

**UNIVERSIDADE FEDERAL DO PARÁ  
NÚCLEO DE INOVAÇÃO E TECNOLOGIAS  
APLICADAS A ENSINO E EXTENSÃO  
PROGRAMA DE PÓS-GRADUAÇÃO CRIATIVIDADE  
E INOVAÇÃO EM METODOLOGIAS DE ENSINO SUPERIOR**

**JOSÉ AUGUSTO DE SENA QUARESMA**

**UM FRAMEWORK GAMIFICADO PARA A DISCIPLINA  
ALGORITMOS OU EQUIVALENTE**



**BELÉM  
2019**



**JOSÉ AUGUSTO DE SENA QUARESMA**

**UM FRAMEWORK GAMIFICADO PARA A DISCIPLINA  
ALGORITMOS OU EQUIVALENTE**

Dissertação apresentada ao Programa de Pós-Graduação Criatividade e Inovação em Metodologias de Ensino Superior do Núcleo de Inovação em Tecnologias Aplicadas a Ensino e Extensão da Universidade Federal do Pará, como requisito parcial à obtenção do título de Mestre em Ensino.

Área de Concentração: Metodologias de Ensino-Aprendizagem.  
Linha de Pesquisa: Criatividade e Inovação em Processos e Produtos Educacionais (CIPPE).

Orientadora: Prof. Dra. Marianne Kogut Eliasquevici  
Coorientador: Prof. Dr. Sandro Ronaldo Bezerra Oliveira

**BELÉM  
2019**

O46a QUARESMA, José Augusto de Sena

Um Framework gamificado para a Disciplina Algoritmos ou equivalente /José Augusto de Sena Quaresma. Belém: UFPA, 2019.

f.250

Dissertação (Mestrado Profissional em Ensino) – Universidade Federal do Pará. Belém, 2019.

1. Algoritmos 2. Gamificação 3. Processo de ensino e aprendizagem.

I. Um Framework gamificado para a disciplina Algoritmos ou equivalente.

C.D.D. 371.102

**JOSÉ AUGUSTO DE SENA QUARESMA**

**UM FRAMEWORK GAMIFICADO PARA A DISCIPLINA  
ALGORITMOS OU EQUIVALENTE**

Dissertação apresentada ao Programa de Pós-Graduação Criatividade e Inovação em Metodologias de Ensino Superior do Núcleo de Inovação em Tecnologias Aplicadas a Ensino e Extensão da Universidade Federal do Pará, como parte das exigências para obtenção do grau de Mestre em Ensino.  
Área de concentração: Metodologias de Ensino-Aprendizagem.

Orientadora: Prof. Dra. Marianne Kogut Eliasquevici  
Coorientador: Prof. Dr. Sandro Ronaldo Bezerra Oliveira

RESULTADO: ( ) APROVADO ( ) REPROVADO

Data: Belém, \_\_\_/\_\_\_/\_\_\_\_\_.

---

Prof. Dra. Marianne Kogut Eliasquevici - Orientadora  
Programa de Pós-Graduação Criatividade e  
Inovação em Metodologias de Ensino Superior  
Universidade Federal do Pará

---

Prof. Dr. Sandro Ronaldo Bezerra Oliveira - Coorientador  
Programa de Pós-Graduação em Ciência da Computação  
Universidade Federal do Pará

---

Prof. Dra. Silvana Rossy de Brito – Examinador externo  
Universidade Federal Rural da Amazônia

---

Prof. Dra. Marcelle Pereira Mota – Examinador externo  
Programa de Pós-Graduação em Ciência da Computação  
Universidade Federal do Pará

---

Prof. Dra. Netília Silva dos Anjos Seixas - Examinador interno  
Programa de Pós-Graduação Criatividade e  
Inovação em Metodologias de Ensino Superior  
Universidade Federal do Pará

**BELÉM  
2019**

*Dedico totalmente à Maria Santíssima,  
Nossa Senhora de Fátima! Ao seu  
auxílio que nunca me faltou...*

## AGRADECIMENTOS

À Deus, pela sabedoria e forças concedidas, as quais me fizeram concluir mais uma fase da minha vida acadêmica, e à Nossa Senhora de Fátima, pela intercessão e exemplo.

Aos meus pais, José Domingos e Ana Celina, pela dedicação, esforço e motivação para sempre buscar meus sonhos.

Aos meus amigos e colegas de curso do PPGCIMES que me acompanharam nessa jornada acadêmica.

Aos meus orientadores, professora Marianne Kogut Eliasquevici e professor Sandro Ronaldo Bezerra Oliveira, por toda a ajuda, incentivo, parceria e amizade nesse tempo de orientação.

À Universidade Federal do Pará (UFPA), por me proporcionar o ambiente acadêmico necessário para alcançar mais uma etapa de minha formação intelectual.

Ao Núcleo de Inovação em Tecnologias aplicadas a Ensino e Extensão (NITAE<sup>2</sup>), por me proporcionar o espaço necessário para alcançar mais esse grau de conhecimento, bem como a disponibilização da ferramenta LimeSurvey para a construção da pesquisa.

Ao Programa de Pós-Graduação Criatividade e Inovação em Metodologias de Ensino Superior (PPGCIMES) e a equipe de professores, pela contribuição a minha formação.

Ao Projeto Software Process Improvement: Development and Research (SPIDER), da UFPA, por me proporcionar um espaço propício para realizar as pesquisas necessárias para a realização desse trabalho.

Ao Grupo de Oração Universitário Maranató, por me permitir a experiência da cultura de Pentecoste conciliando a fé e razão no ambiente acadêmico.

Aos amigos e companheiros do Ministério Universidades Renovadas, por me proporcionarem uma vivência fraterna no tempo de pós-graduação e para a vida inteira.

À minha noiva, Milena Santos, por toda a motivação para finalizar esse trabalho e por se fazer presente nas necessidades.

A todos aqueles que acreditaram no meu potencial durante essa jornada, o meu muito obrigado.

*“Enquanto houver vontade de  
lutar haverá esperança de  
vencer”*

*Santo Agostinho*

## RESUMO

Um *framework gamificado* é a nossa proposta de produto para o processo de ensino e aprendizagem da disciplina Algoritmos ou equivalente. O componente curricular em questão é fundamental para os cursos na área de tecnologia da informação, pois fundamenta o estudante e futuro profissional aos conhecimentos de desenvolvimento de software. Entretanto, verificamos um nível elevado de reprovação e evasão dos alunos nesta disciplina, por meio de consultas na literatura e na Faculdade de Computação da Universidade Federal do Pará (UFPA). Partindo deste contexto, o estudo propõe a seguinte questão-foco: *Como usar a gamificação como abordagem para o processo de ensino e aprendizagem de conteúdos estudados na disciplina Algoritmos ou equivalente nos cursos de graduação da área de Tecnologias de Informação?* Ancorado na questão-foco, tem-se como objetivo geral desenvolver um *framework gamificado* para apoiar o ensino e a aprendizagem dos conteúdos estudados na disciplina Algoritmos ou equivalente presente nos cursos de graduação da área de Tecnologia da Informação. Como forma de alcançar os objetivos traçados, foi definida uma metodologia mista de pesquisa abrangendo: revisão sistemática da literatura; *survey*; observação participante; revisão por pares; e validação por meio da comparação das notas adquiridas nos processos *gamificado* e o tradicionalmente adotado na disciplina. Os dados provenientes da pesquisa permitiram contemplar a literatura sobre a temática, a percepção do panorama brasileiro sobre a disciplina, na visão docente e discente e a observação da interação professor, aluno, monitor e conteúdo na realidade da UFPA. Por meio dos resultados alcançados, foi planejado, concebido e testado o produto dessa dissertação de mestrado com o conteúdo de estrutura de dados homogêneos, apontado para a realidade de aplicação como o de mais difícil aprendido por parte dos alunos e ensino por parte dos professores. Logo, por meio do experimento, corroborado pela Revisão Sistemática da Literatura, ao comparar a abordagem *gamificada*, ao que vem sendo empregado na disciplina Algoritmos, dos cursos da Faculdade de Computação da UFPA, identificamos: (i) maior participação dos alunos em sala; (ii) maior colaboração entre os alunos durante a resolução dos desafios; (iii) maior engajamento para resolução das listas de exercícios; (iv) melhora no desempenho dos alunos no que diz respeito a nota avaliativa; (v) emprego de práticas dinâmicas e lúdicas em sala de aula; (vi) feedback em tempo hábil do progresso dos alunos na disciplina; (vii) aceitação da abordagem por parte dos alunos; e (viii) necessidade de discutir os limites da *gamificação* diante da ocorrência de alunos com deficiência ou de temperamento introvertido. Com esses resultados, avaliamos como oportuno e importante, *gamificar* o processo de ensino e aprendizagem de algoritmos como forma de auxiliar a minimizar os índices de reprovação e evasão na disciplina Algoritmos ou equivalente.

**Palavras-chave:** Algoritmos. Gamificação. Processo de ensino e aprendizagem.



## ABSTRACT

A framework gamified is our product proposal for the teaching and learning process of the Algorithms or equivalent discipline. The curricular component in question is fundamental for the courses in the area of information technology, since it bases the student and professional future on the knowledge of software development. However, we observed a high level of student disapproval and avoidance in this discipline, through consultations in the literature and at the Faculty of Computer Science of the Federal University of Pará (UFPA). From this context, the study proposes the following question-focus: How to use a strategy to get the process of teaching and learning of studied content Algorithms or systems of information undergraduation? Anchored in the question-focus has as general objective the conception of a framework gamified for the discipline Algorithms or equivalent. As a way to reach the objectives outlined, a mixed research methodology was defined, covering systematic review of the literature; survey; participant observation; peer review; and validation by means of the comparison of the acquired grades in the gamified processes and the traditionally adopted in the discipline. The data from the research allowed to contemplate the literature on the subject, the perception of the Brazilian panorama about the discipline, the teacher and student view and the observation of the teacher, student, monitor and content interaction in the UFPA reality. Through the results achieved, the product of this master's thesis with the homogeneous data structure contents was designed, designed and tested, pointing to the reality of application as the most difficult to learn by the students and teaching by the teachers. Therefore, through the experiment, corroborated by the Systematic Review of Literature, when comparing the gamified approach, to what has been used in the Algorithms of the courses of the Faculty of Computation of UFPA, we identified: (i) greater participation of students in the classroom; (ii) greater collaboration among students during the resolution of the challenges; (iii) greater engagement in resolving the lists of exercises; (iv) improvement in students' performance regarding the assessment grade; (v) use of dynamic and playful classroom practices; (vi) timely feedback on students' progress in the discipline; (vii) acceptance of the approach by students; (viii) the need to discuss the limits of gamification in the face of the occurrence of students with disabilities or of introverted temperament. With these results, we evaluated how timely and important it is to spread the teaching and learning process of the Algorithms or equivalent discipline, to help minimize the failure and avoidance rates in the discipline.

**Keywords:** Algorithms. Gamification. Teaching and learning process.

## LISTA DE QUADROS

Quadro 1 – Matriz de criatividade do <i>framework gamificado</i> para a disciplina Algoritmos ou equivalente.....	23
Quadro 2 – Critério de inclusão e exclusão de participantes do <i>survey</i> .....	42
Quadro 3 – Dimensões dos instrumentos e referenciais teóricos de base .....	43
Quadro 4 – Categorias dos questionários .....	43
Quadro 5 – Questões exclusivas para alunos.....	44
Quadro 6 – Requisitos do <i>framework</i> referente às características.....	71
Quadro 7 – Requisitos do <i>framework</i> referentes às práticas .....	72
Quadro 8 – Requisitos do <i>framework</i> referente às recompensas.....	73
Quadro 9 – Requisitos do <i>framework</i> referentes aos recursos e estratégias de ensino.....	73
Quadro 10 – Descrição de papéis .....	75
Quadro 11 – Quesitos de bonificação por comportamento .....	78
Quadro 12 – Quesitos de penalidade por comportamento.....	79
Quadro 13 – Faixa de bonificação para concessão de recurso estrela.....	80
Quadro 14 – Critério, explicação e pontuação atribuída a prática do Dojo do tipo <i>Randori</i> ...	82
Quadro 15 – Critério, explicação e pontuação para as práticas de LAB e teste teórico.....	83
Quadro 16 – Atribuição do <i>avatar</i> aos alunos por prática em sala de aula.....	85
Quadro 17 – Síntese dos dados da revisão por pares.....	88
Quadro 18 – Matriz SWOT adaptada: ATP primeiro dia.....	97
Quadro 19 – Matriz SWOT adaptada: ATP segundo dia .....	98
Quadro 20 – Matriz SWOT adaptada: Dojo <i>Randori</i> primeira implementação.....	99
Quadro 21 – Matriz SWOT adaptada: Dojo <i>Randori</i> segundo dia .....	101
Quadro 22 – Matriz SWOT adaptada: Prática LAB.....	103
Quadro 23 – Fontes Automáticas .....	135
Quadro 24 – <i>String</i> de busca .....	137
Quadro 25 – Critérios de seleção dos estudos primários.....	138
Quadro 26 – Questões para avaliação da qualidade dos estudos.....	139
Quadro 27 – Escala Likert-5 para avaliação da qualidade de estudos.....	139
Quadro 28 – Exemplo de avaliação para os estudos .....	140
Quadro 29 – Avaliação da Qualidade.....	142
Quadro 30 – Abordagens adotadas para <i>gamificar</i> a disciplina Algoritmos ou equivalente .	146

Quadro 31 – <i>Frameworks</i> adotados como base para o processo de <i>gamificação</i> e os ids dos artigos .....	148
Quadro 32 – Elementos de <i>games</i> adotados para <i>gamificar</i> o processo de ensino-aprendizagem de algoritmos .....	149
Quadro 33 – Métodos de Validação adotados nos artigos inclusos .....	152
Quadro 34 – Rastreabilidade dos novos requisitos do <i>framework</i> .....	154
Quadro 35 – Exemplo de uma questão trabalhada na Aula Teórica e Prática antes e depois da mudança.....	155
Quadro 36 – Informações sobre a seção de medalhas da <i>gamificação</i> e premiação no plano de ensino.....	155
Quadro 37 – Exemplo de uma questão trabalhada na prática Dojo <i>Randori</i> antes e depois da mudança.....	156

## LISTA DE FIGURAS

Figura 1 – Histórico de reprovações e aprovações na disciplina Algoritmos na Faculdade de Computação da UFPA (2010-2017) .....	22
Figura 2 – Metodologia da pesquisa.....	26
Figura 3 – <i>Framework Octalysis</i> .....	33
Figura 4 – Pirâmide DMC .....	35
Figura 5 – <i>Mechanics, Dynamics and Emotions</i> (MDE).....	36
Figura 6 – Pesquisa pelo termo <i>gamificação</i> ao longo dos anos .....	42
Figura 7 – Participação na disciplina: (a) quantidade de participação dos discentes;.....	46
Figura 8 – Auto perfil identificado pelos discentes .....	46
Figura 9 – Abordagem didática mais significativa.....	47
Figura 10 – Recursos utilizados para apoiar o ensino .....	48
Figura 11 – Recursos significativos para ensino e aprendizagem .....	48
Figura 12 – Escolha da forma como são divididos os discentes em sala de aula.....	49
Figura 13 – Formato da divisão dos alunos .....	50
Figura 14 – Formas de resolução dos algoritmos em sala pelo professor .....	50
Figura 15 – Uso de linguagens de programação na disciplina .....	51
Figura 16 – Participação e apoio do monitor na disciplina .....	52
Figura 17 – Uso de bonificação na disciplina .....	53
Figura 18 – Formas de bonificação empregadas pelos docentes.....	53
Figura 19 – Formas de bonificação dentro da disciplina identificadas pelos discentes .....	54
Figura 20 – Motivação durante o uso da estratégia de bonificação.....	54
Figura 21 – <i>Feedback</i> das atividades da disciplina .....	55
Figura 22 – Pré-requisitos necessários para o aprendizado de algoritmos .....	56
Figura 23 – Nível de facilidade/dificuldade para ministrar os conteúdos pelo professor .....	57
Figura 24 – Nível de facilidade/dificuldade de assimilar os conteúdos pela turma na visão do professor .....	58
Figura 25 – Grau de facilidade/dificuldade de assimilar os conteúdos na visão discente.....	59
Figura 26 – Nota dos alunos da primeira observação.....	67
Figura 27 – Nota dos alunos coletadas da segunda observação .....	67
Figura 28 – Visualização exemplo de bonificação por comportamento .....	79
Figura 29 – Visualização exemplo de penalidades por comportamento .....	80
Figura 30 – Visualização de parte da aba responsável pela geração de estrela.....	81

Figura 31 – Visualização da aba lista de exercício para as aulas teóricas e práticas da planilha <i>gamificada</i> .....	81
Figura 32 – Visualização da aba de pontuação aula Dojo do tipo <i>Randori</i> .....	82
Figura 33 – Visualização da pontuação por questão da aula LAB e prova teórica .....	83
Figura 34 – Visualização de parte da aba de pontuação da aula LAB .....	84
Figura 35 – Visualização da aba score. ....	84
Figura 36 – Visualização exemplo de atribuição de Medalhas .....	85
Figura 37 – Visualização das abas de conversão das pontuações e estrelas em notas .....	86
Figura 38 – Exemplo de visualização do <i>ranking</i> .....	91
Figura 39 – Presença dos discentes durante as aulas.....	105
Figura 40 – Participações dos discentes durante as aulas.....	105
Figura 41 – Sugestões dos discentes em sala de aula .....	106
Figura 42 – Colaborações dos discentes durante as aulas .....	107
Figura 43 – Faltas dos discentes no decorrer das aulas .....	107
Figura 44 – Penalidade “atraso em 10 minutos” no decorrer das aulas .....	108
Figura 45 – Uso do celular durante as aulas .....	108
Figura 46 – Uso do computador para outras atividades .....	109
Figura 47 – Penalidade por atrapalhar a aula .....	110
Figura 48 – <i>Scoreplay</i> da <i>gamificação</i> .....	111
Figura 49 – Resultados da Aula Teórica e Prática primeiro dia.....	111
Figura 50 – Resultados da Aula Teórica e Prática segundo dia .....	112
Figura 51 – Resultados da prática Dojo <i>Randori</i> .....	113
Figura 52 – Resultados da Lista de Exercício para casa.....	113
Figura 53 – Resultados da prática Laboratório de Programação.....	114
Figura 54 – Resultados LAB questões individuais.....	115
Figura 55 – Resultados do Teste teórico .....	115
Figura 56 – Resultados das questões individuais Teste Teórico .....	116
Figura 57 – Ranking dos discentes .....	117
Figura 58 - Aquisição recurso estrela - Aula Teórica e Prática primeiro dia .....	118
Figura 59 – Resultados recurso estrela - Aula Teórica e Prática segundo dia.....	118
Figura 60 – Resultados recurso estrela - Dojo <i>Randori</i> primeiro dia.....	119
Figura 61 – Resultados recurso estrela - Dojo <i>Randori</i> segundo dia .....	119
Figura 62 – Resultado recurso estrela aula de <i>feedback</i> .....	120
Figura 63 – Quantitativo total de estrela por discente .....	120

Figura 64 – Quantidade de estrelas atribuídas no decorrer do processo .....	121
Figura 65 – Medalha <i>Power</i> atribuída .....	122
Figura 66 – Medalha Participativo .....	122
Figura 67 – Notas dos discentes no uso do <i>framework gamificado</i> .....	126
Figura 68 – Apresentação do desempenho do discente na disciplina inteira .....	127
Figura 69 – Notas disciplina Algoritmos 2016.....	128
Figura 70 – Distribuição dos trabalhos em relação aos anos.....	143
Figura 71 – Distribuição por tipo de publicação .....	144
Figura 72 – Uso de ferramentas para embasar a <i>gamificação</i> .....	150
Figura 73 – Informação sobre o sucesso no uso de <i>gamificação</i> como base para o processo de ensino-aprendizagem de Algoritmos .....	151
Figura 74 – O uso de validação e/ou avaliação nos estudos incluídos .....	151

## **LISTA DE TABELAS**

Tabela 1 – Seleção dos estudos primários .....	143
Tabela 2 – Qualidade dos estudos primários .....	145

## SUMÁRIO

INTRODUÇÃO.....	177
JUSTIFICATIVA E RELEVÂNCIA .....	200
MOTIVAÇÃO.....	244
METODOLOGIA.....	255
ESTRUTURA DA DISSERTAÇÃO .....	277
1    FUNDAMENTAÇÃO TEÓRICA.....	28
1.1    ENSINO E APRENDIZAGEM DE ALGORITMOS .....	28
1.1.1    Como ensinar Algoritmos .....	28
1.1.2    As dificuldades ao ensinar algoritmos .....	29
1.2    GAMIFICAÇÃO .....	311
1.2.1 <i>Framework</i> de <i>gamificação</i> .....	322
1.2.2    Os elementos de jogos .....	377
1.3    TRABALHOS RELACIONADOS .....	38
2    CONCEPÇÃO DO FRAMEWORK GAMIFICADO .....	400
2.1 <i>SURVEY</i> .....	400
2.1.1 <i>Design do Survey</i> .....	411
2.1.2 <i>Instrumentação do survey</i> .....	422
2.1.3    Resultados e discussão do <i>survey</i> .....	455
2.1.4    Considerações sobre o <i>survey</i> .....	59
2.2    OBSERVAÇÃO PARTICIPANTE.....	600
2.2.1    Estrutura da disciplina .....	600
2.2.2    Apresentando o estudo.....	600
2.2.3    A experiência vivida .....	611
2.2.4    Resultados e discussão.....	622
2.2.5    Considerações sobre a observação participante.....	68
2.3    FRAMEWORK PROJETO SPIDER.....	69
3    FRAMEWORK GAMIFICADO VERSÃO 01 .....	711
3.1    REQUISITOS .....	711
3.2    PLANO DE ENSINO .....	744
3.2.1    Características da <i>Gamificação</i> .....	744
3.2.2    Práticas.....	766
3.2.3    Recompensas .....	766
3.2.4    Recursos e Estratégias de Ensino.....	777
3.3    PLANILHA GAMIFICADA.....	78
3.3.1    Bonificações e penalidades por comportamento.....	78
3.3.2    Recurso estrela.....	800
3.3.3    Pontuação por práticas desenvolvidas .....	811
3.3.4 <i>Avatar</i> .....	844
3.3.5    Resultados finais da <i>gamificação</i> .....	855
3.4    REVISÃO POR PARES .....	866
3.4.1    Planejamento.....	866
3.4.2    Condução .....	877
3.4.3    Análise .....	877
3.5    MELHORIAS .....	900
4    O USO DO FRAMEWORK GAMIFICADO: O EXPERIMENTO .....	922
4.1    PLANEJAMENTO DO EXPERIMENTO .....	922
4.1.1    Cenário.....	922
4.1.2    Personagens .....	922



4.1.3	Desenvolvimento das aulas.....	933
4.1.4	O <i>design</i> da <i>gamificação</i> .....	944
4.2	AVALIAÇÃO DO EXPERIMENTO.....	954
4.2.1	Discussão e resultados qualitativos.....	954
4.2.2	Discussão e resultados quantitativos.....	1100
4.2.3	Casos particulares .....	1222
4.2.4	Melhorias .....	1255
4.3	VALIDAÇÃO DO FRAMEWORK GAMIFICADO .....	1266
5	CONCEPÇÃO DA SEGUNDA VERSÃO DO FRAMEWORK GAMIFICADO	
	12929	
5.1	REVISÃO SISTEMÁTICA DA LITERATURA.....	12929
5.1.1	Metodologia da Revisão Sistemática.....	12929
5.1.2	Estratégia da Revisão Sistemática .....	1311
5.1.3	Sumarização dos resultados da Revisão Sistemática.....	1433
5.1.4	Ameaças à validade da RSL .....	1522
5.1.5	Conclusões da RSL.....	1533
5.2	REQUISITOS .....	1533
5.3	FRAMEWORK GAMIFICADO VERSÃO 02: O PRODUTO FINAL .....	1544
	CONSIDERAÇÕES FINAIS .....	1577
	REFERÊNCIAS BIBLIOGRÁFICAS .....	1622
	APÊNDICE A – INSTRUMENTO SURVEY DOCENTE .....	16767
	APÊNDICE B – INSTRUMENTO SURVEY DISCENTE.....	1711
	APÊNDICE C – INSTRUMENTO DA OBSERVAÇÃO PARTICIPANTE.....	1766
	APÊNDICE D – PLANO DE ENSINO-APRENDIZAGEM GAMIFICADO.....	178
	APÊNDICE E – FORMULÁRIO DE REVISÃO POR PARES.....	1955
	APÊNDICE F – PROTOCOLO DA REVISÃO SISTEMÁTICA DA LITERATURA.....	2011
	APÊNDICE G – LISTA DE REFERÊNCIAS DOS ARTIGOS SELECIONADOS (EM ORDEM CRONOLÓGICA) .....	2211
	APÊNDICE G – PLANO DE ENSINO-APRENDIZAGEM GAMIFICADO – VERSÃO FINAL DO PRODUTO.....	22727

## INTRODUÇÃO

A disciplina Algoritmos é um componente curricular básico pertencente ao núcleo Fundamentos de Computação para cursos de graduação em Tecnologia da Informação (SOCIEDADE BRASILEIRA DA COMPUTAÇÃO, 2005). Muitos são os nomes atribuídos a esta disciplina, tais como Introdução à Programação, Introdução à Ciência da Computação, Introdução à Programação de Computadores, Lógica de Programação, entre outros. Para fins deste trabalho, consideramos o nome da disciplina como Algoritmos ou equivalente.

O aprendizado de algoritmos, muitas vezes trata-se do primeiro contato do estudante com a programação, o que nem sempre é algo fácil, visto os estudantes precisarem desenvolver competências para solução de problemas. Assim, por algoritmos define-se, “[...] a criação de uma sequência de operações que, quando executada, produz o resultado do problema [...]” (CARVALHO, 1986, p. 13). Quando os estudantes, logo no início do seu percurso acadêmico, já apresentam dificuldades de entendimento das representações algorítmicas e abstrações, podem ser levados à desistência, reprovação, retenção, desmotivação ou falta de estímulo para seguir em frente.

Conforme Gomes (2010, p. 1),

O ensino e aprendizagem de programação constituem um enorme desafio para alunos e professores. Os elevados níveis de insucesso nas disciplinas introdutórias de programação, em qualquer grau e sistema de ensino, em qualquer parte do mundo, são tema de preocupação e alvo de variadas pesquisas, ao longo dos tempos, resultando também em muitas propostas, sem que, contudo tenham sido reportadas melhorias generalizadas.

Logo, a busca por abordagens didáticas e metodológicas que facilitem o ensino de algoritmos por parte do professor e o aprendizado, por parte dos estudantes, é algo necessário que já vem sendo perseguido. Dentre as abordagens possíveis, no decorrer das pesquisas para a realização do trabalho em tela, identificamos o uso e a defesa da *gamificação* na educação que tem como objetivo proporcionar um maior engajamento do estudante, por meio da inserção de elementos de jogos no contexto de sala de aula, tais como pontuação, *ranking*, *scoreplay*, gerência de recursos, missões, narrativas entre outros (VIANNA *et al.*, 2013; GROTZ, 2016, HSU; LIN, 2016; ELGRABLY; OLIVEIRA, 2018; SANTOS; OLIVEIRA, 2018; WERBACH, 2012).

O termo *gamificação* é conceituado como o uso de *design*, elementos e características de jogos em contextos diferentes de jogos (DETERDING *et al.*, 2011). O estudo de McGonigal

(2011) define qualquer jogo por suas características básicas, quais sejam: meta, regras, sistema de *feedback* e participação voluntária. A meta é o resultado específico que os jogadores vão trabalhar. As regras impõem as limitações em como os jogadores devem atingir a meta. O sistema de *feedback* informa aos jogadores a atual situação do jogo e o quão perto estão de conseguir alcançar a meta. Já a participação voluntária, exige que cada um dos jogadores aceite, consciente e voluntariamente, a meta, as regras e o sistema de *feedback*.

O mapeamento sistemático proposto por Diecheva *et al.* (2015) identificou várias pesquisas que empregaram a *gamificação* como estratégia didática e a grande maioria dos autores destacados no mapeamento concordou que a *gamificação* pode melhorar a aprendizagem, se for bem projetada e utilizada corretamente.

Diante do exposto, tendo por base a realidade da oferta da disciplina Algoritmos da Faculdade de Computação, da Universidade Federal do Pará (UFPA), e as inquietações da minha experiência como aluno de um curso de graduação na área de Tecnologia da Informação, este trabalho partiu da seguinte questão-foco:

*Como usar a gamificação como abordagem para o processo de ensino e aprendizagem de conteúdos estudados na disciplina Algoritmos ou equivalente nos cursos de graduação da área de Tecnologias de Informação?*

A pesquisa empreendida para responder à questão-foco teve como objetivo geral:

- Desenvolver um *framework gamificado* para apoiar o ensino e a aprendizagem dos conteúdos estudados na disciplina Algoritmos ou equivalente presente nos cursos de graduação da área de Tecnologia da Informação.

Os objetivos específicos delineados foram:

- Identificar abordagens *gamificadas* para apoiar o processo de ensino e aprendizagem de algoritmos por meio de uma Revisão Sistemática da Literatura (RSL) referente a estratégias que utilizem *gamificação* no contexto dos cursos de graduação na área de TI.
- Identificar, por meio de um *survey* realizado com discentes e docentes envolvidos com a disciplina Algoritmos ou equivalente, a forma como vem sendo trabalhada a disciplina nos cursos superiores da área de TI, referentes ao ensino, à aprendizagem e ao conteúdo.

- Compreender como ocorre a interação entre docente, discentes e o conteúdo durante as aulas da disciplina Algoritmos, ofertada pela Faculdade de Computação, da Universidade Federal do Pará, por meio de observação participante.
- Conceber e implementar um *framework gamificado* para disciplina Algoritmos ou equivalente com base nos requisitos levantados na RSL, *survey* e observação participante.
- Avaliar o *framework* com especialistas, na área de algoritmos e *gamificação*, adotando a técnica de revisão por pares.
- Validar o *framework gamificado* com a sua execução na disciplina Algoritmos ou equivalente, de preferência ofertada pelos cursos de graduação na área de TI.

Para tal, no presente trabalho apresentamos como concebemos, desenvolvemos, testamos e validamos um *framework gamificado* para a disciplina Algoritmos ou equivalente. Por *framework*, entendemos como um grupo de práticas executadas em sequência para alcançar determinado objetivo; já *gamificado* são os elementos de jogos realizados em ordem, com a finalidade de gerar o processo de ensino e aprendizagem. O *framework* proposto como produto educacional contempla um plano de ensino *gamificado* e uma planilha eletrônica de coleta e geração de visualização para os estudantes, tal qual resumimos a seguir:

- Plano de ensino *gamificado*: descreve o processo de *gamificação* da disciplina, detalhando as abordagens trabalhadas em sala de aula. Além disso, explicita o cronograma de atividades, as regras, as formas de pontuação, a bonificação, as penalidades e a visualização que será disponibilizada aos estudantes. O plano gerencia os recursos utilizados em sala de aula e explica a conversão da *gamificação* para compor a nota na disciplina.
- Planilha eletrônica: descrita no plano de ensino, calcula automaticamente os elementos *gamificados*, tais como pontuações das atividades, *ranking*, recursos, *avatar* e bonificações de comportamento.

Embora esta seja uma disciplina também presente nos cursos técnicos da área de Tecnologia da Informação (TI), compreendemos que, dado o escopo do Programa de Pós-Graduação Criatividade e Inovação em Metodologias do Ensino Superior (PPGCIMES), mais especificamente a linha de pesquisa Criatividade e Inovação em Processos e Produtos Educacionais (CIPPE), focamos o desenvolvimento de um produto visando cursos de graduação. De acordo com o sítio do PPGCIMES, a linha CIPPE está orientada à

[...] conceber e desenvolver processos e produtos criativos para o ensino-aprendizagem, configurados a partir de demandas das diferentes áreas do conhecimento. Para tanto, os recursos comunicacionais, educacionais e os das Tecnologias da Informação e Comunicação serão explorados na geração de soluções inovadoras e factíveis. Os processos e produtos desenvolvidos serão acompanhados de pesquisas qualitativas e/ou quantitativas orientadas às testagens e validações das soluções (PPGCIMES, 2018).

Logo, as tecnologias de informação e comunicação (TICs) se fizeram presentes no produto implementado por meio das planilhas eletrônicas alimentadas pelo pesquisador no percurso da disciplina. Ao final de cada aula, gráficos de *feedback* foram gerados, como é o caso do *ranking* da turma em relação à pontuação e o individual de cada estudante.

O público alvo do *framework* é composto por discentes da graduação que tenham a disciplina Algoritmos ou equivalente em seu currículo base, de preferência da área de Tecnologia da Informação, bem como, docentes de instituições de ensino superior, habilitados a ministrar este componente curricular.

## JUSTIFICATIVA E RELEVÂNCIA

O processo de ensino e aprendizagem de algoritmos está presente na construção básica dos cursos da área de TI e é um eixo fundamental da formação dos futuros profissionais, pois agrega a eles a capacidade de desenvolver e analisar a lógica de sistemas de informação computacionais.

No documento referente à formação dos cursos superiores em Computação (SBC, 2017), é possível visualizar que para o profissional adquirir competências em alguns eixos de formação, como exemplo o desenvolvimento de sistemas e resolução de problemas, precisa-se do conteúdo da disciplina Algoritmos ou equivalente desde a fase inicial do curso. Logo, é possível inferirmos que para o bom seguimento em disciplinas futuras, o conhecimento sobre como se estrutura os algoritmos e as competências adquiridas para alcançar este conhecimento, se fazem necessários.

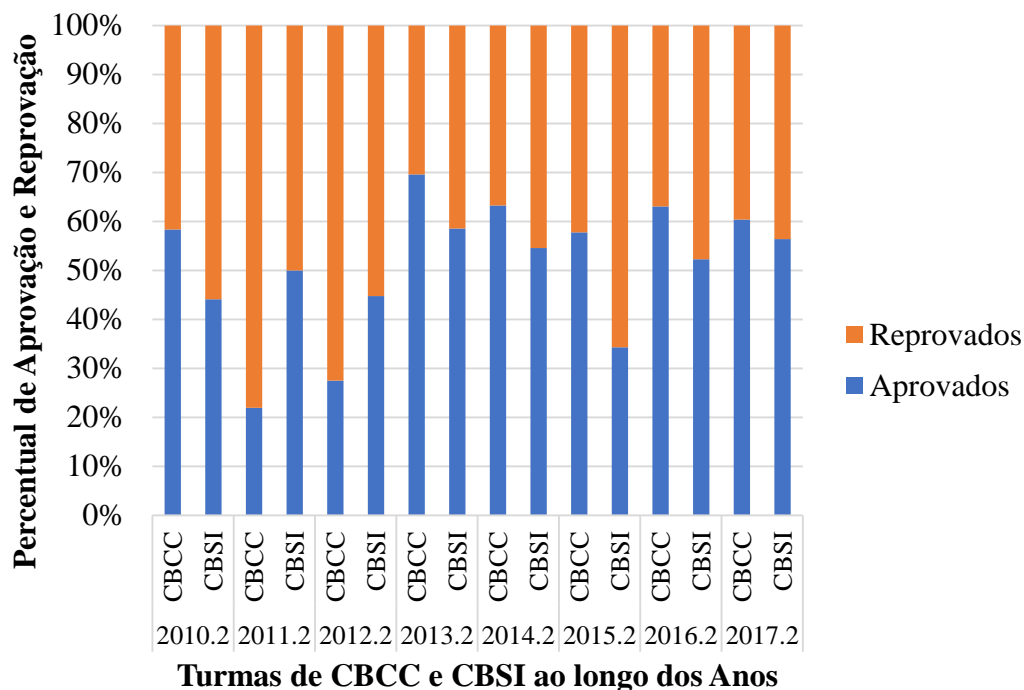
Em geral, a disciplina é apresentada aos estudantes nos semestres iniciais do curso e, conseqüentemente, estabelece a primeira interação dos discentes com o desenvolvimento de *software*. O objetivo é desenvolver no aluno a capacidade de elaborar soluções lógicas mediante problemas genéricos em uma sequência de passos (DIAS JUNIOR; MERCADO, 2016). Segundo o documento referente à formação dos cursos superiores em Computação (SOCIEDADE BRASILEIRA DA COMPUTAÇÃO, 2017), tais competências e habilidades são bases para a formação dos futuros profissionais na área de TI, com especial ênfase no eixo de desenvolvimento de *software*.

Não existe uma única forma como a disciplina pode ser abordada. Dentre as técnicas para o ensino mais comuns estão: (i) uso de problemas com enunciados textuais, os quais o professor busca trabalhar a destreza do aluno em entender o enunciado para em seguida elaborar uma solução algorítmica em pseudocódigo ou em fluxogramas; (ii) o aluno realizar um “teste de mesa” para simular passo a passo a execução do algoritmo e verificar se está alcançando o objetivo proposto; (iii) pedir aos alunos que passem para o papel o que raciocinaram mentalmente; e (iv) o professor acompanhar os alunos quando estes desenvolverem seus exercícios individualmente, entre várias outras. Algumas podem ser mais interessantes do que outras, mas o que se observa é que ainda persistem dificuldades de aprendizagem pelos estudantes (DIAS JUNIOR; MERCADO, 2016; BOSSE; GEROSA, 2015; MARCUSSI *et al.*, 2016).

As dificuldades presentes na disciplina remetem a índices altos de reprovação (BOSSE; GEROSA, 2015; MARCUSSI *et al.*, 2016) e evasão (GIRAFFA; MORA, 2016), que podem ocorrer por diversos motivos, desde o pouco conhecimento prévio de lógica de programação, entendimento dos problemas propostos e dos conteúdos na disciplina, ou até a própria falta de tempo para estudar.

O panorama observado na literatura sobre reprovações e evasão na disciplina também foi perceptível dentro do contexto de oferta da disciplina nos cursos de Bacharelado em Ciência da Computação (CBCC) e Bacharelado em Sistemas de Informação (CBSI) da Faculdade de Computação, da Universidade Federal do Pará (UFPA), durante os anos de 2010 a 2017, conforme expresso na Figura 1.

Figura 1 – Histórico de reprovações e aprovações na disciplina Algoritmos na Faculdade de Computação da UFPA (2010-2017)



Fonte: Dados oriundos do SIGAA UFPA (2018).

É visível os índices de reprovações, tanto no CBCC, com 46%, quanto no CBSI, com 50%. Embora os motivos de reprovação sejam distintos, a maioria dos casos está relacionada à não obtenção do conceito mínimo necessário, à frequência ou à evasão da disciplina, conforme conversas informais com professores da matéria e observação participante realizada em duas ofertas da mesma durante a pesquisa.

Outro aspecto importante que colaborou para o desenvolvimento de um produto para a disciplina Algoritmos ou equivalente, foi a análise das observações participantes, efetuadas por mim, durante Estágio Supervisionado, em duas ofertas da disciplina Algoritmos, do CBCC, da UFPA. Foi possível observar a interação entre professor e alunos, a forma como o assunto foi abordado em sala e as estratégias utilizadas para reter a atenção dos discentes, demonstrando ser viável e relevante a utilização de uma abordagem *gamificada* para o ensino e aprendizagem dos conteúdos presentes na disciplina em questão.

Em relação à *gamificação*, percebemos em Schlemmer (2016) a defesa do seu uso por proporcionar ao aluno um aprendizado fluido, por meio do sentimento de confiança e autoestima na sala de aula. No estudo de Li *et al.* (2012) observamos que o uso de materiais *gamificados* para o ensino aumenta o engajamento subjetivo, proporciona níveis de prazer para o usuário e contribui para o processo de criação do conhecimento, pois aprimora a aprendizagem do aluno.

Quanto ao caráter inovador do produto desenvolvido, nos apropriamos do conceito de Fainholc (2009, p. 241, tradução livre) de que “o processo inovador é definido como um sistema ou projeto destinado a melhorar e mudar as práticas, a razão pela qual está aberto a inter-relação com outros sistemas”. A mesma autora aponta que

[...] a inovação se refere tanto aos processos quanto às mudanças de ideias, materiais ou práticas consolidadas. [...] É frequente [...] a alusão à mudança de modelos, estratégias e recursos didáticos ou práticas. Trata-se de introduzir na realidade do currículo novas dinâmicas que alteram as ideias, metas, papéis, conteúdos, metodologia, organização espacial ou temporal, recursos de avaliação. É o que ocorre com a articulação das TICs ao currículo (FAINHOLC, 2009, p. 242, tradução livre)<sup>1</sup>.

A partir da afirmação de Fainholc (2009), é possível concluir que a abordagem *gamificada* para o ensino de algoritmos, da maneira como foi concebida, permite propiciar uma modificação de práticas já estabelecidas, trazendo inovação ao processo de ensino e aprendizagem de algoritmos.

Com relação ao aspecto criativo, sabemos que não há uma única forma de definir criatividade, assim, como, tampouco, não existe um consenso do que seja um produto criativo. Houve tentativas de sistematizar algumas dimensões e critérios para qualificar um produto como criativo, tais quais as propostas por Besemer e O’Quin (2010), que nos ajudaram a balizar e compreender a criatividade presente na proposta (Quadro 1).

Quadro 1 – Matriz de criatividade do *framework gamificado* para a disciplina Algoritmos ou equivalente (Continua...)

Dimensão novidade: presença de características que traduzem modos diferentes de apresentação do produto em relação aos existentes		
Critério	Descrição	Explicação em relação ao produto
Originalidade	O produto não é usual ou comum quando considerado em contexto que envolva produtos desenvolvidos por pessoas com treinamento e experiências similares (BESEMER ; TREFFINGER, 1981)	Podemos identificar o caráter original da proposta, por estar sendo adotada uma tabela para gerar os dados <i>gamificados</i> dentro da disciplina Algoritmos, algo que ainda não foi feito no contexto em que a <i>gamificação</i> será testada.

Fonte: Elaboração própria (2019).

1 [...] la innovación se refiere tanto a los procesos como a los cambios de ideas, materiales o prácticas consolidadas. [...] Es frecuente [...] la alusión al cambio de modelos, estrategias y recursos didáticos o prácticas. Se trata de introducir en la realidad del currículo nuevas dinámicas que alteran las ideas, metas, papeles, contenidos, metodología, organización espacial o temporal, recursos de evaluación. Es lo que ocurre con la articulación de las TIC al currículo.



Quadro 1 - Matriz de criatividade do *framework gamificado* para a disciplina Algoritmos ou equivalente (Conclusão)

Dimensão novidade: presença de características que traduzem modos diferentes de apresentação do produto em relação aos existentes		
Critério	Descrição	Explicação em relação ao produto
Transformacional	A maneira pela qual o produto força uma nova maneira de ver o mundo. (BESEMER; TREFFINGER, 1981)	A proposta do produto pretende transformar uma realidade do ponto de vista do ensino e aprendizado de algoritmos na UFPA.
Dimensão resolução de problemas: inclui critérios que dizem respeito à exatidão ou correção da solução para a situação problemática		
Critério	Descrição	Explicação em relação ao produto
Útil	Identifica se um produto é operável pelos seus usuários (BESEMER; TREFFINGER, 1981).	A proposta pretende alterar um quadro histórico de reprovações e evasão.
Lógico	Este critério é definido como a medida com que o produto é cientificamente "verdadeiro" ou "válido", ao passo que a solução é consistente com os fatos, ou a extensão em que o produto segue as "regras do jogo" (BESEMER; TREFFINGER, 1981).	O produto será validado cientificamente, pois foi desenvolvido sobre referências bibliográficas e uma série de métodos e metodologias amplamente aceitas pela comunidade científica.
Dimensão elaboração e síntese: envolve considerações sobre estilo.		
Critério	Descrição	Explicação em relação ao produto
Elegância	Uma marca do produto extremamente criativo. O critério pode ser definido como à medida que o produto fornece uma solução discreta ou econômica (BESEMER; TREFFINGER, 1981).	O produto irá gerar gráficos de <i>feedback</i> bem trabalhados por meio de uma planilha eletrônica, contendo os elementos de jogos indicados no plano de ensino.

Fonte: Elaboração própria (2019).

Portanto, segundo a análise realizada por meio da matriz de criatividade (vide Quadro 1) podemos visualizar a criatividade presente na proposta. Outra realidade presente é a inovação por tratar de uma modificação de prática já existente.

## MOTIVAÇÃO

Ao entrar como discente no Programa Pós-Graduação Criatividade e Inovação em Metodologias de Ensino Superior, em um primeiro momento, fiquei preocupado por não ter experiência com a docência, tal qual outros colegas de turma, e, desta forma, não ter uma vivência na qual o meu produto poderia emergir, visto que havia a necessidade de mudar a proposta inicial apresentada no momento da seleção. Após muita conversa com meus

orientadores, fui instigado, então, a recordar das dificuldades que senti ou vivenciei como aluno de graduação da área de TI, as quais poderiam servir de motivação para o desenvolvimento do meu produto. Foi quando me recordei da disciplina Algoritmos, conforme relato a seguir.

Fui discente dos cursos de Bacharelado em Sistemas de Informação (CBSI), da Faculdade de Computação, da Universidade Federal do Pará (UFPA) e Técnico em Informática, do Instituto Federal do Pará (IFPA). Em ambas as ocasiões pude vivenciar a experiência de ser aluno da disciplina Algoritmos ou equivalente, o que me permitiu perceber o quanto alguns alunos tinham dificuldades em compreender os conteúdos, em especial na lógica necessária para a resolução do problema algorítmico. Era comum os alunos com dificuldades sentirem-se desmotivados para aprender e entender como desenvolver um algoritmo, levando-os a não passarem na disciplina. Na minha turma da graduação, também convivi com alunos que já estavam cursando a disciplina pela segunda vez e outros que estavam retidos.

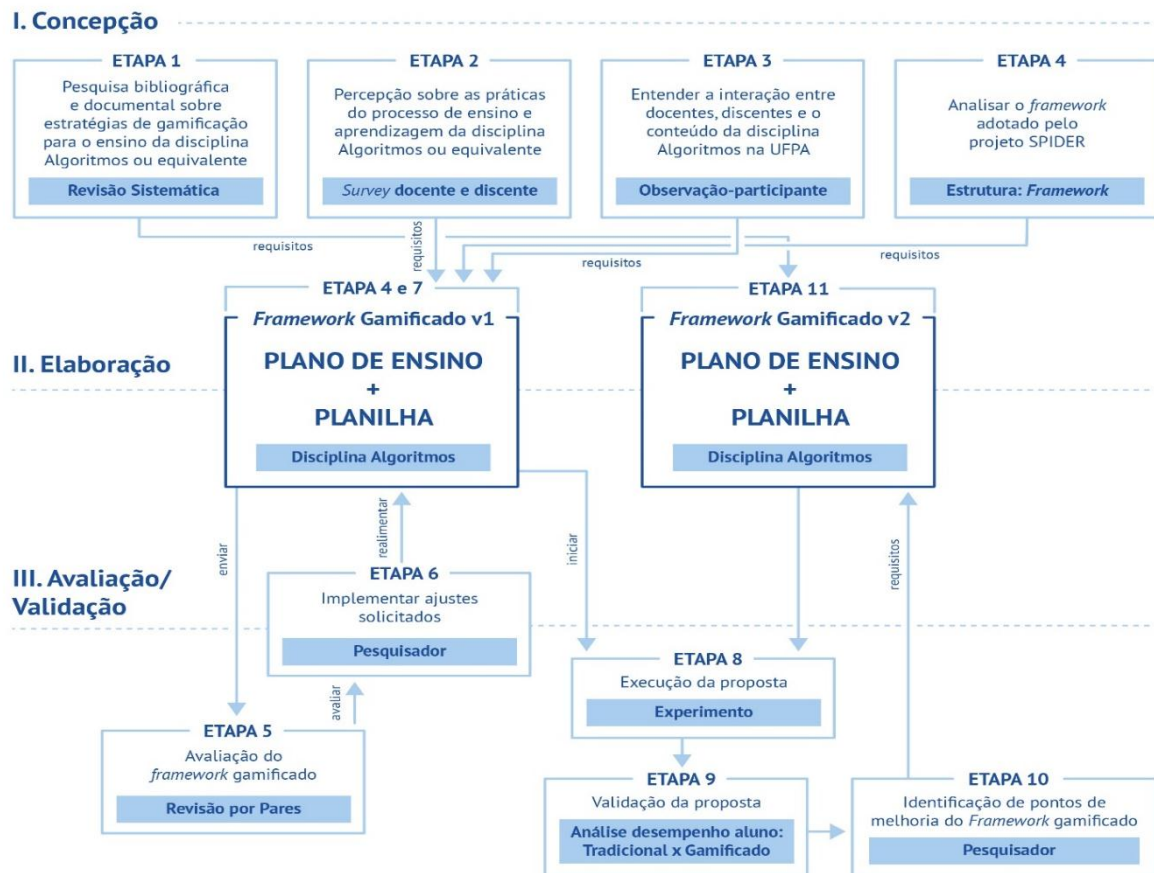
As experiências relatadas, principalmente com a graduação, já seriam suficientes como motivação para pensar em um produto, mas ao ler mais sobre o assunto, pude verificar que o que vivenciei no meu curso não é um caso isolado. Infelizmente, é uma realidade que atinge várias instituições de ensino superior e, por isso, virou alvo de pesquisas ao redor do mundo. Assim, estabeleci como objetivo ajudar a mudar a realidade, a partir da proposta de desenvolver um *framework gamificado* que pudesse atuar diretamente na disciplina Algoritmos ou equivalente.

Tive contato com a abordagem *gamificada*, quando cursei duas disciplinas optativas de Engenharia de *Software*, ofertadas pelo Programa de Pós-Graduação em Ciência da Computação da UFPA, em que as aulas eram nesse método. Ao visualizar como a abordagem gerava engajamento e envolvimento nos alunos participantes, percebi que também poderia ser uma abordagem interessante para a disciplina Algoritmos.

## METODOLOGIA

A metodologia proposta para esse trabalho configura-se como procedimentos de técnicas mistas, uma vez que aglutina aspectos tanto dos métodos quantitativos quanto dos qualitativos (CRESWELL, 2007). Durante a pesquisa inicial para o desenvolvimento desse trabalho foi definido um fluxo metodológico, presente na Figura 2.

Figura 2 – Metodologia da pesquisa



Fonte: Elaboração própria (2019).

Na primeira fase do fluxo metodológico temos as seguintes bases para a concepção do produto: a Revisão Sistemática da Literatura (RSL); o *survey*; a observação-participante; a análise do *framework* adotado pelo Projeto SPIDER (PROJETO SOFTWARE PROCESS IMPROVEMENT: DEVELOPMENT AND RESEARCH - UFPA, 2009) da UFPA para *gamificar* salas de aula no campo da Engenharia de *Software*; e a revisão informal da literatura para aprofundarmos a base teórica e identificarmos os trabalhos relacionados. Vale ressaltar que os procedimentos metodológicos da RSL foram os primeiros a serem iniciados, porém pelo seu tempo de demora os requisitos extraídos foram utilizados para validar a segunda versão do produto.

Após a fase de concepção, temos a fase de elaboração em que tivemos a disponibilização das versões do *framework gamificado* para a disciplina Algoritmos ou equivalente. Na fase de avaliação e validação foi realizada a ponderação por meio da revisão por pares e dos dados coletados pelo experimento. A validação ocorreu de duas formas: (i) pela coleta do progresso do aluno na disciplina; e (ii) pela comparação com uma disciplina de Algoritmos ofertada em outro semestre, em que foi adotado como parâmetro o quesito nota. Cada um dos procedimentos

metodológicos realizados por essa pesquisa e visualizados na Figura 2 está descrito em maiores detalhes nas subseções desta dissertação.

Além disso, foi desenvolvido um experimento com o objetivo de ensinar o conteúdo de estrutura de dados homogêneos em uma sala de aula *gamificada*, em que, ao término, os resultados foram analisados e discutidos. Depois desse momento, os resultados alcançados foram comparados aos adquiridos pela Revisão Sistemática da Literatura e realizada a proposição de implementações para o desenvolvimento da segunda versão do produto da pesquisa de mestrado. O percurso metodológico detalhado está no corpo do trabalho.

## ESTRUTURA DA DISSERTAÇÃO

A dissertação está organizada, tal qual a seguir:

- Capítulo 1: apresenta-se a fundamentação teórica com os trabalhos relacionados ao produto de mestrado, bem como das metodologias utilizadas.
- Capítulo 2: descreve-se o percurso metodológico de concepção do *framework gamificado*.
- Capítulo 3: discorre-se sobre a primeira versão do produto de mestrado, com seus requisitos e a forma como foi avaliada por pares.
- Capítulo 4: relata-se o experimento da primeira versão do *framework gamificado* e o processo de validação realizado.
- Capítulo 5: apresenta-se o resultado da Revisão Sistemática da Literatura, como base para o desenvolvimento da versão final do produto da pesquisa de mestrado.
- Considerações finais: descreve-se os objetivos alcançados, as dificuldades encontradas, entre outras análises importantes, além de deixarmos indicado sugestões para trabalhos futuros.

## 1 FUNDAMENTAÇÃO TEÓRICA

O capítulo apresenta a fundamentação teórica desta dissertação, discutindo os seguintes conceitos: ensino e aprendizagem de algoritmos e *gamificação*. Além disso, ao final, apresenta os trabalhos relacionados.

### 1.1 ENSINO E APRENDIZAGEM DE ALGORITMOS

O processo de ensino e aprendizagem de Algoritmos está presente na construção básica dos cursos da área de TI e é um eixo fundamental da formação dos futuros profissionais, pois, agrega ao profissional a capacidade de desenvolver e analisar a lógica de sistemas de informação computacionais.

No documento referente à formação dos cursos superiores em Computação (SBC, 2017), é possível visualizar que para o profissional adquirir competências em alguns eixos de formação, a exemplo do desenvolvimento de sistemas e resolução de problemas, precisa do conteúdo da disciplina Algoritmos ou equivalente desde a fase inicial do curso. Logo, é possível inferirmos, para o bom seguimento em disciplinas futuras, como se estrutura os algoritmos e as competências adquiridas para alcançar este conhecimento que se fazem necessários.

Em geral, a disciplina é apresentada aos estudantes nos semestres iniciais do curso e, conseqüentemente, estabelece a primeira interação dos discentes com o desenvolvimento de software. O seu objetivo é desenvolver no aluno a capacidade de elaborar soluções lógicas mediante problemas genéricos em uma sequência de passos (DIAS JUNIOR; MERCADO, 2016). Segundo o documento referente à formação dos cursos superiores em Computação, tais competências e habilidades são bases para a formação dos futuros profissionais nas áreas de Tecnologia da Informação, com especial ênfase no eixo de desenvolvimento de software (SBC, 2017).

Logo, nas subseções a seguir apresentaremos como tem sido o ensino de algoritmos e as principais dificuldades encontradas no seu processo de ensino e aprendizagem.

#### 1.1.1 Como ensinar Algoritmos

O ensino de Algoritmos, segundo Timmermann e González (2016), deve ser capaz de permitir ao aluno elaborar estratégias como soluções para os problemas, as quais são escritas em linguagem algorítmica. Durante o processo de ensino e aprendizagem de algoritmos, o docente tem como objetivo ensinar técnicas, estratégias e metachecimento da forma mais

produtiva possível para conseguir uma aprendizagem significativa dos alunos neste campo (TIMMERMANN; GONZÁLEZ, 2016).

Os autores Rapkiewicz *et al.* (2007) relatam que a disciplina aborda os princípios da lógica de programação, com o objetivo de desenvolver a capacidade de análise e resolução de problemas dos alunos por meio da descrição destes na forma de algoritmos computacionais.

Não existe uma única forma como a disciplina pode ser abordada. Dentre as abordagens mais comuns para ensino estão: (i) o uso de problemas com enunciados textuais, os quais o professor procura trabalhar a destreza do aluno em entender o enunciado e após elaborar uma solução algorítmica em pseudocódigo ou em fluxogramas; (ii) o aluno realizar um “teste de mesa” para simular passo a passo o que o algoritmo deveria fazer para verificar se está alcançando o objetivo proposto; (iii) o professor acompanhar os alunos quando estes desenvolverem seus exercícios individualmente; (iv) os alunos utilizarem uma linguagem de programação, já no computador, para visualizar o seu progresso; e (v) o auxílio de jogos educacionais, entre várias outras. Algumas podem ser mais interessantes do que outras, mas o que se observa é que ainda persistem dificuldades de aprendizagem pelos estudantes.

Além das nossas percepções sobre como pode ser trabalhado o processo de ensino e aprendizagem de algoritmos, a revisão da literatura proposta por Dias Júnior e Mercado (2016), também aponta algumas possibilidades: (i) ensino presencial mediado por tecnologias; (ii) *electronic learning*; (iii) ensino presencial mediado por jogos ou design de jogos; (iv) ensino presencial mediado por robôs/*photosnap board*; (v) *blended learning*; e (v) *mobile learning*.

### **1.1.2 As dificuldades ao ensinar algoritmos**

As dificuldades apresentadas na disciplina Algoritmos são bastante evidentes na literatura, como podemos observar no estudo de Bosse e Gerosa (2015). O artigo apresenta principalmente os índices de aprovação (70%), reprovação por nota e/ou frequência (27%) e desistência (3%) dentro da disciplina Algoritmos e equivalentes ministradas na Universidade de São Paulo, no período de 2010 a 2014. Além disso, discute algumas das dificuldades encontradas no processo de ensino e aprendizagem da disciplina em questão, sendo uma delas a compreensão e assimilação dos conteúdos abordados por parte dos estudantes.

Na visão docente, temos o estudo de Dias Júnior e Mercado (2016) em que é apresentada a importância da introdução de práticas de simulação para o aprendizado de algoritmo. De acordo com os autores, os alunos sentem dificuldades de compreender e abstrair o funcionamento do algoritmo unicamente por meio da fala, leitura e explicação do professor.

Timmermann e González (2016) destacam outras dificuldades do processo de ensino e aprendizagem, que dizem respeito à interação entre os professores, os alunos e os conteúdos. Dentre estas, ressaltam a dificuldade no aprendizado do conteúdo pelos estudantes que não tiram dúvidas durante a aula e tampouco o professor consegue estabelecer relação com eles para auxiliá-los no processo.

O estudo de Giraffa e Mora (2016) indica, por meio de pesquisas informais na literatura, que em sua maioria as pesquisas encontradas apresentam a opinião dos docentes em relação à desistência e cancelamento da disciplina. Entre os vários fatores que impactam no cancelamento da disciplina pelos alunos, os mais relevantes e indicados são: compreender as aulas; compreender o enunciado dos exercícios; e desorganização pessoal para gerenciar os estudos.

Há também o trabalho de Vieira *et al.* (2015) que busca entender as dificuldades dos alunos na disciplina Algoritmos. A partir de questionários preenchidos pelos alunos sobre a disciplina, foi possível perceber como dificuldades as seguintes opções: raciocínio lógico; capacidade de abstração; leitura e interpretação de textos e conhecimentos matemáticos. Outro questionamento realizado pelos autores em suas turmas foi a dificuldade na aprendizagem do conteúdo, sendo os mais difíceis: estrutura de dados homogêneos, matrizes e vetores.

Para Gomes *et al.* (2008), existem problemas referentes à prática docente ao ensino de algoritmos, tais como: o ensino não ser personalizado; as estratégias de aprendizagem dos professores não contemplarem, habitualmente, todos os estilos de aprendizagem dos alunos; o ensino de conceitos dinâmicos é, normalmente, realizado por meio de materiais de natureza estática; e, os professores estão, normalmente, mais concentrados em ensinar uma linguagem de programação e os seus detalhes sintáticos, do que em promover a resolução de problemas usando uma linguagem de programação. No que se refere aos discentes, os autores indicam: a programação/algoritmo exige um estudo muito prático e intensivo e os estudantes não estudam o suficiente para adquirir as competências necessárias para programação.

No estudo de Raabe e Silva (2005), foi realizada uma análise sistemática das relações de ensino e aprendizagem na disciplina de Algoritmos durante nove ofertas, entre os anos de 2000 e 2004, em que os autores apontaram alguns aspectos que contribuem para as dificuldades na disciplina e os conglomeraram em três grupos distintos: (i) problemas de natureza didática; (ii) problemas de natureza cognitiva; e (iii) problemas de natureza afetiva. Como problemas de natureza didática, têm-se: (i) grande número de alunos em sala; (ii) dificuldade de o professor compreender a lógica desenvolvida pelo aluno; (iii) diferença de experiência e ritmo de aprendizagem entre os alunos; (iv) o ambiente para a realização das provas, segundo os autores o tempo limitado, a pressão e o stress, os quais em geral acompanham a realização da prova

não favorecem o desenvolvimento do raciocínio; (v) pouca presença de monitores na disciplina, e mesmo quando esta existe, há baixa procura por parte dos alunos dos monitores disponíveis; (vi) ausência de bons materiais, pois o conteúdo trabalhado nos livros e artigos disponíveis não são adequados para o aluno com dificuldade de compreender; e (vii) alunos desorientados na escolha do curso, ou seja, ainda não definiram os cursos da área de tecnologia da informação como carreira. Quanto aos problemas de natureza cognitiva, têm-se: (i) alunos sem perfil para solução de problemas; (ii) alunos sem base operatório-formal; (iii) conteúdo sem proximidade com o conteúdo escolar. Já no grupo referente aos problemas de natureza afetiva têm-se: (i) ocasionais, isto é, problemas de ordem pessoal que afetam o aluno; e (ii) constantes, ou seja, problemas de ordem afetiva manifestados no decorrer da disciplina.

Atender adequadamente às necessidades e dificuldades dos discentes, ao levar em consideração a diversidade de problemas possíveis é uma tarefa desafiadora para os docentes e com grande demanda de trabalho. Nesse contexto, observamos por meio da *gamificação* uma forma de permitir o engajamento dos estudantes para auxiliar no processo de ensino e aprendizagem, tema que trataremos na próxima seção desse referencial teórico.

## 1.2 GAMIFICAÇÃO

Dentro da área educacional, o estudo de Kapp (2012) conceitua *gamificação* como o emprego de elementos de jogos, tais como mecânica, estratégias e pensamentos, com a finalidade de motivar indivíduos à ação, auxiliando na solução de problemas e promovendo a aprendizagem.

Na visão de Fardo (2013), a abordagem *gamificada* proporciona um sistema em que os estudantes conseguem visualizar o efeito de suas ações e aprendizagens, à medida que compreendem a relação das partes com o todo, como acontece nos jogos. Portanto, um dos objetivos principais da abordagem é desenvolver nos indivíduos o sentimento de contribuição para algo maior e mais importante, por meio de suas ações.

Vianna *et al.* (2013) defendem o uso de elementos de jogos, por meio da abordagem *gamificada* na educação, como alternativa às abordagens tradicionais, com o intuito de familiarizar pessoas com novas tecnologias, agilizar processos de aprendizado ou de treinamento para tornar mais agradáveis as tarefas consideradas tediosas ou repetitivas. O estudo de Li *et al.* (2012) observou que o uso de *gamificação* por meio do material disponibilizado para o ensino aumenta o engajamento, pois proporciona níveis de prazer ao



usuário e contribui para o processo de criação do conhecimento, o que aprimora a aprendizagem do aluno.

Com o intuito de estimular o participante a desenvolver determinadas atividades, em especial por gerar no participante o sentimento de estar contribuindo para algo maior, tal qual a mudança de uma realidade, dentro do contexto da *gamificação*, podemos identificar nas abordagens descritas na literatura, diferentes *frameworks*, os quais definem a forma pela qual a *gamificação* será desenvolvida.

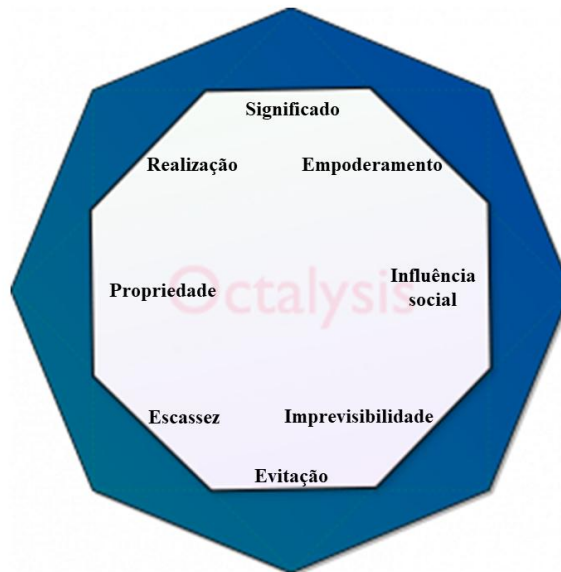
### 1.2.1 *Framework de gamificação*

Como já citado, por *framework gamificado* entendemos como um grupo de elementos de jogos realizados em ordem, com a finalidade de gerar engajamento. Para o trabalho em tela, adotamos como base os trabalhos de Landsell e Hägglund, (2016) e Wiklund e Wakerius, (2016), como indicações de *frameworks* de desenvolvimento de abordagens *gamificadas*, sendo eles: *Octalysis; Dynamics, Mechanics and Component (DMC) Pyramid; Mechanics, Dynamics and Aesthetics (MDA) e Mechanics, Dynamics e Emotions (MDE)* e (LANDSELL; HÄGGLUND, 2016; WIKLUND; WAKERIUS, 2016), os quais retrataremos nas subseções a seguir.

#### 1.2.1.1 *Octalysis*

O *framework Octalysis* foi concebido como uma ferramenta para construir estratégias e analisar a implementação de abordagens *gamificadas* (CHOU, 2013). A motivação para sua construção ocorreu pelo desenvolvedor, ao reconhecer sua necessidade durante as consultorias em treinamento para indústria (CHOU, 2013). No *framework*, o autor descreve oito tipos diferentes de núcleos que nos motivam a fazer atividades e compreender os conteúdos. Conforme o autor Chou (2013), o *framework* pode ser organizado e distribuído em duas formas: por meio de hemisférios e categorias.

Visualmente, o *Octalysis* tem uma forma de octógono, em que as unidades centrais são representadas em cada canto (Figura 3). Chou (2013) registra que todos os núcleos favorecem a motivação extrínseca ou intrínseca, sendo que as unidades no hemisfério direito da imagem representam os aspectos criativos, artísticos e sociais. Já as unidades do hemisfério esquerdo, referem-se a aspectos lógicos e intelectuais.

Figura 3 – *Framework Octalysis*

Fonte: adaptado de CHOU (2013).

Segundo Chou (2013), o seu *framework* (Figura 3) também pode ser descrito pelas seguintes categorias:

- *WhiteHat*: estão no lado superior do octógono e geram motivação positiva. Os seus núcleos e considerações são os seguintes:
  - **Significado** épico e chamado (*Epic Meaning & Calling*): faz com que os participantes acreditem estar realizando algo superior ou que foram escolhidos para executar aquela ação.
  - Desenvolvimento e **realização** (*Development & Accomplishment*): conduz os progressos e desenvolve as habilidades do jogador.
  - **Empoderamento** da criatividade e *feedback* (*Empowerment of Creativity & Feedback*): favorece o engajamento do usuário em um processo criativo, em que ele repetidamente descobre novas possibilidades e combinações dentro do jogo.
  - **Propriedade** e posse (*Ownership & Possession*): quando o usuário se sente motivado a ter ou controlar algo.

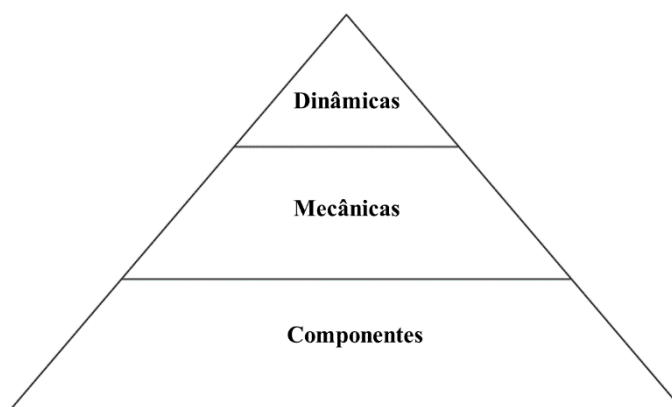
- **Influência social** e pertencimento (*Social Influence & Relatedness*): que são elementos sociais que buscam motivar os participantes.
- *BlackHat*: lado inferior do octógono, com motivação negativas, por meio do medo de perder, por dois motivos: curiosidade do elemento vindouro e o esforço para alcançar as metas ou recursos que ele/ela não pode ter. Os núcleos desse tipo de motivação são os seguintes com as considerações:
  - **Escassez** e impaciente (*Scarcity & Impatience*): o anseio por algo difícil de alcançar.
  - **Imprevisibilidade** e curiosidade (*Unpredictability & Curiosity*): a curiosidade de eventos futuros desconhecidos, bem como a emoção de possíveis recompensas.
  - Perda e **evitação** (*Loss & Avoidance*): o medo de perder algo negativo para ocorrer, além disso, o personagem se sente motivado pela excitação.

Vale destacar que o autor (CHOU, 2013) identifica que *gamificações* bem-sucedidas precisam de considerações e aplicação de todos os núcleos.

#### 1.2.1.2 *Dynamics, Mechanics and Component (DMC) Pyramid*

A pirâmide do DMC, em português Dinâmicas, Mecânicas e Componentes (Figura 4), é uma forma de *gamificação* proposta por Werbach e Hunter (2012), a qual é construída em torno de três categorias de elementos do jogo, identificadas no próprio nome da proposta. A chave para o desenvolvimento de uma experiência *gamificada* utilizando esse *framework* é o entendimento dos três tipos de elementos de jogo, para que se possa empregá-los, da melhor forma possível.

Figura 4 – Pirâmide DMC



Fonte: adaptado de Werbach e Hunter (2012).

Conforme a Figura 4, no topo da pirâmide encontra-se a dinâmica que descreve elementos de jogo que provocam a imersão do jogador na experiência *gamificada*.

Na segunda camada da pirâmide do DMC, encontra-se a mecânica do jogo, a qual tem por função estimular e envolver um jogador para realizar uma ação com o intuito de alcançar uma dinâmica (WERBACH; HUNTER, 2012). Por exemplo, no decorrer do jogo ou ambiente *gamificado*, uma recompensa mecânica pode aparecer para o jogador e isso, por sua vez, estimula a curiosidade e a felicidade, dinâmicas, para o jogador.

Na terceira camada da pirâmide do DMC estão os componentes, os quais têm a função de tornar mais clara para o jogador a dinâmica ou a mecânica, por isso registra-se como o menor nível de abstração. Por exemplo, um jogador tentando resolver um desafio, mecânica, pode ganhar pontos, componente, se conseguir resolver o desafio (WERBACH; HUNTER, 2012).

#### 1.2.1.3 *Mechanics, Dynamics and Aesthetics* (MDA)

Na pesquisa realizada por Hunicke *et al.* (2004), com o título de “MDA: uma abordagem formal para o design de jogos e pesquisa de jogos”, os autores identificam como trabalhar e entender o desenvolvimento de jogos, por meio de um *framework*. Nesse contexto, apresentam três componentes para a projeção de um jogo, são eles: mecânicas, dinâmicas e estéticas. No primeiro tipo são partes específicas do jogo, como a representação de dados adotada. As dinâmicas referem-se aos possíveis comportamentos adotados pelo usuário interagindo com as mecânicas. A estética é uma descrição das respostas emocionais adotadas pelo jogador no decorrer do uso do jogo.

Sobre a estética, os autores do *framework* registram a necessidade de se afastarem dos termos “diversão” e “jogabilidade” e se aproximarem da seriedade do desenvolvimento de

jogos. Portanto, a estética deve gerar as seguintes experiências *gamificadas*: (i) sensação: o jogo como prazer dos sentidos; (ii) fantasia: o faz de conta; (iii) narrativa: um drama gerado; (iv) desafio: obstáculos para os jogadores superarem; (v) companheirismo: formar equipes e interação entre os personagens; (vi) descoberta: o território para ser explorado dentro do jogo; (vii) expressão: o jogo se tornar um ambiente de descoberta para os jogadores; e (viii) submissão: o ato de jogar se torna um passatempo.

#### 1.2.1.4 *Mechanics, Dynamics and Emotions* (MDE)

Desde a concepção e publicação do *framework* MDA, diferentes autores sugeriram modificações para adequá-lo a diferentes contextos. Uma das adequações deu origem ao *framework* MDE, em português: mecânica, dinâmica e emoção (Figura 5), em que o conceito de estética é substituído por emoções para descrever a experiência do usuário. Robson *et al.* (2015, p. 413) argumentam que a estética descreve “as respostas emocionais desejáveis (por exemplo, fantasia, submissão, companheirismo, descoberta) evocadas nos jogadores quando eles interagem com o jogo”. Portanto, a estética é substituída por emoções, já que “é melhor vinculado aos resultados de engajamento que as empresas podem obter dos funcionários e clientes” (ROBSON *et al.*, 2015, p. 413).

Figura 5 – *Mechanics, Dynamics and Emotions* (MDE)



Fonte: adaptado de Robson *et al.* (2015).

Robson *et al.* (2015), ressaltam a importância, para uma *gamificação* bem-sucedida, da compreensão da relação entre os seguintes princípios: mecânica, dinâmica e emoções:

- **Mecânica:** são definições adotadas no início da *gamificação*, as quais não são modificadas, tais como regras, configuração, contexto, tipos de interações e os

limites. As mecânicas referem-se aos aspectos fundamentais da experiência *gamificada*, pois determinam quem são as partes principais, como interação, como ganhar ou perder, e onde e quando a experiência ocorre. Além disso, são subdivididas em três tipos: configuração, regras e progressão.

- **Dinâmica:** refere-se ao tipo de comportamento do jogador que surge ao participar da experiência *gamificada*. É difícil prever as dinâmicas, pois pode levar a comportamentos e resultados não intencionais, que podem ser positivos ou negativos por natureza. Portanto, o desafio para os projetistas é antecipar os tipos de dinâmica possíveis que podem emergir, para poder desenvolver a mecânica da experiência apropriadamente.
- **Emoção:** registra as reações evocadas entre os jogadores individuais quando participam de uma experiência *gamificada*. São um produto de como os jogadores seguem a mecânica e geram dinâmicas. Para os autores, as interações, além de serem divertidas geram uma mistura de sentimentos, sendo eles positivos ou negativos.

### 1.2.2 Os elementos de jogos

A construção de um jogo ou uma *gamificação* é realizada por meio de suas partes, definidas na literatura por elementos. Werbach e Hunter (2012) definem o elemento de *games* como um kit de ferramentas para criar um jogo. Para os autores, existem três tipos de elementos:

- **Dinâmica:** (i) restrições: limitações existentes para os jogadores nas atividades realizadas na *gamificação*; (ii) emoções: sentimentos gerados no jogador no decorrer da *gamificação*, como por exemplo competição, curiosidade, frustrações e felicidade; (iii) narrativa: uma história para ser vivenciada com enredo contínuo e persistente; (iv) progressão: desenvolvimento de um jogador na experiência *gamificada*; e (v) relacionamentos: as relações sociais possibilitadas pela interação entre os jogadores.
- **Mecânicas:** (i) desafios: tarefas que exigem esforço do jogador para resolver; (ii) chance: situações aleatórias que podem ser incorporadas ao sistema *gamificado*; (iii) concorrência: ambientes competitivos com vencedores e perdedores; (iv)

cooperação: alcançar objetivos com trabalho em equipe entre os jogadores; (v) comentário: *feedback* de como está o progresso do jogador no jogo ou *gamificação*; (vi) aquisição de recursos: os jogadores no decorrer do jogo ou *gamificação* adquirirão itens valiosos, os quais podem auxiliá-los em momentos futuros, como por exemplo mais tempo para realizar um desafio, ou então uma dica para a solução; (vii) recompensas: os jogadores são beneficiados por comportamentos e realizações ; (viii) transações: negociações entre jogadores sobre itens adquiridos.

- Componentes: (i) *avatares*: representação visual da identidade e da mudança do personagem em relação ao jogo ou *gamificação*; (ii) combate: durante a jornada o jogador ter lutas de curta duração para vencer desafios propostos no jogo ou *gamificação*; (iii) desbloqueio de conteúdo: algumas partes específicas da *gamificação* ou jogo ficam disponíveis quando um jogador alcança um certo objetivo, como por exemplo: uma parte do mapa, um item de ajuda, uma fase bônus; (iv) *gifting*: a liberdade de compartilhar os recursos ou objetos entre os personagens; (v) gráficos sociais: interação dos jogadores por meio das redes sociais na plataforma do jogo ou experiência *gamificada*; (vi) equipes: grupo de jogadores que cooperam para realizar metas juntos; e (vii) bens virtuais: itens com valor monetário dentro da experiência *gamificada* ou jogo.

Deterding *et al.* (2011) alertam que pessoas que acreditam na *gamificação* como simples adição de recompensas, incentivos com pontos e medalhas dentro do sistema estão destinadas a fracassar no desenvolvimento de suas abordagens. Portanto, a melhor forma de *gamificação* é a combinação de vários elementos de jogo nos seus mais variados tipos para gerar o engajamento e a experiência diferencial para os participantes.

### 1.3 TRABALHOS RELACIONADOS

Em relação ao uso de *gamificação* na aprendizagem, no trabalho de Freitas *et al.* (2016) há a aplicação da abordagem em um contexto mais específico, em que ela é realizada em uma disciplina chamada Fundamentos de Arquitetura de Computadores, que se utilizou da sala de aula, de um espaço virtual e do jogo desenvolvido. A aplicação teve como principal objetivo realizar duelos de conhecimento entre os alunos, em que um conjunto de duelos constituía uma

batalha, sempre relacionada a um ou mais tópicos da disciplina. Como resultado, houve maior interesse e motivação dos envolvidos no jogo, além de uma melhoria em relação à aprendizagem e à absorção de conhecimento por parte dos alunos.

Na pesquisa realizada por Medeiros e Figueiredo (2015), há a apresentação de uma ferramenta, utilizada para a criação de disciplinas *gamificadas*, intitulada de “Game in Class”, cuja principal função é planejar e construir disciplinas utilizando técnicas e elementos de jogos. A ferramenta é direcionada para professores do ensino superior, que tem como interesse diversificar o ensino das suas disciplinas, e que queiram ministrá-las de uma forma diferenciada e inédita por meio da *gamificação*.

O artigo de Monteiro, Oliveira e Martins (2015) estudou a possibilidade do ensino de programação por meio da utilização de elementos da *gamificação*, associando a abordagem ao ensino a distância. As tarefas que foram executadas no curso foram divididas em quatro fases, em que foram apresentados os conceitos iniciais acerca da construção de algoritmos, e após cada fase, assim como em um jogo, o aluno avançaria de nível até chegar na missão final, em que todo o conhecimento adquirido seria utilizado para resolver o desafio. Com isso, notou-se que a abordagem utilizada de fato contribuiu e favoreceu o engajamento, além de ter melhorado o nível de aprendizagem dos alunos.

O trabalho de Falcão, Leite e Tenório (2014) objetivou o desenvolvimento de uma ferramenta para apoiar o ensino presencial, utilizando a *gamificação* e o *design* de jogos. O estudo propôs um ambiente que estimulasse o interesse e a atenção do aluno fora da sala de aula, por meio da aproximação dos conteúdos trabalhados com o meio digital. Além disso, a pesquisa levou em consideração a grande demanda da disciplina Algoritmos ou equivalente nas instituições de ensino, como justificativa para o desenvolvimento da ferramenta.

No estudo de Gonçalves *et al.* (2016), é apresentado um modelo conceitual de apoio ao planejamento da *gamificação*, em que são levados em consideração o contexto em que a abordagem deverá ser implementada, os objetivos educacionais, as habilidades que serão necessárias, os comportamentos e as interações que são esperadas. Além disso, é realizada uma descrição do processo no contexto da educação, para que a *gamificação* seja realizada da forma correta.

Em algumas das abordagens que foram citadas nessa dissertação, os autores optaram pelo desenvolvimento de ferramentas ou protótipos para a aplicação dos elementos da *gamificação*, já em outros, são utilizados apenas os conceitos referentes às dinâmicas e à mecânica dos jogos. No estudo em tela, utilizamos os conceitos presentes nesta abordagem para realizar o ensino da disciplina algoritmos ou equivalente.



## 2 CONCEPÇÃO DO FRAMEWORK GAMIFICADO

Durante a fase de concepção foram definidos alguns procedimentos metodológicos de pesquisa com a finalidade de identificar os requisitos necessários para a concepção e construção de um *framework gamificado* para o processo de ensino e aprendizagem da disciplina Algoritmos ou equivalente, levando em consideração a questão-foco da pesquisa e os objetivos a serem alcançados. Este capítulo apresenta as seguintes subseções: um survey de abrangência nacional com docentes e discentes; uma observação participante realizada na Universidade Federal do Pará; e a estrutura de *gamificação* adotada pelo projeto SPIDER dentro dos cursos de graduação e pós-graduação na UFPA em disciplinas da área de Engenharia de Software.

### 2.1 SURVEY

O *survey* é um “tipo de *delineamento de pesquisa descritiva* cujo objetivo é o de verificar o estado atual de dado fenômeno [...] basicamente consiste na *coleta de dados* seguida de uma descrição dos mesmos, por meio das técnicas da *chamada estatística descritiva* [...]” (APPOLINÁRIO, 2009, p. 122). Segundo Mafra e Travassos (2006), a execução desse formato de pesquisa permite capturar um “retrato instantâneo” da atual situação a ser pesquisada.

O relatório de Mafra e Travassos (2006) aponta que os principais meios de coleta dos dados para um *survey* são as entrevistas ou questionários, sendo que essa coleta precisa representar uma amostra significativa da população e os resultados, ao serem analisados, precisam gerar conclusões genéricas para a população.

O *survey* realizado como parte da pesquisa para a concepção do produto teve como principais objetivos:

- Identificar quais abordagens e recursos vem sendo utilizados a nível nacional, para ministrar a disciplina Algoritmos ou equivalente.
- Indicar perfis de comportamento/atitude dos discentes durante a realização da disciplina Algoritmos ou equivalente.
- Levantar o grau de contribuição da estratégia de bonificação para a motivação do estudante.
- Apurar quais são os conhecimentos prévios necessários para o aprendizado de algoritmos.

- Apontar quais são os conteúdos mais difíceis pertinentes à disciplina tanto para ensinar quanto aprender.

### 2.1.1 *Design do Survey*

O *survey* foi construído na ferramenta LimeSurvey (LIMESURVEY, 2018), específica para a concepção de *survey* com divulgação via internet, disponibilizada pelo Núcleo de Inovação e Tecnologias Aplicadas a Ensino e Extensão, da Universidade Federal do Pará (NITAE<sup>2</sup> - UFPA). O questionário do *survey* ficou disponível para docentes, no período de 22 de novembro de 2017 a 15 de janeiro de 2018, e para os discentes, no período de 04 de dezembro de 2017 a 15 de janeiro de 2018.

As populações-alvo foram professores e estudantes de cursos de graduação na área de TI de instituições de ensino superior públicas ou privadas brasileiras, que tiveram contato com a disciplina Algoritmos ou equivalente. Quanto ao *design* de coleta de dados, o *survey* pode ser classificado como de corte transversal, em que os participantes relatam informações de suas experiências passadas e suas visões dentro de um determinado contexto (MAFRA; TRAVASSOS, 2006).

Como estratégia de divulgação para docentes, utilizamos listas de *e-mails* disponíveis pela Sociedade Brasileira da computação (SBC), a saber: lista geral da entidade; Ensino de Algoritmos e Programação; Olimpíada Brasileira de Informática; Pesquisa e Desenvolvimento; Assuntos Ensino e Pesquisa; Comissões Especiais de Linguagem de Programação; e Comissão Especial de Informática na Educação. Além disso, se utilizaram também *e-mails* de cunho privado vinculados a professores de Instituições de Ensino Superior (IES) do estado do Pará e outras, os quais encontravam-se disponíveis ao acesso público em páginas de faculdades. Após o primeiro convite, foram remetidas às listas três lembretes, com o intuito de coletar dados de professores que, a priori, não haviam sido alcançados nas primeiras tentativas.

Em relação à estratégia de divulgação para os discentes, utilizamos o grupo de *e-mails* dos cursos CBCC, CBSI e Engenharia da Computação da UFPA. Além disso, um convite para participar do *survey* foi publicado em diversos grupos de estudantes na mídia social Facebook, tais como: Sistemas de Informação; Engenharia de Software PUC Rio e Minas; Computação do Centro Universitário Luterano de Palmas, entre outros.

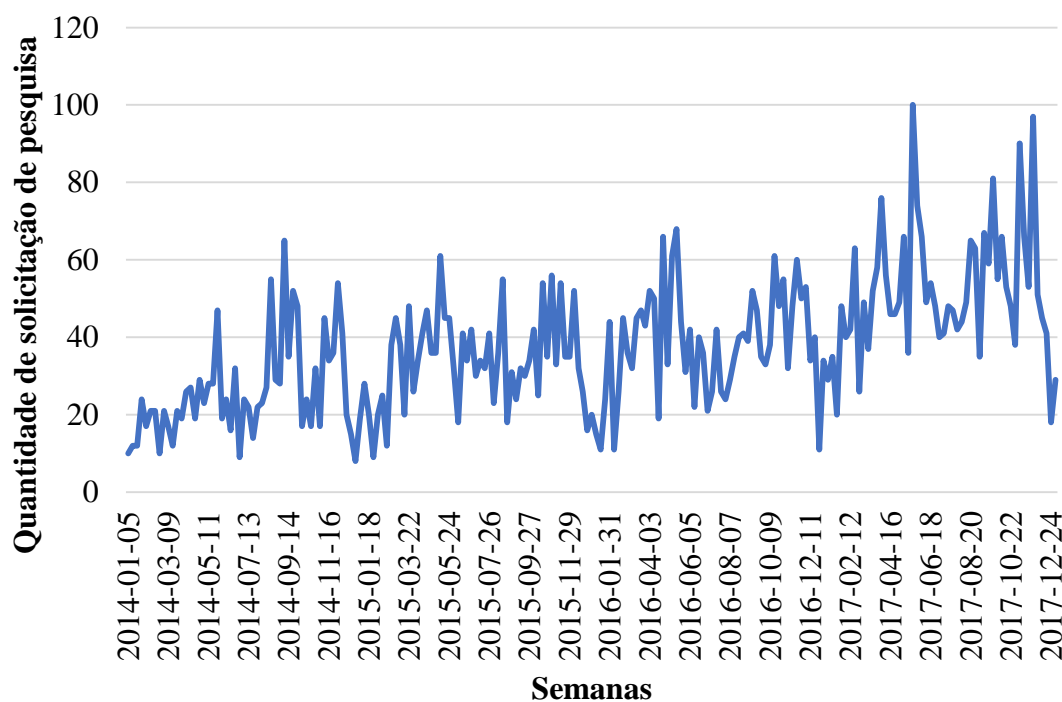
Este *survey* limitou-se a coletar dados quantitativos com relação à opinião pessoal e experiência acadêmica. Critérios de inclusão e exclusão de participantes do *survey* foram definidos, conforme o Quadro 2.

Quadro 2 – Critério de inclusão e exclusão de participantes do *survey*

Critério de inclusão	Consideração
O docente ou discente ter tido contato com a disciplina no período compreendido entre 2015 até 2017.	Este período foi escolhido devido ao fato de o termo <i>gamificação</i> estar mais em evidência, sendo esta abordagem a ser usada na pesquisa.
Critério de exclusão	Consideração
O participante que não estivesse motivado a participar da pesquisa.	Os participantes que não preencheram o questionário em sua totalidade foram excluídos.

Fonte: Elaboração própria (2019).

A definição do critério de inclusão, apresentada no Quadro 2, levou em consideração o contato do participante com a disciplina no período entre 2015 e 2017 e a escolha por esse ciclo se deu a partir do levantamento que fizemos pelo termo *gamificação* ao longo dos anos na ferramenta *Google Trends*. Como ilustrado na Figura 6, percebemos uma ampliação na quantidade de resultados da pesquisa dentro do período compreendido.

Figura 6 – Pesquisa pelo termo *gamificação* ao longo dos anos

Fonte: (GOOGLE TRENDS, 2019).

### 2.1.2 Instrumentação do *survey*

Durante a aplicação do *survey* empregamos dois instrumentos com questões objetivas e uma subjetiva, um para docentes e outro para discentes (Apêndices A e B). Para a composição dos questionários, foram pesquisadas referências sobre o processo de ensino e aprendizagem

da disciplina Algoritmos ou equivalente, bem como sobre a abordagem de *gamificação*, sistematizados no Quadro 3. Posteriormente, definimos que as questões deveriam perpassar pelas dimensões de ensino, aprendizagem e conteúdo.

Quadro 3 – Dimensões dos instrumentos e referenciais teóricos de base

Dimensões	Descrição	Referencial teórico
Ensino e aprendizagem de algoritmos	Estavam presentes questões referentes a abordagens, recursos, linguagens de programação utilizadas, pré-requisitos, ferramentas e formas de bonificação utilizadas na disciplina.	<ul style="list-style-type: none"> <li>• (BOSSE; GEROSA, 2015).</li> <li>• (MARCUSI <i>et al.</i>, 2016).</li> <li>• (DIAS JÚNIOR; MERCADO, 2016).</li> <li>• (RAMOS <i>et al.</i>, 2015).</li> <li>• (RAAB; SILVA, 2005).</li> </ul>
Conteúdo	Estavam presentes questões envolvendo os conteúdos constantes na disciplina, considerando principalmente as informações disponíveis em um Projeto Político Pedagógico e o currículo base da Sociedade Brasileira da Computação.	<ul style="list-style-type: none"> <li>• Currículo base da Sociedade Brasileira de Computação<sup>2</sup> (SOCIEDADE BRASILEIRA DA COMPUTAÇÃO, 2005).</li> <li>• Projeto Político Pedagógico do CBCC da UFPA (UNIVERSIDADE FEDERAL DO PARÁ, 2010)</li> <li>• Disciplina Conceitos Fundamentais de Programação “<i>Fundamental Programming Concepts</i>”, tradução feita pelo autor (ASSOCIATION FOR COMPUTING MACHINERY/ INSTITUTE OF ELECTRICAL AND ELECTRONIC ENGINEERS, 2013)<sup>3</sup></li> </ul>
<i>Gamificação</i>	Presente apenas no instrumento do aluno, constava questões referentes à abordagem <i>gamificada</i> .	<ul style="list-style-type: none"> <li>• (WERBACH, 2012)</li> <li>• (MCGONIAL, 2011)</li> <li>• (COSTA; MARCHIORI, 2016)</li> </ul>

Fonte: Elaboração própria (2019).

Com base nos referenciais e dimensões citados, definimos as questões do *survey*. O Quadro 4 apresenta a divisão por categoria das questões elaboradas para docentes e discentes.

Quadro 4 – Categorias dos questionários (Continua...)

Categoria	Conteúdo das questões
Informações básicas	Questões referentes ao tempo de contato com a disciplina e quantidade de vezes ministrada ou de participação.

Fonte: Elaboração própria (2019).

Quadro 4 – Categorias dos questionários (Conclusão)

Categoria	Conteúdo das questões
-----------	-----------------------

<sup>2</sup> A disciplina Algoritmos é trabalhada em conjunto com os conteúdos da disciplina Estrutura de dados.

<sup>3</sup> Os conteúdos propostos pela disciplina *Fundamental Programming Concepts* são equivalentes aos da disciplina Algoritmos.

Em relação ao processo de ensino	Questões sobre abordagem didática, recursos necessários e/ou significativos para aprendizagem, formato de divisão dos alunos, presença do monitor, forma na qual docente resolve o problema em sala, uso de bonificação, utilização de linguagens de programação e <i>feedback</i> sobre as atividades desenvolvidas na disciplina.
Em relação ao conteúdo da disciplina	Questões sobre os conteúdos da disciplina e o nível de dificuldade de cada um tanto para ensinar, quanto para aprender.

Fonte: Elaboração própria (2019).

No Quadro 5 descrevemos as questões feitas com exclusividade para os discentes, com o objetivo de avaliar o contato destes com a abordagem de *gamificação* e dos principais elementos de jogos.

Quadro 5 – Questões exclusivas para alunos

Questão	Considerações
O que mais lhe motiva a jogar um <i>game</i> ?	A elaboração desta questão trouxe alguns elementos sobre mecânicas de jogos, com o intuito de identificar, para a construção da abordagem proposta, quais destes poderiam ser significativos para motivar o aluno no processo de ensino e aprendizagem.
Durante a sua graduação, em alguma disciplina cursada, o professor fez uso da <i>gamificação</i> como estratégia de ensino?	Esta questão procurou identificar um prévio contato com a abordagem que se deseja construir.
O uso da <i>gamificação</i> contribuiu para melhorar o seu processo de aprendizagem do conteúdo?	Para os estudantes que responderam “sim”, foi investigado se esta abordagem foi considerada válida para a aprendizagem.

Fonte: Elaboração própria (2019).

Antes de disponibilizar o preenchimento dos questionários para os participantes, o instrumento docente foi submetido a uma avaliação por especialistas, por meio de envio para endereço eletrônico. No *e-mail* enviado aos revisores foi informado o objetivo geral do estudo, o pertencimento ao PPGCIMES da UFPA, as outras metodologias adotadas como alinhamento para o desenvolvimento da pesquisa, bem como os seguintes critérios para avaliação do questionário: (i) clareza e entendimento das questões e respostas; (ii) quantitativo de questões presentes no instrumento; (iii) navegação e preenchimento do instrumento via *web*; (iv) identificação de questões que podem ser suprimidas, reformuladas e/ou que precisam ser incorporadas ao instrumento; e (v) se os dados gerados pelo instrumento permitem levantar requisitos para traçar estratégias de *gamificação* para a disciplina Algoritmos ou equivalentes.

O perfil dos especialistas deveria ser de professores da disciplina Algoritmos ou equivalentes com pesquisas desenvolvidas e publicadas na área de *gamificação*. Diante do perfil informado, foram convidados três revisores, dos quais apenas dois retornaram. A resposta da análise dos avaliadores foi de concordância com todos os itens solicitados, apenas com a

seguinte observação: acrescentar no nome disciplina Algoritmos a palavra equivalente, pois a disciplina recebe outras denominações nos mais variados cursos de tecnologia da informação. No que se refere ao instrumento discente, como o quantitativo e as questões eram um reflexo do docente e este último já havia sido avaliado positivamente, foi considerado avaliado e validado. O tempo total disponibilizado para avaliação foi de uma semana, tendo início no dia 26 de outubro de 2019, com a devolutiva ocorrida com três dias.

### **2.1.3 Resultados e discussão do *survey***

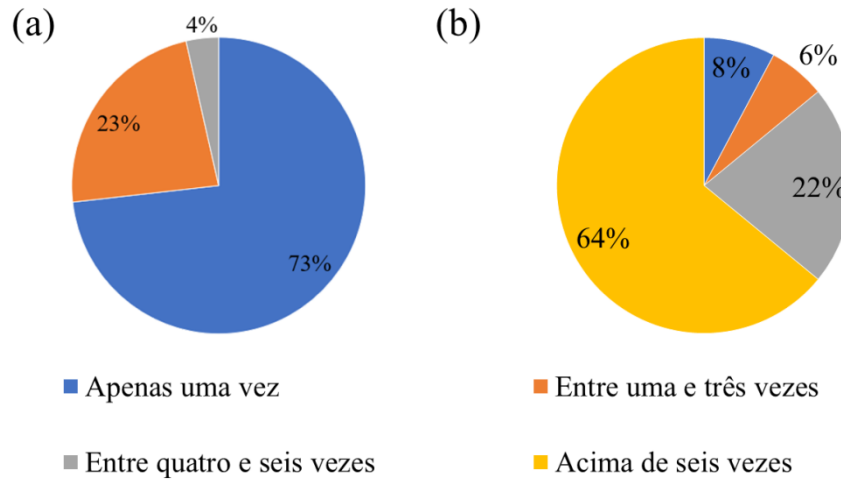
Esta seção apresenta os resultados obtidos com a aplicação do *survey* para os docentes e discentes da disciplina Algoritmos ou equivalentes, bem como sua análise e discussão.

#### **2.1.3.1 O perfil dos participantes**

Entre discentes e docentes, foram recebidas respostas de 109 participantes provenientes de instituições de ensino superior públicas e privadas, sendo que após a execução do refino pelo critério de inclusão e exclusão dos respondentes totais, chegamos a um total de 56 respostas discentes e 64 respostas docentes. Podemos considerar o *survey* como de abrangência nacional, pela estratégia de divulgação citada anteriormente.

Dentre os professores que participaram do *survey* (conforme Figura 7), 64% indicaram que já haviam ministrado a disciplina mais de seis vezes e 22% entre quatro e seis vezes, o que foi considerado um resultado positivo, pela possível qualidade das respostas, dada a grande experiência acumulada na docência dos conteúdos de algoritmos. Com relação aos discentes, 73% declararam ter cursado a disciplina apenas uma vez, 23% duas ou três vezes e 4% quatro ou cinco vezes. Este também é um dado importante, pois demonstra uma diversidade na amostragem.

Figura 7 – Participação na disciplina: (a) quantidade de participação dos discentes; (b) quantidade de vezes ministrada pelos docentes

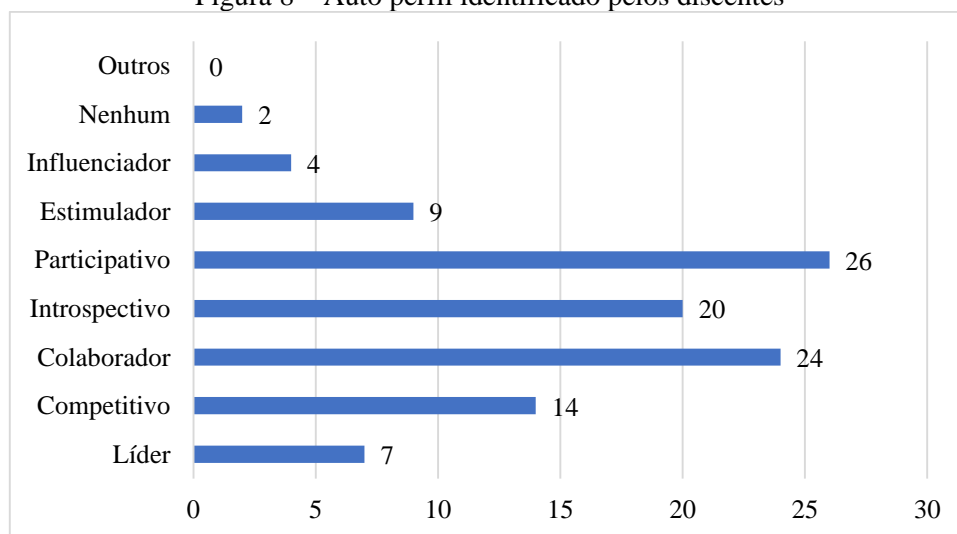


Fonte: Dados da pesquisa (2019).

### 2.1.3.2 Perfis discentes identificados pelo *survey*

Na aplicação dos questionários referentes à identificação de perfis do discente em sala de aula, foram enumeradas diferentes categorias e os perfis “participativo”, “colaborador” e “introspectivo” foram os mais apontados. Perfis descritos por “competitivo” e “estimulador” também foram registrados como relevantes. Os demais não representam margem significativa à pesquisa, conforme demonstra a Figura 8.

Figura 8 – Auto perfil identificado pelos discentes

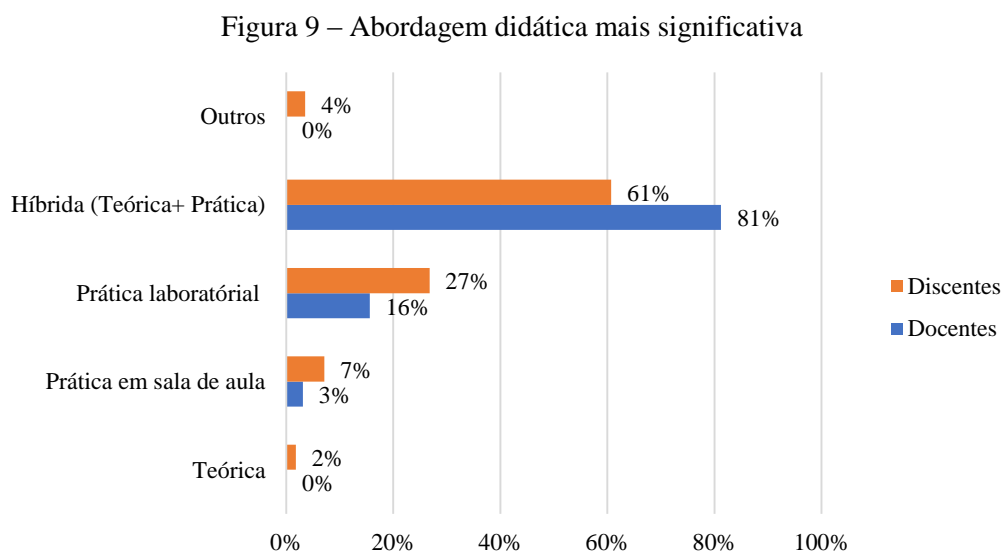


Fonte: Dados da pesquisa (2019).

Já para os docentes, os perfis mais indicados, na questão aberta, foram: “competidor” (11), “colaborador” (9), “líder” (7), “participativo” (7) e “introspectivo” (5). Estes são perfis cujas características são de cunho comportamentais. Assim, a partir destes perfis, para a proposta de *gamificação*, foram utilizados como elementos para bonificar os alunos: a participação e a colaboração, uma vez que foram os principais perfis apontados tanto pelos docentes como pelos discentes. Durante a análise dos dados dos docentes, foi possível identificar, ainda, perfis mais direcionados ao conteúdo, os quais foram empregados na proposta *gamificada* como *avatars*. São eles: analista, programador, aprendiz e amador.

### 2.1.3.3 Abordagens e recursos utilizados na disciplina

Na Figura 9 são confrontadas as informações dos professores e dos alunos em relação à abordagem didática. Podemos perceber que, tanto na visão docente, 81%, quanto discente, 61%, a abordagem identificada como efetiva é a híbrida, isto é, teórica junto à prática. A segunda indicação é a prática laboratorial, em que 27% dos alunos e 16% dos professores escolheram este tipo de formato de aula.

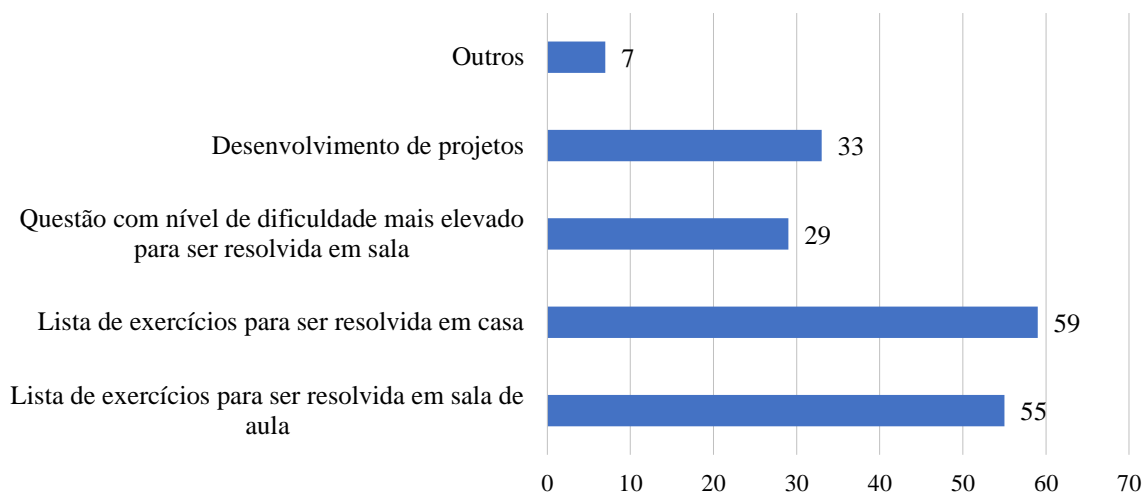


Fonte: Dados da pesquisa (2019).

No quesito recursos de apoio à prática de ensino, os docentes participantes do *survey* elencaram a adoção de listas com exercícios para resolução em casa (59 apontamentos), como o recurso principal. Posteriormente, optou-se por exercícios resolvidos em sala de aula e desenvolvimento de projetos, 55 e 33 registros respectivamente (vide Figura 10).



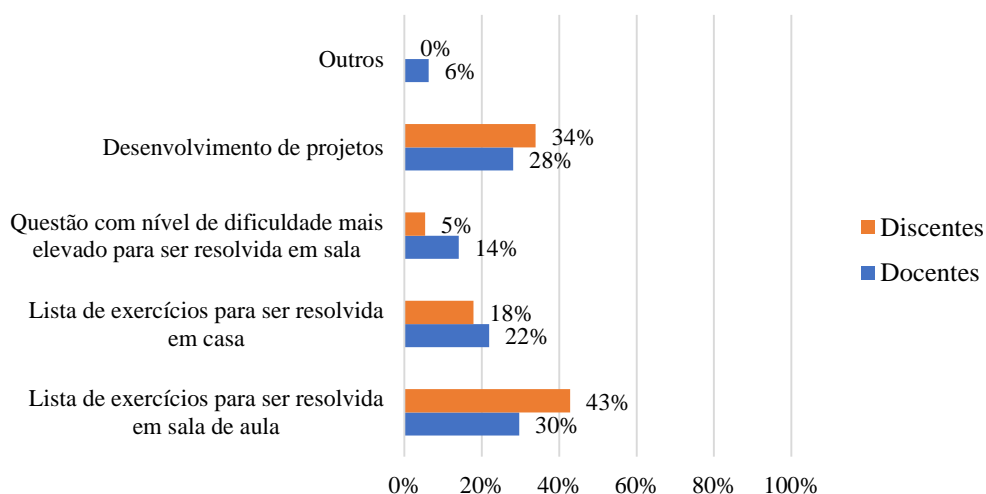
Figura 10 – Recursos utilizados para apoiar o ensino



Fonte: Dados da pesquisa (2019).

Já no que se refere aos recursos mais significativos para a aprendizagem, a primeira indicação é a Lista de exercício para ser resolvida em sala de aula, cujas respostas de docentes foram de 43% e de discentes 30%. Na segunda está o desenvolvimento de projetos, em que as respostas dos docentes foram de 34% e de discentes de 28%. Na terceira posição dos registros, está a lista de exercícios para ser resolvida em casa, a qual conta com 18% das indicações feitas por discentes e 22% por docentes (vide Figura 11).

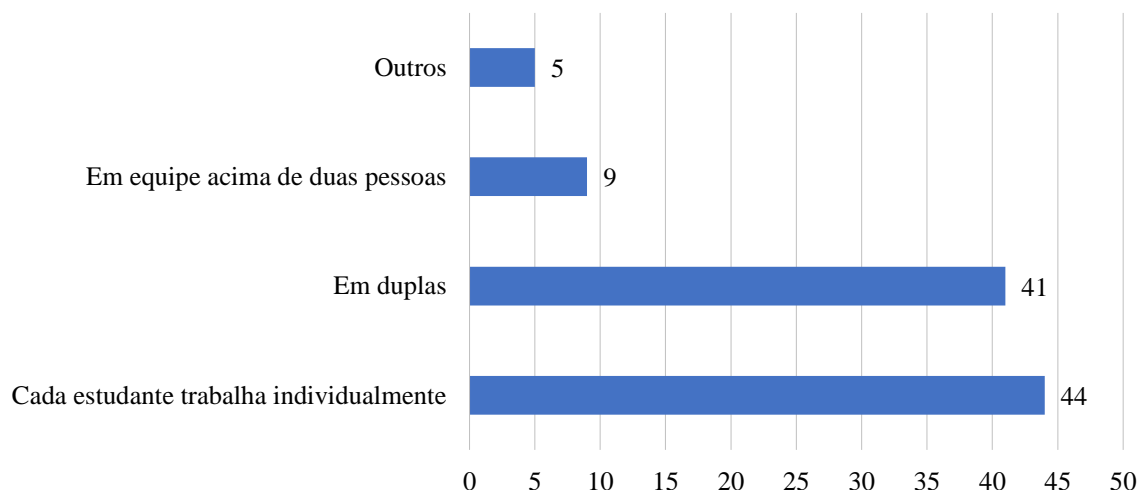
Figura 11 – Recursos significativos para ensino e aprendizagem



Fonte: Dados da pesquisa (2019).

No que tange à divisão dos alunos para a resolução de algoritmos, exposta na Figura 12, a pesquisa demonstra que 44 docentes adotam o trabalho individual e 41 optam pela realização de trabalhos em dupla.

Figura 12 – Escolha da forma como são divididos os discentes em sala de aula

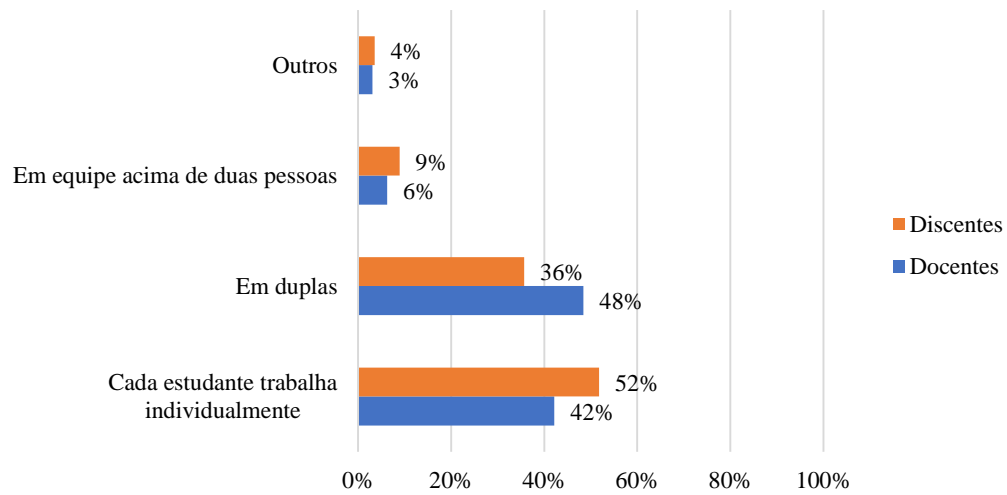


Fonte: Dados da pesquisa (2019).

No que diz respeito à divisão dos alunos para a resolução das atividades na disciplina, o formato que os docentes consideram mais significativo para a aprendizagem é em dupla, com 48% das respostas. Na segunda colocação está o trabalho individual, com 42%. Entretanto, os discentes apontaram o inverso, em que na primeira posição está o trabalho individual, com 52% das respostas e, em segundo, o trabalho em dupla com 36%. Os dados são mostrados na Figura 13.

Em relação à divergência de informação, os docentes indicam o formato em duplas. Já os discentes preferem o entendimento e resolução individual. O controle da disciplina é do docente, então, alguns discentes da pesquisa podem não ter tido contato com alguns formatos.

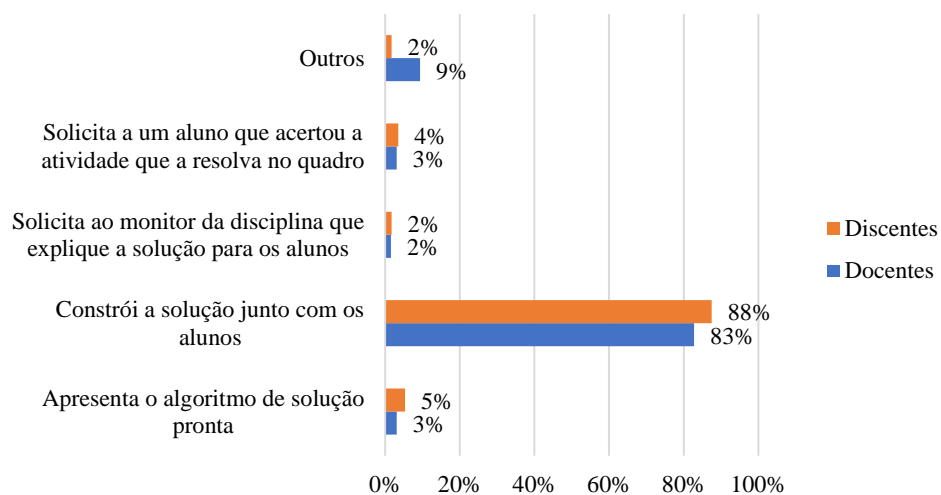
Figura 13 – Formato da divisão dos alunos



Fonte: Dados da pesquisa (2019).

Em relação à preferência da resolução de problemas em sala de aula, a maioria escolheu a opção de construir a solução junto com os alunos, em que o percentual de respostas dos docentes foi de 83% e dos discentes de 88% (vide Figura 14). Este é um resultado interessante visto que a construção coletiva de algoritmos pode permitir com que os alunos adquiram a competência fundamental da disciplina, que é a escrita desse algoritmo. Segundo Jesus e Brito (2009), uma das suas características é a não existência de uma única solução para um determinado problema.

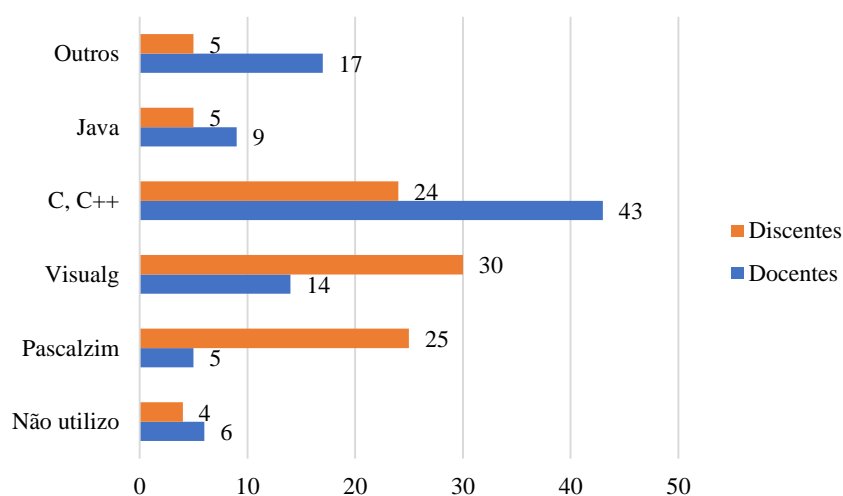
Figura 14 – Formas de resolução dos algoritmos em sala pelo professor



Fonte: Dados da pesquisa (2019).

No que se refere ao uso de *softwares* ou linguagens de programação para facilitar o processo de aprendizagem da disciplina, visualizadas a partir da Figura 15, os docentes informaram que adotam as seguintes linguagens em ordem de apontamentos quantitativos: *C/C++*, *Visualg* e *Java*. Quanto aos discentes, o apontamento é de que seus professores adotaram as seguintes linguagens em ordem de registros quantitativos: *Visualg*, *Pascalzim* e *C/C++*. Na opção “outros”, para docentes, foram identificados alguns registros como *Python*, *Portugol*, *Javascript*, *Devpascal* e *Scratch*.

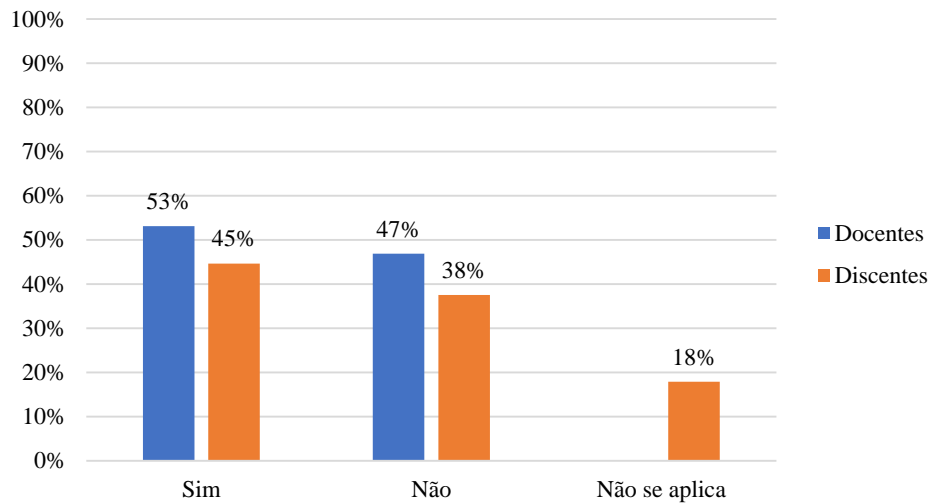
Figura 15 – Uso de linguagens de programação na disciplina



Fonte: Dados da pesquisa (2019).

A Figura 16 traz informações sobre a presença e apoio de monitores na disciplina. Assim, ressaltamos que a opção “não se aplica” estava disponível apenas para os alunos, pois durante a participação na disciplina eles poderiam não ter tido contato com o monitor. Os resultados presentes na Figura 16 indicam que 53% dos docentes afirmaram contar com a participação do monitor no decorrer da disciplina, enquanto 47% não. Já no caso dos discentes, 45% indicaram procurar pelo apoio do monitor para auxílio nas atividades, 38% não e 18% registraram o não contato com o monitor durante a disciplina.

Figura 16 – Participação e apoio do monitor na disciplina



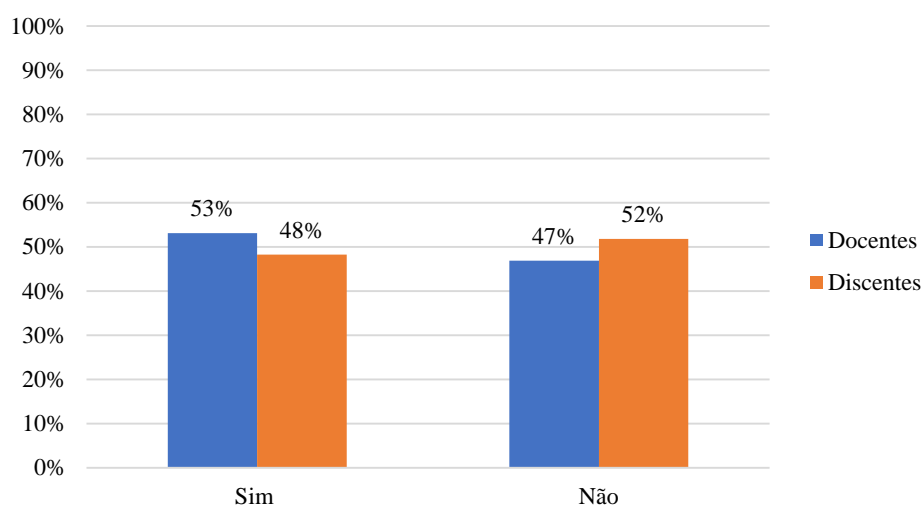
Fonte: Dados da pesquisa (2019).

#### 2.1.3.4 Contribuição da estratégia de bonificação

Como a intenção era o desenvolvimento de uma abordagem *gamificada* para a disciplina Algoritmos ou equivalente, foram coletadas informações sobre a ocorrência da bonificação durante as aulas. Para tal, primeiramente, questionou-se sobre o seu uso, em que 53% dos docentes responderam positivamente, enquanto 47% relataram negativamente. Entre os discentes a situação é diferente, pois 48% apontaram que os professores utilizam este recurso, mas 52% não tiveram contato (vide Figura 17).

A divergência em relação ao uso ou não da bonificação na disciplina pode estar relacionada à prática docente, visto que nem todos os docentes podem achar este recurso significativo para a aprendizagem. Em relação aos discentes, como já exposto, o controle de turma é do professor, ou seja, cabe a ele decidir adotar ou não a bonificação em sala de aula. Outro possível motivo pode estar relacionado ao fato de o discente não ter a compreensão sobre a estratégia, não conseguindo identificá-la nas práticas docentes, como, por exemplo, a lista de exercício pontuada, ou ponto “extra” pela resolução de uma atividade no quadro.

Figura 17 – Uso de bonificação na disciplina

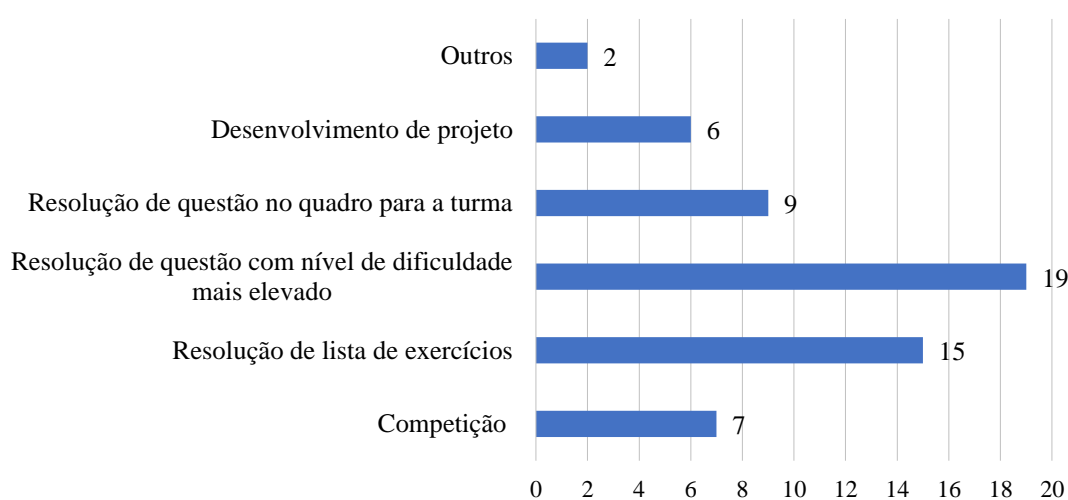


Fonte: Dados da pesquisa (2019).

Durante o preenchimento, aos que informaram ter feito o uso de bonificação, foram disponibilizadas duas outras questões: a primeira sobre a utilização da estratégia; e a segunda referente ao seu fator de impacto no aprendizado.

Com relação aos docentes, durante a análise apresentada na Figura 18, verificamos que 19 bonificam os alunos que resolveram questões com maior nível de dificuldade. A resolução de lista de exercícios e a resolução de questão no quadro para a turma também foram apontadas como estratégias, com 15 e 9 indicações respectivamente.

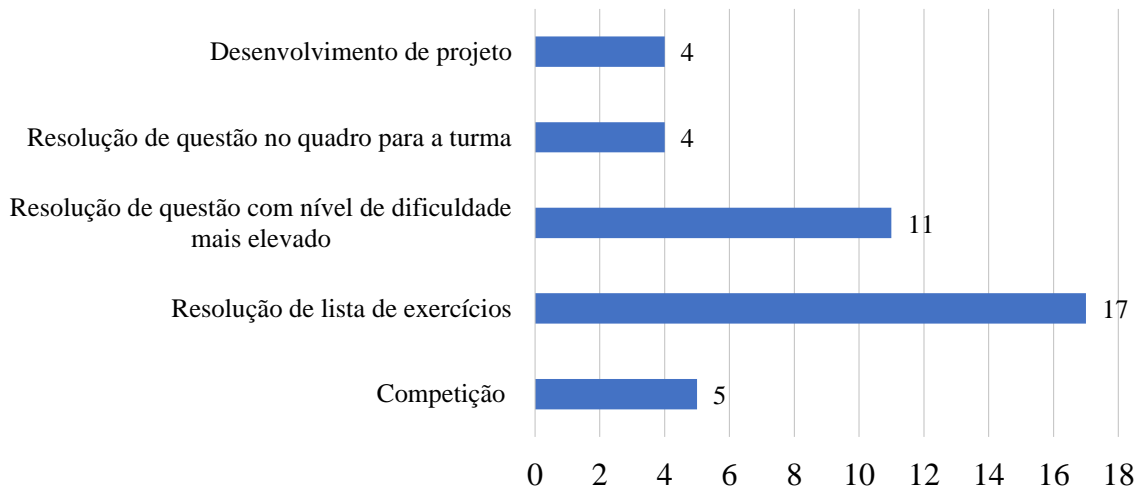
Figura 18 – Formas de bonificação empregadas pelos docentes



Fonte: Dados da pesquisa (2019).

Com relação aos discentes, 17 alunos registraram que os professores se utilizam da estratégia de bonificação na disciplina durante a resolução de lista de exercício, e 11 afirmaram durante a resolução de questão com nível de dificuldade mais elevado (vide Figura 19).

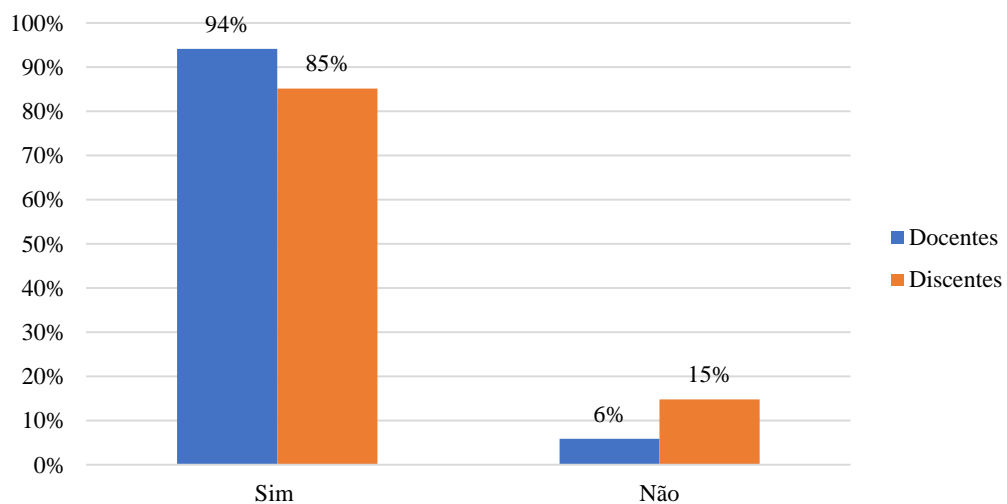
Figura 19 – Formas de bonificação dentro da disciplina identificadas pelos discentes



Fonte: Dados da pesquisa (2019).

Dentre os que informaram a ocorrência de bonificação na disciplina, 94% dos docentes apontaram que os alunos se sentem motivados com essa estratégia e 85% dos discentes afirmaram se sentirem motivados (vide Figura 20).

Figura 20 – Motivação durante o uso da estratégia de bonificação

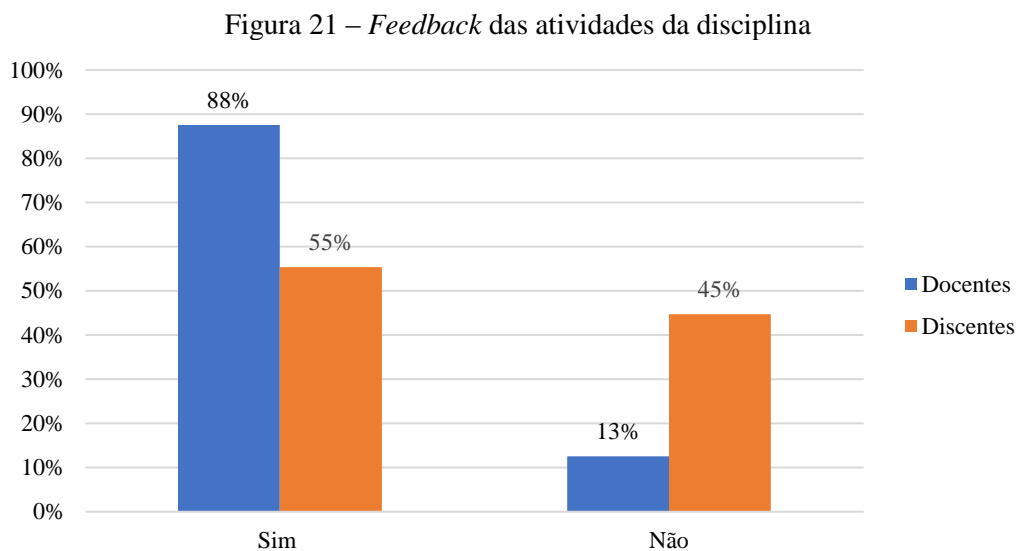


Fonte: Dados da pesquisa (2019).

### 2.1.3.5 *Feedbacks* das atividades

Em relação ao retorno das atividades desenvolvidas durante a disciplina, o *feedback* em tempo hábil permite o progresso de aprendizagem discente. Na literatura, o estudo de Raabe e Silva (2005) disserta sobre a importância do *feedback* para o aprendizado do aluno, porém, os autores constatam que a não devolutiva das atividades em tempo hábil pelos professores é devido a quantidade de alunos nas turmas. Além disso, quando se aborda sobre jogos e *gamificação*, ressaltamos que o *feedback* constante é uma característica fundamental para a sua construção (MCGONIGAL, 2011).

Portanto, percebe-se a importância na devolução das atividades corrigidas, tanto para o ensino e aprendizagem no contexto da disciplina quanto nos jogos. Sobre o *feedback*, verificamos nos resultados dos docentes e discentes um resultado divergente. Enquanto os professores informam, em sua maioria, a emissão dos *feedbacks* (88%), os alunos apontam de maneira balanceada, 55% positivamente e 45% negativamente (vide Figura 21).



Fonte: Dados da pesquisa (2019).

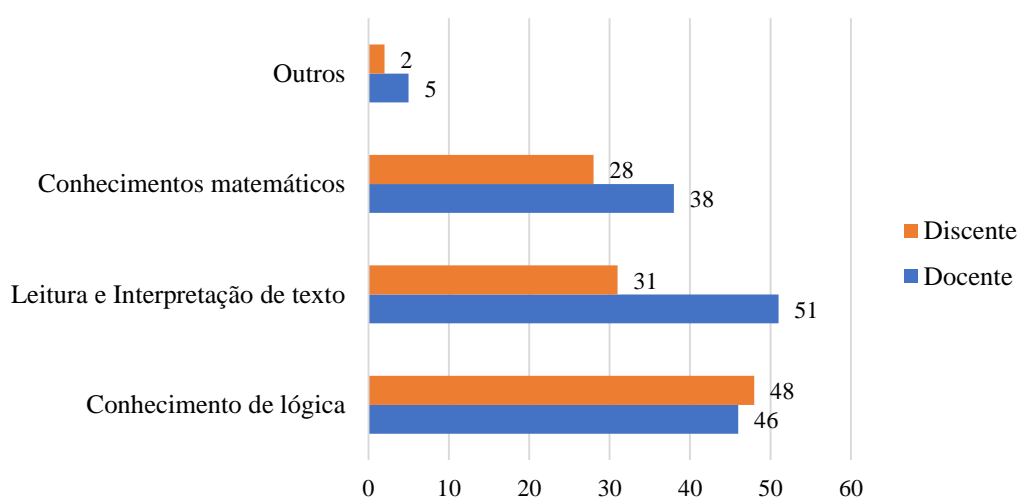
O conflito nos dados, apresentado pelos discentes, reforça a informação presente na literatura de que os *feedbacks* “são difíceis por motivos didático-organizacionais, como por exemplo, a grande quantidade de alunos em turma e o grande número de avaliações sugeridas pela instituição” (RAABE; SILVA, 2005).



### 2.1.3.6 Conhecimentos prévios necessários para a disciplina

Um conjunto de pré-requisitos, adaptados do estudo de Giraffa e Mora (2016) foram apresentados no instrumento aos discentes e docentes, de forma que elencassem quais consideram como necessários ao aprendizado da disciplina Algoritmos ou equivalente. Dentre estes, na pesquisa em tela se destacaram: leitura e interpretação de texto, conhecimentos matemáticos e conhecimento de lógica (vide Figura 22).

Figura 22 – Pré-requisitos necessários para o aprendizado de algoritmos



Fonte: Dados da pesquisa (2019).

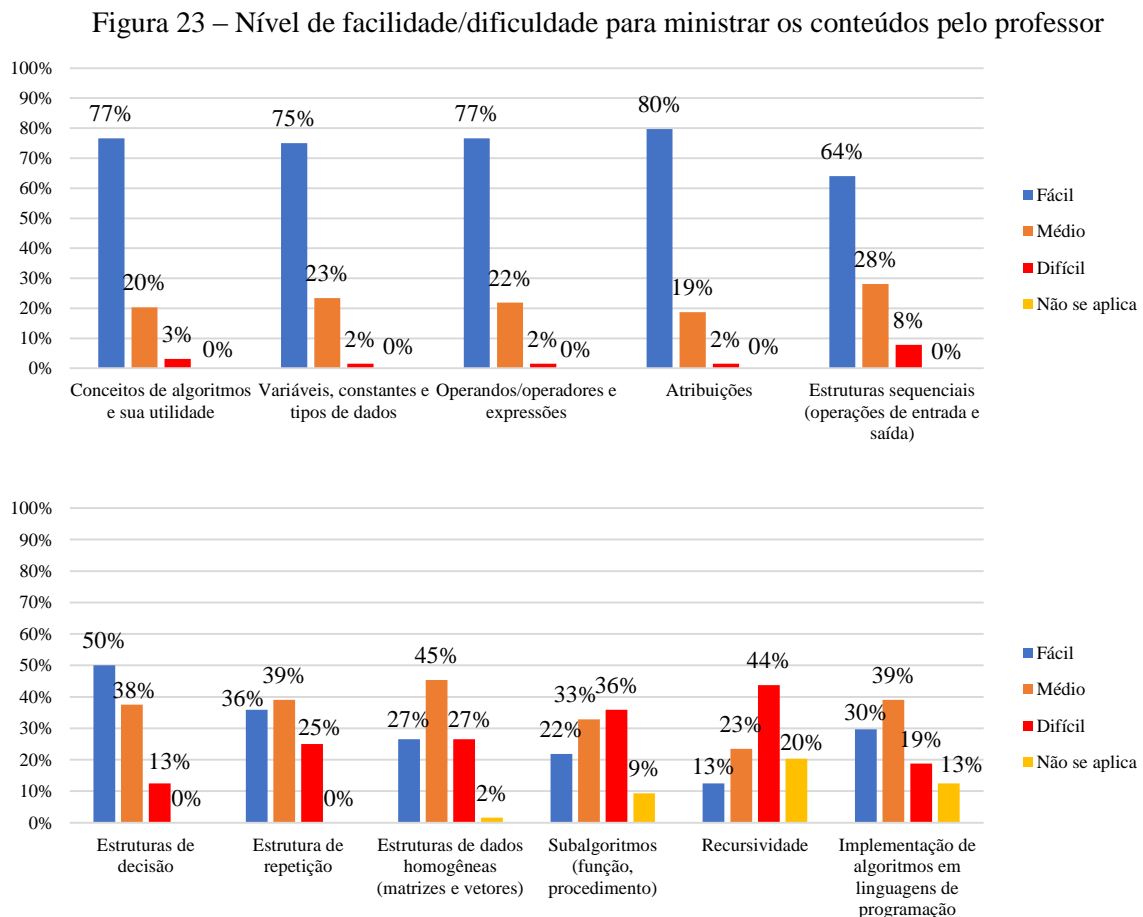
### 2.1.3.7 Conteúdos pertinentes à disciplina

Para esta seção, foram analisadas as respostas dos participantes da disciplina Algoritmos ou equivalente em relação às dificuldades com os conteúdos previstos. Como forma de fazer esta identificação, foi disponibilizada uma escala *likert-3*, com as seguintes opções: “Fácil”, “Médio” e “Difícil”. Também era possível escolher a opção “não se aplica”, pois conforme citado anteriormente, alguns conteúdos podem não estar presentes na ementa do componente curricular de cursos de graduação específicos.

Os conteúdos questionados aos discentes e docentes foram retirados do *currículum base* (SOCIEDADE BRASILEIRA DE COMPUTAÇÃO, 2005) com os ajustes pertinentes ao Projeto Político Pedagógico do Curso de Bacharelado em Ciência da Computação da UFPA (UNIVERSIDADE FEDERAL DO PARÁ, 2010), quais sejam: Conceitos de Algoritmos e sua Utilidade; Variáveis, Constantes e Tipos de Dados; Operandos/Operadores e Expressões; Atribuição; Estruturas Sequenciais (Operação de Entrada e Saída); Estrutura de Decisão;

Estrutura de Repetição; Estrutura de Dados Homogênea (Matrizes e Vetores); Subalgoritmos (Função e Procedimento); Recursividade; Implementação de Algoritmos em Linguagens de Programação.

Observamos, a partir da Figura 23, que, do ponto de vista docente, os conteúdos considerados mais fáceis de ministrar são: “Atribuições” (80%), seguido de “Conceitos de Algoritmos e sua Utilidade” e “Operando/Operadores e Expressões”, ambos com 77% de indicações. Já com relação aos conteúdos considerados mais difíceis de ensinar estão: “Recursividade” (44%), seguido de “Subalgoritmos” (36%) e “Estrutura de Dados Homogênea” (27%).

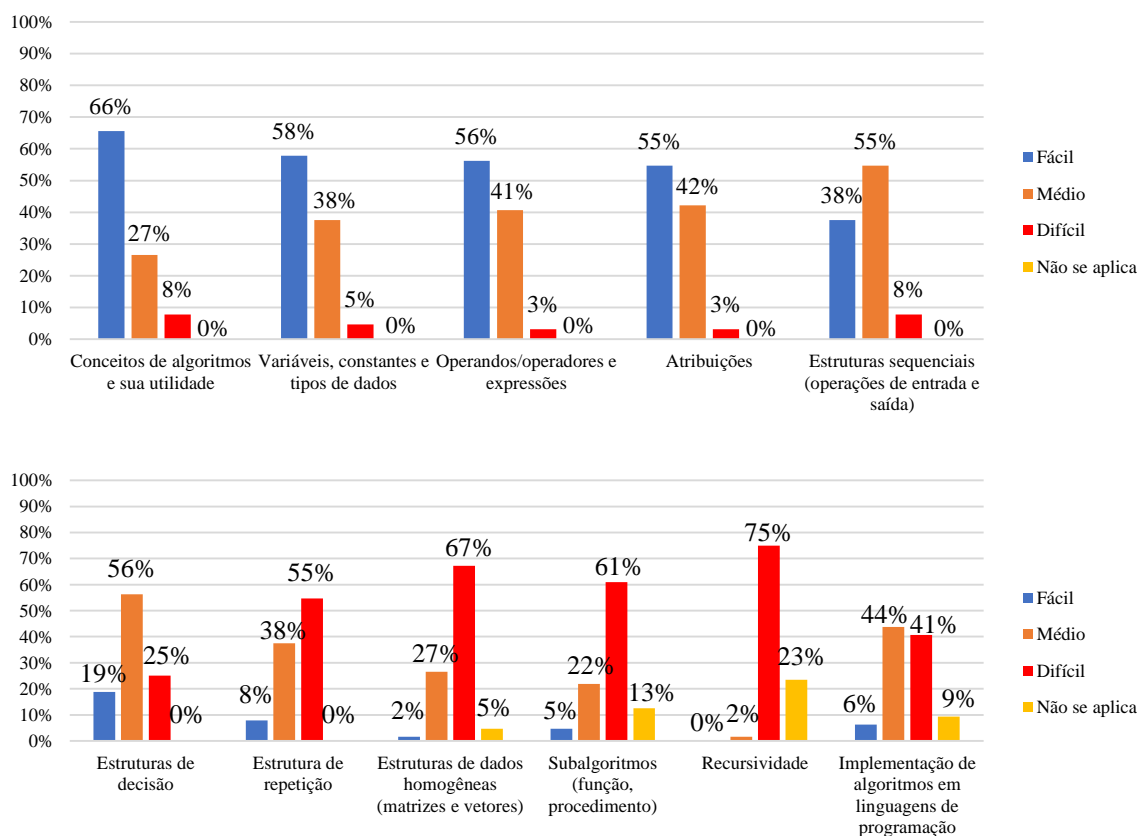


Fonte: Dados da pesquisa (2019).

Ainda sobre os conteúdos, foi também questionado aos participantes sobre o nível de dificuldade/facilidade de assimilação destes pela turma (vide Figura 24). Neste quesito, o conteúdo indicado como de mais fácil assimilação é o “Conceito de Algoritmos e sua utilização” (66%), seguindo de “Variáveis, Constantes e Tipos de Dados” (58%) e “Operando/Operadores e Expressões” (56%). O conteúdo “Recursividade” foi considerado o

mais difícil em relação ao nível de assimilação pela turma (75% dos registros), seguido de “Estrutura de dados homogênea” (67% dos apontamentos) e “Subalgoritmos” (61% de indicações).

Figura 24 – Nível de facilidade/dificuldade de assimilar os conteúdos pela turma na visão do professor

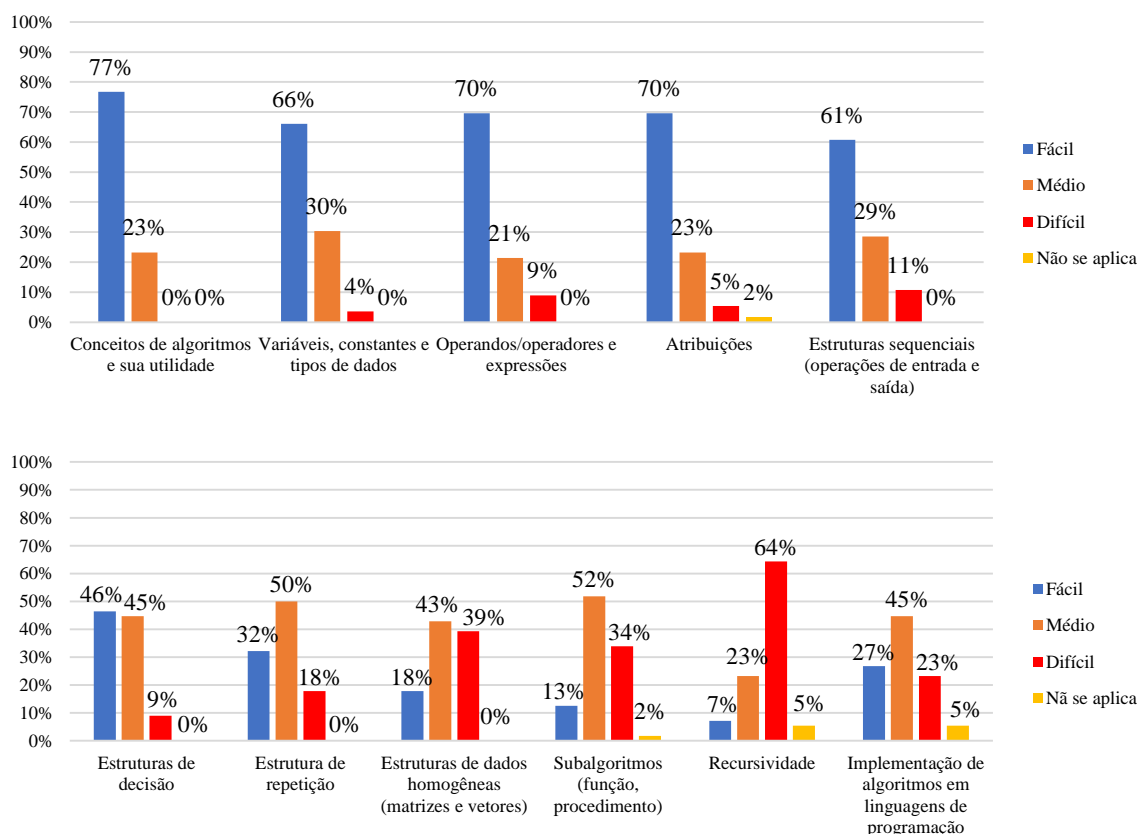


Fonte: Dados da pesquisa (2019).

Os alunos também foram questionados sobre que conteúdos consideram mais fáceis e difíceis de aprender. Segundo a Figura 25, observamos que o “Conceito de algoritmos e sua utilidade” foi considerado, por 77% dos discentes, como o mais fácil, seguido de “Operandos/operadores e expressões” e “Atribuições”, ambos com 70% das respostas. Como terceiro registro mais presente, está o conteúdo de “Variáveis, Constantes e Tipos de Dados”, com um total 66% dos apontamentos.

No grau de difícil aprendido, o conteúdo destacado é sobre “Recursividade”, com 64% das respostas. A segunda unidade de maior complexidade é de “Estrutura de Dados Homogênea”, com 39% dos apontamentos. Em terceiro lugar, com 34% das respostas, está o conteúdo “Subalgoritmos”.

Figura 25 – Grau de facilidade/dificuldade de assimilar os conteúdos na visão discente



Fonte: Dados da pesquisa (2019).

### 2.1.4 Considerações sobre o *survey*

A abordagem didática considerada mais significativa para o ensino e aprendizagem da disciplina Algoritmos ou equivalente é a híbrida. Dentre os recursos apontados pelo *survey*, o mais trabalhado é a lista de exercícios resolvida em sala. Além deste, os participantes da pesquisa indicaram o desenvolvimento de projetos, o qual permite a integração entre os alunos e a aproximação da prática profissional. O formato da divisão dos alunos para as atividades do componente pode ocorrer de forma individual ou em dupla.

Outras percepções sobre a atual realidade da disciplina, relacionadas aos elementos de jogos, foram o emprego da bonificação e o *feedback* das atividades desenvolvidas. Para ambos os elementos, percebemos aceitação e facilitação na aprendizagem dos alunos. Além disso, verificamos o uso de linguagens de programação como recurso para facilitar o aprendizado do discente.

No decorrer da análise dos resultados apresentados no *survey*, tivemos a percepção sobre o conteúdo mais difícil no que tange ao ensino, à assimilação pela turma e ao aprendizado, que é a Recursividade. Ao levar em consideração apenas a aprendizagem, na visão do aluno, e a

assimilação pela turma, na visão do professor, identificamos o segundo mais difícil, como sendo o conteúdo de Estrutura de Dados Homogênea (Matrizes e Vetores).

## 2.2 OBSERVAÇÃO PARTICIPANTE

Como forma de entender a interação entre alunos, professores e o conteúdo em uma realidade prática da disciplina Algoritmos, realizei o Estágio Supervisionado, componente curricular obrigatório da grade curricular do mestrado profissional em ensino, em duas ofertas da disciplina Algoritmos do CBCC da Faculdade de Computação da UFPA. O primeiro Estágio se deu no segundo semestre de 2017 e o segundo no primeiro semestre de 2018.

Além deste objetivo, era importante, também, que as observações realizadas em momentos de aula, pudessem me auxiliar a eliciar os requisitos para trabalhar o *framework gamificado* considerando a realidade e necessidades do curso.

Para atender a estes objetivos, durante o Estágio, foi empregado o procedimento metodológico da observação participante, a partir dos estudos de Creswell (2007) e May (2004).

### 2.2.1 Estrutura da disciplina

A disciplina Algoritmos é um dos componentes curriculares básicos dos cursos de tecnologia de informação (SOCIEDADE BRASILEIRA DE COMPUTAÇÃO, 2005). Na Universidade Federal do Pará, a disciplina possui carga horária de 68 horas, sendo quatro horas semanais, e tem oferta regular no primeiro semestre dos cursos de Bacharelado em Sistemas de Informação e Ciência da Computação. A ementa curricular está presente no Plano Político Pedagógico dos Cursos. Na do CBCC (UNIVERSIDADE FEDERAL DO PARÁ, 2010), por exemplo, segue o Currículo de referência para os cursos de Tecnologia da Informação (SOCIEDADE BRASILEIRA DE COMPUTAÇÃO, 2005) e tem equivalência à disciplina *Fundamental Programming Concepts* do currículo proposto pela Association for Computing Machinery/ Institute of Electrical and Electronic Engineers (2013), como já informado no *survey*.

### 2.2.2 Apresentando o estudo

Anterior ao início da observação e como forma de sistematizar a coleta dos dados, definimos um roteiro (Apêndice C), porém, à medida que as aulas iam se desenvolvendo, outros registros foram sendo acrescentados ao roteiro original.

O papel adotado por mim, durante a coleta de dados foi o de observador participante. Fui apresentado no primeiro dia de aula, pela professora, para os estudantes, como estagiário. Portanto, a participação para esse estudo é do tipo conhecida (LEININGER, 1985). Esta apresentação ocorreu nas duas ofertas da disciplina pela Faculdade de Computação da UFPA, no curso de Bacharelado em Ciência da Computação, nos períodos de 25 de setembro de 2017 a 31 de janeiro de 2018 e 19 de março de 2018 a 06 de junho de 2018, em que atuei como estagiário e observador.

Durante a primeira oferta, a disciplina estava sendo ministrada às segundas e quartas-feiras no horário de 13h00 às 14h40. A segunda oferta ocorreu nos mesmos dias da semana, no horário de 14h40 às 16h30. Além do ambiente presencial de sala de aula, foi utilizada como suporte a plataforma Moodle instalada na Universidade Federal do Pará.

### **2.2.3 A experiência vivida**

A minha experiência foi um grande diferencial na minha formação, pois ainda não havia tido a oportunidade de realizar pesquisa qualitativa. Acredito que este fato se deva por eu ser um pesquisador iniciante, ou por meu curso de graduação ser da área de exatas, a qual, em geral, adota a pesquisa quantitativa. Assim, precisava fazer uso de alguma estratégia para me ajudar a superar essa primeira barreira para o andamento da pesquisa. Meus orientadores me auxiliaram no processo, indagando a necessidade e a justificativa para o desenvolvimento de uma observação participante, e como primeira resposta a esses questionamentos, definimos juntos um objetivo de pesquisa para esse procedimento metodológico e os dados aos quais eu deveria coletar.

O objetivo definido foi compreender como ocorre a interação entre docente, discentes e o conteúdo durante as aulas da disciplina Algoritmos, ofertada pela Faculdade de Computação, da Universidade Federal do Pará. Em nossas reuniões, resolvemos definir um formulário semiestruturado (Apêndice C), o qual serviria de guia para observar e coletar as informações. O instrumento desenvolvido foi avaliado por um pedagogo, o qual contribuiu com algumas sugestões, aceitas e disponibilizadas na versão final. O processo relatado é descrito por May (2004) como a fase de acesso ao campo.

Durante as primeiras semanas em sala de aula, estive atento ao questionário desenvolvido. Porém, após conversas com minha orientadora e professora da disciplina, foi possível identificar que a observação participante não deveria estar limitada ao instrumento. Com isso, passei a buscar também informações sobre o cenário, os personagens, a interação

entre eles e as nuances do processo de ensino e aprendizagem. Buscava, ainda, analisar e identificar possíveis elementos de jogos trabalhados em sala de aula.

Inicialmente observei como era o tratamento da professora para com os alunos, assim como o inverso. Tinha como objetivo aprender práticas e estratégias para ensinar os discentes no desenvolvimento de códigos e no entendimento dos algoritmos. Algo bastante enriquecedor durante este processo, foi o desenvolvimento de minhas habilidades para a docência, nos períodos em que assumia a sala de aula ou então acompanhava os discentes na resolução dos exercícios propostos em sala, com prévia autorização e supervisão da docente da disciplina. As atividades desenvolvidas por mim no decorrer do Estágio Supervisionado e na observação participante permitiram-me adquirir a confiança necessária para coordenar atividades e, por consequência, conduzir o experimento com o produto da minha pesquisa de mestrado.

Vale destacar que nas duas observações fiquei encarregado da proposição do plano de ensino, em que estava disponível: o objetivo da disciplina, as competências e habilidades, a ementa, o conteúdo programático, os recursos didáticos e materiais necessários, a metodologia e a programação. Por meio do objetivo da disciplina, foi possível identificar quais as metas para o desenvolvimento de uma abordagem *gamificada*.

#### **2.2.4 Resultados e discussão**

Os resultados e discussões dos dados coletados por mim, por meio da observação participante, serão descritos nas próximas subseções tendo como base quatro perspectivas: (i) o cenário: local onde ocorreu a observação; (ii) os atores: quem interagiu na disciplina; (iii) os eventos: o que aconteceu em sala de aula presencial e virtual; e (iv) o processo: qual a natureza dos acontecimentos dentro do ambiente do componente curricular (CRESWELL, 2007).

##### **2.2.4.1 Cenário**

O cenário em que as disciplinas foram desenvolvidas, eram laboratórios de informática da Faculdade de Computação, situados no Instituto de Ciências Exatas e Naturais da UFPA, com a seguinte estrutura: um quadro branco, computadores suficientes para os discentes e para a docente, monitor e *Datashow*.

A situação permitia a implementação dos algoritmos em linguagem de programação e, por consequência, os discentes conseguiam visualizar as codificações desenvolvidas. Na

ocasião das duas ofertas da disciplina, a linguagem de programação adotada pela professora foi a Pascal.

A aplicação das provas teóricas acontecia em salas de aula da Faculdade, com as cadeiras dos discentes distribuídas no espaço em formato de fileiras. Nesse dia, era proibido o uso de aparelhos eletrônicos e os discentes conversarem entre si. A prova era feita no papel e nos era permitido responder dúvidas sobre o entendimento das questões, sem apontar a solução.

Como recurso auxiliar, contou-se com o apoio de uma sala virtual disponibilizada na plataforma Moodle, no domínio da UFPA. Nesta sala, os alunos podiam acessar os materiais de aula, listas de exercícios, gabaritos, assim como realizar atividades disponibilizadas pela professora. As listas de exercício enviadas via plataforma, eram corrigidas pelo monitor da disciplina, o qual emitia uma nota e o *feedback* sobre como estava o andamento do discente em relação a resolução.

#### 2.2.4.2 Os atores

Os atores presentes na observação participante foram os discentes matriculados na disciplina, a professora ministrante, um monitor e eu, como estagiário. A professora, possuía a função de ensinar algoritmos, explicando aos alunos os conteúdos da disciplina, bem como auxiliar na implementação dos códigos. Ela é professora associada vinculada à Faculdade de Computação e Coordenadora do Programa de Pós-Graduação Criatividade e Inovação em Metodologias do Ensino Superior (PPGCIMES) da UFPA. Tem formação base em Bacharelado em Matemática na modalidade de Informática, com mestrado em Engenharia de Sistemas e doutorado em Ciências – Desenvolvimento Socioambiental. A professora possui experiência com a disciplina Algoritmos há mais de 20 anos.

A função do monitor era a de auxiliar os alunos nas atividades desenvolvidas em sala, além de corrigir os exercícios enviados pela plataforma Moodle. Em cada uma das ofertas da disciplina observadas tivemos diferentes pessoas nessa função, os quais já tinham sido alunos da disciplina na Faculdade de Computação. Na primeira interação, o monitor era discente do Curso de Bacharelado em Ciência da Computação. Já na segunda, era estudante do curso de Bacharelado em Sistemas de Informação.

Eu ocupava a função de estagiário nas duas ofertas da disciplina, com as seguintes responsabilidades: proposição de listas de exercício; sugerir questões para as provas; preparar programação para os dias de aula; auxiliar os discentes nas atividades desenvolvidas em sala; e conduzir as aulas quando necessário, em geral nos dias de resolução de exercícios. Todas as



tarefas descritas eram supervisionadas e validadas pela professora antes de serem submetidas aos discentes. Como já citado, possuo formação base de Bacharel em Sistemas de Informação, com Especialização em Gestão da Tecnologia da Informação e, atualmente, sou discente do Mestrado Profissional em Ensino do PPGCIMES da UFPA.

Por último, os discentes tinham como função interagir com os outros personagens e o conteúdo, com o objetivo de aprender a desenvolver algoritmos. Na primeira observação estavam presentes 14 discentes repetentes da disciplina, que também estavam retidos em outras disciplinas. Na segunda, participaram das aulas, 20 discentes, sendo, três, repetentes. A maioria dos discentes presentes em sala era do Curso de Bacharelado em Ciência da Computação.

#### 2.2.4.3 Os eventos

Em minha pesquisa foram categorizados os eventos da seguinte maneira: os acontecimentos físicos e virtuais. Os eventos virtuais eram via sala de aula estendida na plataforma Moodle. Os principais eventos que ocorriam por meio da sala de aula estendida era o envio das listas de exercício, em que podíamos perceber, em parte, o interesse dos alunos pelo aprendizado de algoritmos, pois, a plataforma permitia aos alunos enviar suas atividades, quando solicitados. Além disso, o monitor emitia um *feedback* sobre a correção das questões enviadas. Quando o monitor indicava questões que haviam sido difíceis de serem desenvolvidas nas listas, a professora utilizava do espaço de sala de aula para explicar, melhor, a forma como resolver e, após a explanação, acompanhava a resolução pelos discentes com o monitor e o estagiário.

Durante a segunda oferta da disciplina, por meio das listas, o monitor percebeu - embora os discentes estivessem desenvolvendo seus algoritmos - que não estavam se utilizando de boas práticas em programação, tais como, por exemplo, uso de indentação e comentários nos códigos produzidos. Para amenizar esta situação, foi realizado uma conversa entre o monitor, a professora e eu, para que fosse feito um reforço, em sala, pelo monitor, com a minha supervisão, sobre boas práticas de programação de algoritmos.

Os eventos físicos ocorriam durante as aulas, por meio da interação entre os personagens, como forma de propiciar o ensino e a aprendizagem de algoritmos. Destaco, nas duas ofertas, a escolha da abordagem de ensino híbrida pela professora, com uso principalmente de listas de exercícios, para serem resolvidas em sala ou em casa, como recurso de aprendizagem. As listas de exercício eram compostas por questões sobre o desenvolvimento de

algoritmos e realização de testes de mesa para relatar o resultado da execução do algoritmo disponibilizado.

A aula com abordagem híbrida geralmente iniciava com um conteúdo novo da disciplina. Nestes dias, o conteúdo era explanado nos 20 primeiros minutos das aulas e, em seguida, havia a prática por parte dos alunos, com exercícios sobre o assunto. Em outros dias, a aula era apenas prática, com questões para os alunos desenvolverem com o software Pascalzim, ou então, para elaborarem no papel como é o caso do teste de mesa. Importante destacar que a professora também, durante as aulas, promovia vários desafios com os alunos valendo ponto extra.

No decorrer das aulas, foi possível identificar comportamentos negativos e positivos dos discentes, os quais poderiam servir de base para penalizações ou reforços em uma abordagem *gamificada* da disciplina. Como comportamentos negativos, identifiquei: (i) faltar as aulas, fazendo com que o discente não tivesse contato com o conteúdo ministrado; (ii) atraso, uma vez que o conteúdo era ministrado, em geral, nos primeiros 20 minutos da aula; e (iii) distração durante a aula, tais como: uso do celular, conversa entre discentes e uso do computador sem ser para programação ou acesso à plataforma Moodle.

Como comportamentos positivos percebi: (i) a presença constante em sala de aula, pois o discente tinha contato com os personagens e conteúdo da disciplina; e (ii) participação em sala, pois demonstrava o interesse do discente pelo conteúdo da disciplina.

Durante os momentos de prática dos alunos, a professora solicitava ao monitor e a mim, que caminhássemos pela turma para identificar algum aluno que estivesse com problema, para auxiliar com dicas para a resolução das questões, configurando-se como a prática de *coaching*.

No processo de auxílio aos discentes, quando se observava que uma dica era passada para mais de duas pessoas, essa informação era transmitida para a turma inteira. Dentro da coleta dos dados foi possível identificar, também, a prática de *mentoring*, pois alguns discentes tinham dificuldade para o desenvolvimento mesmo após as dicas. Então, a pessoa que estava acompanhando o discente, no caso professora ou monitor, codificava ou iniciava a interação do teste de mesa junto com ele.

#### 2.2.4.4 O processo

O processo, segundo Creswell (2007), é a natureza evolutiva dos atores em relação aos eventos, no caso de ensino e aprendizagem de algoritmos. Nesse ponto, foi possível observar a progressão dos discentes em relação ao conteúdo da disciplina. Tanto os monitores quanto eu

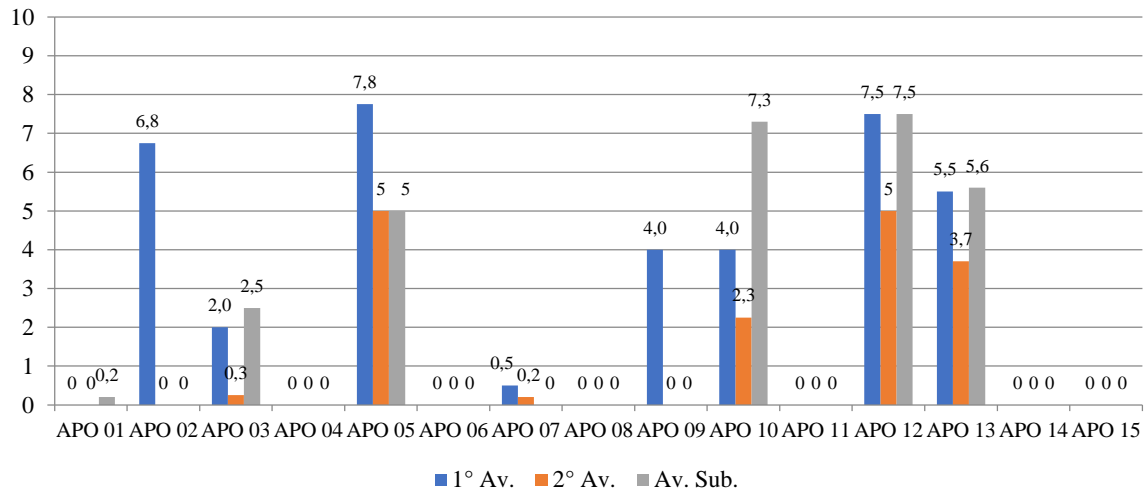
íamos adquirindo confiança para auxiliar os alunos com dicas, codificações e explicações de questões no quadro.

A disciplina tem como característica conter conteúdos cumulativos, ou seja, conteúdos bases que são necessários para o entendimento e progressão na assimilação de conteúdos médios e difíceis. Portanto, é de suma importância o discente ter as bases na disciplina. Segundo o Plano Político Pedagógico do CBCC (UNIVERSIDADE FEDERAL DO PARÁ, 2010), o aluno perpassa pelos seguintes conteúdos: Conceitos de algoritmos e sua utilidade; Variáveis, constantes e tipos de dados; Operandos/operadores e expressões; Atribuição; Estruturas sequenciais (operação de entrada e saída); Estrutura de decisão; Estrutura de repetição; Estrutura de dados homogêneas (matrizes e vetores); Subalgoritmos (função, procedimento); Recursividade; Implementação de algoritmos em linguagens de programação. No entanto, nas duas ofertas observadas, a professora fez a escolha de ministrar o conteúdo até Estrutura de dados homogêneos, porque os tópicos mais avançados seriam mais bem visualizados com o auxílio das implementações em linguagens de programação nas futuras disciplinas de desenvolvimento de software.

A disciplina foi estruturada para uma divisão de três provas avaliativas, além de uma substitutiva ao final. A professora, para compor a nota final dos alunos, fez uso, ainda do desempenho nas listas de exercício para pontos extras, caso necessário.

A Figura 26 apresenta o desempenho dos alunos presentes na primeira observação. O acrônimo APO significa aluno primeira observação. Ao analisar a progressão dos alunos no quesito nota, percebemos que os alunos de números 02 e 10 aumentaram suas notas, enquanto de números 12 e 13 mantiveram a mesma. Já os alunos que tiveram queda no desempenho são os de números 05, 07 e 09. Os seguintes discentes zeraram as atividades avaliativas: 01, 04, 06, 08, 11, 14 e 15. Do total de 15 alunos, sete foram reprovados, sendo três por frequência, pois não estavam presentes em sala em 75% das aulas ministradas (porcentagem de frequência obrigatória descrita dentro do regimento interno da UFPA).

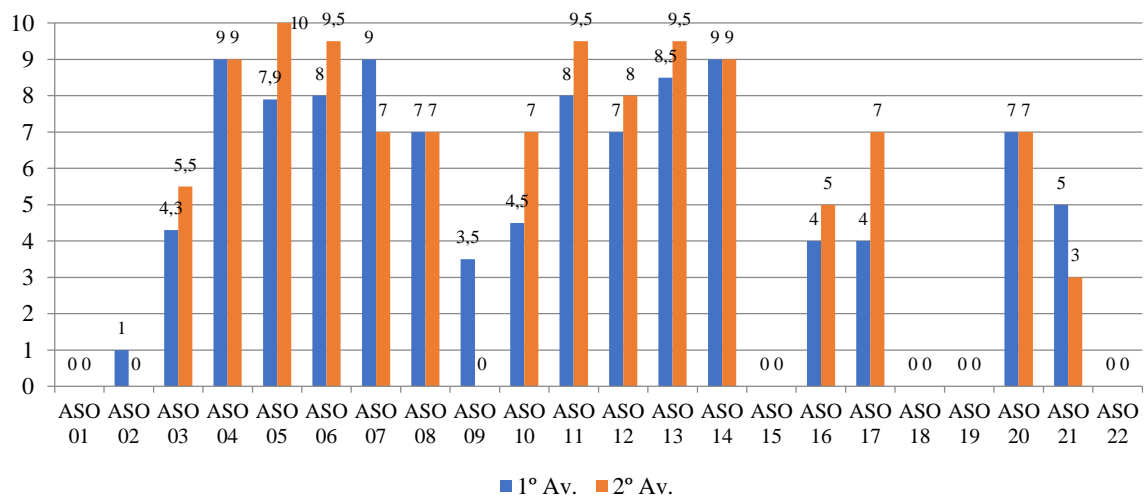
Figura 26 – Nota dos alunos da primeira observação



Fonte: Dados da pesquisa (2019).

Para a segunda observação, utilizamos o acrônimo ASO o qual significa aluno segunda observação. Os dados da Figura 27 indicam que os alunos de número 03, 05, 06, 10, 11, 12, 13, 16 e 17, obtiveram progresso ascendente em suas notas e os discentes 02, 07, 09 e 21, tiveram desenvolvimento em decréscimo. Já os estudantes de número 04, 08, 14 e 20 mantiveram as suas notas. Nessa segunda interação, alguns personagens zeraram as notas durante as atividades, são eles: 01, 15, 18, 19 e 22. Quatro alunos reprovados tiveram frequência inferior ao que a UFPA, no seu regimento interno, solicita para a aprovação na disciplina.

Figura 27 – Nota dos alunos coletadas da segunda observação



Fonte: Dados da pesquisa (2019).

As notas das duas observações apresentam uma das realidades da disciplina e dentro delas podemos identificar situações já relatadas no texto, pois a primeira observação foi realizada em uma turma com a sua maioria de repetentes. Já na segunda tínhamos discentes no primeiro contato com a disciplina. Porém, mesmo com o primeiro contato, apontamos a dificuldade no processo de ensino e aprendizagem, pois 12 alunos em pelo menos uma ou nas duas avaliações não alcançaram a nota mínima para passar.

### **2.2.5 Considerações sobre a observação participante**

A minha experiência com a observação participante foi de extrema importância para o meu percurso no mestrado como pesquisador. Por meio desta vivência, tive oportunidade de enxergar a disciplina por um ângulo diferente, além de me possibilitar adquirir a confiança necessária para o desenvolvimento e condução do experimento ao qual precisava para o uso do produto da minha dissertação de mestrado. Ressalto, também, que proporcionou experienciar a docência no ensino de graduação, o que foi extremamente rico para a minha formação.

A seguir descrevo a sistematização de minhas observações, as quais serviram de base como requisitos para o desenvolvimento do meu produto de mestrado:

- A prática de uma conversa inicial da professora com a turma.
- O formato para explicar os conteúdos da disciplina, bem como a resolução de questões.
- Os objetivos de ensino e aprendizagem trabalhados na disciplina.
- O uso de pontuação extra durante as aulas, bem como bonificar as listas de exercícios para casa, apresentando *feedback* para os discentes sobre sua aprendizagem.
- A ocorrência de comportamentos dos discentes considerados negativos que podem gerar penalizações: atraso; falta, conversa em sala; uso do computador para outro propósito; e uso do celular retirando a atenção na aula.

- A ocorrência de comportamentos positivos dos discentes que podem gerar bonificações: presença constante; pontualidade; participação e interação entre os personagens com o intuito de gerar conhecimento.
- A atribuição de pontos extras durante a disciplina.
- A ocorrência de *mentoring*.
- O costume dos alunos em utilizar a plataforma Moodle e a linguagem de programação Pascal.

### 2.3 FRAMEWORK PROJETO SPIDER

O Projeto *Software Process Improvement: Development and Research* (SPIDER), coordenado pelo professor Dr. Sandro Ronaldo Bezerra Oliveira, institucionalizado no Instituto de Ciências Exatas e Naturais (ICEN) da UFPA, tem como foco principal

apresentar um levantamento das ferramentas de software livre com características adequadas para possibilitar a criação de produtos de trabalhos (artefatos) derivados dos resultados esperados descritos nos objetivos dos processos do modelo MPS.BR - Melhoria do Processo de Software Brasileiro, mantido pela SOFTEX - Associação para Promoção da Excelência do Software Brasileiro, e das práticas específicas descritas nos objetivos das áreas de processo do modelo CMMI – Capability Maturity Model Integration, mantido pelo SEI – Software Engineering Institute, o que evidenciam a implementação do programa da qualidade organizacional (Projeto SPIDER, 2009).

O referido projeto conta com uma linha de pesquisa voltada para abordagens de ensino *gamificada* em engenharia de Software na UFPA, nos cursos oferecidos pela Faculdade de Computação (CBCC e CBSI), por meio do Programa de Pós-graduação em Ciência da Computação.

O *framework*, em desenvolvimento pelos membros dessa linha de pesquisa do projeto, já é objeto de duas publicações de relatos da experiência *gamificada* para ensino. A primeira intitulada “*Gamification and Evaluation of the Use the Agile Tests in Software Quality Subjects: the Application of Experiments*” (ELGRABLY; OLIVEIRA, 2018), retrata o ensino de testes ágeis de *software* via *gamificação*. Já o trabalho de Santos e Oliveira (2018), intitulado “*Gamificação* como Ferramenta de Suporte de Apoio ao Ensino da Técnica Análise de Pontos

por Função em uma Turma de Pós-Graduação: Um Estudo de Caso”, discorre sobre um relato de ensino *gamificado* para turma de pós-graduação sobre análise de pontos por função.

Para o embasamento teórico das salas de aula *gamificadas*, descritas nos relatos de experimentos, foi utilizado o trabalho de McGonigal (2011), o qual define jogos por meio de quatro características básicas: metas, regras, *feedback* e participação voluntária. Os seguintes núcleos propostos no trabalho de Chou (2013) também são empregados, para justificar os elementos de elementos de jogos: significado épico e chamado, desenvolvimento e realização, empoderamento da criatividade e *feedback*, propriedade e posse, e influência social e pertencimento.

A forma como as atividades ocorreram em sala nas duas disciplinas, assim como o exemplo das planilhas e framework, serviram de base para a concepção da primeira proposta de produto que será abordada adiante. As salas de aula *gamificadas*, desenvolvidas para os dois experimentos, tinham como núcleo o elemento competição. Além disso, as práticas adotadas para sala de aula eram: (i) aula teórica: em que era exposto o conteúdo da disciplina; (ii) Dojo do tipo *Randori*: uma atividade colaborativa em ambiente seguro de aprendizagem. Os personagens (piloto e copiloto) interagem com a plateia demonstrando a solução adotada para a atividade proposta; (iii) Laboratório de programação (LAB): prática pareada com acesso ao computador para resolução de questões desafio sobre o conteúdo *gamificado*; (iv) teste teórico: prova formulada com questões de concurso com o intuito de analisar o conhecimento dos discentes adquiridos por meio da *gamificação* do conteúdo; e (v) aula de *feedback*: prática para coletar as opiniões dos discentes sobre as aulas e recursos trabalhados no *framework*.

### 3 FRAMEWORK GAMIFICADO VERSÃO 01

O produto da pesquisa de mestrado desta dissertação materializou-se em um *framework gamificado* para o ensino-aprendizado da disciplina Algoritmos ou equivalente, composto por dois documentos base: plano de ensino e planilha *gamificada*. As bases que nos permitiram desenvolver a lista de requisitos para a concepção do produto, foram: o *survey* (Seção 2.1); a observação participante (Seção 2.2); os modelos de *framework* já adotados no projeto SPIDER (Seção 3.3); e uma revisão informal da literatura (Seção 1.2). A primeira versão da proposta, bem como a forma como foi avaliada por especialistas estão descritas nas seções deste capítulo.

#### 3.1 REQUISITOS

Os requisitos utilizados como base para o desenvolvimento do *framework gamificado* estão sistematizados no Quadro 6, Quadro 7, Quadro 8 e Quadro 9. Cada quadro possui as seguintes colunas: (i) itens do *framework*, em que é apresentado os elementos de jogos; (ii) insumo, descreve o elemento ou como ele é trabalhado dentro do *framework*; e (iii) origem, em que é explicitada a fonte de dados que compõe o requisito.

No Quadro 6 são apresentados os elementos de jogos adotados no *framework* referente às características da *gamificação*.

Quadro 6 – Requisitos do *framework* referente às características (Continua...)

Itens do <i>framework</i>	Insumo	Origem
<b>Conteúdo</b>	<ul style="list-style-type: none"> <li>Definição do conteúdo de Estrutura de Dados Homogêneos (<i>Array</i>).</li> </ul>	<ul style="list-style-type: none"> <li><i>Survey</i>.</li> <li>Observação participante.</li> </ul>
<b>Metas</b>	<ul style="list-style-type: none"> <li>Objetivos de ensino.</li> <li>Objetivos de aprendizagem.</li> </ul>	<ul style="list-style-type: none"> <li>Observação participante.</li> <li><i>Framework</i> Projeto SPIDER.</li> </ul>
<b>Regras</b>	<ul style="list-style-type: none"> <li>Aquisição de pontos por meio de resolução de desafios.</li> <li>Concessão de <i>avatar</i> diário e geral da <i>gamificação</i>.</li> <li>Concessão e/ou retirada de bônus.</li> <li>Transformação de bônus em recurso estrela.</li> <li>Conversão de pontos e estrelas em notas.</li> <li>Definição das abordagens/práticas trabalhadas em sala de aula, bem como fora dela, no caso a lista de exercício para casa.</li> </ul>	<ul style="list-style-type: none"> <li>Revisão informal da literatura.</li> <li><i>Survey</i>.</li> <li>Observação participante</li> <li><i>Framework</i> Projeto SPIDER.</li> </ul>

Fonte: Elaboração própria (2019).



Quadro 6 – Requisitos do *framework* referente às características (Conclusão)

Itens do <i>framework</i>	Insumo	Origem
<b>Feedback</b>	<ul style="list-style-type: none"> <li>• Devolutiva da correção das listas de exercício para os discentes.</li> <li>• Apresentação do <i>ranking</i> no início das aulas.</li> <li>• Disponibilização dos gabaritos das atividades.</li> </ul>	<ul style="list-style-type: none"> <li>• <i>Survey</i>.</li> <li>• Observação participante.</li> <li>• <i>Framework</i> Projeto SPIDER.</li> </ul>
<b>Personagens</b>	<ul style="list-style-type: none"> <li>• Definição dos personagens: mentora, <i>master</i>, <i>coach</i>, <i>juiz/scoreplay</i> e <i>players</i>.</li> </ul>	<ul style="list-style-type: none"> <li>• Revisão informal da literatura.</li> <li>• Observação participante.</li> </ul>

Fonte: Elaboração própria (2019).

No Quadro 7 são apresentados os requisitos referentes às práticas que foram adotadas no decorrer da abordagem *gamificada*.

Quadro 7 – Requisitos do *framework* referentes às práticas

Itens do <i>framework</i>	Insumo	Origem
<b>Conversa inicial</b>	<ul style="list-style-type: none"> <li>• Forma escolhida para informar aos alunos sobre como seria a condução das aulas com a abordagem <i>gamificada</i>, visto que as aulas iriam ser ministradas de outra forma.</li> </ul>	<ul style="list-style-type: none"> <li>• Observação participante.</li> <li>• <i>Framework</i> Projeto SPIDER.</li> </ul>
<b>Aula teórica e prática</b>	<ul style="list-style-type: none"> <li>• Foi escolhida para conduzir as aulas, visto ter sido apontada por discentes e docentes como o método mais significativo para o ensino-aprendizado de algoritmos.</li> </ul>	<ul style="list-style-type: none"> <li>• <i>Survey</i>.</li> </ul>
<b>Dojo</b>	<ul style="list-style-type: none"> <li>• Uma dentre as práticas diferenciadas escolhidas para exercitação, por incentivar a programação em dupla, envolvendo a turma inteira, em ambiente seguro.</li> </ul>	<ul style="list-style-type: none"> <li>• <i>Survey</i>.</li> <li>• <i>Framework</i> Projeto SPIDER.</li> </ul>
<b>Laboratório de programação (LAB)</b>	<ul style="list-style-type: none"> <li>• Uma dentre as práticas escolhidas para exercitação, por incentivar a programação em dupla e permitir a obtenção de um retorno sobre o aprendizado do conteúdo ministrado.</li> </ul>	<ul style="list-style-type: none"> <li>• <i>Survey</i>.</li> <li>• <i>Framework</i> Projeto SPIDER.</li> </ul>
<b>Prova teórica</b>	<ul style="list-style-type: none"> <li>• Forma escolhida para avaliar a aprendizagem individual do conteúdo.</li> </ul>	<ul style="list-style-type: none"> <li>• Observação participante.</li> <li>• <i>Framework</i> Projeto SPIDER.</li> </ul>
<b>Aula de <i>feedback</i></b>	<ul style="list-style-type: none"> <li>• Forma escolhida para coletar opiniões dos alunos em relação ao <i>framework</i>.</li> </ul>	<ul style="list-style-type: none"> <li>• <i>Framework</i> Projeto SPIDER.</li> </ul>

Fonte: Elaboração própria (2019).

O Quadro 8 apresenta os elementos de jogos adotados no *framework* como recompensas aos discentes.

Quadro 8 – Requisitos do *framework* referente às recompensas

Itens do <i>framework</i>	Insumo	Origem
<b>Bônus</b>	<ul style="list-style-type: none"> <li>• Uso de bonificação para apoiar o comportamento. Quesitos avaliados: presença, participação, sugestão e pergunta.</li> </ul>	<ul style="list-style-type: none"> <li>• Observação participante.</li> <li>• <i>Framework</i> Projeto SPIDER.</li> </ul>
<b>Penalidades</b>	<ul style="list-style-type: none"> <li>• Emprego de penalizações por comportamento: falta, atraso, uso do celular e uso do computador.</li> </ul>	<ul style="list-style-type: none"> <li>• Observação participante.</li> <li>• <i>Framework</i> Projeto SPIDER.</li> </ul>
<b>Recurso estrela</b>	<ul style="list-style-type: none"> <li>• As bonificações adquiridas pelos discentes no decorrer de cada aula são convertidas em recurso estrela.</li> </ul>	<ul style="list-style-type: none"> <li>• Revisão informal da literatura.</li> </ul>
<b>Pontos</b>	<ul style="list-style-type: none"> <li>• Atribuição de pontos pela resolução de desafios durante a disciplina.</li> </ul>	<ul style="list-style-type: none"> <li>• Observação participante.</li> <li>• <i>Framework</i> Projeto SPIDER.</li> </ul>
<b>Medalha</b>	<ul style="list-style-type: none"> <li>• Identificação das três primeiras posições em relação aos elementos: pontos e estrela.</li> <li>• Pontos, Medalhas <i>Power</i> e Participativo, Recurso estrela.</li> </ul>	<ul style="list-style-type: none"> <li>• <i>Framework</i> projeto SPIDER</li> </ul>
<b>Colaboração</b>	<ul style="list-style-type: none"> <li>• Identificado como perfil e comportamento adotado pelo discente na disciplina, portanto ele será bonificado por esse comportamento.</li> </ul>	<ul style="list-style-type: none"> <li>• <i>Survey</i>.</li> </ul>
<b>Avatar</b>	<ul style="list-style-type: none"> <li>• Perfil: analista, programador, aprendiz e amador.</li> </ul>	<ul style="list-style-type: none"> <li>• <i>Survey</i>.</li> </ul>

Fonte: Elaboração própria (2019).

O Quadro 9 apresenta os elementos de jogos adotados como os requisitos do *framework* referente aos recursos e estratégias de ensino utilizados no decorrer da *gamificação*.

Quadro 9 – Requisitos do *framework* referentes aos recursos e estratégias de ensino (Continua...)

Itens do <i>framework</i>	Insumo	Origem
<b>Lista de desafios</b>	<ul style="list-style-type: none"> <li>• Uso de lista com questões para exercitação em sala do conteúdo.</li> </ul>	<ul style="list-style-type: none"> <li>• <i>Survey</i>.</li> </ul>
<b>Lista de exercício para casa</b>	<ul style="list-style-type: none"> <li>• Uso de lista com questões para serem resolvidas fora da sala de aula, por ser uma prática adotada por professores e indicada por alunos como efetiva para o aprendizado de programação.</li> </ul>	<ul style="list-style-type: none"> <li>• <i>Survey</i>.</li> </ul>
<b>Coaching</b>	<ul style="list-style-type: none"> <li>• Auxílio dos outros personagens com dicas para os jogadores na resolução dos desafios.</li> </ul>	<ul style="list-style-type: none"> <li>• Revisão informal da literatura.</li> <li>• Observação participante.</li> </ul>

Fonte: Elaboração própria (2019).

Quadro 9 – Requisitos do *framework* referentes aos recursos e estratégias de ensino (Conclusão)

Itens do <i>framework</i>	Insumo	Origem
<b><i>Mentoring</i></b>	<ul style="list-style-type: none"> <li>• Adotado para resolver com o aluno o desafio com o propósito de sanar dúvidas.</li> </ul>	<ul style="list-style-type: none"> <li>• Revisão informal da literatura.</li> <li>• Observação participante.</li> </ul>
<b><i>Juiz /Scoreplay</i></b>	<ul style="list-style-type: none"> <li>• O personagem <i>juiz/scoreplay</i> atribui pontos e bônus aos alunos no decorrer das aulas.</li> </ul>	<ul style="list-style-type: none"> <li>• <i>Framework</i> Projeto SPIDER.</li> </ul>

Fonte: Elaboração própria (2019).

Logo, os quadros permitem visualizar a identificação dos elementos de jogos adotados em relação à fonte de dados que foi relacionado.

### 3.2 PLANO DE ENSINO

O plano de ensino *gamificado* (disponível no Apêndice D), conforme já citado, é o documento base que contém a descrição dos conteúdos, atividades e regras da *gamificação* que serão executadas e acompanhadas pela planilha. Nele estão detalhados: a identificação da disciplina; os conteúdos que serão trabalhados; os objetivos de ensino e de aprendizagem; as habilidades e competências desejadas; o cronograma de aulas; a descrição das atividades realizadas em sala; e o planejamento e a descrição do processo *gamificado* (regras para pontuação, concessão de *avatar*, bonificação, penalidades, aquisição de recurso estrela, conversão dos pontos e a atribuição de estrelas para nota na disciplina).

Com base nos requisitos estabelecidos, serão descritos a seguir as características, as práticas, as recompensas e os recursos que estão presentes no plano de ensino *gamificado* proposto.

#### 3.2.1 Características da *Gamificação*

A disciplina como um todo propõe-se a desenvolver no aluno a capacidade de compreender e elaborar algoritmos com qualidade e eficiência para a solução de problemas computacionais, independente de uma linguagem de programação (UNIVERSIDADE FEDERAL DO PARÁ, 2018). Mas, para o desenvolvimento da primeira versão, quanto ao conteúdo, optou-se por não trabalhar com todos, visto que o intuito era averiguar a eficácia da *gamificação* no processo de ensino-aprendizagem de algoritmos. Sendo assim, definiu-se por trabalhar com Estrutura de Dados Homogêneos (vetor e matriz), segundo conteúdo apontado no *survey* como o mais difícil de ensino-aprendizado, uma vez que Recursividade, primeiro conteúdo apontado, raramente consegue ser ministrado no contexto da Faculdade de

Computação da UFPA, embora presente na ementa. Isto foi verificado a partir de uma observação participante realizada pelo pesquisador em duas ofertas da disciplina, bem como em conversas não sistemáticas com professores da disciplina.

As metas, regras e *feedbacks* são características definidas no estudo de McGonigal (2011), tal como foram descritos no referencial teórico. Dentro da proposta, estabeleceram-se como metas os seguintes objetivos da *gamificação*:

- Objetivos de ensino: capacitar o discente da disciplina, após a conclusão dos estudos sobre o conteúdo, no desenvolvimento e análise de algoritmos com emprego de Estrutura de Dados Homogênea;
- Objetivos de aprendizado: (i) identificar estrutura de dados homogêneos em algoritmos; (ii) implementar estrutura de dados homogêneos em algoritmos e na linguagem de programação adotada para a disciplina; e (iii) realizar teste para analisar qual o resultado do processamento da estrutura de dados homogêneos.

As regras são adotadas para impor limitações de como os jogadores alcançarão as metas (MCGONIGAL, 2011). A forma como as regras foram implementadas serão detalhadas e explicadas na seção da planilha *gamificada*.

O *feedback* é o retorno emitido ao aluno em relação ao seu progresso dentro da *gamificação*. Como forma de desenvolver esse elemento, foi disponibilizada a planilha *gamificada* para os discentes no início de cada aula e via sala de aula virtual. Além disso, para todas as listas de desafios e de exercícios para casa empregadas foram disponibilizados os gabaritos.

Dentro das características da *gamificação* identificou-se o elemento de jogo personagem, no qual as pessoas envolvidas na disciplina receberam funções e nomes diferenciados, com o intuito de gerar uma experiência lúdica. As informações sobre personagens estão sistematizadas no Quadro 10.

Quadro 10 – Descrição de papéis

Identificação	Função
<b>Master</b>	Professor/ Pesquisador
<b>Mentora</b>	Professora da Disciplina
<b>Coach</b>	Acompanhamento na resolução de problemas
<b>Juiz/scoreplay</b>	Colaborador para desenvolvimento do score da turma
<b>Players</b>	Jogadores/ estudantes

Fonte: Elaboração própria (2019).

### 3.2.2 Práticas

Para compor as atividades da sala de aula *gamificada*, foram adotadas as seguintes práticas para o ensino e avaliação da aprendizagem dos discentes:

- **Conversa Inicial:** momento anterior ao início das atividades planejadas, em que foram explicadas a proposta da pesquisa, as metas e as regras da *gamificação* e a identificação dos personagens do jogo, em que foi descrito o papel de cada integrante dentro da sala de aula;
- **Aula Teórica e Prática:** aulas em sala, cujo objetivo era explicar o conteúdo de Estrutura de Dados Homogêneos (vetor e matriz). Para tal, primeiro era realizada uma apresentação do conteúdo, seguindo a ordem lógica de metáfora, conceito, implementação na linguagem e exercício exemplo e, em seguida, os alunos praticavam os conceitos aprendidos por meio de uma lista de atividades para resolver em sala de aula;
- **Dojo *Randori*:** dinâmica para ensino de programação em ambiente seguro, em que o discente aprende por tentativa e erro, com o auxílio de seus pares (LUZ; NETO, 2012). Durante a dinâmica, os discentes, em rodízio, assumem os seguintes papéis: (i) piloto, responsável pelo desenvolvimento do código; (ii) copiloto, com a função de narrar e auxiliar no planejamento e na codificação do programa; e (iii) plateia, que interage entre si com a intenção de cooperar com o piloto e copiloto durante o desenvolvimento;
- **Laboratório de Programação (LAB):** momento em que os alunos foram desafiados com problemas computacionais para a construção de soluções algorítmicas em duplas (ELGRABLY; OLIVEIRA, 2018);
- **Teste Teórico:** momento em que foi requerido o conhecimento individual dos discentes em relação aos conteúdos apresentados em sala, por meio de prova;
- **Aula de *Feedback*:** dinâmica em que o professor fez questionamentos para os discentes sobre o processo de *gamificação*, com a intenção de encontrar melhorias a serem implementadas na versão seguinte.

### 3.2.3 Recompensas

Os elementos enquadrados nessa categoria reforçam comportamentos positivos e procuram inibir os negativos. São eles:

- Bônus: concebido para ser uma recompensa positiva aos discentes que demonstraram comportamentos positivos, em relação aos critérios estabelecidos no plano de ensino, adaptados do *framework* do Projeto SPIDER. O aluno conseguia bonificar quando: (i) estava presente em sala; (ii) participava da aula; (iii) sugestionava mudanças ou informações importantes sobre o conteúdo; e (iv) questionava durante a aula;
- Penalidades: consideradas como recompensas negativas, que foram empregadas para remover bonificações já atribuídas aos alunos, nas seguintes situações: (i) falta não justificada em aula; (ii) uso do computador, exceto para acesso à plataforma Moodle ou *software* Pascalzim; e (iii) uso do celular;
- Recurso Estrela: identificado na revisão informal da literatura, presente na pesquisa (WERBACH; HUNTER, 2012), sendo que foi adotado como forma de sintetizar todas as bonificações diárias acumuladas pelos alunos;
- Ponto: recompensa positiva atribuída aos alunos que conseguiam resolver os desafios presentes nas práticas da *gamificação*. Como regra para esse elemento foi estabelecido que à medida que fosse concedido, não poderia mais ser retirado;
- Medalhas: dois tipos de medalhas foram adicionados à proposta, com o propósito de recompensar positivamente os alunos ao final da abordagem: (i) *Power*, concedida para os alunos com as maiores pontuações na resolução dos desafios da disciplina; e (ii) Participativo, atribuída aos discentes com maior aquisição de recurso estrela;
- Colaboração: quesito utilizado para conceder bonificação para os alunos;
- *Avatar*: representação visual ou simbólica do personagem que o participante pode receber ou se identificar durante o jogo. Este recurso ajuda na imersão do participante na dinâmica e pode ser empregado como motivador das ações, permitindo ao participante uma outra forma de acompanhar a sua progressão. Para o *framework*, configurou-se como uma recompensa positiva de *feedback* diário, o qual demonstra para o discente o seu progresso na resolução dos desafios.

### 3.2.4 Recursos e Estratégias de Ensino

Os recursos adotados para a proposta *gamificada* foram:

- Lista de desafios: lista com questões resolvidas em sala.
- Lista de exercício para casa: lista visando a exercitação fora do horário da aula.
- *Coaching*: auxílio concedido aos discentes por meio de dicas na resolução das questões pelos seguintes personagens: master, mentora, *coach*.

- *Mentoring*: resolução feita de um personagem (master, mentora, *coach*) com algum aluno, com o propósito do entendimento do desafio, ao qual esteja com dificuldades;
- *Juiz/scoreplay*: personagem responsável por atribuir e/ou remover os pontos e/ou bônus dos discentes na planilha *gamificada*.

### 3.3 PLANILHA GAMIFICADA

A planilha *gamificada* foi o instrumento empregado no decorrer das aulas para coletar informações sobre o desempenho e o comportamento dos alunos. As regras do jogo serviram de base para a organização das abas (guias) da planilha e sua programação de forma automática e integrada. Isto é, conforme os valores iam sendo preenchidos, a planilha calculava o recurso estrela para o comportamento e a pontuação para indicar o *avatar* correspondente.

#### 3.3.1 Bonificações e penalidades por comportamento

No Quadro 11 são apresentados os quesitos adotados para a bonificação por comportamento e a quantidade de bônus atribuída.

Quadro 11 – Quesitos de bonificação por comportamento

Quesito	Quantidade de bônus
<b>Presença</b>	10
<b>Participação</b>	2
<b>Questionamento</b>	2
<b>Sugestão</b>	2
<b>Colaboração</b>	2

Fonte: Elaboração própria (2019).

Na Figura 28 é visualizada a aba da planilha em que o juiz/*scoreplay* atribuía o elemento bônus para cada quesito estabelecido, quantas vezes fossem necessárias. Por regra, a planilha limitava a contabilização de cada quesito a 10 bônus, durante a conversão para recurso estrela (vide Figura 30).

Figura 28 – Visualização exemplo de bonificação por comportamento

Presença								
Membro	Aula 1	Aula 2	Aula 3	Aula 4	Aula 5	Aula 6	Aula 7	TOTAL
Aluno 01								0
Aluno 02								0
Aluno 03								0
Aluno 04								0
Aluno 05								0
Aluno 06								0
Aluno 07								0
Aluno 08								0
Aluno 09								0
Aluno 10								0

Fonte: Elaboração própria (2019).

O Quadro 12 apresenta os quesitos adotados para penalizar os discentes por comportamento e a quantidade de bônus retirada.

Quadro 12 – Quesitos de penalidade por comportamento

Quesito	Quantidade de bônus
Falta	10
Atraso em 10 minutos	2
Uso do celular	2
Uso do computador	2

Fonte: Elaboração própria (2019).

A planilha contém uma aba específica para o juiz/*scoreplay* atribuir a retirada do elemento bônus para cada quesito estabelecido, quantas vezes se fizer necessário (vide Figura 29). Por regra, a planilha limitava unicamente o atraso em dez minutos até quatro penalizações, durante a conversão para recurso estrela (vide Figura 30).



Figura 29 – Visualização exemplo de penalidades por comportamento

Atraso em 10 minutos								
Membro	Aula 1	Aula 2	Aula 3	Aula 4	Aula 5	Aula 6	Aula 7	TOTAL
Aluno 01								0
Aluno 02								0
Aluno 03								0
Aluno 04								0
Aluno 05								0
Aluno 06								0
Aluno 07								0
Aluno 08								0
Aluno 09								0
Aluno 10								0

Fonte: Elaboração própria (2019).

### 3.3.2 Recurso estrela

O recurso estrela refere-se às bonificações concedidas a cada aula. Por regra, foi estabelecida uma faixa de pontuação para concessão do recurso, conforme descrito no Quadro 13.

Quadro 13 – Faixa de bonificação para concessão de recurso estrela

Faixa de bonificação	Quantidade de estrelas
0 – 4	0
5 – 16	1
17 – 33	2
34 – 50	3

Fonte: Elaboração própria (2019).

A aba da planilha responsável pela conversão das bonificações diárias dos discentes em recurso estrela está ilustrada na Figura 30.

Figura 30 – Visualização de parte da aba responsável pela geração de estrela

Membro	Estrelas Aula 01											TOTAL BÔNUS	TOTAL ESTRELAS
	Presença	Participações	Sugestão	Pergunta	Colaboração	Falta	Atraso	Uso Cel	Uso PC	Atrapalhar aula			
Aluno 01	0	0	0	0	0	0	0	0	0	0	0,00	0	
Aluno 02	0	0	0	0	0	0	0	0	0	0	0,00	0	
Aluno 03	0	0	0	0	0	0	0	0	0	0	0,00	0	
Aluno 04	0	0	0	0	0	0	0	0	0	0	0,00	0	
Aluno 05	0	0	0	0	0	0	0	0	0	0	0,00	0	
Aluno 06	0	0	0	0	0	0	0	0	0	0	0,00	0	
Aluno 07	0	0	0	0	0	0	0	0	0	0	0,00	0	
Aluno 08	0	0	0	0	0	0	0	0	0	0	0,00	0	
Aluno 09	0	0	0	0	0	0	0	0	0	0	0,00	0	
Aluno 10	0	0	0	0	0	0	0	0	0	0	0,00	0	

Fonte: Elaboração própria (2019).

### 3.3.3 Pontuação por práticas desenvolvidas

Em acordo com o plano de ensino já citado, os discentes poderiam ganhar pontos durante as práticas desenvolvidas. Esta pontuação era preenchida em várias abas presentes na planilha.

A Figura 31 ilustra a aba da planilha de concessão de pontuação para os dias da aula teórica e prática. Nessa aba, o juiz/scoreplay deveria preencher o valor um para cada questão concluída pelo discente e a planilha acrescentava automaticamente 25 pontos à coluna total.

Figura 31 – Visualização da aba lista de exercício para as aulas teóricas e práticas da planilha *gamificada*

Membros	Resultado Pontuação Aula Teórica e Prática - 1º Dia				TOTAL	AVATAR DIA
	1º Questão	2º Questão	3º Questão	4º Questão		
Aluno 01					0	AMADOR
Aluno 02					0	AMADOR
Aluno 03					0	AMADOR
Aluno 04					0	AMADOR
Aluno 05					0	AMADOR
Aluno 06					0	AMADOR
Aluno 07					0	AMADOR
Aluno 08					0	AMADOR
Aluno 09					0	AMADOR
Aluno 10					0	AMADOR

Fonte: Elaboração própria (2019).

Os critérios, a explicação e a máxima pontuação atribuída para a prática de Dojo *Randori* estão presentes no Quadro 14.

Quadro 14 – Critério, explicação e pontuação atribuída a prática do Dojo do tipo *Randori*

Critério (coluna)	Explicação	Pontuação atribuída
Concluir uma funcionalidade	Averiguar a capacidade de o discente finalizar uma funcionalidade da questão proposta.	20
Utilizar o fluxo programar-testar	Identificar se o discente entendeu e adotou a forma de desenvolvimento requerido pela prática.	30
Narrar como copiloto	Apontar a capacidade de o discente analisar o código produzido.	100
Nota geral das questões	Pontuação atribuída pelo <i>master</i> em relação à corretude e completude do código desenvolvido por todos os discentes ao final.	50

Fonte: Elaboração própria (2019).

Na Figura 32 pode-se visualizar a aba da planilha em que o *master* preenchia a pontuação atribuída ao aluno durante a prática do Dojo *Randori*, de acordo com os critérios estabelecidos no Quadro 14.

Figura 32 – Visualização da aba de pontuação aula Dojo do tipo *Randori*

Membros	Dojo				TOTAL	AVATAR
	Uma funcionalidade concluída	Utilizou o fluxo Programar-Testar	Narrou como copiloto	Nota geral das questões		
Aluno 01					0	AMADOR
Aluno 02					0	AMADOR
Aluno 03					0	AMADOR
Aluno 04					0	AMADOR
Aluno 05					0	AMADOR
Aluno 06					0	AMADOR
Aluno 07					0	AMADOR
Aluno 08					0	AMADOR
Aluno 09					0	AMADOR
Aluno 10					0	AMADOR

Fonte: Elaboração própria (2019).

No Quadro 15 são apresentados os critérios, a explicação e as pontuações atribuídas para as práticas de LAB e teste teórico.

Quadro 15 – Critério, explicação e pontuação para as práticas de LAB e teste teórico

Critério (coluna)	Explicação	Pontuação LAB	Pontuação teste teórico
Tentativa	Tentativa do discente em resolver o desafio proposto	20	7,5
Nota do desafio	Análise da corretude e completude do código desenvolvido pelo discente	30	67,5

Fonte: Elaboração própria (2019).

Na Figura 34 é ilustrada a aba da planilha responsável pela pontuação recebida com a resolução dos desafios nas práticas de LAB e teste teórico. Cabia ao *master* preencher esta aba, considerando a tentativa do discente e a análise do código desenvolvido. O valor 1 para a tentativa era convertido automaticamente pela pontuação indicada no Quadro 15. A nota atribuída poderia variar de 0 até o máximo estabelecido no Quadro 15. Ao final, a planilha calculava o total para cada aluno.

Figura 33 – Visualização da pontuação por questão da aula LAB e prova teórica

Membros	LAB - 1º Desafio		TOTAL
	Tentativa	Nota desafio	
Aluno 01			0
Aluno 02			0
Aluno 03			0
Aluno 04			0
Aluno 05			0
Aluno 06			0
Aluno 07			0
Aluno 08			0
Aluno 09			0
Aluno 10			0

Fonte: Elaboração própria (2019).

A aba responsável pela visualização do resultado da prática LAB está na Figura 34. As colunas adotadas nessa prática são similares para o resultado do teste teórico, exceto a coluna Total Lista Exercício Casa.

Figura 34 – Visualização de parte da aba de pontuação da aula LAB

Membros	Resultado Pontuação LAB				TOTAL	TOTAL LISTA EXERCÍCIO CASA	TOTAL GERAL LAB	AVATAR DIA
	1º Desafio	2º Desafio	3º Desafio	4º Desafio				
Aluno 01	0	0	0	0	0	0	0	AMADOR
Aluno 02	0	0	0	0	0	0	0	AMADOR
Aluno 03	0	0	0	0	0	0	0	AMADOR
Aluno 04	0	0	0	0	0	0	0	AMADOR
Aluno 05	0	0	0	0	0	0	0	AMADOR
Aluno 06	0	0	0	0	0	0	0	AMADOR
Aluno 07	0	0	0	0	0	0	0	AMADOR
Aluno 08	0	0	0	0	0	0	0	AMADOR
Aluno 09	0	0	0	0	0	0	0	AMADOR
Aluno 10	0	0	0	0	0	0	0	AMADOR

Fonte: Elaboração própria (2019).

Na Figura 35 é visualizada a aba *score*, com informações da pontuação adquirida pelo aluno em cada uma das práticas e o total das pontuações na *gamificação*. Essa aba foi projetada para os discentes terem acesso a um *feedback* sobre as pontuações recebidas.

Figura 35 – Visualização da aba score.

Membro	Lista Exercício 01	Lista Exercício 02	Dojo	LAB	Prova Teórica	TOTAL	AVATAR FINAL
Aluno 01	0	0	0	0	0	0	AMADOR
Aluno 02	0	0	0	0	0	0	AMADOR
Aluno 03	0	0	0	0	0	0	AMADOR
Aluno 04	0	0	0	0	0	0	AMADOR
Aluno 05	0	0	0	0	0	0	AMADOR
Aluno 06	0	0	0	0	0	0	AMADOR
Aluno 07	0	0	0	0	0	0	AMADOR
Aluno 08	0	0	0	0	0	0	AMADOR
Aluno 09	0	0	0	0	0	0	AMADOR
Aluno 10	0	0	0	0	0	0	AMADOR

Fonte: Elaboração própria (2019).

### 3.3.4 Avatar

O *avatar diário* é calculado automaticamente pela planilha *gamificada* com a quantidade de pontos que o discente recebeu a cada prática. O Quadro 16 apresenta as faixas de pontuação implementadas na planilha para a concessão do elemento citado.

Quadro 16 – Atribuição do *avatar* aos alunos por prática em sala de aula

Avatar	Faixa de Pontuação				
	ATP	Dojo	LAB	Teste	Geral
Analista	90 – 100	180 - 200	270 - 300	270 - 300	900 - 1000
Programador	70 – 89	150 - 179	230 - 269	230 - 269	700 – 899
Aprendiz	50 – 69	120 - 149	190 - 229	190 - 229	500 – 699
Amador	0 – 49	0 - 119	0 - 189	0 - 189	0 – 499

Fonte: Elaboração própria (2019).

### 3.3.5 Resultados finais da *gamificação*

Na Figura 36 é visualizada a representação do elemento medalha implementado na planilha *gamificada*. Os campos presentes são a posição, o valor conquistado, o nome do jogador e o tipo de medalha. Para a sala de aula *gamificada*: ouro é para o primeiro lugar; a prata para o segundo; e o bronze para o terceiro.

Figura 36 – Visualização exemplo de atribuição de Medalhas

Medalhas – Power			
Posição	Valor	Nome	Tipo
1º Lugar	0	Aluno 01	Ouro
2º Lugar	0	Aluno 01	Prata
3º Lugar	0	Aluno 01	Bronze

Fonte: Elaboração própria (2019).

A planilha *gamificada* foi projetada para os alunos terem a todo o momento o *feedback* do seu desempenho na *gamificação*. Então, a aba representada pela Figura 37 apresenta a conversão dos elementos de jogos em nota para os alunos, nesse caso os pontos e as estrelas.

Figura 37 – Visualização das abas de conversão das pontuações e estrelas em notas

Nota Final Pontuações	Total Pontos	Total Nota	Total Estrelas	Conversão Nota	Nota Final Conteúdo
Aluno 01	0	0,00	0	0,0	0,00
Aluno 02	0	0,00	0	0,0	0,00
Aluno 03	0	0,00	0	0,0	0,00
Aluno 04	0	0,00	0	0,0	0,00
Aluno 05	0	0,00	0	0,0	0,00
Aluno 06	0	0,00	0	0,0	0,00
Aluno 07	0	0,00	0	0,0	0,00
Aluno 08	0	0,00	0	0,0	0,00
Aluno 09	0	0,00	0	0,0	0,00
Aluno 10	0	0,00	0	0,0	0,00

Fonte: Elaboração própria (2019).

### 3.4 REVISÃO POR PARES

Após a concepção e o desenvolvimento da primeira versão do *framework gamificado*, entendemos que seria necessário passar por uma avaliação de especialistas para que estes identificassem pontos de melhoria e apontamentos sobre a estratégia para trabalhar com *gamificação* e algoritmos no produto. Optamos por utilizar a revisão por pares, por ser um mecanismo de avaliação de propostas por meio de uma análise realizada por especialistas. Este tipo de avaliação pode ser definido como “o mecanismo mais efetivo e eficaz para garantir a qualidade, confiabilidade, integridade e consistência da literatura acadêmica” (NASSI-CALÒ, 2015). A revisão por pares do *framework gamificado* ocorreu em três etapas. São elas: planejamento, condução e análise.

#### 3.4.1 Planejamento

Durante a etapa de planejamento, definiu-se o instrumento que iria ser adotado e os especialistas que iriam participar da revisão. Quanto ao instrumento (vide Apêndice E), este foi adaptado do Projeto SPIDER (SOFTWARE PROCESS IMPROVEMENT: DEVELOPMENT AND RESEARCH - UFPA, 2009) contemplando as seguintes seções: (i) apresentação; (ii) objetivos da Revisão por Pares; (iii) instruções para execução; (iv) dados de identificação do revisor; (v) perfil do revisor; (vi) apresentação da proposta; e (vii) revisão do produto.

Como revisores foram escolhidos quatro professores/pesquisadores, de duas distintas especialidades:

- Algoritmos: professores da Faculdade Computação da Universidade Federal do Pará que já ministraram a disciplina, em que foram convidados dois, mas apenas um aceitou participar da revisão.
- *Gamificação*: pesquisadores do Projeto SPIDER envolvidos em pesquisas sobre a especialidade, em que foram convidados dois e estes aceitaram.

### 3.4.2 Condução

Para a revisão em si, houve duas sessões de apresentação e avaliação do produto, uma para cada grupo de especialistas, em que foram distribuídos os formulários de revisão e o plano de ensino *gamificado*. Em cada sessão foi explanado pelo pesquisador responsável pela *gamificação*: (i) os procedimentos da revisão; (ii) o uso do produto pelos personagens envolvidos; e (iii) o funcionamento da planilha *gamificada*.

### 3.4.3 Análise

Após a finalização da revisão por pares, os dados coletados foram sintetizados no Quadro 17, construído segundo a ordem de entrega dos instrumentos pelos revisores, contendo seis colunas:

- ID: representada pelo acrônimo RP (Revisão por Pares), associado ao número da revisão empreendida.
- Categoria: identifica a situação da consideração realizada pelo revisor, distinta nas seguintes opções no formulário de revisão:
  - ⇒ **TA (Técnico Alto)**: apontava que havia sido encontrado um problema em um item que, se não fosse alterado, comprometeria o bom andamento das atividades previstas no *framework*.
  - ⇒ **TB (Técnico Baixo)**: apontava que havia sido encontrado um problema em um item que seria conveniente alterar.
  - ⇒ **E (Editorial)**: apontava a ocorrência de um erro de português ou que o texto poderia ser melhorado.
  - ⇒ **Q (Questionamento)**: apontava a ocorrência de dúvidas quanto ao conteúdo do plano de ensino ou da planilha disponibilizados.
  - ⇒ **G (Geral)**: apontava um comentário geral em relação ao produto;
  - ⇒ **S (Sugestão)**: indicava uma sugestão para a proposta.



- Item: assinalava a parte do documento que deveria ser alterada.
- Comentário/justificativa: registra as considerações do revisor que precisam ser implementadas no item.
- Sugestão: aponta um caminho para implementar a solicitação realizada, campo preenchido pelo revisor.
- Parecer: indicava a avaliação dos autores sobre a alteração ou não no item pelas considerações dos revisores.

Quadro 17 – Síntese dos dados da revisão por pares (Continua...)

ID	Categoria (Revisor)	Item (Revisor)	Comentário/Justificativa (Revisor)	Sugestão (Revisor)	Parecer (Pesquisador e orientadores)
RP 01	TA	Penalidade	Aluno não é penalizado por atrapalhar a aula.	Inserir uma penalidade para essa punição.	Adicionado ao <i>framework</i> .
RP 02	TB	Planilha	Melhorar o preenchimento da planilha.	Adicionar campos para observações sobre cada uma das práticas.	Adicionado ao <i>framework</i> .
RP 03	Q	Elementos	O experimento tem mais elementos de jogos.	Yu-kai-Chou (Usar como referência o estudo de CHOU (2013)).	Não adicionado ao <i>framework</i> .
RP 04	TA	Prática do Dojo	Avaliar refatoração e Dojo.	Retirar refatoração	Foi retirado do <i>framework</i> , sendo que sua pontuação foi transferida para o quesito programar-testar.
RP 05	TB	Planilha gamificada	Adicionar Tempo real.	Planilha em tempo real, por meio da ferramenta <i>Google Drive</i> .	Adicionado ao <i>framework</i> .
RP 06	TB	Personagens (item 5.1 no plano de ensino)	Nomes mais lúdicos.	Trocar (professora) por outro nome.	Adicionado ao <i>framework</i> , foi alterado de professora para mentora.

Fonte: Elaboração própria (2019).

Quadro 17 – Síntese dos dados da revisão por pares (Conclusão)

ID	Categoria (Revisor)	Item (Revisor)	Comentário/ Justificativa (Revisor)	Sugestão (Revisor)	Parecer (Pesquisador e orientadores)
RP 07	TA	Avatar (item 5.2 – Tabela 1 no plano de ensino)	Avatares são insígnias.	Substituir o termo <i>avatar</i> por insígnia ou conquista.	Não foi alterado no <i>framework</i> , por entendermos que o <i>avatar</i> diário dará a noção para o participante do seu desenvolvimento dentro das atividades propostas.
RP 08	G	Pontos nos desafios da aula teórica e prática (item 5.2 no plano de ensino)	Questões (Certo ou errado).	Mais flexibilidade na correção da conquista.	Não foi alterado no <i>framework</i> , pois a carga de trabalho inviabilizaria o feedback automatizado para os jogadores.
RP 09	TB	Planilha gamificada	Ausência de <i>Ranking</i>	Devido ao <i>feedback</i> instantâneo, torna-se interessante o uso de técnicas de visualização, como gráfico de barras.	Adicionado ao <i>framework</i> , por meio de uma aba na planilha para que alunos pudessem visualizar um gráfico sobre suas pontuações.
RP 10	TB	Planilha gamificada	Seria bom o uso de visualização para acompanhar o progresso diário em todas as dimensões.	Não foi informada sugestão pelo revisor.	Adicionado ao <i>framework</i> , por meio de duas abas inseridas na planilha gamificada, uma para o <i>ranking</i> de pontuação e uma para ver a quantidade de estrela diária.
RP 11	S	Geral	Aplicar o experimento à turma noturna de Algoritmos no curso de Sistema de Informação.	Comparativo ao experimento na turma de Algoritmos do curso de Ciência da Computação.	Não é possível realizar, pois para essa situação teríamos que ter a aprovação do docente responsável. Além de que o pesquisador está em sua primeira execução de uma proposta gamificada.

Fonte: Elaboração própria (2019).

### 3.5 MELHORIAS

Como resultado da avaliação por pares, foram implementadas as solicitações apontadas pelos especialistas consideradas relevantes ao trabalho, com impacto direto no plano de ensino e planilha. As solicitações que não foram atendidas estão justificadas no próprio Quadro 17. A seguir as melhorias que foram implementadas:

- RP 01: foi identificado como a possibilidade de penalizar os alunos que estiverem atrapalhando a aula. Esta situação foi adicionada ao *framework* por meio do quesito penalidade por atrapalhar aula. O aluno que for enquadrado nesse quesito perderá dois bônus.
- RP 02: as observações sobre ocorrências ainda não previstas na planilha ou no plano de ensino foram adicionadas ao produto a partir da planilha *gamificada*. O pesquisador responsável pela *gamificação* informou para o juiz/*scoreplay* anotar as ocorrências em sala de aula, tais como, mas não limitadas a: (i) maior tempo de duração da prática; (ii) o porquê das bonificações e penalidades atribuídas ao aluno; (iii) mudança de regras de pontuação para auxiliar a turma; e (iv) observações sobre os elementos de jogos no experimento.
- RP 03: Não foi adicionado ao relato do produto *gamificado*, pois foram adotados outros autores para a explicação sobre os elementos de jogos adicionados.
- RP 04: a retirada do requisito refatoração foi considerada uma dica relevante para o desenvolvimento do experimento, pois o conceito de refatoração é avançado dentro da linha de desenvolvimento de software para alunos que ainda estão iniciando no curso.
- RP 05: disponibilizada a planilha *gamificada* aos discentes a partir da ferramenta Google Drive, e o *link* para acesso, a qualquer momento, estava disponível na plataforma Moodle.
- RP 06: o nome da professora no plano de ensino foi substituído por Mentora. Identificamos que a substituição de nomes permite, assim como os outros elementos de

*game*, fazer com que os alunos, no caso os jogadores, sintam-se em um ambiente diferenciado.

- RP 09: adicionada a aba *ranking* na planilha *gamificada* para os alunos acompanharem graficamente o seu andamento na *gamificação* (vide Figura 38) em relação ao elemento pontos. A planilha calculava o desempenho dos discentes à medida em que as pontuações fossem sendo adquiridas no decorrer das práticas.

Figura 38 – Exemplo de visualização do *ranking*



Fonte: Elaboração própria (2019).

- RP 10: foi adicionada outra aba com informações sobre a quantidade de estrelas diárias conquistadas pelos alunos e a quantidade total adquirida.

## 4 O USO DO FRAMEWORK GAMIFICADO: O EXPERIMENTO

No capítulo 04 apresentamos como planejamos e executamos um experimento com a sala de aula *gamificada* (detalhada no Capítulo 3); os resultados quantitativos e qualitativos alcançados com um experimento; bem como apontamos melhorias indicadas pelo pesquisador responsável pela *gamificação* e discentes que participaram do experimento.

O experimento com o uso do *framework gamificado* foi a forma encontrada para avaliar e validar nossa proposta. O contexto de realização foi a disciplina Algoritmos (código: EN05209), com carga horária de 68 horas, ofertada aos calouros no primeiro semestre letivo de 2018, pela Faculdade de Computação da Universidade Federal do Pará, em específico no período em que foi abordado o conteúdo de Estrutura de Dados Homogêneos (*Array*).

Com o intuito de nortear e analisar a execução do experimento, elaboramos uma questão norteadora para tal prática: *O uso de gamificação auxilia o engajamento da turma e colabora para a aprendizagem sobre o conteúdo de estrutura de dados homogêneos?*

### 4.1 PLANEJAMENTO DO EXPERIMENTO

O planejamento levou em consideração o plano de ensino e a planilha *gamificada* descritas no Capítulo 3 da Dissertação. Os itens do planejamento estão descritos a seguir.

#### 4.1.1 Cenário

A sala de aula é um dos núcleos mais importantes da *gamificação*, visto que foi nela o cenário do experimento e do jogo. As aulas ocorreram em um laboratório de informática, contendo: quadro magnético, destinado à explicação; projetor multimídia; e computadores para o docente e discentes. No dia da prova teórica, os discentes foram para outro local, sem acesso à internet e sem o uso do computador.

#### 4.1.2 Personagens

Uma turma de graduação do curso de Bacharelado em Ciência da Computação da UFPA contemplando 20 discentes, um monitor, a professora da disciplina e um estagiário, responsável pela *gamificação*.

### 4.1.3 Desenvolvimento das aulas

Conforme o plano de ensino, o experimento contabilizou oito aulas, assim descritas:

- **Aula 01:** destinada a uma conversa inicial sobre *gamificação*, em que foi apresentado o plano de ensino, em formato de apresentação, contendo: a meta, as regras e os personagens da *gamificação*. Foram explicados, também, como os discentes ganhariam pontos e bonificariam, assim como o cronograma das aulas. Neste dia não foi atribuída pontuação aos discentes, apenas bonificação.
- **Aulas 02 e 03:** foi adotada a abordagem teórica e prática (ATP) com o conteúdo estrutura de dados homogêneos. Na segunda aula foi trabalhado o conteúdo vetores e na terceira abordou-se sobre matrizes. A explicação estruturou-se da seguinte forma: o conceito, uma metáfora, a implementação na linguagem Pascal e, por fim, os discentes foram convidados a resolver questões referentes ao conteúdo. O discente recebia 25 pontos por resolução de desafio, com quatro desafios por dia, totalizando 100 pontos, além disso, ele poderia bonificar por comportamento.
- **Aulas 04 e 05:** foi realizada a aplicação da técnica do Dojo do tipo *Randori*, que é uma atividade realizada em pares em um ambiente seguro de programação. Os discentes tiveram um período inicial de dez minutos para verificar e analisar a questão proposta. Após esse período, um discente assumia como primeiro piloto e depois de sete minutos ocorria a troca por outro discente, até que todos os discentes participassem da atividade. A posição de copiloto era opcional, reservada para caso algum discente desejasse ajudar o piloto na codificação. Após três minutos do piloto codificando, a plateia podia interagir com ele, bem como o copiloto voluntariar-se para ajudar. Como material necessário, foram utilizados um computador e um projetor multimídia. No que tange a pontuação atribuída aos alunos na prática do Dojo *Randori*, seguiu-se os seguintes requisitos com as pontuações devidas: funcionalidade concluída, 20 pontos; uso do fluxo programar-testar, 30 pontos, narrar a programação, 100 pontos, nota geral para a questão desenvolvida pelos alunos, 50 pontos.

- **Aula 06:** houve a prática do LAB com pareamento, técnica na qual os discentes foram divididos em duplas por meio de sorteio, para resolver quatro problemas computacionais a partir de soluções algorítmicas, do conteúdo estrutura de dados homogêneos. As orientações foram repassadas pelos personagens: master e mentora, logo, no início da prática. Como regra estabelecida, não foi permitido o acesso à internet, porém foi permitido *coaching* como forma de ajudar as duplas. O material utilizado foi um computador para cada dupla. Essa prática foi adaptada do *framework* proposto em Elgrably e Oliveira (2018). No que se refere a pontuação concedida pela prática do LAB, pontuou-se pela tentativa (20 pontos) e pelo código desenvolvido (30 pontos), em um total de quatro questões, logo, 200 pontos pela prática em sala. Além disso, houve uma lista de exercício para casa, a qual era para entregar via plataforma Moodle nesse mesmo dia, que continha dez questões valendo dez pontos para cada, se respondida corretamente. Portanto, o total atribuído pela prática LAB foi de 300 pontos: 200 em sala e 100 via plataforma Moodle (lista de exercícios).
- **Aula 07:** os discentes foram submetidos a uma atividade avaliativa visando verificar e analisar o conhecimento e o aprendizado adquirido: um teste teórico contendo quatro questões subjetivas, em que duas questões versaram sobre vetor e duas foram sobre matrizes. Em relação à pontuação atribuída pelo teste teórico, era concedido ao aluno, a cada desafio, 7,5 pontos pela tentativa e 67,5 pontos pelo código desenvolvido, em um total de quatro desafios, logo, 300 pontos pela atividade.
- **Aula 08:** Os discentes respondiam a questionamentos em relação ao *framework gamificado*, com o intuito de registrar *feedbacks* sobre a abordagem para o processo de ensino e aprendizagem. Para esse dia não foi atribuída pontuação, apenas bonificação.

#### 4.1.4 O design da gamificação

A colaboração foi um dos elementos norteadores da proposta de *gamificação*, por compreender a necessidade de motivar os discentes da turma a uma prática cada vez mais presente no mercado de trabalho, qual seja, o trabalho em equipe. Neste estudo, como já

mencionado, o objetivo era pontuar, bonificar e avaliar os conhecimentos adquiridos sobre o conteúdo Estrutura de Dados Homogêneos. Para isso, como também já informado, foram utilizados o plano de ensino e a planilha *gamificada*.

À medida que os discentes adquiriam pontuações, a planilha eletrônica disponibilizada gerava os *avatares* diários automáticos, segundo o padrão estabelecido nas regras presentes no plano de ensino. No que tange à bonificação, o discente poderia ganhá-la ou perdê-la, conforme critérios já estabelecidos. Por dia, o elemento bônus podia gerar até três estrelas (vide Apêndice D – Plano de Ensino).

Um dos instrumentos que possibilitavam o *feedback* para os discentes sobre o progresso na *gamificação* foi a disponibilização da planilha *gamificada* na sala de aula estendida, assim como sua apresentação no início de cada aula. A planilha permitia aos discentes visualizar o seu progresso dentro do conteúdo *gamificado* na disciplina.

## 4.2 AVALIAÇÃO DO EXPERIMENTO

Para avaliar o uso do *framework*, utilizamos uma combinação das abordagens de cunho qualitativo e quantitativo, conforme descrevemos a seguir. Além disso, identificamos por meio das análises dos dados: casos particulares na abordagem *gamificada* e registro das melhorias para o produto dessa dissertação de mestrado.

### 4.2.1 Discussão e resultados qualitativos

Segundo Creswell (2007), os dados qualitativos vêm das percepções do pesquisador em relação ao que está observando. Portanto, para uma melhor compreensão, dividimos nossos resultados e discussão em dois grupos: Metodologias de ensino – SWOT Diário e dados sobre bonificações e penalidades. O primeiro grupo foi coletado por meio de anotações estruturadas das percepções do pesquisador responsável pela condução da *gamificação*, com contribuições sugestionadas pela professora e pelo monitor da disciplina, bem como pelas gravações de áudio, previamente aprovadas pelos discentes e monitor participantes. O segundo grupo foi coletado por meio da planilha *gamificada* dividida em abas referentes às bonificações e penalidades diárias, acrescida das percepções do juiz/*scoreplay*.



#### 4.2.1.1 Metodologias de Ensino – SWOT Diário

A avaliação das metodologias de ensino empregadas e da *gamificação* empreendida, tomou por base a matriz de análise SWOT (*Straights, Weaknesses, Opportunities, Threats*) e os resultados da aula de *feedback*. A matriz de análise SWOT, também conhecida como FOFA (Forças, Fraquezas, Oportunidades e Ameaças), tem como função, segundo (BALDNER, DECOURT, NEVES, 2012), “analisar e cruzar as forças da empresa e as oportunidades do mercado, as fraquezas da empresa e as ameaças do mercado”. Essa forma de análise é adotada nas ciências sociais com o intuito de aprimorar o desenvolvimento estratégico e operacional da empresa, como podemos perceber na citação dos autores. Utilizamos essa estratégia de análise, pois no estudo de Santos (2018) foi adotada para avaliar e aprimorar uma abordagem *gamificada* para o ensino e aprendizagem de Análise de Pontos por Função. Portanto, foi a forma utilizada para o registro de informações acerca de cada uma das práticas realizadas em sala de aula para avaliar essa abordagem.

No contexto do nosso experimento, ao final de cada aula *gamificada* fazíamos uma roda de conversa direcionada à construção da matriz SWOT. Os participantes envolvidos nesse momento eram: a docente responsável pela disciplina, os monitores, o pesquisador que estava desenvolvendo o experimento e o juiz/*scoreplay*.

Durante as sessões eram levantados:

- Pontos fortes: os elementos, as mecânicas e formas trabalhadas em sala que foram positivas para o processo de ensino e aprendizagem;
- Pontos fracos: o que não teve boa aceitação e, por consequência, foi prejudicial ao desempenho dos discentes;
- Melhorias: as oportunidades de mudanças para o uso futuro do *framework gamificado*;
- Observações: anotações sobre possíveis mudanças no planejamento dentro de sala de aula.

Assim, dos elementos que compõem a matriz de análise SWOT, alteramos apenas o item ameaças para observações, com o intuito de aprimorar a nossa avaliação sobre o processo.

No Quadro 18 podemos observar a sistematização dos dados sobre a SWOT adaptada, produzida após o primeiro dia de aula teórica e prática (ATP).

Quadro 18 – Matriz SWOT adaptada: ATP primeiro dia

PONTOS FORTES	MELHORIAS
<ul style="list-style-type: none"> <li>• Colaboração como quesito para bonificação dos discentes.</li> <li>• O uso de um quadro em papel para ajudar na localização dos discentes facilitou a pontuação.</li> <li>• A solicitação aos discentes que haviam terminado seus exercícios, em tempo hábil, para se sentarem ao lado de um colega com dificuldades para auxiliar nas atividades, permitiu não só bonificar o discente por colaboração, como um aprendizado pareado.</li> </ul>	<ul style="list-style-type: none"> <li>• Obter e levar para sala de aula mais exemplos de códigos fontes e mais vetores desenhados.</li> <li>• Utilizar função do software Microsoft Excel para a planilha computar de forma automática as faltas dos discentes.</li> </ul>
PONTOS FRACOS	OBSERVAÇÕES
<ul style="list-style-type: none"> <li>• Não foi possível pontuar o quesito sugestão, pois o juiz/scoreplay não conseguiu identificar no comportamento dos discentes.</li> <li>• A quantidade de slide na aula teórica e prática, não permitiu uma maior participação dos discentes.</li> <li>• Exemplos em códigos fontes e vetores desenhados insuficientes para instigar todos os alunos.</li> <li>• A planilha não penaliza de forma automática as faltas dos discentes.</li> </ul>	<ul style="list-style-type: none"> <li>• O tempo de aula foi acrescido de cinco minutos para que alguns alunos pudessem finalizar o exercício.</li> </ul>

Fonte: Elaboração própria (2019).

Podemos identificar no Quadro 18, como pontos fortes, a colaboração como quesito de bonificação, pois permitiu com que os discentes melhor se dispusessem para auxiliar os colegas, viabilizando aprendizagem por pares. O uso de um quadro em papel para identificar o local onde os discentes estavam sentados durante a aula facilitou para o juiz/scoreplay bonificar e pontuar os discentes. Os pontos fracos encontrados foram: não foi possível identificar o quesito sugestão para bonificação, a quantidade de *slides* da aula teórica e prática não foi suficiente para gerar participação dos discentes; e a planilha não penaliza automaticamente a falta.

Na análise dos pontos fracos foi possível identificarmos como possíveis melhorias para a próxima aula teórica e prática: obter e usar em sala de aula, mais exemplos de códigos, fontes e estrutura de dados homogêneos desenhados para o momento teórico da aula com a finalidade de estimular a participação dos alunos; e o uso de função no *software* Microsoft Excel com o intuito de penalizar as faltas de maneira automática, para gerar uma nova versão da planilha *gamificada*. Nas nossas observações registramos: o acréscimo de cinco minutos no tempo de aula com o intuito de disponibilizar tempo para os alunos finalizarem as atividades.

O Quadro 19 apresenta a matriz SWOT adaptada para o segundo dia de aula teórica e prática.

Quadro 19 – Matriz SWOT adaptada: ATP segundo dia

PONTOS FORTES	MELHORIAS
<ul style="list-style-type: none"> <li>• Aluno bonificar por sugestão caso ele desenvolvesse um código com mais funcionalidades disponíveis que o solicitado pela questão.</li> <li>• Uso do pincel e o quadro branco como recurso no ensino, pois permitiu maior participação dos alunos em aula.</li> </ul>	<ul style="list-style-type: none"> <li>• Verificar a corretude dos slides aplicados em sala.</li> <li>• Elaborar slides com maior quantidade de exemplos para explicar o conceito de matriz e suas aplicações.</li> <li>• Alterar a planilha para a conversão em recurso estrela, limitar e permitir ao juiz/<i>scoreplay</i> pontuar todos os comportamentos previstos como bonificações em sala de aula.</li> </ul>
PONTOS FRACOS	OBSERVAÇÕES
<ul style="list-style-type: none"> <li>• A ocorrência de slides com erros.</li> <li>• Uso do pincel e o quadro branco durante a apresentação do conceito, pois o maior tempo da aula foi utilizado para apresentação da teoria.</li> <li>• Planilha não programada para permitir ao juiz/<i>scoreplay</i> anotar todas as bonificações do discente, pois quebraria a regra estabelecida do máximo de bonificações por quesito.</li> </ul>	<ul style="list-style-type: none"> <li>• O tempo de aula foi acrescido de dez minutos para que alguns discentes pudessem finalizar os exercícios propostos.</li> <li>• Foi disponibilizada a versão 3 da planilha <i>gamificada</i> com as alterações necessárias para o juiz/<i>scoreplay</i>.</li> </ul>

Fonte: Elaboração própria (2019).

No segundo dia de aula teórica e prática (vide Quadro 19), os pontos fortes apontados foram: o quesito sugestão ser utilizado para bonificar os discentes que desenvolviam um código com mais funcionalidades que o solicitado pela questão, como exemplo: uma questão solicitava apresentar o resultado de uma operação com matriz e caso os alunos codificassem para apresentar o resultado em formato matricial, bonificava por sugestão; outra força registrada foi o uso do pincel e quadro branco para desenhar formatos de matrizes e formular questionamento aos discentes como forma de instigar a participação deles. Como pontos fracos, registramos: a ocorrência de erro nos *slides*; o uso do pincel e quadro durante a apresentação, pois fez com que a aula estivesse mais centrada na teoria, o que impossibilitou a alguns discentes terminarem a lista de desafios, determinada para o dia; e a planilha não estar programada para o juiz/*scoreplay* registrar todos os comportamentos bonificáveis adotados pelos discentes em sala de aula.

As melhorias indicadas para os próximos usos e versão do *framework gamificado* foram: verificar a corretude dos *slides* aplicados em sala; elaborar mais *slides* com exemplos e aplicações de matriz, com o intuito de gerar participação; alterar a aba da planilha responsável pela conversão em recurso estrela, para que de maneira automática limitasse as bonificações e penalidade ao estabelecido na regra do plano de ensino. Nesse dia tivemos um acréscimo de dez minutos ao tempo de aula para os discentes tentarem concluir a atividade e a terceira versão da planilha foi disponibilizada para o juiz/*scoreplay* com as alterações registradas na primeira aula teórica e prática.

Quanto às aulas teóricas e práticas, de maneira geral, os discentes participaram ativamente a partir do estímulo feito pelo docente e monitores da disciplina no momento da explicação, por meio de perguntas abertas sobre o conteúdo. Além disso, foi possível coletar dos discentes durante a aula de *feedback*, as seguintes informações: (i) o método adotado para a explicação estava compreensível; (ii) as questões estavam com um nível de dificuldade bom; (iii) era possível participar dentro da aula; e (iv) era preciso trazer mais exemplos.

Referente ao Dojo *Randori*, foram realizadas várias adaptações com o intuito de aprimorar a experiência de ensino e aprendizado para os discentes. O primeiro formato de execução tem sua avaliação descrita no Quadro 20.

Quadro 20 – Matriz SWOT adaptada: Dojo *Randori* primeira implementação (Continua...)

PONTOS FORTES	MELHORIAS
<ul style="list-style-type: none"> <li>• Discentes desenvolveram o código em papel e utilizaram quando estavam como piloto ou copiloto na prática.</li> <li>• Colaboração entre membros da plateia para resolução das funcionalidades.</li> </ul>	<ul style="list-style-type: none"> <li>• Melhorar a preparação da aula.</li> <li>• Formular uma questão aberta para ser resolvida.</li> <li>• Retirar o papel do copiloto e permitir a plateia interagir com o piloto após três minutos de sua codificação.</li> <li>• Reforçar o papel do piloto para os discentes.</li> <li>• Para uma próxima versão ou uso do <i>framework</i> colocar no papel do piloto e copiloto o monitor e o professor.</li> <li>• Pontuação por colaboração apenas quando o conhecimento for difundido para a plateia inteira.</li> <li>• Adequar fontes de exibição do software Pascalzim para apresentação em <i>datashow</i>.</li> <li>• Disponibilizar em papel impresso a questão ou questões trabalhadas na prática para todos os discentes.</li> </ul>

Fonte: Elaboração própria (2019).

Quadro 20 – Matriz SWOT adaptada: Dojo *Randori* primeira implementação (Conclusão)

PONTOS FRACOS	OBSERVAÇÕES
<ul style="list-style-type: none"> <li>• Durante as informações iniciais da prática faltou uma melhor explicação sobre o fluxo programar-testar.</li> <li>• Pouca participação dos discentes no decorrer da prática.</li> <li>• Questões estruturadas não possibilitam uma maior quantidade de participações dos estudantes.</li> <li>• A função de copiloto não funcionou como o esperado.</li> <li>• Poucos estudantes na função de piloto interagiam com os copilotos para narrarem o desenvolvimento da solução.</li> <li>• Discentes que normalmente participavam nas aulas, não participaram interagindo com o piloto e copilo.</li> <li>• Houve a necessidade de indicar alguns discentes para realizar a prática.</li> <li>• Houve a ocorrência de grupos de discentes buscando soluções entre si, sem compartilhar com os demais colegas.</li> <li>• Faltou o exercício impresso para alguns discentes.</li> <li>• A linguagem de programação Pascal tem pouca possibilidade de isolamento de código.</li> <li>• As fontes de saída do software Pascalzim no projetor não estavam visíveis para a maioria da plateia.</li> </ul>	<ul style="list-style-type: none"> <li>• A aula terminou com dez minutos de antecedência.</li> </ul>

Fonte: Elaboração própria (2019).

Observamos, do Quadro 20, como pontos fortes: a escrita no papel pelos discentes dos códigos desenvolvidos como proposição para a resolução da questão, enquanto estavam na posição de plateia para utilizarem quando assumiam a posição de piloto e copiloto; e a colaboração entre discentes para a resolução dos problemas propostos.

Os pontos negativos para esse dia de prática foram: não explicar sobre o fluxo programar-testar, o qual consiste no desenvolvimento de pequenas funcionalidades e logo após a realização do teste; pouca participação dos discentes no decorrer da prática, em geral o ambiente de Dojo solicita uma abordagem participativa das pessoas envolvidas, porém os alunos tinham pouca interação com quem estava no papel do piloto e copiloto; Questões estruturadas não estimulam a participação dos alunos, uma vez que o comando indica o caminho a ser tomado para a solução do problema, por consequência deduzimos ser um dos motivos para a pouca participação dos personagens envolvidos; A função de copiloto não funcionou como o esperado, pois no decorrer da prática os discentes que estavam no papel de piloto não informavam os próximos passos para que o copiloto conseguisse auxiliar e, por consequência, narrar para a plateia; Algumas pessoas tiveram que ser indicadas para a realização da prática; Houve a formação de pequenos grupos, os quais buscavam a solução entre si e não

compartilhavam com os outros integrantes da plateia; Os papéis com os desafios para serem desenvolvidos foram distribuídos para grupos de discentes, porém consideramos para um próximo uso, a necessidade de disponibilização de uma impressão por discente; A linguagem de programação Pascal tem pouco isolamento de código, e por consequência, dificulta, mas não impossibilita o fluxo programar-testar planejado para atividade; A fonte de saída do programa Pascalzim não estava disponível para toda a plateia.

Como foi possível analisar, a prática não se configurou bem para o processo de ensino e aprendizagem. Assim, adotamos como pontos de melhoria para o segundo dia de prática: Foi preparada a aula e nesse processo olhou-se para todas as variáveis envolvidas, inclusive a impressão de cópias das questões para todos os participantes, e a fonte de saída do Pascalzim em relação à apresentação via *datashow*; formulou-se uma questão aberta, para que a turma conseguisse pensar em uma solução por meio de Estrutura de Dados Homogênea; Retirado o papel do copiloto e permitir interação com a plateia do piloto após três minutos de análise e codificação pessoal da questão; reforçar a explicação sobre o papel de piloto, e agora com a prerrogativa dele narrar para a plateia o seu desenvolvimento; só bonificar colaboração quando a solução proposta for repassada para a plateia como um todo e não em grupos isolados na plateia. Nesse dia, a aula finalizou com dez minutos de antecedência.

O segundo dia da prática do Dojo, já com as melhorias identificadas, tem sua análise no Quadro 21.

Quadro 21 – Matriz SWOT adaptada: Dojo *Randori* segundo dia (Continua...)

PONTOS FORTES	MELHORIAS
<ul style="list-style-type: none"> <li>• Prática do Dojo funcionou no processo de ensino e aprendizagem de Algoritmos.</li> <li>• Discentes participaram mais do que no primeiro dia da prática.</li> <li>• Houve maior colaboração entre os discentes.</li> <li>• Um aluno foi ao quadro para explicar um formato de solução.</li> <li>• A turma estava envolvida na prática, colaborou para o bom andamento e finalizar a questão proposta.</li> <li>• A prática foi divertida.</li> <li>• Comando e apoio do professor na atividade.</li> <li>• A questão aberta proporcionou melhoria e alteração de código</li> <li>• Os discentes se sentiram motivados a finalizar o programa.</li> <li>• Pascalzim com fontes de saída visíveis para toda a plateia.</li> </ul>	<ul style="list-style-type: none"> <li>• Bonificar por sugestão os discentes que enviarem via plataforma Moodle até uma data estabelecida outras formas de solução para a questão trabalhada.</li> <li>• Utilizar uma questão semiestruturada para o desenvolvimento da prática.</li> <li>• Quando os discentes não adotarem postura voluntária para a prática, indicar mesclando discentes com bom desempenho nas atividades com os outros.</li> <li>• Explicar melhor o sentido e uso do <i>baby step</i> em algoritmos.</li> <li>• Utilizar unicamente o código desenvolvido para pontuar.</li> <li>• Empregar o uso de bonificação para os papéis adotados pelos discentes, no caso da prática Dojo: piloto, copiloto e plateia.</li> </ul>

Fonte: Elaboração própria (2019).

Quadro 21 – Matriz SWOT adaptada: Dojo *Randori* segundo dia (Conclusão)

PONTOS FRACOS	OBSERVAÇÕES
<ul style="list-style-type: none"> <li>• Durante a prática não foi possível observar como pontuar sugestão.</li> <li>• Questão aberta inibiu alguns alunos de participarem e não possibilitou uma sequência de funcionalidades programadas.</li> <li>• Linguagem de programação Pascal tem pouco isolamento de código.</li> <li>• Discentes não conseguiam visualizar possibilidades de teste para as funcionalidades codificadas como solução para a questão.</li> </ul>	<ul style="list-style-type: none"> <li>• A prática do Dojo funciona para o processo de ensino e aprendizagem de Algoritmos.</li> </ul>

Fonte: Elaboração própria (2019).

Constatamos, com base no Quadro 21, que, a partir das mudanças realizadas, a prática funcionou de forma satisfatória. Além disso, como pontos fortes: discentes participaram mais que no primeiro dia como é possível identificar nos dados sobre bonificação e penalidades e nos resultados quantitativos, seções 4.2.1 e 4.2.2, respectivamente; a prática foi divertida, observamos que os discentes estavam se divertindo programando; o comando e o apoio do professor na atividade, estimulavam os discentes para a participação dentro da prática; questão aberta permitiu com que vários discentes fizessem melhoria e alterações no código, bem como um aluno que foi ao quadro para apresentar o pensamento dele adotado para a resolução; discentes se envolveram, ou seja, estavam dispostos a finalizar o programa em aula, vale o destaque para os que tiveram participação voluntária. Com pontos fracos identificamos: quesito sugestão não foi possível bonificar; a questão aberta impossibilitou a participação de todos os discentes, bem como conseguirem cada um desenvolver uma funcionalidade; a linguagem de programação adotada, que no caso é o Pascal, tem pouco isolamento de código; os discentes não conseguiam visualizar possibilidades de teste na questão.

Para o segundo dia, analisamos os dados e propomos como melhorias: a bonificação por sugestão ser por meio do envio de novas formas de resolução da questão desenvolvida no conjunto da turma; Elaborar uma questão semiestruturada para o próxima prática de Dojo; e disponibilizar o monitor para auxiliar os discentes com sugestão para gerar o fluxo programar-testar; Explicar melhor sobre o fluxo programar-testar com dois exemplos desenvolvidos com eles; Adotar para o próximo uso ou versão do *framework* nessa prática bonificação pela função dos papéis e não pontuação, bem como pontuar unicamente o código desenvolvido pelos discentes. Portanto, de maneira geral, com as mudanças adotadas, observamos que a prática Dojo funciona para o ensino de algoritmos e aprendizagem de algoritmos.

A avaliação da prática LAB está disposta no Quadro 22.

Quadro 22 – Matriz SWOT adaptada: Prática LAB

PONTOS FORTES	MELHORIAS
<ul style="list-style-type: none"> <li>• Uso da prática de <i>coaching</i>.</li> <li>• Os alunos resolverem as questões propostas em duplas.</li> <li>• Sorteio dos pares em sala de aula, por meio de uma ferramenta online.</li> <li>• Proibir o acesso à internet.</li> </ul>	<ul style="list-style-type: none"> <li>• Equilibrar o nível de dificuldade das questões na prática com o de aprendizado da turma.</li> <li>• Uso da bonificação por: (i) interação entre as duplas; (ii) narrar para o colega de dupla o código desenvolvido; e (iii) trocar durante a prática o papel de piloto e copiloto nas duplas.</li> </ul>
PONTOS FRACOS	OBSERVAÇÕES
<ul style="list-style-type: none"> <li>• Questões com o nível de dificuldade maior que o de aprendizado da turma.</li> <li>• Não foi bonificado a interação entre as duplas na prática.</li> <li>• Não foi possível observar narração dos códigos desenvolvidos e troca de função nas duplas formadas.</li> </ul>	<ul style="list-style-type: none"> <li>• Permitido a prática de <i>coaching</i>.</li> <li>• Durante a prática foi possível observar duas funções nas duplas: (i) piloto, estava no computador e programava a solução; e (ii) copiloto, estava acompanhando o desenvolvimento do código e propunha melhorias.</li> <li>• No início da prática um grupo estava com três discentes. Após dez minutos foi desfeito para formar duas duplas, pois outro discente chegou.</li> </ul>

Fonte: Elaboração própria (2019).

Conforme **Quadro 22**, dentre os pontos positivos destacamos: (i) a prática de *coaching*, no momento os outros personagens da *gamificação* auxiliavam os jogadores com dicas para a resolução dos desafios; (ii) a resolução dos desafios ocorrerem em duplas, pois permitiu no decorrer da prática uma análise do problema computacional bem como sua resolução por dois estudantes; (iii) sorteio para a formação das duplas por meio de uma ferramenta *online*, com o intuito de diminuir o viés; e (iv) proibir o acesso à internet, permitiu aos discentes focarem em desenvolver estratégias de resolução dos problemas. Como pontos fracos, temos: (i) as questões desafio com nível de dificuldade muito elevado, sendo este fato retratado pelos discentes também na aula de *feedback*; e (ii) não foi bonificado o comportamento dos discentes no decorrer da prática, bem como os discentes que iniciaram com o comando do computador finalizaram a prática.

Como melhorias da prática de LAB, descrevemos: equivaler às questões da prática com o nível de aprendizado da turma; bonificar o comportamento dos alunos na prática em relação: (i) interação entre as duplas; (ii) narrar para o colega de dupla o código desenvolvido; e (iii) trocar durante a prática o papel de piloto e copiloto nas duplas.

Na aula de *feedback* sobre a prática LAB, os discentes levantaram como pontos de melhoria: (i) para as questões trabalhadas em sala de aula aumentar gradativamente o nível de dificuldade na sequência de exibição; e (ii) em relação à lista de exercício para casa, informaram a necessidade de um tempo dentro da aula teórica e prática para discutir como resolver cada uma das questões, com possibilidades de dicas.

Em relação aos cenários de aprendizagem adotados, no caso o Dojo *Randori* e o LAB, os discentes, durante a aula de *feedback*, afirmaram ser uma experiência positiva para o



aprendizado dentro da disciplina. Vale ressaltar que eles solicitaram que tais práticas pudessem ser adotadas em todos os conteúdos trabalhados na disciplina.

No caso do teste teórico e da aula de *feedback*, não foram elaboradas matrizes SWOTs adaptadas, mas coletamos informações sobre elas. A primeira prática permitiu a avaliação de desempenho individual dos discentes após as práticas *gamificadas*, e tem seu formato regido, como o professor já trabalhava esse elemento em sala. A segunda foi uma experiência diferencial para os discentes em que apontaram sugestões e reafirmaram o uso das novas práticas para o ensino e aprendizado de Algoritmos.

De maneira geral, sobre as metodologias utilizadas no experimento, os discentes afirmaram que a utilização de uma sala de aula *gamificada* para o ensino do conteúdo de estrutura de dados na disciplina de Algoritmos foi muito válida. Por meio da abordagem, os discentes tiveram a visão teórica e prática necessárias do assunto. Vale ressaltar que todos os pontos levantados pelos personagens da *gamificação* foram amplamente discutidos e sistematizados nesta dissertação com a finalidade de aprimorar o *framework gamificado* para o processo de ensino e aprendizagem na disciplina Algoritmos.

#### 4.2.1.2 Dados sobre bonificações e penalidades

Na Figura 38 temos informações a respeito da presença dos discentes em sala de aula. Conforme já citado, cada aula frequentada representava dez bônus aos discentes, cujo propósito era estimular a presença deles em sala. Porém, ainda assim, houve alunos que faltaram às aulas *gamificadas*.

Figura 39 – Presença dos discentes durante as aulas

Membro	Presença							TOTAL
	Aula 1	Aula 2	Aula 3	Aula 4	Aula 5	Aula 6	Aula 7	
Aluno 01	0	1	1	1	0	1	1	50
Aluno 02	1	1	0	1	1	1	0	50
Aluno 03	1	1	1	1	1	1	1	70
Aluno 04	1	1	1	1	1	1	1	70
Aluno 05	1	1	1	1	1	1	1	70
Aluno 06	1	1	1	1	1	1	1	70
Aluno 07	1	1	1	1	1	1	1	70
Aluno 08	1	1	1	1	0	1	1	60
Aluno 09	1	1	1	1	1	1	1	70
Aluno 10	1	1	1	1	1	1	1	70
Aluno 11	1	1	1	1	1	1	1	70
Aluno 12	1	1	1	1	1	1	1	70
Aluno 13	1	1	1	1	1	1	1	70
Aluno 14	1	1	1	1	0	1	1	60
Aluno 15	1	1	1	1	1	1	1	70
Aluno 16	1	1	1	1	1	1	1	70
Aluno 17	0	1	0	1	0	0	0	20
Aluno 18	1	1	1	0	1	1	1	60
Aluno 19	1	0	1	1	0	1	1	50
Aluno 20	1	0	1	1	1	1	0	50

Fonte: Elaboração própria (2019).

A Figura 40 apresenta a planilha de participação dos discentes em sala de aula. De acordo com a figura, observamos que, mesmo quando os discentes conseguiam alcançar a meta diária (bonificação até cinco vezes por aula), continuavam participando das aulas. Esse envolvimento pode ser percebido, por exemplo, na Aula 4 (segundo dia da prática de Dojo Randori).

Figura 40 – Participações dos discentes durante as aulas

Membro	Participações							TOTAL
	Aula 1	Aula 2	Aula 3	Aula 4	Aula 5	Aula 6	Aula 7	
Aluno 01							3	6
Aluno 02	2			2				8
Aluno 03	3	4	1	3			6	34
Aluno 04	4	1	1	5				22
Aluno 05	2	2	3	8			3	36
Aluno 06	2			1			5	16
Aluno 07	4	5	4	7				40
Aluno 08	3	5	4	9				42
Aluno 09	2	1	2				5	20
Aluno 10	3	1	1	2			4	22
Aluno 11	2		1					6
Aluno 12	5	3	3	10			8	58
Aluno 13	2			1				6
Aluno 14	2			1				6
Aluno 15	2		1					6
Aluno 16	2	1	3	4			11	42
Aluno 17								0
Aluno 18	2	2					3	14
Aluno 19	2		4	7			4	34
Aluno 20	2			1				6

Fonte: Elaboração própria (2019).

As bonificações referentes ao quesito sugestão, atribuída pelo juiz/*scoreplay* ao discente, está disponível na Figura 41. Percebe-se que a meta prevista (até cinco bonificações por dia) não foi alcançada pelos discentes em nenhum dos dias de aula. Destacamos que nas Aulas 3, 4 e 5, houve um acréscimo nas bonificações, por ter sido permitido o envio de sugestões para a melhoria de código produzido em sala durante um período de 72 horas após o encontro. Na Aula 5 este envio não foi computado como recurso estrela, porque na regra da *gamificação*, nessa aula não seria concedido este recurso.

Figura 41 – Sugestões dos discentes em sala de aula

Membro	Sugestões							TOTAL
	Aula 1	Aula 2	Aula 3	Aula 4	Aula 5	Aula 6	Aula 7	
Aluno 01				1				2
Aluno 02								0
Aluno 03			1	1			4	12
Aluno 04		1	3	3	3			20
Aluno 05		4		1	3			16
Aluno 06		1			2		2	10
Aluno 07	1	1		2	1			10
Aluno 08		1	1	2	1			10
Aluno 09			1				2	6
Aluno 10							2	4
Aluno 11								0
Aluno 12		2	3		4			18
Aluno 13					1		3	8
Aluno 14								0
Aluno 15								0
Aluno 16				1			3	8
Aluno 17								0
Aluno 18								0
Aluno 19				1				2
Aluno 20								0

Fonte: Elaboração própria (2019).

Durante as aulas teóricas e práticas, ao analisarmos a Figura 42, em conjunto às matrizes de análise SWOTs adaptadas das aulas, observamos a importância do elemento de jogo, bonificação por colaboração, para o processo de ensino e aprendizado no *framework gamificado*. A colaboração, como quesito de bonificação, estimulou um comportamento positivo entre os discentes, pois possibilitou o aprendizado pareado. Na Aula 3, Dojo *Randori*, a colaboração era bonificada quando pequenos grupos de discentes socializavam entre si a informação. Já na aula 4, em que foi proposto alterações no formato da prática, a bonificação era registrada pelo juiz/*scoreplay* ao passo que a solução era compartilhada com toda a turma.

Figura 42 – Colaborações dos discentes durante as aulas

Membro	Colaboração							TOTAL
	Aula 1	Aula 2	Aula 3	Aula 4	Aula 5	Aula 6	Aula 7	
Aluno 01		2	1	1				8
Aluno 02		3		1				8
Aluno 03		2	1	2				10
Aluno 04	5	5	3	1				28
Aluno 05	4	5	3	2				28
Aluno 06		2		2				8
Aluno 07	4	1	1	2				16
Aluno 08	5	5	1	1				24
Aluno 09	5	2		1				16
Aluno 10	1			1				4
Aluno 11	1			1				4
Aluno 12	5		5	1				22
Aluno 13	1		5	1				14
Aluno 14		2		1				6
Aluno 15				1				2
Aluno 16		2		1				6
Aluno 17		1		1				4
Aluno 18	1	2						6
Aluno 19			3	1				8
Aluno 20			1	1				4

Fonte: Elaboração própria (2019).

A Figura 43 apresenta o elemento ‘faltas’, configurado para esse *framework* como penalidade. Logo, a imagem apresenta os alunos ausentes na abordagem *gamificada*.

Figura 43 – Faltas dos discentes no decorrer das aulas

Membro	Falta							TOTAL
	Aula 1	Aula 2	Aula 3	Aula 4	Aula 5	Aula 6	Aula 7	
Aluno 01	1	0	0	0	1	0	0	20
Aluno 02	0	0	1	0	0	0	1	20
Aluno 03	0	0	0	0	0	0	0	0
Aluno 04	0	0	0	0	0	0	0	0
Aluno 05	0	0	0	0	0	0	0	0
Aluno 06	0	0	0	0	0	0	0	0
Aluno 07	0	0	0	0	0	0	0	0
Aluno 08	0	0	0	0	1	0	0	10
Aluno 09	0	0	0	0	0	0	0	0
Aluno 10	0	0	0	0	0	0	0	0
Aluno 11	0	0	0	0	0	0	0	0
Aluno 12	0	0	0	0	0	0	0	0
Aluno 13	0	0	0	0	0	0	0	0
Aluno 14	0	0	0	0	1	0	0	10
Aluno 15	0	0	0	0	0	0	0	0
Aluno 16	0	0	0	0	0	0	0	0
Aluno 17	1	0	1	0	1	1	1	50
Aluno 18	0	0	0	1	0	0	0	10
Aluno 19	0	1	0	0	1	0	0	20
Aluno 20	0	1	0	0	0	0	1	20

Fonte: Elaboração própria (2019).

A Figura 44 apresenta o elemento atraso em dez minutos com no máximo quatro por aula. Esse quesito foi inserido como forma de estímulo positivo para que os discentes chegassem cedo à aula. Ao analisar os dados da observação participante e o formato adotado na explicação teórica do *framework*, foi verificado que o discente atrasado perdia a explicação sobre o conteúdo, a qual em geral se discorria nos primeiros vinte minutos de aula. Vale

ressaltar que apenas os Alunos 01, 02 e 20 sofreram essa penalidade. Ou seja, os outros discentes absorveram como estímulo para a sua presença em sala.

Figura 44 – Penalidade “atraso em 10 minutos” no decorrer das aulas

Membro	Atraso em 10 minutos							TOTAL
	Aula 1	Aula 2	Aula 3	Aula 4	Aula 5	Aula 6	Aula 7	
Aluno 01			1					2
Aluno 02	2							4
Aluno 03								0
Aluno 04								0
Aluno 05								0
Aluno 06								0
Aluno 07								0
Aluno 08								0
Aluno 09								0
Aluno 10								0
Aluno 11								0
Aluno 12								0
Aluno 13								0
Aluno 14								0
Aluno 15								0
Aluno 16								0
Aluno 17								0
Aluno 18								0
Aluno 19								0
Aluno 20	2							4

Fonte: Elaboração própria (2019).

A Figura 45 apresenta a penalidade pelo uso do celular em sala de aula. Como podemos perceber, apenas três discentes sofreram essa pena: os Alunos 14, 17 e 20. É importante destacar que após sofrerem a sanção uma vez, não adotaram esse comportamento de novo.

Figura 45 – Uso do celular durante as aulas

Membro	Uso de Celular							TOTAL
	Aula 1	Aula 2	Aula 3	Aula 4	Aula 5	Aula 6	Aula 7	
Aluno 01								0
Aluno 02								0
Aluno 03								0
Aluno 04								0
Aluno 05								0
Aluno 06								0
Aluno 07								0
Aluno 08								0
Aluno 09								0
Aluno 10								0
Aluno 11								0
Aluno 12								0
Aluno 13								0
Aluno 14				1				2
Aluno 15								0
Aluno 16								0
Aluno 17	1							2
Aluno 18								0
Aluno 19								0
Aluno 20			1					2

Fonte: Elaboração própria (2019).

A penalidade por utilizar o computador para outras atividades está presente na Figura 46. Nesse quesito se estabeleceu como regra o acesso à *internet* unicamente para visualizar a

plataforma Moodle da UFPA, sala de aula estendida da disciplina, e o uso do Pascalzim, software adotado como recurso de ensino. Destacamos nesse item o registro de apenas dois discentes no dia da aula de *feedback*, os quais estavam reexecutando códigos em Pascal. Mas como a prática do dia era a resposta dos alunos em relação ao *feedback* e não à codificação ou visualização de programas, então os discentes foram penalizados.

Figura 46 – Uso do computador para outras atividades

Membro	Uso do Computador							TOTAL
	Aula 1	Aula 2	Aula 3	Aula 4	Aula 5	Aula 6	Aula 7	
Aluno 01								0
Aluno 02								0
Aluno 03								0
Aluno 04							1	2
Aluno 05								0
Aluno 06								0
Aluno 07								0
Aluno 08								0
Aluno 09								0
Aluno 10								0
Aluno 11								0
Aluno 12								0
Aluno 13								0
Aluno 14								0
Aluno 15								0
Aluno 16								0
Aluno 17								0
Aluno 18							1	2
Aluno 19								0
Aluno 20								0

Fonte: Elaboração própria (2019).

A Figura 47 apresenta a planilha referente às penalidades por atrapalhar a aula. Nesse quesito observamos três ocorrências, uma na Aula 3 e duas na Aula 4. Vale destacar a aplicação da penalidade para os discentes que estavam conversados no decorrer da atividade.

Figura 47 – Penalidade por atrapalhar a aula

Membro	Penalidade por atrapalhar a aula							TOTAL
	Aula 1	Aula 2	Aula 3	Aula 4	Aula 5	Aula 6	Aula 7	
Aluno 01								0
Aluno 02								0
Aluno 03								0
Aluno 04								0
Aluno 05								0
Aluno 06								0
Aluno 07								0
Aluno 08								0
Aluno 09								0
Aluno 10								0
Aluno 11								0
Aluno 12								0
Aluno 13								0
Aluno 14								0
Aluno 15								0
Aluno 16				1				2
Aluno 17								0
Aluno 18								0
Aluno 19			1	1				4
Aluno 20								0

Fonte: Elaboração própria (2019).

No que tange às penalidades, de maneira geral, podemos observar que os discentes seguiram as regras estabelecidas na *gamificação*. De maneira geral, os discentes não sofreram as penalidades previstas, pois não adotaram comportamentos passíveis de penalidade no decorrer do processo em sala de aula.

#### 4.2.2 Discussão e resultados quantitativos

Os dados quantitativos apoiam os qualitativos a fim de avaliar o *framework gamificado* para desenvolver a próxima versão assim como para os próximos usos. Nessa discussão, a seção foi dividida nas seguintes subseções: resultados e discussão em relação ao conteúdo; dados relacionados à atribuição de pontos e *avatar*; resultados e discussão em relação ao comportamento; informações relativas ao elemento bônus e ao recurso estrela adquiridos pelos discentes; o elemento medalha, prêmio para os discentes mais bem colocados.

##### 4.2.2.1 Resultados e discussão em relação ao conteúdo

O *score* de *gamificação* é formado pelo desempenho do discente em relação ao conteúdo. Nesse caso, à medida que os discentes respondiam as listas de desafio, ganhavam pontos dentro da estrutura *gamificada* (vide Figura 48). Em geral, a *gamificação* possibilitou uma progressão de aprendizagem no conteúdo da disciplina e tal situação é perceptível ao observarmos os alunos 12 e 20. Os discentes com o desempenho considerado não muito

satisfatório, no caso, receberam o *avatar* Amador e Aprendiz. Em geral, são discentes não presentes em sala de aula (vide seção 4.2.1.2 do trabalho em tela).

Figura 48 – Scoreplay da gamificação

	A	B	C	D	E	F	G	H	I
1	Membro	Lista Exercício 01	Lista Exercício 02	Lista Exercício Casa	Dojo	LAB	Prova Teórica	TOTAL	AVATAR FINAL
2	Aluno 01	0	75	0	66,75	0	0	141,75	AMADOR
3	Aluno 02	100	75	0	143,75	85	11,25	415	AMADOR
4	Aluno 03	100	50	0	193,75	155	95	593,75	APRENDIZ
5	Aluno 04	100	100	99	177,5	200	290	966,5	ANALISTA
6	Aluno 05	100	100	86	193,75	198	300	977,75	ANALISTA
7	Aluno 06	100	100	67	198,5	180	300	945,5	ANALISTA
8	Aluno 07	100	100	86	197	140	275	898	PROGRAMADOR
9	Aluno 08	100	100	80	193,75	0	275	748,75	PROGRAMADOR
10	Aluno 09	100	62,5	0	177,75	170	55	565,25	APRENDIZ
11	Aluno 10	100	75	0	193,75	50	185	603,75	APRENDIZ
12	Aluno 11	100	100	58	143,75	150	300	851,75	PROGRAMADOR
13	Aluno 12	100	75	88	195,5	200	300	958,5	ANALISTA
14	Aluno 13	100	100	68	193,75	198	275	934,75	ANALISTA
15	Aluno 14	75	75	0	112,75	0	275	537,75	APRENDIZ
16	Aluno 15	75	75	0	127,75	145	245	667,75	APRENDIZ
17	Aluno 16	100	62,5	0	183,75	85	75	506,25	APRENDIZ
18	Aluno 17	0	50	0	66,75	0	0	116,75	AMADOR
19	Aluno 18	75	75	0	66,75	50	155	421,75	AMADOR
20	Aluno 19	100	0	0	193,75	140	260	693,75	APRENDIZ
21	Aluno 20	100	0	24	162,75	70	147,5	504,25	APRENDIZ
22									

Fonte: Elaboração própria (2019).

Os resultados obtidos no primeiro dia da aula teórica e prática estão na Figura 49. Observamos que a maioria dos discentes presentes em sala de aula conseguiu finalizar a atividade em sala, exceto os Alunos 14, 15 e 18. Nesse dia o elemento de colaboração se destacou, favorecendo a interação entre os discentes, que se ajudaram para finalizar as questões.

Figura 49 – Resultados da Aula Teórica e Prática primeiro dia

	A	B	C	D	E	F	G
1	Membros	Resultado Pontuação Aula Teórica e Prática - 1º Dia				TOTAL	AVATAR DIA
2		1º Questão	2º Questão	3º Questão	4º Questão		
3	Aluno 01	0	0	0	0	0	AMADOR
4	Aluno 02	1	1	1	1	100	ANALISTA
5	Aluno 03	1	1	1	1	100	ANALISTA
6	Aluno 04	1	1	1	1	100	ANALISTA
7	Aluno 05	1	1	1	1	100	ANALISTA
8	Aluno 06	1	1	1	1	100	ANALISTA
9	Aluno 07	1	1	1	1	100	ANALISTA
10	Aluno 08	1	1	1	1	100	ANALISTA
11	Aluno 09	1	1	1	1	100	ANALISTA
12	Aluno 10	1	1	1	1	100	ANALISTA
13	Aluno 11	1	1	1	1	100	ANALISTA
14	Aluno 12	1	1	1	1	100	ANALISTA
15	Aluno 13	1	1	1	1	100	ANALISTA
16	Aluno 14	1	1	1	0	75	PROGRAMADOR
17	Aluno 15	1	1	1	0	75	PROGRAMADOR
18	Aluno 16	1	1	1	1	100	ANALISTA
19	Aluno 17	0	0	0	0	0	AMADOR
20	Aluno 18	1	1	1	0	75	PROGRAMADOR
21	Aluno 19	1	1	1	1	100	ANALISTA
22	Aluno 20	1	1	1	1	100	ANALISTA

Fonte: Elaboração própria (2019).



No segundo dia de aula teórica e prática temos resultados apresentados na Figura 50. Para esse dia tivemos uma alteração, na qual a atribuição para a resolução da questão foi de 0,5, o que concedia ao aluno a pontuação de 12,5 pontos. Essa alteração foi concedida após uma conversa do pesquisador responsável pela *gamificação* com a professora, e atribuída para os discentes que não completavam a resolução da questão. Pois a planilha estava programada para preenchimento automático do *scoreplay*. Nesse dia também 50% da turma não conseguiu finalizar toda a atividade, um dos pontos para esse acontecimento foi registrado no SWOT adaptado diário da atividade (vide Quadro 19).

Figura 50 – Resultados da Aula Teórica e Prática segundo dia

23	24	Membros	Resultado Pontuação Aula Teórica e Prática - 2º Dia				TOTAL	AVATAR DIA
			1º Questão	2º Questão	3º Questão	4º Questão		
25	Aluno 01	1	1	1	0	75	PROGRAMADOR	
26	Aluno 02	1	1	1	0	75	PROGRAMADOR	
27	Aluno 03	1	1	0	0	50	AMADOR	
28	Aluno 04	1	1	1	1	100	ANALISTA	
29	Aluno 05	1	1	1	1	100	ANALISTA	
30	Aluno 06	1	1	1	1	100	ANALISTA	
31	Aluno 07	1	1	1	1	100	ANALISTA	
32	Aluno 08	1	1	1	1	100	ANALISTA	
33	Aluno 09	1	1	0,5	0	62,5	APRENDIZ	
34	Aluno 10	1	1	1	0	75	PROGRAMADOR	
35	Aluno 11	1	1	1	1	100	ANALISTA	
36	Aluno 12	1	1	1	0	75	PROGRAMADOR	
37	Aluno 13	1	1	1	1	100	ANALISTA	
38	Aluno 14	1	1	1	0	75	PROGRAMADOR	
39	Aluno 15	1	1	1	0	75	PROGRAMADOR	
40	Aluno 16	1	1	0,5	0	62,5	APRENDIZ	
41	Aluno 17	1	1	0	0	50	AMADOR	
42	Aluno 18	1	1	1	0	75	PROGRAMADOR	
43	Aluno 19	0	0	0	0	0	AMADOR	
44	Aluno 20	0	0	0	0	0	AMADOR	

Fonte: Elaboração própria (2019).

A prática de Dojo *Randori* aconteceu em dois dias e o resultado das pontuações pode ser visto na Figura 51. Os preenchimentos decimais dentro da aba da planilha ocorreram quando os discentes no decorrer da prática não atendiam inteiramente o quesito solicitado, e à medida que era preenchida, a planilha calculava a proporcionalidade e acrescentava ao valor total. Após os resultados apresentados e a forma como a prática é descrita na literatura como melhoria para a próxima execução do *framework gamificado*, decidimos pontuar unicamente pelo código desenvolvido e remover os outros critérios que serão devidamente bonificados à medida da execução da prática.

Figura 51 – Resultados da prática Dojo *Randori*

	A	B	C	D	E	F	G	H
1	DOJO							
2	Membros	Uma funcionalidade concluída	Utilizou o fluxo Programar-Testa	Narrou a programação	Nota geral das questões	TOTAL	AVATAR	
3	Aluno 01	0,2	0,3	0,1	43,75	66,75	AMADOR	
4	Aluno 02	1	1	0,5	43,75	143,75	APRENDIZ	
5	Aluno 03	1	1	1	43,75	193,75	ANALISTA	
6	Aluno 04	1	0,3	1	48,5	177,5	PROGRAMADOR	
7	Aluno 05	1	1	1	43,75	193,75	ANALISTA	
8	Aluno 06	1	1	1	48,5	198,5	ANALISTA	
9	Aluno 07	1	1	1	47	197	ANALISTA	
10	Aluno 08	1	1	1	43,75	193,75	ANALISTA	
11	Aluno 09	0,2	1	1	43,75	177,75	PROGRAMADOR	
12	Aluno 10	1	1	1	43,75	193,75	ANALISTA	
13	Aluno 11	1	1	0,5	43,75	143,75	APRENDIZ	
14	Aluno 12	1	1	1	45,5	195,5	ANALISTA	
15	Aluno 13	1	1	1	43,75	193,75	ANALISTA	
16	Aluno 14	0,5	0,3	0,5	43,75	112,75	AMADOR	
17	Aluno 15	0,2	1	0,5	43,75	127,75	APRENDIZ	
18	Aluno 16	0,5	1	1	43,75	183,75	ANALISTA	
19	Aluno 17	0,2	0,3	0,1	43,75	66,75	AMADOR	
20	Aluno 18	0,2	0,3	0,1	43,75	66,75	AMADOR	
21	Aluno 19	1	1	1	43,75	193,75	ANALISTA	
22	Aluno 20	0,5	0,3	1	43,75	162,75	PROGRAMADOR	
23								

Fonte: Elaboração própria (2019).

No *framework gamificado*, a lista de exercício para casa é pontuada em conjunto com a prática LAB e seus resultados estão apresentados na Figura 52. O recurso alcançou 50 por cento (50%) dos estudantes e nenhum discente conseguiu alcançar os 100 pontos destinados à lista. Sobre essa situação, durante a aula de *feedback*, os discentes afirmaram que a lista estava com um nível de complexidade muito alto, por isso, nem tentaram entregar. Entretanto, informaram à necessidade de um tempo na aula teórica e prática para dicas na resolução do recurso.

Figura 52 – Resultados da Lista de Exercício para casa

	A	B	C	D	E	F	G	H	I	J	K	L	M
45	Resultado Pontuação Lista Exercício Casa												
46	Membros	1° Questão	2° Questão	3° Questão	4° Questão	5° Questão	6° Questão	7° Questão	8° Questão	9° Questão	10° Questão	TOTAL	AVATAR DIA
47	Aluno 01	0	0	0	0	0	0	0	0	0	0	0	AMADOR
48	Aluno 02	0	0	0	0	0	0	0	0	0	0	0	AMADOR
49	Aluno 03	0	0	0	0	0	0	0	0	0	0	0	AMADOR
50	Aluno 04	1	1	1	0,9	1	1	1	1	1	1	99	ANALISTA
51	Aluno 05	0,7	1	0	0,9	1	1	1	1	1	1	86	PROGRAMADOR
52	Aluno 06	0,8	0	1	1	0,9	1	1	1	0	0	67	APRENDIZ
53	Aluno 07	1	1	1	1	1	1	1	0,7	0	0,9	86	PROGRAMADOR
54	Aluno 08	1	1	1	1	1	1	1	1	0	0	80	PROGRAMADOR
55	Aluno 09	0	0	0	0	0	0	0	0	0	0	0	AMADOR
56	Aluno 10	0	0	0	0	0	0	0	0	0	0	0	AMADOR
57	Aluno 11	0	0	0	0,8	1	1	1	1	0	1	58	APRENDIZ
58	Aluno 12	1	1	0,8	1	1	1	1	1	0	1	88	PROGRAMADOR
59	Aluno 13	1	0	0,8	0	1	1	1	1	0	1	68	APRENDIZ
60	Aluno 14	0	0	0	0	0	0	0	0	0	0	0	AMADOR
61	Aluno 15	0	0	0	0	0	0	0	0	0	0	0	AMADOR
62	Aluno 16	0	0	0	0	0	0	0	0	0	0	0	AMADOR
63	Aluno 17	0	0	0	0	0	0	0	0	0	0	0	AMADOR
64	Aluno 18	0	0	0	0	0	0	0	0	0	0	0	AMADOR
65	Aluno 19	0	0	0	0	0	0	0	0	0	0	0	AMADOR
66	Aluno 20	0	0,9	0	0,5	0	1	0	0	0	0	24	AMADOR
67													

Fonte: Elaboração própria (2019).

O resultado da prática LAB está apresentado na planilha da Figura 53. O *avatar* analista foi concedido para apenas três discentes, são eles 04, 05 e 12. O fato colabora com a informação

já presente no SWOT diário da prática (Quadro 22) e relatada na aula de *feedback*. Os discentes apontaram que as questões desafios do dia estavam com um nível muito elevado para a turma.

Figura 53 – Resultados da prática Laboratório de Programação

89	A	Resultado Pontuação LAB				F	G	H	I
		B	C	D	E				
90	Membros	1° Desafio	2° Desafio	3° Desafio	4° Desafio	TOTAL	TOTAL LISTA EXERCÍCIO CASA	TOTAL GERAL LAB	AVATAR DIA
91	Aluno 01	0	0	0	0	0	0	0	AMADOR
92	Aluno 02	50	35	0	0	85	0	85	AMADOR
93	Aluno 03	50	50	30	25	155	0	155	AMADOR
94	Aluno 04	50	50	50	50	200	99	299	ANALISTA
95	Aluno 05	50	50	48	50	198	86	284	ANALISTA
96	Aluno 06	50	30	50	50	180	67	247	PROGRAMADOR
97	Aluno 07	50	50	20	20	140	86	226	APRENDIZ
98	Aluno 08	0	0	0	0	0	80	80	AMADOR
99	Aluno 09	50	50	45	25	170	0	170	AMADOR
100	Aluno 10	50	0	0	0	50	0	50	AMADOR
101	Aluno 11	50	0	50	50	150	58	208	APRENDIZ
102	Aluno 12	50	50	50	50	200	88	288	ANALISTA
103	Aluno 13	50	50	48	50	198	68	266	PROGRAMADOR
104	Aluno 14	0	0	0	0	0	0	0	AMADOR
105	Aluno 15	50	50	45	0	145	0	145	AMADOR
106	Aluno 16	50	35	0	0	85	0	85	AMADOR
107	Aluno 17	0	0	0	0	0	0	0	AMADOR
108	Aluno 18	50	0	0	0	50	0	50	AMADOR
109	Aluno 19	50	50	20	20	140	0	140	AMADOR
110	Aluno 20	50	20	0	0	70	24	94	AMADOR
111									

Fonte: Elaboração própria (2019).

Na Figura 54 temos o resultado em relação ao desempenho dos discentes em cada uma das questões da prática do LAB. Para essa atividade, configuramos o elemento tentativa, em que se o estudante ao menos tivesse tentando resolver a questão, era atribuída uma pontuação para o mesmo; na figura é representado pela coluna atividade concluída, e possuía como preenchimento 0 ou 1. A outra coluna era a nota do desafio, no caso com preenchimento de 0 a 30, pontuação concedida ao discente pelo código desenvolvido. Essa informação era disponibilizada aos alunos por meio da planilha *gamificada*.

No que tange a informação de algumas pessoas que pontuaram unicamente na nota do desafio, ocorreu, pois, a permissão para os alunos enviarem as atividades que não tinham conseguido terminar em sala via plataforma Moodle. Porém, para essa interação não seria mais atribuído o ponto pela tentativa, uma vez que estaria fora do contexto da prática. Alteração efetiva de nota só tivemos do aluno 03. Em relação ao envio de código tivemos também o aluno 06, a qual bonificou por sugestão por já ter concluído a questão em sala e ter proposto uma nova forma de resolução.

A melhoria para a próxima versão do *framework* é desenvolver desafios compatíveis ao nível de aprendizado da turma. Sobre a extensão do prazo para o envio das atividades não observamos como algo válido.

Figura 54 – Resultados LAB questões individuais

LAB - 1º Desafio				LAB - 2º Desafio				LAB 02 - 3º Desafio				LAB 02 - 4º Desafio			
Membros	Atividade concluída	Nota desafio	TOTAL	Membros	Atividade concluída	Nota desafio	TOTAL	Membros	Atividade concluída	Nota desafio	TOTAL	Membros	Atividade	Nota desafio	TOTAL
Aluno 01	0	0	0	Aluno 01	0	0	0	Aluno 01	0	0	0	Aluno 01	0	0	0
Aluno 02	1	30	50	Aluno 02	1	15	35	Aluno 02	0	0	0	Aluno 02	0	0	0
Aluno 03	1	30	50	Aluno 03	1	30	50	Aluno 03	0	30	30	Aluno 03	0	25	25
Aluno 04	1	30	50	Aluno 04	1	30	50	Aluno 04	1	30	50	Aluno 04	1	30	50
Aluno 05	1	30	50	Aluno 05	1	30	50	Aluno 05	1	28	48	Aluno 05	1	30	50
Aluno 06	1	30	50	Aluno 06	0	30	30	Aluno 06	1	30	50	Aluno 06	1	30	50
Aluno 07	1	30	50	Aluno 07	1	30	50	Aluno 07	1	0	20	Aluno 07	1	0	20
Aluno 08	0	0	0	Aluno 08	0	0	0	Aluno 08	0	0	0	Aluno 08	0	0	0
Aluno 09	1	30	50	Aluno 09	1	30	50	Aluno 09	1	25	45	Aluno 09	0	25	25
Aluno 10	1	30	50	Aluno 10	0	0	0	Aluno 10	0	0	0	Aluno 10	0	0	0
Aluno 11	1	30	50	Aluno 11	0	0	0	Aluno 11	1	30	50	Aluno 11	1	30	50
Aluno 12	1	30	50	Aluno 12	1	30	50	Aluno 12	1	30	50	Aluno 12	1	30	50
Aluno 13	1	30	50	Aluno 13	1	30	50	Aluno 13	1	28	48	Aluno 13	1	30	50
Aluno 14	0	0	0	Aluno 14	0	0	0	Aluno 14	0	0	0	Aluno 14	0	0	0
Aluno 15	1	30	50	Aluno 15	1	30	50	Aluno 15	1	25	45	Aluno 15	0	0	0
Aluno 16	1	30	50	Aluno 16	1	15	35	Aluno 16	0	0	0	Aluno 16	0	0	0
Aluno 17	0	0	0	Aluno 17	0	0	0	Aluno 17	0	0	0	Aluno 17	0	0	0
Aluno 18	1	30	50	Aluno 18	0	0	0	Aluno 18	0	0	0	Aluno 18	0	0	0
Aluno 19	1	30	50	Aluno 19	1	30	50	Aluno 19	1	0	20	Aluno 19	1	0	20
Aluno 20	1	30	50	Aluno 20	1	0	20	Aluno 20	0	0	0	Aluno 20	0	0	0

Fonte: Elaboração própria (2019).

A Figura 55 apresenta o resultado da atividade avaliativa individual realizada dentro do *framework* como o teste teórico. Nove dos 20 alunos conseguiram chegar ao *avatar* analista e dois ao programador. Essa informação indica o sucesso de se trabalhar a *gamificação* no processo de ensino e aprendizagem de Algoritmos. Mais de 50% dos discentes obtiveram um desempenho de bom para excelente, em uma atividade avaliativa já usada na abordagem tradicional da disciplina. Além disso, o Aluno 17 tinha dificuldade na disciplina e conseguiu escrever um algoritmo correto e completo para a solução de um problema computacional, no caso a segunda questão.

Figura 55 – Resultados do Teste teórico

	A	B	C	D	E	F	G
89	Membros	Resultado Pontuação Teste Teórico				TOTAL	AVATAR DIA
90		1º Questão	2º Questão	3º Questão	4º Questão		
91	Aluno 01	0	0	0	0	0	AMADOR
92	Aluno 02	3,75	7,5	0	0	11,25	AMADOR
93	Aluno 03	15	30	50	0	95	AMADOR
94	Aluno 04	75	75	65	75	290	ANALISTA
95	Aluno 05	75	75	75	75	300	ANALISTA
96	Aluno 06	75	75	75	75	300	ANALISTA
97	Aluno 07	50	75	75	75	275	ANALISTA
98	Aluno 08	50	75	75	75	275	ANALISTA
99	Aluno 09	15	25	15	0	55	AMADOR
100	Aluno 10	40	75	70	0	185	AMADOR
101	Aluno 11	75	75	75	75	300	ANALISTA
102	Aluno 12	75	75	75	75	300	ANALISTA
103	Aluno 13	50	75	75	75	275	ANALISTA
104	Aluno 14	50	75	75	75	275	ANALISTA
105	Aluno 15	40	75	55	75	245	PROGRAMADOR
106	Aluno 16	0	75	0	0	75	AMADOR
107	Aluno 17	0	0	0	0	0	AMADOR
108	Aluno 18	65	75	15	0	155	AMADOR
109	Aluno 19	45	75	75	65	260	PROGRAMADOR
110	Aluno 20	7,5	75	40	25	147,5	AMADOR
111							

Fonte: Elaboração própria (2019).

Uma visualização mais detalhada sobre o resultado do teste teórico está disponível na Figura 56; nela está presente a coluna atividade concluída e nota da questão para cada um dos desafios da prática, preenchida em relação a resposta do discente. A primeira coluna, referente à tentativa realizada na resolução da questão. Já a segunda, identificava a pontuação alcançada pelo código desenvolvido. Nessa visualização podemos observar o desempenho dos alunos no teste teórico com pontuações distantes, mas a maioria da turma conseguiu pontuação acima de 65% previstos para a prática.

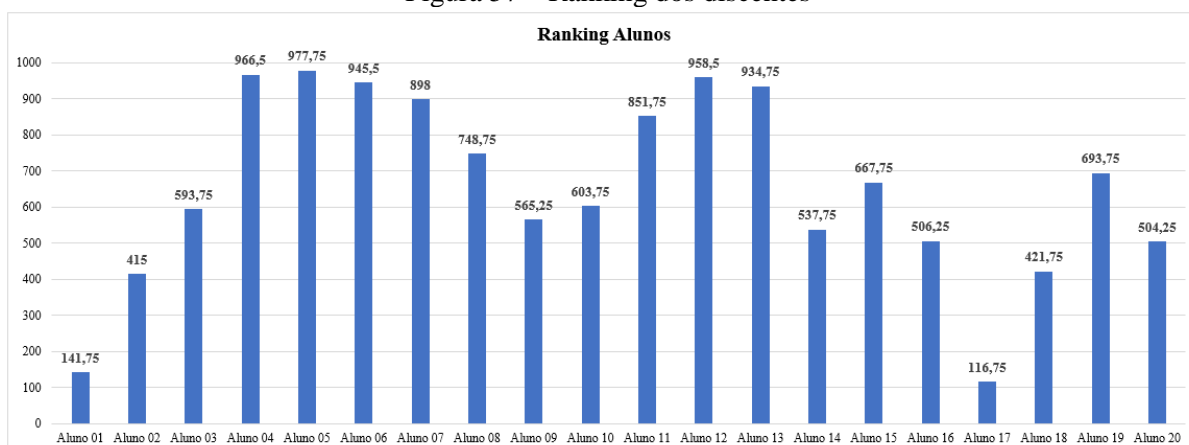
Figura 56 – Resultados das questões individuais Teste Teórico

Teste Teórico - 1°				Teste Teórico - 2°				Teste Teórico - 3°				Teste Teórico - 4°			
Membros	Atividade concluída	Nota Questão	TOTAL	Membros	Atividade concluída	Nota Questão	TOTAL	Membros	Atividade concluída	Nota Questão	TOTAL	Membros	Atividade concluída	Nota Questão	TOTAL
Aluno 01	0	0	0	Aluno 01	0	0	0	Aluno 01	0	0	0	Aluno 01	0	0	0
Aluno 02	0,5	0	3,75	Aluno 02	1	0	7,5	Aluno 02	0	0	0	Aluno 02	0	0	0
Aluno 03	1	7,5	15	Aluno 03	1	22,5	30	Aluno 03	1	42,5	50	Aluno 03	0	0	0
Aluno 04	1	67,5	75	Aluno 04	1	67,5	75	Aluno 04	1	57,5	65	Aluno 04	1	67,5	75
Aluno 05	1	67,5	75	Aluno 05	1	67,5	75	Aluno 05	1	67,5	75	Aluno 05	1	67,5	75
Aluno 06	1	67,5	75	Aluno 06	1	67,5	75	Aluno 06	1	67,5	75	Aluno 06	1	67,5	75
Aluno 07	1	42,5	50	Aluno 07	1	67,5	75	Aluno 07	1	67,5	75	Aluno 07	1	67,5	75
Aluno 08	1	42,5	50	Aluno 08	1	67,5	75	Aluno 08	1	67,5	75	Aluno 08	1	67,5	75
Aluno 09	1	7,5	15	Aluno 09	1	17,5	25	Aluno 09	1	7,5	15	Aluno 09	0	0	0
Aluno 10	1	32,5	40	Aluno 10	1	67,5	75	Aluno 10	1	62,5	70	Aluno 10	0	0	0
Aluno 11	1	67,5	75	Aluno 11	1	67,5	75	Aluno 11	1	67,5	75	Aluno 11	1	67,5	75
Aluno 12	1	67,5	75	Aluno 12	1	67,5	75	Aluno 12	1	67,5	75	Aluno 12	1	67,5	75
Aluno 13	1	42,5	50	Aluno 13	1	67,5	75	Aluno 13	1	67,5	75	Aluno 13	1	67,5	75
Aluno 14	1	42,5	50	Aluno 14	1	67,5	75	Aluno 14	1	67,5	75	Aluno 14	1	67,5	75
Aluno 15	1	32,5	40	Aluno 15	1	67,5	75	Aluno 15	1	47,5	55	Aluno 15	1	67,5	75
Aluno 16	0	0	0	Aluno 16	1	67,5	75	Aluno 16	0	0	0	Aluno 16	0	0	0
Aluno 17	0	0	0	Aluno 17	0	0	0	Aluno 17	0	0	0	Aluno 17	0	0	0
Aluno 18	1	57,5	65	Aluno 18	1	67,5	75	Aluno 18	1	7,5	15	Aluno 18	0	0	0
Aluno 19	1	37,5	45	Aluno 19	1	67,5	75	Aluno 19	1	67,5	75	Aluno 19	1	57,5	65
Aluno 20	1	0	7,5	Aluno 20	1	67,5	75	Aluno 20	1	32,5	40	Aluno 20	1	17,5	25

Fonte: Elaboração própria (2019).

Os discentes podiam ver o crescimento do seu desempenho, bem como o da turma por meio de uma representação visual, o *ranking*, apresentado na Figura 57. A representação visual permitida pelo *ranking* colabora para o elemento de jogo de competição e a característica de *feedback*, retratada na literatura e nas metodologias de coleta de dados. Nenhum dos discentes conseguiu alcançar os 1000 pontos dentro da *gamificação*. Porém, vários discentes se aproximaram, como é o caso do discente ganhador da medalha *Power* (vide seção 4.2.2.3), Aluno 05, com 977,75.

Figura 57 – Ranking dos discentes



Fonte: Elaboração própria (2019).

De maneira geral o elemento ‘pontos’ permitiu aos discentes um *feedback* positivo sobre o seu andamento dentro do conteúdo *gamificado*. Porém, indicamos algumas sugestões para a próxima versão, tais como: (i) permitir uma maior liberdade na prática do Dojo por meio da atribuição de pontos unicamente pelo código coletivo desenvolvido em sala; e (ii) definir uma regra para a pontuação por tentativa na prática do LAB.

#### 4.2.2.2 Resultados e discussão sobre comportamento

Os resultados sobre comportamento estavam dispostos em uma aba da planilha *gamificada* para os discentes. Esta tinha preenchimento automático, após o *scoreplay* preencher as abas de bonificações e penalidades. Podemos visualizar o total de bônus adquirido e o total de estrela por dia. Esse conjunto de informação estava disponível para os discentes durante cada uma das aulas do *framework*, exceto, nas aulas de teste teórico e LAB.

A Figura 58 apresenta como dados o recurso estrela atribuído no primeiro dia de aula teórica e prática. Na figura identificamos que sete discentes alcançaram duas estrelas, quatro atingiram a máxima bonificação do quesito colaboração e um na participação. A única penalidade atribuída para os discentes foi a falta, portanto, não tivemos comportamentos considerados negativos a essa *gamificação* nesse dia em sala de aula.

Figura 58 - Aquisição recurso estrela - Aula Teórica e Prática primeiro dia

Membro	Estrelas Aula 01										TOTAL BÔNUS	TOTAL ESTRELAS
	Presença	Participações	Sugestão	Pergunta	Colaboração	Falta	Atraso	Uso Cel	Uso PC	Atrapalhar aula		
Aluno 01	0	0	0	0	0	10	0	0	0	0	-10,00	0
Aluno 02	10	4	0	0	0	0	4	0	0	0	10,00	1
Aluno 03	10	6	0	0	0	0	0	0	0	0	16,00	1
Aluno 04	10	8	0	2	10	0	0	0	0	0	30,00	2
Aluno 05	10	4	0	0	8	0	0	0	0	0	22,00	2
Aluno 06	10	4	0	0	0	0	0	0	0	0	14,00	1
Aluno 07	10	8	2	4	8	0	0	0	0	0	32,00	2
Aluno 08	10	6	0	2	10	0	0	0	0	0	28,00	2
Aluno 09	10	4	0	0	10	0	0	0	0	0	24,00	2
Aluno 10	10	6	0	2	2	0	0	0	0	0	20,00	2
Aluno 11	10	4	0	0	2	0	0	0	0	0	16,00	1
Aluno 12	10	10	0	2	10	0	0	0	0	0	32,00	2
Aluno 13	10	4	0	0	2	0	0	0	0	0	16,00	1
Aluno 14	10	4	0	0	0	0	0	0	0	0	14,00	1
Aluno 15	10	4	0	0	0	0	0	0	0	0	14,00	1
Aluno 16	10	4	0	0	0	0	0	0	0	0	14,00	1
Aluno 17	0	0	0	0	0	10	0	2	0	0	-12,00	0
Aluno 18	10	4	0	0	2	0	0	0	0	0	16,00	1
Aluno 19	10	4	0	0	0	0	0	0	0	0	14,00	1
Aluno 20	10	4	0	0	0	0	4	0	0	0	10,00	1

Fonte: Elaboração própria (2019).

No segundo dia da aula teórica e prática, os dados coletados pelo juiz/*scoreplay* estão na Figura 59. Oito discentes alcançaram duas estrelas. Um dado interessante sobre esse segundo dia é que um discente alcançou oito bônus em relação ao elemento sugestão. Além disso, três discentes alcançaram a máxima bonificação do quesito colaboração, e dois em participações.

Figura 59 – Resultados recurso estrela - Aula Teórica e Prática segundo dia

Membro	Estrelas Aula 02										TOTAL BÔNUS	TOTAL ESTRELAS
	Presença	Participações	Sugestão	Pergunta	Colaboração	Falta	Atraso	Uso Cel	Uso PC	Atrapalhar aula		
Aluno 01	10	0	0	0	4	0	0	0	0	0	14,00	1
Aluno 02	10	0	0	0	6	0	0	0	0	0	16,00	1
Aluno 03	10	8	0	0	4	0	0	0	0	0	22,00	2
Aluno 04	10	2	2	0	10	0	0	0	0	0	24,00	2
Aluno 05	10	4	8	0	10	0	0	0	0	0	32,00	2
Aluno 06	10	0	2	0	4	0	0	0	0	0	16,00	1
Aluno 07	10	10	2	0	2	0	0	0	0	0	24,00	2
Aluno 08	10	10	2	0	10	0	0	0	0	0	32,00	2
Aluno 09	10	2	0	2	4	0	0	0	0	0	18,00	2
Aluno 10	10	2	0	0	0	0	0	0	0	0	12,00	1
Aluno 11	10	0	0	0	0	0	0	0	0	0	10,00	1
Aluno 12	10	6	4	6	0	0	0	0	0	0	26,00	2
Aluno 13	10	0	0	0	0	0	0	0	0	0	10,00	1
Aluno 14	10	0	0	0	4	0	0	0	0	0	14,00	1
Aluno 15	10	0	0	0	0	0	0	0	0	0	10,00	1
Aluno 16	10	2	0	4	4	0	0	0	0	0	20,00	2
Aluno 17	10	0	0	0	2	0	0	0	0	0	12,00	1
Aluno 18	10	4	0	4	4	0	0	0	0	0	22,00	2
Aluno 19	0	0	0	0	0	10	0	0	0	0	-10,00	0
Aluno 20	0	0	0	0	0	10	0	0	0	0	-10,00	0

Fonte: Elaboração própria (2019).

No terceiro dia do *framework* tivemos a primeira execução da prática Dojo *Randori*, o recurso estrela atribuído nesse dia está disponível na Figura 60. Nesse dia percebemos pouca participação dos estudantes em sala de aula. No entanto, dois discentes conseguiram a bonificação máxima por colaboração, pois compartilhavam com pequenos grupos de colegas a resolução das atividades. Outro dado importante é o de que quatro discentes conseguiram alcançar o recurso estrela e pela primeira vez as penalidades em sala de aula foram aplicadas: no caso o atraso, Aluno 01, e o uso do celular, discente 20.

Figura 60 – Resultados recurso estrela - Dojo *Randori* primeiro dia

Membro	Estrelas Aula 03										TOTAL BÔNUS	TOTAL ESTRELAS
	Presença	Participações	Sugestão	Pergunta	Colaboração	Falta	Atraso	Uso Cel	Uso PC	Atrapalhar aula		
Aluno 01	10	0	0	0	2	0	2	0	0	0	10,00	1
Aluno 02	0	0	0	0	0	10	0	0	0	0	-10,00	0
Aluno 03	10	2	2	2	2	0	0	0	0	0	18,00	2
Aluno 04	10	2	6	0	6	0	0	0	0	0	24,00	2
Aluno 05	10	6	0	0	6	0	0	0	0	0	22,00	2
Aluno 06	10	0	0	0	0	0	0	0	0	0	10,00	1
Aluno 07	10	8	0	0	2	0	0	0	0	0	20,00	2
Aluno 08	10	8	2	0	2	0	0	0	0	0	22,00	2
Aluno 09	10	4	2	0	0	0	0	0	0	0	16,00	1
Aluno 10	10	2	0	0	0	0	0	0	0	0	12,00	1
Aluno 11	10	2	0	0	0	0	0	0	0	0	12,00	1
Aluno 12	10	6	6	0	10	0	0	0	0	0	32,00	2
Aluno 13	10	0	0	0	10	0	0	0	0	0	20,00	2
Aluno 14	10	0	0	0	0	0	0	0	0	0	10,00	1
Aluno 15	10	2	0	0	0	0	0	0	0	0	12,00	1
Aluno 16	10	6	0	0	0	0	0	0	0	0	16,00	1
Aluno 17	0	0	0	0	0	10	0	0	0	0	-10,00	0
Aluno 18	10	0	0	0	0	0	0	0	0	0	10,00	1
Aluno 19	10	8	0	0	6	0	0	0	0	2	22,00	2
Aluno 20	10	0	0	0	2	0	0	2	0	0	10,00	1

Fonte: Elaboração própria (2019).

A Figura 61 apresenta as informações do recurso estrela em relação ao segundo dia de atividades do Dojo *Randori*. A participação dos discentes em sala foi satisfatória, pois identificamos bonificações por meio do elemento sugestão e pergunta. Nesse dia era visível o envolvimento dos discentes com a prática, ao ponto que, apenas dois discentes foram penalizados por conversarem sobre um assunto que não dizia respeito à aula. São eles, os discentes: 16 e 20. No mesmo dia tivemos 6 estudantes com a máxima nota em participações e 9 discentes com 2 estrelas. Por meio desses dados quantitativos corroborados com a SWOT (vide Quadro 21), afirmamos o sucesso da nova estrutura de Dojo para o ensino de Algoritmos.

Figura 61 – Resultados recurso estrela - Dojo *Randori* segundo dia

Membro	Estrelas Aula 04										TOTAL BÔNUS	TOTAL ESTRELAS
	Presença	Participações	Sugestão	Pergunta	Colaboração	Falta	Atraso	Uso Cel	Uso PC	Atrapalhar aula		
Aluno 01	10	0	2	2	2	0	0	0	0	0	16,00	1
Aluno 02	10	4	0	0	2	0	0	0	0	0	16,00	1
Aluno 03	10	6	2	2	4	0	0	0	0	0	24,00	2
Aluno 04	10	10	6	0	2	0	0	0	0	0	28,00	2
Aluno 05	10	10	2	2	4	0	0	0	0	0	28,00	2
Aluno 06	10	2	0	2	4	0	0	0	0	0	18,00	2
Aluno 07	10	10	4	0	4	0	0	0	0	0	28,00	2
Aluno 08	10	10	4	2	2	0	0	0	0	0	28,00	2
Aluno 09	10	0	0	0	2	0	0	0	0	0	12,00	1
Aluno 10	10	4	0	0	2	0	0	0	0	0	16,00	1
Aluno 11	10	0	0	0	2	0	0	0	0	0	12,00	1
Aluno 12	10	10	0	0	2	0	0	0	0	0	22,00	2
Aluno 13	10	2	0	0	2	0	0	0	0	0	14,00	1
Aluno 14	10	2	0	0	2	0	0	2	0	0	12,00	1
Aluno 15	10	0	0	0	2	0	0	0	0	0	12,00	1
Aluno 16	10	8	2	4	2	0	0	0	0	2	24,00	2
Aluno 17	10	0	0	0	2	0	0	0	0	0	12,00	1
Aluno 18	0	0	0	0	0	10	0	0	0	0	-10,00	0
Aluno 19	10	10	2	0	2	0	0	0	0	2	22,00	2
Aluno 20	10	2	0	2	2	0	0	0	0	0	16,00	1

Fonte: Elaboração própria (2019).

As bonificações da aula de *feedback* estão presentes na Figura 62. Sete discentes conseguiram 2 estrelas, três deles, tendo 10 participações. Dois discentes sofreram a



penalização por estarem utilizando o computador em outras atividades, pois nesse dia não se previa o uso do computador, porém não foi obrigado que eles fossem desligados.

Figura 62 – Resultado recurso estrela aula de *feedback*

Membro	Estrelas Aula 07										TOTAL BÔNUS	TOTAL ESTRELAS
	Presença	Participações	Sugestão	Pergunta	Colaboração	Falta	Atraso	Uso Cel	Uso PC	Atrapalhar aula		
Aluno 01	10	6	0	0	0	0	0	0	0	0	16,00	1
Aluno 02	0	0	0	0	0	10	0	0	0	0	-10,00	0
Aluno 03	10	10	8	0	0	0	0	0	0	0	28,00	2
Aluno 04	10	0	0	0	0	0	0	0	2	0	8,00	1
Aluno 05	10	6	0	0	0	0	0	0	0	0	16,00	1
Aluno 06	10	10	4	0	0	0	0	0	0	0	24,00	2
Aluno 07	10	0	0	0	0	0	0	0	0	0	10,00	1
Aluno 08	10	0	0	0	0	0	0	0	0	0	10,00	1
Aluno 09	10	10	4	0	0	0	0	0	0	0	24,00	2
Aluno 10	10	8	4	0	0	0	0	0	0	0	22,00	2
Aluno 11	10	0	0	0	0	0	0	0	0	0	10,00	1
Aluno 12	10	10	0	0	0	0	0	0	0	0	20,00	2
Aluno 13	10	0	6	2	0	0	0	0	0	0	18,00	2
Aluno 14	10	0	0	0	0	0	0	0	0	0	10,00	1
Aluno 15	10	0	0	0	0	0	0	0	0	0	10,00	1
Aluno 16	10	10	6	6	0	0	0	0	0	0	32,00	2
Aluno 17	0	0	0	0	0	10	0	0	0	0	-10,00	0
Aluno 18	10	6	0	0	0	0	0	0	2	0	14,00	1
Aluno 19	10	8	0	2	0	0	0	0	0	0	20,00	2
Aluno 20	0	0	0	0	0	10	0	0	0	0	-10,00	0

Fonte: Elaboração própria (2019).

A Figura 63 apresenta a aba relativa ao recurso estrela que os discentes iam recebendo no decorrer da *gamificação* preenchida automaticamente. Ela foi pensada como *feedback* mais imediato para o discente em relação ao seu ganho do recurso estrela em todo o processo.

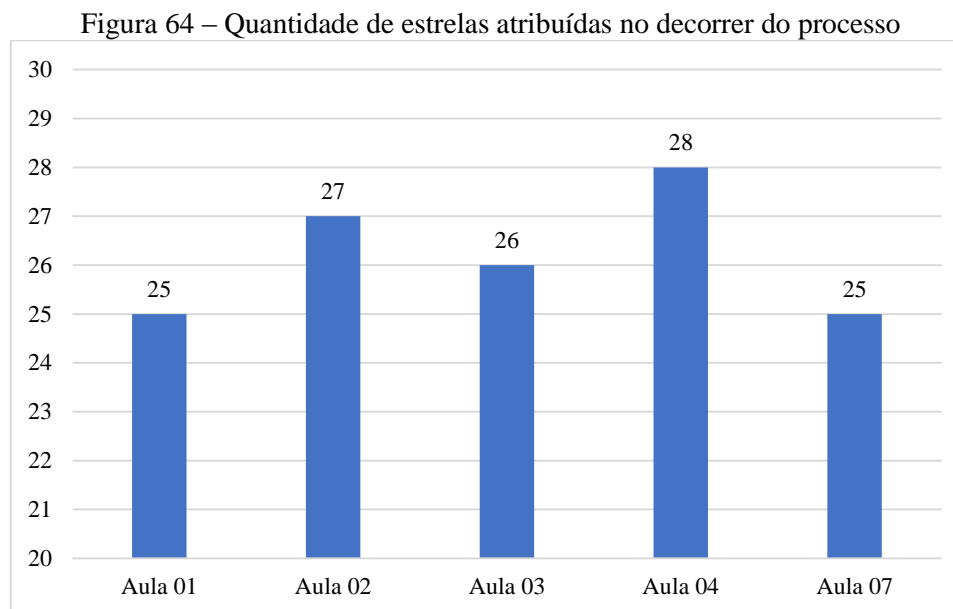
Figura 63 – Quantitativo total de estrela por discente

Membro	Quantidade de estrelas finais							TOTAL
	Aula 1	Aula 2	Aula 3	Aula 4	Aula 5	Aula 6	Aula 7	
Aluno 01	0	1	1	1	0	0	1	4
Aluno 02	1	1	0	1	0	0	0	3
Aluno 03	1	2	2	2	0	0	2	9
Aluno 04	2	2	2	2	0	0	1	9
Aluno 05	2	2	2	2	0	0	1	9
Aluno 06	1	1	1	2	0	0	2	7
Aluno 07	2	2	2	2	0	0	1	9
Aluno 08	2	2	2	2	0	0	1	9
Aluno 09	2	2	1	1	0	0	2	8
Aluno 10	2	1	1	1	0	0	2	7
Aluno 11	1	1	1	1	0	0	1	5
Aluno 12	2	2	2	2	0	0	2	10
Aluno 13	1	1	2	1	0	0	2	7
Aluno 14	1	1	1	1	0	0	1	5
Aluno 15	1	1	1	1	0	0	1	5
Aluno 16	1	2	1	2	0	0	2	8
Aluno 17	0	1	0	1	0	0	0	2
Aluno 18	1	2	1	0	0	0	1	5
Aluno 19	1	0	2	2	0	0	2	7
Aluno 20	1	0	1	1	0	0	0	3

Fonte: Elaboração própria (2019).

Na Figura 64 temos um dado para a discussão sobre a progressão do recurso estrela em todo o processo da *gamificação*. A aula 4, Dojo *Randori* segundo dia, atingiu o maior número de alcance do recurso estrela. Na prática de *feedback*, aula 07, a qual imaginamos ser o dia em que mais se distribuiria estrelas, ficou empatada com a primeira aula teórica e prática, a aula

01. Então, é importante uma discussão futura de novas formas para coletar informações dos discentes sobre o *framework*, durante a aula de *feedback*.



Fonte: Elaboração própria (2019).

De uma maneira geral, sobre o elemento recurso estrelas atribuídas, nenhum dos discentes conseguiu as 3 estrelas como meta diária. Podemos identificar novas formas de bonificação, ou então retirar as limitações de bonificação com o intuito de permitir aos discentes que consigam alcançar o objetivo diário para o componente nos próximos usos do *framework gamificado*.

#### 4.2.2.3 Medalhas da *gamificação*

Outro elemento da *gamificação* utilizado com o intuito de analisar o desempenho dos discentes em relação ao conteúdo e a participação foi a medalha. Para o experimento relatado adotou-se as categorias: *Power* e Participativo. A primeira refere-se à quantidade de pontos que o discente tinha acumulado no decorrer da *gamificação*, ou seja, o desempenho (vide Figura 57). A *Power* subdividia-se em: ouro, prata e bronze. Na *gamificação* o discente 05 conseguiu a medalha *Power*, a de ouro, por sua vez, foi atribuída ao discente 05, porque adquiriu a maior pontuação.

Figura 65 – Medalha *Power* atribuída

Medalhas - Power			
Posição	Valor	Nome	Tipo
1º Lugar	977,75	Aluno 05	Ouro
2º Lugar	966,5	Aluno 04	Prata
3º Lugar	958,5	Aluno 12	Bronze

Fonte: Elaboração própria (2019).

A segunda medalha indicava o discente com maior quantidade de estrelas, ou seja, o que mais participou no processo (vide Figura 64). A segunda medalha tinha três subdivisões, similar a *Power*. O Discente 12 ganhou a categoria de ouro, pois foi ele quem mais participou do processo. No segundo e terceiro lugar, tivemos 7 discentes com 9 estrelas, dentre eles, o discente que ganhou a categoria ouro na medalha *Power*: o discente 05.

Figura 66 – Medalha Participativo

Medalhas - Participativo			
Posição	Valor	Nome	Tipo
1º Lugar	10	Aluno 12	Ouro
2º Lugar	9	EMPATE	Prata
3º Lugar	9	EMPATE	Bronze

Fonte: Elaboração própria (2019).

O elemento medalha foi adicionado ao *framework* com o intuito de estimular a competição entre os discentes. Porém, na aula de *feedback*, eles informaram a colaboração como elemento central. O ponto de melhoria seria a disponibilização de prêmios para os três primeiros lugares nas medalhas, com o intuito de estímulo para os discentes.

### 4.2.3 Casos particulares

Esta seção de avaliação do experimento descreve os casos especiais, ou seja, pessoas que no decorrer da *gamificação* não se adaptaram a nova abordagem, ou então, tiveram um ganho de desempenho expressivo dentro do processo de ensino e aprendizagem de Algoritmos. Os dados descritos nessa seção são visualizados por meio da análise dos dados qualitativos e quantitativos, coletados durante o experimento e também por meio da observação participante, realizada antes do uso do *framework* em sala de aula.

## Aluno 14

Para o Aluno 14, a *gamificação* pode não ser considerada a melhor opção para o desenvolvimento do processo de ensino e aprendizagem. Essa realidade pode ser constatada, pois o discente em questão faltou em algumas atividades, as quais são consideradas importantes para a atividade proposta, tais como a prática do LAB, em que o discente poderia conseguir até 200 pontos. Vamos observar na seção da validação, que o discente, no seu desempenho tradicional, tem desempenho superior ao *gamificado*.

Outro ponto que é importante frisar foi o não envio da lista de exercício para casa, situação essa a qual demonstra a falta de estímulo para participar nas atividades e que colaborou para o baixo rendimento na *gamificação*.

No dia da aula de *feedback* o discente não comentou várias das perguntas abertas. Porém, depois falou diretamente e particularmente ao pesquisador responsável pela *gamificação*, quando questionado sobre suas faltas no processo; então informou a limitação com o elemento colaboração, e principalmente o fato de não gostar de falar e se apresentar em público. Vale ressaltar que no primeiro dia de aula o discente foi instigado várias vezes para poder ser ajudado por outra pessoa com o intuito de fluir a colaboração e ele rejeitou. Observamos também que o discente tinha certo medo de apresentações em público, tanto é que não conseguiu se concentrar na resolução desenvolvida no dia que estava atuando como piloto dentro da prática do Dojo, além de ter faltado no dia da prática LAB.

## Alunos 01, 02 e 17

No caso dos Alunos 01, 02 e 17, podemos perceber que a *gamificação* proporcionou pontuação para eles, por meio das tentativas e colaboração em sala de aula. Porém, em relação ao aprendizado, não identificamos mudanças significativas. Ao observamos o desempenho individual desses alunos em relação ao conteúdo, em especial à prática do teste teórico, identificamos os seguintes pontos: (i) o Aluno 01 zerou a atividade; (ii) o Aluno 02 conseguiu a pontuação por tentativa, ou seja, conseguiu fazer a construção básica do algoritmo; e (iii) o Aluno 17 faltou a essa prática. Vale ressaltar que os três discentes são “repetentes” na disciplina.

Porém, o que cabe uma análise mais aprofundada é sobre o Aluno 02, pois tivemos a percepção de que à medida que o discente visualizava suas pontuações, ele se fazia presente em sala. Como prova para essa situação, a maioria das pontuações adquiridas são na sala de aula. Na primeira aula teórica e prática o pesquisador responsável pela *gamificação*, bem como os

monitores, perceberam no discente ao qual estava colaborando com a Aluna 01, no caso o Aluno 04, que ele teve todo o cuidado de explicar a regra de funcionamento dos algoritmos. Consideramos essa ação como algo de extrema relevância para o processo de ensino e aprendizagem, e forma de aprendizagem colaborativa.

Então, tivemos a impressão de que se o conteúdo *gamificado* fosse em um momento inicial da disciplina, seria possível que o Aluno 02 estivesse mais engajado para a sua construção de conhecimento na disciplina Algoritmos.

### **Aluno 18**

No caso do Aluno 18, tivemos algo peculiar, um salto impressionante por meio da *gamificação*, pois averiguamos a aplicação do discente no seu desenvolvimento durante as práticas do experimento. Vale ressaltar que o discente tinha baixa visão. E, quando olhamos para o desenvolvimento dele na parte tradicional da disciplina, vimos que ele zerou nas duas provas, destacando-se a falta em uma.

Durante as aulas teóricas e práticas identificamos a aprendizagem, por pares por meio do item ‘colaboração’, acontecendo com o discente mencionado. No decorrer das resoluções ele conseguiu resolver três das quatro questões propostas para sala de aula. Para a lista de exercício feita para casa, na aula de Dojo *Randori* e LAB, por mais que ele não tenha enviado ele estava atento. Outro ponto importante a se mencionar é no teste teórico do conteúdo *gamificado* que podemos registrar pontuação pela tentativa e pelo código desenvolvido. Quando comparado com as outras provas da disciplina, o discente não tinha conseguido. Logo, é importante acrescentar que o aluno de baixa visão é amparado pela legislação da UFPA, uma vez que tem direito de um tempo a mais para o desenvolvimento do teste teórico.

### **Alunos 09 e 12**

No caso dos Alunos 09 e 12, tivemos um ganho expressivo no processo de ensino e aprendizagem desses discentes. Podemos observar que os dois cumpriram as listas de exercício em sala de aula, colaboraram com os colegas, e no caso do Aluno 09, em algumas questões, teve outros colegas colaborando com ele. As diferenças entre eles são nas notas obtidas durante as práticas do LAB e Teste teórico em que podemos observar melhor ao analisar os dados quantitativos (vide seção 4.2.2); e no ganho de estrela, em que o Aluno 12 ganhou em cada uma

das aulas do processo, duas estrelas, ou seja, o máximo atingido pelos discentes, enquanto o Aluno 09, apenas na aula de Dojo, ganhou apenas uma, porém nas outras ganhou duas.

Em relação ao desempenho, no processo tradicional, este não foi tão satisfatório para os dois discentes, constatados por meio das suas notas (vide seção 4.3). Enquanto, na *gamificação* tiveram um ganho de performance, pois ao perceberem as nuances da abordagem, eles começaram a competir e colaborar não só para ganhar bônus ou pontos, como observamos na visualização das bonificações (vide seção 4.2.1.2), mas estavam ativos e engajados no processo.

#### 4.2.4 Melhorias

Dentro da execução do experimento foi possível verificar melhorias as quais serão adotadas em uma segunda execução do *framework*, tais como:

Na aula teórica e prática, trabalhar com o fluxo programar e testar. Além disso, já desenvolver com os discentes algumas questões de escopo aberto para se familiarizarem com a prática do Dojo.

Na construção das listas de desafios as questões devem ter níveis diferenciais de dificuldade, seguindo da mais fácil para a mais difícil. Vale ressaltar que essas listas são as questões utilizadas em sala de aula para os discentes resolverem em casa, no Laboratório de Programação e no teste teórico.

No caso lista de exercício para casa, é importante ter um tempo, em torno de dez minutos, na aula teórica e prática para explicar aos discentes sobre como resolver. Outra melhoria seria disponibilizar dicas de resolução das questões.

Para o Dojo *Randori* as sugestões foram: (i) tornar opcional o papel do copiloto, e bonificar o discente que aceitar o desafio; (ii) a atividade ser pontuada unicamente pelo código desenvolvido pelos discentes, sendo que a forma como executam os papéis e codificam será bonificada; (iii) permitir com que os discentes envie sugestões ao código desenvolvido pela turma e bonificar por sugestão, porém com um limite de tempo, como o de dois dias para submeter, por exemplo; e (iv) possibilitar aos discentes dez minutos iniciais para compreender o desafio proposto na atividade.

Na proposta LAB fazer duas execuções: uma sem pontuação, para os discentes ganharem prática, e outra com pontuação.

Para os desafios da prova teórica e LAB, fixar uma regra de atribuição da pontuação pela tentativa de resolver a questão. No caso do primeiro experimento foi adotado como norma:

(i) o discente conseguir fazer leitura e escrita por meio do programa e acrescentar cinco por cento; e (ii) utilizar a estrutura laço corretamente cinco por cento.

Vale ressaltar como melhoria para esse experimento, a adaptação do material para um discente com baixa visão, ao qual nos relatou que letras impressas no tamanho 14 ele conseguiria ler. Então, todo o material impresso entregue era formatado nesse tamanho para ele, além de que os slides de apresentação das aulas eram disponibilizados via plataforma Moodle (Moodle, 2018) para que o discente pudesse visualizar com ferramentas de acessibilidade, disponibilizadas pelo sistema operacional Windows, como por exemplo a Lupa.

### 4.3 VALIDAÇÃO DO FRAMEWORK GAMIFICADO

A validação do *framework* aconteceu de duas formas aqui descritas: (i) pela evolução e análise das notas dos discentes dentro de sala de aula, ou seja, no contexto da mesma disciplina e com os mesmos participantes; e (ii) pela comparação do desempenho das notas dos discentes na disciplina *gamificada* com as notas obtidas na disciplina sem *gamificação*.

Na Figura 67 temos presente as notas da disciplina Algoritmos em relação ao conteúdo *gamificado*. Na imagem podemos ver quantos pontos os discentes acumularam no decorrer da abordagem proposta, quantas estrelas e a conversão destes elementos para as notas, apresentadas em uma aba da planilha *gamificada online*. Em todo o tempo de jogo o discente estava acompanhando os pontos adquiridos no seu processo de aprendizagem e a conversão para a nota na disciplina.

Figura 67 – Notas dos discentes no uso do *framework gamificado*

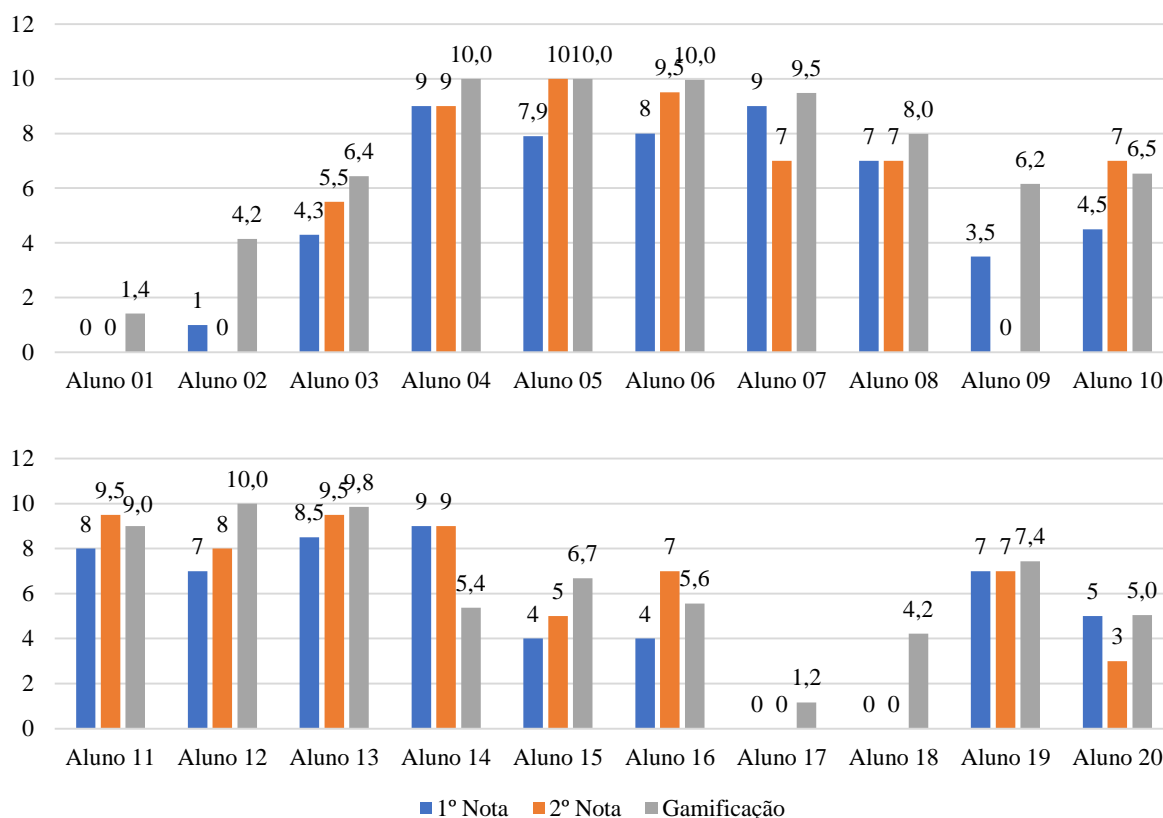
Nota Final	Pontuações	Total Pontos	Total Nota	Total Estrelas	Conversão Nota	Nota Final Conteúdo
Aluno 01		141,75	1,42	4	0,0	1,42
Aluno 02		415	4,15	3	0,0	4,15
Aluno 03		593,75	5,94	9	0,5	6,44
Aluno 04		966,5	9,67	9	0,5	10,17
Aluno 05		977,75	9,78	9	0,5	10,28
Aluno 06		945,5	9,46	7	0,5	9,96
Aluno 07		898	8,98	9	0,5	9,48
Aluno 08		748,75	7,49	9	0,5	7,99
Aluno 09		565,25	5,65	8	0,5	6,15
Aluno 10		603,75	6,04	7	0,5	6,54
Aluno 11		851,75	8,52	5	0,5	9,02
Aluno 12		958,5	9,59	10	1,0	10,59
Aluno 13		934,75	9,35	7	0,5	9,85
Aluno 14		537,75	5,38	5	0,5	5,88
Aluno 15		667,75	6,68	5	0,5	7,18
Aluno 16		506,25	5,06	8	0,5	5,56
Aluno 17		116,75	1,17	2	0,0	1,17
Aluno 18		421,75	4,22	5	0,5	4,72
Aluno 19		693,75	6,94	7	0,5	7,44
Aluno 20		504,25	5,04	3	0,0	5,04

Fonte: Elaboração própria (2019).

A primeira forma adotada para validar o uso do *framework* para apoio ao processo de ensino e aprendizagem de algoritmos vem de uma análise da progressão dos discentes no decorrer da disciplina. Para isso, coletamos as notas advindas da primeira e segunda prova, por meio da observação participante; além disso, aglutinamos em uma imagem com o intuito de realizar a análise (vide Figura 68).

A imagem retrata que vários discentes tiveram ganho de desempenho considerável, como é o caso dos discentes: 03, 04, 07, 08, 09, 12, 15, 18 e 20. Alguns outros mantiveram o seu desempenho e relataram por meio da aula de *feedback*, a excelente experiência de trabalhar com a *gamificação*. Segundo um dos discentes, “Ter o contato com a colaboração em sala de aula, remete a competências do mercado de trabalho”. Assim, enfatizamos no primeiro dia de aula os elementos de jogos aos quais trabalharíamos em sala. Então, a experiência de trabalhar *gamificação* foi um sucesso ao olharmos para a progressão de desempenho dos discentes na disciplina e também para as informações relatadas na aula de *feedback*.

Figura 68 – Apresentação do desempenho do discente na disciplina inteira



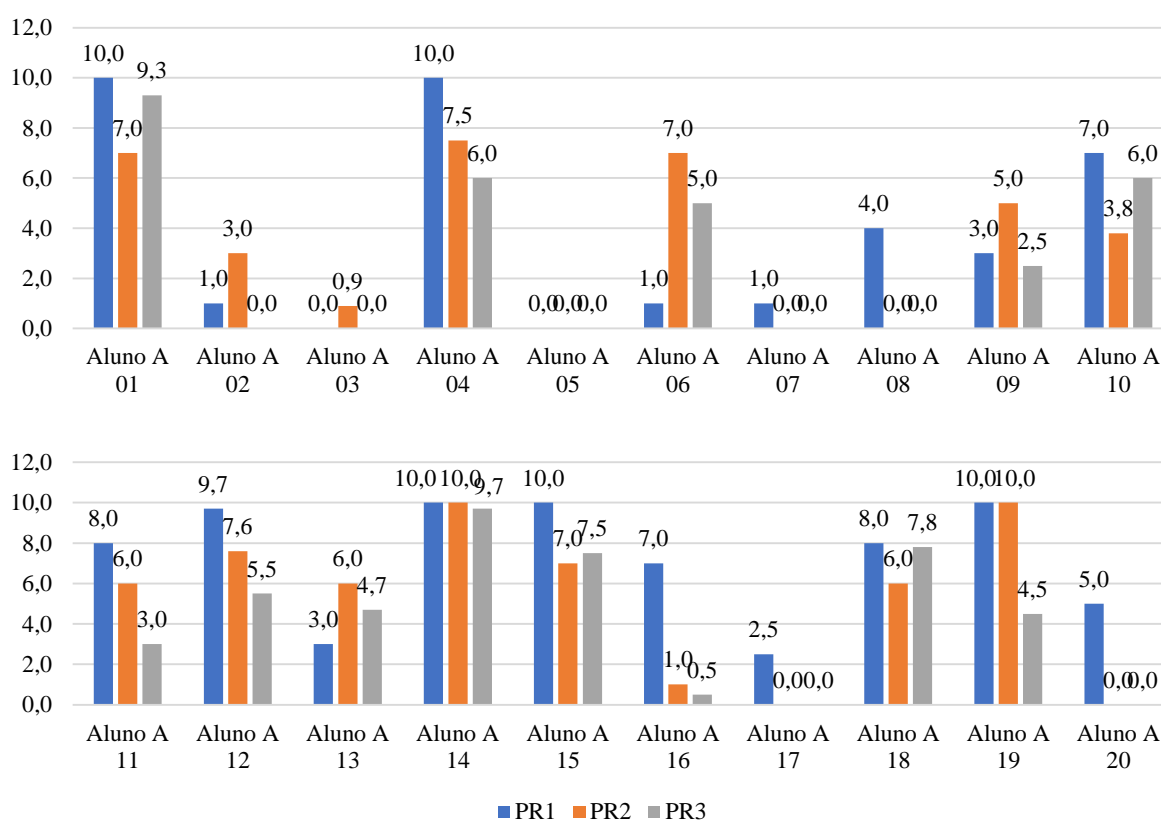
Fonte: Elaboração própria (2019).

A Figura 69 representa as notas coletadas da disciplina Algoritmo oferta pela Faculdade de Computação no ano de 2016. Os discentes na terceira avaliação, a qual equivale com o



conteúdo de estrutura de dados homogêneo, obtiveram um desempenho inferior ao das outras notas. Temos como exemplo os seguintes discentes: 02, 04, 06, 08, 09, 10, 11,12, 13, 15, 16, 17, 19 e 20. Nesse ponto, quando analisamos a relação aos dados coletados por meio do experimento com os da observação participante, observamos que a *gamificação* auxiliou os discentes no seu desempenho em relação à disciplina. Portanto, no que tange a comparação e validação sobre a abordagem *gamificada* em comparação com o qual é adotado na disciplina, indicamos sucesso de nossa proposição.

Figura 69 – Notas disciplina Algoritmos 2016



Fonte: Elaboração própria (2019).

Logo, podemos inferir que o *framework gamificado* para o processo de ensino e aprendizagem da disciplina Algoritmos é caracterizado por sucesso, pois permite aos discentes ganhos de desempenho, além de engajá-los no conhecimento dos conteúdos da disciplina, sendo válido para uso em outros conteúdos, bem como na disciplina inteira. Vale destacar a aula de *feedback* e o uso do SWOT adaptado, em que podemos coletar pontos de melhoria para uma próxima interação do produto desenvolvido.

## 5 CONCEPÇÃO DA SEGUNDA VERSÃO DO FRAMEWORK GAMIFICADO

A segunda versão do produto da dissertação de mestrado foi desenvolvida a partir das melhorias identificadas na avaliação e validação ocorrida com o uso da primeira versão, bem como com os requisitos levantados na Revisão Sistemática da Literatura. Nas próximas subseções serão apresentados: o procedimento metodológico da Revisão sistemática; os requisitos; bem como a segunda versão do *framework gamificado*.

### 5.1 REVISÃO SISTEMÁTICA DA LITERATURA

Nesta seção relatamos o desenvolvimento e os resultados provenientes dos procedimentos metodológicos da Revisão Sistemática da Literatura. Será discutida a metodologia, a estratégia, a sumarização dos resultados e as ameaças da Revisão Sistemática, bem como as conclusões para a elaboração da segunda versão do *framework gamificado*.

#### 5.1.1 Metodologia da **Revisão Sistemática**

A Revisão Sistemática da Literatura (RSL) é uma metodologia de pesquisa que objetiva identificar, avaliar e interpretar a maior quantidade possível de estudos relevantes e disponíveis para uma particular questão de pesquisa, tópico de pesquisa ou fenômeno de interesse (KITCHENHAM; CHARTERS, 2007). O principal objetivo consiste em realizar uma pesquisa exaustiva na literatura em busca de evidências que possam apoiar uma determinada hipótese, ou simplesmente a busca por conhecimento aprofundado acerca de certo fenômeno de interesse.

Portanto, a metodologia da RSL é considerada um estudo secundário, por utilizar como base estudos primários previamente publicados. Deste modo, pode ser feita a integração de diversos estudos experimentais, de forma a comparar seus resultados, visto que nenhum estudo individualmente pode ser considerado definitivo (TRAVASSOS; BIOLCHINI, 2007), sendo, assim, necessária a confirmação de resultados obtidos a partir da análise de um número maior de estudos.

A metodologia em questão teve início na Medicina, em que foi possível constatar que uma pesquisa criteriosa e exaustiva em busca de melhores práticas e cuidados com os pacientes, baseando-se na comparação entre métodos já existentes, presentes em estudos publicados, resultava em uma melhoria significativa na tomada de decisão por parte dos profissionais de saúde no momento de escolher o melhor tratamento para seus pacientes. Conforme Mafra *et al.*

(2006), o trabalho de Kitchenham *et al.* (2004) foi o primeiro a estabelecer um paralelo entre a Medicina e a Engenharia de Software.

A RSL conduzida nessa pesquisa tem como base a Engenharia de Software Baseada em Evidências (*Evidence Based Software Engineering – EBSE*), a qual busca prover meios pelos quais melhores evidências provenientes da pesquisa possam ser integradas com experiência prática e valores humanos no processo de tomada de decisão, considerando o desenvolvimento e a manutenção do software (KITCHENHAM *et al.*, 2004). A essência do paradigma baseado em evidência é coletar e analisar sistematicamente todos os dados disponíveis sobre determinado fenômeno para obter uma perspectiva mais completa e mais ampla do que se pode captar a partir de um estudo individual (COSTA, 2010).

O guia de Kitchenham e Charters (2007) aponta algumas razões para se realizar uma RSL, quais sejam: (i) sumarizar evidências existentes sobre um fenômeno, de modo a aumentar a precisão e a confiabilidade dos resultados; (ii) identificar lacunas na pesquisa atual; (iii) identificar na literatura o quanto evidências empíricas suportam ou contradizem hipóteses teóricas; (iv) fornecer um arcabouço para posicionar novas pesquisas; e (v) apoiar a geração de novas hipóteses.

Por ser uma metodologia de pesquisa organizada e metódica, a RSL possui, segundo Mafra *et al.* (2006), as seguintes características:

- Abrangência: por englobar a grande maioria dos estudos relevantes à questão de pesquisa.
- Não-tendenciosa: possui um protocolo de revisão, não sendo dirigida por interesses pessoais de seus pesquisadores. O protocolo estabelecido permite diminuir o viés da pesquisa.
- É suscetível a variações nos resultados obtidos de acordo com a análise da literatura.
- É replicável, por possuir um protocolo de revisão definido na fase de planejamento.

O processo desenvolvido na realização de uma RSL possui três fases distintas, propostas por Kitchenham e Charters (2007): planejamento, condução e apresentação dos resultados. As tarefas de cada uma destas etapas estão apresentadas de forma breve, a seguir:

- Planejamento da revisão: identifica a necessidade da revisão; define a comissão de pesquisa (não realizada nessa pesquisa); especifica as questões de pesquisa; desenvolve o protocolo de revisão; avalia o protocolo de revisão.
- Condução da revisão: identifica a pesquisa; seleciona os estudos primários; avalia a qualidade dos estudos; extrai e monitora os dados; sintetiza os dados.
- Apresentação dos resultados: define métodos pelos quais a pesquisa será divulgada; formula o relatório da revisão sistemática, contendo todos os dados obtidos por meio da pesquisa, além da análise feita pelo próprio pesquisador; avalia o relatório da revisão sistemática.

### 5.1.2 Estratégia da Revisão Sistemática

O protocolo da RSL (Apêndice F) é o documento que descreve as estratégias adotadas e planejadas para a condução desta Revisão Sistemática. No protocolo encontram-se registrados os objetivos, o escopo, as restrições, os critérios de seleção e avaliação, entre outras estratégias definidas para a condução com sucesso da Revisão Sistemática. As subseções a seguir relatam essas estratégias.

#### 5.1.2.1 Objetivo da Revisão Sistemática

O principal objetivo desta Revisão Sistemática é identificar as abordagens de apoio ao processo de ensino-aprendizagem de algoritmos que façam uso da *gamificação*, no contexto do ensino superior, para os cursos de graduação da área de TI. Por abordagens entende-se: metodologias, métodos, técnicas e ferramentais. Desta maneira, têm-se a seguinte estrutura, proposta por Santos (2010):

- **Analisar:** relatos de experiência e publicações científicas por meio de um estudo baseado em Revisão Sistemática.
- **Com o propósito de:** identificar abordagens *gamificadas* para apoiar o processo de ensino-aprendizagem de algoritmos.
- **Com relação à:** definição e uso de metodologias, métodos, técnicas, ferramentais e demais instrumentos empregados como apoio ao processo de ensino-aprendizagem de algoritmos em cursos de graduação na área de tecnologia da informação (TI).
- **Do ponto de vista de:** dos pesquisadores.

- **No contexto:** acadêmico, instituições de ensino superior públicas ou privadas.

### 5.1.2.2 Questão de pesquisa principal

O objetivo principal desse estudo é identificar e avaliar os principais relatos de experiência com abordagens *gamificadas* para o ensino-aprendizagem de Algoritmos no contexto dos cursos superiores. Assim, uma questão de pesquisa é o primeiro passo para o desenvolvimento de uma RSL, devendo ser específica e de relevância científica. Com a intenção de atingir esse objetivo, foi definida a seguinte questão principal:

QP01: Quais as principais abordagens de apoio ao processo de ensino e aprendizagem *gamificado* da disciplina Algoritmos no contexto de cursos de nível superior de tecnologia da informação?

#### 5.1.2.2.1 Estrutura da questão principal

Durante a construção da estrutura da questão principal foi utilizada a recomendação de Kitchenham (2007), que a organiza como: *Population, Intervention, Context, Outcomes, Comparison* (PICOC). Para essa pesquisa, foram empregados apenas os itens população, intervenção e resultados, o que a torna uma revisão QUASI sistemática da literatura, de acordo com Santos (2010).

Têm-se por objetivo, por meio dessa questão, identificar as abordagens de ensino empregadas na disciplina algoritmos e afins, em especial as que utilizam técnicas de *gamificação* (Intervenção), aplicáveis no ensino superior (Contexto), utilizadas em sala de aula por professores (população). Assim, tem-se a seguinte estrutura da questão principal:

- **População (P):** trabalhos publicados em conferências e periódicos indexados pelo sistema Qualis-CAPES, com foco em cursos de tecnologia da informação.
- **Intervenção (I):** abordagens para apoiar o processo de ensino-aprendizagem de algoritmos, em especial as que empregam técnicas de *gamificação*.
- **Resultados (O):** metodologias, métodos, técnicas, ferramentais e demais instrumentos empregados como apoio ao processo de ensino-aprendizagem de algoritmos.

### 5.1.2.3 Questões secundárias

Com base na questão principal, foi estabelecido um conjunto de questões secundárias visando esclarecer detalhes importantes da revisão, as quais são respondidas durante a fase de extração de informações, a saber:

- QSO1. Qual a abordagem mais utilizada para *gamificar* o processo de ensino-aprendizagem da disciplina Algoritmos ou equivalente?
- QSO2. Quais as principais dificuldades encontradas ao se *gamificar* o ensino da disciplina Algoritmos ou equivalente?
- QSO3. Quais as principais dificuldades encontradas no aprendizado *gamificado* da disciplina Algoritmos ou equivalente?
- QSO4. Como se realiza o processo *gamificado* de ensino-aprendizado da disciplina Algoritmos ou equivalente?
- QSO5. Existem relatos de sucesso do processo *gamificado* de ensino-aprendizagem da disciplina Algoritmos ou equivalente?
- QSO6. Existem relatos de insucesso do processo *gamificado* de ensino-aprendizagem da disciplina Algoritmos ou equivalente?
- QSO7. Existem relatos de avaliação do processo *gamificado* de ensino-aprendizagem da disciplina Algoritmos ou equivalente?
- QSO8. Existem relatos de validação do processo *gamificado* de ensino-aprendizagem da disciplina Algoritmos ou equivalente?
- QSO9. Como foi feita a validação do processo *gamificado* de ensino-aprendizagem da disciplina Algoritmos ou equivalente?

### 5.1.2.4 Escopo e recursos da pesquisa

Com a finalidade de tornar a pesquisa exequível, foi definido o seguinte escopo da pesquisa:

- I. Para a seleção das fontes de busca, foram definidos os seguintes critérios, adaptados de Souza *et al.* (2017):
  - Disponibilidade para consultas web.
  - Disponibilidade para busca de artigos a partir do domínio da UFPA.

- Disponibilidade de artigos na íntegra por meio do domínio da UFPA ou a partir da utilização da *engine* de busca Google e/ou Google Scholar.
  - Disponibilidade de artigos em inglês ou português.
  - Presença de mecanismo de busca que faça uso de palavras-chave.
  - Relevância da fonte.
  - Boa relação entre estudos retornados e estudos selecionáveis.
- II. Devido ao caráter da pesquisa e os critérios de seleção, também foram adotados os critérios de restrições, adaptados de Souza *et al.* (2017), a saber:
- A pesquisa não pode incorrer em ônus financeiro aos pesquisadores. Portanto, serão selecionadas apenas as fontes que possibilitam consultas de forma gratuita pelo domínio da UFPA.
  - Os estudos foram obtidos por meio das fontes selecionadas e em conformidade com os critérios de inclusão e exclusão.
  - Foram apenas considerados estudos nas línguas: inglês e português. A primeira por ser a língua de maior difusão no mundo, considerando os estudos de maior relevância e abrangência mundial. Já a segunda, por ser a língua oficial onde o estudo está sendo realizado (Brasil), para entender como está o panorama nacional em relação a pesquisas desse cunho.
  - A pesquisa restringiu-se aos resultados publicados entre 01 de janeiro de 2006 até dezembro de 2016, contemplando, desta forma, um período de 10 anos. Isso se deve ao fato do termo *gamificação* ter sido definido em 2008 e referendado na literatura no estudo de Deterding *et al.* 2011. Porém, para a maior completude do estudo retroagiu-se mais dois anos para completar o ciclo já informado.
- III. Os seguintes recursos foram utilizados no decorrer da execução desta Revisão Sistemática da Literatura:
- Dois pesquisadores (um aluno de mestrado do PPGCIMES e um graduando do curso de Bacharelado em Sistemas de Informação da UFPA).
  - Acesso às fontes de pesquisa por meio do domínio da Universidade Federal do Pará;
  - Validações sobre documentos e procedimentos da realização da Revisão Sistemática por meio de reuniões com a Profa. Dr. Marianne Kogut Eliasquevici (Docente do

PPGCIMES) e Prof. Dr. Sandro Ronaldo Bezerra Oliveira (Docente Programa de Pós-Graduação em Ciência da Computação).

#### 5.1.2.5 Busca primária

A busca primária constitui a etapa em que os pesquisadores adotam métodos previamente definidos para encontrar os estudos primários nas fontes de busca selecionadas. As buscas podem ser feitas de duas formas: Manuais e Automáticas.

Para este estudo foi definido unicamente o método das buscas automáticas, devido ter sido uma busca abrangente e com o objetivo de eliminar o viés da seleção. Por busca automática entende-se como as definidas para fontes que possuem uma *engine* de busca, ou seja, as que possuem trabalhos indexados em repositórios de estudos *online*.

A busca automática foi realizada a partir das máquinas de busca (buscadores automáticos, como exemplo Google e Google Scholar), disponíveis nas bases de dados selecionadas. A partir da inserção das *strings* de buscas específicas, nessas máquinas são retornados os estudos, os quais possuem presentes as palavras-chave no seu título, *abstract* ou resumo. Após essa fase, cada estudo será validado a partir dos critérios de seleção e restrição para serem inclusos ou excluídos do estudo.

#### 5.1.2.6 Seleção das fontes de busca

As buscas foram realizadas apenas em fontes de busca automatizadas disponíveis na Internet e que possuem parceria com a Universidade Federal do Pará para acesso sem ônus financeiro aos pesquisadores. O Quadro 34 resume as bases de dados selecionadas para esta RSL.

Quadro 23 – Fontes Automáticas

Engenho de Busca	Link
IEEEExplore	<a href="http://ieeexplore.ieee.org">http://ieeexplore.ieee.org</a>
ScienceDirect	<a href="http://www.sciencedirect.com">http://www.sciencedirect.com</a>
El Compendex	<a href="http://www.engineeringvillage.com">http://www.engineeringvillage.com</a>
Elsevier Scopus	<a href="http://www.scopus.com/home.url">http://www.scopus.com/home.url</a>
ACM Digital Library	<a href="http://dl.acm.org">http://dl.acm.org</a>

Fonte: Elaboração própria (2019).



### 5.1.2.7 Identificação das palavras-chave e sinônimos

A formulação das palavras-chave e sinônimos é realizada para facilitar a concepção da *string* de busca. Para a sua identificação foi utilizada por base a questão principal da pesquisa, definida na Subseção 5.1.2.2.1 deste documento, respeitando a estrutura PICOC, proposta por Kitchenham e Charters (2007), sendo que apenas os itens População, Intervenção e Resultados foram considerados. Em consonância com as restrições da pesquisa, os idiomas definidos para as palavras-chave e sinônimos são o português e o inglês. Portanto, a partir da questão da pesquisa, têm-se a seguinte listagem de palavras identificadas:

- **POPULAÇÃO:**

Inglês: *learning programming, Computer Engineering, Computer Science, Information Systems, Information Technology, Software Engineering, Computation Degree, academic, teaching, learning, under graduation.*

Português: aprendizado de programação, Engenharia da Computação, Ciência da Computação, Sistemas de Informação, Tecnologia da Informação, Engenharia de Software, Licenciatura em Computação, acadêmico, ensino, aprendizado, graduação.

- **INTERVENÇÃO:**

Inglês: *process of teaching and learning in algorithms, introduction to programming, gamification.*

Português: processo de ensino e aprendizagem em algoritmos, introdução a programação, *gamificação*.

- **RESULTADOS:**

Inglês: *model, framework, method, technique, methodology, approach, knowledge, activity, tool, teaching tools, procedure, system, program, environment, task, software.*

Português: Modelo, *framework*, método, técnica, metodologia, abordagem, conhecimento, atividade, ferramenta, ferramentas de ensino, procedimento, sistema, programa, ambiente, tarefa, software.

### 5.1.2.8 Geração da *string* de busca

A composição da *string* de busca foi realizada dentro de três itens: População, Intervenção e Resultados. Dentro de cada um dos itens foram alocadas palavras sinônimas separadas pelo operador <OR>, o qual define pela lógica de programação que qualquer um dos

termos entrará na busca. Entre os itens foi utilizado o operador <AND>, pois determina que tais itens devem estar na pesquisa. Portanto, para a estrutura dessa pesquisa, tem-se a seguinte regra de filtragem de artigos {(População) AND (Intervenção) AND (Resultados)}. Logo, de acordo com a estrutura PIC) (ou PIO no caso desta pesquisa), conforme (SANTOS, 2010).

Portanto, a *string* de busca (vide Quadro 24) foi gerada a partir de termos relacionados às palavras-chave, os quais foram estruturados conforme o critério PICOC (*Population, Intervention, Comparison, Outcomes, Context*) proposto em (KITCHENHAM; CHARTERS, 2007), excluindo *Comparison* e *Context*, pois não fazem parte do escopo desta pesquisa. As palavras-chave foram extraídas das questões de pesquisa, a saber: *Population* (Algoritmos e graduação em tecnologia da informação); *Intervention* (Processo de ensino e aprendizagem de algoritmos e *gamificação*); e *Outcomes* (Abordagens e Softwares).

Quadro 24 – *String* de busca

<i>String</i> de busca
("learning programming" OR "Computer Engineering" OR "Computer Science" OR "Information Systems" OR "Information Technology" OR "Software Engineering" OR "Computation Degree" OR "academic" OR "teaching" OR "learning" OR "undergraduation") AND ("process of teaching and learning in algorithms" OR "introduction to programming") AND ("model" OR " <i>framework</i> " OR "method" OR "technique" OR "methodology" OR "approach" OR "knowledge" OR "activity" OR "tool" OR "teaching tools" OR "procedure" OR "system" OR "program" OR "environment" OR "task" OR "software").

Fonte: Elaboração própria (2019).

Para fins dessa revisão, entendeu-se por abordagem: métodos, metodologia, técnicas, ferramentas. Já quando se trata de cursos de nível superior, na área de tecnologia da informação, delimitou-se, mas não se restringe, aos cursos de Engenharia da Computação, Ciência da Computação, Sistemas de Informação, Tecnologia da Informação, Engenharia de Software, Licenciatura em Computação.

#### 5.1.2.9 Estratégia e seleção dos estudos primários

Após a execução das buscas, os estudos primários relevantes às questões de pesquisa foram selecionados por dois diferentes pesquisadores, tendo por base os critérios dispostos no Quadro 25, de acordo com os procedimentos, a seguir: (i) 1º Filtro para retirada de artigos, leitura independente do título, palavras-chave e resumo; e (ii) reunião de consenso, em caso de divergência entre os pesquisadores.

Além disso, a ferramenta Jabref de gerenciamento de referências foi utilizada para coletar os artigos e eliminar os duplicados de uma planilha eletrônica, utilizada para fazer a extração e identificação dos dados.

Quadro 25 – Critérios de seleção dos estudos primários

ID	Critério de Inclusão (CI) ou Critério de Exclusão (CE)
CI 1	Publicações que relatem metodologias educacionais no processo de ensino-aprendizagem de algoritmos com apoio da <i>gamificação</i> .
CE 1	Publicações que não tenham foco em cursos superior de TI, não restrito à Engenharia da Computação, Ciência da Computação, Sistemas de Informação, Tecnologia da Informação, Engenharia de Software e Licenciatura em Computação.
CE 2	Publicações do tipo relatórios de workshops, pôster, apresentações, <i>keynotes speaker</i> , livros, teses e dissertações.
CE 3	Publicações que não estejam acessíveis na íntegra na Web.
CE 4	Artigos publicados fora do período definido para a busca.
CE 5	Duplicidade de publicações (indexados) terão apenas sua primeira ocorrência considerada.
CE 6	Artigos que claramente não atendam as questões de pesquisa.
CE 7	Artigos não apresentados em uma das línguas aceitas (Inglês e Português).

Fonte: Elaboração própria (2019).

#### 5.1.2.10 Processo de seleção dos estudos primários

Durante a condução desta revisão sistemática, os estudos primários foram identificados conforme o processo seguinte:

1. A partir da leitura de resumo, introdução e conclusão, os artigos deverão ser avaliados quanto aos critérios de inclusão e exclusão e o resultado deve ser registrado.
2. Os dois pesquisadores responsáveis pela seleção dos artigos devem entrar em consenso, quando necessário. Isso ocorre quando não há uma unanimidade na inclusão de um artigo e pelo menos um pesquisador decidir incluir.
3. Na fase de consenso, em caso de discordância sobre a inclusão de algum estudo, o mesmo deverá ser incluído.

Os estudos primários identificados serão posteriormente lidos em totalidade e então será aplicada a avaliação de qualidade e a estratégia de extração de dados, conforme descrito nas subseções seguintes.

#### 5.1.2.11 Estratégia de avaliação da qualidade dos estudos primários

Segundo Kitchenham e Charters (2007), em adição aos critérios de inclusão e exclusão, é importante avaliar a qualidade dos estudos primários. Para tanto, dentro dessa revisão, os critérios de qualidade empregados foram adaptados da dissertação de Costa (2010), por descrever critérios suficientemente abrangentes para cobrir o escopo dos estudos a serem considerados. As modificações foram realizadas ao se considerar as questões de pesquisa que estão dispostas no Quadro 26.

Quadro 26 – Questões para avaliação da qualidade dos estudos

Item	Critérios de Qualidade
<b>Introdução/Planejamento</b>	
01	Os objetivos ou questões do estudo são claramente definidos (incluindo justificativas para a realização do estudo)?
02	O tipo de estudo está definido claramente?
<b>Desenvolvimento</b>	
03	Existe uma clara descrição do contexto no qual a pesquisa foi realizada?
04	O trabalho é bem/adequadamente referenciado (apresenta trabalhos relacionados ou semelhantes e baseia-se em modelos e teorias da literatura)?
<b>Conclusão</b>	
05	O estudo relata de forma clara e não ambígua os resultados?
06	Os objetivos ou questões do estudo são alcançados?
<b>Critérios para a questão de investigação</b>	
07	O estudo lista primária ou secundariamente os modelos, processos, métodos, técnicas, metodologia e afins, para apoiar o processo <i>gamificado</i> de ensino-aprendizagem de algoritmos?
08	O estudo apresenta ferramentas de apoio ao processo <i>gamificado</i> de ensino-aprendizagem de algoritmos?
<b>Critério específico para estudos experimentais</b>	
9	Existe um método ou um conjunto de métodos descrito para a realização do estudo?
<b>Critério específico para estudos teóricos</b>	
10	Existe um processo não tendencioso na escolha dos estudos?
<b>Critério específico para Revisões Sistemáticas</b>	
11	Existe um protocolo rigoroso, descrito e seguido?
<b>Critério específico para relato de experiência no ambiente acadêmico</b>	
12	Existe uma descrição sobre a(s) instituição(ões) onde foi conduzido o estudo?

Fonte: Adaptado de Costa, 2010.

#### 5.1.2.12 Processo de avaliação da qualidade

Os estudos primários selecionados por meio dos critérios de inclusão e exclusão foram lidos em sua totalidade e avaliados quanto aos critérios de qualidade. Para identificar o grau de adequação a esses critérios, foi adotada a escala Likert-5, estratégia proposta por Costa (2010) em que são possibilitadas respostas gradativas, sendo elas: 0 (discordo totalmente) a 4 (concordo totalmente). O Quadro 27 apresenta a escala de avaliação a ser utilizada, de acordo com a proposta de Costa (2010).

Quadro 27 – Escala Likert-5 para avaliação da qualidade de estudos (Continua...)

Numeração atribuída	Consideração
<b>Concordo totalmente (4)</b>	Deve ser concedido no caso em que o trabalho apresente no texto os critérios que atendam totalmente a questão.
<b>Concordo parcialmente (3)</b>	Deve ser concedido no caso em que o trabalho atenda parcialmente aos critérios da questão.
<b>Neutro (2)</b>	Deve ser concedido no caso em que o trabalho não deixe claro se atende ou não a questão.

Fonte: Elaboração própria (2019).

Quadro 27 – Escala Likert-5 para avaliação da qualidade de estudos (Conclusão)

Numeração atribuída	Consideração
<b>Discordo parcialmente (1)</b>	Deve ser concedido no caso em que os critérios contidos na questão não são atendidos pelo trabalho avaliado.
<b>Discordo totalmente (0)</b>	Deve ser concedido no caso em que não existe nada no trabalho que atenda aos critérios da questão.

Fonte: Elaboração própria (2019).

Para auxiliar a avaliação, seguindo a escala Likert-5, foi elaborado o Quadro 28, adaptado de Malcher (2015), no qual é considerado o critério para definir a aderência do estudo dentro da escala.

Quadro 28 – Exemplo de avaliação para os estudos (Continua...)

Escala por Critério	
Critério	Escala
<b>1</b>	4 - Define e justifica o estudo claramente. 3 - Define claramente o estudo, porém a justificativa não é clara. 2 - Define claramente o estudo, mas não justifica. 1 - A definição dos objetivos do estudo não é clara. 0 - Não define o estudo.
<b>2</b>	4 - Define o tipo de estudo, referenciando na literatura a metodologia. 3 - Define o tipo de estudo, porém sem referenciar a metodologia. 2 - Não define o tipo de estudo. É possível inferir facilmente. 1 - Não define o tipo de estudo. É possível inferir com dificuldade. 0 - Não é possível inferir o tipo de estudo.
<b>3</b>	4 - Define claramente uma seção com o contexto da pesquisa. 3 - O contexto da pesquisa está incluído em uma seção não exclusiva. 2 - O contexto da pesquisa está disperso ao longo do texto. 1 - O contexto da pesquisa está disperso e é insubstancial. 0 - O contexto da pesquisa não é abordado.
<b>4</b>	4 - O texto apresenta uma seção de trabalhos relacionados. 3 - O texto apresenta trabalhos relacionados em uma seção não exclusiva. 2 - O texto apresenta trabalhos relacionados dispersos ao longo do texto. 1 - O texto não apresenta trabalhos relacionados, mas se apoia na literatura. 0 - O texto não apresenta trabalhos relacionados nem se apoia na literatura.
<b>5</b>	4 - Resultados são claramente apresentados na seção de conclusão. 3 - Resultados são claramente referenciados na seção de conclusão. 2 - Resultados apresentados na conclusão não são claros. 1 - Resultados referenciados na conclusão não são claros. 0 - Não são apresentados resultados.
<b>6</b>	4 - Os resultados estão totalmente aderentes ao objetivo do estudo. 3 - Os resultados estão aderentes ao objetivo do estudo, no entanto o autor faz ressalvas. 2 - Os resultados são parcialmente aderentes ao objetivo do estudo. 1 - Os resultados não estão aderentes ao objetivo do estudo. 0 - Não é alcançado nenhum resultado.

Fonte: Elaboração própria (2019).

Quadro 28 – Exemplo de avaliação para os estudos (Conclusão)

Escala por Critério	
Critério	Escala
<b>7</b>	4 – Algum dos elementos é claramente descrito. 3 – Algum dos elementos é apresentado, mas não aprofundado. 2 – Algum dos elementos é avaliado, porém não descrito. 1 – Algum dos elementos é citado indiretamente. 0 – Nenhum dos elementos é apresentado direta ou indiretamente.
<b>8</b>	4 – Apresenta uma ferramenta disponível para uso em versão funcional. 3 – Apresenta ferramenta em versão funcional, porém não disponível para uso. 2 – Apresenta um projeto ou proposta de ferramenta, incluindo a descrição de funcionalidades. 1 – Apresenta apenas algoritmos de apoio ou análises de ferramentas de suporte ao processo de ensino e aprendizagem de Algoritmos. 0 – Não são apresentadas ferramentas de apoio.
<b>9</b>	4 – O método de experimento é definido e referenciado claramente. 3 – O método de experimento é definido claramente. 2 – O método de experimento é citado. 1 – O método de experimento não é citado, porém é possível inferir. 0 – Não é possível inferir o método de experimento.
<b>10</b>	4 – O texto descreve critérios para a escolha dos estudos. 3 – O texto não descreve critérios para a escolha dos estudos, porém apresenta estudos que discordam do estudo apresentado. 2 – O texto descreve apenas estudos aderentes ao estudo apresentado. 1 – O texto descreve estudos insuficientes. 0 – O texto não descreve estudos base.
<b>11</b>	4 – O protocolo de revisão é apresentado, descrito e seguido. 3 – O protocolo de revisão é apresentado e descrito, porém há evidências de que não foi seguido adequadamente. 2 – O protocolo de revisão não foi suficientemente descrito. 1 – O protocolo de revisão apenas foi citado ao longo do texto. 0 – Não há um protocolo de revisão.
<b>12</b>	4 – A área de atuação, tamanho e origem da organização são informados. 3 – Apenas duas das características do item 4 são informadas. 2 – Apenas uma das características do item 4 é informada. 1 – Nenhuma das características do item 4 é informada. 0 – O estudo não foi conduzido em uma ou mais organizações.

Fonte: Elaboração própria (2019).

No estudo de Beecham (2007), sobre RSL, define-se cinco níveis de qualidade para enquadrar os estudos que foram avaliados no critério de qualidade, sendo eles: excelente, muito boa, boa, média e baixa. Já em Costa (2007) tem-se uma construção de um intervalo estatístico para cada um dos níveis, apresentados no Quadro 29.

O cálculo para classificar o estudo no intervalo estatístico seguiu similar aos procedimentos já definidos por Costa (2007). Então, para a Revisão Sistemática desenvolvida, ao ser feita a análise dos critérios de qualidade pelos revisores, atribuímos um número da escala

para cada critério. Ao final efetuamos a soma e verificamos a porcentagem para enquadrar o estudo segundo os níveis de qualidade, descritos no Quadro 29.

Quadro 29 – Avaliação da Qualidade

Faixa de Notas	Avaliação
<b>Excelente</b>	>86%
<b>Muito Boa</b>	66%-85%
<b>Boa</b>	46%-65%
<b>Média</b>	26%-45%
<b>Baixa</b>	< 26%

Fonte: (COSTA, 2010).

#### 5.1.2.13 Estratégia de extração e síntese dos dados

No decorrer desta pesquisa todos os estudos primários após o emprego dos critérios de inclusão e exclusão foram lidos na íntegra para classificar a qualidade e com a finalidade de identificar informações relevantes para a extração dos dados. A extração foi planejada e desenvolvida de forma estruturada segundo o processo sugerido por Cruzes e Dybå (2011). Para tanto, adotou-se uma planilha eletrônica para a coleta dos seguintes dados: (i) publicação (autor, referência, editor e ano); (ii) contexto (tipo de estudo, métodos de pesquisa, análise dos dados, tamanho da amostra); e (iii) evidências (processo de avaliação e validação, uso de ferramentas, trechos interessantes do texto e referente a uma das questões de pesquisa). Logo, foi possível identificar a partir dos dados coletados as informações relevantes dos estudos para responder as questões da pesquisa.

A técnica de análise temática dos dados, segundo as recomendações indicadas por Cruzes e Dybå (2011), foi o método escolhido para a síntese dos dados. Com a leitura integral dos artigos selecionados e do uso da planilha de extração foram registrados: (i) as abordagens *gamificadas* adotadas; (ii) o processo de validação e/ou avaliação; (iii) o uso de ferramenta para suportar a *gamificação*; (iv) o resultado do processo *gamificado*; e (v) os elementos de jogos utilizados. As informações registradas foram agrupadas em categorias para a criação de modelos temáticos com o objetivo de responder as questões de pesquisa.

### 5.1.3 Sumarização dos resultados da Revisão Sistemática

Esta pesquisa selecionou um total de 61 estudos primários relevantes ao tópico desta investigação, após a aplicação dos critérios de seleção, conforme ilustrado na Tabela 1. Esses estudos primários estão listados por meio de identificadores (ID), com o acrônimo *Primary Study* (PS), para uma melhor referência ao longo do texto, conforme pode ser visualizado acessando o Apêndice G.

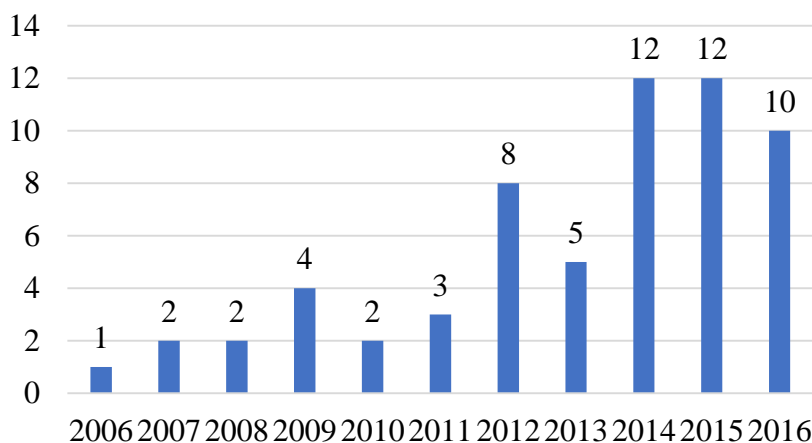
Tabela 1 – Seleção dos estudos primários

Bases	Quantidade entradas	CI 01	CE 01	CE 02	CE 03	CE 04	CE 05	CE 06	CE 07
<b>IEEE</b>	3200	27	48	42	0	0	0	3083	0
<b>Science Direct</b>	720	5	21	22	0	0	0	669	3
<b>Engineering Village</b>	1754	10	51	88	0	0	21	1584	0
<b>Scopus</b>	2444	8	52	86	0	0	17	2281	0
<b>ACM</b>	1909	11	52	29	5	0	0	1812	0
<b>Total</b>	10027	61	224	267	5	0	38	9429	3

Fonte: Dados da pesquisa (2019).

Como forma de mapear os artigos aceitos, foi feita uma busca quantitativa dos resultados, os quais apresentaram uma distribuição dos estudos em relação ao ano de publicação. Pode-se perceber uma crescente publicação no decorrer dos anos, ou seja, apresenta-se um interesse pelas publicações sobre *gamificação* no contexto de ensino-aprendizagem de Algoritmos em cursos de graduação em Tecnologias da Informação. Os dados estão presentes na Figura 70.

Figura 70 – Distribuição dos trabalhos em relação aos anos

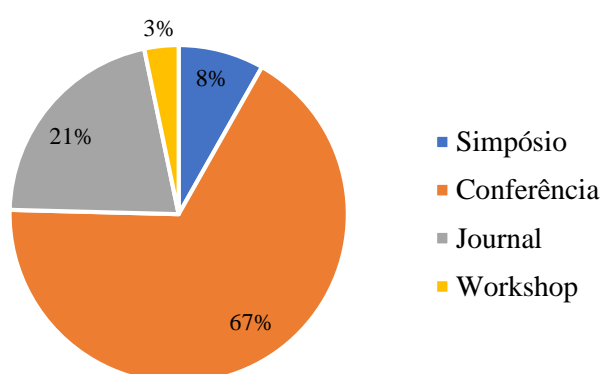


Fonte: Dados da pesquisa (2019).



Um dos dados elaborados por essa revisão sistemática é a distribuição dos artigos pelo tipo de publicação. Esse dado apresenta uma realidade da área de ciência da computação, em que a maioria dos artigos está presente em eventos identificados pelos resultados como Simpósio, Conferência e Workshop, com um total de 79%. Já no que diz respeito a periódicos, foi obtido apenas 21% de artigos, sendo os seus identificadores: PS006, PS007, PS008, PS016, PS020, PS024, PS039, PS044, PS045, PS046, PS050, PS055, PS056. Os dados sobre a distribuição dos tipos de publicações estão dispostos na Figura 71.

Figura 71 – Distribuição por tipo de publicação



Fonte: Dados da pesquisa (2019).

A Revisão Sistemática da Literatura gerou um trabalho de conclusão de curso (MENEZES, 2018), intitulado: “Abordagens *gamificadas* para o ensino e aprendizagem de algoritmos ou equivalente: um estudo a partir do Mapeamento Sistemático da Literatura”, no qual o discente gerou um Mapeamento Sistemático da Literatura. No estudo em questão estão descritos e apresentados com maiores detalhes os dados quantitativos da Revisão Sistemática da Literatura.

#### 5.1.3.1 Qualidade dos estudos primários

A fim de garantir dados empíricos de bons estudos primários, para a avaliação da qualidade desses estudos, foi utilizada a escala *Likert-5* para os 11 critérios de qualidade adaptados de (COSTA, 2010), ilustrados no Quadro 26. Os estudos foram enquadrados em cinco níveis de qualidade: Excelente, muito boa, boa, Média e baixa, como apresentado na Tabela 2.

Tabela 2 – Qualidade dos estudos primários

Qualidade	Quantidade de Estudos
Excelente	3
Muito boa	7
Boa	28
Baixa	0

Fonte: Elaboração própria (2019).

Os estudos considerados excelentes são: PS003, PS004 e PS005. Um registro interessante a ser feito é que não teve nenhum artigo considerado com o nível de qualidade baixa, e a maior quantidade de artigos está presente na categoria boa.

#### 5.1.3.2 Abordagem mais utilizada para *gamificar* o processo de ensino-aprendizagem da disciplina Algoritmos ou equivalente (QS01)

Durante a extração e a síntese dos dados apresentados pelos estudos inclusos, foi possível perceber que a maioria desenvolve uma nova abordagem para trabalhar o processo de ensino-aprendizagem de algoritmos. Porém, foram encontradas quatro abordagens, as quais são adotadas em dois estudos. São elas: Trakla2; Light 2.0 e CodeCombat; Programe o seu Robô; e Wu'sCastle; as quais são relacionadas com os ids do estudo na Quadro 30.

Nos estudos PS020 e PS035 a abordagem *gamificada* utiliza a plataforma Trakla2 como base para o seu experimento. Nesse contexto, os professores adotam alguns dos elementos de *game*, como é o caso de medalhas, recompensas e insígnias. Outro dado interessante é a adoção do grupo de controle e sem controle para a validação. No que tange ao resultado, este foi relatado que as notas não foram afetadas, porém em relação ao comportamento dos alunos foi alterado positivamente.

Os artigos PS021 e PS022 relatam a avaliação do uso dos Lightbot 2.0 e CodeCombat para o ensino de programação de computadores, disciplina equivalente a Algoritmos. Nestes estudos os pesquisadores identificam a contribuição dos jogos para o ensino de programação e relatam a necessidade de alguns elementos que auxiliam os jogadores nesse processo, tais como: tutoriais, *feedback* e os níveis de dificuldade.

Nos estudos definidos por PS045 e PS046 são apresentados o jogo Programe o seu Robô para ensino de conceitos básico de Ciência da Computação, dentre eles os de algoritmos. Portanto, para essa abordagem, foi desenvolvido um jogo em que os alunos adquirem conhecimentos sobre algoritmos no decorrer do jogo. Como forma de avaliar a abordagem, um teste foi realizado com alunos com a finalidade de coletar *feedbacks* para melhorias.

Por fim, na abordagem *Wu's Castle*, presente nos artigos PS053 e PS057, os autores relatam o desenvolvimento e teste com o uso do jogo para o ensino de Estrutura de Dados Homogêneos e estrutura de repetição, conteúdos da disciplina Algoritmos. O teste de eficiência para o jogo foi a partir de uma comparação entre alunos jogadores e não jogadores na prova da disciplina e no desenvolvimento dos algoritmos. Outro ponto retratado é o *feedback* como elemento de jogo, pois o intuito era permitir ao jogador visualizar o seu progresso ao programar.

Quadro 30 – Abordagens adotadas para *gamificar* a disciplina Algoritmos ou equivalente

Abordagens adotadas	Artigos
Trakla2	PS020 e PS035
Lightbot 2.0 e CodeCombat	PS021 e PS022
Programe o seu Robô	PS045 e PS046
Wu'sCastle	PS053 e PS057

Fonte: Elaboração própria (2019).

### 5.1.3.3 Principais dificuldades encontradas ao se *gamificar* o ensino da disciplina Algoritmos ou equivalente (QS02)

Para responder a essa questão de pesquisa, foi realizado um mapeamento dos estudos, em que foi possível encontrar as seguintes dificuldades no processo apresentado:

- No estudo PS005 ao se *gamificar* foi relatada a dificuldade na estrutura da classe, pois era adotada uma forma tradicional, diferente da *gamificada*.
- No artigo PS009 a ferramenta desenvolvida exigia alto desempenho computacional, não tinha confiabilidade e apresentava erros em algumas funções solicitadas pelo usuário. Outra situação descrita pelos autores foi a documentação para a aplicação, que era escassa, logo, à medida que o aluno progredia na ferramenta não conseguia realizar ações de maior complexidade.
- Na pesquisa PS010 os autores coletaram alguns *feedbacks* dos alunos sobre o jogo desenvolvido e um apontamento relevante foi a necessidade de aprimorar os tutoriais disponíveis.
- A dificuldade encontrada no estudo PS048 e informada pelos usuários na primeira aplicação foi o de disponibilizar no *software* um *feedback* para os usuários. E, em uma

segunda coleta de informações, os alunos relataram a necessidade do desenvolvimento de um tutorial.

- Já no artigo PS010 a melhoria solicitada pelos usuários foi no agente inteligente da aplicação, denominado como tutor, pois apresentava alguns erros no auxílio aos estudantes dentro da aplicação.

Em resumo, as principais dificuldades encontradas nos estudos ao se *gamificar* o ensino da disciplina Algoritmos ou equivalente dizem respeito: (i) ao elemento *feedback*, ou seja, informar ao aluno o nível que alcançou na *gamificação*, seus erros e o que precisa melhorar no desenvolvimento do seu processo de ensino e aprendizagem; e (ii) à elaboração da documentação para auxiliar os discentes a usarem o software disponível durante a sala de aula *gamificada*.

#### 5.1.3.4 Principais dificuldades encontradas no aprendizado *gamificado* da disciplina Algoritmos ou equivalente (QS03)

Assim como foi realizado o mapeamento para as dificuldades de ensino no decorrer da extração dos dados, realizou-se a coleta das dificuldades em relação ao aprendizado. Sobre essas dificuldades, dois artigos chamaram atenção:

- O artigo de PS011 relatou que o elemento competitividade superou o objetivo de ensino proposto, revisão de conteúdo. Essa realidade foi percebida em apenas um aluno da turma, em que tal situação permite inferir que a competitividade ainda pode ser utilizada como um elemento de jogo nas abordagens *gamificadas*.
- No estudo PS059 o registro de problema foi quando os discentes precisaram utilizar o software para a programação, pois estavam acostumados com a aplicação que promovia auxílio na sala de aula *gamificada*. Porém, a situação em questão foi resolvida a partir da aprendizagem pareada com ambiente colaborativo, em que os discentes, que tinham maior conhecimento sobre os conteúdos e uso do software da disciplina, auxiliavam os outros.

### 5.1.3.5 Processo *gamificado* de ensino-aprendizado da disciplina Algoritmos ou equivalente (QS04)

As abordagens *gamificadas* para o ensino-aprendizagem da disciplina Algoritmos ou equivalentes compartilham dois componentes básicos: o *framework* base para a *gamificação* e os elementos de jogos trabalhados. Então, com o intuito de responder a essa questão de pesquisa foram identificados nos estudos esses dois componentes.

O Quadro 31 apresenta os *frameworks* base para o desenvolvimento dos estudos, bem como os seus identificadores para esta Revisão Sistemática. Em alguns dos estudos selecionados não foi relatada a informação disposta no quadro. Porém, vale ressaltar que todos os estudos apresentam fundamentação teórica para o processo *gamificado* elaborado. Os mais citados e utilizados são: as características disponíveis no livro de Kapp (2012); e o software para elaboração de abordagens *gamificadas* *Karel the Robot*.

Quadro 31 – *Frameworks* adotados como base para o processo de *gamificação* e os ids dos artigos

<i>Framework</i> adotado para embasar a <i>gamificação</i>	Artigos
Kapp (2012)	PS011 e PS018
Karel the Robot	PS029 e OS 059
Pex4Fun	PS037
Rummyor Go Fish	PS048
Resource Craft	PS055
Game2Learn's	PS057

Fonte: Elaboração própria (2019).

O segundo dado trabalhado para responder a essa questão de pesquisa foi o mapeamento dos elementos de *games* adotados nos estudos primários (vide Quadro 32). Os principais elementos de *games* trabalhados são *feedback*, pontos, ranking, *score*, níveis de dificuldade, desafios e o uso de *avatar*.

O elemento *feedback* está relatado nos dados de *game* disponíveis nessa revisão sistemática e também é apresentado como uma das dificuldades no processo tradicional de ensino-aprendizagem de Algoritmos, relatado na revisão informal da literatura. Portanto, como na primeira versão do produto dessa dissertação já foi adotado sendo um dos elementos de jogos, observamos a sua importância na continuidade das versões construídas.

Portanto, temos como base para a *gamificação* as características apontadas no livro (KAPP, 2012), bem como o software para elaboração de abordagens *gamificadas* *Karel the*

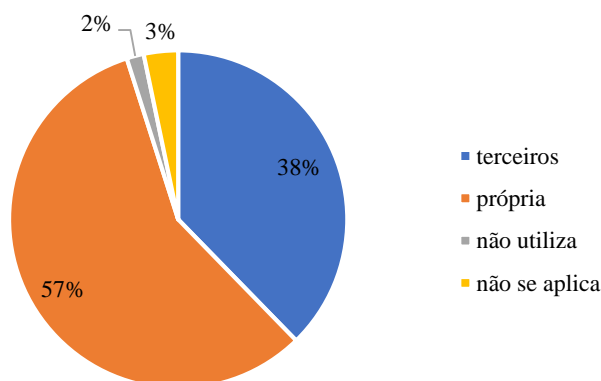
*Robot*. Outra informação relevante disponível é o catálogo de elementos de jogos utilizados na literatura com sucesso para *gamificar* a sala de aula da disciplina Algoritmos ou equivalente.

Quadro 32 – Elementos de *games* adotados para *gamificar* o processo de ensino-aprendizagem de algoritmos

Elementos de jogos	Artigos
<b>Feedback</b>	PS002, PS003, PS004, PS005, PS048, PS051, PS052, PS053, PS054, PS055, PS057, PS058
<b>Pontos</b>	PS001, PS009, PS050, PS054, PS055, PS057, PS058
<b>Ranking</b>	PS001, PS009, PS011, PS053, PS055, PS057
<b>Score</b>	PS001, PS009, PS011, PS053, PS055, PS057
<b>Níveis de dificuldades</b>	PS001, PS048, PS050, PS053, PS054, PS058
<b>Desafios</b>	PS001, PS048, PS052, PS056, PS057, PS058
<b>Avatar</b>	PS050, PS053, PS054, PS057
<b>Simulação</b>	PS002, PS003, PS050
<b>Bônus</b>	PS009, PS053
<b>Recompensas</b>	PS009, PS052
<b>Missões</b>	PS001, PS056
<b>Narrativa Épica</b>	PS053, PS054
<b>Fases</b>	PS054
<b>Prêmio</b>	PS001
<b>Emblemas</b>	PS009
<b>Leaderboards</b>	PS009
<b>Equipe</b>	PS011
<b>Penalidades</b>	PS050
<b>Regras</b>	PS052
<b>Metas</b>	PS052
<b>História</b>	PS052
<b>Diversão</b>	PS052
<b>Recurso</b>	PS055
<b>Participação Voluntária</b>	PS057

Fonte: Dados da pesquisa (2019).

A Figura 72 apresenta o uso de ferramentas para embasar a *gamificação*, na qual obtemos a seguinte situação: (i) 57% dos estudos desenvolveram uma nova ferramenta; (ii) 38% dos estudos adotaram de terceiros; (iii) 2% dos estudos não utilizam; e (iv) 3% dos estudos não se aplicam, ou seja, são as revisões da literatura. No relato dos estudos podemos observar que 51% está disponível para uso, enquanto em 49% não.

Figura 72 – Uso de ferramentas para embasar a *gamificação*

Fonte: Dados da pesquisa (2019).

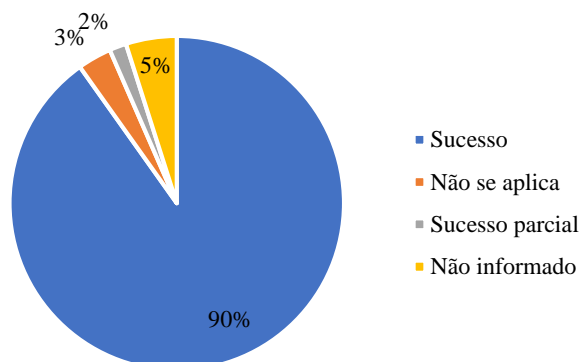
A informação descrita pelo uso de ferramentas para embasar a *gamificação* colabora com os elementos de jogos, bem como com os *frameworks* base para poder responder como *gamificar* a disciplina Algoritmos. Nesse aspecto, um dos trabalhos futuros aos quais essa Revisão Sistemática aponta para a pesquisa de mestrado é a construção de uma ferramenta e a sua disponibilização para embasar o uso do *framework gamificado* para o ensino-aprendizagem de Algoritmos.

#### 5.1.3.6 Sucesso do processo *gamificado* de ensino-aprendizagem da disciplina Algoritmos ou equivalente (QS05)

Após a extração dos estudos, verificou-se que em 90% destes o processo de *gamificação* da disciplina Algoritmos ou equivalente foi considerada bem sucedida pelos próprios autores. Os outros 10% foram divididos em: (i) não se aplica (5%), em que dois estudos (PS016 e PS028), por serem revisão sistemáticas da literatura, não relataram diretamente a intervenção no processo; (ii) não informada (2%), visto que os autores de três estudos (PS014, PS056 e PS059) não se posicionaram sobre a avaliação do processo; e (iii) sucesso parcial (2%), visto que os autores do estudo (PS018) informam que o uso da abordagem ainda está acontecendo. A Figura 73 sistematiza em gráfico estas informações.

Vale ressaltar que no que tange ao sucesso e insucesso, estamos levando em consideração as informações discutidas pelos autores dos artigos.

Figura 73 – Informação sobre o sucesso no uso de *gamificação* como base para o processo de ensino-aprendizagem de Algoritmos



Fonte: Dados da pesquisa (2019).

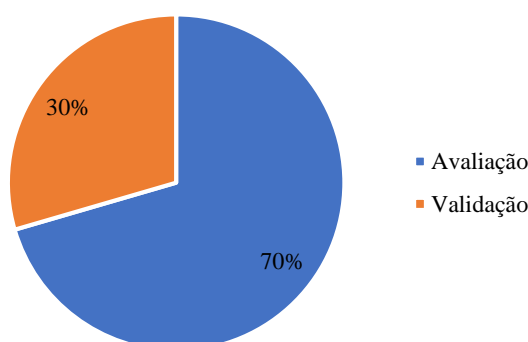
#### 5.1.3.7 Insucesso do processo *gamificado* de ensino e aprendizagem da disciplina Algoritmos ou equivalente (QS06)

Dentro dos estudos selecionados por esta revisão sistemática não foram encontrados relatos de insucesso nos estudos, porém, como informado já na Figura 73, tivemos um estudo com sucesso parcial.

#### 5.1.3.8 Ocorrência de avaliação do processo *gamificado* de ensino-aprendizagem da disciplina Algoritmos ou equivalente (QS07)

A Figura 74 apresenta quantitativamente o processo pelo qual foram analisados os estudos selecionados, sendo que em: 70% ocorreu avaliação; e 30% foi realizado validação.

Figura 74 – O uso de validação e/ou avaliação nos estudos incluídos



Fonte: Dados da pesquisa (2019).



### 5.1.3.9 Ocorrência de validação do processo *gamificado* de ensino-aprendizagem da disciplina Algoritmos ou equivalente (QS08)

Durante a síntese dos dados da revisão sistemática foi possível verificar que em 18 estudos foram adotados processos de validação, segundo o relato realizado pelos autores dos trabalhos. Portanto, respondemos positivamente essa questão de pesquisa. Um relato mais detalhado sobre o método adotado e os identificadores dos artigos que realizaram a validação está presente na próxima subseção.

### 5.1.3.10 Validação do processo *gamificado* de ensino-aprendizagem da disciplina Algoritmos ou equivalente (QS09)

O Quadro 33 apresenta os métodos descritos nos estudos primários aceitos. São eles: validação com grupos especialistas, grupo com controle e sem controle, comparação entre duas ferramentas disponíveis; e a validação do *design* adotado ao processo *gamificado*.

Então, é possível averiguar o método mais utilizado pela literatura para atestar a validade da proposta por meio de grupo de controle e sem controle.

Quadro 33 – Métodos de Validação adotados nos artigos inclusos

Método de Validação	Quantidade	Artigos
Validação em grupos	01	PS003
Grupo com controle e sem controle		PS004, PS005, PS006, PS007, PS008, PS014, PS031, PS035, PS039, PS049, PS053, PS054, PS057
Entre duas ferramentas	01	PS021
Em relação ao design	01	PS022

Fonte: Elaboração própria (2019).

### 5.1.4 Ameaças à validade da RSL

Em revisões sistemáticas, o viés de seleção dos estudos primários é uma das principais ameaças. Para minimizar essa questão, adotou-se a busca automática e sucessíveis testes para a *string* de busca. Além disso, todas as etapas da revisão sistemática foram realizadas por dois pesquisadores, com supervisão de outros dois.

Outra estratégia adotada foram reuniões de consenso caso houvesse divergência na etapa de seleção dos estudos primários e de avaliação da qualidade. Essa reunião acontecia entre os dois pesquisadores, que estavam realizando a leitura e avaliação dos estudos, em que

informavam o motivo da nota atribuída ao estudo e analisavam qual a melhor para caracterizar o critério avaliado.

Uma possível dificuldade que poderia existir, seria a categorização das abordagens *gamificadas*, porém, como foi adotada a técnica de Síntese e Análise Temática (CRUZES; DYBÅ, 2011), esse problema não se apresentou nessa pesquisa.

### 5.1.5 Conclusões da RSL

Como forma de atender ao objetivo de identificar abordagens *gamificadas* de apoio ao processo de ensino-aprendizagem de algoritmos nos cursos de graduação em tecnologia da informação existentes na literatura, uma RSL foi conduzida, no qual questões de pesquisas foram estabelecidas e respondidas ao longo deste estudo. Como resultado foram selecionados 61 estudos primários relevantes para serem avaliados.

Um total de 57 abordagens *gamificadas* para o processo de ensino-aprendizagem na disciplina Algoritmos ou equivalentes foram identificadas, sendo que apenas 4 são trabalhadas em duas pesquisas, são elas *Trakla2*, *Lightbot 2.0* e *Code Combate*, *Programo o seu Robô* e *Wu's Castle*. Outro importante dado mapeado por essa RSL foram os elementos de jogos, em um total de 24, em que os mais citados foram: *feedback*, pontos, *ranking*, *score*, níveis de dificuldade, desafios e *avatar*.

Observamos que o método mais utilizado na literatura para validar o processo foi o de grupo com controle e sem controle, e que em sua grande maioria (90%) a intervenção na disciplina de algoritmos via *gamificação* é bem sucedida.

Logo, essa Revisão Sistemática apresenta o seguinte conjunto de informações sobre a *gamificação* do processo de ensino-aprendizagem da disciplina Algoritmos e equivalentes: (i) os *frameworks* base adotados; (ii) os elementos de jogos trabalhados; (iii) a utilização de ferramentas para suportar a *gamificação*; (iv) as dificuldades em relação ao ensino e aprendizagem; e (v) as formas de validação das propostas.

## 5.2 REQUISITOS

A segunda versão do *framework gamificado* tem como base a primeira versão com a inclusão dos novos requisitos vindos das melhorias apontadas pela análise do experimento e dos resultados apontados pela Revisão Sistemática da Literatura. Como forma de dispor as informações em relação a sua origem, no que tange às melhorias, foi construído o Quadro 34.

Quadro 34 – Rastreabilidade dos novos requisitos do *framework*

Itens do <i>framework</i>	Insumo	Origem
Simulação	Regras de teste para a lista de desafio com o intuito de concessão de pontos	Revisão Sistemática da Literatura e Experimento
Níveis de dificuldade	Acrescentar um nível crescente de dificuldade na resolução das listas de desafios no decorrer das aulas	Revisão Sistemática da Literatura e Experimento
Prêmio	Concessão de prêmio aos mais pontuados e bonificados na <i>gamificação</i>	Revisão Sistemática da Literatura e Experimento
Histórico (Escopo aberto)	Na formulação dos desafios presentes na disciplina utilizar questões de escopo aberto bem como exemplos de aplicações reais dos conceitos.	Revisão Sistemática da Literatura e Experimento
Formato do Dojo <i>Randori</i>	As mudanças indicadas pelo experimento, tais como: Papel do copiloto opcional, mudança das pontuações por bonificações das atribuições realizadas no dia e atribuição de pontuação unicamente pelo código desenvolvido pela turma.	Experimento
Prática LAB com e sem pontuação	Dois dias para a prática do LAB, um para que os participantes tenham contato e vejam como será a prática e o segundo dia com a pontuação devida para o dia.	Experimento

Fonte: Elaboração própria (2019).

Portanto, as alterações coletadas, tanto pelo experimento quanto pelos requisitos apresentados na Revisão Sistemática da Literatura, convergem para as inclusões e alterações dos mesmos itens do *framework gamificado* (vide Quadro 34).

### 5.3 FRAMEWORK GAMIFICADO VERSÃO 02: O PRODUTO FINAL

O produto educacional da pesquisa em tela na sua segunda versão tem a estrutura já definida e desenvolvida para a primeira versão com as implementações dos requisitos levantados pelo primeiro experimento e a Revisão sistemática da Literatura. Os componentes base do *framework* são o plano de ensino e a planilha *gamificada*, (vide Capítulo 3). Portanto, as implementações de melhoria foram trabalhadas nesses documentos e explanadas nesta seção.

O primeiro requisito é o elemento de jogo simulação. Vale ressaltar o seu uso na primeira versão, com os testes de mesa e também o uso do Pascalzim para visualizar o programa em execução. A mudança é nas listas de desafios das aulas teóricas e práticas, em que para cada desafio terá indicações de testes para assegurar se a questão foi completada. A mudança afetará

as listas de desafios de cada uma das práticas trabalhadas (Apêndices A e B do plano de ensino), como exemplo temos o Quadro 35.

Quadro 35 – Exemplo de uma questão trabalhada na Aula Teórica e Prática antes e depois da mudança

Antes	Depois
1) Desenvolva um algoritmo que leia do usuário um vetor de 10 posições e calcule e exiba o número armazenado mais o índice da posição que ele está armazenado.	1) Desenvolva um algoritmo que leia do usuário um vetor de 10 posições e calcule e exiba o número armazenado mais o índice da posição que ele está armazenado.  Condição para concessão da pontuação: Inserir no programa: 1 2 3 4 5 6 7 8 9 10 O programa deve apresentar: 1 1 2 2 3 3 4 4 5 5 6 6 7 7 8 8 9 9 10 10

Fonte: Elaboração própria (2019).

Os níveis de dificuldades serão também implementados dentro das listas de desafios das aulas teóricas e práticas, em desafios com níveis crescentes de dificuldades. A implementação aconteceu no plano de ensino com a descrição do elemento e a regra de atribuição de pontuação proporcional ao nível de complexidade da questão, a alteração está disponível na seção 5.2.1 do plano de ensino. A planilha *gamificada* está modelada para a concessão automática dos pontos para o estudante que completar a atividade.

A mecânica de jogo competição é instigada pelo elemento prêmio. Para a segunda versão ele será implementado dentro do plano de ensino com um relato do mesmo e concederá aos ganhadores das medalhas *Power* e Participativo um curso online na plataforma Udemy (UDEMY, 2018). Como ponto de melhoria na planilha *gamificada*, para a aquisição da medalha Participativo e, por conseguinte um dos prêmios, foi definido um critério de desempate para essa conquista. O critério é acumulado de bonificações sem a conversão para o recurso estrela. Informações sobre essa inclusão estão disponíveis na seção 5.7 do plano de ensino.

Quadro 36 – Informações sobre a seção de medalhas da *gamificação* e premiação no plano de ensino

Seção medalhas da <i>gamificação</i> e premiação no plano de ensino
A concessão de medalha para a <i>gamificação</i> acontecerá no final das aulas e poderá ser acompanhada pelos discentes por meio da aba Medalhas. Para a <i>gamificação</i> da nossa sala de aula, teremos dois tipos de medalha: Power, referente as pontuações adquiridas no decorrer do processo e a Participativo, em relação ao recurso estrela adquirido. Para cada medalha teremos três posições, são elas: 1º lugar, ouro; 2º lugar, prata; 3º lugar, bronze. O critério de desempate para a medalha Participativo será o número total de bonificações sem a conversão para recurso estrela.  Os primeiros lugares tanto na medalha Power quanto na medalha Participativo ganharão o seguinte prêmio: um curso online pela plataforma Udemy.

Fonte: Elaboração própria (2019).

A componente história refere-se a uma narrativa construída sobre como o algoritmo deve se comportar em relação ao que lhe é solicitado. O escopo aberto tem por definição a construção da atividade, possibilitando ao aluno, vários caminhos para a solução. Portanto, consideramos como um único requisito, impactando na elaboração da lista de desafios da prática do Dojo *Randori*. Ele refletiu no plano de ensino (apêndice D do plano de ensino) e também na planilha *gamificada*, com aquisição de bonificação por sugestão para diferentes soluções propostas pelos alunos em relação aos questionamentos. Um exemplo de questão de como era antes e como está na nova versão, segue disponível no quadro.

Quadro 37 – Exemplo de uma questão trabalhada na prática Dojo *Randori* antes e depois da mudança

Antes	Depois
Desenvolva um programa em Pascal que contenha uma lista de supermercado de uma pessoa, usando um vetor de string com 10 posições. As funcionalidades que o programa precisa apresentar ao usuário, por meio de um menu com 06 opções, são: 1) Incluir todos os itens da lista. 2) Imprimir toda a lista. 3) Apresentar um item específico da lista, por meio do índice. 4) Incluir um item da lista, a partir de um índice específico. 5) Excluir um item da lista, a partir de um índice específico. 6) “Zerar” a lista (todos os elementos vão conter branco dentro). Ao digitar o número 0, o programa finaliza a execução.	Uma empresa foi contratada para criar um programa para uma loja. O programa tem que ser capaz de: incluir e excluir as vendas de cada semana dos meses do ano; alterar as vendas de uma determinada semana de um mês específico; exibir o total vendido em cada mês do ano; exibir o total vendido no ano e, ainda, em que mês(es) a loja alcançou o mínimo desejado para vendas (R\$ 5.000,00). Para fins de simplificação, considere que cada mês do ano possui somente 04 semanas

Fonte: Elaboração própria (2019).

A segunda versão do *framework gamificado* encontra-se disponível para uso no link para download:

- Plano de ensino: <http://twixar.me/w811>
- Planilha *gamificada*: <http://twixar.me/b811>

Logo, podemos identificar que não temos tantas mudanças dentro da estrutura do produto, porém registramos a significância das mesmas, porque acrescentamos um elemento estimulador da competição, ampliamos o escopo de resolução de questões, o qual pode suprir a dificuldade encontrada para pontuar a bonificação (vide Capítulo 4). Além disso, contaremos com testes pré-definidos como demonstração da assertividade das questões.

## CONSIDERAÇÕES FINAIS

A disciplina Algoritmos ou equivalente é um importante componente curricular para a continuidade dos discentes nos cursos de graduação em tecnologia da informação, com especial ênfase no desenvolvimento de software. Dentro da literatura e na Universidade Federal do Pará podemos constatar índices elevados de reprovação e evasão na disciplina. Além disso, verificamos várias dificuldades tanto para o ensino, quanto para o aprendizado dentro desse componente curricular.

Para auxiliar a alterar essa realidade, no decorrer da pesquisa, identificamos na abordagem *gamificada* uma estratégia possível para o professor empregar em sala de aula no desenvolvimento de suas atividades. Por meio do uso de elementos de jogos, a *gamificação* estimula os personagens a realizar atividades ou aprender os conteúdos.

Diante deste contexto e a partir da questão-foco estabelecida, propusemos um *framework gamificado* para a disciplina Algoritmos ou equivalente, o qual foi concebido, desenvolvido, testado, avaliado e validado por diferentes procedimentos metodológicos:

- *Survey*: retrato sobre a disciplina Algoritmos tanto na visão docente, quanto na visão discente, com informações sobre: a abordagem didática mais significativa para o ensino e aprendizagem, a lista de exercício como recurso mais adotado, o formato de divisão dos discentes para resolução das atividades, uso de bonificação, *feedback* das atividades e o conteúdo mais difícil. A pesquisa foi disponibilizada *online*, por meio de questionários eletrônicos. Os resultados permitiram visualizar como a disciplina vem sendo trabalhada e vivenciada a nível nacional, constituindo-se como uma das bases para a elaboração do *framework gamificado*.
- Observação participante: forma de entendimento da interação entre o conteúdo e os personagens da disciplina (professor, monitor e alunos). Esse procedimento metodológico foi adotado para a coleta de informações em duas ofertas da disciplina da Faculdade de Computação, da Universidade Federal do Pará. Após a coleta, conseguimos identificar requisitos para o desenvolvimento do produto dessa dissertação de mestrado.

- Revisão por pares: avaliação realizada no *framework gamificado* para o processo de ensino e aprendizagem de Algoritmos, feito por um grupo de especialistas em Algoritmos e *gamificação*.
- Validação do produto: A validação do produto dessa dissertação de mestrado foi por meio do procedimento metodológico de grupo de controle e sem controle. Foi adotado como parâmetro de comparação das notas adquiridas. O grupo de controle aconteceu na turma de 2016 e o grupo sem controle foi na turma de 2018. As turmas foram ofertadas pela Faculdade de Computação da Universidade Federal do Pará. Adotou-se o conteúdo de estrutura de dados homogêneos para a comparação e identificação do sucesso da abordagem *gamificada*.
- Revisão sistemática da Literatura: com ajuda de um protocolo, possibilitou identificar 57 abordagens *gamificadas* para trabalhar o processo de ensino e aprendizagem da disciplina Algoritmos, em um total de 61 artigos inclusos na pesquisa. Além disso, originou um Trabalho de Conclusão do Curso de Bacharelado em Sistemas de Informação da Universidade Federal do Pará.

Os resultados de cada etapa da pesquisa, mais especificamente o experimento e a validação da abordagem *gamificada*, possibilitaram responder a questão-foco, bem como atingir ao objetivo geral traçado. Assim, dentre os principais resultados alcançados, destacamos:

- Maior participação dos alunos em sala.
- Maior colaboração entre os alunos durante a resolução dos desafios propostos em sala.
- Maior engajamento para resolução das listas de exercícios.
- Melhora no desempenho da maior parte dos alunos no que diz respeito a nota avaliativa.
- Emprego de práticas dinâmicas e lúdicas em sala de aula.
- *Feedback* em tempo hábil do progresso dos alunos na disciplina.
- Aceitação da abordagem por parte dos alunos.

O *framework gamificado* como proposta de produto dessa dissertação, após todas as alterações, mostrou-se adequado e oportuno para mitigar problemas de ensino e aprendizado na disciplina Algoritmos ou equivalente. Entretanto, também possui limitações percebidas principalmente durante o experimento realizado durante a disciplina Algoritmos no curso de Bacharelado em Ciência da Computação da Faculdade de Computação da Universidade Federal do Pará. Como exemplos citam-se:

- O professor para fazer um uso adequado do *framework* necessita de uma infraestrutura mínima de recursos, que nem sempre está disponível; tais como: laboratório de informática, apoio de monitor e uma pessoa para atuar como juiz/scoreplay.
- O *framework* proposto não pressupõe a presença de alunos com deficiências e com temperamento introvertido, mas não se sabe definir se é uma limitação do produto ou da própria abordagem *gamificada*.

Além das limitações descritas sobre o produto, relato também as minhas dificuldades no desenvolvimento dessa pesquisa de mestrado. Inicialmente senti preocupação, dada a minha pouca experiência com planejamento e elaboração de pesquisa. Esta dificuldade inicial foi suprida por meio das sessões de orientação, estudo sobre os conteúdos trabalhados e elaboração dos trabalhos requeridos pelas disciplinas realizadas no percurso do mestrado profissional. Durante a condução e realização da Revisão Sistemática, as minhas limitações recaíram, principalmente, na disciplina para a leitura de todos os resumos, títulos e palavras chaves, uma vez que foram considerados para o primeiro filtro 10.027 estudos, que foram superadas com o tempo e a experiência de organização da leitura.

Esta pesquisa traz contribuições acadêmicas, por meio das três publicações já aceitas e disponibilizadas: (i) no “15th International Conference on Information Systems and Technology Management” (CONTECSI 2018), a respeito do *survey* desenvolvido; e no (ii) “XXIII Conferência Internacional sobre Informática Educativa” foram dois: um sobre o *framework gamificado* e o segundo sobre a avaliação da primeira versão do produto desenvolvido nesta investigação.



Esta dissertação de mestrado se mostrou abrangente e relevante, ao ponto de originar novos trabalhos a respeito do estudo aqui relatado. Para tanto, destacamos as seguintes sugestões de prosseguimento do estudo descrito, com possíveis evoluções que podem torná-lo mais completo e adequado:

- Evolução da Revisão sistemática: a RSL identificou diversas abordagens *gamificadas* para o ensino e aprendizagem de Algoritmos em diversos cursos de graduação em Tecnologia da Informação, porém, podemos identificar algumas outras abordagens *gamificadas*, seja por meio da expansão para outras fontes de busca, ou pela ampliação do período de pesquisa realizada pela Revisão Sistemática. Logo, uma possível evolução no protocolo pode auxiliar a encontrar outras evidências, as quais possivelmente não foram inclusas nesta pesquisa.
- Avaliação da segunda versão do *framework gamificado*: o uso da segunda versão do *framework gamificado* na disciplina Algoritmos ou equivalentes dentro de uma das suas ofertas regulares, permite a coleta de novas informações sobre um possível aprimoramento do *framework*, o qual gerará o desenvolvimento de uma terceira versão.
- Avaliação do *framework* em outros conteúdos da disciplina Algoritmos: o uso do *framework* para outros conteúdos da disciplina permite visualizar o comportamento das práticas em outros momentos dela, e por meio desse uso, permite a evolução e desenvolvimento de uma nova versão do produto dessa dissertação de mestrado.
- Avaliação do *framework* para toda a disciplina Algoritmos: uma adaptação do *framework* permite utilizá-lo em toda a disciplina e por consequência *gamificar* todo o conteúdo.
- Desenvolvimento de uma ferramenta para auxiliar no desenvolvimento do *framework gamificado*: como possível evolução para o processo de ensino e aprendizagem, utilizando a proposta da dissertação, seria o desenvolvimento de uma ferramenta ao qual simula ao funcionamento da planilha e disponibiliza o plano de ensino como uma central de ajuda sobre o bom andamento da disciplina.

Portanto esta dissertação de mestrado constituiu-se como um desafio para o seu desenvolvimento, dado a sua abordagem abrangente, por uso de vários procedimentos metodológicos de pesquisa tanto qualitativos, quanto quantitativos.

## REFERÊNCIAS BIBLIOGRÁFICAS

APPOLINÁRIO, F. *Dicionário de metodologia científica: um guia para produção do conhecimento científico*. 1.ed. São Paulo: Atlas, 2009.

ASSOCIATION FOR COMPUTING MACHINERY/ INSTITUTE OF ELECTRICAL AND ELECTRONIC ENGINEERS. *Curriculum Guidelines for Undergraduate Degree Programs in Computer Science*. Association for Computing Machinery/ Institute of Electrical and Electronic Engineers. New York, p. 1 - 518. 2013. (ISBN: 978-1-4503-2309-3 / ACM: 999133).

BALDNER, P. R.; DECOURT, F.; NEVES, H. D. R. *Planejamento e gestão estratégica*. Rio de Janeiro: Fundação Getúlio Vargas (FGV), 2012.

BESEMER, S. P.; TREFFINGER, D. J. *Analysys of Creative Products: Review and Synthesis.. The Journal of Creative Behavior*, v. XV, n. 3, p. 158 - 178, 1981.

BOSSE, Y.; GEROSA, M. A. *Reprovações e Trancamentos nas Disciplinas de Introdução à Programação da Universidade de São Paulo: Um Estudo Preliminar.. WEI-Workshop sobre Educação em Computação*, São Paulo, 2015. 1 - 10.

CHOU, Y.-K. *Actionable Gamification: Beyond Points, Badges and Leaderboards*. [S.l.]: [s.n.], 2013.

COSTA, A. C. S.; MARCHIORI, P. Z. *Revista de Ciência da Informação e Documentação. Gamificação, elementos de jogos e estratégia: uma matriz de referência.*, Ribeirão Preto, v. VI, n. 2, p. 44-65, Outubro 2015. ISSN ISSN 2178-2075.

COSTA, C. S. *Uma abordagem baseada em evidências para o gerenciamento de projetos no desenvolvimento distribuído de software*. Dissertação de Mestrado – Programa de Pós-Graduação em Ciência da Computação - Universidade Federal de Pernambuco. Recife. 2010.

CRESWELL, J. W. *Uma estrutura para projeto*. \_\_\_\_\_. *Projeto de pesquisa: métodos qualitativo, quantitativo e misto*. Porto Alegre: Artmed, 2007.

CRUZES, D. S.; DYBÅ, T. Recommended steps for thematic synthesis in software engineering. *In 2011 International Symposium on Empirical Software Engineering and Measurement*, Lund-Suécia, Setembro 2011. 275-284.

DETERDING, S.; DIXON, D.; KHALED, R.; NACKE, L.. From game design elements to gamefulness: defining gamification. *Proceedings of the 15th international academic MindTrek conference: Envisioning future media environments.*, Tampere, Finland, 28 - 30 Setembro 2011. 9-15.

DIAS JÚNIOR, M.; MERCADO, L. A Importância da Estratégia de Ensino por Simulação para a Disciplina de Algoritmos. *5º Congresso Ibero-Americano em Investigação Qualitativa (CIAIQ 2016)*, 08 jul. 2016. 85 - 94.

ELGRABLY, I. S.; OLIVEIRA, S. R. B. Gamification and Evaluation of the Use the Agile Tests in Software Quality Subjects: the Application of Experiments. *13th ENASE - Evaluation of Novel Approaches to Software Engineering*, Madeira, 2018.

FALCÃO, A. P.; LEITE, M. D.; TENÓRIO, M. M. Ferramenta de apoio ao ensino presencial utilizando gamificação e design de jogos. *XXV SBIE*, Dourados, Brasil., 2014.

FARDO, M. L. A Gamificação Aplicada em Ambientes de Aprendizagem. *RENTE*. Revista Novas Tecnologias na Educação, v. 11, p. 1, 2013.

FREITAS, S.; LIMA, T.; CANEDO, E.; COSTA, R. L. Gamificação e avaliação do engajamento dos estudantes em uma disciplina técnica de curso de graduação. *XXVII SBIE*, Uberlândia, Brasil., 2016.

GIRAFFA, M. M.; MORA, M. C. Evasão na disciplina de algoritmo e programação: um estudo a partir dos fatores intervenientes na perspectiva do aluno. *Congressos CLABES*, Quito, 9 - 11 Novembro 2016.

GOMES, A.; AREIAS, C.; HENRIQUES, J.; MENDES, AJ. Aprendizagem de programação de computadores: dificuldades e ferramentas de suporte. *Revista portuguesa de Pedagogia*, Coimbra, p. 161-179, 2008.

GONÇALVES, L.; GIACOMAZZO, G., RODRIGUES, F.; MACAIA, B. Gamificação na Educação: um modelo conceitual de apoio ao planejamento em uma proposta pedagógica. *XXVII SBIE*. , Uberlândia, Brasil., 2016.

HIJON-NEIRA, R. B.; VELÁZQUEZ-ITURBIDE, Á.; PIZARRO-ROMERO, C. Game programming for improving learning experience. In: *Proceedings of the 2014 conference on Innovation & technology in computer science education.*, Uppsala, 23 - 25 Junho 2014. 225-230.

HUNICKE, Robin; LEBLANC, Marc; ZUBEK, Robert. MDA: A formal approach to game design and game research. In: **Proceedings of the AAAI Workshop on Challenges in Game AI**. 2004. p. 1722.

HUOTARI, Kai; HAMARI, Juho. Defining gamification: a service marketing perspective. In: **Proceeding of the 16th international academic MindTrek conference**. ACM, 2012. p. 17-22.

\_\_\_\_\_, Kai; HAMARI, Juho. A definition for gamification: anchoring gamification in the service marketing literature. **Electronic Markets**, v. 27, n. 1, p. 21-31, 2017.

IKLUND, Emil; WAKERIUS, Victor. The gamification process: a framework on gamification. 2016

JABREF. JABREF. *JABREF*, 2018. Disponível em: <<http://www.jabref.org/>>. Acesso em: 22 Setembro 2017.

JESUS, A.; BRITO, G. S. Concepção de ensino-aprendizagem de algoritmos e programação de computadores: a prática docente. *Varia Scientia*, Cascavel, v. IX, n. 16, p. 149-158, 2009.

JUUL, Jesper. The game, the player, the world: Looking for a heart of gameness. **PLURAIIS-Revista Multidisciplinar**, v. 1, n. 2, 2018.

KAPP, K. M. *The Gamification of Learning and Instruction: Game-based Methods and Strategies for Training and Education*. 1º. ed. [S.l.]: Pfeiffer & Company, 2012. ISBN 1118096347 9781118096345.

KITCHENHAM, B.; CHARTERS, S. *Guidelines for performing Systematic Literature Reviews in Software Engineering*. 3º. ed. -2007-01,: EBSE Technical Report, v. ||, 2007.

LANDSELL, Julia; HÄGGLUND, Emelie. Towards a Gamification Framework: Limitations and opportunities when gamifying business processes. 2016.

LANGE, A.; KARAWEJCZYK, T. COACHING NO PROCESSO DE DESENVOLVIMENTO. *DIÁLOGO*, Canoas, p. 39-56, Abril 2014. ISSN 2238-9024.

LEININGER, M. M. *Qualitative research methods in nursing*. Orlando: Grune & Stratton, 1985.

LI, W.; GROSSMAN, T.; FITZMAURICE, G. Proceedings of the 25th annual ACM symposium on User interface software and technology. *GamiCAD: a gamified tutorial system for first time autocad users.*, 2012. 103-112.

LIMA JUNIOR, J. A. T.; VIEIRA, C. E. C.; PAULA VIEIRA, P. Dificuldades no Processo de Aprendizagem de Algoritmos: uma Análise dos Resultados na Disciplina de AL1 do Curso de Sistemas de Informação da FAETERJ–Campus Paracambi. *Cadernos UniFOA*, Volta Redonda, v. X, n. 27, p. 5 - 15 , Abril 2015.

LIMESURVEY. Professional online surveys with LimeSurvey. *LimeSurvey*, 2018. Disponível em: <<https://www.limesurvey.org/>>. Acesso em: 15 Outubro 2018.

LUZ, R. B.; NETO, A. Using Programming Dojos for Test-Driven Development Teaching.. *XXIII Brazilian Symposium on Informatics in Education (SBIE 2012)*. , Rio de Janeiro, 2012. 25-35.

MAFRA, S. N.; TRAVASSOS, G. H. *Estudos Primários e Secundários apoiando a busca por Evidência em Engenharia de Software*. Universidade Federal do Rio de Janeiro. Rio de Janeiro, p. 22. 2006.

MARCON, S. S.; ELSÉN, I. Estudo qualitativo utilizando observação participante - análise de. *Acta Scientiarum. Biological Sciences*, Maringá, n. 22, p. 637 - 647, 2000. ISSN ISSN: 1679-9283.

MARCUSSI, L. D. et al. Pesquisa no ensino de algoritmos e programação nas engenharias: estudos e resultados preliminares. *Simpósio de Engenharia de produção: Perspectivas e soluções para a indústria e mercado de trabalho*, Londrina, 2016.

MAY, T. *Pesquisa Social, Questões, métodos e processos*. Tradução de Carlos Alberto Silveira Netto Soares. 3ª. ed. Porto Alegre: Artmed, 2004.

MCGONIGAL, J. *Reality Is Broken: Why Games Make Us Better and How They Can Change The World*. Nova Iorque: The Penguin Press, 2011.

MEDEIROS, J. S.; FIGUEIREDO, K. Game in Class: Criando Disciplinas Gamificadas. *Workshops do IV CBIE*, Maceió, Brasil., 2015.

MONTEIRO, W. M.; OLIVEIRA, T. M.; MARTINS, D. J. S. Gamificação na Educação: Possibilidades para o ensino de programação. *Revista Tecnologias na Educação*, v. VII, n. 13, 2015.

MOODLE. Moodle. *Moodle*, 2018. Disponível em: <[https://moodle.org/?lang=pt\\_br](https://moodle.org/?lang=pt_br)>. Acesso em: out. 2018.

NASSI-CALÒ, L. Avaliação por pares: modalidades, prós e contras. Scielo em perspectiva. *Scielo em Perspectiva*, 2015. Disponível em: <<http://blog.scielo.org/blog/2015/03/27/avaliaca-o-por-pares-modalidades-pros-e-contras/>>. Acesso em: 10 abr. 2018.

PINSONNEAULT, A.; KRAEMER, K. Survey research methodology in management information systems: an assessment. *Journal of management information systems*, v. 10, n. 2, p. 75-105, 1993.

PROJETO SOFTWARE PROCESS IMPROVEMENT: DEVELOPMENT AND RESEARCH - UFPA. Projeto SPIDER - Tool Suite for Quality. *Projeto SPIDER*, 2009. Disponível em: <<http://www.spider.ufpa.br/>>. Acesso em: 2017.

PROJETO SPIDER. Início. *Projeto SPIDER*, 2009. Disponível em: <<http://www.spider.ufpa.br/>>. Acesso em: 01 Novembro 2018.

QUARESMA, J. A. S., ELIASQUEVICI, M. K., MENEZES, J.; Oliveira, S. R. B. (2018) “Um Estudo sobre a Disciplina Algoritmos ou Equivalente dos Cursos de Graduação quanto ao Ensino, Aprendizado e Conteúdo: Uma Aplicação de Survey”. 15th CONTECSI. São Paulo.

RAABE, A. L. A.; SILVA, J. M. C. Um ambiente para atendimento as dificuldades de aprendizagem de algoritmos. *XIII Workshop de Educação em Computação (WEI'2005)*, São Leopoldo, 2005.

RAMOS, V.; WAZLAWICK, R.; GALIMBERTI, M.; FREITAS, M.; MARIANI, A. C. A Comparação da Realidade Mundial do Ensino de Programação para Iniciantes com a Realidade Nacional: Revisão sistemática da literatura em eventos brasileiros. *Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação-SBIE)*, Maceió, 26 - 30 Outubro 2015. 318 - 327.

REGINALDO, L. Pascalzim - Compilador Pascal. *Pascalzim*, 2016. Disponível em: <<http://pascalzimbr.blogspot.com/p/blog-page.html>>. Acesso em: Outubro 2018.

ROBSON, K.; PLANGGER, K.; KIETZMANN, J. H.; MCCARTHY, I.; PITT, L. Is it all a game? Understanding the principles of gamification. **Business Horizons**, v. 58, n. 4, p. 411-420, 2015.

SANTOS, E. D.; OLIVEIRA, S. R. B. Gamificação como Ferramenta de Suporte de Apoio ao Ensino da Técnica Análise de Pontos por Função em um Turma de Pós-Graduação: Um Estudo de Caso. *15th CONTECSI - International Conference on Information Systems and Technology Management*. , São Paulo, 23 - 25 Maio 2018.

SANTOS, G. Revisão Sistemática, Mini-Curso. *Simpósio Brasileiro de Qualidade de Software-SBQS.*, Belém-PA, 2010.

SCHLEMMER, E. Games e Gamificação: uma alternativa aos modelos de EaD. *Revista Iberoamericana de Educación a Distancia*, 2016. 107-124.

SILVA, C. R. E. Orientação Profissional, mentoring, coaching e counseling: Algumas singularidades e similaridades em práticas. *Revista Brasileira de Orientação Pro*, São Paulo, p. 299-309, Dezembro 2010. ISSN 1679-3390.

SOCIEDADE BRASILEIRA DE COMPUTAÇÃO. Currículo de Referência para cursos de graduação em Bacharelado em Ciência da Computação e Engenharia da Computação, 2005.

\_\_\_\_\_. Referenciais de formação para os cursos de graduação em Computação 2017, 2017.

SPRINT, G.; COOK, D. Enhancing the CS1 student experience with gamification. *In: Integrated STEM Education Conference (ISEC)*, 7 Março 2015. 94-99.

TANUR, J. M. *Advances in methods for large-scale surveys and experiments*. In R. Mcadams, N. J. Smelser, and D. J. Treiman (eds.), *Behavioral and Social Science Research: A National Resource*. Washington: National Academy Press, v. II, 1982.

TIMMERMANN, G. L. K.; GONZÁLEZ, F. J. Mediações que os professores e alunos estabelecem com o conteúdo da disciplina de Algoritmos de cursos superiores: estudo de caso. *Anais dos Workshops do V Congresso Brasileiro de Informática na Educação (CBIE 2016)*, Uberlândia, 24 a 27 Outubro 2016. 1295 - 1305.

TRAVASSOS, G.; BIOLCHINI, J. Revisões sistemáticas aplicadas a engenharia de software. *XXI SBES-Brazilian Symposium on Software Engineering.*, João Pessoa, 15 - 19 Outubro 2007.

UDEMY. Udemy. *Udemy*, 2018. Disponível em: <<https://www.udemy.com>>. Acesso em: 05 dez. 2018.

UNIVERSIDADE FEDERAL DO PARÁ. *PROJETO PEDAGÓGICO DO CURSO DE BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO - UFPA*. UNIVERSIDADE FEDERAL DO PARÁ. Belém, p. 1 - 86. 2010.

UNIVERSIDADE FEDERAL DO PARÁ. EaD UFPA. *Moodle UFPA*, 2018. Disponível em: <<http://www.aedmoodle.ufpa.br/>>.

VIANNA, Y.; TANAKA, S.; VIANNA, M.; MEDINA, B. *Gamification, Inc: como reinventar empresas a partir de jogos*. [S.l.]: MJV Press., 2013. 15 p.10

WERBACH, K.; HUNTER, D. *For the win: How game thinking can revolutionize your business*. Philadelphia: Wharton Digital Press, 2012.

WIKLUND, Emil; WAKERIUS, Victor. *The gamification process: a framework on gamification*. 2016.

## APÊNDICE A – INSTRUMENTO SURVEY DOCENTE

**Título provisório do Estudo:** Desenvolvimento de estratégias de *gamificação* para apoio ao ensino e aprendizagem de algoritmos

**Pesquisador:** José Augusto de Sena Quaresma (Aluno do Programa de Pós-Graduação Criatividade e Inovação em Metodologias do Ensino Superior - PPGCIMES da Universidade Federal do Pará) – [jq.quaresma12@gmail.com](mailto:jq.quaresma12@gmail.com).

**Orientadora:** Profa. Dr.<sup>a</sup> Marianne Kogut Eliasquevici.

**Coorientador:** Prof. Dr. Sandro Ronaldo Bezerra Oliveira.

### TERMO DE CONSENTIMENTO LIVRE E ESCLARECIDO

Você está sendo convidado (a) para participar de uma pesquisa cujo objetivo é desenvolver estratégias de *gamificação* para apoio ao ensino e aprendizagem de algoritmos. Ao final, espera-se contribuir para melhorar os índices de aprovação da disciplina Algoritmos ou equivalente presente nos cursos da área de Tecnologia da Informação (Sistemas de Informação, Licenciatura em Computação, Ciência da Computação, Engenharia da Computação, Engenharia de Software ou outros cursos relacionados). Esta pesquisa é parte integrante de uma dissertação do Programa de Pós-Graduação Criatividade e Inovação em Metodologias de Ensino Superior, conforme indicações a seguir:

**A quem se destina:** Docentes envolvidos na disciplina Algoritmos ou equivalente.

**Objetivo:** Desenvolver estratégias de *gamificação* para apoio ao ensino e aprendizagem de algoritmos.

**Riscos potenciais:** Não há riscos conhecidos neste estudo.

**Confidencialidade:** Ao participar deste estudo, você reconhece e concorda que suas respostas e contribuições serão registradas anonimamente para fins de análise de dados mais confiáveis.

**Disseminação dos resultados:** Os resultados deste estudo aparecerão na dissertação final e poderão, também, fazer parte de publicações de caráter científico.

**Direito de se retirar:** A participação neste estudo é voluntária, você pode decidir não participar a qualquer momento do preenchimento do questionário.

Você concorda com esses termos?

\*Ao preencher e enviar este questionário, seu consentimento livre e esclarecido é implícito e indica que você entende as condições de participação neste estudo, mencionadas acima.

Se você aceitar participar, esse questionário é composto por 20 perguntas e o seu preenchimento não tomará mais de 10 minutos do seu tempo.

\* Required

Consentimento \*

Sim

### Informações Básicas

1. Quantas vezes ministrou a disciplina Algoritmos ou equivalente?

( ) Apenas uma vez

( ) Entre uma e três vezes

( ) Entre quatro e seis vezes



Acima de seis vezes

2. Você ministrou a disciplina Algoritmos ou equivalente nos últimos dois anos?

Sim

Não

### Em relação ao processo de ensino

**3. Qual abordagem didática você utiliza predominantemente ao ministrar a disciplina Algoritmos ou equivalente? \***

Teórica

Prática em sala de aula

Prática laboratorial

Híbrida (Teórica+ Prática)

Outro:

**4. Qual(is) recurso(s) você utiliza para apoiar o ensino de algoritmos? (Você pode marcar mais de uma opção) \***

Lista de exercícios para ser resolvida em sala de aula

Lista de exercícios para ser resolvida em casa

Questão com nível de dificuldade mais elevado para ser resolvida em sala

Desenvolvimento de projetos

Outro:

**5. Qual dentre os recursos apontados acima você considera mais significativo para o ensino de algoritmos? \***

Lista de exercícios para ser resolvida em sala de aula

Lista de exercícios para ser resolvida em casa

Questão com nível de dificuldade mais elevado para ser resolvida em sala

Desenvolvimento de projetos

Outro:

**6. Como você costuma dividir os alunos para a resolução de algoritmos durante a aula? (Você pode marcar mais de uma opção) \***

Cada estudante trabalha individualmente

Em duplas

Em equipe acima de duas pessoas

Outro:

**7. Qual dentre os formatos apontados acima você considera mais significativo para a aprendizagem? \***

Cada estudante trabalha individualmente.

Em duplas

Em equipes acima de duas pessoas

Outro:

**8. Você costuma ter o auxílio de um monitor durante a disciplina Algoritmos ou equivalente? \***

Sim

Não

**9. Durante a Disciplina, você costuma identificar perfis de estudantes? \***

- Sim  
 Não

**9.1 Caso sim, descreva os perfis que já conseguiu identificar separados por ponto e vírgula (Exemplos: líder; competidor; colaborador; introspectivo; participativo; etc.).****10. Durante a sua aula, para a resolução de problemas em sala você: \***

- Apresenta o algoritmo de solução pronta  
 Constrói a solução junto com os alunos  
 Solicita ao monitor da disciplina que explique a solução para os alunos  
 Solicita a um aluno que acertou a atividade que a resolva no quadro  
 Outro:

**11. Durante a Disciplina você faz uso de bonificação? (Exemplos: pontuação extra, prêmio, etc.). \***

- Sim  
 Não

**11.1 Caso sim, de que forma(s)?**

- Competição  
 Resolução de lista de exercícios  
 Resolução de questão com nível de dificuldade mais elevado  
 Resolução de questão no quadro para a turma  
 Desenvolvimento de projeto  
 Outros

**11.2 Caso sim, você percebe motivação entre os estudantes com este tipo de estratégia?**

- Sim  
 Não

**12. Você fornece feedback sobre as atividades desenvolvidas?**

- Sim  
 Não

**13. Qual(is) fator(es) você considera como pré-requisitos para o aprendizado da disciplina Algoritmos ou equivalente? (Você pode marcar mais de uma opção)**

- Conhecimentos de lógica  
 Leitura e Interpretação de texto  
 Conhecimentos matemáticos  
 Outro:

**14. Durante a Disciplina, qual(is) software(s)/linguagem(ns) você utiliza para facilitar o processo de ensino de algoritmos? (Você pode marcar mais de uma opção) \***

- Não utilizo  
 Pascalzim  
 Visualg  
 C, C++  
 Java

( ) Outro:

### Em relação ao conteúdo da Disciplina

Para cada um dos conteúdos abaixo, presentes no desenvolvimento da Disciplina, assinale qual o grau de facilidade que você sentiu tanto ao ensinar quanto os estudantes em assimilar.

#### 15. Ensino pelo professor: \*

	Fácil	Médio	Difícil	Não se aplica
Conceitos de algoritmos e sua utilidade				
Variáveis, constantes e tipos de dados				
Operandos/operadores e expressões				
Atribuições				
Estruturas sequenciais (operações de entrada e saída)				
Estruturas de decisão				
Estrutura de Repetição				
Estruturas de dados homogêneas (matrizes e vetores)				
Subalgoritmos (função, procedimento)				
Recursividade				
Implementação de algoritmos em linguagens de programação				

#### 16. Assimilação pela turma: \*

	Fácil	Médio	Difícil	Não se aplica
Conceitos de algoritmos e sua utilidade				
Variáveis, constantes e tipos de dados				
Operandos/operadores e expressões				
Atribuições				
Estruturas sequenciais (operações de entrada e saída)				
Estruturas de decisão				
Estrutura de repetição				
Estruturas de dados homogêneas (matrizes e vetores)				
Subalgoritmos (função, procedimento)				
Recursividade				
Implementação de algoritmos em linguagens de programação				

17. Utilize as linhas a seguir, caso sinta necessidade de registrar outras considerações que julgar pertinentes:

---

Obrigado por contribuir com nossa pesquisa!

## APÊNDICE B – INSTRUMENTO SURVEY DISCENTE

**Título provisório do Estudo:** Desenvolvimento de estratégias de *gamificação* para apoio ao ensino e aprendizagem de algoritmos

**Pesquisador:** José Augusto de Sena Quaresma (Aluno do Programa de Pós-Graduação Criatividade e Inovação em Metodologias do Ensino Superior - PPGCIMES da Universidade Federal do Pará) – [jq.quaresma12@gmail.com](mailto:jq.quaresma12@gmail.com).

**Orientadora:** Profa. Dr.<sup>a</sup> Marianne Kogut Eliasquevici.

**Coorientação:** Prof. Dr. Sandro Ronaldo Bezerra Oliveira.

### TERMO DE CONSENTIMENTO LIVRE E ESCLARECIDO

Você está sendo convidado(a) para participar de uma pesquisa cujo objetivo é desenvolver estratégias de *gamificação* para apoio ao ensino e aprendizagem de algoritmos. Ao final, espera-se contribuir para melhorar os índices de aprovação da disciplina Algoritmos ou equivalente presente nos cursos da área de Tecnologia da Informação. Esta pesquisa é parte integrante de uma dissertação do Programa de Pós-Graduação Criatividade e Inovação em Metodologias de Ensino Superior, conforme indicações a seguir:

**A quem se destina:** discentes envolvidos na disciplina Algoritmos ou equivalente.

**Objetivo:** desenvolver estratégias de *gamificação* para apoio ao ensino e aprendizagem de algoritmos.

**Riscos potenciais:** não há riscos conhecidos neste estudo.

**Confidencialidade:** ao participar deste estudo, você reconhece e concorda que suas respostas e contribuições serão registradas anonimamente para fins de análise de dados mais confiáveis.

**Disseminação dos resultados:** os resultados deste estudo aparecerão na dissertação final e poderão, também, fazer parte de publicações de caráter científico.

**Direito de se retirar:** a participação neste estudo é voluntária, você pode decidir não participar a qualquer momento do preenchimento do questionário.

#### Você concorda com esses termos?

\*Ao preencher e enviar este questionário, seu consentimento livre e esclarecido é implícito e indica que você entende as condições de participação neste estudo, mencionadas acima.

Se você aceitar participar, esse questionário é composto por 20 perguntas e o seu preenchimento não tomará mais de 10 minutos do seu tempo.

\* Required

Consentimento \*

Sim

#### Informações Básicas

##### 1. Quantas vezes cursou a disciplina Algoritmos ou equivalente? \*

( ) Apenas uma vez

( ) Duas ou três vezes

( ) Quatro ou cinco vezes

( ) Acima de seis vezes

**2. Você cursou a disciplina Algoritmos ou equivalente nos últimos dois anos? \***

- Sim
- Não

**Em relação ao processo de ensino**

**3. Qual abordagem didática você prefere que o professor utilize ao ministrar a disciplina Algoritmos ou equivalente? \***

- Teórica
- Prática em sala de aula
- Prática laboratorial
- Híbrida (Teórica+ Prática)
- Outro:

**4. Qual dentre os recursos apontados a seguir você considera mais significativo para a sua aprendizagem de algoritmos? \***

- Lista de exercícios para ser resolvida em sala de aula
- Lista de exercícios para ser resolvida em casa
- Questão com nível de dificuldade mais elevado para ser resolvida em sala
- Desenvolvimento de projetos
- Outro:

**5. Qual dentre os formatos apontados a seguir você considera mais significativo para a sua aprendizagem de algoritmos? \***

- Cada estudante trabalha individualmente.
- Trabalhar em dupla
- Em equipes acima de duas pessoas
- Outro:

**6. Você procura o auxílio do monitor durante a disciplina Algoritmos ou equivalente? \***

- Sim
- Não
- Não se aplica

**7. Diante de sua participação nas aulas de algoritmos, dos perfis a seguir, como você se considera? Marque até dois principais \***

- Líder
- Competitivo
- Colaborador
- Introspectivo
- Participativo
- Estimulador
- Influenciador
- Nenhum
- Outro: \_\_\_\_\_

**8. Durante a aula, para a resolução de problemas em sala você prefere que o professor: \***

- Apresente o algoritmo de solução pronta
- Construa a solução junto com os alunos
- Solicite ao monitor da disciplina que explique a solução para os alunos

- Solicite a um aluno que acertou a atividade que a resolva no quadro
- Outro:

**9. Durante a Disciplina o professor faz/fez uso de bonificação? (Exemplos: pontuação extra, prêmio, etc.). \***

- Sim
- Não

**9.1 Caso sim, de que forma(s)?**

- Competição
- Resolução de lista de exercícios
- Resolução de questão com nível de dificuldade mais elevado
- Resolução de questão no quadro para a turma
- Desenvolvimento de projeto
- Outros

**9.2 Caso sim, você se sente motivado ao estudo com este tipo de estratégia?**

- Sim
- Não

**10. Você recebe/recebeu feedback sobre as atividades desenvolvidas na disciplina Algoritmos ou equivalente?**

- Sim
- Não

**11. Qual(is) fator(es) você considera como pré-requisitos para o aprendizado na disciplina Algoritmos ou equivalente?**

- Conhecimentos de lógica
- Leitura e Interpretação de texto
- Conhecimentos matemáticos
- Outro:

**12. Durante a Disciplina, qual(is) software(s)/linguagem o professor utiliza/utilizou para facilitar sua aprendizagem na disciplina Algoritmos ou equivalente? \***

- Não utilizo
- Pascalzim
- Visualg
- C/ C++
- Java
- Outro:

**13. O uso das linguagens ou ambiente de desenvolvimento facilita/facilitou o seu processo de aprendizagem? \***

- Sim
- Não

### Em relação ao conteúdo da Disciplina

**14. Para cada um dos conteúdos abaixo, presentes no desenvolvimento da disciplina Algoritmos ou equivalente, assinale qual o grau de facilidade que você sente/sentiu tanto ao aprender.**

	Fácil	Médio	Difícil	Não se aplica
Conceitos de algoritmos e sua utilidade				
Variáveis, constantes e tipos de dados				
Operandos/operadores e expressões				
Atribuições				
Estruturas sequenciais (operações de entrada e saída)				
Estruturas de decisão				
Estrutura de repetição				
Estruturas de dados homogêneas (matrizes e vetores)				
Subalgoritmos (função, procedimento)				
Recursividade				
Implementação de algoritmos em linguagens de programação				

### Jogos

**15. O que mais lhe motiva ao jogar um game? \***

- As recompensas adquiridas (medalhas, troféus)
- O desafio das missões
- A competição com outros jogadores
- A cooperação com outros jogadores
- O ranking do jogo
- Customizar um personagem ou *avatar*
- Compartilhar minhas premiações
- Receber um feedback sobre minhas ações no jogo
- Coletar recursos
- Gerenciar recursos
- Não jogo
- Outro:

**16. Durante a sua graduação, em alguma das disciplinas cursadas, o professor fez uso da *gamificação* como estratégia de ensino? \***

- Sim
- Não

**16.1 Caso sim, o uso da *gamificação* contribuiu para melhorar o seu processo de aprendizado do conteúdo?**

- Sim
- Não
- Em parte

**Outras informações**

**17. Utilize as linhas a seguir, caso sinta necessidade de registrar outras considerações que julgar pertinentes:**

---

Obrigado por contribuir com nossa pesquisa!



## APÊNDICE C – INSTRUMENTO DA OBSERVAÇÃO PARTICIPANTE

Data: \_\_\_\_ / \_\_\_\_ / \_\_\_\_

Hora de início:

Hora de término:

Quantidade de estudantes iniciais (até 20 minutos de aula):

Quantidade de estudantes finais:

Presentes na aula: ( ) professor ( ) monitor

Responsável pela condução da aula: ( ) professor ( ) monitor

Conteúdo ministrado:

### PROCEDIMENTOS METODOLÓGICOS

1) Os objetivos da aula foram apresentados para a turma? ( ) sim ( ) não

2) Formato/Metodologia utilizada durante a aula:

( ) Avaliação

( ) Resolução de exercícios

( ) Expositiva

( ) Dialogada

( ) \_\_\_\_\_

3) Estratégias pedagógicas utilizadas:

( ) competição

( ) resolução de exercícios em grupo

( ) resolução de exercícios individual

( ) uso de outros recursos tecnológicos

( ) \_\_\_\_\_

4) A aula apresenta-se de forma estimulante e desafiadora?

5) As atividades propostas para a turma são, em sua maioria, de natureza individual ou coletiva?

6) Há a retomada de conhecimentos trabalhados em aulas anteriores como um ponto de partida para facilitar novas aprendizagens?

7) Como o professor prepara os estudantes para a próxima aula? Cria expectativas?

8) Como está organizado o tempo da aula? Foram reservados períodos de duração suficiente para os estudantes fazerem anotações, exporem as dúvidas, debaterem e resolverem problemas?

Outras observações:

### RELAÇÃO PROFESSOR X ESTUDANTE

O professor aguarda os estudantes terminarem os exercícios para apresentar os resultados?

Há uma relação harmônica entre estudantes e professor?

### INTERAÇÃO ENTRE OS ESTUDANTES E O CONTEÚDO

O tempo foi suficiente para o conteúdo proposto na aula?

Principais dúvidas feitas pelos estudantes:

( ) interpretação do comando

( ) base matemática

( ) raciocínio lógico

( ) capacidade de abstração

( ) sintaxe

( ) semântica

( ) \_\_\_\_\_

### **RELAÇÃO MONITOR X ESTUDANTE**

Os estudantes chamam muito pelo apoio do monitor?

Ele foi peça chave para a realização das atividades?

### **RELAÇÃO ENTRE OS ESTUDANTES**

Existe clima de cooperação entre os estudantes?

Os trabalhos em grupo/dupla são facilmente desenvolvidos?

### **OUTRAS OBSERVAÇÕES:**

## APÊNDICE D – PLANO DE ENSINO-APRENDIZAGEM GAMIFICADO

Universidade Federal do Pará  
Instituto de Ciências Exatas e Naturais  
Faculdade de Computação



### Plano de Ensino-Aprendizagem *Gamificado*

#### 1 Identificação

Professora: Marianne Kogut Eliasquevici  
Disciplina: Algoritmos  
Curso: Bacharelado em Ciência da Computação  
Semestre: 1º semestre – 2018/2  
Ano: 2018  
Carga horária: 68 horas (4 horas semanais – 2 encontros na semana)  
Elaboração: José Augusto de Sena Quaresma  
Marianne Kogut Eliasquevici  
Sandro Ronaldo Bezerra Oliveira

#### 2 Conteúdo

Unidade 4 – Estrutura de dados homogêneas

- 4.1 – Vetor
- 4.2 – Matriz

#### 3 Objetivo da Disciplina

Desenvolver no aluno a capacidade de compreender e elaborar algoritmos com qualidade e eficiência para a solução de problemas computacionais, independente de uma linguagem de programação.

##### 3.1 Objetivo de Ensino

Capacitar o discente da Disciplina após a conclusão dos estudos sobre o conteúdo, em desenvolver e analisar Estrutura de dados homogênea.

##### 3.2 Objetivos de Aprendizagem

1. Identificar estrutura de dados homogêneos em algoritmos.

2. Implementar estrutura de dados homogêneos em algoritmos e na linguagem de programação adotada para a disciplina.
3. Realizar teste para analisar qual o resultado do processamento da estrutura de dados homogêneos.

### 3.3 Habilidades e competências (específicas)

Especificar, modelar, implantar e avaliar estrutura de dados homogêneas em algoritmos.

## 4 Planejamento

Para o conteúdo de que trata esse plano de ensino foi adotado um *framework gamificado*, adaptado de Elgrably e Oliveira (2018) e Santos e Oliveira (2018). As informações referentes ao processo de *gamificação*, bem como as atividades, os recursos metodológicos e a forma como serão avaliados os alunos está presente na seção 5 desse plano de ensino.

## 5 Processo de *gamificação*

A *gamificação* é definida segundo Deterding *et al.* (2011) como o uso de elementos de jogos em ambientes que não são de jogos. A partir dessa premissa pretendemos *gamificar* as atividades em sala de aula ao se trabalhar com algumas dinâmicas, mecânicas e elementos de games para ensinar o conteúdo de estrutura de dados homogêneos.

Portanto, a avaliação e planejamento do processo de ensino e aprendizagem no conteúdo de estrutura de dados homogêneos serão baseados na *gamificação* aqui descrita. De maneira inicial informamos que os alunos serão avaliados de duas maneiras: uma em relação ao conteúdo, ao qual ele receberá pontos e um *avatar* a cada aula. E, a outra em relação a comportamento, ao qual o aluno receberá estrelas a cada aula.

### 5.1 Personagens

Nessa seção será descrito os personagens participantes dessa *gamificação*, a qual está descrito no Quadro 1.

Quadro 1 – Descrição de papéis

Identificação	Função	Nome
<b>Master</b>	Professor/ Pesquisador	<<Responsável>>
<b>Mentora</b>	Professora da Disciplina	<<Nome da professora>>
<b>Coach</b>	Acompanhamento na resolução de problemas	<<Monitor>>
<b>Juiz/ Scoreplay</b>	Colaborador para desenvolvimento do score da turma	<<Nome do responsável pela planilha>>
<b>Players</b>	Jogadores/ estudantes	Alunos da disciplina Algoritmos

## 5.2 Descrição das atividades no jogo com o cronograma

Para efeito dessa *gamificação* identificamos como os dias de aula, atividades práticas desenvolvidas e a prova como abordagens. A informação referente a data, máxima pontuação e estrela que podem ser alcançadas, bem como uma rápida descrição de cada abordagem está apresentada no Quadro 2.

O local onde acontecerão as aulas e atividades do *framework gamificado* será o Laboratório de Computação 1 da Faculdade de Computação, localizado no terreio do Instituto de Ciências Exatas e Naturais da Universidade Federal do Pará.

Os recursos que estarão disponíveis para essa *gamificação* será a estrutura do laboratório, o uso de um notebook pessoal do pesquisador e pinceis atômicos.

Quadro 2 – Cronograma das fases e a pontuação atribuída (Continua...)

Data	Abordagem	Atividade	Pontuação	Quantidade de Estrelas
<b>06/06/2018</b>	Conversa inicial sobre a <i>gamificação</i>	Não terá atividades	0	0
<b>11/06/2018</b>	Aula teórica e prática – Array Unidimensional (Seção 5.2.1)	Lista exercício sala – 4 questões	100	3
<b>13/06/2018</b>	Aula teórica e prática – Array Bidimensional	Lista exercício sala – 4 questões	100	3
<b>18/06/2018</b>	Dojo Randori (Seção 5.2.3)	Atividade em dupla rotativa avaliando a narrativa (copiloto) e a execução (piloto)	200	3
<b>20/06/2018</b>	Dojo Randori	Atividade em dupla rotativa avaliando a narrativa (copiloto) e a execução (piloto)		3
<b>25/06/2018</b>	Atividade Lab (Seção 5.2.4)	Atividade em dupla fixa, resolvendo 4 questões. Lista de exercício de casa (4 questões)	300 (200 – Atividade em sala + 100 – Lista de exercício de casa)	0

Quadro 2 – Cronograma das fases e a pontuação atribuída (Conclusão)

Data	Abordagem	Atividade	Pontuação	Quantidade de Estrelas
02/07/2018 <sup>4</sup>	Prova Teórica (Seção 5.2.5)	Teste individual (4 questões)	300	0
04/07/2018	Feedback (Seção 5.2.6)	Bonificação sobre a análise do ensino de <i>gamificação</i>		3

### 5.2.1 Aula Teórica e Prática

A aula teórica e prática é uma abordagem híbrida de ensino em que o professor mescla conceitos teóricos com práticas de programação. Para essa abordagem foram reservados dois dias de aula, no primeiro dia será focado em vetor e no segundo dia os estudos serão voltados para matriz.

Como se trata de um processo *gamificado* o aluno ganhará pontuação e estrelas no dia, sendo que o processo de ganho de estrela está descrito na seção 5.5 desse documento. O aluno pontuará ao resolver as questões do exercício proposto. Para essa pontuação, utilizaremos a seguinte regra: O aluno ganha 25 pontos por cada questão que resolver; A lista de exercício proposta por dia possui quatro questões (Apêndices A e B), então diariamente o aluno terá um total de 100 pontos; À medida que o aluno receber a pontuação ele receberá um *avatar* diário, segundo a Quadro 3.

Quadro 3 – Avatar diário aula teórica e prática

Avatar	Faixa de pontuação
<b>Analista</b>	90 – 100
<b>Programador</b>	70 – 89
<b>Aprendiz</b>	50 – 69
<b>Amador</b>	0 – 49

#### 5.2.1.1 Procedimentos metodológicos

As duas aulas teóricas e práticas terão a mesma estrutura lógica de apresentação do conteúdo, contendo: uma Metáfora sobre o assunto; apresentação do conceito e a forma como é implementado em pascal; um exercício exemplo para o professor fazer junto com os alunos; uma lista de exercício com quatro questões; Por fim, serão retomados os pontos abordados na aula.

---

<sup>4</sup> Caso tenha Jogo do Brasil na copa do mundo as 11 horas desse dia a prova será no dia 04/07/2018 e por consequência o dia da aula de feedback será alterada para o dia 09/07/2018.

### 5.2.2 Dojo *Randori*

A abordagem Dojo *Randori* é uma prática de ensino de programação em ambiente seguro. O aluno aprenderá por tentativa e erro, com o auxílio de seus pares. Tal prática é adaptada do *framework* apresentado por Luz e Neto (2012). A abordagem possui os seguintes papéis: (i) piloto, responsável pelo desenvolvimento do código; (ii) copiloto, com a função de narrar e auxiliar no planejamento e codificação do programa; (iii) plateia, interage entre si com a intenção de cooperar com o piloto e copiloto durante o desenvolvimento. No contexto da sala de aula será uma das nossas dinâmicas. Contará com dois dias de desenvolvimento por conta do tamanho da turma e cada um dos alunos passará pelos três papéis já descritos.

Na atividade em questão os alunos serão avaliados por comportamento segundo os quesitos definidos na seção 5.5 desse documento. No que tange o conteúdo o aluno receberá pontuação segundo os quesitos estabelecidos na Quadro 4. O quesito pontuações pelas questões desenvolvidas pela turma será uma nota atribuída pelo professor em relação a codificação das funções realizadas pela turma, podendo ir de 0 até 50 pontos, ou seja, nesse atributo toda a turma receberá nota igual. As questões para a atividade do Dojo *Randori* estão presentes no Apêndice D.

Quadro 4 – Pontos adquiridos no Dojo *Randori*

Quesito	Pontos adquiridos
Uma questão concluída	20
Utilizou o fluxo Programar-Testar	30
Narrou como copiloto	100
Pontuações pelas questões desenvolvidas pela turma	Até 50

O *avatar* atribuído para a atividade do Dojo *Randori* seguirá a regra estabelecida na Quadro 5.

Quadro 5 – Avatar Dojo *Randori*

Avatar	Faixa de pontuação
<b>Analista</b>	180 – 200
<b>Programador</b>	150 – 179
<b>Aprendiz</b>	120 – 149
<b>Amador</b>	0 – 119

### 5.2.2.1 Procedimentos Metodológicos

As duas aulas de Dojo seguirão o mesmo procedimento metodológico, o qual será: o professor explicará sobre as regras do Dojo *Randori* e apresentará a lista de exercício proposta para que os alunos programem em par; Após esse momento será solicitado que dois voluntários assumam os papéis de piloto e copiloto; Eles iniciam a codificação, após 5 minutos da atividade é permitido a plateia emitir sugestão na programação; Depois de dois minutos o piloto retorna a plateia, o copiloto assume a posição de piloto e um voluntário da plateia assume a posição de copiloto. O ciclo se repete até que todos os alunos tenham participado da atividade. Ao final, o professor recolherá os códigos gerados para análise.

### 5.2.3 Atividade LAB

A atividade Laboratório de Programação, abreviada como LAB, é uma abordagem proposta no *framework* de Elgrably e Oliveira (2018) e Santos e Oliveira (2018). Na proposta em questão, os alunos serão divididos em duplas cada uma com acesso a um computador, o qual deve estar sem acesso à internet, e terão uma lista com quatro desafios para responder.

Os alunos ganharam nesse dia pontos por cada desafio, sendo dividido da seguinte maneira: (i) 20 pontos para o desafio concluído; (ii) 30 pontos atribuída pelo professor em relação a nota da codificação; (iii) A pontuação máxima por desafio será de 50 pontos. Como cada dupla receberá uma lista contendo quatro questões (Apêndice E) a pontuação máxima para o LAB será de 200 pontos. Além disso, para esse dia foi reservado a entrega para o exercício feito em casa (Apêndice C), o qual pontuará segundo as regras descritas: (i) 25 pontos por questão resolvida de maneira correta; (ii) um total de 100 pontos para as quatro questões resolvidas de maneira correta. Sendo assim, a pontuação total do dia será a soma dos pontos adquiridos tanto na atividade LAB quanto para a lista de exercício feito em casa, então para esse dia teremos o *avatar* diário segundo condições estabelecidas na Quadro 6. Para esse dia não os alunos não receberão recurso estrela como descrito, na seção 5.5 desse documento.

Quadro 6 – *Avatar* da Atividade LAB em conjunto com a lista de exercício feita em casa

<i>Avatar</i>	Faixa de pontuação
<b>Analista</b>	270 – 300
<b>Programador</b>	230 – 269
<b>Aprendiz</b>	190 – 229
<b>Amador</b>	0 – 189



### 5.2.3.1 Procedimentos metodológicos

Na atividade LAB será solicitado aos alunos para que se dividam em duplas e após a divisão serão organizados em um computador para cada dupla. Cada dupla receberá uma folha com as quatro questões referentes. Após a conclusão o professor recolherá os códigos fontes das duplas para analisar. O tempo para a execução dessa atividade será de 1 hora e 30 minutos.

### 5.2.4 Teste Teórico

O teste teórico verifica o conhecimento do aluno individualmente em relação ao conteúdo estudado na disciplina. Nesse dia os alunos não receberão recurso estrela, segundo descrição da seção 5.5 desse documento. Porém, os alunos pontuarão segundo as seguintes regras para cada questão: (i) 25 pontos pela questão concluída; (ii) somados a de 0 a 50 pontos atribuídos pela análise do código feito pelo professor. O teste teórico conterà quatro questões (Apêndice F) cada uma com o valor máximo de 75 pontos, portando o nível máximo de pontuação do aluno será de 300 pontos. E, como estamos em um processo *gamificado* o aluno ganhará um *avatar* pelo seu desempenho na prova segundo regras definidas na Quadro 7.

Quadro 7 – Avatar teste teórico

Avatar	Faixa de pontuação
<b>Analista</b>	270 – 300
<b>Programador</b>	230 – 269
<b>Aprendiz</b>	190 – 229
<b>Amador</b>	0 – 189

### 5.2.4.1 Procedimentos metodológicos

Para o teste teórico os alunos farão a atividade individualmente e terão o tempo de 1 hora e 30 minutos para resolver quatro questões sobre o conteúdo estrutura de dados homogêneos.

### 5.2.5 Aula de Feedback

A aula de *feedback* é uma abordagem prevista no *framework* de Elgrably e Oliveira (2018) e Santos e Oliveira (2018), em que os alunos serão questionados pelo professor sobre o processo de ensino e aprendizagem via *gamificação*. As perguntas desse dia se encontram no Apêndice G. Para esse dia os alunos ganharam apenas recurso estrela, segundo regras descritas na seção 5.5 desse documento.

### 5.2.5.1 Procedimentos Metodológicos

Na aula de feedback o professor instigará os alunos por meio de uma apresentação em Power point sobre as diversas atividades que aconteceram no *framework gamificado*, com o intuito de melhorar para uma próxima turma.

### 5.3 Mecânicas de bônus

A bonificação é um recurso de o aluno ganhará por seu comportamento em sala de aula, após a contabilização de todos os bônus e penalidades que o aluno sofreu, esse recurso será transformado em estrela, segundo as regras descritas na seção 5.5. As regras que definem a bonificação do aluno estão descritas no Quadro 8.

Quadro 8 – Distribuição das bonificações

Nome da dimensão	Explicação da dimensão	Bônus por dimensão	Máxima Bonificação por dimensão	Justificativa
<b>Presença</b>	O aluno estar presente pontualmente no horário da aula.	10 Bônus	10 Bônus	A presença é considerada importante para compreender o conteúdo
<b>Participação</b>	0 – Caso o aluno não tenha participado; 2 bônus a cada participação até no máximo 10 bônus de participação (Ex: Feito algum comentário para o professor, respondido a uma pergunta aberta do professor a turma).	2 Bônus	10 Bônus	A participação em sala de aula indica que o aluno está atento ao conteúdo
<b>Sugestão</b>	0 – Caso o aluno não tenha sugerido; 2 bônus a cada sugestão com até no máximo 10 bônus de sugestão.	2 Bônus	10 Bônus	A sugestão mostra que o aluno está buscando o conceito além da sala de aula.
<b>Pergunta</b>	0 – Caso o aluno não tenha perguntado; 2 – bônus a cada pergunta com até no máximo 10 bônus de pergunta	2 Bônus	10 Bônus	A pergunta caracteriza o interesse pelo assunto abordado.
<b>Colaboração</b>	0 – Caso o aluno não tenha ajudado algum colega; 2 bônus a cada ajuda ao colega na resolução das atividades proposta com até no máximo 10 bônus de colaboração.	2 Bônus	10 Bônus	A colaboração nutre no aluno o sentimento de auxílio ao colega.

#### 5.4 Mecânicas de Penalidades

A penalidade é uma infração cometida pelo aluno a qual será penalizada com a perda de bonificação. As regras que definem a bonificação do aluno estão descritas no Quadro 9.

Quadro 9 – Distribuição das penalidades

Nome da dimensão	Explicação da dimensão	Perda de Bônus por dimensão	Justificativa
<b>Falta</b>	-10 pontos caso falte a disciplina	-10	A falta caracteriza a perda do conteúdo passado no dia.
<b>Atraso em 10 minutos</b>	O aluno perderá 2 bônus a cada atraso. Podendo ter a penalidade aplicada a no máximo 4 atrasos totalizando perda de 8 bônus.	-2	O atraso em 10 minutos identifica o desinteresse do aluno pela disciplina
<b>Uso do celular</b>	A cada uso do celular em horário de aula o aluno perderá 2 bônus	-2	O uso do celular indica a não atenção a atividade ou dinâmica desenvolvida em sala
<b>Uso do computador</b>	Utilizar o computador em sites e ferramentas não previstas nos procedimentos metodológicos	-2	O uso do computador indica a não atenção a atividade ou dinâmica desenvolvida em sala
<b>Penalidade por atrapalhar a aula</b>	Quando o aluno interromper a aula com brincadeiras ou não permitir que outro aluno assista a aula perderá 2 bônus.	-2	Atrapalhar a aula indica a não atenção a atividade também pode atrapalhar o desempenho dos outros alunos

#### 5.5 Conversão de bônus para estrela

Após os alunos bonificarem e sofrerem as penalidades, diariamente receberão o recurso estrela exceto nos casos já previstos. As regras para concessão do recurso estrela estão descritas no Quadro 10.

Quadro 10 – Conversão de bônus para recurso estrela

Recurso	Descrição	Intervalo de bônus	Quantidade
<b>Estrela</b>	O recurso estrela é atribuído ao jogador conforme a sua bonificação	34 – 50	3
		17 – 34	2
		5 – 17	1

## 5.6 Conversão da pontuação para nota na disciplina

A pontuação final do jogo ocorrerá mediante a transformação do recurso estrela em pontos, o qual acontece segundo Quadro 11. E, a transformação do processo *gamificado* em nota será o número de pontos dividido por 100.

Quadro 11 – Conversão das estrelas para pontuação

Faixa de pontuação	Nota extra
15 – 10	<b>100</b>
9 – 5	<b>50</b>
4 – 0	<b>0</b>

### Referência Bibliográfica Básica

CARVALHO, S. E.R. **Introdução à Programação com PASCAL**. 4.ed. Rio de Janeiro: Campus, 1986.

DETERDING, S.; DIXON, D.; KHALED, R.; NACKE, L.. From game design elements to gamefulness: defining gamification. In: **Proceedings of the 15th international academic MindTrek conference: Envisioning future media environments**. ACM, 2011. p. 9-15.

EIGRABLY, I. S.; OLIVEIRA, S. R. B. "Gamification and Evaluation of the Use the Agile Tests in Software Quality Subjects: the Application of Experiments". **13th ENASE - Evaluation of Novel Approaches to Software Engineering, Madeira, Portugal**, 2018.

LUZ, R. B.; NETO, A. Using Programming Dojos for Test-Driven Development Teaching. **XXIII Brazilian Symposium on Informatics in Education (SBIE 2012)**. Rio de Janeiro. p.25-35, 26, 2012.

EVARISTO, Jaime. Programando com Pascal. **2ª Edição. Ed. Book Express**, 2004.

SANTOS, E. D.; OLIVEIRA, S. R. B. Gamificação como Ferramenta de Suporte de Apoio ao Ensino da Técnica Análise de Pontos por Função em um Turma de Pós-Graduação: Um Estudo de Caso. **15th CONTECSI – International Conference on Information Systems and Technology Management**. São Paulo, Brasil.2018.

### Referência Bibliográfica Complementar

MEDINA, Marco; FERTING, Cristina. **Algoritmos e programação: teoria e prática**. São Paulo: Novatec Editora, 2006.

## Apêndice A -Lista de Exercício 01

Conteúdo: Estrutura de dados homogêneos (vetor)

Observação:

Lista disponível para os alunos no dia 11/06/2018

A lista precisa ser feita em aula para os alunos ganharem pontuação

- 1) Desenvolva um algoritmo que leia do usuário um vetor de 10 posições e calcule e exiba o número armazenado mais o índice da posição que ele está armazenado.
- 2) Desenvolva um algoritmo em pascal que leia um vetor de 7 posições e calcule e exiba primeiro os números armazenados com índice ímpar e depois em índice par.
- 3) Desenvolva um algoritmo em pascal que leia dois vetores de 5 posições, calcule um vetor com a soma dos elementos em índices iguais e apresente ao usuário.
- 4) Desenvolva um algoritmo em pascal que leia os 12 meses do ano e à medida que o usuário digitar um número informe o mês correspondente, quando o usuário digitar 0 finalize o programa, caso digite outro número informe mês inválido.

## Apêndice B -Lista de Exercício 02

Conteúdo: Estrutura de dados homogêneos (matriz)

Observação:

Lista disponível para os alunos no dia 13/06/2018

A lista precisa ser feita em aula para os alunos ganharem pontuação

- 1) Desenvolva um algoritmo que leia do usuário um conjunto de números de uma matriz quadrada de ordem 3 e a presente ao usuário os números armazenados. Regra de constituição de matriz quadrada número de colunas = número de linhas.
- 2) Desenvolva um algoritmo que calcule e exiba o traço e a diagonal da matriz informada na primeira questão. A regra do cálculo do traço da matriz é a soma da dos elementos da diagonal principal dela. A diagonal principal é formada pelos elementos cujo índice de colunas é igual ao índice de linhas.
- 3) Desenvolva um algoritmo para armazenar a agenda do usuário, portanto o programa deve receber os seguintes dados pessoais dos contatos do usuário: nome, endereço, CEP, Bairro e Telefone. O tipo de dado que essa matriz deve receber é do tipo String. Considere que o usuário poderá ter apenas 7 contatos na agenda e o usuário informará na primeira interação todos os contatos.
- 4) Desenvolva um algoritmo em pascal que monte uma nota fiscal, para isso utilize a estrutura de matriz, na primeira coluna deve armazenar a quantidade de produto solicitado, na segunda linha o valor unitário e na terceira linha o valor total por produto. Considere que a nota fiscal pode ter 10 linhas e o usuário informará na primeira interação os 10 produtos.

## Apêndice C – Lista de exercício para casa

Conteúdo: Estrutura de dados homogêneos

Observação:

Lista disponível para os alunos no dia 13/06/2018

A lista será feita em casa para os alunos ganharem pontuação

1) Desenvolva um programa em Pascal que receba do usuário um vetor de 10 posições, multiplique os valores em índice ímpar por 10 e divida os valores em índice par por 10. Depois apresente ao usuário o vetor resultante.

2) Desenvolva um programa em Pascal que leia do usuário dois vetores de 5 posições do tipo inteiro, multiplique o primeiro pelo inverso do segundo e apresente um terceiro vetor com o resultado para o usuário.

3) Elabore um programa em Pascal que leia dois vetores com 5 elementos do tipo inteiro cada, intercale os elementos dos dois vetores, formando um novo vetor com 10 elementos.

EX:

6	1	8
---	---	---

3	4	7
---	---	---

6	3	1	4	8	7
---	---	---	---	---	---

4) Criar um programa em Pascal que, dado um vetor de 5 elementos do tipo inteiro, crie um outro que seja o inverso do primeiro e apresente os dois vetores para o usuário.

(Ex.: vet1 → 1 2 3 4 5; vet2 → 5 4 3 2 1)

5) Escrever um programa em Pascal que lê um vetor N com 20 elementos do tipo inteiro e o escreve. Em seguida, troque o 1º elemento com o último, o 2º com o penúltimo até o 10º com o 11º e escreva o vetor N assim modificado.

6) Desenvolva um programa em Pascal que implemente uma matriz 3x3, receba do usuário números reais, coloque os números na matriz e depois apresente para o usuário apenas os elementos da diagonal principal.

7) Elabore um programa em Pascal que crie uma matriz 7X8 em que cada elemento é a soma dos índices de sua posição dentro da matriz. Ao final, apresente para o usuário a matriz gerada em formato matricial.

8) Criar um programa em Pascal que possa armazenar as alturas de dez atletas de cinco delegações que participarão dos jogos de verão. Imprimir a maior altura de cada delegação.

9) Elabore um algoritmo em pascal que receba números inteiros informados pelo usuário armazene em uma matriz 3x3 identifique se existem números repetidos, quantas vezes se repetem e apresente ao usuário.

10) Faça um programa que leia uma matriz 10X10, calcule e escreva a soma dos elementos que estão na área marcada com '\*'.

*									
*	*								
*	*	*							
*	*	*	*						
*	*	*	*	*					
*	*	*	*	*	*				
*	*	*	*	*	*	*			
*	*	*	*	*	*	*	*		
*	*	*	*	*	*	*	*	*	
*	*	*	*	*	*	*	*	*	*

### Apêndice D – Lista de exercícios para o Dojo *Randori*

Conteúdo: Estrutura de dados homogêneos

Observação:

Lista disponível para os alunos no dia 18/06/2018

#### Atividade 01

Desenvolva um programa em Pascal que contenha uma lista de supermercado de uma pessoa, usando um vetor de string com 10 posições. As funcionalidades que o programa precisa apresentar ao usuário, por meio de um menu com 06 opções, são:

- 1) Incluir todos os itens da lista.
- 2) Imprimir toda a lista.
- 3) Apresentar um item específico da lista, por meio do índice.
- 4) Incluir um item da lista, a partir de um índice específico.
- 5) Excluir um item da lista, a partir de um índice específico.
- 6) “Zerar” a lista (todos os elementos vão conter branco dentro).

Ao digitar o número 0, o programa finaliza a execução.

#### Atividade 02

Desenvolva um programa em Pascal que contenha um boletim com as notas dos alunos, usando uma matriz 10x4, sendo o índice de linha corresponde ao número do aluno e o de coluna a nota bimestral. As funcionalidades que o programa precisa apresentar ao usuário, por meio de um menu com 07 opções, são:

- 1) Inserir todas as notas informando o número do aluno.
- 2) Apresentar o Boletim em formato de matriz.
- 3) Inserir uma nota específica informando o número do aluno e o bimestre.
- 4) Apresentar todas as notas de um aluno específico a partir do seu número.
- 5) Calcular e apresentar a média de um aluno específico da turma a partir do seu número.
- 6) Apresentar o aluno que teve a maior nota na turma.
- 7) Apresente o aluno que teve a menor nota na turma.

Ao digitar o número 0, o programa finaliza a execução.

Lista de exercício para o segundo dia Dojo *Randori* (20/06/2018)

### Atividade 01

Uma empresa foi contratada para criar um programa para uma loja. O programa tem que ser capaz de: incluir e excluir as vendas de cada semana dos meses do ano; alterar as vendas de uma determinada semana de um mês específico; exibir o total vendido em cada mês do ano; exibir o total vendido no ano e, ainda, em que mês(es) a loja alcançou o mínimo desejado para vendas (R\$ 5.000,00). Para fins de simplificação, considere que cada mês do ano possui somente 04 semanas.

### Apêndice E – Atividade Laboratório de Programação

Conteúdo: Estrutura de dados homogêneos

Observação:

Atividade disponível para os alunos no dia 27/06/2018

A atividade será realizada em dupla

Não será liberada consulta a internet, aos materiais disponibilizados e nem a dupla ao lado

1) Desenvolva um algoritmo em Pascal que contenha a lista de preços para um leilão *online*. O programa recebe todos os lances e apresenta para o usuário o lance com preço maior, menor e a média dos lances que foram realizados. O vetor receberá 15 lances.

2) Elabore um programa em Pascal que armazene os 40 primeiros números primos em um vetor de inteiros. Depois apresente lado a lado os números de índice ímpar e os números de índice par. Ex:

Ímpar		Par
2		3
5		7

3) Elabore um algoritmo em pascal que leia do usuário números inteiros e os armazene em uma matriz de ordem 4, depois calcule o traço da diagonal principal. Limpe a tela do *prompt* e apresente ao usuário os números que estão acima da diagonal principal e que são maiores que o traço, em formato horizontal separado por “|”.

4) Sua empresa foi contratada por uma rede de cinemas para gerar um programa para reservar as cadeiras para uma sessão. Como bom desenvolvedor verificou que as filas começam com a numeração 1 e vão até a 12, também observou que em cada fila da sessão tem 12 poltronas, logo, uma matriz de ordem 12.

Desenvolva um algoritmo em pascal que realize a “reserva” de lugares (informando a fileira e a cadeira), com o nome do cliente, caso a cadeira já tenha sido reservada exibir uma mensagem para o cliente, exibir em formato matricial as cadeiras disponíveis, permitir uma opção para o cliente verificar se a sessão já está cheia, e ao digitar a palavra “fim” o programa finaliza.



## Apêndice F – Teste teórico

Conteúdo: Estrutura de dados homogêneos

Observação:

Atividade disponível para os alunos no dia 02/07/2018

A atividade será realizada individualmente.

Não será liberada consulta a internet, aos materiais disponibilizados e nem o colega ao lado.

O aluno realizará a atividade no papel com caneta, sem o auxílio de computador.

- 1) Faça um programa em Pascal que leia um vetor de 5 posições de números inteiros, multiplique todos os elementos pelo maior valor do vetor e apresente o novo vetor.
- 2) Faça um programa em Pascal que leia um vetor de seis elementos numéricos inteiros, calcule e mostre:
  - a) A quantidade de números pares
  - b) Quais os números pares
  - c) A quantidade de números ímpares
  - d) Quais os números ímpares
- 3) Desenvolva um algoritmo em Pascal que leia os valores de uma matriz de ordem 3 de inteiros, calcule e apresente a média dos números digitados, assim como a matriz.
- 4) Desenvolva um algoritmo em Pascal que leia os valores de uma matriz de ordem 4 e apresente todos os elementos da diagonal secundária, bem como a soma deles.

## Apêndice G – Atividade *feedback*

Data da atividade: 04/07/2018

As questões que serão feitas a turma para análise do *framework gamificado* foram divididas em categorias e seguirão a ordem abaixo:

### ***Feedback* sobre as metodologias de ensino escolhidas:**

- Foram positivas para o aprendizado? Por quê?
- Seria necessário mais tempo de aula teórica e prática? Por quê?
- Necessita de mais tempo na aula de Dojo *Randori*? Por quê?
- Qual a melhor forma do Dojo na sua opinião Tipo 01 ou Tipo 02? Por quê?
- Necessitaria de mais tempo na aula prática LAB (Laboratório de Programação)? Por quê?
- Você acha que seria necessário algum pré-requisito para um melhor aproveitamento dos alunos nas metodologias escolhidas para o ensino? Quais?

***Feedback sobre a aula teórica e prática:***

- A aula expositiva e prática foi satisfatória para o aprendizado de Estrutura de dados homogêneos?
- Qual a dificuldade vocês acharam para as listas de exercício?
- Vocês consideram válida a experiência do fluxo de ensino:
- Metáfora -> conceito -> A implementação na linguagem Pascal -> Um exercício exemplo
- O fluxo de ensino foi satisfatório ou gostariam de outra forma de ensino?

***Feedback sobre a aula de Dojo Randori:***

- A prática do Dojo foi considerada satisfatória?
- Existia a necessidade de mais tempo para a explicação sobre as regras e o funcionamento do Dojo?
- Qual a dificuldade que vocês acharam na atividade?
- Vocês acham que o tempo destinado para atividade foi apropriado? Ou queriam mais tempo dessa atividade?
- Vocês têm alguma sugestão para melhoria da prática do Dojo?

***Feedback sobre a prática de LAB:***

- A prática do LAB foi considerada satisfatória?
- O tempo que foi disponibilizado para a prática foi suficiente?
- Qual a dificuldade encontrada na atividade?
- Fazer a técnica utilizando pareamento foi positivo?
- Vocês têm alguma sugestão para melhoria da prática do LAB?

***Feedback sobre a lista de exercício feita em casa:***

- Você achou interessante ter uma lista de exercício para ser desenvolvida em casa?
- Qual a dificuldade vocês acham para a lista de exercício?

**Feedback sobre a importância de Estrutura de dados homogêneos:**

- Houve interesse em se aprofundar em estrutura de dados homogêneos?
- A utilização de Dojo e LAB, facilitou o entendimento da importância da prática da implementação e análise de Estrutura de dados homogêneos?
- Houve interesse dos alunos em utilizar a estrutura de dados homogêneos em projetos pessoais, no estágio ou no emprego?

**APÊNDICE E – FORMULÁRIO DE REVISÃO POR PARES****Histórico de Revisões**

<b>Data</b>	<b>Versão</b>	<b>Descrição</b>	<b>Autor</b>
02/03/2018	1.0	Elaboração do Formulário de Revisão por Pares	José Augusto
09/03/2018	1.1	Revisão e ajustes	Sandro Bezerra e José Augusto

# Revisão por Pares

## 1. Apresentação

Esta revisão por pares tem como foco a avaliação do *framework gamificado* para a disciplina Algoritmos ou equivalentes, o qual contém os seguintes itens:

- O plano de ensino *gamificado*;
- A planilha de *gamificação*.

## 2. Objetivo da revisão por pares

O objetivo primordial da revisão é avaliar o *framework* de *gamificação* da disciplina Algoritmos ou equivalente, bem como sua estrutura e a comunicação entre os itens que o compõem. Portanto, devem ser revisados o plano de ensino *gamificado*, as atividades, a planilha com os elementos de *gamificação*, a estrutura adotada para apresentação das aulas, as listas de exercício e o teste teórico.

## 3. Instruções para a execução da revisão por pares

- a) Preencha a sua Identificação e o seu Perfil como especialista na disciplina Algoritmos ou em *Gamificação* (Seções 4 e 5).
- b) Leia o *framework* de *gamificação* da disciplina Algoritmos ou equivalentes (em anexo), analisando se os dados informados são válidos, com relação a clareza, ortografia e as competências e habilidades requeridas para o conteúdo da disciplina. Avalie se cada um dos elementos descritos atende ao que se propõem em relação ao ensino pelo professor e ao aprendizado do aluno.
- c) Durante a leitura, registre pontos para as quais você deseja fazer um comentário, o qual pode ser em relação ao conteúdo presente no plano de ensino e nas atividades, ou então nos elementos de jogos presentes na planilha.
- d) Utilize a Tabela constante no final deste documento (Seção 7) para registrar seus comentários:
  - A coluna **ID** representa um campo auto incremental de considerações provenientes das Revisões;
  - A coluna **Categoria** representa o tipo de consideração da Revisão. Estes tipos serão mais bem explicados na Seção 7 deste documento;
  - A coluna **Item** representa o nome da técnica constante no catálogo apresentado e que possui alguma consideração proveniente da Revisão;
  - A coluna **Comentário com a Justificativa** representa a consideração do Revisor quanto à criação do catálogo;
  - A coluna **Sugestão** representa a proposta do Revisor para contornar o problema, caso aplicável.
- e) Após concluir a análise do documento em anexo preencha a avaliação objetiva e quantitativa da proposta (Seção 6).
- f) Ao concluir a revisão, por favor, envie seu documento de revisão para: [jq.quaresma12@gmail.com](mailto:jq.quaresma12@gmail.com).

#### 4. Dados de identificação do revisor

Nome do revisor:

Data da revisão:

#### 5. Perfil do revisor do *framework gamificado*

##### 5.1. Para revisores sobre *gamificação*

a) Qual seu nível de conhecimento em *Gamificação*?

Alto

Médio

Baixo

Nenhum

b) Você ministrou alguma disciplina utilizando técnicas de *Gamificação* nos últimos três anos (2015 – 2018)?

Sim.

Não

c) Caso sim para a questão anterior. Quais as técnicas de *Gamificação* adotadas?

---

---

d) Caso sim a questão anterior. Quais elementos de *Gamificação* adotados?

---

---

##### 5.2. Para revisores sobre *Algoritmos*

a) Qual seu nível de conhecimento em Algoritmos?

Alto

Médio

Baixo

Nenhum

c) Já ministrou a disciplina Algoritmos ou equivalentes em Instituições de Ensino Superior?

Sim. Quantas vezes:

Não

d) Você ministrou a disciplina Algoritmos ou equivalentes nos últimos três anos (2015 – 2018)?

Sim.  Não

e) Você já ministrou a disciplina Algoritmos ou equivalentes utilizando técnicas de *gamificação*?

Sim. Quantas vezes:  Não

f) Caso sim a questão anterior. Quais elementos?

---

---

## 6. Apresentação da proposta

a) Como você considera a proposta do *framework* da disciplina Algoritmos ou equivalente (descrição, atividades propostas, planilha, elementos *gamificados*)?

Completa  Incompleta  
 Inconsistente  Não sei

### Observações:

b) Como você considera a proposta do plano de ensino *gamificado*?

Completa  Incompleta  
 Inconsistente  Não sei

### Observações:

c) Como você considera as atividades proposta para a disciplina?

Completa  Incompleta  
 Inconsistente  Não sei

### Observações:





- **E (Editorial)**, indicando que foi encontrado um erro de português ou que o texto pode ser melhorado;
- **Q (Questionamento)**, indicando que houve dúvidas quanto ao conteúdo das considerações;
- **G (Geral)**, indicando que o comentário é geral em relação às considerações.

**Tabela 01. Exemplo para preenchimento**

ID	Categoria	Item	Comentário com a Justificativa	Sugestão
1	G	<i>Framework</i>	Acredito que a proposta está com poucos elementos <i>gamificados</i>	Aumentar a quantidade de elementos <i>gamificados</i> dentro da planilha e explicar no plano de ensino

**Tabela 02. Preenchimento do revisor**

ID	Categoria	Item	Comentário com a Justificativa	Sugestão
1				
2				
3				
4				
5				
6				
7				
8				
9				

## APÊNDICE F – PROTOCOLO DA REVISÃO SISTEMÁTICA DA LITERATURA

### HISTÓRICO DO DOCUMENTO

<b>Data</b>	<b>Versão</b>	<b>Descrição</b>	<b>Autor</b>
21/08/2017	0.1	Início da concepção do protocolo da Revisão Sistemática	José Augusto Quaresma e Sandro Bezerra
21/08/2017	0.2	Construção do Protocolo	José Augusto, Marinne Kogut Eliasquevici e Júlio Menezes
28/08/2017	0.3	Inserção do Contexto	José Augusto
06/09/2017	0.4	Inserção tabelas e formulários, ajuste nas strings de busca	José Augusto
10/09/2017	0.5	Revisão	Marianne Kogut Eliasquevici
12/12/2018	1.0	Ajuste para a versão final do protocolo	José Augusto

## 1 CONTEXTO

O programa de Pós-Graduação Criatividade e Inovação em Metodologias de Ensino Superior (PPGCIMES), da Universidade Federal do Pará (UFPA), foi aprovado em dezembro de 2016 na 169ª reunião do Conselho Técnico-Científico da Educação Superior (CTC-ES). Tem por área de concentração “Metodologias de Ensino-Aprendizagem” e encontra-se subdividido em duas linhas de pesquisa: Inovações Metodológicas no Ensino Superior (INOVAMES) e Criatividade e Inovação em Processos e Produtos Educacionais (CIPPE).

A linha de pesquisa CIPPE, segundo o edital de seleção PPGCIMES 01/2017 do programa (SITE PPGCIMES, 2017, p.2), é orientada à

[...] conceber e desenvolver processos e produtos criativos para o ensino-aprendizagem, configurados a partir de demandas das diferentes áreas do conhecimento. Para tanto, os recursos comunicacionais, educacionais e os das Tecnologias da Informação e Comunicação serão explorados na geração de soluções inovadoras e factíveis. Os processos e produtos desenvolvidos serão acompanhados de pesquisas qualitativas e/ou quantitativas orientadas às testagens e validações das soluções.

No contexto do ensino superior têm-se os cursos na área de computação, os quais segundo o documento “Referências de Formação para os Cursos de Graduação em Computação” (SBC, 2017) podem ser divididos em: bacharelado em ciência da computação, bacharelado em engenharia da computação, bacharelado em engenharia de software, licenciatura em computação, bacharelado em sistemas de informação e os cursos superiores de tecnologia. Esses cursos têm um Currículo de base desenvolvido pela Sociedade Brasileira de Computação (SBC) em 2005.

O curriculum base dos cursos de computação divide as disciplinas em dois núcleos específicos, que são: os fundamentos da computação e tecnologia de computação. E quatro núcleos de outras áreas do conhecimento, que são: matemática, ciências básicas, eletrônica, e contexto social e profissional.

O núcleo de fundamentos de computação, segundo o curriculum base proposto pela SBC (2005), compreende o conjunto de matérias que envolvem a parte científica e as técnicas fundamentais à formação sólida dos egressos do curso de computação. Sendo assim, compreende as disciplinas de: análise de algoritmos; algoritmos e estrutura de dados; arquitetura e organização de computadores; circuitos digitais; fundamentos de sistemas; linguagens de programação; linguagens formais autômatos e computabilidade; organização de arquivos e dados; sistemas operacionais; E, teoria dos gráficos.

Por ser introdutória, de acordo com Matos e Oshiro (2017, p.12), “se o aluno não tiver uma boa base nessa disciplina, ele pode ter dificuldades em disciplinas futuras de programação, ou pode ficar desmotivado para continuar no curso”. Este fato demonstra a importância desta disciplina para cursos em que ela é ministrada.

No intuito de minimizar as dificuldades de aprendizagem encontradas pelos discentes ao longo da disciplina Algoritmos, o projeto de pesquisa em tela, ligado à linha CIPPE-PPGCIMES, tem o propósito de gerar um produto de software para apoiar o processo de ensino-aprendizagem de algoritmos via *gamificação*. Para tal, foram elencadas quatro fontes de dados que, após analisadas, irão gerar a lista de requisitos do produto de software: (i) uma revisão sistemática da literatura com base no protocolo detalhado adiante; (ii) dois *surveys* (um para discentes e outro para docentes); e (iii) a observação de uma oferta da disciplina Algoritmos no curso de Ciência da Computação da Universidade Federal do Pará.

O presente protocolo, como parte essencial da Revisão Sistemática da Literatura, e uma adaptação dos protocolos realizados nas pesquisas de Souza *et al.* (2017) e Malcher (2015), mesmo que em contextos diferenciados. Cabe ressaltar, que para efeito do protocolo em questão, sempre que for referenciado o termo algoritmos, também será introdução à programação.

## 2 OBJETIVO

A Revisão Sistemática da Literatura (RSL), que dá origem ao presente protocolo, tem por objetivo identificar as abordagens de apoio ao processo de ensino-aprendizagem de algoritmos que façam uso da *gamificação*, no contexto do ensino superior, para os cursos de graduação da área de TI. Por abordagens entende-se: metodologias, métodos, técnicas e ferramentais. Desta maneira, têm-se a seguinte estrutura, proposta por Santos (2010):

- **Analisar:** relatos de experiência e publicações científicas por meio de um estudo baseado em Revisão Sistemática.
- **Com o propósito de:** identificar abordagens *gamificadas* para apoiar o processo de ensino e aprendizagem de algoritmos.
- **Com relação à:** definição e uso de metodologias, métodos, técnicas, ferramentais e demais instrumentos empregados como apoio ao processo de ensino-aprendizagem de algoritmos em cursos de graduação na área de tecnologia da informação (TI).
- **Do ponto de vista de:** dos pesquisadores.

- **No contexto:** acadêmico, instituições de ensino superior públicas ou privadas.

## REFERÊNCIAS BIBLIOGRÁFICAS

BEECHAN, M. et. al. Motivation in Software Engineering: A systematic literature review. **Information and Software Technology**, v. 50, n.1, p. 860-878, 2007.

COSTA, C. S. **Uma abordagem baseada em evidências para o gerenciamento de projetos no desenvolvimento distribuído de software**. Dissertação de Mestrado – Programa de Pós-Graduação em Ciência da Computação - Universidade Federal de Pernambuco, Recife, 2010.

DETERDING, S.; DIXON, D.; KHALED, R.; NACKE, L. From game design elements to gamefulness: defining gamification. In: **Proceedings of the 15th international academic MindTrek conference: Envisioning future media environments**. ACM, 2011. p. 9-15.

GOMES, A.; AREIAS, C.; HENRIQUES, J.; MENDES, A. J.. Aprendizagem de programação de computadores: dificuldades e ferramentas de suporte. **Revista Portuguesa de Pedagogia**, n.42-2, p. 161-179, jul. 2008. Disponível em: <<http://impactum-journals.uc.pt/rppedagogia/article/view/1242>>. Acesso em: 05 set. 2017.

PPGCIMES. **Áreas de Concentração e Linhas de Pesquisa**. Disponível em: <<http://ppgcimes.propesp.ufpa.br/index.php/br/programa/areas-de-concentracao-e-linhas-de-pesquisa>>. Acesso em 22 de novembro de 2017.

KITCHENHAM, B.; CHARTERS, S.. **Guidelines for performing Systematic Literature Reviews in Software Engineering**. Vol 2.3 EBSE Technical Report, EBSE-2007-01, 2007.

MCGONIGAL, Jane. **Reality Is Broken: Why Games Make Us Better and How They Can Change The World**. Nova Iorque: The Penguin Press, 2011.

SANTOS, G. Revisão Sistemática, Mini-Curso. **Simpósio Brasileiro de Qualidade de Software–SBQS. Belém–PA**, 2010.

SHELDON, Lee. **The multiplayer classroom: Designing coursework as a game**. Cengage Learning, 2011.

## 3 FORMULAÇÃO DA PERGUNTA

Dentro do escopo da RSL, as questões de pesquisa são a base para a construção do protocolo. Portanto, no decorrer dessa pesquisa tem-se por objetivo investigar as propostas da literatura no que tange à seguinte indagação:

**Quais as principais abordagens podem ser adotadas em cursos de nível superior nas áreas de tecnologia da informação para apoio ao processo de ensino e aprendizagem de algoritmos?**

Para fins dessa revisão, entende-se por abordagem: métodos, metodologia, técnicas, ferramentas. Já quando se trata de cursos de nível superior na área de tecnologia da informação, delimita-se, mas não se restringe, aos cursos de Engenharia da Computação, Ciência da Computação, Sistemas de Informação, Tecnologia da Informação, Engenharia de Software, Licenciatura em Computação. A partir do delimitado, foi determinada a seguinte questão de pesquisa, a qual servirá como guia desta revisão sistemática:

**(Q1) Quais as principais abordagens de apoio ao processo de ensino e aprendizagem gamificado da disciplina algoritmos no contexto de cursos de nível superior de tecnologia da informação?**

### 3.1 ESTRUTURA DA QUESTÃO PRINCIPAL

Durante a construção da estrutura da questão principal foi utilizada a recomendação de Kitchenham e Charters (2007), que a organiza como: *Population, Intervention, Context, Outcomes, Comparison* (PICOC). Para essa pesquisa, foram empregados apenas os itens população, intervenção e resultados, o que a torna uma revisão QUASI sistemática da literatura, de acordo com Santos (2010).

#### 3.1.1 Para Q1

Têm-se por objetivo, por meio dessa questão, identificar as abordagens de ensino empregadas na disciplina algoritmos e afins, em especial as que utilizam técnicas de *gamificação* (Intervenção), aplicáveis no ensino superior (Contexto), utilizadas em sala de aula por professores (população).

- **População (P):** trabalhos publicados em conferências e periódicos indexados pelo sistema Qualis-CAPES, com foco em cursos de tecnologia da informação.
- **Intervenção (I):** abordagens para apoiar o processo de ensino e aprendizagem de algoritmos, em especial as que empregam técnicas de *gamificação*.
- **Resultados (O):** metodologias, métodos, técnicas, ferramentais e demais instrumentos empregados como apoio ao processo de ensino e aprendizagem de algoritmos.

### 3.2 QUESTÕES SECUNDÁRIAS

Com base na questão principal, foi estabelecido um conjunto de questões secundárias, as quais devem ser respondidas durante a fase de extração de informações. Tais questões têm por objetivo esclarecer detalhes importantes que esta revisão procura identificar:

QSO10. Qual a abordagem mais utilizada para *gamificar* o processo de ensino e aprendizagem da disciplina algoritmos ou equivalente?

QSO11. Quais as principais dificuldades encontradas ao se *gamificar* o ensino da disciplina Algoritmos ou equivalentes?

QSO12. Quais as principais dificuldades encontradas no aprendizado *gamificado* da disciplina Algoritmos ou equivalentes?

QSO13. Como se realiza o processo *gamificado* de ensino e aprendizado da disciplina de Algoritmos ou equivalente?

QSO14. Existem relatos de sucesso o processo *gamificado* de ensino e aprendizagem da disciplina Algoritmos ou equivalente?

QSO15. Existem relatos de insucesso o processo *gamificado* de ensino e aprendizagem da disciplina Algoritmos ou equivalente?

QSO16. Existem relatos de avaliação do processo *gamificado* de ensino e aprendizagem da disciplina Algoritmos ou equivalente?

QSO17. Existem relatos de validação do processo *gamificado* de ensino e aprendizagem da disciplina Algoritmos ou equivalente?

QSO18. Como foi feita a validação do processo *gamificado* de ensino e aprendizagem da disciplina Algoritmos ou equivalente?

## 4 ESCOPO DA PESQUISA

Com a finalidade de tornar a pesquisa exequível, foram definidos critérios de seleção das fontes e algumas restrições associadas.

### 4.1 CRITÉRIOS DE SELEÇÃO DAS FONTES

Para a seleção das fontes de pesquisa, foram definidos os seguintes critérios, adaptados de Souza *et al.* (2017):

- Disponibilidade para consultas web.
- Disponibilidade para busca de artigos por meio do domínio da UFPA.
- Disponibilidade de artigos na íntegra por meio do domínio da UFPA ou a partir da utilização da *engine* de busca Google e/ou Google Scholar.
- Disponibilidade de artigos em inglês ou português.
- Presença de mecanismo de busca que faça uso de palavras-chave.
- Relevância da fonte.
- Boa relação entre estudos retornados e estudos selecionáveis.

Caso os artigos retornados por meio da busca automática não estejam disponíveis na íntegra, serão utilizadas as máquinas de pesquisa dos *sites* do Google (<http://www.google.com.br>) e Google Scholar (<http://scholar.google.com.br>) com o objetivo de adquirir os artigos completos.

#### 4.2 RESTRIÇÕES

Devido ao caráter da pesquisa e os critérios de seleção, também foram adotados os critérios de restrições a seguir, adaptados de Souza *et al.* (2017):

- A pesquisa não pode incorrer em ônus financeiro aos pesquisadores. Portanto, apenas serão selecionadas as fontes que possibilitam consultas de forma gratuita pelo domínio da UFPA.
- Serão apenas considerados os estudos obtidos por meio das fontes selecionadas e em conformidade com os critérios de inclusão e exclusão.
- Serão apenas considerados estudos nas línguas inglês e português. A primeira, por ser a língua de maior difusão no mundo, considera dos estudos de maior relevância e abrangência mundial. Já a segunda, por ser a língua oficial onde o estudo está sendo realizado (Brasil), para entender como está o panorama nacional em relação a pesquisas desse cunho.

A pesquisa será restrita aos resultados publicados entre 01 de janeiro de 2006 até dezembro de 2016, contemplando, desta forma, um período de 10 anos. Pois, o termo *gamificação* é definido em 2008 e referendado na literatura no estudo de Deterding *et al.* 2011.



Porém, para a maior completude do estudo se retroage mais 2 anos para completar o ciclo já informado.

## 5 MÉTODOS DE BUSCA NA FONTE

O presente protocolo de revisão sistemática tem definido para o seu escopo apenas o método de busca nas fontes de formato automático.

### 5.1 MÉTODO DE BUSCA AUTOMÁTICA

A busca automática será realizada por meio das máquinas de busca (buscadores automáticos, como exemplo Google e Google Scholar), disponíveis nas bases de dados selecionadas. Por meio da inserção das *strings* de busca específicas nessas máquinas são retornados os artigos, aos quais tiverem presentes as palavras-chave no seu título, *abstract* ou resumo. Após essa fase, cada artigo será validado a partir dos critérios de seleção e restrição para serem inclusos ou excluídos do estudo. O Quadro 01 resume as bases de dados selecionadas para esta RSL.

Quadro 01 – Fontes Automáticas

Engenho de Busca	Link
<b>IEEEExplore</b>	<a href="http://ieeexplore.ieee.org">http://ieeexplore.ieee.org</a>
<b>ScienceDirect</b>	<a href="http://www.sciencedirect.com">http://www.sciencedirect.com</a>
<b>El Compendex</b>	<a href="http://www.engineeringvillage.com">http://www.engineeringvillage.com</a>
<b>ElsevierScopus</b>	<a href="http://www.scopus.com/home.url">http://www.scopus.com/home.url</a>
<b>ACM Digital Library</b>	<a href="http://dl.acm.org">http://dl.acm.org</a>

### 5.2 PALAVRAS-CHAVE E SINÔNIMOS

A formulação das palavras-chave e sinônimos é realizada para facilitar a concepção da *string* de busca. Para a sua identificação foi utilizada por base a questão principal da pesquisa, definida na seção 4 do protocolo, respeitando a estrutura PICOC, proposta por Kitchenham (2007), sendo que, apenas os itens População, Intervenção e Resultados foram considerados. Em consonância com as restrições da pesquisa, os idiomas definidos para as palavras-chave e sinônimos é o português e o inglês. Portanto, a partir da questão da pesquisa, têm-se a seguinte listagem de palavras identificadas:

- **POPULAÇÃO**

Inglês: *learning programming, Computer Engineering, Computer Science, Information Systems, Information Technology, Software Engineering, Computation Degree, academic, teaching, learning, under graduation.*

Português: aprendizado de programação, Engenharia da Computação, Ciência da Computação, Sistemas de Informação, Tecnologia da Informação, Engenharia de Software, Licenciatura em Computação, acadêmico, ensino, aprendizado, graduação.

- **INTERVENÇÃO**

Inglês: *process of teaching and learning in algorithms, introduction to programming.*

Português: processo de ensino e aprendizagem em algoritmos, introdução a programação, *gamificação*.

- **RESULTADOS**

Inglês: *model, framework, method, technique, methodology, approach, knowledge, activity, tool, teaching tools, procedure, system, program, environment, task, software.*

Português: Modelo, *framework*, método, técnica, metodologia, abordagem, conhecimento, atividade, ferramenta, ferramentas de ensino, procedimento, sistema, programa, ambiente, tarefa, software.

### 5.3 STRING DE BUSCA

Na composição da *string* de busca, será utilizada a definição das palavras-chave e sinônimos, seção 6.2. Conforme observado, a identificação das mesmas é realizada dentro de três itens, sendo que as palavras dentro dos itens são sinônimas, portanto, separadas pelo operador <OR>, o qual segundo a lógica de programação define se pode entrar na pesquisa apenas um, ou os dois. E, no caso das palavras entre itens foi utilizado o operador <AND>, que de acordo com a lógica de programação define que os dois itens devem entrar na pesquisa. Portanto para a estrutura dessa pesquisa temos a seguinte regra de filtragem de artigos {(População) AND (Intervenção) AND (Resultados)}. Logo, de acordo com a estrutura PIC (ou PIO no caso desta pesquisa) conforme Santos (2010).

Na presente revisão, conforme já explicitado na seção 4, ao formular a questão principal, o item Comparação, da estrutura proposta por Kitchenham (2007), não se aplica ao contexto dessa revisão. Complementarmente, as palavras-chave no item referente à Intervenção são relevantes ao contexto do processo de apoio ao ensino e aprendizado de algoritmos ou introdução a programação com *gamificação*, portanto são suas palavras-chave. Já, no caso do item População, é importante o âmbito acadêmico restrito ao nível superior, com ênfase às graduações de TI. Em relação ao item Resultados, foi levado em consideração os formatos de abordagens metodológicas para a pesquisa.

Para a questão principal da pesquisa será considerado duas *strings* de busca (uma em inglês e a outra em português).

### 5.3.1 Para Q1

- (“aprendizado de programação” OR “Engenharia da Computação” OR “Ciência da Computação” OR “Sistemas de Informação” OR “Tecnologia da Informação” OR “Engenharia de Software” OR “Licenciatura em Computação” OR “acadêmico” OR “ensino” OR “aprendizado” OR “graduação”) AND (“processo de ensino e aprendizagem em algoritmos” OR “introdução a programação”) AND (“Modelo” OR “*framework*” OR “método” OR “técnica” OR “metodologia” OR “abordagem” OR “conhecimento” OR “atividade” OR “ferramenta” OR “ferramentas de ensino” OR “procedimento” OR “sistema” OR “programa” OR “ambiente” OR “tarefa” OR “software”).
- (*learning programming*” OR “*Computer Engineering*” OR “*Computer Science*” OR “*Information Systems*” OR “*Information Technology*” OR “*Software Engineering*” OR “*Computation Degree*” OR “*academic*” OR “*teaching*” OR “*learning*” OR “*undergraduation*”) AND (“*process of teaching and learning in algorithms*” OR “*introduction to programming*”) AND (“*model*” OR “*framework*” OR “*method*” OR “*technique*” OR “*methodology*” OR “*approach*” OR “*knowledge*” OR “*activity*” OR “*tool*” OR “*teaching tools*” OR “*procedure*” OR “*system*” OR “*program*” OR “*environment*” OR “*task*” OR “*software*”).

Algumas adaptações são necessárias de acordo com as necessidades específicas de cada fonte de pesquisa. A seguir são apresentadas as tabelas de constituição e *strings* de busca de cada fonte.

#### **Para IEEE**

- ((undergraduation OR education OR learn OR teach) AND (Programming OR Algorithm OR codification OR code) AND (Gam\* OR Funware OR Ludification OR edutainment))

#### **Para Science Direct**

- title-abstr-key ((undergraduation OR education OR learn OR teach) AND (Programming OR Algorithm OR codification OR code) AND (Gam\* OR Funware OR Ludification OR edutainment))

#### **Para El Compendex (Engineering Village)**

- (((undergraduation OR education OR learn OR teach) AND (Programming OR Algorithm OR codification OR code) AND (Gam\* OR Funware OR Ludification OR edutainment)) WN KY) AND ({English} WN LA))

#### **Para Scopus**

- ((TITLE-ABS-KEY ((undergraduation OR education OR learn OR teach) AND (Programming OR Algorithm OR codification OR code) AND (Gam\* OR Funware OR Ludification OR edutainment))) AND (PUBYEAR AFT 2005) AND (PUBYEAR BEF 2017))

#### **Para ACM**

- recordAbstract((undergraduation OR education OR learn OR teach) AND (Programming OR Algorithm OR codification OR code) AND (Gam\* OR Funware OR Ludification OR edutainment))

## 6 TIPOS DE ARTIGO

Para o escopo dessa revisão sistemática, serão consideradas pesquisas do tipo experimental, teórica, relato de experiência, *surveys* e revisões sistemáticas da literatura na forma de artigos completos de periódicos ou conferências.

## 7 IDIOMA DOS ARTIGOS

No âmbito dessa RSL, serão considerados os artigos escritos em português e inglês. Conforme já explicitado os motivos no item 5.2 que fala sobre as restrições.

## 8 CRITÉRIOS DE INCLUSÃO E EXCLUSÃO

Os critérios de inclusão e exclusão aqui presentes são adaptados de Souza *et al.* (2017). A seguir estão os critérios de exclusão a serem utilizados:

- CE.1)** Publicações que não tenham foco em cursos superior de TI, não restrito à Engenharia da Computação, Ciência da Computação, Sistemas de Informação, Tecnologia da Informação, Engenharia de Software e Licenciatura em Computação.
- CE.2)** Publicações do tipo relatórios de workshops, pôster, apresentações, *keynotes speaker*, livros, teses e dissertações.
- CE.3)** Publicações que não estejam acessíveis na íntegra na Web.
- CE.4)** Artigos publicados fora do período definido para a busca.
- CE.5)** Duplicidade de publicações (indexados) terão apenas sua primeira ocorrência considerada.
- CE.6)** Artigos que claramente não atendam as questões de pesquisa.
- CE.7)** Artigos não apresentados em uma das linguagens aceitas (Inglês e Português).

Como critério de inclusão, tem-se:

- CI.1)** Publicações que relatem metodologias educacionais no processo de ensino-aprendizagem de algoritmos com apoio da *gamificação*.

## 9 CRITÉRIOS DE QUALIDADE PRIMÁRIOS

Segundo Kitchenham (2007), em adição aos critérios de inclusão e exclusão, é importante avaliar a qualidade dos estudos primários. Para tanto, dentro dessa revisão, os critérios de qualidade são adaptados de Costa (2010). A autora descreve critérios suficientemente abrangentes para cobrir o escopo dos estudos a serem considerados. Para fins dessa revisão foram feitas as modificações necessárias ao se considerar as questões de pesquisa, apresentadas no Quadro 02.

Quadro 02 – Questões para avaliação da qualidade dos estudos

Item	Critérios de Qualidade
<b>Introdução/Planejamento</b>	
01	Os objetivos ou questões do estudo são claramente definidos (incluindo justificativas para a realização do estudo)?
02	O tipo de estudo está definido claramente?
<b>Desenvolvimento</b>	
03	Existe uma clara descrição do contexto no qual a pesquisa foi realizada?
04	O trabalho é bem/adequadamente referenciado (apresenta trabalhos relacionados ou semelhantes e baseia-se em modelos e teorias da literatura)?
<b>Conclusão</b>	
05	O estudo relata de forma clara e não ambígua os resultados?
06	Os objetivos ou questões do estudo são alcançados?
<b>Critérios para a questão de investigação</b>	
7	O estudo lista primária ou secundariamente os modelos, processos, métodos, técnicas, metodologia e afins, para apoiar o processo <i>gamificado</i> de ensino-aprendizagem de algoritmos?
8	O estudo apresenta ferramentas de apoio ao processo <i>gamificado</i> de ensino-aprendizagem de algoritmos?
<b>Critério específico para estudos experimentais</b>	
9	Existe um método ou um conjunto de métodos descrito para a realização do estudo?
<b>Critério específico para estudos teóricos</b>	
10	Existe um processo não tendencioso na escolha dos estudos?
<b>Critério específico para Revisões Sistemáticas</b>	
11	Existe um protocolo rigoroso, descrito e seguido?
<b>Critério específico para relato de experiência no ambiente acadêmico</b>	
12	Existe uma descrição sobre a(s) instituição(ões) onde foi conduzido o estudo?

## 10 PROCESSOS DE BUSCA NA FONTE

No decorrer da condução desta revisão sistemática os métodos de busca dos estudos primários definidos anteriormente seguirão os seguintes processos (Quadro 03):

Quadro 03 – Algoritmo do processo de busca

Busca	Algoritmo
Automática	1 – Gerar <i>string</i> de busca específica pela fonte selecionada. 2 – Executar a <i>string</i> de busca na fonte selecionada. 3 – Catalogar estudos retornados pelas <i>engines</i> de busca na ferramenta JabRef. 4 – Após leitura prévia do estudo, analisar sua relevância. Caso o estudo tenha relevância, mantê-lo na pesquisa. Caso contrário, descartar estudo.

## 11 PROCESSO DE SELEÇÃO DOS ESTUDOS PRIMÁRIOS

Os seguintes recursos serão utilizados no decorrer da execução desta revisão sistemática da literatura:

- Dois pesquisadores (um aluno de mestrado do PPGCIMES e um graduando do curso de Bacharelado em Sistemas de Informação da UFPA).
- Acesso às fontes de pesquisa por meio do domínio da Universidade Federal do Pará.
- Validações sobre documentos e procedimentos da realização da revisão sistemática por meio de reuniões com a Profa. Dr. Marianne Kogut Eliasquevici (Docente do PPGCIMES) e Prof. Dr. Sandro Ronaldo Bezerra Oliveira (Docente Programa de Pós-Graduação em Ciência da Computação).

Durante a condução desta revisão sistemática, os estudos primários serão identificados conforme o processo seguinte:

1. A partir da leitura de resumo, introdução e conclusão, os artigos deverão ser avaliados quanto aos critérios de inclusão e exclusão e o resultado deve ser registrado.
2. Os dois pesquisadores responsáveis pela seleção dos artigos devem entrar em consenso, quando necessário. Isso ocorre quando não há uma unanimidade na inclusão de um artigo e pelo menos um pesquisador decidir incluir.
3. Na fase de consenso, em caso de discordância sobre a inclusão de algum estudo, o mesmo deverá ser incluído.
4. Os estudos primários identificados serão posteriormente lidos em totalidade e então será aplicada a avaliação de qualidade e a estratégia de extração de dados, conforme descrito nas subseções seguintes.

## 12 AVALIAÇÃO DA QUALIDADE DOS ESTUDOS PRIMÁRIOS

A avaliação de qualidade é o segundo filtro para refinar os estudos que entraram na pesquisa. Na qual, é considerada a observação do pesquisador sobre o artigo, como forma de refinamento e validação de sua pesquisa, em que é considerado o grau de importância individual do estudo para possíveis comparações durante a síntese dos dados (KITCHENHAM; CHARTERS, 2007).

Em complementação, uma avaliação de qualidade pode ser utilizada como recomendação de estudos para futuras pesquisas segundo Kitchenham e Charters (2007).

Para manter neutralidade na pesquisa, os artigos excluídos por critérios de qualidade devem ser citados, conjuntamente com as razões de sua exclusão.

O presente protocolo de revisão sistemática conta com 12 critérios de qualidade, os quais são formados por dois conjuntos. O primeiro conjunto, os critérios de 1 a 8, são critérios vinculados ao artigo. E o segundo conjunto, os critérios de 9 a 12 são específicos para relatos de artigo que trabalhem com metodologia específica, definida pelo critério.

O presente protocolo de revisão sistemática conta com 12 critérios de qualidade, sendo que os critérios de 1 a 8 se aplicam aos estudos primários, e os de 9 a 12 somente se aplicam aos tipos específicos de trabalho, já mencionados nos critérios.

### 12.1 PROCESSO DE AVALIAÇÃO DE QUALIDADE

Os estudos primários selecionados por meio dos critérios de inclusão e exclusão serão lidos em suas totalidades e avaliados quanto aos critérios de qualidade. Para identificar o grau de adequação aos critérios de qualidade, será adotada a escala Likert-5, estratégia proposta por Costa (2010), em que são possibilitadas respostas gradativas, sendo elas: 0 (discordo totalmente) à 4 (concordo totalmente). O Quadro 04 apresenta a escala de avaliação a ser utilizada, de acordo com a proposta por Costa (2010).

Quadro 04 – Escala Likert-5 para avaliação da qualidade de estudos (Continua...)

Escala de Likert-5	
Concordo totalmente (4)	Deve ser concedido no caso em que o trabalho apresente no texto os critérios que atendam totalmente a questão.
Concordo parcialmente (3)	Deve ser concedido no caso em que o trabalho atenda parcialmente aos critérios da questão.



Quadro 04 – Escala Likert-5 para avaliação da qualidade de estudos (Conclusão)

Escala de Likert-5	
Neutro (2)	Deve ser concedido no caso em que o trabalho não deixe claro se atende ou não a questão.
Discordo parcialmente (1)	Deve ser concedido no caso em que os critérios contidos na questão não são atendidos pelo trabalho avaliado.
Discordo totalmente (0)	Deve ser concedido no caso em que não existe nada no trabalho que atenda aos critérios da questão.

Para auxiliar a avaliação, seguindo a escala Likert-5, foi elaborado o Quadro 05, adaptado de Malcher (2015), no qual é considerado o critério para definir a aderência do estudo dentro da escala.

Quadro 05 – Exemplo de avaliação de um estudo (Continua...)

Escala por Critério	
Critério	Escala
1	4 - Define e justifica o estudo claramente. 3 - Define claramente o estudo, porém a justificativa não é clara. 2 - Define claramente o estudo, mas não justifica. 1 - A definição dos objetivos do estudo não é clara. 0 - Não define o estudo.
2	4 - Define o tipo de estudo, referenciando na literatura a metodologia. 3 - Define o tipo de estudo, porém sem referenciar a metodologia. 2 - Não define o tipo de estudo. É possível inferir facilmente. 1 - Não define o tipo de estudo. É possível inferir com dificuldade. 0 - Não é possível inferir o tipo de estudo.
3	4 - Define claramente uma seção com o contexto da pesquisa. 3 - O contexto da pesquisa está incluído em uma seção não exclusiva. 2 - O contexto da pesquisa está disperso ao longo do texto. 1 - O contexto da pesquisa está disperso e é insubstancial. 0 - O contexto da pesquisa não é abordado.
4	4 - O texto apresenta uma seção de trabalhos relacionados. 3 - O texto apresenta trabalhos relacionados em uma seção não exclusiva. 2 - O texto apresenta trabalhos relacionados dispersos ao longo do texto. 1 - O texto não apresenta trabalhos relacionados, mas se apoia na literatura. 0 - O texto não apresenta trabalhos relacionados nem se apoia na literatura.
5	4 - Resultados são claramente apresentados na seção de conclusão. 3 - Resultados são claramente referenciados na seção de conclusão. 2 - Resultados apresentados na conclusão não são claros. 1 - Resultados referenciados na conclusão não são claros. 0 - Não são apresentados resultados.
6	4 - Os resultados estão totalmente aderentes ao objetivo do estudo. 3 - Os resultados estão aderentes ao objetivo do estudo, no entanto o autor faz ressalvas. 2 - Os resultados são parcialmente aderentes ao objetivo do estudo. 1 - Os resultados não estão aderentes ao objetivo do estudo. 0 - Não é alcançado nenhum resultado.

Quadro 05 – Exemplo de avaliação de um estudo (Conclusão)

Escala por Critério	
Critério	Escala
7	4 – Algum dos elementos é claramente descrito. 3 – Algum dos elementos é apresentado, mas não aprofundado. 2 – Algum dos elementos é avaliado, porém não descrito. 1 – Algum dos elementos é citado indiretamente. 0 – Nenhum dos elementos é apresentado direta ou indiretamente.
8	4 – Apresenta uma ferramenta disponível para uso em versão funcional. 3 – Apresenta ferramenta em versão funcional, porém não disponível para uso. 2 – Apresenta um projeto ou proposta de ferramenta, incluindo a descrição de funcionalidades. 1 – Apresenta apenas algoritmos de apoio ou análises de ferramentas de suporte ao processo de ensino e aprendizagem de Algoritmos. 0 – Não são apresentadas ferramentas de apoio.
9	4 – O método de experimento é definido e referenciado claramente. 3 – O método de experimento é definido claramente. 2 – O método de experimento é citado. 1 – O método de experimento não é citado, porém é possível inferir. 0 – Não é possível inferir o método de experimento.
10	4 – O texto descreve critérios para a escolha dos estudos. 3 – O texto não descreve critérios para a escolha dos estudos, porém apresenta estudos que discordam do estudo apresentado. 2 – O texto descreve apenas estudos aderentes ao estudo apresentado. 1 – O texto descreve estudos insuficientes. 0 – O texto não descreve estudos base.
11	4 – O protocolo de revisão é apresentado, descrito e seguido. 3 – O protocolo de revisão é apresentado e descrito, porém há evidências de que não foi seguido adequadamente. 2 – O protocolo de revisão não foi suficientemente descrito. 1 – O protocolo de revisão apenas foi citado ao longo do texto. 0 – Não há um protocolo de revisão.
12	4 – A área de atuação, tamanho e origem da organização são informados. 3 – Apenas duas das características do item 4 são informadas. 2 – Apenas uma das características do item 4 é informada. 1 – Nenhuma das características do item 4 é informada. 0 – O estudo não foi conduzido em uma ou mais organizações.

No estudo de Beecham (2007) sobre RSL ele define cinco níveis de qualidade para enquadrar os estudos que foram avaliados no critério de qualidade, sendo eles: excelente, muito boa, boa, média e baixa. Já em Costa (2007) temos uma construção de um intervalo estatístico para cada um dos níveis, apresentados no Quadro 06.

O cálculo para classificar o estudo no intervalo estatístico seguirá em similar os procedimentos já definidos por Costa (2007). Então, para esse protocolo ao ser feita a análise dos critérios de qualidade pelos revisores, será atribuído um número da escala para cada critério.

E, ao final será feita a soma e verificado a porcentagem para enquadrar o estudo segundo os níveis de qualidade, descritos no Quadro 6.

Quadro 06 – Avaliação da Qualidade

Faixa de Notas	Avaliação
Excelente	>86%
Muito Boa	66%-85%
Boa	46%-65%
Média	26%-45%
Baixa	< 26%

Fonte: (COSTA, 2010).

### 13 ESTRATÉGIA DE EXTRAÇÃO DE INFORMAÇÕES

No decorrer do processo de extração das informações será utilizada a *string* de busca para buscar os estudos nas bases de dados e, fazer o *download* do arquivo bib com os estudos, com seus respectivos títulos, abstracts e palavras chaves.

Após a extração será preenchido o formulário de inclusão e exclusão, segundo os critérios já definidos na seção 9, exemplificado no Quadro 07. Em seguida, será lido na integra os estudos que foram incluídos na pesquisa e avaliados segundo os critérios de qualidade definidos na seção 13.1 (Quadro 08).

Quadro 07 – Formulário para inclusão e exclusão de estudos

Estudo (ID)	CE01	CE02	CE03	CE05	CE06	CE07	CI01	Resultado	Observação

Para o preenchimento da Tabela 6 serão ajustados os estudos, após, serão avaliados os critérios de exclusão de 1 a 7, e o critério de inclusão. No campo, resultado, será colocado incluído, caso o estudo tenha sido aprovado no critério de inclusão ou não reprovado em nenhum de exclusão. No caso da observação serão anotados fatores justificantes de inclusão ou exclusão do estudo.

Quadro 08 – Formulário de avaliação de qualidade

Estudo (ID)		Pesquisador		Data Avaliação	
Título do Trabalho					
Autores					
Fonte				Ano	
Tipo de estudo				Local publicação	
Item	Critério Qualidade				Valor
1					
2					
3					
N					
Resultado					

### 13.1 EXTRAÇÃO DE RESULTADOS

Durante a condução da RSL, na etapa de extração dos resultados, após a análise do estudo primário, será elaborado um resumo, no qual deverá ser identificado o formato da abordagem do processo de ensino e aprendizagem de algoritmos, procurando responder também as questões secundárias de pesquisa. O Quadro 08 descreve a forma como as questões secundárias serão analisadas.

Quadro 09 – Análise das questões secundárias (Continua...)

Questão secundária	Forma de análise
QS01: Qual a abordagem mais utilizada para <i>gamificar</i> o processo de ensino e aprendizagem da disciplina Algoritmos ou equivalente?	Para as abordagens retornadas, identificar e resumir a principal metodologia de apoio ao processo de ensino e aprendizagem, para essa situação será escolhido a mais referenciada ou citada.
QS02: Quais as principais dificuldades encontradas ao se <i>gamificar</i> o ensino da disciplina Algoritmos ou equivalente?	Para cada relato de experiência sobre o processo de ensino e nos cursos de nível superior, um dos pontos tratado será a dificuldade, será analisado se nos diferentes relatos são encontradas dificuldades semelhantes.
QS03: Quais as principais dificuldades encontradas no aprendizado <i>gamificado</i> da disciplina Algoritmos ou equivalente?	Durante os relatos de experiência sobre o processo de aprendizagem de algoritmos, um dos principais tópicos é a dificuldade, em que será analisado se nos diferentes relatos são encontradas dificuldades semelhantes.

Quadro 09 – Análise das questões secundárias (Conclusão)

Questão secundária	Forma de análise
QS04: Como se realiza o processo <i>gamificado</i> de ensino e aprendizado da disciplina Algoritmos ou equivalente?	Para os relatos de experiência serão investigados dois pontos no que tange a essa questão, são os <i>frameworks</i> adotados como base para a abordagem e os elementos de jogos trabalhados dentro da <i>gamificação</i> .
QS05: Existem relatos de sucesso do processo <i>gamificado</i> de ensino e aprendizagem da disciplina Algoritmos ou equivalente?	Para cada relato de experiência foi averiguado, segundo a posição dos autores, cada sucesso e foi relatado numericamente nesse ponto.
QS06: Existem relatos de insucesso do processo <i>gamificado</i> de ensino e aprendizagem da disciplina Algoritmos ou equivalente?	Semelhante ao QS05 só que o ponto investigado será o insucesso.
QS07: Existem relatos de avaliação do processo <i>gamificado</i> de ensino e aprendizagem da disciplina Algoritmos ou equivalente?	Para cada um dos estudos será averiguado se foi realizado o processo de avaliação e nessa questão será descrita numericamente.
QS08: Existem relatos de validação do processo <i>gamificado</i> de ensino e aprendizagem da disciplina Algoritmos ou equivalente?	Semelhante a QS07 porém investigando o termo validação.
QS09: Como foi feita a validação do processo <i>gamificado</i> de ensino e aprendizagem da disciplina Algoritmos ou equivalente?	Para cada relato de experiência foi constatada um formato.

## 14 SUMARIZAÇÃO DOS RESULTADOS

Na apresentação dos resultados dessa RSL, o principal gráfico é o do quantitativo sobre as abordagens *gamificadas* do ensino de algoritmos. A partir de cada questão de pesquisa poderá ser gerado os seguintes gráficos:

- Quantidade de abordagem do ensino de algoritmos.
- Quantitativo de estudos por tipo (Experimental, Teórico, Revisão Sistemática).
- Evolução ao longo do tempo a quantidade de estudos selecionados.
- Artigos selecionados por base de dados.

## APÊNDICE G – LISTA DE REFERÊNCIAS DOS ARTIGOS SELECIONADOS (EM ORDEM CRONOLÓGICA)

(PS001) REGALADO, Murilo Rocha; ARANHA, Eduardo; DA SILVA, Thiago Reis. Gamifying an online approach for promoting game development Learning and contest: An experience report. In: **Frontiers in Education Conference (FIE)**, 2016 IEEE. IEEE, 2016. p. 1-8.

(PS002) MILJANOVIC, Michael A.; BRADBURY, Jeremy S. Robot ON!: a serious game for improving programming comprehension. In: **Games and Software Engineering (GAS), 2016 IEEE/ACM 5th International Workshop on**. IEEE, 2016. p. 33-36.

(PS003) SAITO, Daisuke; WASHIZAKI, Hironori; FUKAZAWA, Yoshiaki. Analysis of the learning effects between text-based and visual-based beginner programming environments. In: **Engineering Education (ICEED), 2016 IEEE 8th International Conference on**. IEEE, 2016. p. 208-213.

(PS004) BOGDANOVYCH, Anton; TRESKAK, Tomas. Teaching Programming Fundamentals to Modern University Students. In: **CSEDU (2)**. 2016. p. 308-317.

(PS005) CHANG, Chih-Tsan; TSAI, Cheng-Yu; CHENG, Po-Jen; YU, Pao-Ta. Toward a teaching strategy design for game-based learning. In: **Workshop Proceeding of the 24<sup>th</sup> International Conference on Computers in Education**. 2016. P. 26-32.

(PS006) FOTARIS, Panagiotis; MASTORAS, Theodoros; LEINFELLNER, Richard; ROSUNALLY, Yasmine. Climbing up the leaderboard: An empirical study of applying gamification techniques to a computer programming class. **Electronic Journal of e-learning**, v. 14, n. 2, p. 94-110, 2016.

(PS007) HOOSHYAR, Danial; AHMAD, Rodina B.; YOUSEFI, Moslem; FATHI, Moein; HORNG, Shi-Jinn; HEUISEOK, Lim. Applying an online game-based formative assessment in a flowchart-based intelligent tutoring system for improving problem-solving skills. **Computers & Education**, v. 94, p. 18-36, 2016.

(PS008) MLADENović, Saša; KRPAŃ, Divna; MLADENović, Monika. Using games to help novices embrace programming: from elementary to higher education. **International journal of engineering education**, v. 32, n. 1B, p. 521-531, 2016.

(PS009) PAIVA, José Carlos; LEAL, José Paulo; QUEIRÓS, Ricardo Alexandre. Enki: A pedagogical services aggregator for learning programming languages. In: **Proceedings of the 2016 ACM Conference on Innovation and Technology in Computer Science Education**. ACM, 2016. p. 332-337.

(PS010) RIGO, Sandro José; DIEHL, Pablo. Tri-Lua: Using Gamification as Support Learning Programming Language. In: **European Conference on Games Based Learning**. Academic Conferences International Limited, 2016. p. 922.

(PS011) SPRINT, Gina; COOK, Diane. Enhancing the CS1 student experience with gamification. In: **Integrated STEM Education Conference (ISEC), 2015 IEEE**. IEEE, 2015. p. 94-99.

(PS012) BITTENCOURT, Roberto A.; SANTOS, David M. B.; RODRIGUES, C. A.; BATISTA, Washington P.; CHARLEGRE, Henderson S. Learning programming with peer support, games, challenges and scratch. In: **2015 IEEE Frontiers in Education Conference (FIE)**. IEEE, 2015. p. 1-9.

(PS013) SAITO, Daisuke; WASHIZAKI, Hironori; FUKAZAWA, Yoshiaki. Work in progress: A comparison of programming way: Illustration-based programming and text-based programming. In: **Teaching, Assessment, and Learning for Engineering (TALE), 2015 IEEE International Conference on**. IEEE, 2015. p. 220-223.

(PS014) SAJANA, A.; BIJLANI, Kamal; JAYAKRISHNAN, R. An interactive serious game via visualization of real life scenarios to learn programming concepts. In: **Computing, Communication and Networking Technologies (ICCCNT), 2015 6th International Conference on**. IEEE, 2015. p. 1-8.

(PS015) YUE, Wong Seng; WAN, Wong Lai. The effectiveness of digital game for introductory programming concepts. In: **Internet Technology and Secured Transactions (ICITST), 2015 10th International Conference for**. IEEE, 2015. p. 421-425.

(PS016) AZMI, Shahdatunnaim; IAHAD, Noorminshah A.; AHMAD, Norasnita. Gamification in online collaborative learning for programming courses: A literature review. **ARPN Journal of Engineering and Applied Sciences**, v. 10, n. 23, p. 1-3, 2015.

(PS017) DIETRICH, Jens; TANDLER, Johannes; SUI, Li; MEYER, Manfred. The primegame revolutions: A cloud-based collaborative environment for teaching introductory programming. In: **Proceedings of the ASWEC 2015 24th Australasian Software Engineering Conference**. ACM, 2015. p. 8-12.

(PS018) FIGUEIREDO, Karen da S. Gamified Courses in Information Systems Program. In: **Proceedings of the annual conference on Brazilian Symposium on Information Systems: Information Systems: A Computer Socio-Technical Perspective-Volume 1**. Brazilian Computer Society, 2015. p. 81.

(PS019) FOTARIS, Panagiotis; MASTORAS, Theodoros; LEINFELLNER, Richard; ROSUNALLY, Yasmine. From hiscore to high marks: Empirical study of teaching programming through gamification. In: **European Conference on Games Based Learning**. Academic Conferences International Limited, 2015. p. 186.

(PS020) HAKULINEN, Lasse; AUVINEN, Tapio; KORHONEN, Ari. The effect of achievement badges on students' behavior: An empirical study in a university-level computer science course. **International Journal of Emerging Technologies in Learning (IJET)**, v. 10, n. 1, p. 18-29, 2015.

(PS021) OREHOVAČKI, Tihomir; BABIĆ, Snježana. Evaluating the quality of games designed for learning programming by students with different educational background: An empirical study. In: **Information and Communication Technology, Electronics and Microelectronics (MIPRO), 2015 38th International Convention on**. IEEE, 2015. p. 963-968.

(PS022) OREHOVAČKI, Tihomir; BABIĆ, Snježana. Inspecting quality of games designed for learning programming. In: **Learning and Collaboration Technologies**. Springer, Cham, 2015. p. 620-631.

(PS023) HAKULINEN, Lasse; AUVINEN, Tapio. The effect of gamification on students with different achievement goal orientations. In: **Teaching and Learning in Computing and Engineering (LaTiCE), 2014 International Conference on**. IEEE, 2014. p. 9-16.

(PS024) IBÁÑEZ, María-Blanca; DI-SERIO, Angela; DELGADO-KLOOS, Carlos. Gamification for engaging computer science students in learning activities: A case study. **IEEE Transactions on learning technologies**, v. 7, n. 3, p. 291-301, 2014.

(PS025) DRUMOND, Rafael Rego; DE DAMASCENO, André Luiz Brandão; NETO, Carlos de Salles Soares. Wanda: a framework to develop card based games to help motivate programming students. In: **Computer Games and Digital Entertainment (SBGAMES), 2014 Brazilian Symposium on**. IEEE, 2014. p. 158-164.

(PS026) ZHANG, Jinghua; ATAY, Mustafa; SMITH, Emanuel; CARDWELL, Elvira R.; JONES, Elva J. Using a game-like module to reinforce student understanding of recursion. In: **Frontiers in Education Conference (FIE), 2014 IEEE**. IEEE, 2014. p. 1-7.

(PS027) HIJON-NEIRA, Raquel; VELÁZQUEZ-ITURBIDE, Ángel; PIZARRO-ROMERO, Celeste; CARRIÇO, Luís. Serious games for motivating into programming. In: **Frontiers in Education Conference (FIE), 2014 IEEE**. IEEE, 2014. p. 1-8.

(PS028) VAHLDICK, Adilson; MENDES, António José; MARCELINO, Maria José. A review of games designed to improve introductory computer programming competencies. In: **Frontiers in Education Conference (FIE), 2014 IEEE**. IEEE, 2014. p. 1-7.

(PS029) PARALIČ, M.; PIETRIKOVÁ, E. Learning by game creation in introductory programming course: 5-Year-long study. In: **Emerging eLearning Technologies and Applications (ICETA), 2014 IEEE 12th International Conference on**. IEEE, 2014. p. 391-396.

(PS030) FERREIRA, João F.; MENDES, Alexandra. The magic of algorithm design and analysis: teaching algorithmic skills using magic card tricks. In: **Proceedings of the 2014 conference on Innovation & technology in computer science education**. ACM, 2014. p. 75-80.

(PS031) HIJON-NEIRA, Raquel B.; VELÁZQUEZ-ITURBIDE, Ángel; PIZARRO-ROMERO, Celeste. Game programming for improving learning experience. In: **Proceedings of the 2014 conference on Innovation & technology in computer science education**. ACM, 2014. p. 225-230.

(PS032) KNUTAS, Antti; IKONEN Jouni; NIKULA, Uolevi; PORRAS, Jari. Increasing collaborative communications in a programming course with gamification: a case study. In: **Proceedings of the 15th International Conference on Computer Systems and Technologies**. ACM, 2014. p. 370-377.



(PS033) MALLIARAKIS, Christos; SATRATZEMI, Maya; XINOGALOS, Stelios. CMX: Implementing an MMORPG for learning programming. In: **8th European Conference on Games Based Learning: ECGBL2014**. 2014. p. 346-355.

(PS034) PICCIONI, Marco; ESTLER, Christian; MEYER, Bertrand. SPOC-supported introduction to programming. In: **Proceedings of the 2014 conference on Innovation & technology in computer science education**. ACM, 2014. p. 3-8.

(PS035) HAKULINEN, Lasse; AUVINEN, Tapio; KORHONEN, Ari. Empirical study on the effect of achievement badges in TRAKLA2 online learning environment. In: **Learning and Teaching in Computing and Engineering (LaTiCE), 2013**. IEEE, 2013. p. 47-54.

(PS036) ZHANG, Jinghua; ATAY, Mustafa; CALDWELL, Elvira R.; JONES, Elva J. Visualizing loops using a game-like instructional module. In: **2013 IEEE 13th International Conference on Advanced Learning Technologies (ICALT)**. IEEE, 2013. p. 448-450.

(PS037) TILLMANN, Nikolai; HALLEUX, Jonathan; XIE, Tao; GULWANI, Sumit; BISHOP, Judith. Teaching and learning programming and software engineering via interactive gaming. In: **Proceedings of the 2013 International Conference on Software Engineering**. IEEE Press, 2013. p. 1117-1126.

(PS038) ADAMO-VILLANI, Nicoletta; HALEY-HERMIZ, Thomas; CUTLER, Robb. Using a serious game approach to teach 'operator precedence' to introductory programming students. In: **Information Visualisation (IV), 2013 17th International Conference**. IEEE, 2013. p. 523-526.

(PS039) KISS, Gabor. Teaching programming in the higher education not for engineering students. **Procedia-Social and Behavioral Sciences**, v. 103, p. 922-927, 2013.

(PS040) BAKER, Amos; ZHANG, Jinghua; CALDWELL, Elvira R. Reinforcing array and loop concepts through a game-like module. In: **Computer Games (CGAMES), 2012 17th International Conference on**. IEEE, 2012. p. 175-179.

(PS041) MARQUES, Bradley RC; LEVITT, Stephen P.; NIXON, Ken J. Video games as a medium for software education. In: **Games Innovation Conference (IGIC), 2012 IEEE International**. IEEE, 2012. p. 1-4.

(PS042) JANTAN, Siti Robaya; ALJUNID, Syed Ahmad. An experimental evaluation of scaffolded educational games design for programming. In: **Open Systems (ICOS), 2012 IEEE Conference on**. IEEE, 2012. p. 1-6.

(PS043) CAMBRAIA, Adão Caron; OLIVEIRA, Maria Angélica F. Learning to program: a game or a boogeyman. In: **Anais 2nd International Conference on Design and Modeling in Science, Education and Technology. Orlando, Flórida, USA**. 2012.

(PS044) DOUADI, Bourouaieh; TAHAR, Bensebaa; HAMID, Seridi. Smart edutainment game for algorithmic thinking. **Procedia-Social and Behavioral Sciences**, v. 31, p. 454-458, 2012.

(PS045) KAZIMOGLU, Cagin; KIERNAN, Mary; BACON, Liz; MACKINNON, Lachlan. A serious game for developing computational thinking and learning introductory computer programming. **Procedia-Social and Behavioural Sciences**, v. 47, p. 1991-1999, 2012.

(PS046) KAZIMOGLU, Cagin; KIERNAN, Mary; BACON, Liz; MACKINNON, Lachlan. Learning programming at the computational thinking level via digital game-play. **Procedia Computer Science**, v. 9, p. 522-531, 2012.

(PS047) LJUNGKVIST, Peter; MOZELIUS, Peter. Educational games for self learning in introductory programming courses-a straightforward design approach with progression mechanisms. In: **Proceedings Of The 6th European Conference On Games Based Learning, ECGBL**. 2012. p. 285-293.

(PS048) MASSO, Nicholas; GRACE, Lindsay. Shapemaker: A game-based introduction to programming. In: **Computer Games (CGAMES), 2011 16th International Conference on**. IEEE, 2011. p. 168-171.

(PS049) YEH, Kuo-Chuan Martin; CHEN, Wei-Fan. WIP: Using a Computer Gaming Strategy to facilitate Undergraduates' Learning in a Computer Programming Course: An Experimental Study.

(PS050) COELHO, António; KATO, Enrique; XAVIER, João; GONÇALVES, Ricardo. Serious game for introductory programming. In: **International Conference on Serious Games Development and Applications**. Springer, Berlin, Heidelberg, 2011. p. 61-71.

(PS051) ANGOTTI, Robin; HILLYARD, Cinnamon; PANITZ, Michael; SUNG, Kelvin; MARINO, Keri et al. Game-themed instructional modules: a video case study. In: **Proceedings of the Fifth International Conference on the Foundations of Digital Games**. ACM, 2010. p. 9-16.

(PS052) IBRAHIM, Roslina; SEMARAK, Jalan; JAAFAR, Azizah. Using educational games in learning introductory programming: A pilot study on students' perceptions. In: **Information Technology (ITSim), 2010 International Symposium in**. IEEE, 2010. p. 1-5.

(PS053) EAGLE, Michael; BARNES, Tiffany. Experimental evaluation of an educational game for improved learning in introductory computing. In: **ACM SIGCSE Bulletin**. ACM, 2009. p. 321-325.

(PS054) EAGLE, Michael; BARNES, Tiffany. Evaluation of a game-based lab assignment. In: **Proceedings of the 4th International Conference on Foundations of Digital Games**. ACM, 2009. p. 64-70.

(PS055) JIAU, Hewijin Christine; CHEN, Jinghong Cox; SSU, Kuo-Feng. Enhancing self-motivation in learning programming using game-based simulation and metrics. **IEEE Transactions on Education**, v. 52, n. 4, p. 555-562, 2009.

(PS056) MURATET, Mathieu; TORGUET, Patrice; JESSEL, Jean-Pierre; VIALLET, Fabienne. Towards a serious game to help students learn computer programming. **International Journal of Computer Games Technology**, v. 2009, p. 3, 2009.

(PS057) EAGLE, Michael; BARNES, Tiffany. Wu's castle: teaching arrays and loops in a game. In: **ACM SIGCSE Bulletin**. ACM, 2008. p. 245-249.

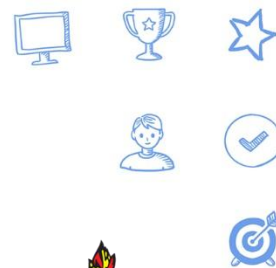
(PS058) EDIRISINGHE, E. M. Teaching students to identify common programming errors using a game. In: **Proceedings of the 9th ACM SIGITE conference on Information technology education**. ACM, 2008. p. 95-98.

(PS059) ANDERSON, Eike Falk; MCLOUGHLIN, Leigh. Critters in the classroom: a 3D computer-game-like tool for teaching programming to computer animation students. In: **ACM SIGGRAPH 2007 educators program**. ACM, 2007. p. 7.

(PS060) LEUTENEGGER, Scott; EDGINGTON, Jeffrey. A games first approach to teaching introductory programming. In: **ACM SIGCSE Bulletin**. ACM, 2007. p. 115-118.

(PS061) BAYLISS, Jessica D.; STROUT, S. Games as a "flavor" of CS1. In: **ACM SIGCSE Houston**. ACM, 2006. p. 500-504.

**APÊNDICE G – PLANO DE ENSINO-APRENDIZAGEM GAMIFICADO – VERSÃO  
FINAL DO PRODUTO**



**UNIVERSIDADE FEDERAL DO PARÁ  
NÚCLEO DE INOVAÇÃO E TECNOLOGIAS  
APLICADAS A ENSINO E EXTENSÃO  
PROGRAMA DE PÓS-GRADUAÇÃO CRIATIVIDADE  
E INOVAÇÃO EM METODOLOGIAS DE ENSINO SUPERIOR**

**UM FRAMEWORK GAMIFICADO PARA A DISCIPLINA ALGORITMOS OU  
EQUIVALENTE**

**PLANO DE ENSINO**

**DISCENTE: JOSÉ AUGUSTO DE SENA QUARESMA  
ORIENTADORA: PROFA. DRA. MARIANNE KOGUT ELIASQUEVICI  
COORIENTADOR: PROF. DR. SANDRO RONALDO BEZERRA OLIVEIRA**

**BELÉM  
2019**



## APRESENTAÇÃO

Este Plano de Ensino é parte integrante do *Framework Gamificado*, produto educacional, fruto da pesquisa de mestrado do discente José Augusto de Sena Quaresma, sob orientação da professora Doutora Marianne Kogut Eliasquevici e coorientação do professor Doutor Sandro Ronaldo Bezerra Oliveira, no Programa de Pós-Graduação Criatividade e Inovação em Metodologias de Ensino Superior (PPGCIMES), do Núcleo de Inovação e Tecnologias Aplicadas a Ensino e Extensão (NITAE<sup>2</sup>), da Universidade Federal do Pará (UFPA). A dissertação de mestrado intitula-se: “Um *framework gamificado* para a disciplina algoritmos ou equivalente”.

Para uso dos(as) docentes, o produto encontra-se disponível acessando os *links* a seguir:

- Plano de Ensino: <http://twixar.me/w811>
- Planilha *Gamificada*: <http://twixar.me/b811>

Tanto o Plano de Ensino, quanto a Planilha *Gamificada*, podem ser adaptadas, desde que citadas as fontes de referência.

Esperamos que a experiência seja enriquecedora para todos(as)!

<<Nome da Universidade>>

<<Nome da Faculdade>>

<<  
Logo  
Instituição  
>>

## PLANO DE ENSINO-APRENDIZAGEM *GAMIFICADO*<sup>5</sup>

### 1 IDENTIFICAÇÃO

**Professor(a):** <<Nome do professor(a)>>

**Disciplina:** <<Nome da disciplina. Ex: Algoritmos>>

**Curso:** <<Nome do Curso>>

**Semestre:** <<Semestre de aplicação>>

**Ano:** <<Ano de aplicação>>

**Carga horária:** <<Quantidade de horas geral, semanal e encontros semanais>>

### 2 CONTEÚDO

- Estrutura de dados homogêneas:
  - Vetor
  - Matriz

### 3 OBJETIVO GERAL DA DISCIPLINA

Desenvolver no aluno a capacidade de compreender e elaborar algoritmos com qualidade e eficiência para a solução de problemas computacionais, independente de uma linguagem de programação.

#### 3.1 OBJETIVO DE ENSINO PARA ESTA UNIDADE DE CONTEÚDO

Capacitar o discente da Disciplina após a conclusão dos estudos sobre o conteúdo, em desenvolver e analisar Estrutura de dados homogênea.

---

<sup>5</sup> Ao usar este Plano de Ensino, é importante citar que foi adaptado desta dissertação de mestrado defendida no Programa de Pós-Graduação Criatividade e Inovação em Metodologias de Ensino Superior.

### 3.2 OBJETIVOS DE APRENDIZAGEM PARA ESTA UNIDADE DE CONTEÚDO

1. Identificar estrutura de dados homogêneos em algoritmos.
2. Implementar estrutura de dados homogêneos em algoritmos e na linguagem de programação adotada para a disciplina.
3. Realizar teste para analisar qual o resultado do processamento da estrutura de dados homogêneos.

### 3.3 HABILIDADES E COMPETÊNCIAS (ESPECÍFICAS)

Especificar, modelar, implantar e avaliar estrutura de dados homogêneas em algoritmos.

## 4 PLANEJAMENTO

Para o conteúdo de que trata esse plano de ensino, foi adotado um *framework gamificado*, adaptado de Elgrably e Oliveira (2018) e Santos e Oliveira (2018). As informações referentes ao processo de *gamificação*, bem como as atividades, os recursos metodológicos e a forma como serão avaliados os alunos está presente na seção 5 desse plano de ensino.

## 5 PROCESSO DE GAMIFICAÇÃO

A *gamificação* é definida segundo Deterding *et al.* (2011) como o uso de elementos de jogos em ambientes que não são de jogos. A partir dessa premissa pretendemos *gamificar* as atividades em sala de aula ao se trabalhar com algumas dinâmicas, mecânicas e elementos de games para ensinar o conteúdo de estrutura de dados homogêneos.

Portanto, a avaliação e planejamento do processo de ensino e aprendizagem no conteúdo de estrutura de dados homogêneos serão baseados na *gamificação* aqui descrita. De maneira inicial informamos que os alunos serão avaliados de duas maneiras: uma em relação ao conteúdo, ao qual ele receberá pontos e um *avatar* a cada aula. E, a outra em relação a comportamento, ao qual o aluno receberá estrelas a cada aula.

### 5.1 JOGADORES

Nessa seção será descrito os personagens participantes dessa *gamificação*, a qual está descrito no Quadro 1.



Quadro 1 – Descrição de papéis

Identificação	Função	Nome
<b>Master</b>	Professor/ Pesquisador	<<Nome>>
<b>Mentora</b>	Professora da Disciplina	<<Nome>>
<b>Coach</b>	Acompanhamento na resolução de problemas	<<Nome do monitor>>
<b>Juiz/ Scoreplay</b>	Colaborador para desenvolvimento do score da turma	<<Nome>>
<b>Players</b>	Jogadores/ estudantes	Alunos da disciplina Algoritmos

## 5.2 DESCRIÇÃO DAS ATIVIDADES NO JOGO COM O CRONOGRAMA

Para efeito dessa *gamificação* identificamos como os dias de aula, atividades práticas desenvolvidas e a prova como abordagens. A informação referente a data, máxima pontuação e estrela que podem ser alcançadas, bem como uma rápida descrição de cada abordagem, estão apresentadas no Quadro 2.

O local onde acontecerão as aulas e atividades do *framework gamificado* será o <<Identifique a localização do laboratório ou sala de aula>>, localizado no <<Instituto e/ou Universidade aonde está localizado>>.

Os recursos que estarão disponíveis para essa *gamificação* será a estrutura do laboratório, o uso de um notebook pessoal do pesquisador e pinceis atômicos.

Quadro 2 – Cronograma das fases e a pontuação atribuída

Data	Abordagem	Atividade	Pontuação	Quantidade de Estrelas
<b>dd/mm/aaaa</b>	Conversa inicial sobre a <i>gamificação</i>	Não terá atividades	0	0
<b>dd/mm/aaaa</b>	Aula teórica e prática – Array Unidimensional (Seção 5.2.1)	Lista exercício sala – 4 questões	100	3
<b>dd/mm/aaaa</b>	Aula teórica e prática – Array Bidimensional	Lista exercício sala – 4 questões	100	3
<b>dd/mm/aaaa</b>	Dojo Randori (Seção 5.2.3)	Atividade em dupla rotativa avaliando a narrativa (copiloto) e a execução (piloto)	200	3
<b>dd/mm/aaaa</b>	Dojo Randori	Atividade em dupla rotativa avaliando a narrativa (copiloto) e a execução (piloto)		3
<b>dd/mm/aaaa</b>	Atividade Lab (Seção 5.2.4)	Atividade em dupla fixa, resolvendo 4 questões. Lista de exercício de casa (4 questões)	300 (200 – Atividade em sala + 100 – Lista de exercício de casa)	0
<b>dd/mm/aaaa</b>	Prova Teórica (Seção 5.2.5)	Teste individual (4 questões)	300	0

Data	Abordagem	Atividade	Pontuação	Quantidade de Estrelas
dd/mm/aaaa	Feedback (Seção (5.2.6))	Bonificação sobre a análise do ensino de <i>gamificação</i>		3

### 5.2.1 Aula Teórica e Prática

A aula teórica e prática é uma abordagem híbrida de ensino em que o professor mescla conceitos teóricos com práticas de programação. Para essa abordagem foram reservados dois dias de aula, no primeiro dia será focado em vetor e no segundo dia os estudos serão voltados para matriz.

Como trata-se de um processo *gamificado* o aluno ganhará pontuação e estrelas no dia, sendo que o processo de ganho de estrela está descrito na seção 5.5 desse documento. O aluno pontuará ao resolver as questões do exercício proposto. Para essa pontuação, utilizaremos a seguinte regra: O aluno ganha 10 pontos pela resolução da primeira questão, 20 pela segunda, 30 pela terceira e 40 pela quarta, por possuírem níveis de dificuldades diferentes; A lista de exercício proposta por dia possui quatro questões (Apêndices A e B), então diariamente o aluno terá um total de 100 pontos; A medida que o aluno receber a pontuação ele receberá um *avatar* diário, segundo a Quadro 3.

A condicional de teste para validar a pontuação do desafio pela resolução do aluno estará unicamente disponível para o professor e monitor da disciplina.

Quadro 3 – Avatar diário Aula Teórica e Prática

Avatar	Faixa de pontuação
<b>Analista</b>	90 – 100
<b>Programador</b>	70 – 89
<b>Aprendiz</b>	50 – 69
<b>Amador</b>	0 – 49

#### 5.2.1.1 Procedimentos Metodológicos

As duas aulas teóricas e práticas terão a mesma estrutura lógica de apresentação do conteúdo, contendo: uma Metáfora sobre o assunto; apresentação do conceito e a forma como é implementado em pascal; um exercício exemplo para o professor fazer junto com os alunos; uma lista de exercício com quatro questões; Por fim, serão retomados os pontos abordados na aula.

### 5.2.2 Dojo *Randori*

A abordagem Dojo *Randori* é uma prática de ensino de programação em ambiente seguro. O aluno aprenderá por tentativa e erro, com o auxílio de seus pares. Tal prática é adaptada do *framework* apresentado por Luz e Neto (2012). A abordagem possui os seguintes papéis: (i) piloto, responsável pelo desenvolvimento do código; (ii) copiloto, com a função de narrar e auxiliar no planejamento e codificação do programa; (iii) plateia, interage entre si com a intenção de cooperar com o piloto e copiloto durante o desenvolvimento. No contexto da sala de aula será uma das nossas dinâmicas. Contará com dois dias de desenvolvimento por conta do tamanho da turma e cada um dos alunos passará pelos três papéis já descritos.

Na atividade em questão os alunos serão avaliados por comportamento segundo os quesitos definidos na seção 5.5 desse documento. No que tange o conteúdo o aluno receberá pontuação segundo os quesitos estabelecidos na Quadro 4. O quesito pontuações pelas questões desenvolvidas pela turma será uma nota atribuída pelo professor em relação a codificação das funções realizadas pela turma, podendo ir de 0 até 50 pontos, ou seja, nesse atributo toda a turma receberá nota igual. As questões para a atividade do Dojo *Randori* estão presentes no Apêndice D.

Quadro 4 – Pontos adquiridos no Dojo *Randori*

Quesito	Pontos adquiridos
Uma questão concluída	20
Utilizou o fluxo Programar-Testar	30
Narrou como copiloto	100
Pontuações pelas questões desenvolvidas pela turma	Até 50

O *avatar* atribuído para a atividade do Dojo *Randori* seguirá a regra estabelecida na Quadro 5.

Quadro 5 – *Avatar* Dojo *Randori*

<i>Avatar</i>	Faixa de pontuação
<b>Analista</b>	180 – 200
<b>Programador</b>	150 – 179
<b>Aprendiz</b>	120 – 149
<b>Amador</b>	0 – 119

#### 5.2.2.1 Procedimentos Metodológicos

As duas aulas de Dojo seguirão o mesmo procedimento metodológico, o qual será: o professor explicará sobre as regras do Dojo *Randori* e apresentará a lista de exercício proposta

pra que os alunos programem em par; Após esse momento será solicitado que dois voluntários assumam os papéis de piloto e copiloto; Eles iniciam a codificação, após cinco minutos da atividade é permitido a plateia emitir sugestão na programação; Depois de dois minutos o piloto retorna à plateia, o copiloto assume a posição de piloto e um voluntário da plateia assume a posição de copiloto. O ciclo se repete até que todos os alunos tenham participado da atividade. Ao final, o professor recolherá os códigos gerados para análise.

### 5.2.3 Atividade LAB

A atividade Laboratório de Programação, abreviada como LAB, é uma abordagem proposta no *framework* de Elgrably e Oliveira (2018) e Santos e Oliveira (2018). Na proposta em questão, os alunos serão divididos em duplas cada uma com acesso a um computador, o qual deve estar sem acesso à internet, e terão uma lista com quatro desafios para responder.

Os alunos ganharam, nesse dia, pontos por cada desafio, sendo dividido da seguinte maneira: (i) 20 pontos para o desafio concluído; (ii) 30 pontos atribuídos pelo professor em relação a nota da codificação; (iii) A pontuação máxima por desafio será de 50 pontos. Como cada dupla receberá uma lista contendo quatro questões (Apêndice E) a pontuação máxima para o LAB será de 200 pontos. Além disso, para esse dia, foi reservado a entrega para o exercício feito em casa (Apêndice C), o qual pontuará segundo as regras descritas: (i) 25 pontos por questão resolvida de maneira correta; (ii) Um total de 100 pontos para as quatro questões resolvidas de maneira correta. Sendo assim, a pontuação total do dia será a soma dos pontos adquiridos, tanto na atividade LAB, quanto para a lista de exercício feito em casa. Portanto, para esse dia, teremos o *avatar* diário, segundo condições estabelecidas na Quadro 6. Logo, os alunos não receberão recurso estrela como descrito, na seção 5.5 desse documento.

Quadro 6 – *Avatar* da Atividade LAB em conjunto com a lista de exercício feita em casa

<i>Avatar</i>	Faixa de pontuação
<b>Analista</b>	270 – 300
<b>Programador</b>	230 – 269
<b>Aprendiz</b>	190 – 229
<b>Amador</b>	0 – 189

#### 5.2.3.1 Procedimentos Metodológicos

Na atividade LAB será solicitado aos alunos para que se dividam em duplas e após a divisão serão organizados em um computador para cada dupla. Cada dupla receberá uma folha

com as quatro questões referentes. Após a conclusão o professor recolherá os códigos fontes das duplas para analisar. O tempo para a execução dessa atividade será de 1 hora e 30 minutos.

#### 5.2.4 Teste Teórico

O teste teórico verifica o conhecimento do aluno, individualmente, em relação ao conteúdo estudado na disciplina. Nesse dia os alunos não receberão recurso estrela, segundo descrição da seção 5.5 desse documento. Porém, os alunos pontuarão segundo as seguintes regras para cada questão: (i) 25 pontos pela questão concluída; (ii) Somados a de 0 a 50 pontos atribuídos pela análise do código feito pelo professor. O teste teórico conterá quatro questões (Apêndice F) cada uma com o valor máximo de 75 pontos, portando o nível máximo de pontuação do aluno será de 300 pontos. E, como estamos em um processo *gamificado*, o aluno ganhará um *avatar* pelo seu desempenho na prova segundo regras definidas na Quadro 7.

Quadro 7 – *Avatar* Teste Teórico

<i>Avatar</i>	Faixa de pontuação
<b>Analista</b>	270 – 300
<b>Programador</b>	230 – 269
<b>Aprendiz</b>	190 – 229
<b>Amador</b>	0 – 189

##### 5.2.4.1 Procedimentos Metodológicos

Para o teste teórico, os alunos farão a atividade individualmente e terão o tempo de 1 hora e 30 minutos para resolver quatro questões sobre o conteúdo estrutura de dados homogêneos.

#### 5.2.5 Aula de *Feedback*

A aula de *feedback* é uma abordagem prevista no *framework* de Elgrably e Oliveira (2018) e Santos e Oliveira (2018), em que os alunos serão questionados pelo professor sobre o processo de ensino e aprendizagem via *gamificação*. As perguntas desse dia se encontram no Apêndice G. Nessa ocasião, os alunos ganharam apenas recurso estrela, segundo regras descritas na seção 5.5 desse documento.

### 5.2.5.1 Procedimentos metodológicos

Na aula de feedback o professor instigará os alunos por meio de uma apresentação em Power point sobre as diversas atividades que aconteceram no *framework gamificado*, com o intuito de melhorar para uma próxima turma.

## 5.3 Mecânicas de bônus

A bonificação é um recurso em que o aluno ganhará por seu comportamento em sala de aula, após a contabilização de todos os bônus e penalidades que o aluno sofreu, esse recurso será transformado em estrela, segundo as regras descritas na seção 5.5. As regras que definem a bonificação do aluno estão descritas no Quadro 8.

Quadro 8 – Distribuição das bonificações

Nome da dimensão	Explicação da dimensão	Bônus por dimensão	Máxima bonificação por dimensão	Justificativa
<b>Presença</b>	O aluno estar presente pontualmente no horário da aula.	10 Bônus	10 Bônus	A presença é considerada importante para compreender o conteúdo
<b>Participação</b>	0 – Caso o aluno não tenha participado; 2 bônus a cada participação até no máximo 10 bônus de participação (Ex: Feito algum comentário para o professor, respondido a uma pergunta aberta do professor a turma).	2 Bônus	10 Bônus	A participação em sala de aula indica que o aluno está atento ao conteúdo
<b>Sugestão</b>	0 – Caso o aluno não tenha sugerido; 2 bônus a cada sugestão com até no máximo 10 bônus de sugestão.	2 Bônus	10 Bônus	A sugestão mostra que o aluno está buscando o conceito além da sala de aula.
<b>Pergunta</b>	0 – Caso o aluno não tenha perguntado; 2 – bônus a cada pergunta com até no máximo 10 bônus de pergunta	2 Bônus	10 Bônus	A pergunta caracteriza o interesse pelo assunto abordado.
<b>Colaboração</b>	0 – Caso o aluno não tenha ajudado algum colega; 2 bônus a cada ajuda ao colega na resolução das atividades proposta com até no máximo 10 bônus de colaboração.	2 Bônus	10 Bônus	A colaboração nutre no aluno o sentimento de auxílio ao colega.

## 5.4 Mecânicas de penalidades

A penalidade é uma infração cometida pelo aluno a qual será penalizada com a perda de bonificação. As regras que definem a bonificação do aluno estão descritas no Quadro 9.

Quadro 9 – Distribuição das penalidades

Nome da dimensão	Explicação da dimensão	Perda de bônus por dimensão	Justificativa
<b>Falta</b>	-10 pontos caso falte a disciplina	-10	A falta caracteriza a perda do conteúdo passado no dia.
<b>Atraso em 10 minutos</b>	O aluno perderá 2 bônus a cada atraso. Podendo ter a penalidade aplicada a no máximo 4 atrasos, totalizando perda de 8 bônus.	-2	O atraso em 10 minutos identifica o desinteresse do aluno pela disciplina
<b>Uso do celular</b>	A cada uso do celular em horário de aula o aluno perderá 2 bônus	-2	O uso do celular indica a não atenção a atividade ou dinâmica desenvolvida em sala
<b>Uso do computador</b>	Utilizar o computador em sites e ferramentas não previstas nos procedimentos metodológicos	-2	O uso do computador indica a não atenção a atividade ou dinâmica desenvolvida em sala
<b>Penalidade por atrapalhar a aula</b>	Quando o aluno interromper a aula com brincadeiras ou não permitir que outro aluno assista a aula perderá 2 bônus.	-2	Atrapalhar a aula indica a não atenção a atividade também pode atrapalhar o desempenho dos outros alunos

## 5.5 Conversão de bônus para estrela

Após os alunos bonificarem e sofrerem as penalidades, diariamente receberam o recurso estrela exceto nos casos já previstos. As regras para concessão do recurso estrela estão descritas no Quadro 10.

Quadro 10 – Conversão de bônus para recurso estrela

Recurso	Descrição	Intervalo de bônus	Quantidade
<b>Estrela</b>	O recurso estrela é atribuído ao jogador conforme a sua bonificação	34 – 50	3
		17 – 34	2
		5 – 17	1

## 5.6 Conversão da pontuação para nota na disciplina

A pontuação final do jogo ocorrerá mediante a transformação do recurso estrela em pontos, o qual acontece segundo Quadro 11. E, a transformação do processo *gamificado* em nota será o número de pontos dividido por 100.

Quadro 11 – Conversão das estrelas para pontuação

Faixa de pontuação	Nota extra
15 – 10	100
9 – 5	50
4 – 0	0

## 5.7 Medalhas da *gamificação* e premiação

A concessão de medalha para a *gamificação* acontecerá no final das aulas e poderá ser acompanhada pelos discentes por meio da aba Medalhas. Para a *gamificação* da nossa sala de aula, teremos dois tipos de medalha: Power, referente as pontuações adquiridas no decorrer do processo e a Participativo, em relação ao recurso estrela adquirido. Para cada medalha teremos três posições, são elas: 1º lugar, ouro; 2º lugar, prata; 3º lugar, bronze. O critério de desempate para a medalha Participativo será o número total de bonificações sem a conversão para recurso estrela.

Os primeiros lugares, tanto na medalha Power quanto na medalha Participativo, ganharão o seguinte prêmio: um curso online pela plataforma Udemy.

## 6 REFERÊNCIA BIBLIOGRÁFICA BÁSICA

CARVALHO, S. E.R. **Introdução à Programação com PASCAL**. 4.ed. Rio de Janeiro: Campus, 1986.

DETERDING, S.; DIXON, D.; KHALED, R.; NACKE, L. From game design elements to gamefulness: defining gamification. In: **Proceedings of the 15th international academic MindTrek conference: Envisioning future media environments**. ACM, 2011. p. 9-15.

EIGRABLY, I. S.; OLIVEIRA, S. R. B. "Gamification and Evaluation of the Use the Agile Tests in Software Quality Subjects: the Application of Experiments". **13th ENASE - Evaluation of Novel Approaches to Software Engineering**. Madeira, Portugal, 2018.

EVARISTO, Jaime. Programando com Pascal. **2ª Edição**. Ed. Book Express, 2004.

LUZ, R. B.; NETO, A. Using Programming Dojos for Test-Driven Development Teaching. **XXIII Brazilian Symposium on Informatics in Education (SBIE 2012)**. Rio de Janeiro. p.25-35, 26, 2012.



SANTOS, E. D.; OLIVEIRA, S. R. B. Gamificação como Ferramenta de Suporte de Apoio ao Ensino da Técnica Análise de Pontos por Função em um Turma de Pós-Graduação: Um Estudo de Caso. **15th CONTECSI - International Conference on Information Systems and Technology Management**. São Paulo, Brasil. 2018.

## 7 REFERÊNCIA BIBLIOGRÁFICA COMPLEMENTAR

MEDINA, Marco; FERTING, Cristina. **Algoritmos e programação: teoria e prática**. São Paulo: Novatec Editora, 2006.

### APÊNDICE A - LISTA DE EXERCÍCIO 01

Conteúdo: Estrutura de dados homogêneos (vetor)

Observação:

Lista disponível para os alunos no dia <<Informe o dia da prática>>

A lista precisa ser feita em aula para os alunos ganharem pontuação

1) Desenvolva um algoritmo que leia do usuário um vetor de 10 posições e calcule e exiba o número armazenado mais o índice da posição que ele está armazenado.

Condição para concessão da pontuação:

Inserir no programa: 1 2 3 4 5 6 7 8 9 10

O programa deve apresentar: 1 1 2 2 3 3 4 4 5 5 6 6 7 7 8 8 9 9 10 10

2) Desenvolva um algoritmo em pascal que leia um vetor de 7 posições e calcule e exiba primeiro os números armazenados com índice ímpar e depois em índice par.

Condição para concessão da pontuação:

Inserir no programa: 1 2 3 4 5 6 7

O programa deve apresentar: 1 3 5 7

2 4 6

3) Desenvolva um algoritmo em pascal que leia dois vetores de 5 posições, calcule um vetor com a soma dos elementos em índices iguais e apresente ao usuário.

Condição para concessão da pontuação:

Inserir no programa: 1 2 3 4 5

6 7 8 9 10

O programa deve apresentar: 7 9 11 13 15

4) Desenvolva um algoritmo em pascal que leia os 12 meses do ano e, à medida que o usuário digitar um número, informe o mês correspondente; quando o usuário digitar 0, finalize o programa, caso digite outro número informe mês inválido.

Condição para concessão da pontuação:

Inserir no programa: 1/ 3 /5/ 0

O programa deve apresentar: Janeiro/ Março/ Maio

## **APÊNDICE B - LISTA DE EXERCÍCIO 02**

Conteúdo: Estrutura de dados homogêneos (matriz)

Observação:

Lista disponível para os alunos no dia <<Informe o dia da prática>>

A lista precisa ser feita em aula para os alunos ganharem pontuação

1) Desenvolva um algoritmo que leia do usuário um conjunto de números de uma matriz quadrada de ordem 3 e apresente ao usuário os números armazenados. Regra de constituição de matriz quadrada número de colunas = número de linhas.

Condição para concessão da pontuação:

Inserir no programa: 1 2 3 4 5 6 7 8 9

O programa deve apresentar: 1 2 3 4 5 6 7 8 9

2) Desenvolva um algoritmo que calcule e exiba o traço e a diagonal da matriz informada na primeira questão. A regra do cálculo do traço da matriz é a soma dos elementos da diagonal principal dela. A diagonal principal é formada pelos elementos cujo índice de colunas é igual ao índice de linhas.

Condição para concessão da pontuação:

Inserir no programa: 1 2 3 4 5 6 7 8 9

O programa deve apresentar: 1 5 9

3) Desenvolva um algoritmo para armazenar e exibir a agenda do usuário, portanto o programa deve receber os seguintes dados pessoais dos contatos do usuário: nome, endereço, CEP, Bairro e Telefone. O tipo de dado que essa matriz deve receber é do tipo String. Considere que o usuário poderá ter apenas 7 contatos na agenda e o usuário informará na primeira interação todos os contatos.

Condição para concessão da pontuação:

Inserir no programa: José Rua 66073070 Guamá 9198888888/ Maria Travessa 66073070 São Braz 9197777777/ João Rua 66073070 Benguí 9193333333/ Jorge Avenida 66073070 Doca 9190000000/ Rebeca Travessa 66073070 Guamá 919355678 /Ezequiel Rua 66073070 Guamá 919252852/ Ester Rua 66073070 Marco 9198888888

O programa deve apresentar: José Rua 66073070 Guamá 9198888888/ Maria Travessa 66073070 São Braz 9197777777/ João Rua 66073070 Benguí 9193333333/ Jorge Avenida 66073070 Doca 9190000000/ Rebeca Travessa 66073070 Guamá 919355678/ Ezequiel Rua 66073070 Guamá 919252852/ Ester Rua 66073070 Marco 9198888888

4) Desenvolva um algoritmo em pascal que monte uma nota fiscal, para isso utilize a estrutura de matriz, na primeira coluna deve armazenar a quantidade de produto solicitado, na segunda linha o valor unitário e na terceira linha o valor total por produto. Considere que a nota fiscal pode ter 10 linhas e o usuário informará na primeira interação os 10 produtos.

Condição para concessão da pontuação:

Inserir no programa: 1 2/ 3 4/ 5 6/ 1 2/ 3 4/ 4 5/ 4 5/ 3 4/ 3 3/ 1 2/

O programa deve apresentar: 1 2 2 / 3 4 12/ 5 6 30/1 2 2/ 3 4 12/ 4 5 20/ 4 5 20/ 3 4 12/ 3 3 9/  
1 2 2

## **APÊNDICE C – LISTA DE EXERCÍCIO PARA CASA**

Conteúdo: Estrutura de dados homogêneos

Observação:

Lista disponível para os alunos no dia <<Informe o dia da prática>>

A lista será feita em casa para os alunos ganharem pontuação

1) Desenvolva um programa em Pascal que receba do usuário um vetor de 10 posições, multiplique os valores em índice ímpar por 10 e divida os valores em índice par por 10. Depois apresente ao usuário o vetor resultante.

2) Desenvolva um programa em Pascal que leia do usuário dois vetores de 5 posições do tipo inteiro, multiplique o primeiro pelo inverso do segundo e apresente um terceiro vetor com o resultado para o usuário.

3) Elabore um programa em Pascal que leia dois vetores com 5 elementos do tipo inteiro, cada. Intercale os elementos dos dois vetores, formando um novo vetor com 10 elementos.

EX:

6	1	8
---	---	---

3	4	7
---	---	---

6	3	1	4	8	7
---	---	---	---	---	---

4) Criar um programa em Pascal que, dado um vetor de 5 elementos do tipo inteiro, crie um outro que seja o inverso do primeiro e apresente os dois vetores para o usuário.

(Ex.: vet1 → 1 2 3 4 5; vet2 → 5 4 3 2 1)

5) Escrever um programa em Pascal que lê um vetor N com 20 elementos do tipo inteiro e o escreve. Em seguida, troque o 1º elemento com o último, o 2º com o penúltimo até o 10º com o 11º e escreva o vetor N assim modificado.

6) Desenvolva um programa em Pascal que implemente uma matriz 3x3, receba do usuário números reais, coloque os números na matriz e depois apresente para o usuário apenas os elementos da diagonal principal.

7) Elabore um programa em Pascal que crie uma matriz 7X8 em que cada elemento é a soma dos índices de sua posição dentro da matriz. Ao final, apresente para o usuário a matriz gerada em formato matricial.

- 8) Criar um programa em Pascal que possa armazenar as alturas de dez atletas de cinco delegações que participarão dos jogos de verão. Imprimir a maior altura de cada delegação.
- 9) Elabore um algoritmo em pascal que receba números inteiros informados pelo usuário, armazene em uma matriz 3x3 e identifique se existem números repetidos, quantas vezes se repetem e apresente ao usuário.
- 10) Faça um programa que leia uma matriz 10X10, calcule e escreva a soma dos elementos que estão na área marcada com '\*'.

*									
*	*								
*	*	*							
*	*	*	*						
*	*	*	*	*					
*	*	*	*	*	*				
*	*	*	*	*	*	*			
*	*	*	*	*	*	*	*		
*	*	*	*	*	*	*	*	*	
*	*	*	*	*	*	*	*	*	*

## APÊNDICE D – LISTA DE EXERCÍCIOS PARA O DOJO *RANDORI*

Conteúdo: Estrutura de dados homogêneos

Observação:

Lista disponível para os alunos no dia <<Informe o primeiro dia da prática>>

### Atividade 01

A empresa de desenvolvimento Açai Tecnologia foi contratada para desenvolver um programa em Pascal para uma empresa de ônibus. O programa deve ser capaz: de reservar, pagar e cancelar reserva no Ônibus; o assento reservado guarda o valor do RG da pessoa; caso o assento já tenha sido reservado, informar que o assento não está disponível; exibir para o usuário os assentos disponíveis em formato de matriz; exibir os assentos já ocupados; informar se o ônibus já está lotado. Para fins de simplificação considere que o ônibus possui 60 lugares, com fileiras de 4 lugares disponíveis.

Lista de exercício para o segundo dia <<Informe o segundo dia da prática>>

### Atividade 01

Uma empresa foi contratada para criar um programa para uma loja. O programa tem que ser capaz de: incluir e excluir as vendas de cada semana dos meses do ano; alterar as vendas de uma determinada semana de um mês específico; exibir o total vendido em cada mês do ano; exibir o total vendido no ano e, ainda, em que mês(es) a loja alcançou o mínimo desejado para vendas (R\$ 5.000,00). Para fins de simplificação, considere que cada mês do ano possui somente 04 semanas

### APÊNDICE E – ATIVIDADE LABORATÓRIO DE PROGRAMAÇÃO

Conteúdo: Estrutura de dados homogêneos

Observação:

Atividade disponível para os alunos no dia <<Informe o dia da prática>>

A atividade será realizada em dupla

Não será liberada consulta a internet, aos materiais disponibilizados e nem a dupla ao lado

1) Desenvolva um algoritmo em Pascal que contenha a lista de preços para um leilão *online*. O programa recebe todos os lances e apresenta para o usuário o lance com preço maior, menor e a média dos lances que foram realizados. O vetor receberá 15 lances.

2) Elabore um programa em Pascal que armazene os 40 primeiros números primos em um vetor de inteiros. Depois apresente lado a lado os números de índice ímpar e os números de índice par. Ex:

Ímpar		Par
2		3
5		7

3) Elabore um algoritmo em pascal que leia do usuário números inteiros e os armazene em uma matriz de ordem 4, depois calcule o traço da diagonal principal. Limpe a tela do *prompt* e apresente ao usuário os números que estão acima da diagonal principal e que são maiores que o traço, em formato horizontal separado por “|”.

4) Sua empresa foi contratada por uma rede de cinemas para gerar um programa para reservar as cadeiras para uma sessão. Como bom desenvolvedor verificou que as filas começam com a

numeração 1 e vão até a 12, também observou que em cada fila da sessão tem 12 poltronas, logo, uma matriz de ordem 12.

Desenvolva um algoritmo em pascal que realize a “reserva” de lugares (informando a fileira e a cadeira), com o nome do cliente, caso a cadeira já tenha sido reservada, exibir uma mensagem para o cliente, exibir em formato matricial as cadeiras disponíveis, permitir uma opção para o cliente, verificar se a sessão já está cheia, e ao digitar a palavra “fim” o programa finaliza.

## **APÊNDICE F – TESTE TEÓRICO**

Conteúdo: Estrutura de dados homogêneos

Observação:

Atividade disponível para os alunos no dia <<Informe o dia da prática>>

A atividade será realizada individualmente.

Não será liberada consulta a internet, aos materiais disponibilizados e nem o colega ao lado.

O aluno realizará a atividade no papel com caneta, sem o auxílio de computador.

1) Faça um programa em Pascal que leia um vetor de 5 posições de números inteiros, multiplique todos os elementos pelo maior valor do vetor e apresente o novo vetor.

2) Faça um programa em Pascal que leia um vetor de seis elementos numéricos inteiros, calcule e mostre:

- a) A quantidade de números pares
- b) Quais os números pares
- c) A quantidade de números ímpares
- d) Quais os números ímpares

3) Desenvolva um algoritmo em Pascal que leia os valores de uma matriz de ordem 3 de inteiros, calcule e apresente a média dos números digitados, assim como a matriz.

4) Desenvolva um algoritmo em Pascal que leia os valores de uma matriz de ordem 4 e apresente todos os elementos da diagonal secundária, bem como a soma deles.

## APÊNDICE G – ATIVIDADE FEEDBACK

Data da atividade: <<Informe o dia da prática>>

As questões que serão feitas a turma para análise do *framework gamificado* foram divididas em categorias e seguirão a ordem abaixo:

### **Feedback sobre as metodologias de ensino escolhidas:**

- Foram positivas para o aprendizado? Por quê?
- Seria necessário mais tempo de aula teórica e prática? Por quê?
- Necessita de mais tempo na aula de Dojo *Randori*? Por quê?
- Qual a melhor forma do Dojo na sua opinião Tipo 01 ou Tipo 02? Por quê?
- Necessitaria de mais tempo na aula prática LAB (Laboratório de Programação)? Por quê?
- Você acha que seria necessário algum pré-requisito para um melhor aproveitamento dos alunos nas metodologias escolhidas para o ensino? Quais?

### **Feedback sobre a aula teórica e prática:**

- A aula expositiva e prática foi satisfatória para o aprendizado de Estrutura de dados homogêneos?
- Qual dificuldade vocês acharam para as listas de exercício?
- Vocês consideram válida a experiência do fluxo de ensino:
- Metáfora -> conceito -> A implementação na linguagem Pascal -> Um exercício exemplo
- O fluxo de ensino foi satisfatório ou gostariam de outra forma de ensino?

### **Feedback sobre a aula de Dojo *Randori*:**

- A prática do Dojo foi considerada satisfatória?
- Existia a necessidade de mais tempo para a explicação sobre as regras e o funcionamento do Dojo?
- Qual dificuldade vocês acharam da atividade?
- Vocês acham que o tempo destinado para atividade foi apropriado? Ou queriam mais tempo dessa atividade?



- Vocês têm alguma sugestão para melhoria da prática do Dojo?

**Feedback sobre a prática de LAB:**

- A prática do LAB foi considerada satisfatória?
- O tempo que foi disponibilizado para a prática foi suficiente?
- Qual a dificuldade encontrada na atividade?
- Fazer a técnica utilizando pareamento, foi positivo?
- Vocês têm alguma sugestão para melhoria da prática do LAB?

**Feedback sobre a lista de exercício feita em casa:**

- Você achou interessante ter uma lista de exercício para ser desenvolvida em casa?
- Qual a dificuldade vocês acham para a lista de exercício?

**Feedback sobre a importância de Estrutura de dados homogêneos:**

- Houve interesse em se aprofundar em estrutura de dados homogêneos?
- A utilização de Dojo e LAB, facilitou o entendimento da importância da prática da implementação e análise de Estrutura de dados homogêneos?
- Houve interesse dos alunos em utilizar a estrutura de dados homogêneos em projetos pessoais, no estágio ou no emprego?