

Universidade Federal de Campina Grande
Centro de Engenharia Elétrica e Computação
Coordenação de Pós-Graduação em Ciência da Computação

Tese de Doutorado

Personalizando Atividades para Estimular o
Engajamento de Aprendizes de Programação

Mirna Carelli Oliveira Maia

Campina Grande, Paraíba, Brazil

08/05/2020

Universidade Federal de Campina Grande
Centro de Engenharia Elétrica e Informática
Coordenação de Pós-Graduação em Ciência da Computação

Personalizando Atividades para Estimular o Engajamento de Aprendizes de Programação

Mirna Carelli Oliveira Maia

Tese submetida à Coordenação do Curso de Pós-Graduação em Ciência da Computação da Universidade Federal de Campina Grande - Campus I como parte dos requisitos necessários para obtenção do grau de Doutor em Ciência da Computação.

Área de Concentração: Ciência da Computação

Linha de Pesquisa: Educação e Computação

Dr. Dalton Dario Serey Guerrero

Dr. Jorge Cesar Abrantes de Figueiredo

Campina Grande, Paraíba, Brasil

©Mirna Carelli Oliveira Maia, 08/05/2020

M217p Maia, Mirna Carelli Oliveira.
Personalizando atividades para estimular o engajamento de aprendizes de programação / Mirna Carelli Oliveira Maia. – Campina Grande, 2020.
140 f. : il. color.

Tese (Doutorado em Ciência da Computação) – Universidade Federal de Campina Grande, Centro de Engenharia Elétrica e Informática, 2020.
"Orientação: Prof. Dr. Dalton Dario Serey Guerrero, Prof. Dr. Jorge Cesar Abrantes de Figueiredo".
Referências.

1. Computação – Estudo e Ensino. 2. Programação – Computação. 3. Engajamento do Estudante. 4. Aprendizagem Colaborativa. 5. Educação em Computação. I. Guerrero, Dalton Dario Serey. II. Figueiredo, Jorge Cesar Abrantes de. III. Título.

CDU 004.43(07)(043)

**PERSONALIZANDO ATIVIDADES PARA ESTIMULAR O ENGAJAMENTO DE
APRENDIZES DE PROGRAMAÇÃO**

MIRNA CARELLI OLIVEIRA MAIA

TESE APROVADA EM 08/05/2020

DALTON DARIO SEREY GUERRERO, Dr., UFCG
Orientador(a)

JORGE CESAR ABRANTES DE FIGUEIREDO, Dr., UFCG
Orientador(a)

WILKERSON DE LUCENA ANDRADE, Dr., UFCG
Examinador(a)

ELIANE CRISTINA DE ARAÚJO, Dra., UFCG
Examinador(a)

SEAN WOLFGAND MATSUI SIQUEIRA, Dr., UNIRIO
Examinador(a)

ROBERTO ALMEIDA BITTENCOURT, Dr., UEFS
Examinador(a)

CAMPINA GRANDE - PB

*Painho, eu daria tudo para ver o seu sorriso
de orgulho com essa minha vitória.*

Agradecimentos

Se pudesse, eu trocava a seção de agradecimentos por uma que contasse a "história por trás da tese". Nela, eu colocaria que minha vida virou de cabeça para baixo durante a realização desta pesquisa. Logo no primeiro ano, perdi a visão do olho direito e fui surpreendida com um diagnóstico difícil de esclerose múltipla que transformou como eu iria enfrentar a vida dali por diante. Concluir um doutorado diante de tantas dúvidas, sintomas e efeitos colaterais se tornara praticamente uma missão impossível. Também colocaria que meses depois, me deparei com um novo diagnóstico, dessa vez de painho. Meu mundo acabou. Não teria como seguir em frente sem ele ao meu lado vibrando por cada conquista. E lá se foi o meu maior fã, a minha maior torcida e minha grande inspiração para seguir sendo pesquisadora e professora. Nessa seção, relataria as pausas que tive que fazer, seja para internações, para viagens para tratar a doença ou até por causa de uma greve dos caminhoneiros que parou o país. Finalizaria com a maior e mais crítica de todas as pausas: uma pandemia. Esse inimigo me prendeu em casa, transformando a banca examinadora e a plateia da defesa em "quadrinhos" na tela de meu computador. Infelizmente, as normas da ABNT não permitiriam incluir esse tipo de seção, então deixo registrado aqui apenas meus agradecimentos ;).

Em primeiro lugar, agradeço a Deus, por ter me dado força para superar todos os desafios e me transformado em quem sou hoje. Seguido por meus pais: Painho, te agradeço por ser minha inspiração, por sempre estimular meu lado cientista, por ouvir meus conflitos, por me orientar em todas as minhas decisões, por vibrar por cada vitória e por ter deixado tantas boas memórias; Mainha, obrigada por ser meu suporte, por ficar feliz com minhas conquistas (mesmo sem entender exatamente do que se trata), por cuidar tão bem de mim e de Benjamin e permitir que eu pudesse voar.

Benjamin, eu sei que não é fácil entender como mamãe estuda tanto pra ser doutora mas mesmo assim te explica que não vai ser médica. Um dia você vai se orgulhar de mim. Obrigada por ser esse menino doce, inteligente e que me faz ser a pessoa mais feliz do mundo. Agradeço a Ronaldo por ser meu companheiro, me ajudar a colocar o pé no chão, me acalmar nos momentos difíceis e aguentar todos os meus choros. Mikaela, te agradeço por estar sempre presente em todas as lembranças da minha vida. Amo vocês com todas as minhas forças.

Agradeço a Ana Liz, Eliane e Thaciana, amigas que o doutorado me deu de presente e ajudaram a transformar essa fase em uma jornada mais leve e divertida e menos solitária. Também reconheço a importância dos meus colegas de Splab, especialmente aos companheiros que deixaram boas lembranças das conversas que aconteceram na sala 105.

Meu agradecimento aos *profs* Dalton e Jorge por acreditarem em meu potencial, pela orientação e as valiosas discussões que tivemos durante todos esses anos.

Por fim, mas não menos importante, agradeço a todos os alunos que participaram dessa pesquisa. Aqui tem um pedacinho de cada um de vocês.

Resumo

O processo de ensino-aprendizagem de programação envolve diversos fatores, como estratégias específicas de ensino e avaliação, além daqueles que estão relacionados ao perfil dos estudantes. Nos últimos anos, esse último fator tem se destacado por compreender aspectos importantes nesse processo, como o comportamento e as atitudes dentro e fora da sala de aula, esforço cognitivo e algumas sensações vivenciadas.

Os estudantes de cursos de Computação são considerados pouco engajados quando comparados aos de outros cursos. Essa conclusão quando se compara o nível de engajamento dos estudantes utilizando o instrumento *National Survey of Student Engagement*, que é o mais consolidado na literatura da área. Nessa corrente de pesquisa, a principal crítica dos especialistas diz respeito à falta de adequação dos indicadores de engajamento utilizados pelo instrumento em relação à natureza do curso de computação. Considerando que cursos de Introdução à Programação são apontados como a causa de níveis significativos de ansiedade e frustração, alguns especialistas criticam a escassez de estudos voltados a relacionar o aspecto emocional ao engajamento. Isso se deve, certamente, ao interesse mais expressivo em analisar o que os estudantes fazem do que compreender as respostas afetivas relacionadas ao processo de aprendizagem. Aos poucos, estratégias para estimular os alunos a desenvolver atitudes mais positivas tem recebido considerável atenção na literatura.

Nessa pesquisa, foram investigadas as principais dificuldades encontradas para estimular e mensurar o engajamento de estudantes de computação em atividades referentes à programação. Para isso, foi desenvolvido um modelo de personalização de atividades e a definição de métricas capazes de representar o engajamento de aprendizes de programação. Contiguamente, foram realizados os estudos qualitativos, experimentais e de caso, para compreender os efeitos da aplicação da proposta de personalização no engajamento. Os resultados obtidos indicam que a aplicação de atividades personalizadas influencia no engajamento emocional, acrescentando a presença de emoções positivas, como as vinculadas ao poder, a tranquilidade, a produtividade, e reduzindo a sensação de emoções negativas, como a tristeza e o espanto. Ao passo que não foi possível observar a influência do método nos aspectos relacionados a cognição e comportamento do engajamento quanto às métricas propostas.

Abstract

Programming learning involves several factors, including teaching and assessment strategies, and those related to students, such as engagement. In recent years, engagement has stood out for involving essential aspects, such as behavior and attitudes inside and outside the classroom, cognitive effort, and sensations experienced in the programming learning process.

Computer course students have low engagement if they are compared the level of student engagement using the *National Survey of Student Engagement* instrument of other students. About this, the specialists' criticism is about the adequacy of the engagement indicators used by the tool to the nature of the computing course. Considering that introductory programming courses are identified as the cause of significant levels of anxiety and frustration, some experts criticize the lack of emphasis on studies on the emotional aspect of student engagement. This is because there is a greater interest in analyzing what students do, instead of understanding affective responses to the learning process. Gradually, strategies to help students develop more positive attitudes have received considerable attention in the literature.

In this research, we investigate the difficulty of stimulating and measuring student engagement in programming learning activities. For this, we developed a model for personalizing activities and defined metrics to represent the engagement of programming learners. We carry out qualitative experimental and case studies to understand the effects on student engagement of the application of the personalization proposal. Our results suggest that the application of personalized activities influences emotional engagement, improving positive emotions, such as those linked to power, tranquility, productivity, and reducing the feeling of negative emotions, such as sadness and astonishment. While we are unable to perceive influence on the cognition and behavioral aspects of engagement when implemented with the proposed metrics.

Sumário

1	Introdução	1
1.1	Declaração do Problema	2
1.2	Perguntas de Pesquisa	3
1.3	Visão Geral	4
1.4	Contribuições	5
1.5	Estrutura do Documento	5
2	Referencial Teórico	7
2.1	Engajamento de Estudantes	7
2.1.1	Definição	8
2.1.2	Engajamento Comportamental	10
2.1.3	Engajamento Cognitivo	11
2.1.4	Engajamento Emocional	12
2.1.5	Instrumentos para Medição de Engajamento	13
2.2	Pedagogia da Contribuição do Estudante	14
2.2.1	Características	15
2.2.2	Exemplos de Atividades	17
2.3	Conclusão	18
3	Modelo de Personalização de Atividades de Programação	20
3.1	Visão Geral	20
3.2	Atividade Personalizada de Programação	23
3.3	Atribuição de Papéis	25
3.4	Tipos de Atividades Personalizadas	26

3.4.1	Criação de Novo Exercício	27
3.4.2	Resolução de Exercício Produzido por Estudante	29
3.4.3	Sugestão de Casos de Teste	29
3.4.4	Revisão de Código	29
3.4.5	Revisão de Uma Atividade Existente	30
3.5	Conclusão	32
4	Métricas de Engajamento em Atividades de Programação	33
4.1	Engajamento Comportamental	33
4.2	Engajamento Cognitivo	35
4.3	Engajamento Emocional	38
4.4	Conclusão	39
5	Avaliação do Engajamento de Aprendizes de Programação	41
5.1	Ambiente Experimental	42
5.1.1	<i>Design</i> da Disciplina	42
5.1.2	Ferramentas	44
5.2	Engajamento do Colaborador na Criação de Exercício	44
5.2.1	Materiais e Métodos	45
5.2.2	Resultados	51
5.3	Engajamento do Colaborador na Realização de Atividades Personalizadas	56
5.3.1	Materiais e Métodos	57
5.3.2	Resultados	62
5.4	Aplicação das Métricas de Engajamento de Aprendizes de Programação	71
5.4.1	Materiais e Método	74
5.4.2	Resultados	77
5.5	Conclusão	83
6	Discussão	87
6.1	Desafios da Pesquisa	87
6.2	Implicações	88
6.3	Efeito no Engajamento de Estudantes	90

6.4	<i>Feedback</i> dos Estudantes	91
6.5	Trabalhos Relacionados	93
6.5.1	Engajamento de Estudantes	93
6.5.2	Pedagogia de Contribuição do Estudante	94
6.6	Conclusão	96
7	Conclusão	97
7.1	Trabalhos Futuros	99
A	Parecer do Comitê de Ética	106
B	<i>Online Student Engagement: A Case Study in Teaching of Programming</i>	110
C	Questionário do Grupo Controle	121
D	Questionário do Grupo Experimental	123
E	Atividades Personalizadas: Grupo Controle	125
F	Atividades Personalizadas: Grupo Experimental	130
G	Feedback dos Estudantes	135
H	<i>Learning Styles in Programming Education: a Systematic Mapping Study</i>	140

Lista de Figuras

1.1	Visão Geral da Solução	4
3.1	Cenário 1: Ciclo que compreende o processo de compartilhamento da dificuldade de aprendizagem pelo aluno e a elaboração de uma nova atividade personalizada pelo colaborador	21
3.2	Cenário 2: Ciclo em que o professor percebe a dúvida que não foi compartilhada e gera uma meta-atividade, que é repassada ao colaborador	21
3.3	Cenário 3: Ciclo em que a dificuldade do aluno é identificada por meio de um Ambiente de Aprendizagem que acompanha o progresso do estudante	22
3.4	Visão Geral do Modelo de Personalização	23
3.5	Exemplo de Atividade Personalizada	24
3.6	Contexto de Atividade Personalizada	25
3.7	Meta-atividade de Criação de Atividades	28
3.8	Atividade Personalizada de Criação de Exercício	28
3.9	Atividade de Revisão de Código	30
3.10	Atividade Personalizada de Revisão de Texto	31
4.1	Instrumento Roda das Emoções	39
5.1	Estudos para Avaliação de Engajamento	42
5.2	Interação entre Alunos via Slack	45
5.3	Organização da Sala Durante o Experimento	48
5.4	Informação Compartilhada com os Participantes do Experimento I	49
5.5	Enunciado Exibido para o Grupo de Controle	49
5.6	Enunciado Exibido para o Grupo Experimental	50

5.7	Questionário Sobre a Intensidade de Sensações	50
5.8	Número de Exercícios Criados por Estudantes	51
5.9	Número de Exercícios Criados por Autor	52
5.10	Intervalo entre as Submissões de Questões	53
5.11	Dificuldade dos Exercícios Gerados pelo Grupo de Controle	53
5.12	Dificuldade dos Exercícios Gerados pelo Grupo Experimental	54
5.13	Como o Participante se Sentiu Envolvido, Estimulado, Incluído, Interessado e Motivado)	54
5.14	Como o Participante se Sentiu Admirado, Feliz, Orgulhoso e Satisfeito	55
5.15	Como o Participante se Sentiu Influyente, Produtivo e Útil	56
5.16	Como o Participante se Sentiu Cauteloso, Chateado e Envergonhado	57
5.17	Organização da Sala Durante o Experimento	60
5.18	Informação Compartilhada com os Participantes do Experimento II	61
5.19	Enunciado Exibido Para o Grupo Experimental no Segundo Experimento	62
5.20	Horário das Submissões	64
5.21	Horário das Submissões	64
5.22	As Emoções Relatadas pelo Grupo de Controle	66
5.23	As Emoções Relatadas pelo Grupo Experimental	66
5.24	Comparação entre as Emoções	67
5.25	Comentário do Grupo de Controle (1)	68
5.26	Comentário do Grupo de Controle (2)	69
5.27	Comentário do Grupo Experimental (1)	70
5.28	Comentário do Grupo Experimental (2)	71
5.29	Comentário do Grupo Experimental (3)	72
5.30	Comentário do Grupo Experimental (4)	73
5.31	Intervalo de Tempo Entre os Eventos	76
5.32	Duração das Sessões de Estudo	77
5.33	Tarefas Concluídas	78
5.34	Exercícios Resolvidos em Simulados	79
5.35	Hábito de Estudo	79
5.36	Ritmo de Aprendizagem	80

5.37 (a) Total de Questões Realizadas; (b) Tarefas de Casa	83
5.38 Número de Eventos	84
5.39 Duração das Sessões em Minutos	84
5.40 Questões em Simulados	85
5.41 Ritmo de Aprendizagem	85
G.1 Opinião Sobre Atividades Personalizadas (I)	136
G.2 Opinião Sobre Atividades Personalizadas (II)	137
G.3 Opinião Sobre Atividades Personalizadas (III)	138
G.4 Opinião Sobre Atividades Personalizadas (IV)	139

Lista de Tabelas

2.1	Exemplos de Engajamentos Positivos e Negativos (Trowler, 2010)	10
4.1	Indicadores de Engajamento Comportamental	34
4.2	Ritmo de Aprendizagem	37
5.1	Unidades de Conteúdo	43
5.2	Características dos Grupos de Controle e Experimental	51
5.3	Características dos Grupos de Controle e Experimental	61
5.4	Características dos Participantes	62
5.5	Questões Concluídas Durante o Experimento	63
5.6	Questões Concluídas por Tipo	63
5.7	Tamanho dos Exercícios	65
5.8	Correlação entre Engajamento e Desempenho	82
6.1	Comentários de Participantes	92

Capítulo 1

Introdução

A aprendizagem de programação envolve diversos fatores, incluindo as estratégias de ensino e avaliação, e os que são relacionados diretamente aos estudantes. Nos últimos anos, o engajamento tem se destacado por compreender aspectos importantes, como o comportamento, cognição e emoção no processo de aprendizagem de programação (Farrington et al. 2012, Vihavainen, Airaksinen e Watson 2014, Luxton-Reilly et al. 2018).

De forma geral, educadores usam o termo engajamento para se referir a esforço e participação ou habilidades cognitivas como pensamento crítico, resolução de problemas, síntese e avaliação (Kuh 2001). No contexto educativo em programação, engajamento é comumente utilizado para relatar *comportamentos* que podem ser observados durante a experiência de aprendizagem do estudante (Fredricks e McColskey 2012, Christenson, Reschly e Wylie 2012, Luxton-Reilly et al. 2018).

Contudo, comportamento é apenas um dos aspectos do engajamento estudantil. Dentre suas definições, o engajamento é considerado como “a interação entre tempo, esforço e outros recursos relevantes investidos pelos alunos e suas instituições, com o objetivo de otimizar a experiência do aluno e melhorar os resultados de aprendizagem, desenvolvimento e desempenho dos alunos” (Trowler 2010).

O engajamento estudantil é um meta-constructo que inclui dimensões com aspectos comportamentais, cognitivos e emocionais (Fredricks, Blumenfeld e Paris 2004, Christenson, Reschly e Wylie 2012, Linnenbrink e Pintrich 2003), em que o engajamento comportamental está relacionado com o envolvimento em atividades acadêmicas, sociais ou extracurriculares e boa conduta (Fredricks, Blumenfeld e Paris 2004). O engajamento cog-

nitivo é associado ao esforço do aluno na aprendizagem e sua habilidade de auto-regulação. Enquanto isso, o engajamento emocional relaciona reações, positivas ou negativas, com os professores, colegas de classe e com a instituição e que podem influenciar para que os alunos mantenham vínculo com a instituição (Fredricks e McColskey 2012).

As três dimensões do engajamento são fundamentais e complementares. Nesta pesquisa de doutorado, definimos engajamento ou envolvimento do estudante, como sendo:

Todos os aspectos que influenciam a aprendizagem de programação do aprendiz, incluindo as atitudes dentro e fora da sala de aula, esforço cognitivo e sensações vivenciadas ao estudar programação.

Engajamento pode ser estimado por auto-relato do aluno, observação ou relato do professor (Fredricks e McColskey 2012). Dentre os instrumentos de medição, o questionário de auto-relato *National Survey of Student Engagement* (NSSE) é o mais conhecido (Kuh 2003). Ao estimar pelo NSSE e comparar o nível de engajamento dos estudantes de programação com de outros cursos, especialistas concluíram que estudantes de computação são os menos engajados (Sinclair et al. 2015, Butler et al. 2016, Morgan et al. 2018, Morgan et al. 2018).

Outra crítica dos especialistas quanto ao engajamento dos estudantes aprendizes de programação é a pouca atenção a aspectos afetivos e emocionais (Scaico 2018, Luxton-Reilly et al. 2018). O que pode estar relacionado ao fato do aprendizado de programação ressaltar a importância da prática de exercícios de programação para bons resultados acadêmicos (Corney, Teague e Thomas 2010, Pears 2010, Araujo et al. 2013).

Embora seja essencial conhecer o que os alunos fazem, é preciso também se aprofundar nos demais aspectos envolvidos no processo de aprendizagem, portanto a aplicação de práticas direcionadas a promover apenas uma dimensão do engajamento pode negligenciar aspectos igualmente importantes. De acordo com a literatura, mais de 75% dos estudos estão interessados apenas em conhecer o que os alunos podem fazer, bem como analisar os dados sobre as tarefas e avaliações com intenção de medir o seu desempenho (Sheard et al. 2009).

1.1 Declaração do Problema

Manter os alunos envolvidos e persistindo mesmo diante das dificuldades é um desafio para os educadores. Sobretudo porque o engajamento influencia na aprendizagem dos estudantes,

além de estar relacionado com os índices de abandono e reprovação (Finn 1989).

Considerando que cursos de introdução à programação são apontados como causas de níveis significativos de ansiedade e frustração, alguns especialistas criticam a pouca ênfase dos estudos no aspecto emocional do engajamento. Isso porque há maior interesse em analisar o que os estudantes fazem, em vez de compreender as respostas afetivas ao processo de aprendizagem. Aos poucos, estratégias para ajudar os alunos a desenvolver atitudes mais positivas têm recebido considerável atenção na literatura.

Esta pesquisa de doutorado aprofunda o conhecimento sobre o engajamento de estudantes aprendizes de programação, investigando como mensurar e principalmente como melhorá-lo, considerando suas dimensões de comportamento, cognição e emoção.

1.2 Perguntas de Pesquisa

O objetivo geral desta pesquisa consiste em investigar como mensurar e quais os efeitos no engajamento de aprendizes ao realizar atividades personalizadas de programação. Para atender esse objetivo, as seguintes perguntas motivadoras foram elencadas para o desenvolvimento do trabalho:

1. **Como mensurar engajamento em atividades de aprendizagem de programação?**
- Definimos métricas que indicam o engajamento de estudantes em atividades personalizadas de programação.
2. **Como estimular o engajamento de estudantes em atividades de aprendizagem de programação?** - Propusemos um modelo de personalização com o objetivo de melhorar o engajamento dos estudantes em atividades de programação.
3. **Quais os efeitos no engajamento de estudantes em atividades personalizadas de programação?** - Realizamos estudos experimentais quantitativos para investigar os efeitos da aplicação do modelo de personalização de atividades no engajamento de estudantes aprendizes de programação.

1.3 Visão Geral

Nessa tese de doutorado, apresentamos como estimular o engajamento de estudantes durante a aprendizagem de programação por meio de um modelo de personalização de atividades. Foi atribuído aos estudantes novos papéis e a partir disso foram definidos novos tipos de atividades de programação. Esse novo cenário de realização proporcionou a criação de novos indicadores para mensurar engajamento em atividades de programação.

A Figura 1.1 apresenta a visão geral da solução proposta, com os principais papéis que os estudantes podem assumir no modelo e como eles se relacionam. Caso o estudante esteja com dúvidas em algum conceito ou dificuldades em resolver um exercício da base de dados do exercício do curso de programação, um colaborador é informado sobre essa situação e recebe detalhes sobre o contexto em que o estudante demonstrou estar com dificuldade de aprendizagem. Idealmente, o colaborador é outro aluno aprendiz de programação, mas também pode ser algum monitor da disciplina ou até outro estudante que está em estágio mais avançado no curso. Para auxiliar o estudante com dificuldades, o colaborador irá desenvolver uma atividade personalizada levando em consideração o conteúdo que o estudante demonstrou não compreender efetivamente.

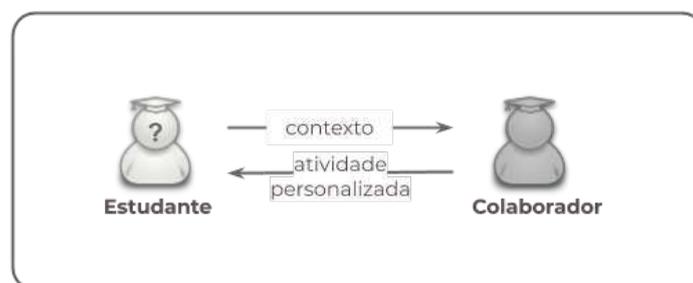


Figura 1.1: Visão Geral da Solução

A atividade personalizada trata-se de um exercício de programação que tem como objetivo ajudar o personagem com dificuldades a refletir sobre a resolução do problema descrito no exercício. Uma vantagem desse modelo de atividade é que a personalização se centraliza na situação real de dificuldade de um estudante em específico. A aplicação do modelo de personalização de atividades de programação permite a definição de novos indicadores de engajamento no contexto da programação com ênfase em seus três aspectos: cognitivo, comportamental e emocional. Com isso, foram definidas métricas para representar dedica-

ção, persistência, hábito de estudo e participação engajamento comportamental. A métrica ritmo de aprendizagem foi definida para representar o engajamento cognitivo e como o estudante de programação está progredindo ao longo do curso. Bem como estabelecemos o uso de um instrumento de auto-relato conhecido como “roda das emoções” para mensurar o engajamento emocional dos estudantes ao realizar atividades de programação.

1.4 Contribuições

Sumarizando, essa tese de doutorado estudou como mensurar e melhorar o engajamento de estudantes de programação por meio de atividades personalizadas.

Diante disso, as principais contribuições deste trabalho de doutorado são:

- Um modelo de personalização descrevendo como elaborar atividades de programação para estimular o engajamento de aprendizes;
- Métricas que indicam o engajamento comportamental, cognitivo e emocional dos estudantes aprendizes de programação;
- Um conjunto de lições aprendidas sobre a colaboração e o engajamento de aprendizes de programação.

Os resultados desta pesquisa inferiram um impacto positivo no engajamento emocional dos estudantes sem prejudicar as aspectos comportamentais e cognitivos. Vale destacar que as discussões e as lições aprendidas durante este trabalho de doutorado, bem como o modelo proposto estão servindo como inspiração para criação de novas estratégias de ensino adotadas pelos professores da disciplina de Programação utilizada como nosso ambiente de experimentação.

1.5 Estrutura do Documento

Esse documento está estruturado da seguinte forma:

Capítulo 2 - Referencial Teórico: Este Capítulo apresenta o referencial teórico que embasou o processo de pesquisa, desde os diversos conceitos que envolvem o engajamento até seus aspectos principais.

Capítulo 3 - Modelo de Personalização: Este Capítulo detalha o modelo proposto, bem como explica os componentes e tipos de atividade personalizada.

Capítulo 4 - Métricas de Engajamento em Atividades de Programação: Este Capítulo detalha as métricas que avaliam aspectos do comportamento, cognição e emoção do engajamento de estudantes aprendizes de programação em atividades personalizadas.

Capítulo 5 - Aplicação do Modelo de Personalização: Este Capítulo se concentra nos estudos sobre o engajamento de estudantes no âmbito de uma disciplina de programação introdutória, descreve o comportamento de aprendizes de programação em um ambiente online de aprendizagem, apresentando cenários de aplicação do modelo de personalização e concomitantemente analisa o seu impacto no engajamento dos estudantes.

Capítulo 6 - Discussão: Este Capítulo discute os resultados dos estudos conduzidos durante esta pesquisa de doutorado e apresenta as implicações práticas em aplicar o modelo de personalização. Como também apresenta a relação dessa pesquisa com outros trabalhos e ressalta algumas ameaças à validade.

Capítulo 7 - Conclusões: Este Capítulo sumariza os resultados e detalha seus pontos positivos e limitações. Além disso, sugere temas como sugestões de trabalhos futuros.

Capítulo 2

Referencial Teórico

Esse capítulo apresenta as diversas definições de engajamento encontradas na literatura, detalhando as dimensões de comportamento, cognição e emoção presentes em algumas destas. Apresenta os principais indicadores e métodos de medição de engajamento, exemplificando os instrumentos mais utilizados. Bem como, estabelece a definição utilizada ao longo deste trabalho.

Além disso, o capítulo introduz as características da abordagem conhecida como Aprendizagem ativa que é uma das intervenções para melhoria de engajamento de estudantes, destacando a Pedagogia de Contribuição do Estudante que é a metodologia de ensino que fundamenta essa pesquisa.

2.1 Engajamento de Estudantes

Os primeiros estudos relativos à compreensão do engajamento concentraram no contexto do ensino médio. A intenção dos especialistas era melhorar as conquistas, o comportamento e o sentimento de pertencimento, para que os estudantes pudessem permanecer vinculados à escola. O engajamento foi especialmente visto como estratégia para recuperar estudantes socioeconomicamente desfavorecidos que sinalizavam ter maiores riscos de abandono do ensino médio (Parsons e Taylor 2011).

Baixos índices de engajamento refletem nas taxas de abandono e reprovação (Finn 1989). Estudantes desengajados se recusam a participar de discussões em sala de aula ou ficam inquietos durante a realização de atividades. Professores podem até intervir para despertar

o interesse, mas o que acontece normalmente, é que a desistência aumenta a medida que se exige mais de estudantes que agem dessa forma (Finn e Zimmer 2012).

De acordo com a experiência dos professores, os alunos desistem de estudar por motivos diversos e tal realidade não se faz presente apenas no ensino médio.. Recentemente, o envolvimento recebeu atenção dos especialistas que visam aprimorar o aprendizado também no ensino superior (Trowler 2010). Uma vez que as instituições de ensino superior vêm enfrentando restrições de orçamento, estratégias para atrair e reter estudantes, garantindo que se formem para serem cidadãos bem-sucedidos e produtivos, são bem vindas. Portanto, o tema “engajamento estudantil” vem ganhando espaço em conferências e encontros mundiais da educação (Trowler 2010).

2.1.1 Definição

Mesmo com inúmeros especialistas estudando o termo “engajamento de estudantes” ao longo dos anos, ainda não tem um consenso sobre sua definição. Essa seção detalha algumas dessas definições e apresenta em qual delas esse trabalho se concentra.

De modo geral, o termo *engajamento* é usado para relacionar participação em atividades como na definição de (Kuh 2007) que sugere “envolvimento dos alunos como participação em práticas educacionalmente eficazes, dentro e fora da sala de aula, o que leva a uma série de resultados”.

Outras definições consideram engajamento como um processo que está associado a medidas de desempenho acadêmico pelo qual as instituições agregam valor aos seus alunos (Carini, Kuh e Klein 2006). Ou como os estudantes parecem estar interessados em seu aprendizado e sua conexão com a instituição e uns aos outros (Axelson e Flick 2010). Ou ainda “quantidade e qualidade da energia física e psicológica que estudantes investem na experiência escolar” (Axelson e Flick 2010). Segundo (Coates 2005), é como um constructo que abrange aspectos acadêmicos e não-acadêmicos, conforme participação em atividades acadêmicas, comunicação com colegas e professores, envolvimento em experiências educacionais e sentir-se integrante da comunidade universitária. Uma das definições mais utilizadas afirma que engajamento está concentrado na interação entre tempo, esforço e outros recursos relevantes investidos pelos estudantes e instituições, com o objetivo de otimizar a experiência do estudante e melhorar os resultados de aprendizagem e desenvolvimento dos

estudantes, o desempenho e a reputação da instituição (Trowler 2010).

Com relação ao número de aspectos importantes no conceito de engajamento, pode ser um modelo bidimensional, incluindo comportamento e emoção (Finn 1989). Como também pode ser um constructo multidimensional que mescla os aspectos comportamentais, cognitivos e de emoções de maneira significativa (Linnenbrink e Pintrich 2003, Fredricks, Blumenfeld e Paris 2004, Christenson, Reschly e Wylie 2012). Em que o aspecto comportamental representa o que é observável, como participação e conclusão de atividades acadêmicas. O engajamento cognitivo envolve o esforço necessário para compreender ideias complexas e desenvolver habilidades difíceis. Enquanto o engajamento emocional engloba reações ao relacionamento com professores, colegas e a instituição de ensino. Pode-se dividir também em aspecto comportamental acadêmico (realização de atividades), e comportamento (participação nas aulas e em atividades extracurriculares) (Grove e Education 2019).

Ainda é necessário aprofundar a discussão com trabalhos científicos, teóricos e empíricos para determinar até que ponto essas diferentes dimensões são construções únicas e, também, se um modelo de três ou quatro componentes descreve com mais precisão a construção do envolvimento do estudante. Contudo, optamos por estabelecer o nosso entendimento do termo “engajamento do estudante de programação”, com base em definições da literatura e em aspectos que julgamos relevantes para o contexto desse trabalho, como sendo:

Todos os recursos que influenciam a experiência de aprendizagem de programação, incluindo suas atitudes dentro e fora da sala de aula, esforço cognitivo e emoções sentidas ao estudar programação.

Trowler et. al (2010) sugerem que essas dimensões apresentam um polo *positivo* e *negativo*, bem como um terceiro representando o não envolvimento. Assim, pode-se envolver positivamente, negativamente ou não se envolver nas dimensões comportamentais, emocionais e cognitivas. Além disso, é possível que um estudante se envolva positivamente em uma ou mais dimensões, enquanto está negativamente envolvido em outra.

A Tabela 2.1 representa exemplos de engajamentos positivos, negativos e não engajados. Percebe-se que, a dimensão comportamental pode ser observada pela maneira como o estudante assiste às aulas. Estar presente, mas não prestar atenção indica que o mesmo está negativamente engajado. Enquanto, não estar presente sinaliza o não envolvimento com a

aula. Para a dimensão emocional, algumas sensações podem representar que o estudante está positivamente engajado como o interesse, em contrapartida o tédio representa que estão negativamente engajados. Nesta dimensão, o não engajamento seria sinalizado pelo sentimento de rejeição.

Tabela 2.1: Exemplos de Engajamentos Positivos e Negativos (Trowler, 2010)

Engajamento	Positivo	Negativo	Não engajado
Comportamental	Participa de aulas com entusiasmo	Ignora as aulas	Boicotes ou perturbação das aulas
Emocional	Interesse	Tédio	Rejeição
Cognitivo	Atende ou excede os requisitos da tarefa	Tarefas atrasadas, apressadas ou ausentes	Redefine parâmetros para tarefas

A literatura registra fatores que ajudam a perceber como o aprendiz está envolvido com o processo de aprendizagem e sinaliza quais os estudantes estão em risco de falha (Luxton-Reilly et al. 2018). Tais fatores ainda realçam a importância dos contextos - famílias, escolas, colegas e comunidades - para intervir e aprimorar o envolvimento dos estudantes (Christenson, Reschly e Wylie 2012).

Alguns dos indicadores de engajamento foram considerados como preditores de abandono (Rumberger e Lim 2008). Outros são características individuais do estudante e, dificilmente, são alteráveis como perfil socioeconômico, famílias monoparentais, entre outros. No entanto, outros indicadores podem ser alterados após uma intervenção efetiva (Christenson, Reschly e Wylie 2012).

As seções seguintes detalham as três dimensões do engajamento sobre as quais nos concentramos neste trabalho, apresentam suas características e descrevem como seus indicadores podem ser estimados.

2.1.2 Engajamento Comportamental

O comportamental é o aspecto observável do engajamento e representa como o estudante *age* durante o processo de aprendizagem. Inclui atitudes, como a participação em sala de aula e o número de atividades realizadas (Christenson, Reschly e Wylie 2012, Fredricks e McColskey 2012).

Pode ser considerado um preditor confiável do desempenho escolar. Comportamentos como frequência nas aulas e a conclusão de tarefas impactam diretamente no desempenho dos estudantes. Pesquisas relatam uma relação positiva entre engajamento comportamental e realização de tarefas (McWilliam, Trivette e Dunst 1985, Finn 1989, Farrington et al. 2012) e relação negativa com problemas de disciplina e desistência (Finn, Pannozzo e Voelkl 1995).

Alguns indicadores são relacionados ao envolvimento comportamental:

- **Frequência** - A presença do aluno, participação em aulas e demais atividades propostas são elementos críticos do processo de aprendizagem e podem ser levados em consideração na avaliação do desempenho acadêmico. As ausências excessivas tendem a afetar negativamente a capacidade de aprender (Farrington et al. 2012);
- **Responsabilidade** - o aprendizado é influenciado pela maneira como um indivíduo participa de atividades educativas (Coates 2005). A responsabilidade do estudante é fundamental para ter experiência de aprendizado bem-sucedida, aceitar a responsabilidade pelo seu progresso acadêmico, gerenciando seu tempo e preparação para a aula (Grove e Education 2019);
- **Abandono e Desistência** - A relação entre a desistência e abandono escolar com o baixo engajamento é um padrão encontrado em vários estudos da área. Isto porque, comportamentos considerados negativos, incluindo a recusa por participação em atividades e baixa frequência, acabam refletindo na desistência do estudante quando o nível de exigência aumenta (Finn e Zimmer 2012);

2.1.3 Engajamento Cognitivo

Engajamento cognitivo é o esforço consciente para compreender ideias, a fim de atingir requisitos de aprendizagem. Os estudantes podem estar comportamentalmente engajados, mas não cognitivamente. Isto porque, estudantes estão cognitivamente engajados quando usam estratégias de metacognição para analisar a aprendizagem e refletir sobre a tarefa que estão desempenhando (Linnenbrink e Pintrich 2003).

Comportamentos que indicam envolvimento cognitivo incluem fazer perguntas coerentes sobre o conceito que está sendo estudado, persistir tentando mesmo diante de tarefas

difíceis, ler mais do que o material sugerido, revisar material de conceitos já vistos e usar a autorregulação para orientar a aprendizagem (Fredricks e McColskey 2012).

O aprendizado exige esforço de qualidade que contribui para resultados desejáveis (Coates 2005). Quanto mais os estudantes gastam tempo de qualidade para estudar, mais eles aprendem. Para o ensino de programação, isso significa que, quanto mais o aprendiz pratica, mais ele aprende (Araujo et al. 2013). Sendo assim, exceder a realização de atividades pode sinalizar um forte indicador de envolvimento cognitivo.

A medição do envolvimento cognitivo é um pouco delicada. Nas técnicas de observação usadas nas salas de aula, é difícil detectar se os estudantes estão tentando realizar um trabalho para dominar o conteúdo ou para concluir uma tarefa. Também existe dificuldade na medição da metacognição de um estudante devido à falta de instrumentos confiáveis (Linnenbrink e Pintrich 2003).

2.1.4 Engajamento Emocional

O engajamento emocional examina os sentimentos e atitudes dos estudantes em relação à instituição de ensino, professores e colegas. Fornece incentivo afetivo para que os estudantes persistam nas atividades de aprendizagem (Christenson, Reschly e Wylie 2012, Fredricks e McColskey 2012). Também pode ser definido como o que as pessoas sentem quando se identificam e fazem um bom trabalho (Kahn 1990). Está relacionado a sentimentos de interesse, felicidade, ansiedade, raiva e frustração dos alunos durante a realização de atividades (Skinner e Belmont 1993).

Pesquisas nessa área utilizam questionários para identificar a maneira que os estudantes se sentem ao realizar atividades, e como se relacionam com o lugar onde estudam, com os professores e os colegas. O problema é que emoções relatadas pelos alunos podem ser atribuídas a uma variedade de fatores acadêmicos, e não exclusivamente para sua experiência de aprendizagem (Fredricks e McColskey 2012).

Experiências emocionais ruins podem afetar negativamente o envolvimento do estudante de uma forma geral. Por isso, para (Kahn 1990), as três dimensões do engajamento deviam ser organizadas em uma hierarquia, com o engajamento emocional como o nível “mais profundo”. Uma vez incorporado o aspecto afetivo, algumas pessoas podem confundir o conceito de engajamento com o de motivação. O engajamento do es-

tudante reflete a sua interação com um contexto ou tarefa, enquanto a motivação se refere a um determinado comportamento individual de alguém sem influência do ambiente no qual está inserido. Assim, diferente da motivação, especialistas sugerem que é possível alterar alguns aspectos do engajamento com a realização de intervenções eficientes (Fredricks, Blumenfeld e Paris 2004, Fredricks e McColskey 2012).

2.1.5 Instrumentos para Medição de Engajamento

A literatura registra cerca de 150 ferramentas para mensurar o nível de engajamento de um estudante. Essas ferramentas seguem cinco metodologias diferentes que estão descritas a seguir (Fredricks e McColskey 2012):

- **Auto-relato do estudante** - Nesse método, há um questionário com uma série de perguntas com o intuito de conhecer a perspectiva do estudante em relação a itens não observáveis do engajamento, principal aspectos emocionais e de cognição. O *National Survey of Student Engagement* (NSSE) é o questionário para auto-relato de estudantes mais consolidado na área de engajamento acadêmico (Fredricks e McColskey 2012). Pois tem como objetivo extrair informações para melhorar o ensino superior em inúmeras universidades ao redor do mundo (Kuh 2003). Suas perguntas alternam sobre o comportamento e perfil do estudante, e a interação com a instituição, colegas e professores (Kuh 2009). Instrumentos do tipo de auto-relato podem ser utilizados para conhecer aspectos comportamentais e emocionais do engajamento;
- **Relato do instrutor** - Nesse método, o instrutor responde a um questionário com uma série de perguntas relacionadas ao comportamento observável de cada estudante. É mais utilizado para medir o engajamento infantil porque esse perfil de estudantes pode apresentar dificuldades em responder os questionários de auto-relato pela limitação da habilidades de leitura e escrita (Fredricks e McColskey 2012). A mensuração do engajamento via relato do professor pode ser utilizada para identificar aspectos observáveis do comportamento do estudante;
- **Experimento por amostragem** - Com este método, os estudantes registram informações sobre o engajamento durante a realização do experimento. Essas informações de-

talham o que e como algo está sendo realizado em um determinado momento do experimento. Pode ser muito útil para conhecer o envolvimento do estudante durante a realização de atividades sugeridas para casa, por exemplo (Fredricks e McColskey 2012). A vantagem desse método é que se reduz o risco de ocorrerem falhas de memória, visto que o questionário é aplicado logo após realizar a atividade proposta no experimento. Esse método pode ser utilizado para identificar aspectos comportamentais e emocionais do engajamento;;

- **Entrevista** - Nas entrevistas, os participantes são convidados a contar suas histórias de formas mais abertas e não estruturadas em perguntas objetivas. Uma vantagem deste método é que, durante a entrevista, o estudante pode se sentir à vontade para descrever com mais detalhes como foi sua experiência de aprendizado e quais fatores podem ter facilitado ou dificultado essa experiência. E como desvantagem, o entrevistador pode afetar a confiabilidade, a profundidade e os tipos de respostas (Fredricks e McColskey 2012).
- **Observação** - Trata-se de um método desenvolvido para avaliar o comportamento de um estudante ou de um grupo deles, durante a realização de uma atividade. Pode ser utilizado para comparar com informações sobre o engajamento coletado a partir de relatos de professores, alunos e de entrevista. A principal desvantagem é a demanda de esforço e tempo para realizar entrevistas, por esse motivo, é utilizada apenas para um pequeno número de estudantes. Deve ser utilizado para reconhecer comportamentos observáveis do estudante;

2.2 Pedagogia da Contribuição do Estudante

A Aprendizagem Ativa (AA) é um termo guarda-chuva que envolve estratégias de aprendizagem em que os estudantes apresentam mais autonomia e controle do desenvolvimento de suas atividades (Anthony 1996). É considerada centrada no aluno porque incentiva sua proatividade e o transforma em protagonista do processo de aprendizagem.

AA inclui “atividades instrucionais envolvendo os alunos em fazer coisas e pensar sobre o que estão fazendo” (Bonwell e Eison 1991, Prince 2004). Alguns componentes são

fundamentais para o seu sucesso, dentre eles (Morán 2015):

- Há menos ênfase na *transmissão de conhecimento* e maior ênfase no desenvolvimento de habilidades do aluno;
- Maior estímulo ao desafio em forma de jogos ou não;
- Os alunos passam mais tempo realizando atividades de aprendizagem do que apenas ouvindo o professor;
- Incentivo ao aprendizado com a interação entre alunos e trabalho em grupo;
- Professor atua como mediador e não como principal responsável pela aprendizagem do aluno;

A aplicação da AA deve seguir princípios que se relacionam com a construção de estudantes mais proativos, incluindo o desenvolvimento da reciprocidade e cooperação entre os alunos.

Esta pesquisa de doutorado se encaixa em uma das estratégias de AA chamada Pedagogia da Contribuição do Estudante (PCE). Nela, o estudante pode contribuir para o material de aprendizagem com base em suas próprias experiências.

2.2.1 Características

A PCE ¹ é uma abordagem que enfatiza atividades que envolvem o estudante como um colaborador ativo das experiências e recursos de aprendizado de si mesmo e de outros. É motivada por menos palestras e mais atividades. Essas atividades podem assumir várias formas e ser realizadas de maneira individual ou em grupo (Collis e Moonen 2006).

A definição dessa pedagogia considera a perspectiva do estudante como quem contribui e quem utiliza a contribuição de outros indivíduos (Collis e Moonen 2005, Hamer et al. 2008):

“Uma pedagogia que incentiva os alunos a contribuir para a aprendizagem da comunidade e a valorizar as contribuições de outros”.

¹Tradução do termo em inglês *Contributing Student Pedagogy*

PCE inclui atividades de aprendizagem que incentivam o compartilhamento de soluções, criação de material de estudo e revisão do trabalho dos colegas. Essas atividades oferecem oportunidades para que os estudantes se envolvam com o material da disciplina e desenvolvam importantes habilidades como pensamento crítico, criatividade e melhoria das relações interpessoais (Hamer et al. 2012, Falkner e Falkner 2012).

No ambiente que estimula a contribuição, o instrutor não exerce o controle sobre todos os aspectos do curso e está disposto a negociar ou delegar algumas decisões aos estudantes. Dessa forma, as atividades baseadas em PCE tendem a atribuir novas responsabilidades ao estudante no processo de aprendizagem. Essas novas responsabilidades englobam a criação de recursos e avaliação da contribuição de outras pessoas (Hamer et al. 2008).

As atividades podem ou não ser realizadas em equipe. O foco do trabalho está nas contribuições, isto é, no recurso resultante das atividades através do qual outros aprendem o conteúdo. Esse tipo de abordagem também incentiva o desenvolvimento de uma comunidade de aprendizagem, em que pessoas produzem e consomem recurso. A tecnologia exerce papel fundamental no ambiente de contribuição. A ferramenta utilizada deve apoiar a colaboração, a comunicação e a geração de materiais pelos estudantes, bem como permite que os instrutores monitorem o desenvolvimento das atividades.

Há muitos benefícios na aplicação de PCE. De acordo com (Collis e Moonen 2006), alguns desses benefícios são:

- Aprender com o trabalho de outros aprendizes;
- Utilizar o trabalho e as experiências de outros alunos como *feedback*;
- Expandir os recursos disponíveis para o curso e que podem ser reutilizados em outros cursos;
- Diálogo e interação com outras pessoas durante a realização atividades;

A implantação do PCE passa a falsa sensação de ser fácil delegar aos estudantes a responsabilidade de “fazer todo o trabalho”. Porém, Kirshner et al. argumentam que há fortes evidências de que aplicar atividades com pouca orientação baseada em ideias construtivistas é menos eficaz do que abordagens que suportam o ensino tradicional. Além disso, afirmam

ainda que a instrução não orientada pode ter efeitos negativos no aprendizado dos estudantes (Kirschner, Sweller e Clark 2006).

Com relação às atividades de PCE, a orientação e o apoio do instrutor são essenciais para garantia do sucesso. Essa orientação pode não estar na forma tradicional de apresentação de palestras, mas pode incluir direcionamento de quais recursos utilizar, promovendo um ambiente seguro para os estudantes apresentarem suas contribuições e apresentando os critérios de avaliação. Esses pontos levam ao alto custo de implantação de atividades baseadas em PCE (Kirschner, Sweller e Clark 2006).

2.2.2 Exemplos de Atividades

A PCE se concentra na construção e avaliação de recursos. O exemplo mais comum para aplicar esse tipo de abordagem é a de instruir estudantes, para criarem uma pergunta que avalie um ou mais resultados de aprendizagem de um curso. Além da pergunta, os estudantes devem escrever uma solução como gabarito. Ambos os recursos devem ser armazenados em um banco de dados, onde ficam disponíveis para uso de outros estudantes como recurso de aprendizado (Luxton-Reilly e Denny 2010).

Outras atividades, além da criação de pergunta, compõem o processo de contribuição. Os estudantes também precisam responder, revisar, fornecer *feedback* e avaliar a produção das outras pessoas. Detalhamos exemplos das atividades de contribuição a seguir:

- **Criação de Perguntas e Solução** - Ao elaborar, o estudante precisa enquadrar o contexto da pergunta no conteúdo, definir critérios de desempenho, consultar o material do curso para encontrar exemplos apropriados e refletir sobre precedência de outros conteúdos. Inclusive, aprendem a se expressar com clareza e desenvolvem habilidades de auto-avaliação. Esse tipo de atividade também exige que o estudante compreenda o conteúdo abordado na questão. Dessa forma, a criação de perguntas e soluções pode expor uma falta de entendimento sobre um tópico. Perguntas criadas por estudantes podem ser de múltipla escolha, de respostas livres (subjetivas) e questões de programação. Questões de múltipla escolha são objetivas e apresentam alternativas para solução, sendo apenas uma correta. Questões subjetivas são aquelas que não exigem um formato pré-estabelecido de resposta. Enquanto as de programação são um caso

especial de pergunta subjetiva em que o autor exige um programa escrito como solução. Dessa forma, deverá escrever especificações com clareza, incluindo um exemplo de entrada e saída.

- **Responder questões criadas por estudantes** - Com a criação de questões criadas por estudantes, um banco de dados é desenvolvido com baixo custo para os professores. A PCE sugere que essas questões sejam utilizadas pelos estudantes para praticar suas habilidades e revisar o material do curso.
- **Avaliar perguntas e respostas** - Os estudantes avaliam trabalho de seus colegas, classificando tanto as perguntas quanto as respostas. Ao avaliar, o estudante fornece *feedback* sobre a qualidade do trabalho de outros indivíduos.
- **Discutir e fornecer *feedback*** - Com esse tipo de atividade, é explorada a capacidade de escrever comentários e pedir esclarecimentos. Ao ler críticas construtivas, o autor pode obter dicas e ideias para melhorar sua pergunta, conseqüentemente aumentando a qualidade dos recursos disponíveis no banco de dados.

2.3 Conclusão

Neste capítulo, vimos como engajamento é conceituado por vários autores e descrevemos como o termo é utilizado ao longo do trabalho. Como também, detalhamos as dimensões de comportamento, cognição e emoção que são mencionadas em algumas das definições. O comportamento é representado por todas as atitudes observáveis dos estudantes. Sendo que a emoção inclui todos os sentimentos que o estudante sente durante a experiência de aprendizagem e, a cognição envolve o esforço necessário para compreender ideias complexas e desenvolver habilidades difíceis. Também apresentamos instrumentos de medição de engajamento e quais os principais métodos utilizados por eles. Além disso, aqui introduzimos as características da abordagem conhecida como Aprendizagem ativa que é uma das intervenções mais eficientes para incentivar o engajamento de estudantes, conhecida como Pedagogia de Contribuição do Estudante. Essa metodologia de aprendizagem que fundamenta essa pesquisa de doutorado. Por meio da aplicação de atividades de contribuição, os estudantes são

incentivados a contribuir para a aprendizagem da comunidade e a valorizar as contribuições de outros. Mostramos ainda exemplos de atividades baseadas na contribuição.

Capítulo 3

Modelo de Personalização de Atividades de Programação

Essa tese de doutorado apresenta um modelo para personalização de atividades com o intuito de estimular o engajamento de estudantes aprendizes em atividades de aprendizagem de programação. Esse modelo se encaixa na Pedagogia da Contribuição do Estudante, um tipo de abordagem de aprendizagem ativa, em que o aprendiz é responsável por contribuir com material de estudo para o próprio aprendizado e o de outros indivíduos.

Neste capítulo, apresenta-se uma visão geral do modelo de personalização, detalhando os tipos de atividades personalizadas de programação, os momentos em que há necessidade de intervenção do professor ou do suporte de uma ferramenta de aprendizagem, além de exemplificar como implantar o novo modelo de atividades.

3.1 Visão Geral

Durante o desenvolvimento dessa pesquisa, foi estimulada a interação e colaboração entre estudantes de programação por meio do desenvolvimento de atividades práticas de aprendizagem. A ideia é que essa interação entre os estudantes aconteça através da criação, revisão e compartilhamento de atividades de programação. O produto que resulta dessa interação é o que foi nomeado como Atividade de Programação Personalizada. O termo personalização é utilizado para representar que a atividade explora o conteúdo que um estudante específico não compreendeu adequadamente, e que a estratégia para resolução também deve atender às

necessidades desse aluno individualmente.

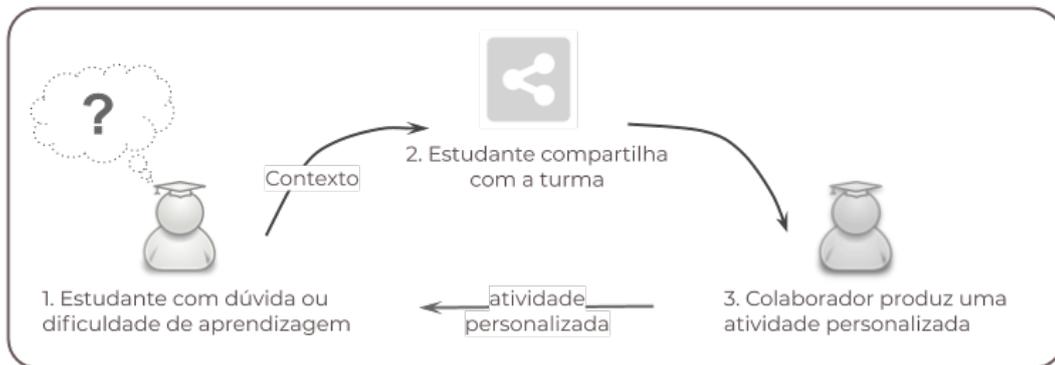


Figura 3.1: Cenário 1: Ciclo que compreende o processo de compartilhamento da dificuldade de aprendizagem pelo aluno e a elaboração de uma nova atividade personalizada pelo colaborador

A Figura 3.1 apresenta o primeiro cenário em que o modelo de personalização pode ser aplicado. O estudante apresenta dúvida em algum conceito ou dificuldade na resolução de um exercício. Em seguida, compartilha com a turma o contexto em que se enquadra a dificuldade que está enfrentando, sinalizando a necessidade de intervenção. Em consequente, outro estudante, chamado de colaborador, interpreta o contexto, reflete sobre o problema do colega com dúvida e elabora uma nova atividade no intuito de sanar a dúvida.

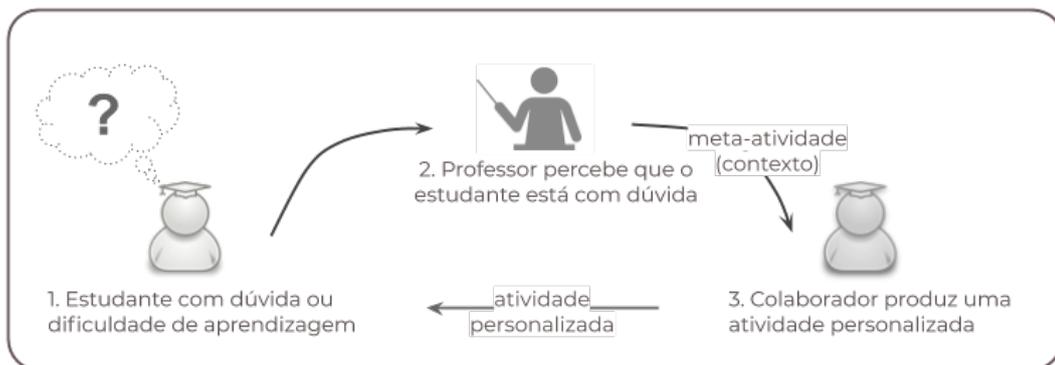


Figura 3.2: Cenário 2: Ciclo em que o professor percebe a dúvida que não foi compartilhada e gera uma meta-atividade, que é repassada ao colaborador

A Figura 3.2 apresenta o segundo cenário em que o modelo pode ser utilizado. Nesse caso, embora a dificuldade não tenha sido comunicada pelo aluno, o professor percebe que o estudante não está evoluindo no mesmo ritmo que os demais em determinado assunto da disciplina. Esse fator o leva a elaborar uma meta-atividade que descreve o contexto que o

estudante está inserido e atribui ao colaborador a responsabilidade de elaborar uma nova atividade personalizada.

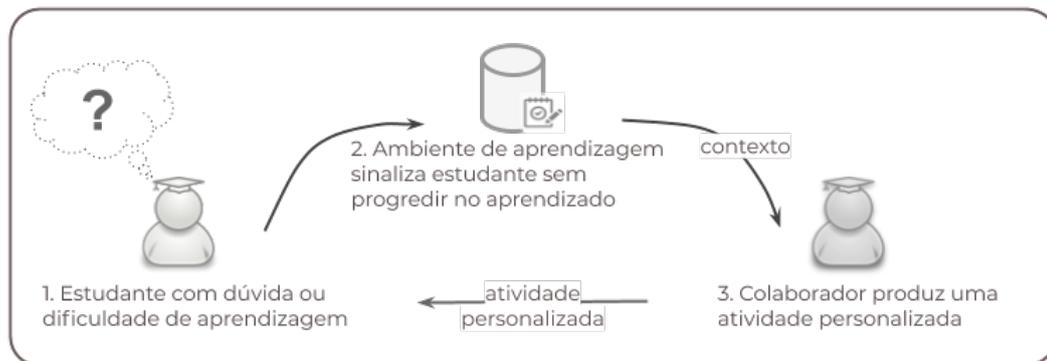


Figura 3.3: Cenário 3: Ciclo em que a dificuldade do aluno é identificada por meio de um Ambiente de Aprendizagem que acompanha o progresso do estudante

A Figura 3.3 apresenta o cenário em que a identificação da dificuldade de progresso do estudante é realizada automaticamente. Nesse contexto, o Ambiente de Aprendizagem deve coletar indicadores de engajamento por meio de atividades de aprendizagem. Esses indicadores, por sua vez, devem contribuir na identificação do nível de progresso de um estudante com dificuldade pelo professor. Nesse cenário, o professor atua de forma participativa, selecionando quais são os indicadores que melhor se encaixam para o contexto da turma. No próximo capítulo, serão apresentados alguns indicadores de engajamento como proposta para utilização em atividades de programação.

A Figura 3.4 une os três cenários apresentados e relaciona os papéis envolvidos no modelo. Sendo assim, tem que o requisitante, que pode ser o estudante, o professor ou um Ambiente de Aprendizagem, é o responsável por notificar um colaborador, contextualizar a dúvida e receber a atividade personalizada de programação. Já o colaborador é um estudante responsável por produzir a nova atividade personalizada baseada no contexto da dúvida e fornecê-la ao requisitante.

O modelo visa estimular a colaboração *de* estudantes *para* estudante. Dessa forma, considera-se que o colaborador desempenha o papel fundamental no modelo, pois tem a responsabilidade de elaborar as atividades personalizadas de programação. Esse papel pode ser atribuído a um monitor da disciplina, um estudante que está em fase mais avançada ou um aluno que cursa a disciplina e sente confortável com a colaboração.

O requisitante é responsável por solicitar a criação de uma nova atividade levando em



Figura 3.4: Visão Geral do Modelo de Personalização

consideração o contexto da dúvida do estudante. Essa solicitação pode ser automática ou manual, em que a automática ocorre quando o curso utiliza um sistema de aprendizagem com recursos e funcionalidades para monitorar a realização de atividades e identificar quando um estudante não está conseguindo avançar nas atividades, e a manual quando o professor ou o próprio estudante percebe que há uma dificuldade em prosseguir e gera a demanda para o colaborador.

3.2 Atividade Personalizada de Programação

A atividade personalizada é um exercício de programação de autoria de um estudante colaborador que se baseou em um contexto específico. Pode ser um exercício de resolução de problemas de programação ou um *quiz* de perguntas objetivas.

O termo *personalizada* é utilizado para representar que o exercício é direcionado para as dificuldades de um estudante, levando em consideração a dúvida no conteúdo que prejudicou o progresso na disciplina e a estratégia que o levaria a refletir sobre o que está dificultando sua aprendizagem. Frente a isso, define-se atividade personalizada de programação como:

Um tipo de atividade ou exercício de programação inspirado em uma situação ou dificuldade enfrentada por um estudante real. A resposta deve auxiliar o estudante na reflexão sobre a dificuldade de aprendizagem, e pode seguir o formato

de indicação de outros exercícios e casos de testes, revisão da solução proposta pelo estudante ou análise do texto do enunciado de uma questão existente.

A atividade de personalizada apresenta dois componentes principais: o estudante personagem e o contexto. O personagem é aquele que apresenta dificuldade na aprendizagem de algum conteúdo relacionado com a programação. Essa dificuldade pode ser exemplificada como a falta de compreensão de um conceito, o uso do comando da linguagem de programação ou a interpretação do erro no código e no texto do enunciado da questão. Esse personagem pode sinalizar que está com essa demanda ou o sistema pode identificar a necessidade de intervenção. O contexto representa o cenário que a dúvida surgiu e pode incluir o exercício que o estudante estava tentando solucionar, a tentativa de resposta e o resultado da avaliação. O tipo de atividade deve ser definido pela análise dos elementos descritos no contexto.

Atividade Personalizada

Preencha a tabela a seguir com novos casos de teste dando exemplos de entrada válida, saída esperada e qual o valor obtido pela execução do seu programa. Insira na primeira coluna exemplos de entradas que atendam aos requisitos definidos no enunciado da questão. Na segunda, acrescente o que deve ser exibido na tela quando a entrada for o valor que está na primeira coluna. Para preencher a terceira coluna, execute o seu programa com os valores que estão na coluna de entrada e verifique o que é exibido na tela. Os valores das colunas 2 e 3 devem ser iguais.

Entrada válida	Saída Esperada	Saída Obtida
0	0	
4		

Figura 3.5: Exemplo de Atividade Personalizada

A Figura 3.5 apresenta um exemplo de atividade personalizada de sugestão de casos de

testes. A atividade é formada por um enunciado que descreve o que deve ser realizado, uma tabela que o aprendiz deve preencher com exemplos de casos de testes e o que o programa produz quando executa, recebendo os dados de entrada como parâmetro. Observe que o colaborador não fornece a resposta pronta, pelo contrário, utiliza exemplos para que o estudante reflita sobre o erro que o sistema encontra em seu código.

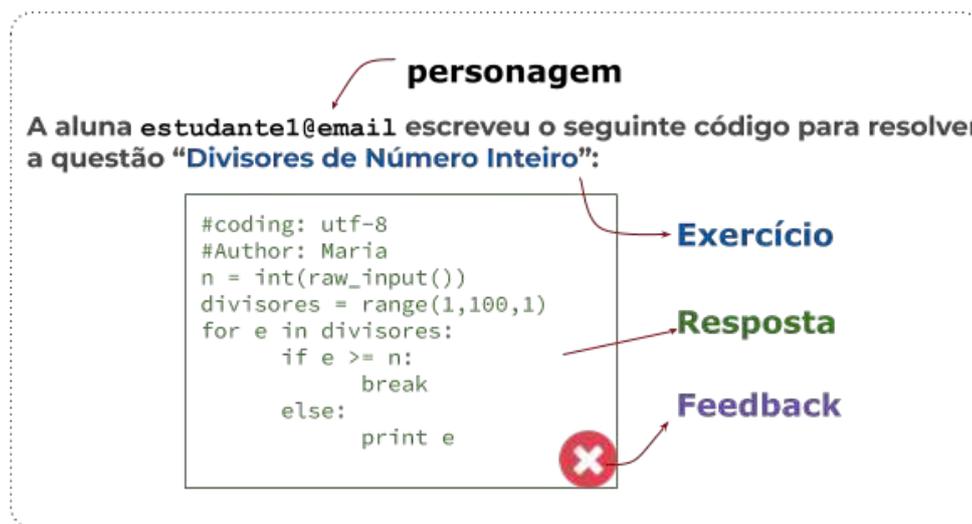


Figura 3.6: Contexto de Atividade Personalizada

A Figura 3.6 apresenta as informações que motivaram a criação da atividade exibida na Figura 3.5. O estudante¹ representa um exemplo do tipo de dificuldade que um estudante real pode apresentar em elaborar uma solução correta para a questão cujo objetivo é encontrar os divisores de um número inteiro. O colaborador também tem acesso à resposta enviada pelo estudante1 e o *feedback* sobre a corretude da solução. De posse dessas informações, o requisitante pode orientar o tipo de atividade personalizada que deve ser gerada, essa também pode ser uma decisão do colaborador.

3.3 Atribuição de Papéis

Existem três papéis importantes no modelo de personalização: o estudante colaborador, o que está com dificuldade de aprendizagem e o requisitante de atividades personalizadas. A mesma pessoa pode assumir mais de um papel em momentos diferentes do processo de cola-

¹Todos os nomes de alunos e e-mails usados neste trabalho são irreais.

boração. Em um cenário, o estudante pode estar com dificuldade no aprendizado e precisar de ajuda e, em outro, atuar como colaborador e desenvolver novas atividades personalizadas.

- O estudante é quem inicia a demanda por colaboração porque está com dificuldade na aprendizagem de algum conteúdo explorado no curso ou não consegue em resolver um problema de programação.
- O colaborador também é um estudante, mas que assume a responsabilidade de participar do processo de aprendizado de outros alunos. Podem ser tanto tutores e monitores do curso, como também outro estudante do curso que tenha interesse em contribuir com o aprendizado de programação dos colegas.
- O requisitante é quem percebe a presença da dificuldade no aprendizado de programação no estudante. Esse requisitante pode ser o professor ou o próprio estudante que percebe que não está conseguindo esclarecer uma dúvida sem ajuda. Em outros cenários, o papel do requisitante pode ser desempenhado por um sistema (ou ambiente) de aprendizagem que apresenta funcionalidades capazes de identificar e sinalizar quando um estudante não está evoluindo no conteúdo.

3.4 Tipos de Atividades Personalizadas

Baseando-se em estratégias consolidadas para o aprendizado de programação e compartilhamento de conhecimento, propõe-se a utilização de cinco tipos de atividades personalizadas. A meta-atividade de criação ou resolução de novos exercícios elaborados por estudantes são provenientes da Pedagogia de Contribuição do Estudante, que une habilidades diversas e efeitos positivos no desempenho do aprendiz. As atividades de revisão de código e sugestão de casos de testes são decorrentes de práticas adotadas em processos de desenvolvimento de software para compartilhamento de informação. Já a revisão do enunciado surgiu da dificuldade dos estudantes em interpretação de texto e de problemas encontrados nos enunciados das atividades.

3.4.1 Criação de Novo Exercício

A criação de novos exercícios é um tipo de meta-atividade, pois se refere à criação de uma nova atividade na qual o enunciado descreve elementos importantes, que estão de acordo com as dificuldades do estudante, fornecendo informações essenciais para estruturação de uma atividade personalizada.

As questões geradas por estudantes podem assumir formatos diversos, incluindo *quizzes*, questões objetivas, subjetivas e de resolução de problemas de programação. Especialmente para o último formato, em contextos em que se adota uma abordagem dirigida por testes, além do enunciado com a descrição do problema, outros componentes são necessários. Exercícios não atrelados a essa abordagem podem ser descritos de uma maneira diferente e omitir os seguintes componentes:

- Descrição do formato dos dados que devem ser considerados válidos para **entrada** do programa;
- Descrição do formato do dado esperado como **saída** do programa;
- Lista de exemplos de **casos de teste** para determinar se os requisitos funcionais dos programas estão sendo atendidos pela solução;
- O código de um programa que resolva o exercício que deve ser usado como **solução de referência**. Isso garante que o autor elaborou uma questão cuja solução é factível, além de levá-lo a repensar e reavaliar os outros componentes;

A Figura 3.7 apresenta um exemplo de meta-atividade de criação de novo exercício. No enunciado, o requisitante apresenta as funções de manipulação de Strings como conteúdo que deve ser explorado, e define que o tipo de exercício desenvolvido deve ser o de resolução de problemas.

A Figura 3.8 apresenta outro exemplo de meta-atividade de criação de novo exercício inspirada em uma dúvida compartilhada por um aluno por meio de uma ferramenta de comunicação utilizada em um curso de programação. O personagem *estudante2* evidencia dúvidas em relação às funções de manipulação de Strings pela mensagem como criar uma função que retorne a quantidade de caracteres digitados que foi a ele atribuída.

Crie um exercício de programação para praticar o uso de funções de manipulação de String.

O enunciado do exercício deve ter:

- Descrição do formato dos dados que devem ser considerados válidos para entrada do programa
- Descrição do formato do dado esperado como saída do programa
- Lista de exemplos de casos de teste para determinar se os requisitos funcionais dos programas estão sendo atendidos pela solução
- O código de um programa que resolva o exercício e que pode ser usado como gabarito ou solução de referência

Figura 3.7: Meta-atividade de Criação de Atividades

Crie um exercício de programação para estudante2@email praticar o uso de funções de manipulação de String.

O enunciado do exercício deve ter:

- Descrição do formato dos dados que devem ser considerados válidos para entrada do programa
- Descrição do formato do dado esperado como saída do programa
- Lista de exemplos de casos de teste para determinar se os requisitos funcionais dos programas estão sendo atendidos pela solução
- O código de um programa que resolva o exercício e que pode ser usado como gabarito ou solução de referência

Figura 3.8: Atividade Personalizada de Criação de Exercício

A diferença entre esses dois exemplos é a informação sobre quem é o estudante que está enfrentando uma situação de dificuldade no aprendizado das funções de manipulação de String. O exemplo da Figura 3.8 foi baseado em uma dúvida divulgada publicamente, por isso percebe-se que em algumas situações esse tipo de informação pode ficar explícita no enunciado. Mas nada impede que o professor possa se inspirar na dúvida de um aluno e requisitar atividades sem mencionar o estudante.

3.4.2 Resolução de Exercício Produzido por Estudante

O resultado da realização das atividade de criação é um novo exercício produzido por estudantes. Em atividades personalizadas de resolução de exercícios, os estudantes devem interpretar o problema e elaborar uma solução no formato estabelecido no enunciado. Atividades de resolução de exercícios já são conhecidas e aplicadas em cursos de programação, mas, geralmente, os exercícios aplicados são elaborados por professores. Esse tipo de estratégia oriunda da PCE tem como objetivo a valorização do trabalho dos estudantes que estão produzindo o conteúdo.

3.4.3 Sugestão de Casos de Teste

Nas atividades de sugestão de casos de teste, o estudante é estimulado a analisar o enunciado do exercício, reconhecer seus requisitos funcionais e analisar os dados que são esperados como saída. Além disso, para sugerir novos testes, o estudante precisa compreender o código e identificar erros.

Por se tratar de um estilo de atividade pouco explorada em disciplinas de programação, é necessário introduzir o conceito sobre teste de software, quais são as suas características e os aspectos que devem ser testados, como as condições de parada de laços, lógica dentro de comandos condicionais, formato de entrada/saída, dentre outros.

3.4.4 Revisão de Código

A atividade personalizada de revisão de código é uma estratégia adotada por processos de desenvolvimento de software para compartilhamento de informação., dado que, ao analisar código, o indivíduo é estimulado a ler outras soluções, conhecer novas estratégias para solucionar um problema e refletir sobre a sua própria solução. Outra vantagem da revisão de código é que permite a discussão de ideias, estimulando a comunicação e o pensamento crítico dos envolvidos.

Assim como na atividade de sugestão de casos de teste, a de revisão de código não é uma prática adotada com frequência nos cursos de programação. Por esse motivo, recomenda-se a apresentação dos critérios utilizados para uma revisão efetiva de código, para que o estudante com dificuldade seja capaz de perceber em qual conteúdo deve se aprofundar .

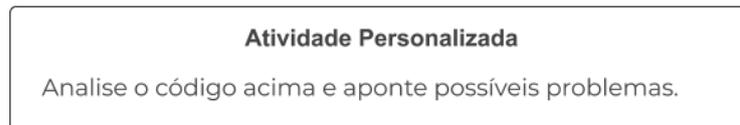


Figura 3.9: Atividade de Revisão de Código

Como exemplo prático, o professor pode criar um roteiro de perguntas em que o estudante reconhece problemas passo-a-passo: o código faz a leitura da entrada de forma correta? Produz a saída conforme esperado? O resultado da execução dos casos de teste está correto? As condições de limite de execução dos laços estão corretas?

A Figura 3.9 apresenta um exemplo de atividade personalizada de revisão de código para o contexto apresentado na Figura 3.6. Para responder a atividade, ao invés de elaborar o código de uma nova solução para o problema dos divisores de um número inteiro, o respondente deve analisar o código e encontrar possíveis problemas.

Certamente, a atividade de revisão requer mais detalhes do que simplesmente “encontrar erro”. É preciso sugerir mudanças para melhoria da qualidade do código, da escolha da estrutura de dados adequada (se houver necessidade), definição do nome das variáveis, dentre outras. Cursos de introdução à programação, em geral, estão mais concentrados em explorar aspectos funcionais, dando menos ênfase ao conhecimento avançado em qualidade de *software*. Por isso, sugere-se que atividades de revisão se concentrem nos requisitos funcionais encontrando problemas no código. Isso pode ajudar o estudante que está com dúvidas a refletir sobre melhorias, direcionando-o a encontrar alternativas de solução.

3.4.5 Revisão de Uma Atividade Existente

Em algumas situações, o estudante não está com dúvida na programação em si, mas sim na interpretação do problema reportado no exercício, nos casos de testes, ou em outros componentes da atividade. Ao revisar o texto, o estudante pode corrigir possíveis erros para melhorar a compreensão da atividade, corrigir falhas de ortografia e gramaticais, além de aprimorar os casos de testes e a solução de referência.

A Figura 3.10 apresenta um exemplo de atividade personalizada de revisão da questão *Restaurante Universitário* de autoria de um estudante. Para responder a atividade, o revisor deve analisar o enunciado, encontrar e reportar possíveis problemas, ao invés de simples-

Restaurante Universitário

O Restaurante Universitário é responsável por fornecer refeições aos alunos da universidade. Após reivindicações do alunos que não comem carne por uma outra opção que substitua a mesma, a direção passou a fazer uma pesquisa para saber a porcentagem de vegetarianos que frequentam o restaurante. Você deve criar um programa que receba uma entrada informando se o aluno come carne ou é vegetariano e mostrar na saída a porcentagem dos alunos vegetarianos em relação ao total.

Entrada

O programa irá receber uma sequência de vogais, que podem ser: "V" (Vegetariano) e "C"(Come carne).

Saída

O programa deve imprimir a média de vegetarianos em relação ao total.

Exemplo de Execução

VCCVCVCVCCC
A porcentagem de alunos vegetarianos e 36.36%.

Testes

Entrada	Saída
VVVVCC	A porcentagem de alunos vegetarianos e 66.6%.
CCCCCCC	A porcentagem de alunos vegetarianos e 0%.



Atividade Personalizada

Revise o enunciado da questão "Restaurante Universitário" e aponte possíveis erros em seu enunciado e nos testes.

Figura 3.10: Atividade Personalizada de Revisão de Texto

mente gerar uma solução correta para esse problema.

A revisão de atividade demanda mais esforço porque absorve o trabalho de outras atividades, incluindo a revisão de código, sugestão de casos de testes e a resolução do problema exibido como contexto. Isso quer dizer que para revisar uma questão, o estudante deve:

- Desenvolver o código de um programa que resolva a atividade. Dessa forma, o apren-

diz pode refletir sobre o problema, desenvolver sua própria solução e identificar possíveis erros no enunciado (texto e casos de testes);

- Revisar o código da solução de referência, caso sejam encontrados problemas no código, o respondente pode sugerir casos de testes que acusem erros que não estavam sendo atingidos na versão atual da questão;
- Revisar os casos de testes e identificar situações não avaliadas pelos casos de testes daquela versão da atividade;
- Revisar o texto do enunciado, pois é possível melhorar a escrita do problema, utilizando outro vocabulário, exemplos mais simples ou correção de erros de português.

3.5 Conclusão

Neste capítulo, além de detalhar o modelo de personalização das atividades que foi utilizado para estimular o engajamento de aprendizes de programação, apresentou-se três cenários em que o modelo pode ser aplicado. Quando o estudante aponta que está com dificuldade aos colegas da turma, professor identifica a dificuldade sem ser comunicado ou o ambiente de aprendizagem sinaliza que há alguém sem progredir da forma esperada. Apresentou-se, também, os papéis envolvidos no processo: estudante com dificuldade de aprendizagem, professor, ambiente de aprendizagem e colaborador. Os três primeiros, chamados de requisitantes, são responsáveis por sinalizar o colaborador sobre o contexto e a dificuldade de aprendizagem que um estudante está enfrentando. O papel do colaborador pode ser assumido tanto por outro estudante matriculados na mesma disciplina, quanto por um monitor ou voluntário que tenha interesse em participar do processo de aprendizagem de outras pessoas. Além disso, a atividade personalizada de programação foi definida como um tipo de atividade ou exercício de programação inspirado em uma situação ou dificuldade enfrentada por um estudante real que a resposta da atividade deve auxiliar o estudante a refletir sobre a dificuldade de aprendizagem. Por fim, foram apresentados os tipos de atividades personalizadas de programação que são estabelecidos pelo formato da resposta esperada por cada atividade e foram chamados de criação ou resolução de novos exercícios, sugestão de casos de testes, revisão do código de uma solução ou revisão do texto do enunciado de uma questão existente.

Capítulo 4

Métricas de Engajamento em Atividades de Programação

Esta tese de doutorado apresenta um modelo para personalizar o exercício visando estimular o engajamento de estudantes aprendizes em atividades de aprendizagem de programação. Com a aplicação do modelo de personalização, é possível identificar indicadores para sinalizar o engajamento de aprendizes em atividades de aprendizagem de programação. Foram apresentados indicadores que levam em consideração o momento, o propósito e o conteúdo da resposta produzida pela realização da atividade. A partir dessas informações, propõe-se indicadores para o aspecto comportamental e cognitivo do engajamento. De modo geral, não é fácil identificar sensações e relacioná-las a um evento específico. Por isso, foi proposta a utilização de um instrumento de auto-relato para auxiliar no reconhecimento de sensações após a realização de atividades personalizadas de programação e, conseqüentemente, conhecer o engajamento emocional do estudante.

4.1 Engajamento Comportamental

Embora seja conhecida a existência de diversos meios para aprendizagem, como a leitura de livros, assistir aulas, dentre outras, o presente estudo parte da premissa de que *o aluno está estudando quando realiza exercícios*. Utilizou-se a característica inerente à habilidade de programação que exige muita prática e foram definidos novos indicadores de engajamento comportamental baseados no desenvolvimento de atividades personalizadas.

A Tabela 4.1 apresenta as características de um estudante engajado em atividades que foram julgadas como importantes para o contexto do aprendizado de programação. A primeira coluna apresenta as características e a segunda as relaciona com indicadores de engajamento.

Tabela 4.1: Indicadores de Engajamento Comportamental

Característica	Indicadores
<i>Dedicação</i>	Número de atividades concluídas, Média do tempo de realização das atividades, Duração da Sessão de estudo, Número de casos de testes corretos
<i>Persistência</i>	Número de questões não resolvidas, Número de casos de testes incorretos, Número de tentativas de respostas
<i>Hábito de Estudo</i>	Número de tarefas realizadas em um período, Questões concluídas em casa
<i>Participação</i>	Número de atividades extra realizadas

Dedicação é a qualidade ou a condição de quem se entrega (Dicio 2020). Assim, para representar que o estudante está dedicado, propõe-se as seguintes métricas:

- **Atividades Concluídas** - uma atividade é considerada concluída quando a resposta enviada pelo estudante estiver correta. No entanto, caso seja uma atividade de resolução de problemas de programação, ela só é considerada concluída quando todos os casos de testes não acusam erros no código da resposta.
- **Tempo de Realização das Atividades** - reflete o intervalo de tempo entre o instante em que o aluno inicia a leitura do enunciado e o que ele a conclui.
- **Sessão de Estudos** - representa o período que o aluno se dedica ao desenvolvimento das atividades de programação.
- **Casos de Testes Corretos** - é uma métrica que só pode ser obtida em atividades de resolução de problemas e que apresentam exemplos de casos de testes para validar a correteza da resposta do aluno.

Segundo o dicionário, persistir significa continuar ou prosseguir, insistir (Dicio 2020). Nessa pesquisa, persistência foi definida como o empenho de resolver um exercício até receber a aceitação absoluta dos testes existentes. Por isso, foram definidas as seguintes métricas como indicador de **persistência**:

- **Exercícios Não Resolvidos** - só deve ser considerado exercício não resolvido aquele que o aluno leu, mas não respondeu. Isso pode sugerir que ele não conseguiu resolver ou que ele encerrou sua sessão de estudo.
- **Casos de Testes incorretos** - só pode ser obtida em atividades de resolução de problemas e que apresentam exemplos de casos de testes para validar a corretude da resposta do aluno. Quando existem casos de testes que acusam erro no código solução, pode significar que o aluno está com dificuldade naquele problema e não persistiu até conseguir desenvolver o código com uma solução completamente correta.
- **Tentativas** - em geral, para alcançar o sucesso em uma questão, o aprendiz faz algumas tentativas. Se há um número alto de tentativas, representa que o aprendiz continuou persistindo mesmo sem conseguir apresentar a resposta correta rapidamente.

O **hábito de estudo** representa um padrão de comportamento desenvolvido para adquirir conhecimento. Com o registro do horário da realização de atividades, é possível identificar as seguintes métricas:

- **Tarefas realizadas em um período** - reconhecer o período que o aluno passa mais tempo estudando.
- **Exercícios concluídos em casa** - quando não forem realizados no horário da aula presencial, considera-se que foram feitos como tarefa de casa.

Enquanto que a **participação em atividades extra** é representada pelo número de exercícios que foram realizados como atividade complementar.

4.2 Engajamento Cognitivo

A inspiração para esse aspecto do engajamento veio do esporte, levando em consideração como os atletas progridem em corridas e maratonas, já que precisam manter um bom rendimento durante todo o percurso, que costuma durar horas. O termo *pace* é comumente utilizado nesse meio para representar o ritmo médio ou velocidade na corrida e, geralmente, é medido pelo tempo que se leva para percorrer 1 km. O maratonista treina com o objetivo de

melhorar o *pace*, buscando alcançar melhores resultados. Considerando que cada indivíduo tem suas peculiaridades e aprende em seu tempo, sugere-se uma métrica para acompanhar o progresso individual chamada de **ritmo de aprendizagem**. Essa métrica representa a velocidade de aprendizagem e indica como o aluno está compreendendo o assunto abordado em um determinado instante do curso. Propõe-se que seu valor seja estimado pela proporção entre quantidade de metas que o estudante alcançou e as metas que ele deveria alcançar acerca de um determinado assunto abordado no curso. A fórmula abaixo calcula o valor do ritmo de aprendizagem:

$$\text{Ritmo de Aprendizagem} = \frac{\text{Qtd de Metas Alcançadas}}{\text{Total de Metas Definidas}}$$

Uma meta representa um conceito que o aluno precisa aprender, ser capaz de fazer ou uma habilidade que deve adquirir. O professor define quais são as metas de cada atividade de aprendizagem. Após a correção da atividade, o professor determina quais foram as metas que estudante conseguiu alcançar.

O conceito de meta de aprendizagem pode ser adaptado para várias situações. Assim, é possível concretizar esse conceito de formas que o professor julgue relevante para o contexto da disciplina. Podem ser considerados: o número de acertos em questões objetivas, número de casos de testes corretos ou o número de envios de códigos. Nessas situações, o ritmo de aprendizagem deve ser adaptado à proporção entre o número atingido pelo aluno e o número total disponível, sejam questões, testes ou envios. Abaixo, apresentamos propostas de outras metas de aprendizagem para atividades sobre conceito fundamentais de algoritmos de programação:

- Compreender o conceito de algoritmo;
- Conhecer as características de um algoritmo;
- Reconhecer exemplos de algoritmos no cotidiano do aluno;

É importante saber que, ao acompanhar o progresso individual representado pelo ritmo de aprendizagem periodicamente após o aluno ser submetido a alguma atividade com metas

definidas, o professor pode conhecer as dificuldades, personalizar o atendimento e ter tempo de adotar abordagens de ensino-aprendizagem de forma efetiva. Dado que, em situações comuns, o progresso só é avaliado pela nota em uma atividade de avaliação de conteúdo.

A Tabela 4.2 apresenta exemplos de valores de ritmo de aprendizagem de alunos após a realização de 4 atividades. A Atividade **A** possui 2 metas de aprendizagem definidas, a **B** tem 4, a **C** tem 3 e a **D** tem 2, totalizando 11 metas. O *aluno1* atingiu 1 meta das atividades **A**, **C** e **D**, e 4 da atividade **B**. O *aluno2* atingiu 2 metas da atividade **A**, 1 da **B**, 2 da **C** e 1 da **D**. E o *aluno3* alcançou 2 da atividade **A**, 3 da **B**, nenhuma meta das atividades **C** e **D**. Ao analisar o valor do ritmo individualmente, é possível afirmar que os assuntos das atividades **A**, **C** e **D** precisam ser revisitados pelo Aluno 1. As metas das atividades **B**, **C** e **D** precisam ser mais aprofundadas pelos alunos 2 e 3.

A informação sobre o ritmo de aprendizagem individual orienta o professor em quais aspectos o estudante está com deficiência e em quais conteúdos ou habilidades devem se concentrar naquele momento do curso.

Tabela 4.2: Ritmo de Aprendizagem

Aluno Metas	Ritmo de Aprendizagem			
	A (2)	B (4)	C (3)	D (2)
<i>Aluno 1</i>	0,5	1	0,33	0,5
<i>Aluno 2</i>	1	0,25	0,66	0,5
<i>Aluno 3</i>	1	0,75	0	0

Em uma situação comum, em que o professor acompanha o progresso dos alunos pela nota em avaliação, a avaliação é genérica para todos os alunos que não conseguiram atingir uma nota específica. Analisando os dados acima, percebe-se que os alunos apresentam dificuldades distintas e precisam visitar conteúdos diferentes.

Outro ponto de destaque é que o ritmo de aprendizagem deve ser analisado periodicamente, principalmente para que o professor tenha tempo para intervir de forma efetiva. O aluno 3, por exemplo, estava progredindo bem nas primeiras atividades, mas não conseguiu manter o mesmo ritmo nas últimas. Outros tipos de indicadores podem sinalizar o esforço cognitivo para desenvolver atividades personalizadas. Quanto maior o seu valor, maior o esforço para produzi-lo. Exemplos desses indicadores são:

- **Tamanho do enunciado da questão** - quanto maior o tamanho do enunciado, maior

o esforço exigido para sua elaboração. Pode ser medido pela quantidade de letras ou palavras;

- **Lines of code e outras métricas de qualidade de código** - quanto melhor a qualidade do código, maior o esforço para produzi-lo;
- **Número de casos de testes sugeridos**
- **Tamanho do texto da revisão do código**
- **Número de correções e ajustes sugeridos no texto de uma questão**

4.3 Engajamento Emocional

O engajamento emocional reflete as reações, positivas ou negativas, aos professores, colegas de classe ou ambiente escolar. Acredita-se que sensações positivas de engajamento emocional favorecem para que o aluno crie vínculos com a instituição e influenciam na vontade deles de estudar (Fredricks e McColskey 2012).

Em (Scaico 2018), o componente afetivo da motivação foi mensurado pelo auto-relato do aluno utilizando o instrumento roda das emoções ¹. Esse questionário exibe de maneira gráfica 20 estados emocionais e ajuda a identificar reações emocionais a objetos, eventos e situações (Silva 2011).

Nem todo mundo tem facilidade de identificar emoções, é por isso que a roda das emoções apresenta várias perspectivas para o mesmo estado emocional. A Figura 4.1 exibe uma versão traduzida para o português do instrumento descrito em (Scherer 2005). Os estados emocionais são divididos em 6 grupos centrais: bravo, triste, tranquilo, poderoso, alegre e assustado. Cada grupo possui 6 emoções diretamente relacionadas ao grupo central e cada uma dessas possui outros 2 estados associados. O indivíduo pode seguir de fora para dentro para encontrar o estado emocional que mais se aproxima à sensação que sentiu ao ser submetido a determinado evento ou situação.

Essa tese se concentrou nas sensações do grupo central. Os grupos foram divididos em emoções positivas e negativas, sendo bravo, triste e assustado como negativas, e tranquilo,

¹Tradução do termo em inglês *Geneva Emotion Wheel*

zação do instrumento de auto-relato para aplicação após a realização das atividades. Para mensurar aspectos do comportamento do engajamento, foram apresentados os indicadores dedicação, persistência, hábito de estudo e participação.

Já em relação ao engajamento cognitivo, foi definido o ritmo de aprendizagem, que representa como o estudante está progredindo no curso. Além disso, foram mostrados outros indicadores relacionados ao esforço cognitivo em realizar as atividades personalizadas, como aspectos de qualidade do código, números de testes sugeridos e tamanho do texto da revisão.

Por fim, o uso da roda das emoções foi sugerido por facilitar a identificação de sensações que estimam o engajamento emocional do estudante.

Capítulo 5

Avaliação do Engajamento de Aprendizes de Programação

Esse capítulo apresenta estudos para avaliar o modelo de personalização e aplicar as métricas de engajamento em atividades de estudantes aprendizes de programação. Todos esses estudos foram realizados em um curso de programação introdutória, que chamamos de ambiente de experimentação. Inicialmente, o capítulo descreve esse ambiente e apresenta como o conteúdo é dividido, as práticas pedagógicas e ferramentas utilizadas pelos instrutores.

A Figura 5.1 sumariza as características dos três estudos realizados para avaliar o engajamento de estudantes de programação, sendo dois experimentais e um estudo de caso. Os experimentais são estudos transversais, quantitativos e de laboratório, que avaliam os efeitos do engajamento de colaboradores na realização de atividades personalizadas de programação. Isso quer dizer que foram pesquisas realizadas em um curto intervalo de tempo, gerando dados numéricos em um ambiente controlado de laboratório. O experimento 1 relatado na Seção 5.2 avalia o engajamento do colaborador na realização de atividades personalizadas do tipo criação de exercícios. O experimento 2 apresentado na Seção 5.3 avalia o engajamento na realização dos outros tipos de atividades personalizadas. O segundo experimento também tem um componente de avaliação quantitativa, que serve para compreender a opinião dos estudantes sobre a realização desse novo modelo de atividades.

O estudo de caso foi realizado com a intenção de concretizar as métricas de engajamento de aprendizes de programação com dados da interação do estudante com um ambiente de aprendizagem. Foi uma pesquisa longitudinal e observacional, realizada durante um semes-

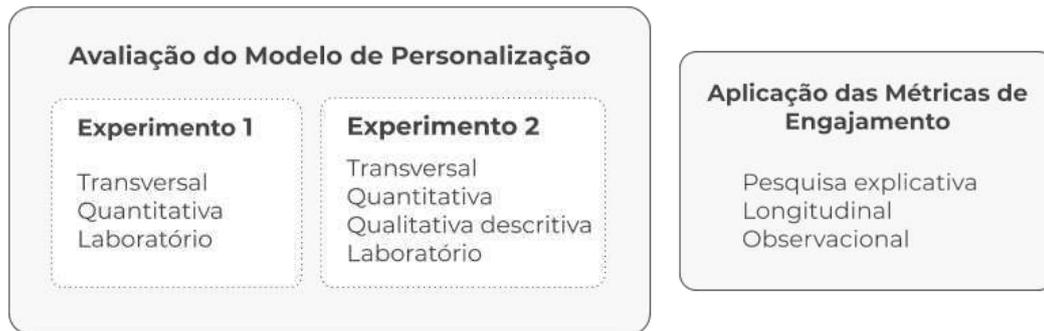


Figura 5.1: Estudos para Avaliação de Engajamento

tre sem intervenção que está registrado na Seção 5.4.

Acrescenta-se ainda que todos os estudos apresentados neste capítulo compõem o projeto Uma Abordagem Colaborativa para Ensino de Introdução à Programação que foi submetido e aprovado pelo Comitê de Ética em Pesquisa da UFCG sob número CAAE 69427916.4.0000.5182 cujo parecer está no Apêndice A.

5.1 Ambiente Experimental

Os estudos foram realizados no âmbito das disciplinas de Programação 1 e Laboratório de Programação 1 do curso de Ciência da Computação da UFCG durante os semestres 2017.2, 2018.2 e 2019.1. Essas duas disciplinas são administradas como apenas uma, com a mesma metodologia de ensino e grupo de docentes, sendo a principal diferença a sala que é destinada às aulas. A disciplina de Programação 1 é ministrada em sala de aula convencional, enquanto Laboratório de Programação 1 tem aulas no laboratório de informática. Dessa forma, durante o texto, será utilizado o termo a *disciplina*, no singular, para se referir às duas disciplinas.

5.1.1 Design da Disciplina

Os professores adotam práticas pedagógicas baseadas na aprendizagem ativa. Isso inclui a aprendizagem individualizada¹, aprendizagem de domínio² e sala de aula invertida³. O aprendizado individualizado respeita as diferenças entre os estudantes e reconhece que cada

¹Tradução dos termos em inglês: *self-paced*

²*mastery learning* e

³*flipped classroom*

sujeito tem um ritmo próprio de aprendizado, com a compreensão sobre o conteúdo em momentos diferentes de outros estudantes da turma. Na aprendizagem de domínio, o conteúdo programático do curso é dividido em pequenas unidades sequenciais. A progressão para a matéria da unidade seguinte só é permitida quando o aluno demonstra ter dominado o conteúdo e as habilidades da unidade atual. Em outros termos, o aluno só estuda a matéria associada à unidade N se, e somente se, demonstrou que conseguiu adquirir o conhecimento abordado na matéria da unidade N - 1. Esse tipo de abordagem ajuda a concentrar a atenção do aluno em um conteúdo específico e evita o acúmulo de dúvidas sobre os assuntos vistos em algum momento passado. O método da sala de aula invertida muda a dinâmica das aulas, porque demanda que os alunos se preparem em casa antes dos professores apresentarem o conteúdo de modo presencial em sala. Assim, as aulas presenciais tornam-se orientadas pelas dúvidas e são destinadas ao desenvolvimento de atividades práticas.

Com a combinação dessas estratégias, a turma pode ficar dividida em grupos que estão estudando assuntos diferentes, mesmo estando matriculados no mesmo curso. Dessa forma, o aluno pode direcionar seus esforços para compreender o assunto que tem dificuldade sem se preocupar com o andamento dos outros alunos da turma. O conteúdo programático engloba os conceitos essenciais de um curso de programação organizado em 10 unidades. Sua organização está apresentada na Tabela 5.1.

Tabela 5.1: Unidades de Conteúdo

Unidade	Conteúdo
1	Conceitos elementares de programação, incluindo comandos de entrada e saída de dados
2	Escrita de programas utilizando os conceitos aprendidos na unidade 1
3	Comandos condicionais e funções simples
4	Tratamento de strings e laços simples
5	Laços indefinidos
6	Funções
7	Estruturas de dados e algoritmos básicos de ordenação
8	Outras funções relacionadas a estruturas de dados lineares (listas, tuplas e strings)
9	Sequências e matrizes
10	Mapas

5.1.2 Ferramentas

O suporte tecnológico ajuda a garantir o acompanhamento, a orientação e o estudo de forma individualizada para cada aluno. Apresentamos as principais ferramentas utilizadas nas disciplinas e que influenciaram, de alguma forma, o desenvolvimento dos nossos estudos.

O TST *online* é um ambiente que gerencia a liberação das questões para o aluno. Já o TST é um conjunto de *scripts* executados por linha de comando que automatizam o teste e a verificação dos exercícios de programação que são acessados pelo TST online (Guerrero 2019). Ambos são utilizados em conjunto para dar suporte à realização de exercícios e provas e o acompanhamento do desempenho do aluno.

O Slack é uma plataforma de comunicação que oferece recursos no estilo IRC, incluindo canais de bate-papo organizados por tópico, grupos privados e mensagens diretas (Technologies 2019). Os professores usam a ferramenta como uma sala de aula virtual, onde apenas as pessoas envolvidas com a disciplina podem participar. Essa ferramenta permite que a comunicação ocorra a qualquer hora e em qualquer dia da semana. Isso quer dizer que todos podem ver quem está online, conversar e discutir assuntos relativos à disciplina com qualquer participante, incluindo professores, tutores e os demais alunos. Mensagens enviadas quando o usuário não está online ficam disponíveis para acesso em um outro momento. Portanto, é possível acompanhar a troca de mensagens mesmo se não estiver online todo o tempo. O Slack foi uma ferramenta importante no desenvolvimento desses estudos porque permitiu acompanhar a interação entre os alunos e facilitou a comunicação com eles.

A Figura 5.2 apresenta uma das discussões que acontecem entre os alunos pelo Slack. Com esse exemplo, pode-se perceber a importância da troca de informações entre os alunos, já que a dúvida de um também pode ser a do outro aluno. O nome e as imagens dos envolvidos foram ocultados para garantir a anonimização dos dados.

5.2 Engajamento do Colaborador na Criação de Exercício

Nesta seção, apresenta-se um estudo quantitativo e experimental, de forma controlada, que investiga o engajamento do estudante colaborador (o que desenvolve novas atividades personalizadas e realiza atividades criadas por outros estudantes).

O experimento foi realizado de forma transversal e controlada. Esse tipo de estudo é

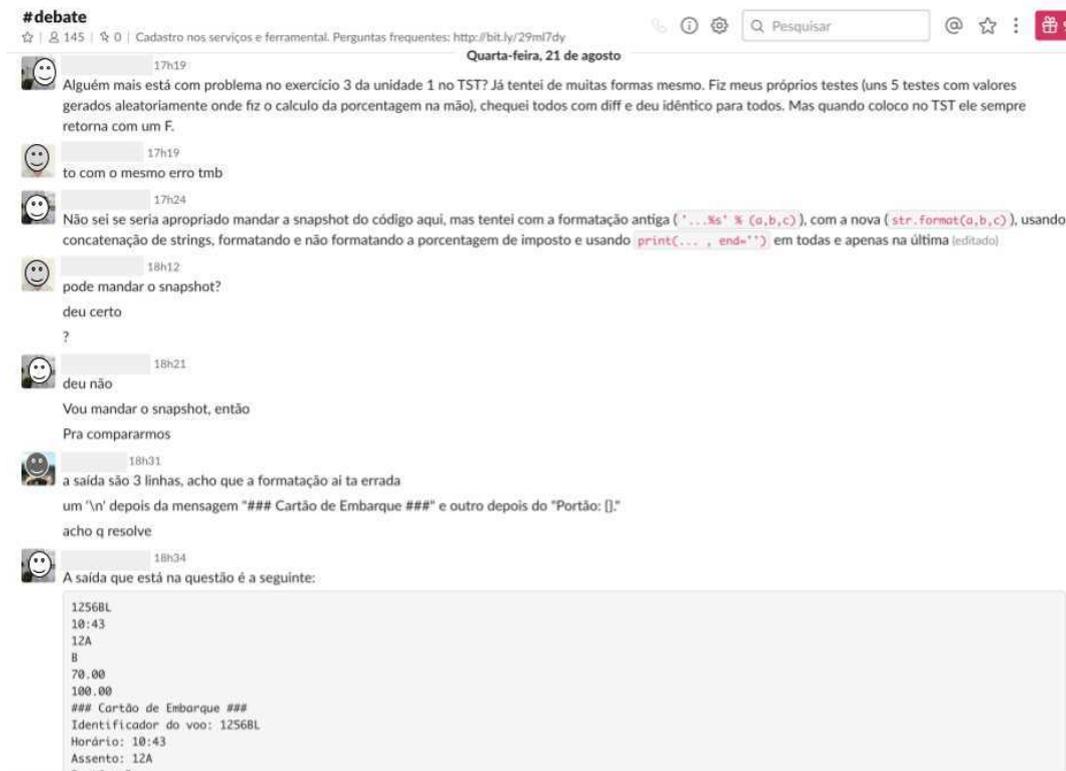


Figura 5.2: Interação entre Alunos via Slack

caracterizado por ser executado durante um curto intervalo de tempo em um laboratório reservado. O procedimento foi realizado dessa forma para garantir o isolamento do tratamento e reduzir a possibilidade de que intervenções na disciplina pudessem influenciar nos resultados do experimento.

5.2.1 Materiais e Métodos

O principal objetivo do experimento foi investigar quais os impactos no engajamento de colaboradores no desenvolvimento de atividades de criação de exercícios e resolução daqueles que são criados por outros estudantes.

Desse modo, investigamos as seguintes hipóteses nula (H_0) e alternativa (H_1):

H_0 : Não há melhoria no engajamento de estudantes colaboradores quando contribuem com novos exercícios para o aprendizado de programação de outros estudantes.

H_1 : Há melhoria no engajamento de estudantes colaboradores quando contribuem com novos exercícios para o aprendizado de programação de outros estudantes.

Para investigar essas hipóteses, as seguintes variáveis foram definidas:

- **Dependente (resultado):** engajamento de estudantes colaboradores em atividades de criação de novos exercícios. Fica estabelecido que o engajamento comportamental deve ser representado pela dedicação e persistência. O aspecto cognitivo do engajamento deve ser representado pela classificação do nível de dificuldade da questão criada pelo participante. Já o aspecto emocional deve ser representado pelo auto-relato de sensações do participante.
- **Independente (preditiva):** estar ciente sobre o resultado da atividade ser efetivamente utilizada por outras pessoas.
- **Intrínseca antecedente:** unidade de conteúdo que o participante estava estudando no instante da realização do experimento.

Dos aspectos de comportamento do engajamento, o estudo se concentra em dois:

- **Dedicação** - concretizada pelo número de questões criadas e intervalo de tempo entre os envios de questões;
- **Persistência** - definida pela duração da sessão de estudo;

O **engajamento cognitivo** foi estimado pelo nível de dificuldade dos exercícios. Utilizou-se essa medida por se acreditar que quanto maior o nível de dificuldade de uma questão, maior o esforço cognitivo para sua criação.

Foi utilizado um questionário de auto-relato para medir o nível de engajamento emocional, baseado em sensações, após a realização de atividades personalizadas.

Diante dessas métricas que foram utilizadas para concretizar o engajamento dos participantes no estudo, as hipóteses H_0 e H_1 em outras:

- H_0^1 : A média de questões produzidas pelo grupo experimental é menor ou igual a de controle.
- H_1^1 : A média de questões produzidas pelo grupo experimental é maior que a de controle.
- H_0^2 : A média do intervalo de tempo entre as submissões do grupo experimental é menor ou igual a de controle.

- H_1^2 : A média do intervalo de tempo entre as submissões do grupo experimental é maior que a de controle.
- H_0^3 : A média da sessão de estudos do grupo experimental é menor ou igual a de controle.
- H_1^3 : A média da sessão de estudos do grupo experimental é maior que a de controle.
- H_0^4 : O número de questões difíceis produzidas pelo grupo experimental é menor ou igual ao de controle.
- H_1^4 : O número de questões difíceis produzidas pelo grupo experimental é maior que o de controle.
- H_0^5 : O número de questões fáceis produzidas pelo grupo experimental é maior que o de controle.
- H_1^5 : O número de questões fáceis produzidas pelo grupo experimental é menor que o de controle.
- H_0^6 : A média de sensações positivas sentidas pelo grupo experimental é menor ou igual a de controle.
- H_1^6 : A média de sensações positivas sentidas pelo grupo experimental é maior que a de controle.
- H_0^7 : A média de sensações negativas sentidas pelo grupo experimental é maior que a de controle.
- H_1^7 : A média de emoções negativas sentidas pelo grupo experimental é menor que a de controle.

Antes da aplicação do questionário, foi criada uma situação falsa de que os professores estavam planejando mudanças no estilo da disciplina de programação. Acrescentou-se que a principal mudança seria a redução do número de exercícios liberados pelo TST, atribuindo aos alunos a responsabilidade de criarem seus próprios exercícios.

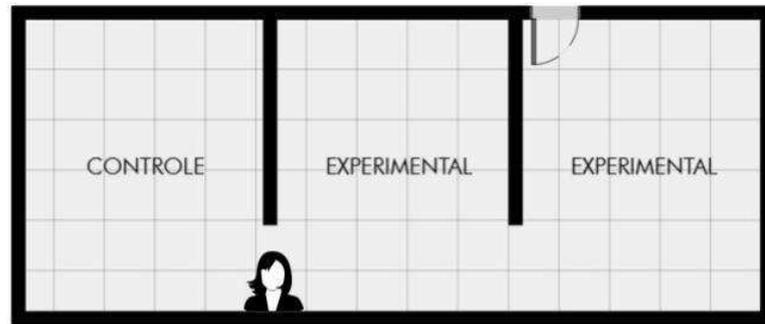


Figura 5.3: Organização da Sala Durante o Experimento

Em seguida, os participantes foram informados que deveriam realizar um simulado para que os professores analisassem a nova metodologia que seria aplicada no semestre seguinte.

Os participantes foram divididos em dois grupos fisicamente separados. Os grupos de controle e experimental estavam setorizados e instruídos a não conversar ou fazer barulho durante o simulado. A Figura 5.3 apresenta a planta baixa do laboratório onde foi realizado o experimento. Por falta de espaço, foi utilizado um único laboratório que possui 3 salas internas subdivididas por uma parede. Para evitar vazamento de informações entre os grupos e reduzir a interação entre os participantes, uma inspetora posicionada na porta dividia os dois grupos.

A distribuição dos participantes em cada grupo foi realizada de maneira aleatória e balanceada, de acordo com a unidade de conteúdo que estavam vendo no momento do experimento. Como o estudo foi realizado com o semestre em andamento, passados 54 dias letivos, não foi possível incluir outros critérios como o desempenho final na disciplina para balancear os participantes dos grupos.

Diferente da rotina de estudos comum dos participantes na disciplina, em que os exercícios são disponibilizados via TST, foi utilizado um serviço cujo acesso aos dados fosse mais simples. Diante disso, utilizou-se o serviço do Google Drive (GDrive) que fornece serviços que facilitam a criação e edição de formulários e coleta de dados (Cloud 2020). Além disso, organiza os dados das resposta respostas do formulário em uma planilha eletrônica, onde é possível acessar metadados das submissões, incluindo data e horário que foram utilizados para concretizar as métricas de engajamento dos participantes do experimento.

Foram elaborados dois formulários no GDrive, sendo que o primeiro representou o si-

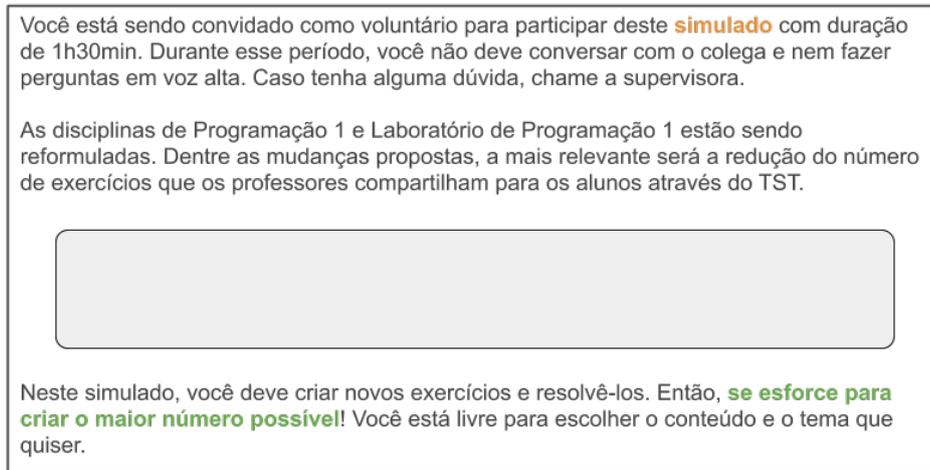


Figura 5.4: Informação Compartilhada com os Participantes do Experimento I

mulado para submissão de atividades e o segundo com o questionário para aplicação após a realização do simulado. Os participantes receberam um endereço para acesso online ao simulado correspondente aos seus respectivos grupos. O enunciado do simulado orientava o que deveria ser realizado e o tempo de duração. A Figura 5.4 apresenta o texto que foi repassado para os dois grupos. O quadro cinza que se encontra no meio da imagem representa o que difere a informação passada de um grupo para outro. Esse quadro foi substituído pela mensagem da Figura 5.5 para o grupo de controle. Com essa informação, é possível justificar porque estava sendo realizado um simulado com um tipo de exercício diferente dos que eles estavam habituados a desenvolver na disciplina. Já a Figura 5.6 apresenta a mensagem que substitui o quadro cinza no grupo experimental, no intuito de informá-los que os novos exercícios seriam, de fato, inseridos no TST e utilizados na disciplina para ajudar outras pessoas a aprender a programar.

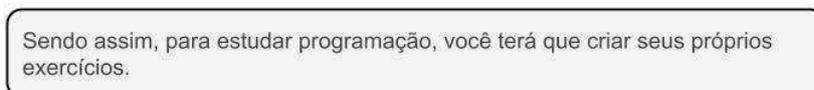


Figura 5.5: Enunciado Exibido para o Grupo de Controle

Ao final do simulado, os participantes foram direcionados para segundo questionário cujo conteúdo encontra-se na Figura 5.7. Esse formulário apresenta as sensações na primeira coluna e as intensidades *nenhum pouco*, *pouco*, *razoável*, *intenso* e *muito intenso* nas colunas seguintes. Os participantes foram orientados a assinalar a coluna que melhor representa a

Sendo assim, para estudar programação você terá que criar seus próprios exercícios. As novas questões serão incorporadas ao material da disciplina, inseridas no TST e ajudarão os colegas a aprender a programar.

Figura 5.6: Enunciado Exibido para o Grupo Experimental

intensidade de como eles se sentiram ao criar novos exercícios.

1) Como você se sentiu ao saber que vai precisar criar exercícios? *

Assinale a coluna que representa a intensidade da emoção. Caso haja alguma emoção ou sentimento que você queira destacar e que não esteja nas linhas abaixo, por favor descrever no espaço da última pergunta deste questionário.

	Nem um pouco	Pouco	Razoável	Intenso	Muito intenso
Interessado(a)	<input type="radio"/>				
Orgulhoso(a)	<input type="radio"/>				
Feliz	<input type="radio"/>				
Satisfeito(a)	<input type="radio"/>				
Admirado(a)	<input type="radio"/>				
Envergonhado(a)	<input type="radio"/>				
Estimulado(a)	<input type="radio"/>				
Cauteloso(a)	<input type="radio"/>				
Chateado(a)	<input type="radio"/>				
Incluído(a)	<input type="radio"/>				
Motivado(a)	<input type="radio"/>				
Importante	<input type="radio"/>				
Poderoso(a)	<input type="radio"/>				
Influente	<input type="radio"/>				
Útil	<input type="radio"/>				
Envolvido(a)	<input type="radio"/>				
Produtivo(a)	<input type="radio"/>				

Figura 5.7: Questionário Sobre a Intensidade de Sensações

5.2.2 Resultados

Estudantes matriculados na disciplina do ambiente de experimentação no semestre 2018.2 participaram desse experimento. Todos os alunos da turma foram convidados a participar, porém, dos cerca de 100 alunos, apenas 27 deles atenderam ao convite. Acreditamos que, com essa estratégia de seleção, apenas estudantes que já são naturalmente engajados aceitaram o convite.

A Tabela 5.2 sumariza as características de ambos os grupos com informações sobre a média de unidade de conteúdo média e o número de participantes de cada um.

Tabela 5.2: Características dos Grupos de Controle e Experimental

Grupo	Unidade de conteúdo (Média)	Núm. de Participantes
<i>Controle</i>	6,38	13
<i>Experimental</i>	6,5	14

A Figura 5.8 apresenta o gráfico boxplot que compara o número de exercícios de cada grupo. Usamos o teste-t de cauda única para comparar a média de criação de questões e analisar H_0^1 : **A média de questões produzidas pelo grupo experimental é menor ou igual a de controle.** Com o $p\text{-value} = 0.8023$, o resultado do teste-t não permite rejeitar a H_0^1 .

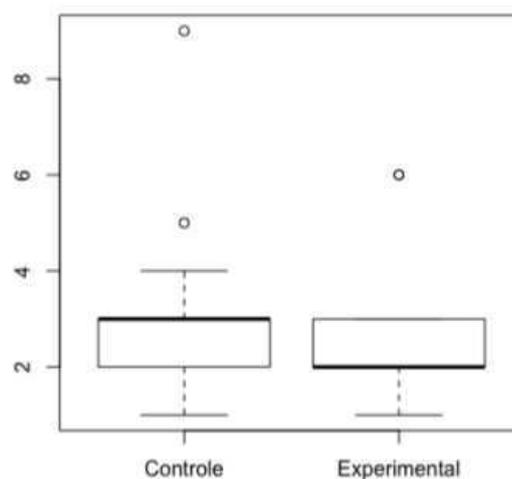


Figura 5.8: Número de Exercícios Criados por Estudantes

Comparou-se também a produção individual dos alunos de cada grupo. A Figura 5.9 exibe o número de questões geradas por cada participante do grupo de controle e experimental. Os participantes do grupo de controle produziram entre 1 a 9 exercícios, com média de

3,3 questões por aluno. Enquanto o experimental produziu entre 1 e 6 questões, com média de 2,7 questões por aluno.

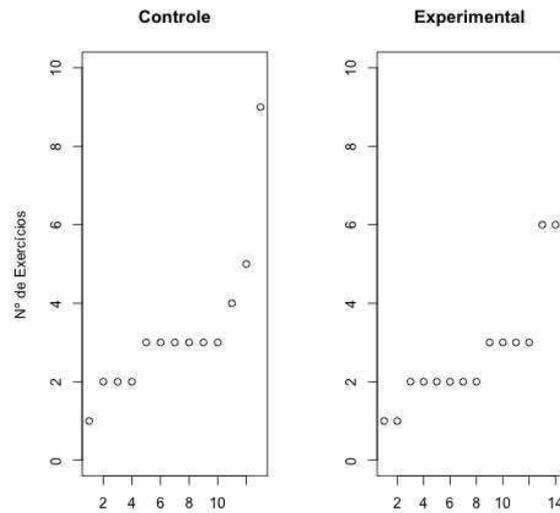


Figura 5.9: Número de Exercícios Criados por Autor

Usamos o intervalo de tempo entre as submissões para investigar se o grupo experimental foi mais rápido do que o de controle e usamos o teste-t de cauda única para investigar a hipótese H_0^2 : **A média do intervalo de tempo entre as submissões do grupo experimental é menor ou igual a de controle.** Com $p\text{-value} = 0.4495$, não podemos rejeitar H_0^2 . Isso também se reflete pela análise visual da Figura 5.10, onde não há indícios suficientes para constatar diferenças entre as médias.

Em relação à duração da sessão de estudos, investigamos a hipótese H_0^3 : A média da sessão de estudos do grupo experimental é menor ou igual a de controle. A intenção com essa hipótese era de observar se houve diferença significativa entre os grupos em relação ao tempo que os alunos se mantiveram na atividade. Fazendo uso do teste-t de cauda única, com $p\text{-value} = 0.9195$, não foram encontradas evidências suficientes para rejeitar H_0^3 .

No simulado, além de criar o novo exercício, o participante precisava classificar o nível de dificuldade do material que gerou. A Figura 5.11 apresenta a classificação do nível de dificuldade das questões criadas pelos participantes do grupo de controle e Figura 5.12 a classificação do grupo experimental. Cerca de 39% das questões do grupo de controle foram classificadas com nível médio ou difícil de dificuldade. Enquanto, aproximadamente, 48% das questões do grupo experimental não foram consideradas fácil, quase 10% a mais.

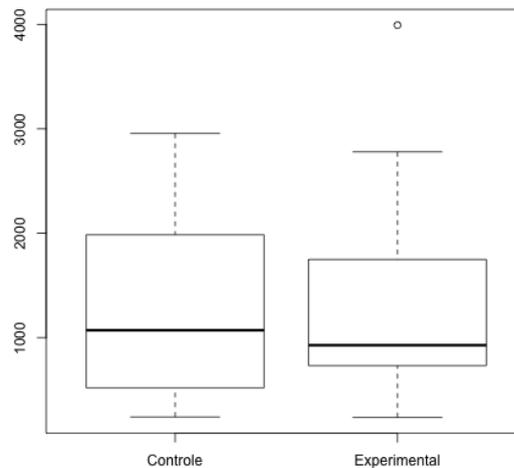


Figura 5.10: Intervalo entre as Submissões de Questões

Ao mesmo tempo que participantes do controle produziram quase 60% de questões fáceis, confrontados com os 53% do grupo experimental.

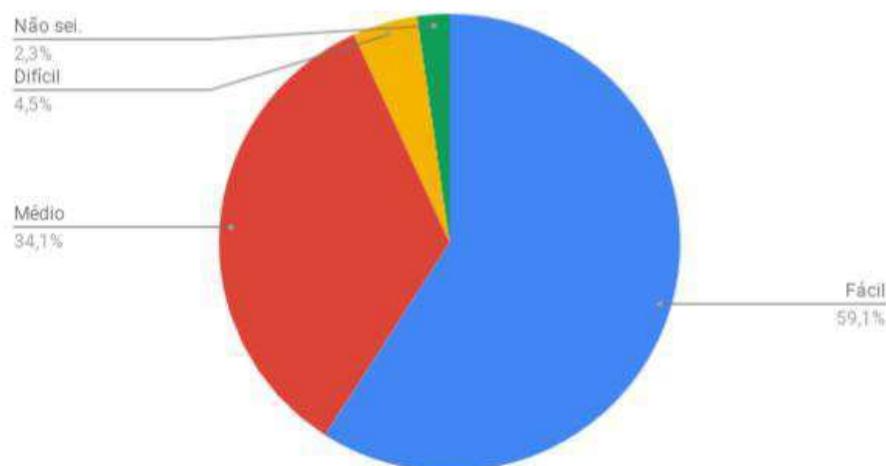


Figura 5.11: Dificuldade dos Exercícios Gerados pelo Grupo de Controle

Em relação às sensações, a Figura 5.13 apresenta a intensidade com que os participantes se sentiram envolvidos, estimulados, incluídos, interessados e motivados. Não foram identificadas diferenças significativas entre as intensidades das sensações envolvido, interessado e motivado. Enquanto cerca de 86% dos participantes do grupo experimental e 62% do de controle disseram estar se sentindo estimulados. Cerca de 7% dos participantes do grupo experimental e 15% do de controle, que corresponde a cerca do dobro do valor, se sentiram

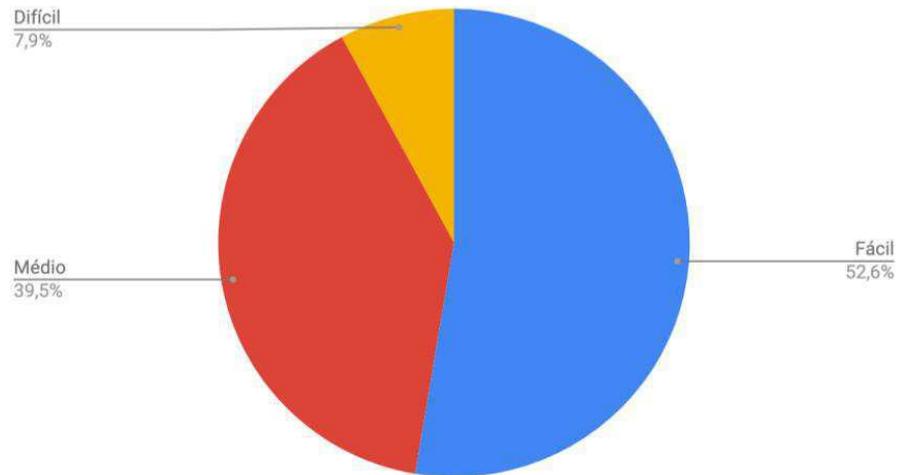


Figura 5.12: Dificuldade dos Exercícios Gerados pelo Grupo Experimental

nenhum pouco ou pouco incluídos.

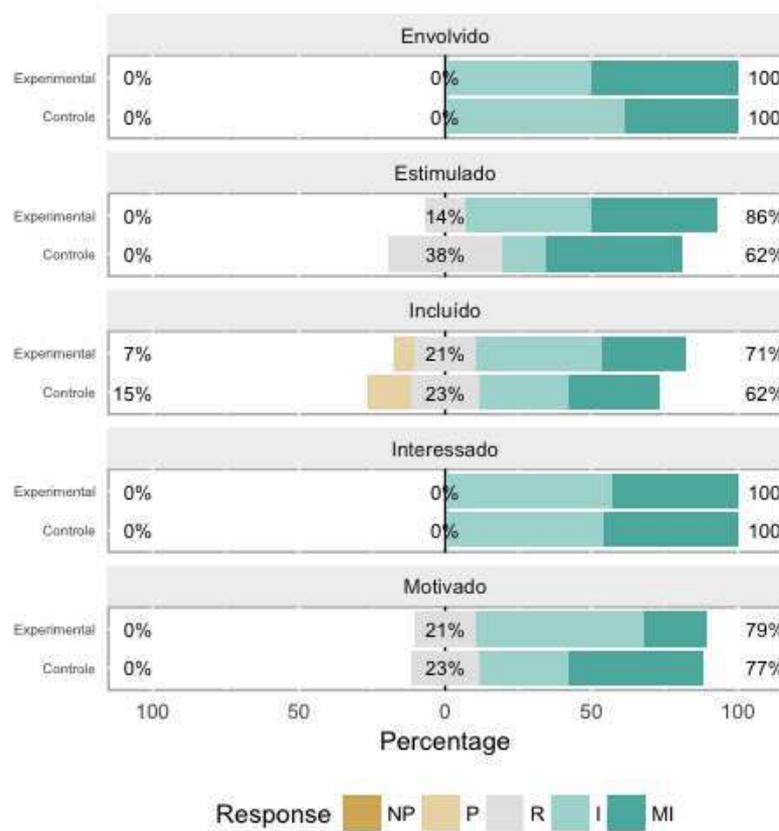


Figura 5.13: Como o Participante se Sentiu Envolvido, Estimulado, Incluído, Interessado e Motivado)

A Figura 5.14 exhibe como os participantes se sentiram admirados, felizes, orgulhosos e

satisfeitos. O grupo de controle se sentiu intensa ou muito intensamente mais admirados e orgulhosos. Enquanto para as sensações de feliz e satisfeito, não houve relato de intensidade nenhum e nenhum pouco apenas para o grupo experimental.

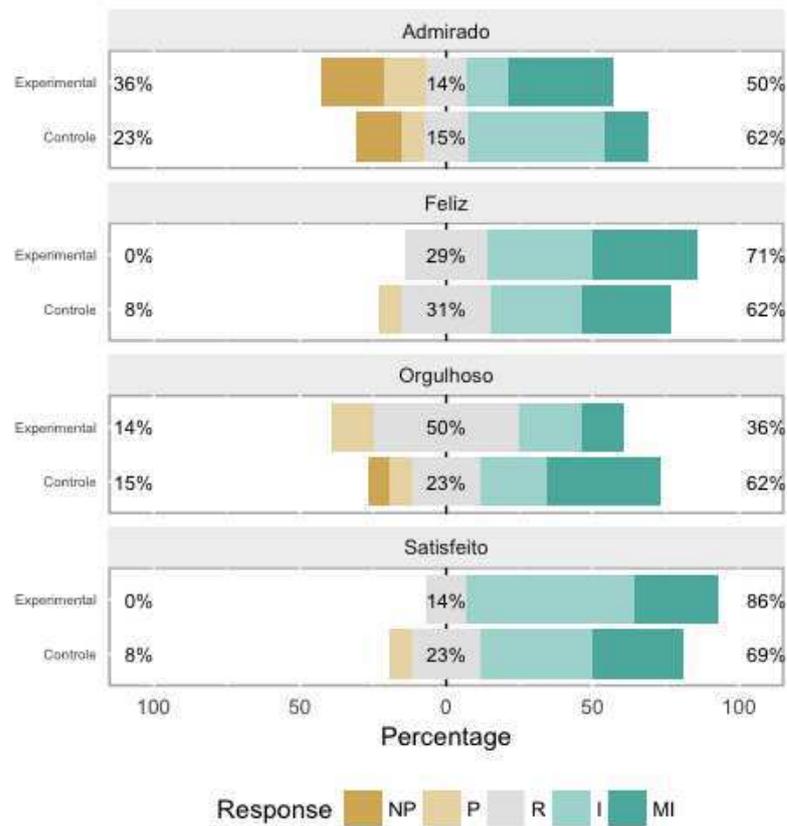


Figura 5.14: Como o Participante se Sentiu Admirado, Feliz, Orgulhoso e Satisfeito

A Figura 5.15 exibe como os grupos relataram as sensações influente, produtivo e útil. O grupo experimental se sentiu mais influente e produtivo. Enquanto o grupo de controle se sentiu mais útil.

Por fim, a Figura 5.16 exibe como os grupos relataram as sensações cauteloso, chateado e envergonhado. Nenhum aluno do grupo experimental se sentiu chateado. Ao mesmo tempo que 29% do grupo experimental se sentiu nenhum ou nenhum pouco cautelosos, o que aconteceu com 8% do grupo de controle. Cerca de 71% do experimental e 54% do grupo de controle se sentiu envergonhado.

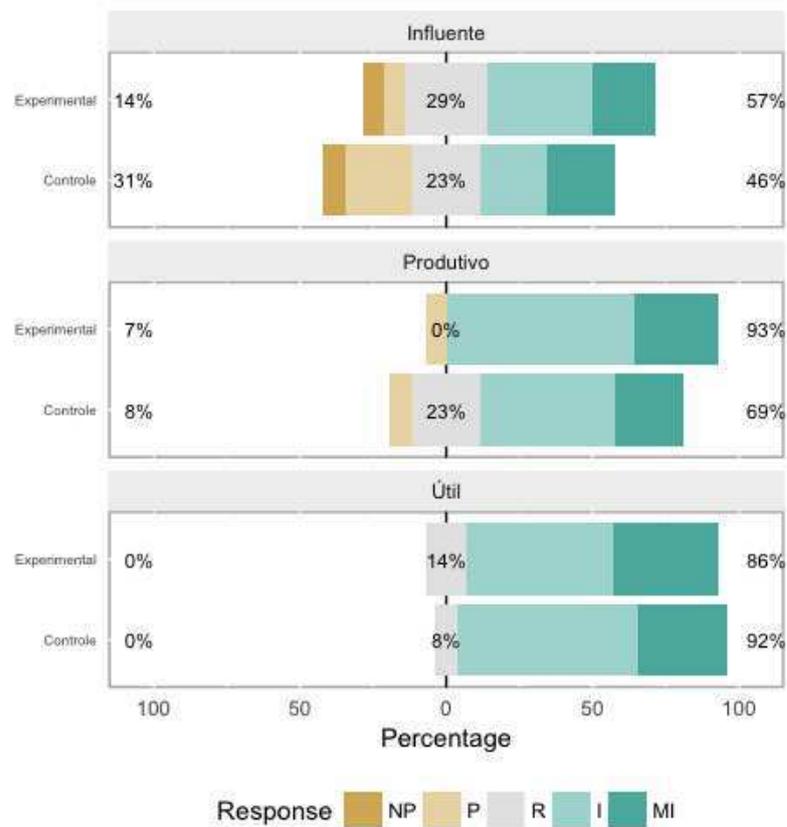


Figura 5.15: Como o Participante se Sentiu Influente, Produtivo e Útil

5.3 Engajamento do Colaborador na Realização de Atividades Personalizadas

Nessa seção, foram estudados os efeitos no engajamento dos colaboradores ao realizar os outros tipos de atividades personalizadas. Foram utilizados os métodos quantitativo experimental e qualitativo descritivo para investigar quais aspectos do engajamento do colaborador são afetados pela realização das atividades personalizadas. Esse estudo foi conduzido de maneira transversal e com ambiente de laboratório controlado.

Assim como no experimento anterior, a decisão sobre o método que conduz o estudo foi guiada pela dificuldade em isolar o tratamento sem que o resultado fosse influenciado por possíveis intervenções na disciplina, caso o experimento fosse longitudinal.

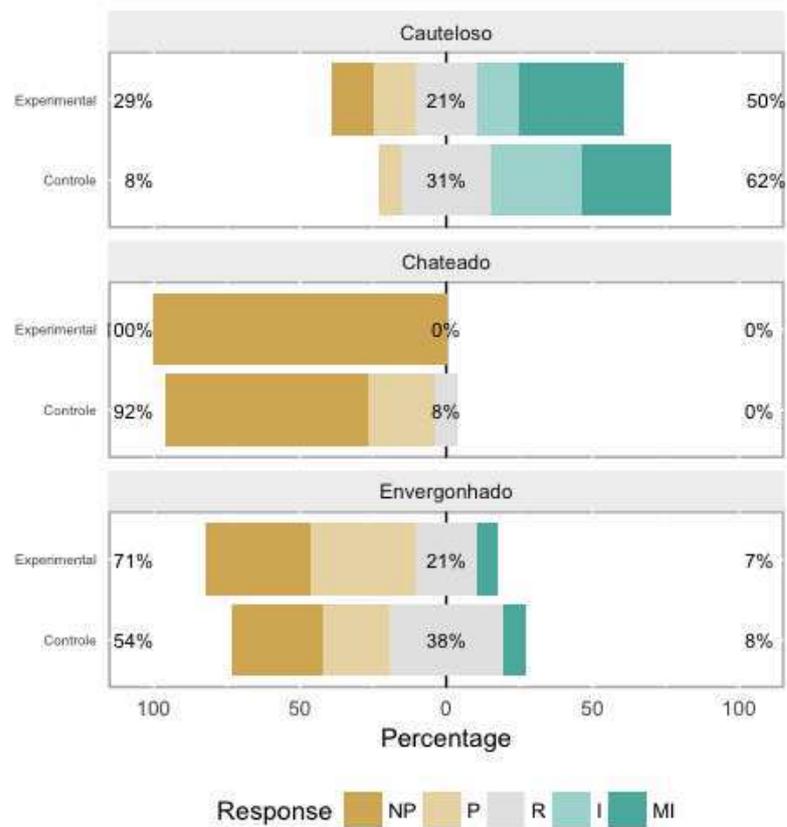


Figura 5.16: Como o Participante se Sentiu Cauteloso, Chateado e Envergonhado

5.3.1 Materiais e Métodos

Esse experimento teve como objetivo principal estudar os efeitos no engajamento do estudante colaborador que realiza todos os tipos de atividades personalizadas de programação. Além disso, buscou conhecer a opinião dos estudantes sobre a realização desse tipo de atividade. Desse modo, esse estudo experimental foi guiado pelas seguintes hipóteses nula e alternativa:

H_0 : Não há melhoria no engajamento de colaboradores que realizam atividades personalizadas de programação.

H_1 : Há melhoria no engajamento de colaboradores que realizam atividades personalizadas de programação.

Para investigar essas hipóteses, definimos as seguintes variáveis:

- **Dependente (resultado):** engajamento de estudantes colaboradores em atividades personalizadas de programação. Estabelecemos que engajamento comportamental

deve ser representado pela dedicação e persistência. O aspecto cognitivo do engajamento deve ser representado pelo tamanho dos enunciados, número de testes criados e o tamanho do texto da revisão de código. Já o emocional deve ser mensurado pelo auto-relato de sensações do participante por meio do instrumento Roda das Emoções.

- **Independente (preditiva):** a ciência de que a contribuição é direcionada para um personagem real.
- **Intrínseca antecedente:** número de faltas, ritmo de aprendizagem e a unidade de conteúdo do participante no instante da realização do experimento.

Esse estudo foi concentrado em dois aspectos de comportamento do engajamento:

- **Dedicação** - concretizada pelo número de questões criadas e intervalo de tempo entre os envios de questões;
- **Persistência** - representada pelas submissões que aconteceram depois do prazo *obrigatório* de 1h30min para realização do simulado.

Foram utilizadas as métricas do Experimento 1 para avaliar a dedicação do engajamento comportamental. A persistência foi concretizada como os envios após o período pré-definido como obrigatório para realização das atividades. Considerou-se as características dos demais tipos de atividades personalizadas como indicadores de engajamento cognitivo, incluindo tamanho dos enunciados, número de casos de testes sugeridos e o tamanho da revisão de código. Como foi observada a dificuldade dos participantes em identificar sensações no experimento apresentado na seção anterior, optou-se por utilizar o instrumento Roda das Emoções para identificar o engajamento emocional. Os participantes foram orientados a assinalar as 3 sensações mais experimentadas durante a realização das atividades. Essas concretizações das métricas de engajamento foram utilizadas para decompor H_0 e H_1 em outras hipóteses:

- H_0^1 : A média de questões produzidas pelo grupo experimental é menor ou igual a de controle.
- H_1^1 : A média de questões produzidas pelo grupo experimental é maior que a de controle.

- H_0^2 : A média do intervalo de tempo entre as submissões do grupo experimental é menor ou igual a de controle.
- H_1^2 : A média do intervalo de tempo entre as submissões do grupo experimental é maior que a de controle.
- H_0^3 : A média da duração da sessão de estudos do grupo experimental é menor ou igual a de controle.
- H_1^3 : A média da sessão de estudos do grupo experimental é maior que a de controle.
- H_0^4 : O tamanho das questões produzidas pelo grupo experimental é menor ou igual ao de controle.
- H_1^4 : O tamanho das questões produzidas pelo grupo experimental é maior que o de controle.
- H_0^5 : O número de casos de testes produzidos pelo grupo experimental é menor ou igual ao de controle.
- H_1^5 : O número de casos de testes produzidos pelo grupo experimental é maior que o de controle.
- H_0^6 : O tamanho das revisões produzidas pelo grupo experimental é menor ou igual ao de controle.
- H_1^6 : O tamanho do texto das revisões produzidas pelo grupo experimental é menor ou igual ao de controle.
- H_0^7 : A média de sensações positivas sentidas pelo grupo experimental é menor ou igual a de controle.
- H_1^7 : A média de sensações positivas sentidas pelo grupo experimental é maior que a de controle.
- H_0^8 : A média de sensações negativas sentidas pelo grupo experimental é maior ou igual que a de controle.

- H_1^8 : A média de sensações negativas sentidas pelo grupo experimental é menor que a de controle.

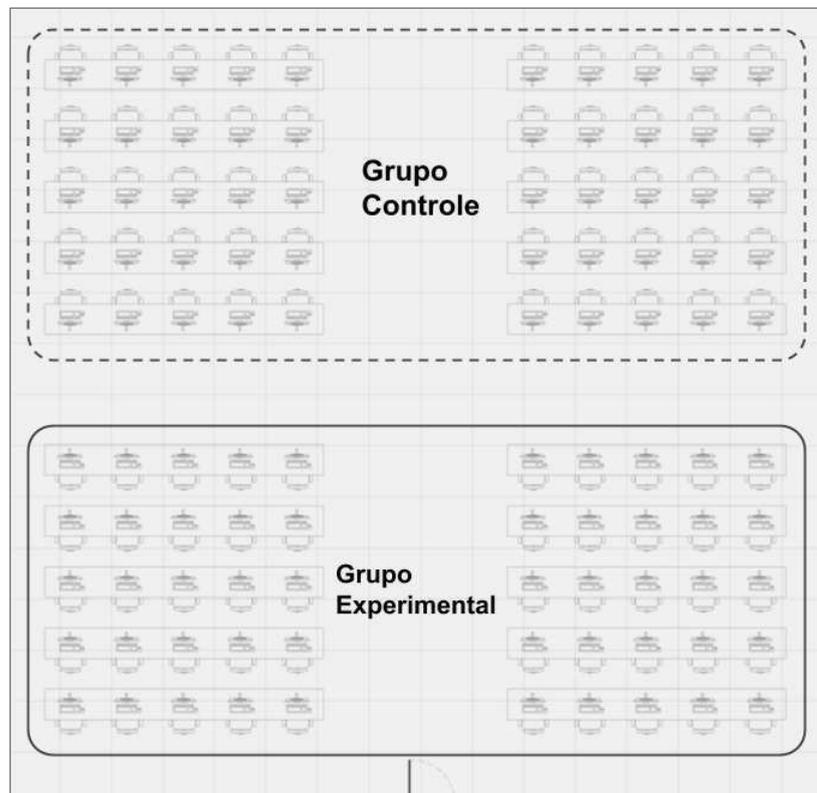


Figura 5.17: Organização da Sala Durante o Experimento

Os participantes foram divididos em dois grupos, controle e experimental, fisicamente separados em setores e instruídos a não conversar ou fazer barulho durante o simulado. A Figura 5.17 apresenta a planta baixa do laboratório onde foi realizado o experimento. Por falta de espaço, utilizamos um único laboratório que possui 4 blocos de computadores separados por corredor. Para evitar vazamento de informações entre os grupos e reduzir a interação entre os participantes, 5 inspetores foram posicionados em cada setor.

A distribuição dos participantes em cada grupo foi realizada de maneira aleatória e balanceada de acordo com a unidade de conteúdo, ritmo de aprendizagem e número de faltas no momento do experimento. Como o estudo foi realizado com o semestre em andamento, passados 48 dias letivos, não foi possível incluir outros critérios como o desempenho final na disciplina para balancear os participantes dos grupos. Em relação às características dos participantes dos grupos controle e experimental, na Tabela 5.3 pode ser observado que eles estão balanceados nos indicadores média de faltas, unidade atual, ritmo de aprendizagem e

número de participantes. Os participantes do grupo de controle apresentavam, em média, 1,2 faltas e o experimental tinha média de 1,1 faltas. Em se tratando da unidade de conteúdo que estavam estudando no dia da realização do experimento, foi observado que o grupo de controle estudava 6,6 em média, e o experimental 6,4. A média do ritmo de aprendizagem do controle era 1,28, e 1,29 no experimental. O grupo experimental tinha 1 participante a mais que o grupo de controle.

Tabela 5.3: Características dos Grupos de Controle e Experimental

Grupo	Núm. de Faltas	Unidade	Ritmo de Aprendizagem	Quantidade de Participantes
<i>Controle</i>	1,2	6,6	1,28	36
<i>Experimental</i>	1,1	6,4	1,29	37

Os participantes receberam a orientação de responder às questões de um simulado que continha questões em estilos diferentes dos que eles estavam habituados. Também foi explicado que o simulado deveria ser acessado em um ambiente diferente do de costume para responder exercícios. A informação foi dada porque, assim como no experimento descrito na seção anterior, o GDrive foi utilizado. Além da orientação verbal, também foram entregues documentos impressos com as recomendações que podem ser consultadas nos Apêndices C e D.

Ambos os grupos receberam a orientação da Figura 5.18. Enquanto o grupo experimental recebeu a informação que está na Figura 5.19 que sugere que o resultado da intenção deve servir para ajudar alguém com dificuldade.

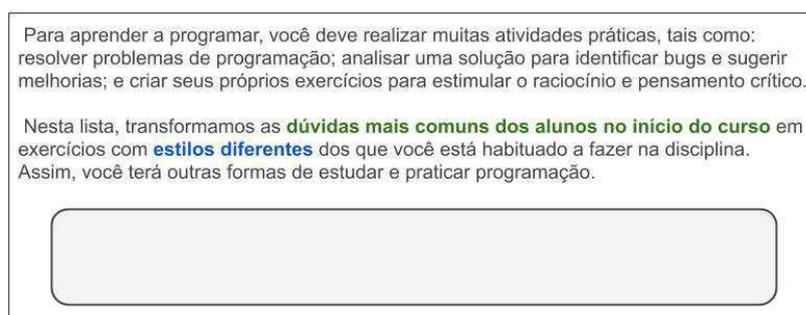


Figura 5.18: Informação Compartilhada com os Participantes do Experimento II

Os alunos foram orientados a responder o simulado com uma lista de exercícios durante 1h30min. Mas havia a possibilidade de ultrapassar esse tempo, caso fosse requisitado.

Assim, você terá outras formas de estudar, praticar programação e **ajudará quem está com dificuldade.**

Figura 5.19: Enunciado Exibido Para o Grupo Experimental no Segundo Experimento

O simulado de ambos os grupos engloba o mesmo número de exercícios, sendo 3 exemplos de cada tipo de atividade personalizada. Isso quer dizer que existem 3 questões de criação de exercícios, 3 de revisão de código, 3 de propostas de testes e 6 de resolução de problemas criados por estudantes. As atividades do simulado podem ser consultadas nos Apêndices E e F.

5.3.2 Resultados

Esse estudo experimental foi realizado no 54º dia do semestre letivo 2019.1. Dos quase 100 alunos matriculados na disciplina naquele semestre, 84 participaram do estudo. Uma parcela pequena da turma não pôde participar do estudo porque não assinou o termo de consentimento que autorizava a participação no projeto de pesquisa ou faltaram à aula no dia que o experimento foi realizado.

A Tabela 5.4 apresenta as características dos participantes em relação aos indicadores de engajamento comportamental e cognitivo. No momento da realização do experimento, os alunos estavam, em média, cursando a unidade 8 da disciplina e tinham apenas 1 falta. Eles já tinham resolvido, em média, cerca de 134 atribuições e estavam avançando no conteúdo com um ritmo de aprendizagem de 1,6 unidades por mini-teste. No momento do experimento, alguns alunos estavam reprovados por falta e outros já tinham concluído as 10 unidades. Ambos os casos não foram incluídos neste estudo.

Tabela 5.4: Características dos Participantes

Média	Faltas	Unidade Corrente	Atribuições Concluídas	Ritmo
<i>Participantes</i>	1	8	134	1,6

Engajamento Comportamental

Nessa seção, apresenta-se uma comparação dos dados produzidos pelos grupos de controle e experimental em termos de aspectos do engajamento que envolvem o comportamento dos estudantes ao realizar as atividades personalizadas.

A Tabela 5.5 exibe a quantidade de questões resolvidas pelos participantes durante o experimento. Percebeu-se que o comportamento de ambos é semelhante, tanto que ambos possuem o mesmo valor de mediana (13) e o valor de diferença entre as médias é muito pequeno (0,03). O que difere os grupos é o comportamento da maioria dos alunos de cada grupo, visto que os de controle responderam entre 12 e 14, enquanto os do experimental entre 10 e 15. O teste t foi utilizado para investigar se há diferença entre os valores do grupo de controle e experimental. Com $p\text{-value} = 0.9427$ e intervalo de confiança de 95%, não é possível afirmar que há diferença significativa entre o número de respostas.

Tabela 5.5: Questões Concluídas Durante o Experimento

Grupo	Mín.	Média	Mediana	Máx.
<i>Experimental</i>	8	12,59	13	15
<i>Controle</i>	4	12,56	13	15

Com a intenção de conhecer o interesse dos alunos por atividade cuja resposta não exige exclusivamente codificação, analisamos o número de respostas para cada tipo de questão. A Tabela 5.6 exibe a média do número de respostas para cada tipo de atividade. Mais uma vez, o teste t foi usado para compará-las, com intervalo de confiança de 95% e $p\text{-values} = 0.2487$ (criação), 0.2924 (revisão), 0.2964 (proposta de testes) e 0.6931 (resolução de exercícios), não pode-se dizer que há diferença significativa entre as médias.

Tabela 5.6: Questões Concluídas por Tipo

Média - Mediana	Criação de Exercícios	Revisão de Código	Proposta de Testes	Resolução de Exercício Criado por Outros Alunos
<i>Experimental</i>	2,78 - 3	2,37 - 3	2,21 - 3	2,70 - 3
<i>Controle</i>	2,69 - 3	2,52 - 2	2,38 - 2	2,75 - 3

Também foram registrados os horários das submissões de respostas que representam o

momento que os alunos concluíram a atividade. O intuito dessa informação foi conhecer o comportamento dos alunos em relação ao tempo que se dedicaram à atividade após o período que era pré-determinado para realização da atividade. A Figura 5.20 apresenta o gráfico com os horários das submissões. Os pontos em vermelho representam as submissões do grupo experimental, enquanto o azul as do controle. O teste-t serviu para verificar se havia diferença significativa entre as médias dos dois grupos. Já pela análise visual do Boxplot exibido na Figura 5.21, percebe-se que a mediana do grupo de controle é maior do que a experimental. Essa também foi a conclusão pelo resultado do teste-t, com $p\text{-value} = 0.2811$ e intervalo de confiança de 95%. Sendo assim, há diferença entre os grupos, e o controle passou, em média, mais tempo (em segundos) dedicado à atividade do que o experimental.

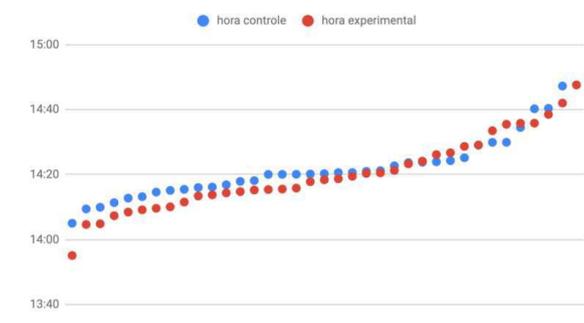


Figura 5.20: Horário das Submissões

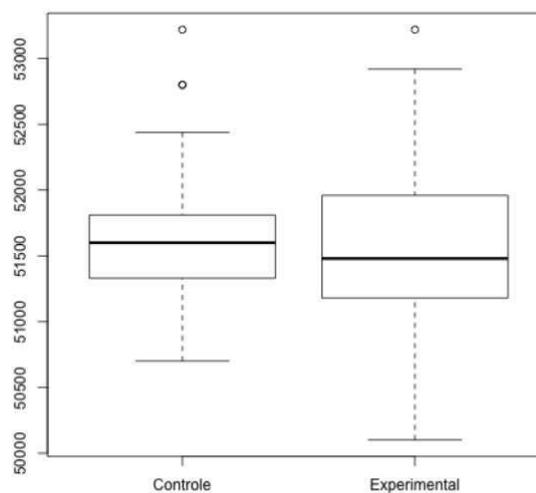


Figura 5.21: Horário das Submissões

Engajamento Cognitivo

Para avaliar o engajamento cognitivo, o tamanho da produção dos alunos foi comparado, tanto o enunciado das novas questões quanto o tamanho do texto das revisões. A Tabela 5.7 apresenta a média e mediana da quantidade de caracteres no enunciado das questões criadas pelos alunos e do texto dos alunos ao revisar o código. Foi utilizado o test-t para verificar se há diferença entre as médias. Com $p\text{-value} = 0.5342$ e intervalo de confiança de 95%, não podemos dizer que há diferença entre as médias de tamanho do enunciado das questões. A mesma conclusão acontece sobre a revisões de código, com $p\text{-value} = 0.357$ e intervalo de confiança de 95%, sendo observada que não há diferença significativa entre os tamanhos dos textos.

Tabela 5.7: Tamanho dos Exercícios

Média - Mediana	Tamanho do Enunciado	Tamanho do Texto da Revisão
<i>Experimental</i>	717,05 - 598	423,97 - 405
<i>Controle</i>	627,02 - 575,5	478,36 - 420,5

Engajamento Emocional

Nessa seção, exibimos as informações sobre as emoções que os alunos relataram estar sentindo ao realizar as atividades desenvolvidas durante o experimento.

Para facilitar a análise, a atenção foi focada nas emoções centrais. Por exemplo, caso o aluno tenha relatado a sensação *com sono*, foi registrado que ele estava *triste*, já para o aluno que relatou estar *interessado*, foi registrado que ele estava *alegre*, e assim por diante.

A Figura 5.22 apresenta as emoções que foram relatadas pelos alunos no grupo de Controle. Em tons de azul, foram destacados aqueles sentimentos que são considerados positivos para a experiência de aprendizado, que inclui alegre, poderoso e tranquilo. Já as sensações negativas, foram destacadas de vermelho e incluem triste, bravo e assustado. Pela análise visual, percebe-se que o espaço em azul do gráfico é maior do que o vermelho, o que pode sugerir que os alunos sentiram mais emoções positivas do que negativas.

A Figura 5.23 apresenta as emoções que foram relatadas pelos alunos no grupo experimental. O mesmo padrão de cores utilizado anteriormente pode ser observado para o gráfico

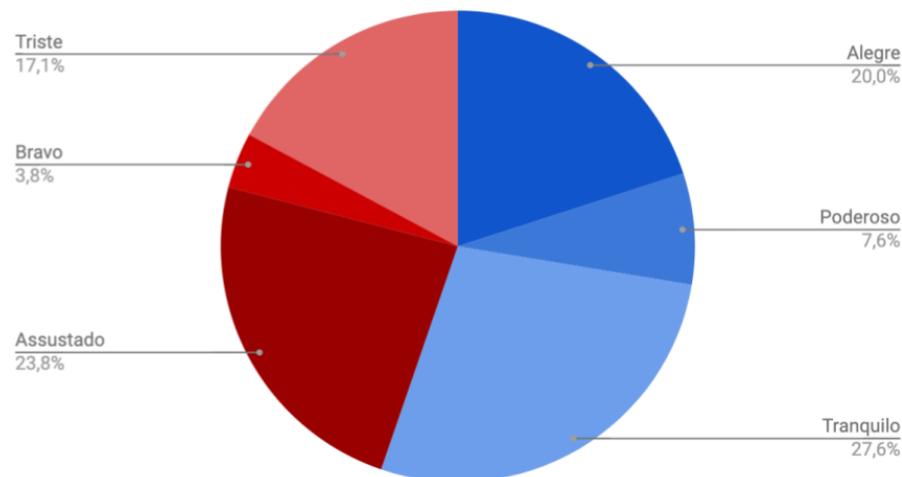


Figura 5.22: As Emoções Relatadas pelo Grupo de Controle

do grupo experimental. Perceba que a porcentagem das emoções em azul são maiores do que as do grupo de controle. As emoções *tranquilo* e *assustado* são as mais relatadas. Há uma redução nas emoções negativas *triste* e *assustado*, quando o gráfico é comparado com o do grupo de controle. Entretanto, há um aumento na emoção negativa *bravo*. Enquanto em relação às emoções positivas, há redução em *alegre* e um aumento nas emoções *poderoso* e *tranquilo*.

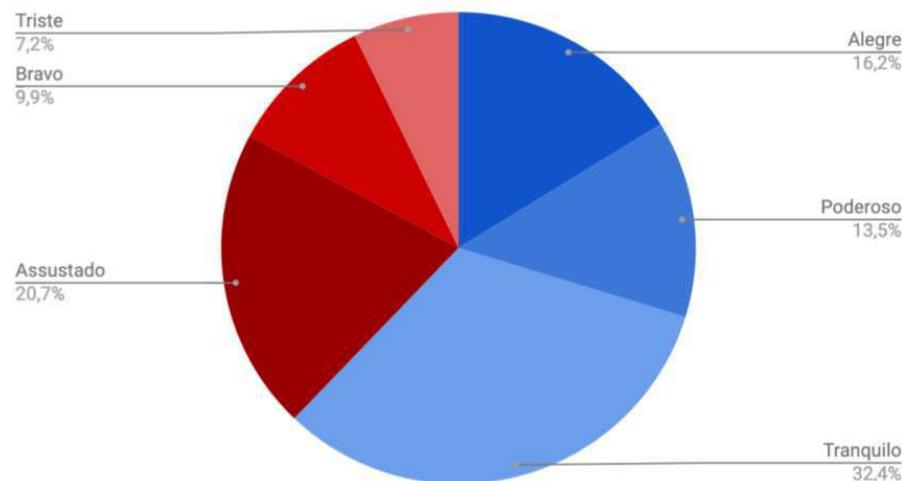


Figura 5.23: As Emoções Relatadas pelo Grupo Experimental

Para facilitar a comparação entre as emoções relatadas, tem-se o gráfico da Figura 5.24. A cor azul foi relacionada aos dados dos alunos do grupo de controle e o vermelho aos do grupo experimental. Percebam que o controle relatou estar mais triste, assustado e alegre

do que o experimental. Enquanto os alunos do experimental relataram estar mais bravos, poderosos e tranquilos.

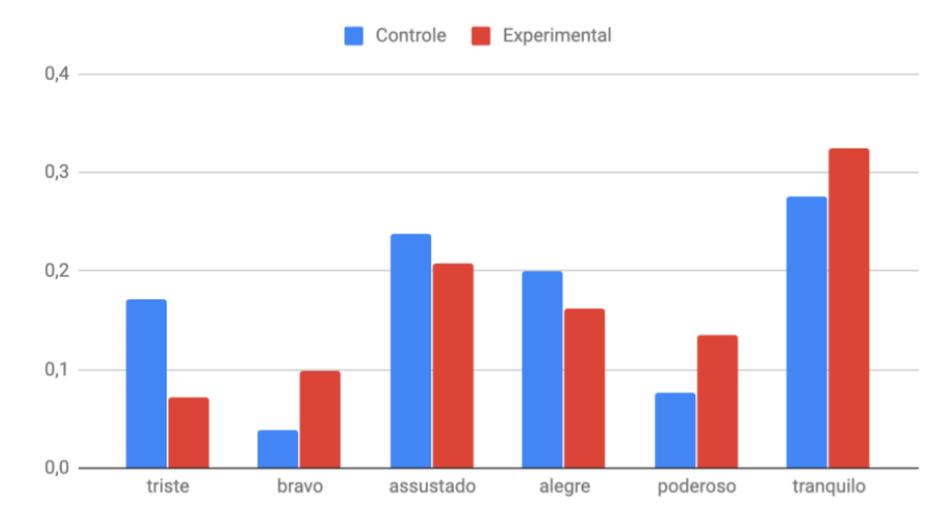


Figura 5.24: Comparação entre as Emoções

Opinião dos Participantes

Com a intenção de coletar mais informações sobre os participantes e compreender alguns dados coletados no experimento, foi solicitado que eles informassem os pontos positivos e negativos encontrados durante a realização daquele tipo de atividade. Esse tipo de *feedback* também foi considerado como aspecto emocional, já que pode direcionar os estudos para possíveis melhorias no modelo de personalização.

A Figura 5.25 apresenta o questionário preenchido por um aluno do grupo de controle que relatou ter sentido apenas sentimentos positivos. Em sua opinião, as atividades de tipos diferentes ajudam a perceber outras dificuldades e melhorar a auto-avaliação. Enquanto a Figura 5.26 apresenta o diário de um aluno que relatou exclusivamente sensações negativas. Esse participante achou a atividade entediante e exaustiva porque “precisa pensar um pouco” e julgou que esse tipo de atividade é “trabalho braçal”.

Já em relação aos participantes do grupo experimental, percebeu-se que aqueles que relataram apenas emoções negativas não tinham interesse em colaborar, pois deixaram evidente a vontade em atingir objetivos individuais. A Figura 5.27 representa a opinião de um participante que relatou apenas sensações negativas e afirma que ficou chateado porque gostaria

Simulado: 20/05/2019

- Use o computador no ambiente de prova para fazer a atividade que está no endereço <http://bit.do/exercicioC>.
- Ao concluir a atividade 1, analise a figura a seguir e assinale as 3 emoções que você experimentou com maior intensidade na aula de hoje.

3. O que você achou da atividade de hoje? Apresente aspecto(s) positivo(s) e negativo(s) desta prática?

A atividade foi muito boa em dois sentidos: porque algumas dificuldades com o curso para criação de problemas e auto-avaliação do que foi aprendido até este ponto da disciplina. Não conseguiu avaliar nenhum ponto negativo forte o suficiente para servir como um contraponto.

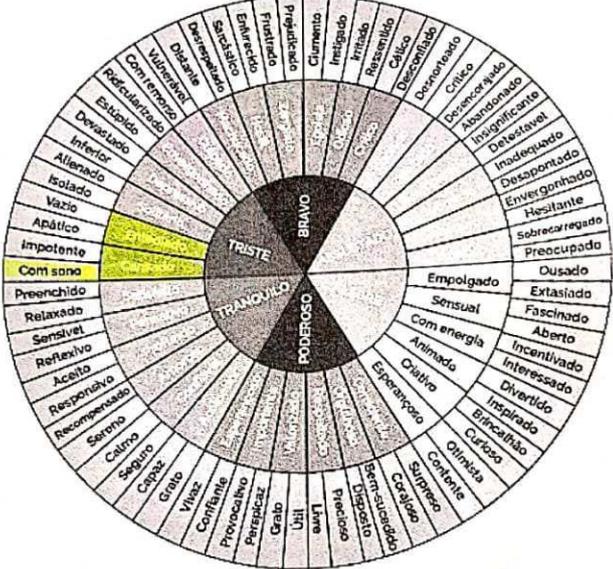
Figura 5.25: Comentário do Grupo de Controle (1)

de ter uma prova, pois não concorda que o estilo de atividade que propomos deva ser usada como estratégia de avaliação.

Esse comportamento com ênfase no objetivo individual também fica evidenciado no diário da Figura 5.28. Embora o aluno tenha relatado emoções que se caracterizam como mais positivas (relaxado, sensível e reflexivo), o seu comentário sobre a atividade foi negativo.

Simulado: 20/05/2019

- Use o computador no ambiente de prova para fazer a atividade que está no endereço <http://bit.do/exercicioC>.
- Ao concluir a atividade 1, analise a figura a seguir e assinale as 3 emoções que você experimentou com maior intensidade na aula de hoje.



3. O que você achou da atividade de hoje? Apresente aspecto(s) positivo(s) e negativo(s) desta prática?

Achei importante forçar o aluno a criar problemas e casos de testes de programas porque faz a gente olhar de uma perspectiva diferente para os problemas do tst, e problemas em geral.

Foi entediante por ser exaustivo e precisar pensar pouco, foi mais um "trabalho braçal".

Figura 5.26: Comentário do Grupo de Controle (2)

Ele afirma que são chatas as atividades de criação de exercícios e questiona se vai receber algum tipo de recompensa com a pergunta “vale nota?”.

A Figura 5.29 apresenta um comentário que se repetiu em vários diários. Nele, o aluno relata dois sentimentos positivos e um negativo. Ao dar a opinião sobre a prática, o aluno evidencia um problema que precisa ser melhorado em relação à aplicação das atividades personalizadas: a qualidade das questões. De fato, os exercícios utilizados precisavam de alguns ajustes e esses erros foram deixados de propósito nas questões desenvolvidas por estudantes para garantir que o conteúdo continuasse fiel ao original.

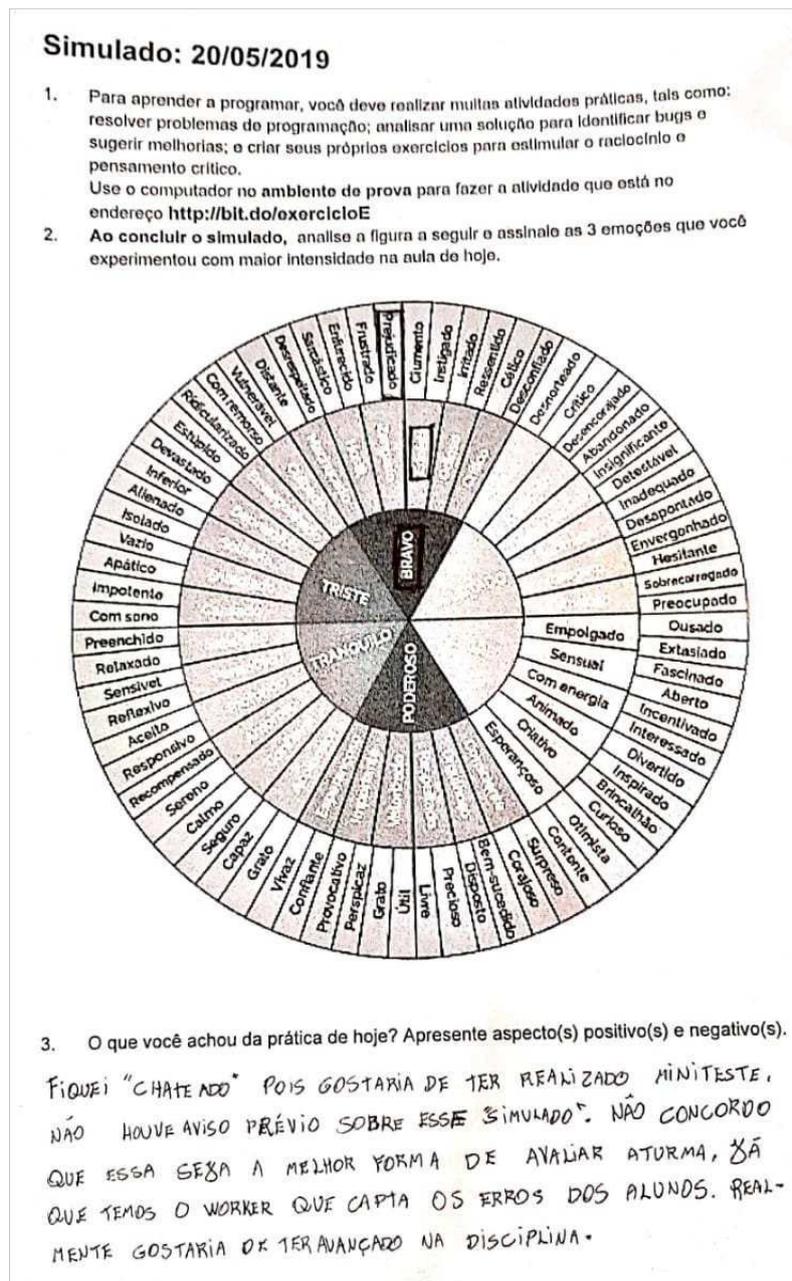
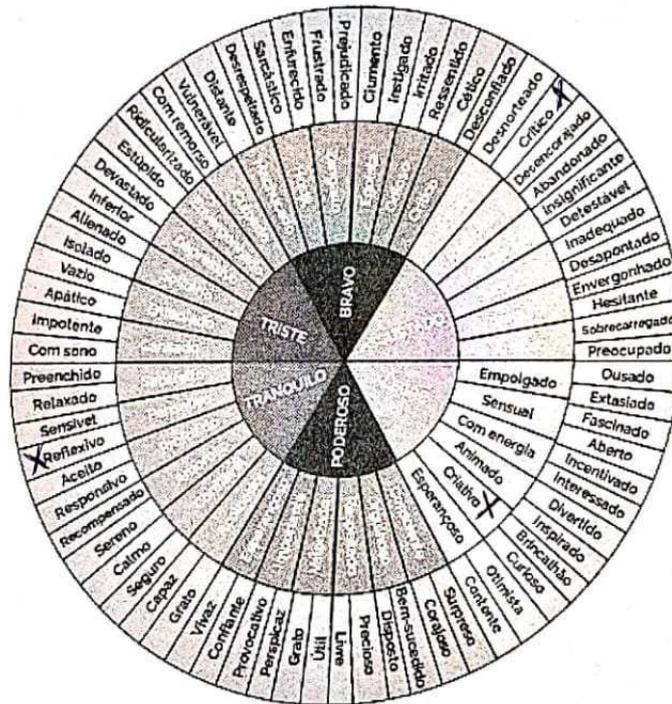


Figura 5.27: Comentário do Grupo Experimental (1)

Por fim, o comentário exibido na Figura 5.30 foi destacado, pois o aluno relatou apenas emoções positivas e afirmou que foi uma das atividades mais legais que fez.

Simulado: 20/05/2019

- Para aprender a programar, você deve realizar muitas atividades práticas, tais como: resolver problemas de programação; analisar uma solução para identificar bugs e sugerir melhorias; e criar seus próprios exercícios para estimular o raciocínio e pensamento crítico.
Use o computador no ambiente de **prova** para fazer a atividade que está no endereço <http://bit.do/exercicioE>
- Ao concluir o simulado, analise a figura a seguir e assinale as 3 emoções que você experimentou com maior intensidade na aula de hoje.



- O que você achou da prática de hoje? Apresente aspecto(s) positivo(s) e negativo(s).

Boa. Um bom método de aprendizado pode ser avaliando o erro das outras e trabalhando a criatividade de resolver os elaborados de problemas.

* Ponto negativo: A apresentação de enunciados incorretos ou incompletos podem levar ao erro de outras estudantes.

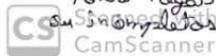


Figura 5.29: Comentário do Grupo Experimental (3)

de um semestre de interação do estudante com um ambiente de aprendizagem e concretizou as métricas de engajamento definidas no Capítulo 4.

Os dados utilizados no estudo resultaram da realização de exercícios de programação no semestre 2017.2. O sistema utilizado para leitura de exercícios foi o TST *online* e a submissão de respostas pelo TST.

Simulado: 20/05/2019

- Para aprender a programar, você deve realizar muitas atividades práticas, tais como: resolver problemas de programação; analisar uma solução para identificar bugs e sugerir melhorias; e criar seus próprios exercícios para estimular o raciocínio e pensamento crítico.
Use o computador no ambiente de prova para fazer a atividade que está no endereço <http://bit.do/exercicioE>
- Ao concluir o simulado, analise a figura a seguir e assinale as 3 emoções que você experimentou com maior intensidade na aula de hoje.

3. O que você achou da prática de hoje? Apresente aspecto(s) positivo(s) e negativo(s).

Uma das mais legais que fiz, todos os professores dizem que a melhor maneira de aprender é ensinando e aqui nós ensinamos.

Figura 5.30: Comentário do Grupo Experimental (4)

Destaca-se que esse estudo de caso resultou no artigo *Online Student Engagement: A Case Study in Teaching of Programming* apresentado no Simpósio Brasileiro de Informática na Educação (SBIE) - 2019 (Maia, Figueiredo e Serey 2019) e que está no Apêndice B deste documento.

5.4.1 Materiais e Método

Nesse estudo, é possível compreender como os estudantes aprendizes de programação estudam. Para isso, partimos da premissa de que o estudante está estudando enquanto realiza atividades de programação, utilizando o TST e TST online para leitura e submissão de respostas.

Diante disso, o estudo foi mediado pelas seguintes questões de pesquisa (QP):

- **QP1:** Como estudantes aprendizes de programação se engajam em um ambiente online de aprendizado?
- **QP2:** Qual a métrica de engajamento de estudantes de programação tem maior correlação com o desempenho?

A fim de responder a QP1, utilizou-se dados para compreender aspectos comportamentais e cognitivos do engajamento dos estudantes em atividades de aprendizagem. Também foram usadas as informações sobre o dia e horário das submissões de respostas para compreender a dedicação dos estudantes ao estudo e como realizam as tarefas de casa. Quanto a resolução da QP2, foi realizada análise da correlação dos valores das métricas com o resultado final da disciplina.

Para o contexto desse estudo, considerou-se que o estudante está estudando quando está interagindo com um dos dois sistemas, TST e TST *online*. Portanto, esses sistemas foram instrumentados para registrar em um arquivo os eventos de leitura e envio de respostas de exercício. Assim, para cada evento desse tipo, foram registrados os seguintes dados:

- Um identificador do **aluno** que realizou a tarefa;
- Um identificador do **exercício**;
- Em que situação ou **modo de execução** a atividade foi realizada, podem assumir os valores exercício comum, simulado, maratona ou avaliação;
- **Unidade** de conteúdo no qual se encaixa o exercício;
- A **data e hora** de ocorrência do evento;

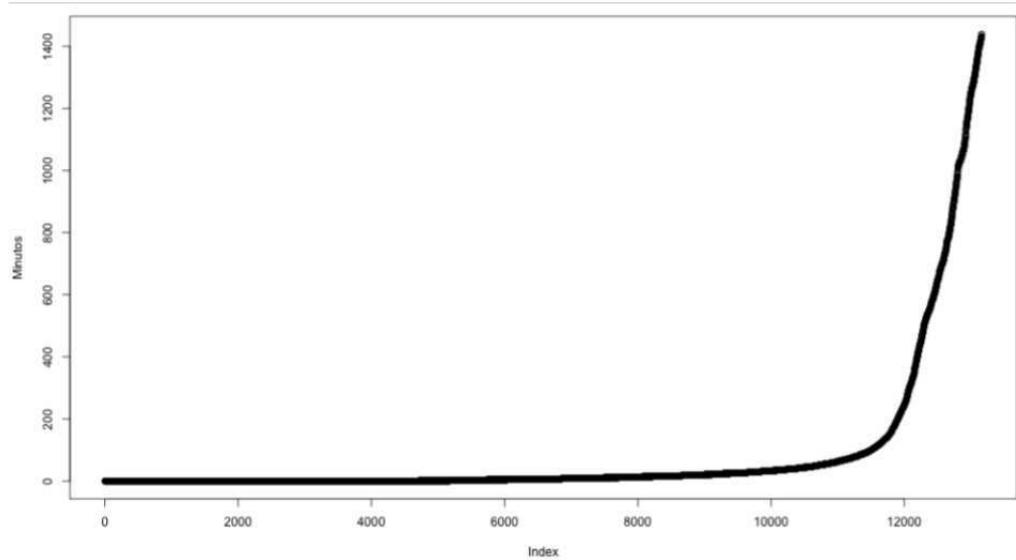
Utilizamos esses dados para concretizar as seguintes métricas:

- **Sessão de Estudos** - Por quanto tempo o estudante permanece interagindo com o sistema, lendo ou respondendo exercícios;
- **Participação** - o interesse dos estudantes por atividades complementares. Consideramos que esse tipo de atividade é todo tipo de tarefa proposta pelo professor que a resolução não é obrigatória. Isto inclui os simulados e maratonas de programação. Simulado é um tipo de atividade utilizada para treinamento, e maratona é uma dinâmica que promove competição entre grupos;
- **Tarefas realizadas em casa (*homework*)** - tarefa realizada fora do ambiente de aula, sem a orientação física do professor;
- **Eventos** - a quantidade de interação do estudante com o sistema;
- **Ritmo de Aprendizagem** - se a tarefa foi concluída em modo de avaliação, consideramos que o estudante a realizou durante o mini-teste. O cálculo do ritmo de aprendizagem foi realizado pelo número de tarefas concluídas por mini-teste. Para simplificar, consideramos a média final dos ritmos em todos os mini-testes como medida de ritmo;

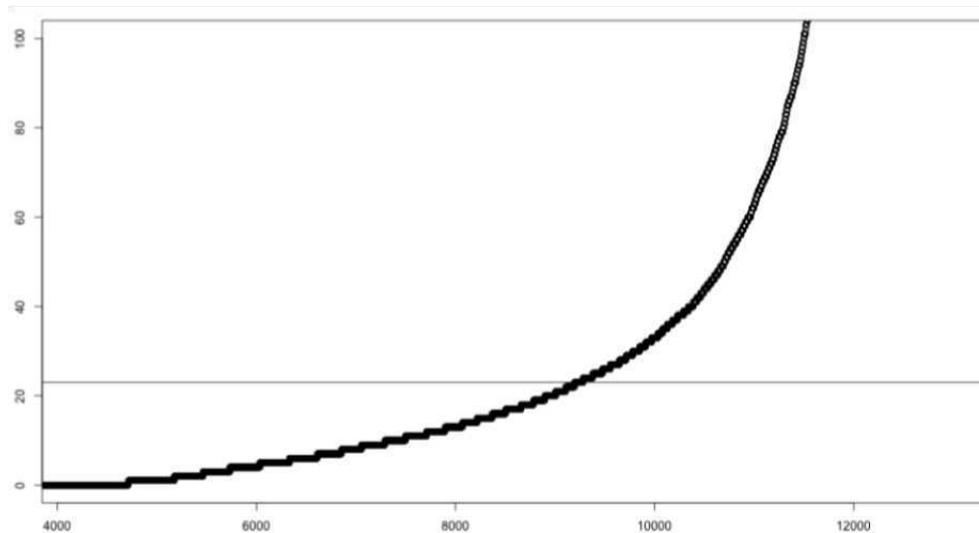
Para conhecer a duração da sessão de estudos, isto é, o tempo que o estudante está se dedicando à realização de exercícios, utilizamos os conceitos de *thinking-time* e *break-time* (Araujo et al. 2013). O conceito representa um período em que o estudante está focado em estudar, desenvolvendo exercícios. O *break-time* é o período no qual o estudante está realizando atividades fora do ambiente de estudo. Em geral, um estudante passa mais tempo em *break-time* do que em *thinking-time*, representando que o break é um período mais longo que *thinking*. O intervalo entre os eventos foi calculado para compreender qual a diferença entre os dois períodos de tempo.

A Figura 5.31-a apresenta o intervalo entre os eventos em minutos e Figura 5.31-b destaca o ponto correspondente a 23 minutos, delimitando o ponto após o qual o intervalo aumenta em proporções maiores. Assim, o tempo de 23 minutos foi considerado como a duração do *thinking-time*, intervalos superiores a esse valor são considerados *break-time* no nosso estudo.

Com o propósito de mensurar as tarefas de casa e a participação em atividades complementares, foi reputado os dias da semana quando aconteciam as aulas e avaliações. No



(a) Intervalo entre Submissões



(b) Destaque para o Intervalo de 23 minutos

Figura 5.31: Intervalo de Tempo Entre os Eventos

semestre 2017.2, quando o estudo foi realizado, as aulas aconteciam segunda-feira e quarta-feira, e as avaliações na quinta-feira no turno matutino. Para conhecer a participação dos estudantes, foram analisado os eventos associados a questões de modo avaliação. O modo de execução também foi utilizado para mensurar a participação em atividades complementares. No semestre em que o estudo de caso foi realizado, foram realizadas apenas atividades extra do tipo simulado.

5.4.2 Resultados

No semestre 2017.2, as turmas continham cerca 100 estudantes matriculados. Entretanto, cerca de 16 deles abandonaram o curso ou atingiram o número máximo de faltas na disciplina e seus dados foram desconsiderados. Assim, 84 alunos atenderam a esses requisitos e solucionaram um total de 450 questões durante 151 dias, resultando em 14320 eventos.

A apresentação do resultado foi guiada pelas respostas às perguntas de pesquisa:

- **QP1: Como estudantes aprendizes de programação se engajam em um ambiente on-line de aprendizado?**

A duração total das sessões de estudo variou entre 4h51m e 44h29m, com uma média de 18h19m. Uma diferença de mais de 900% entre os dois extremos, aqueles alunos que estudaram muito e os que estudaram muito pouco. A Figura 5.32 apresenta o gráfico boxplot que representa a distribuição dos dados referentes à duração total das sessões de estudo em minutos. Durante essas sessões, registrou-se um total de eventos que variaram de 143 a 718, com uma média de 341,4.

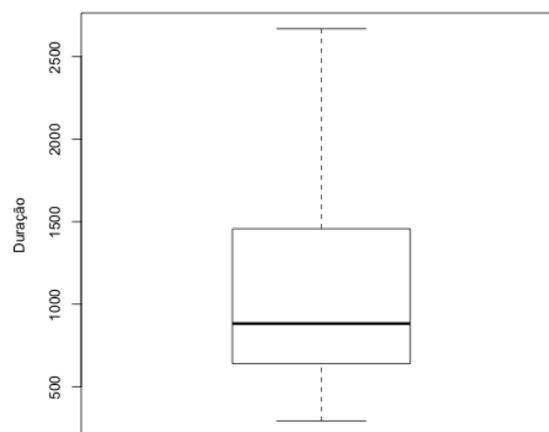


Figura 5.32: Duração das Sessões de Estudo

O número total de tarefas concluídas variou entre 49 e 334, com uma média de 147 questões. A Figura 5.33 exhibe o histograma com os dados relacionados ao número de exercícios concluídos. Um único aluno concluiu o mínimo de questões, enquanto a

realização de exercícios da maioria dos alunos de concentra entre 50 e 100 atividades concluídas.

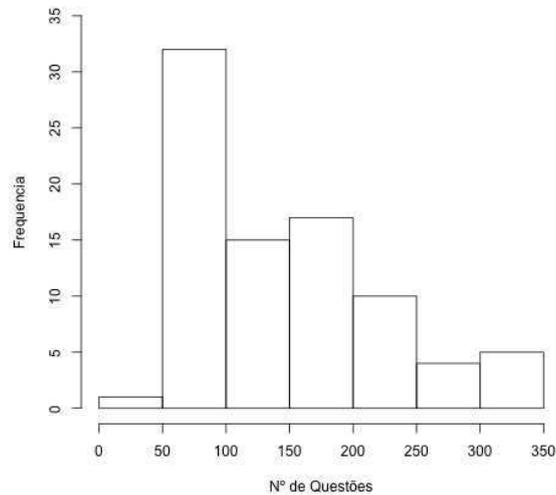


Figura 5.33: Tarefas Concluídas

Com relação ao número de questões concluídas como dever de casa, foi observada uma variação entre 43 a 321, com média de 140 tarefas. Comparando os dados do total de questões concluídas e tarefas feitas como dever de casa, foi perceptível que cerca de 97,5% das questões são realizadas como tarefa de casa, ou seja, fora do horário habitual da aula.

A Figura 5.34 apresenta o gráfico histograma que representa o número de questões concluídas durante os simulados. Esse valor variou entre 0 e 41, com média de 21 questões.

Para entender a rotina de estudos dos participantes, foi analisado como as sessões estão divididas ao longo da semana. A Figura 5.35 apresenta a duração em minutos das sessões de estudos em cada dia da semana, agrupados pelo turno manhã (Ma), tarde (T), noite (N) e madrugada (Ma). Destaca-se que nesse estudo, foi considerado como turno da manhã entre 6 e 12 horas, tarde inicia depois das 12 até às 18 horas, enquanto o turno da noite inicia no fim do período da tarde até 22 horas. A madrugada foi considerada como o período depois das 22 e antes das 6 horas.

Percebe-se que os maiores picos de estudo acontecem no horário das avaliações

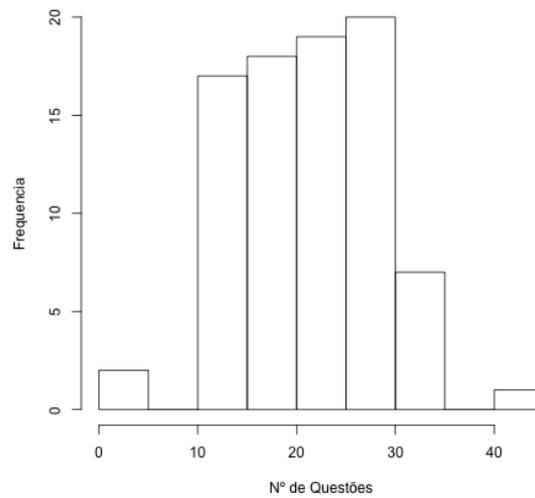


Figura 5.34: Exercícios Resolvidos em Simulados

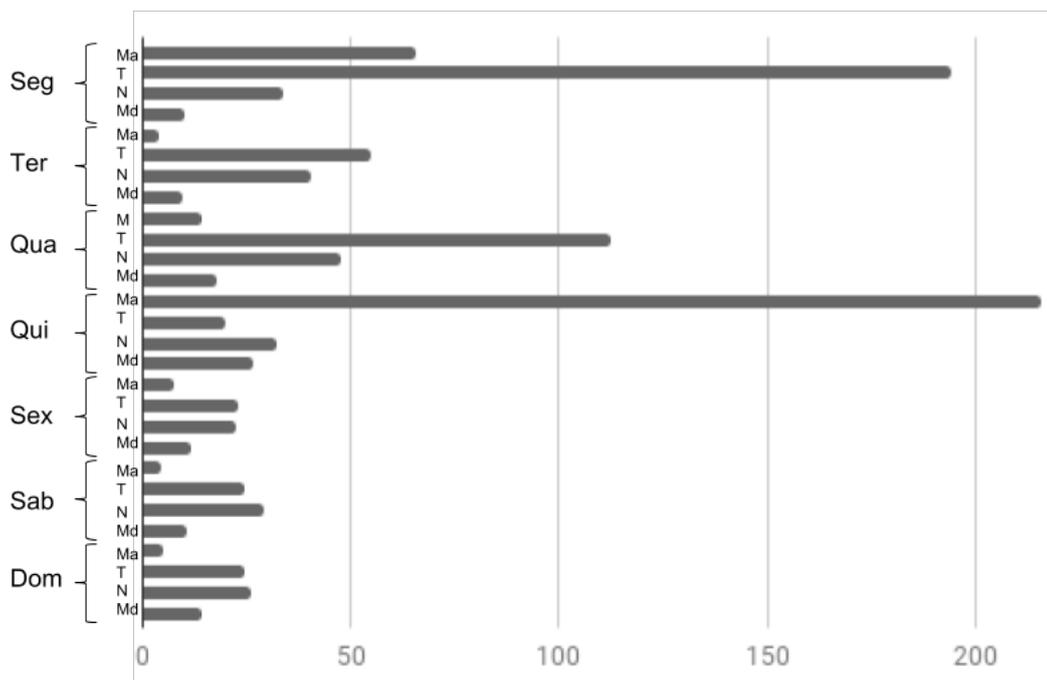


Figura 5.35: Hábito de Estudo

(quinta-feira pela manhã), no turno seguinte à aula (segunda-feira a tarde) e na véspera da avaliação (quarta-feira à tarde). Ou seja, os estudantes dedicam mais tempo de estudo nos dias que antecedem a avaliação (segunda, terça e quarta-feira). Depois da avaliação, especialmente da quinta-feira a tarde até o domingo, ocorre uma diminuição no número de eventos, representando que os estudantes também diminuem o tempo de

estudo.

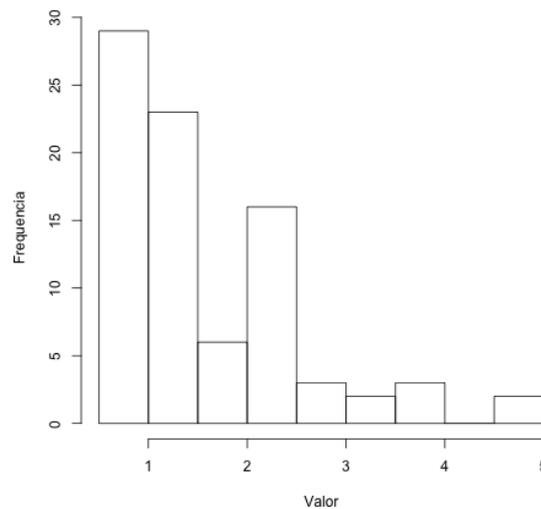


Figura 5.36: Ritmo de Aprendizagem

A Figura 5.36 apresenta o gráfico histograma com a frequência dos valores do ritmo de aprendizagem ao final do curso. Cerca de 54% dos estudantes apresentaram o valor mínimo 0,5 como ritmo de aprendizagem. Inferi-se que, a maioria dos alunos resolveu cerca de uma questão por mini-teste. Enquanto apenas 1 aluno conseguiu avançar 5 unidades em um único mini-teste. A média da turma para o ritmo de aprendizagem foi de 1,6 unidades por teste.

• **QP2: Qual a métrica de engajamento de estudantes de programação tem maior correlação com o desempenho?**

O método utilizado para testar a normalidade dos dados foi o *Shapiro-Wilk*. Foi notado que os dados não seguem uma distribuição normal porque todos os valores de *p-value* foram menores que 0,05.

Isso posto, o método *Kendall* foi escolhido para analisar a correlação dos dados. Esse método calcula o valor do τ que determina o nível de correlação entre os dados baseado nos seguintes critérios:

- $\pm 0.9 \leq \tau \leq \pm 1.0$: correlação muito forte
- $\pm 0.7 \leq \tau \leq \pm 0.9$: correlação forte

- $\pm 0.5 \leq \tau \leq \pm 0.7$: correlação moderada
- $\tau \leq \pm 0.4$: correlação fraca

No semestre em que ocorreu o estudo de caso, o desempenho da disciplina era calculado levando em consideração a nota final dos mini-testes e a avaliação qualitativa realizada dos códigos dos estudantes. Sendo que cada questão respondida corretamente equivale a 0,5 pontos da nota dos mini-testes. O somatório dessas notas corresponde a 70% do desempenho do estudante. O complemento do desempenho que equivale a 30% da nota final é calculado pela avaliação da qualidade do código gerado pelos estudantes durante os mini-testes.

A Tabela 5.8 apresenta o valor do τ resultado da análise de correlação entre as métricas: questões concluídas, participação em simulados, realização de tarefas em casa, duração das sessões, número de eventos e desempenho. Dois valores acima de 0.9 se sobressaíram indicando correlação muito forte entre as seguintes métricas: questões concluídas versus tarefa de casa e questões concluídas versus número de eventos. Valores destacados com * representam outras correlações fortes, como o número de questões concluídas versus duração da sessão de estudos, e tarefa de casa versus número de eventos.

Em cinza, os valores destacados estão relacionados com o desempenho. Observe que a única métrica com $\tau \geq 7.0$, representando uma correlação forte, trata-se do ritmo de aprendizagem. Isso pode representar que os estudantes que progredem rapidamente no curso podem ter a tendência de conseguir bons desempenhos.

Uma outra estratégia que utilizada para verificar a relação entre indicadores de engajamento e o desempenho foi agrupar os dados dos alunos de acordo com as notas obtidas pelos estudantes no final do curso. O grupo A foi formado por estudantes que obtiveram nota final abaixo de 7,0 e grupo B que obtiveram nota maior ou igual. Os dados dos estudantes que desistiram do curso não foram incluídos na análise.

A Figura 5.37(a) apresenta a diferença do número de questões resolvidas entre os dois grupos. Estudantes do grupo A fizeram entre 49 e 258, com média de 118. Ao passo que os do grupo B fizeram entre 64 e 334 questões, com média de 170. O teste *t* foi utilizado para comparar as médias dos dois grupos e confirmar que a média de

Tabela 5.8: Correlação entre Engajamento e Desempenho

	Exerc.	Part. em Sim.	Tarefa de Casa	Ritmo	Dur. da Sessão	No. de Eventos
Part. em Simulados	0,3468					
Tarefa de Casa	0,9372*	0,3265				
Ritmo	0,2569	0,2572	0,2718			
Dur. das Sessão de Estudo	0,8085*	0,3169	0,791	0,3159		
No. de Eventos	0,9198*	0,3578	0,8862*	0,2062	0,7614	
Desempenho	0,273	0,2519	0,2867	0,8238	0,3328	0,222

questões dos estudantes com notas melhores é maior do que a dos demais ($p\text{-value} = 0.0004174$).

Em relação ao número de questões realizadas fora do horário da aula, ou seja, como tarefa de casa, pelo grupo A, é possível observar que esse número variou entre 43 e 247, com média de 108. Enquanto, no grupo B, esse número variou entre 64 e 320, com média de 158. Com o mesmo *teste t*, foi confirmada a análise visual do gráfico da Figura 5.37(b), bem como foi concluído que a média do número de exercícios de casa feitos do grupo A é menor do que grupo B ($p\text{-value} = 0.0003643$).

A Figura 5.39 apresenta a duração das sessões e a diferença entre os dois grupos. O grupo A apresentou sessões com duração entre 291 e 1721, com média de 833,4. Entretanto o grupo B, apresentou sessões com duração entre 456 e 2669, e 1299 de média. Quanto ao número de eventos realizados durante as sessões, vemos na Figura 5.38 que a média dos alunos do grupo A é menor do que do grupo B.

A Figura 5.40 apresenta a diferença do número de questões realizadas em simulados entre os dois grupos. O grupo A fez entre 4 e 29 questões, com média de 19,08. Enquanto o grupo B fez entre 0 e 41, com 23,5 de média. Por meio do *teste t* e com $p\text{-value} = 0.00171$, foi possível concluir que o grupo B fez mais questões em simulados.

A maior diferença entre os dois grupos de alunos aparece na Figura 5.41. O menor ritmo de aprendizagem do grupo A é 0,5 e o maior é 1,42. Já no grupo B, o valor varia

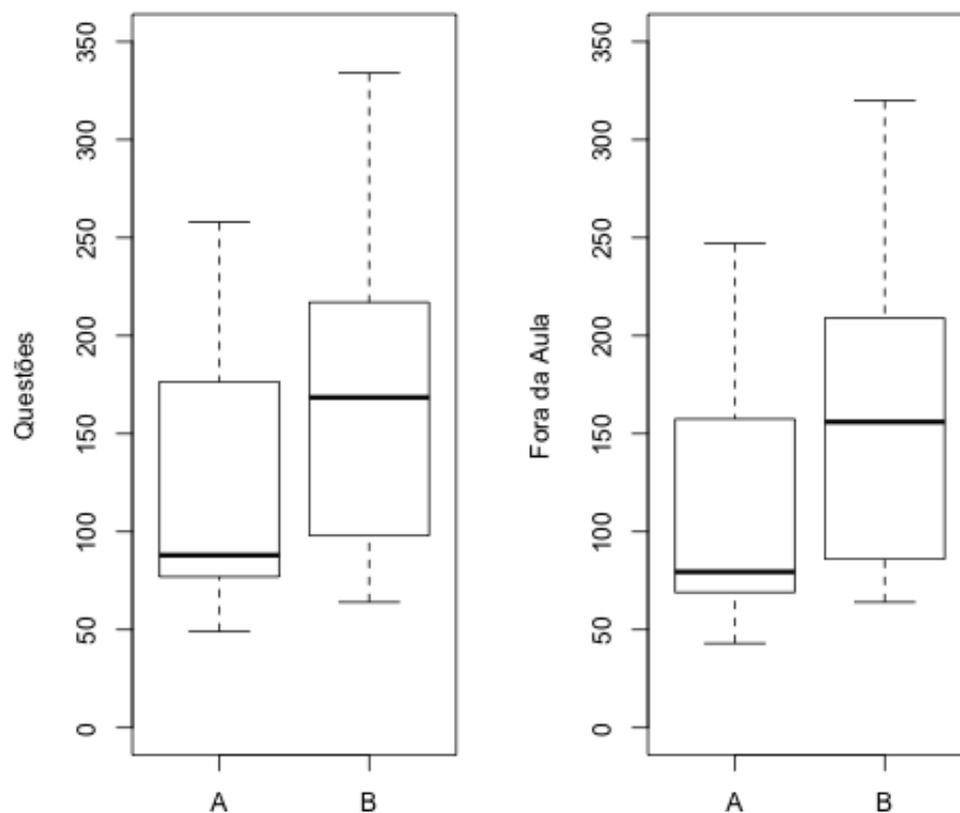


Figura 5.37: (a) Total de Questões Realizadas; (b) Tarefas de Casa

entre 1,08 e 5. Isso representa que os estudantes que avançam mais rápido também apresentam melhores notas no final da disciplina.

5.5 Conclusão

Esse capítulo apresentou os estudos em que foi avaliado o modelo de personalização e a aplicação das métricas de engajamento em atividades de estudantes aprendizes de programação. Como também, foi descrito o ambiente de experimentação no qual todos os estudos foram realizados. Foram utilizados três estudos para avaliar o engajamento de estudantes de programação, sendo dois experimentais e um estudo de caso. Os experimentais foram realizados em ambiente controlado de laboratório durante um curto espaço de tempo para

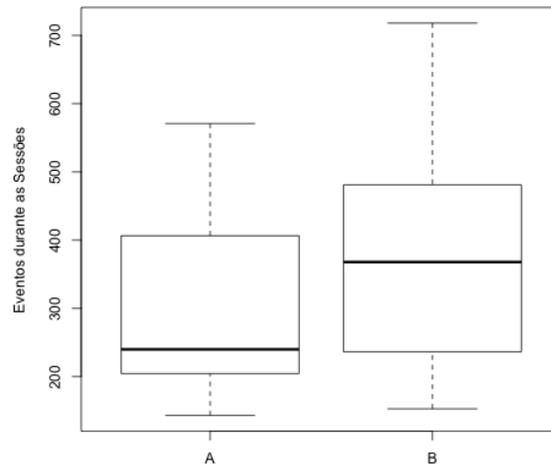


Figura 5.38: Número de Eventos

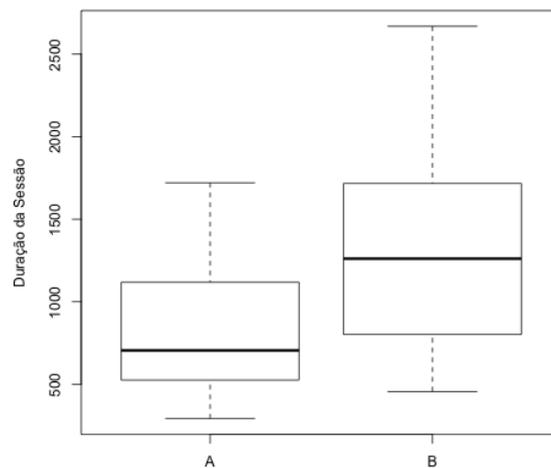


Figura 5.39: Duração das Sessões em Minutos

obtenção dos dados quantitativos sobre o engajamento do colaborador que realiza atividades personalizadas de programação.

O experimento 1 avaliou o engajamento do colaborador na realização de atividades personalizadas do tipo criação de exercícios. O experimento 2 avaliou o engajamento na realização dos outros tipos de atividades personalizadas, bem como apresentou uma avaliação quantitativa para compreender a opinião dos estudantes sobre a realização desse novo modelo de atividades.

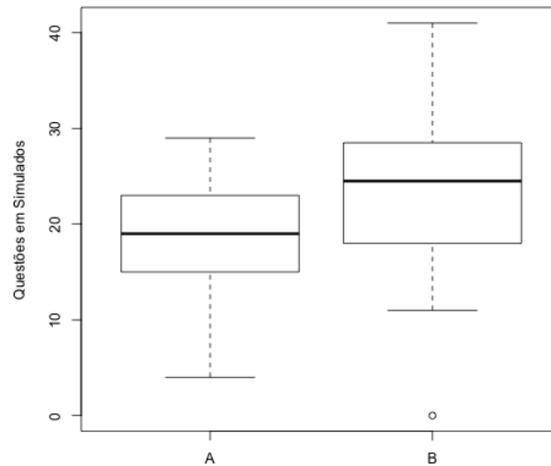


Figura 5.40: Questões em Simulados

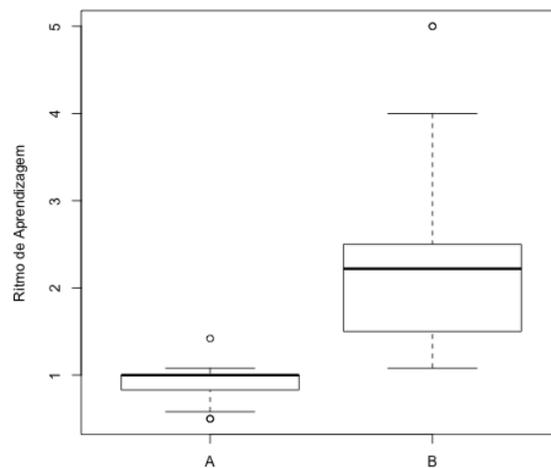


Figura 5.41: Ritmo de Aprendizagem

No experimento 1, foram encontrados indícios de efeitos positivos no engajamento cognitivo do colaborador, e também efeitos favoráveis em sensações positivas e efeitos dimi-
nutivos de sensações negativas do aspecto emocional. Por outro lado, não foram obtidas evidências sobre o efeito no aspecto de comportamento do engajamento, portanto esse estudo experimental apresenta algumas ameaças à validade. A primeira delas é a validade de conclusão devido ao pequeno número de participantes e a forma de seleção da amostra. Tais aspectos foram causados pela seleção da amostra de participantes dificultada pelo

turno disponível para realização da atividade em laboratório. A maioria dos estudantes não comparecem no turno no qual não aconteciam as aulas presenciais. Dessa forma, apenas cerca de 27% participaram do experimento, número pequeno para generalizar o resultado representando uma ameaça à validade de constructo. Por não ter utilizado um instrumento validado para mensuração de sensações, o questionário ignora sensações negativas, como raiva e tédio, e outras positivas, como criatividade e surpresa. Ao mesmo tempo que, reconhecer sentimentos é uma tarefa que exige maturidade e auto-conhecimento. Acredita-se que o instrumento utilizado para coleta desse tipo de informação não foi adequado e pode ter gerado ameaça à validade dos dados.

No experimento 2, não foram encontradas diferenças significativas nas dimensões cognitiva e comportamental do engajamento. Embora, há um aumento em estados emocionais considerados positivos, e também redução de estados negativos. Nos dois experimentos, é preciso ressaltar os diferentes vieses que podem comprometer os resultados. As ameaças à validade desse estudo também estão relacionadas à amostra de população, principalmente em relação ao seu tamanho reduzido. No que diz respeito à seleção dos participantes, todos os estudantes que assinaram o termo de consentimento do projeto de pesquisa e que não faltaram a aula no dia do experimento participaram do estudo.

O estudo de caso foi realizado com a intenção de concretizar as métricas de engajamento de aprendizes de programação com dados da interação do estudante com um ambiente de aprendizagem. Assim como os outros experimentos, não há dados suficientes para generalizar os resultados.

Capítulo 6

Discussão

Nessa pesquisa de doutorado, foi analisado como melhorar o engajamento de aprendizes de programação utilizando atividades personalizadas. Para isso, além de propor e descrever esse modelo de personalização, também foram sugeridas métricas para sinalizar aspectos relevantes do engajamento dos estudantes que desenvolvem atividades personalizadas de programação. Para conhecer o comportamento dos estudantes iniciantes na codificação e para investigar o impacto no engajamento do uso do modelo de personalização foram realizados alguns estudos. Os resultados sugerem que não há impacto significativo nos indicadores comportamentais do engajamento. Ao mesmo tempo que há impacto positivo no engajamento cognitivo em algumas situações e melhoria do engajamento emocional.

Esse capítulo apresenta as principais dificuldades que enfrentamos durante a realização dessa pesquisa de doutorado, discute os principais temas abordados durante o seu desenvolvimento e apresenta implicações práticas em relação à aplicação do modelo de personalização em cursos de programação.

6.1 Desafios da Pesquisa

Durante o desenvolvimento dessa pesquisa, alguns obstáculos foram enfrentados. O primeiro desafio foi cumprir as exigências para autorização de projetos com participantes com menos de 18 anos pelo Comitê de Ética em Pesquisa. Dado que, para participar da pesquisa, além dos estudantes, os pais precisaram autorizar a participação. O problema é que, a maioria deles mora longe dos pais e/ou responsáveis, atrasando o início da pesquisa e dificultando a

seleção de uma amostra com tamanho estatisticamente significativo.

O segundo desafio foi a execução dos estudos no ambiente de experimentação. O calendário letivo do curso, muitas vezes, foi prejudicado por motivos que fogem da alçada da pesquisa, envolvendo até greve dos caminhoneiros. Por esse motivo, o cronograma de atividades e a realização dos experimentos foram alterados inúmeras vezes porque precisavam se adaptar a essas mudanças no calendário. Em paralelo a isso, ainda enfrentamos, a cada semestre, as alterações no próprio ambiente, incluindo mudanças no design da disciplina, do suporte ferramental e até da equipe de instrutores. O estilo de ensino influencia na escolha do melhor momento para realizar os estudos experimentais. À medida que a equipe mudava, as estratégias de ensino também eram afetadas. Dessa forma, essas alterações influenciaram o cronograma de execução e o design dos experimentos. Os estudos só puderam ser executados em determinado momento do semestre, especialmente quando os alunos conseguiam atingir uma unidade de conteúdo mais avançada (a partir da unidade 4). Devido a tais mudanças entre semestres, uma possível comparação de dados de alunos de semestres diferentes e longitudinal foi inviabilizada.

6.2 Implicações

A atividade personalizada de programação empodera o estudante porque o atribui a responsabilidade de elaborar material de estudo para outras pessoas. Nesse contexto, o professor deve atuar como orientador e monitorar como os estudantes desenvolvem as atividades.

Destaca-se a importância de dedicar tempo para passar orientações sobre como criar exercícios em formato de treinamento. Assim, os aprendizes podem aprender quais as características e os tipos do material que será gerado, conteúdo explorado, vocabulário, nível de dificuldade da questão e, principalmente, como usar o ferramental tecnológico. O treinamento também é importante para garantia da qualidade do recurso gerado pelos estudantes.

Não só para criar exercícios, mas também é necessário planejar um treinamento para explicar como ocorre uma revisão eficiente, tanto de texto quanto de código. Além de explicar como criar bons testes.

O suporte tecnológico facilita a aplicação do modelo de personalização. Caso o instrutor deseje implantar o modelo em sua disciplina, recomenda-se fortemente que utilize uma

ferramenta adequada para o contexto do ensino de programação. Principalmente porque atividades de programação possuem características específicas, envolvendo testes e submissão de código fonte como resposta. Tais ferramentas sugeridas devem atender aos seguintes requisitos:

1. **Cadastro de usuários** - devem existir dois tipos de usuários: instrutores e estudantes. Ambos podem cadastrar novas atividades, embora os instrutores tem mais privilégios do que os estudantes porque podem acompanhar a interação entre os estudantes;
2. **Comunicação** entre usuários - entre usuários - pode ser em formato de chat ou em fóruns;
3. **Banco de dados de atividades** - cadastro, consulta, edição e remoção de exercícios;
4. **Inserção de Comentários em Atividades** - esse tipo de função pode ser utilizada para fazer a revisão do código e do texto de uma questão, além de poder ser utilizada na sugestão de testes.

Ao ter a responsabilidade de criar exercícios de programação compartilhada entre os alunos e professores, o material de estudo vai ser sempre uma novidade. Visto que, à medida que os estudantes geram novos exercícios, o banco de dados de questões também vai aumentar. Proporcionando aos estudantes mais opções para praticar programação. Consequentemente, os professores podem reduzir o tempo e esforço dedicados ao planejamento de atividades para a disciplina.

Durante os estudos realizados ao longo dessa pesquisa de doutorado, foram gerados cerca de 260 novos exercícios de programação por 116 estudantes. Todos apresentam enunciado com descrição de entrada e saída válidas, testes e um código solução. Apenas uma parcela de cerca de 20% dos enunciados foi revisada. Portanto, com o BD com essa quantidade de questões, é possível ainda decompor em outras novas 600 atividades, incluindo para revisão do texto, sugestão de novos testes e revisão de código.

6.3 Efeito no Engajamento de Estudantes

Nessa seção, os resultados dos nossos estudos em relação às três dimensões do engajamento de estudantes são discutidos.

- **Comportamental** - Foram comparadas a quantidade de atividades concluídas, duração de sessão de estudos e período entre submissões, de dois grupos. Participantes que compunham o grupo de controle foram expostos a uma lista de exercícios sem indicação de que estavam colaborando realmente com alguém. O outro grupo, realizaram exercícios que mascaravam a colaboração. A comparação entre os indicadores dos grupos não revelou diferença significativa entre o comportamento dos participantes. Esse resultado sugere que os estudantes que colaboram por meio de atividades personalizadas não são prejudicados. Nesses experimentos, as formas de concretizar indicadores e atividades não permitiram demonstrar se há influência no engajamento comportamental.
- **Cognitivo** - A medição desse aspecto foi a tarefa mais desafiadora dos estudos. Principalmente porque não encontramos formas efetivas de mensurar cognição em seções curtas de estudo, cenário que caracteriza os estudos experimentais. Por isso, foram consideradas o nível de dificuldade das questões, o número de novos testes, tamanho do enunciado das questões, entre outros como indicativo de que o estudante está cognitivamente engajado com a atividade. Porém, não é seguro afirmar que o nível de dificuldade, classificado pelo autor da atividade, reflita fielmente a complexidade da questão. Para isso, seria necessário uma análise mais profunda, usando outros aspectos e a opinião de outras pessoas. Por outro lado, por se tratar da avaliação do participante, isso pode refletir no esforço que o mesmo julgou ser necessário para construir aquela questão. Porém, em relação aos demais indicadores cognitivos, estes experimentos e as formas de concretizar atividades não permitiram demonstrar se há influência no engajamento cognitivo.
- **Emocional** - O reconhecimento de estados emocionais requer um nível alto de autoconhecimento e maturidade para conhecer se o evento causou determinada emoção ou se é resultado de outros acontecimentos. A avaliação dos aspectos emocionais do

engajamento foi realizada por meio de questionários de auto-relato com alguns exemplos de emoções. No primeiro estudo, a intensidade de algumas emoções foi avaliada. Enquanto no segundo, foram avaliadas quais eram elas e se estavam ligadas a emoções que consideramos positivas ou negativas. Os estudantes relataram se sentir mais tranquilos e poderosos, e menos tristes e assustados, após a participação nos experimentos. Portanto, aplicar as atividades personalizadas, mesmo que seja em uma única seção de estudo, tem o potencial de melhorar estados emocionais positivos e reduzir a sensação de negativos.

6.4 *Feedback dos Estudantes*

Com a intenção de conhecer a opinião dos estudantes, foi solicitado que os participantes de um dos estudos nos enviassem três fatores positivos e três negativos em relação à atividade a que eles foram submetidos. A Tabela 6.1 apresenta a transcrição fiel de comentários fornecidos por quatro participantes, sendo dois do grupo de controle e dois do experimental. Os questionários completos estão no Apêndice G.

Nesta seção, os fatores positivos e negativos serão discutidos separadamente a seguir:

- **Pontos positivos** - Os estudantes reportaram ter tido uma experiência positiva na realização de atividades que incentiva a produção de conteúdo e estimulam a criatividade. E apontaram que o tipo de atividades a que eles foram submetidos estimulam a reflexão sobre o que conhecem sobre o assunto e a análise de código. Um deles mencionou a reflexão sobre a estrutura, comandos de programação e como solucionar problemas. Destaca-se ainda, a sugestão de permitir compartilhamento de exercícios entre estudantes que foi feita por um participante do grupo de controle, um dos objetivos da aplicação de atividades personalizadas.
- **Pontos negativos** - Os participantes do experimento acreditavam que estavam realizando um simulado. Nesse tipo de evento, os estudantes devem responder a questões durante 1 hora. Geralmente, são questões curtas de escrita de código para resolução de problemas de programação. Ao se deparar com um modelo de atividade diferente, alguns participantes relataram se sentir decepcionados. Essa sensação pode ter sido

Tabela 6.1: Comentários de Participantes

C	"A atividade possibilitou descobrir realmente se conhecia o determinado assunto, verificar se tinha a capacidade de pensar e refletir como os professores na resolução e análise. O aspecto positivo, é que sei como analisar e como pensar na estruturação de um problema relacionado a um determinado comando, e como pensar na resolução e solução dele. O aspecto negativo, é que assuntos mais a frente, prejudicam aqueles que estão mais atrasados, claro, de acordo de como for avaliado tal atividade"
	"É uma boa atividade. Permite que alunos criem suas próprias questões com suas próprias entradas e saídas, fazendo assim, com que ele possa se desafiar mais no futuro. Poderia haver troca de questões entre alunos, um aluno faria a questão de outro e vice-versa"
E	"Pontos positivos: estimula a criatividade e vontade de superar as respostas erradas, e não querer apenas consertar a questão, e sim criar uma nova. Negativos: nível muito alto da maioria das questões e um tempo inferior a quantidade e dificuldade das questões"
	"Achei a atividade interessante, com cunho de construção de conhecimento para ajudar os alunos e a disciplina a evoluírem. O aspecto negativo foi o tamanho da atividade para um simulado e exposição dos e-mails de colegas"

motivada pelos pontos negativos que encontramos em alguns comentários que questionam o nível de dificuldade, o tamanho de resolução das questões e como eles seriam avaliados nesse tipo de questão. Durante esse trabalho, não foi possível tempo suficiente para realizar avaliação da qualidade das atividades usando critérios bem definidos, e nem sugerir estratégias precisas de validação das respostas das atividades personalizadas. Portanto foi elencada como uma sugestão de trabalhos futuros para dar continuidade a essa pesquisa.

Outro ponto negativo levantado foi a exposição do endereço do personagem da atividade. Foram utilizados nomes e e-mails fictícios, embora a situação reportada na atividade personalizada fosse inspirada em uma situação real. Foi possível notar que

o estabelecimento da confiança que está sendo reportada uma situação real não é exatamente a relação entre o respondente e o personagem reportado na atividade, mas o fato de estar colaborando com alguém que realmente está com dificuldades. Avaliar se o personagem retratado na atividade se sente constrangida também é uma sugestão de trabalhos futuros.

6.5 Trabalhos Relacionados

Os trabalhos relacionados a esta pesquisa se concentram em dois temas, aqueles que exploram aspectos de engajamento e da pedagogia de contribuição do estudante, ambos no contexto da computação.

6.5.1 Engajamento de Estudantes

Existem diversos estudos sobre o envolvimento de estudantes especialmente em ambientes de aprendizagem online. Em geral, seu foco está na forma de interação e no design do ambiente de aprendizagem, no entanto foram listadas nessa seção pesquisas que dão atenção especial ao envolvimento do estudante na educação em computação.

Rahila (Umer et al. 2018) teve como objetivo prever os estudantes que estão em risco de falhar no curso. O artigo apresenta uma análise das informações relacionadas às atribuições e dos dados de engajamento. A pesquisa considera engajamento como sendo o resultado da interação do usuário com o sistema através de ações como: visualização de módulos de curso, leitura de postagens em fóruns e submissão de trabalhos. Os resultados mostraram que as tarefas são mais determinantes para o desempenho do que os dados de engajamento. Gray (Gray e DiLoreto 2016) examinaram os fatores que afetam tanto os resultados de aprendizado dos estudantes quanto a satisfação destes em cursos online. O estudo concluiu que a interação do estudante não tem um impacto significativo na satisfação do mesmo. Embora o método de ensino adotado e a frequência de interação com o instrutor apresentaram uma relação direta com o desempenho dos estudantes no curso. Dixson (Dixson 2010) tentou descobrir quais atividades poderiam aumentar o engajamento dos estudantes. A conclusão do estudo foi que não há diferença significativa nos níveis de envolvimento do estudante entre aqueles que relatam atividades ativas versus atividades passivas. O autor sugere que os

instrutores devem criar oportunidades para os estudantes interagirem uns com os outros, não apenas com o conteúdo, como forma de melhorar o engajamento.

Zhang et al. (Zhang e Meng 2016) analisaram o envolvimento de estudantes no trabalho desenvolvido em equipe com o suporte do Slack. Essa ferramenta foi utilizada para se comunicar, coordenar tarefas, fazer *upload* de documentos e propor novas ideias. Todas essas atividades foram vistas por todos os membros da equipe e monitoradas pelo professor. Os resultados sugerem os seguintes aspectos como importantes para incentivar a participação no trabalho em equipe: confiança mútua, influência social, a recompensa do trabalho e o sucesso pessoal.

Budd (Budd 2006) apresenta um relato sobre a experiência inicial de ensino de estrutura de dados utilizando uma abordagem ativa. As aulas começam com uma curta palestra do professor introduzindo o conceito de tabela *hash*. Os estudantes recebem um rascunho de uma implementação de uma tabela *hash* e uma planilha contendo *quizzes*. Essa planilha é utilizada para estimular os estudantes a trabalhar cooperativamente, fazer perguntas, usar o livro e fazer o que for necessário para garantir que todos tenham a resposta correta até o final. O professor atuou como mediador, dando conselhos e guiando os estudantes a progredir. O autor finaliza com a opinião positiva dos estudantes sobre o método de ensino adotado.

A principal diferença entre os artigos mencionados acima e a essa pesquisa são os indicadores de engajamento de estudante. Nesse estudo, foram utilizadas métricas relacionadas com a realização de atividades, enquanto esses trabalhos usam a frequência de interação, não importando a função que está sendo acionada no ambiente de aprendizagem.

6.5.2 Pedagogia de Contribuição do Estudante

Em computação, há registros de trabalhos que estimulam estudantes a criar uma pergunta que avalie um ou mais resultados de aprendizagem de um curso. Além da criação de uma pergunta, os estudantes precisam escrever uma solução e responder perguntas geradas por outros indivíduos. Ao escrever perguntas, os estudantes aprendem a se expressar usando linguagem clara e efetiva, e a avaliar seu próprio entendimento sobre o conteúdo do curso. Especificamente, quando escreve uma pergunta cuja resposta exige um programa, o estudante deve escrever especificações, normalmente incluindo um exemplo de entrada e saída. Além de exigir que o autor reflita se a complexidade e os coman-

dos necessários na resposta estão adequados ao público a quem se destina o exercício (Hamer 2006, Luxton-Reilly e Denny 2010, Falkner e Falkner 2012, Etlinger 2013).

A literatura registra que os benefícios da aplicação da pedagogia de contribuição podem ser causados pelas habilidades do processo de geração e avaliação do material, e da auto-avaliação do conhecimento do conteúdo (Hamer et al. 2008), e também pelas habilidades interpessoais (Gol e Nafalski 2007). Em relação ao entendimento do conteúdo, (Denny et al. 2008) relatam um estudo de correlação com o desempenho em provas de uma turma que usou um banco de perguntas de múltipla escolha geradas por estudantes, como uma ferramenta de aprendizado. Os estudantes que contribuíram com questões tiveram um desempenho melhor no exame do que aqueles que não o fizeram.

Em (Hamer et al. 2010), há uma revisão da literatura sobre Educação em Ciência da Computação para identificar casos em que o PCE é aplicado, identificando as características das ferramentas e quais os tipos de atividades para as quais elas são usadas. Em sua pesquisa, foram encontradas ferramentas utilizadas para diversos propósitos:

- **revisão por pares:** os estudantes analisam e fazem comentários sobre o trabalho um dos outros;
- **comunicação e discussão:** a contribuição ocorre na troca de informações entre os aprendizes;
- **Anotação:** os estudantes comentam e compartilham seus comentários sobre os materiais que *não* foram desenvolvidos por estudantes;
- **construção de recurso:** os estudantes criam novos materiais que devem ser incorporados ao curso para que outros aprendizes possam utilizá-los em seus estudos. Nesse contexto, os artefatos produzidos são um tipo de material estático que os estudantes poderiam obter de um livro, por exemplo;
- **compartilhamento de soluções:** os estudantes compartilham suas próprias soluções com outros estudantes;
- **fazer links:** estudantes que fazem conexões com conceitos conhecidos ou localizam recursos externos relacionados ao conteúdo que está sendo estudado.

- **criação de atividades:** os aprendizes criam atividades para outros alunos.

Duas ferramentas desenvolvidas se destacam no suporte à criação de atividades: *PeerWise* e *StudySieve*. O *PeerWise* permite que um estudante crie perguntas de múltipla escolha para que outros estudantes respondam, forneçam feedback e discutam (Denny et al. 2008). O *StudySieve* apoia a criação de atividades de aprendizado permitindo a formulação de perguntas mais genéricas (Luxton-Reilly et al. 2012). A principal diferença entre ambas é o *feedback* imediato sobre a corretude da solução de quem responde à pergunta. Isso se deve à dificuldade de especificar o gabarito de uma resposta subjetiva, principalmente quando o seu resultado é um programa. Assim, apenas *PeerWise* fornece o *feedback* imediato sobre se a resposta está correta ou não.

6.6 Conclusão

Esse capítulo descreveu as principais dificuldades que enfrentamos durante a realização dessa pesquisa de doutorado, detalhando desafios burocráticos e específicos do tipo de pesquisa realizada. Como também, foram apresentadas as consequências da aplicação do modelo de personalização em um curso de programação que representa parte das lições aprendidas durante o desenvolvimento da pesquisa.

Foi debatido o resultado dos estudos de acordo com o impacto no engajamento em cada uma de suas dimensões. E detalhou-se a opinião dos estudantes ao depararem com atividades personalizadas de programação.

Por fim, foram apresentados trabalhos relacionados a esta pesquisa, com foco em intervenções para melhoria de engajamento de estudantes de programação, e na pedagogia de contribuição.

Capítulo 7

Conclusão

Neste trabalho, foi apresentado um modelo de personalização de atividades para estimular o engajamento de aprendizes de programação. Tal modelo descreve como utilizar dúvidas e experiências vivenciadas por estudantes para elaborar novos estilos de exercícios de programação. Foram descritos quais os tipos de atividades e os atributos necessários para personalizar exercícios de programação, assim como foram apresentados tipos de atividades personalizadas, como sendo criação de novos exercícios de programação, sugestão de novos casos de teste, análise do texto de atividades existentes ou revisão do código de outras pessoas.

A realização de atividades personalizadas resulta em indicadores que sinalizam o envolvimento do estudante de programação, portanto foram sugeridas métricas para mensurar as três dimensões do engajamento. Como engajamento comportamental, foram propostas métricas para sinalizar: dedicação, persistência, hábito de estudo e participação. Assim como medições do esforço cognitivo e estados emocionais.

Destarte, as principais contribuições deste trabalho de doutorado foram:

- O modelo de personalização de exercícios de programação;
- As métricas que sinalizam o engajamento comportamental, cognitivo e os estados emocionais do estudante de programação;
- Um conjunto de lições aprendidas sobre a colaboração e o engajamento de aprendizes de programação;

A principal contribuição dessa tese se baseia nas lições aprendidas com a proposta do modelo de personalização de atividades de programação e sua avaliação em um curso de programação introdutória. As lições e as discussões durante este o desenvolvimento desta pesquisa, bem como o modelo proposto estão servindo como inspiração para criação de novas estratégias de ensino adotadas pelos professores da disciplina de Programação utilizada como nosso ambiente de experimentação.

Os estudos foram realizados em um ambiente de ensino de programação introdutória e foi analisado o engajamento de estudantes em diferentes situações. Foram realizados estudos experimentais para conhecer o impacto no engajamento da aplicação das atividades personalizadas. Os resultados não encontraram diferenças significativas no engajamento comportamental. Apenas em um cenário, foi perceptível melhoria do aspecto cognitivo. No outro, não houve diferença. Com relação ao engajamento emocional, os resultados mostraram que os participantes se sentiram mais poderosos e tranquilos ao realizar atividades personalizadas, e menos tristes e assustados.

Também foi realizado um estudo de caso que mostrou o comportamento de aprendizes de programação ao interagir com um sistema de gerência de aprendizagem. Foi constatado que, contrário ao pensamento comum, os estudantes não só concentram seus estudos na véspera da prova. O horário que os estudantes mais interagem no sistema é aquele turno seguinte às aulas teóricas e práticas.

Além disso, vale ressaltar que, durante o trabalho de doutorado, dois trabalhos foram publicados em conferências reconhecidas na área de educação em computação. O primeiro deles foi resultado de um trabalho de disciplina e teve como objetivo compreender se estilos de aprendizagem influencia no aprendizagem de programação (ver Apêndice H):

- Maia, Mirna Carelli Oliveira, Dalton Serey, and Jorge Figueiredo. "Learning styles in programming education: a systematic mapping study." 2017 IEEE Frontiers in Education Conference (FIE). IEEE, 2017.

O segundo artigo científico apresenta as métricas de engajamento extraídas do ambiente online de aprendizagem e que representam uma de nossas principais contribuições:

- Maia, Mirna Carelli Oliveira, Jorge Figueiredo, and Dalton Serey. "Online Student Engagement: A Case Study in Teaching of Programming." 2019 Simpósio Brasileiro

de Informática na Educação. SBIE 2019.

7.1 Trabalhos Futuros

Essa pesquisa de doutorado desperta oportunidades para trabalhos futuros em diferentes direções. Algumas são orientações para melhoria deste trabalho e outras são propostas de novos estudos sobre os temas que foram aqui abordados.

- **Avaliação dos Efeitos no Engajamento da Personalização de Atividades** - estudo longitudinal para detectar efeitos no engajamento dos estudantes na realização de atividades personalizadas de programação. Em especial, aspectos cognitivos foram pouco explorados durante essa pesquisa de doutorado. Acredita-se que é possível que o impacto na cognição seja melhor investigada com estudo de duração mais longa, como um longitudinal;
- **Engajamento** - a pesquisa em engajamento é considerada recente. À medida que o perfil dos estudantes muda, é preciso definir novas estratégias para garantir que eles se mantenham envolvidos com o processo de aprendizagem mesmo diante das dificuldades. Esse tema abre um campo amplo para pesquisa, desde a sugestão de novas métricas, métodos e instrumentos para detecção de engajamento, até as metodologias de ensino que promovem a sua melhoria. Especialmente, explorar pesquisas envolvendo a métrica ritmo de aprendizagem e como pode ser aplicada em outros contextos;
- **Reputação de Colaborador** - Propor e analisar estratégias que possam indicar a reputação de um colaborador. Essa reputação pode ser determinada por métricas que levariam em consideração a qualidade de suas revisões, testes e novas questões. Ou até mesmo pela avaliação dos demais indivíduos envolvidos no processo;
- **Avaliação da Qualidade de Exercícios** - Propor e avaliar estratégias para avaliar a qualidade dos exercícios propostos por estudantes. Pode ser feita de forma automática ou levando em consideração a opinião dos participantes do processo de colaboração.

Referências Bibliográficas

Anthony 1996 ANTHONY, G. Active learning in a constructivist framework. *Educational studies in mathematics*, Springer, v. 31, n. 4, p. 349–369, 1996.

Araujo et al. 2013 ARAUJO, E. C. de et al. O papel do hábito de estudo no desempenho do aluno de programação. In: *XXI Workshop sobre Educação em Informática*. [S.l.: s.n.], 2013.

Axelson e Flick 2010 AXELSON, R. D.; FLICK, A. Defining student engagement. *Change: The Magazine of Higher Learning*, Routledge, v. 43, n. 1, p. 38–43, 2010. Disponível em: <<https://doi.org/10.1080/00091383.2011.533096>>.

Bonwell e Eison 1991 BONWELL, C. C.; EISON, J. A. *Active Learning: Creating Excitement in the Classroom*. 1991 ASHE-ERIC Higher Education Reports. [S.l.]: ERIC, 1991.

Budd 2006 BUDD, T. A. An active learning approach to teaching the data structures course. *SIGCSE Bull.*, ACM, New York, NY, USA, v. 38, n. 1, p. 143–147, mar. 2006. ISSN 0097-8418. Disponível em: <<http://doi.acm.org/10.1145/1124706.1121388>>.

Butler et al. 2016 BUTLER, M. et al. Comparing international indicators of student engagement for computer science. In: *ACM. Proceedings of the Australasian Computer Science Week Multiconference*. [S.l.], 2016. p. 6.

Carini, Kuh e Klein 2006 CARINI, R. M.; KUH, G. D.; KLEIN, S. P. Student engagement and student learning: Testing the linkages. *Research in higher education*, Springer, v. 47, n. 1, p. 1–32, 2006.

Christenson, Reschly e Wylie 2012 CHRISTENSON, S. L.; RESCHLY, A. L.; WYLIE, C. *Handbook of research on student engagement*. [S.l.]: Springer Science & Business Media, 2012.

Cloud 2020 CLOUD, G. *G Suite*. 2020. Disponível em: <<https://gsuite.google.com.br/intl/pt-BR/features/>>. Acesso em: Fevereiro 2020.

Coates 2005 COATES, H. The value of student engagement for higher education quality assurance. *Quality in Higher Education*, Taylor and Francis Ltd, v. 11, n. 1, p. 25–36, 2005.

Collis e Moonen 2005 COLLIS, B.; MOONEN, J. An on-going journey: Technology as a learning workbench. University of Twente, Netherlands, 9 2005.

Collis e Moonen 2006 COLLIS, B.; MOONEN, J. The contributing student: Learners as co-developers of learning resources for reuse in web environments. In: *Engaged learning with emerging technologies*. [S.l.]: Springer, 2006. p. 49–67.

- Corney, Teague e Thomas 2010 CORNEY, M.; TEAGUE, D.; THOMAS, R. N. Engaging students in programming. In: AUSTRALIAN COMPUTER SOCIETY, INC. *Proceedings of the Twelfth Australasian Conference on Computing Education-Volume 103*. [S.l.], 2010. p. 63–72.
- Denny et al. 2008 DENNY, P. et al. Peerwise: students sharing their multiple choice questions. In: ACM. *Proceedings of the fourth international workshop on computing education research*. [S.l.], 2008. p. 51–58.
- Dicio 2020 DICIO. *Dicionário Online de Português, definições e significados de mais de 400 mil palavras. Todas as palavras de A a Z*. 2020. Disponível em: <<https://www.dicio.com.br/persistencia/>>. Acesso em: Maio 2020.
- Dixson 2010 DIXSON, M. D. Creating effective student engagement in online courses: What do students find engaging? *Journal of the Scholarship of Teaching and Learning*, p. 1–13, 2010.
- Etlinger 2013 ETLINGER, H. A. Adding a contributing student pedagogy component to an introductory database course. In: *Proceeding of the 44th ACM Technical Symposium on Computer Science Education*. New York, NY, USA: ACM, 2013. (SIGCSE '13), p. 299–304. ISBN 978-1-4503-1868-6. Disponível em: <<http://doi.acm.org/10.1145/2445196.2445288>>.
- Falkner e Falkner 2012 FALKNER, K.; FALKNER, N. J. Supporting and structuring contributing student pedagogy in computer science curricula. *Computer Science Education*, v. 22, n. 4, p. 413–443, 2012. Disponível em: <<http://dx.doi.org/10.1080/08993408.2012.727713>>.
- Farrington et al. 2012 FARRINGTON, C. A. et al. *Teaching Adolescents to Become Learners: The Role of Noncognitive Factors in Shaping School Performance—A Critical Literature Review*. [S.l.]: ERIC, 2012.
- Finn 1989 FINN, J. D. Withdrawing from school. *Review of educational research*, Sage Publications Sage CA: Thousand Oaks, CA, v. 59, n. 2, p. 117–142, 1989.
- Finn, Pannozzo e Voelkl 1995 FINN, J. D.; PANNOZZO, G. M.; VOELKL, K. E. Disruptive and inattentive-withdrawn behavior and achievement among fourth graders. *The Elementary School Journal*, University of Chicago Press, v. 95, n. 5, p. 421–434, 1995.
- Finn e Zimmer 2012 FINN, J. D.; ZIMMER, K. S. Student engagement: What is it? why does it matter? In: *Handbook of research on student engagement*. [S.l.]: Springer, 2012. p. 97–131.
- Fredricks, Blumenfeld e Paris 2004 FREDRICKS, J. A.; BLUMENFELD, P. C.; PARIS, A. H. School engagement: Potential of the concept, state of the evidence. *Review of Educational Research*, v. 74, n. 1, p. 59–109, 2004. Disponível em: <<http://dx.doi.org/10.3102/00346543074001059>>.
- Fredricks e McColskey 2012 FREDRICKS, J. A.; MCCOLSKEY, W. The measurement of student engagement: A comparative analysis of various methods and student self-report instruments. In: *Handbook of research on student engagement*. [S.l.]: Springer, 2012. p. 763–782.

- Gol e Nafalski 2007 GOL, O.; NAFALSKI, A. *Collaborative learning in engineering education*. Tese (Doutorado) — Unesco, International Centre for Engineering Education, 2007.
- Gray e DiLoreto 2016 GRAY, J. A.; DILORETO, M. The effects of student engagement, student satisfaction, and perceived learning in online learning environments. *International Journal of Educational Leadership Preparation*, ERIC, v. 11, n. 1, p. n1, 2016.
- Grove e Education 2019 GROVE, A.; EDUCATION, G.-W. U. S. of. *The Teacher's Role in Student Engagement*. Gardner-Webb University, 2019. (Doctor of Education dissertation). Disponível em: <<https://books.google.com.br/books?id=AdtFzQEACAAJ>>.
- Guerrero 2019 GUERRERO, D. D. S. *Students assignments testing*. 2019. Disponível em: <<https://github.com/daltonserrey/tst>>. Acesso em: 15 Setembro 2019.
- Hamer 2006 HAMER, J. Some experiences with the "contributing student approach". In: *Proceedings of the 11th Annual SIGCSE Conference on Innovation and Technology in Computer Science Education*. New York, NY, USA: ACM, 2006. (ITICSE '06), p. 68–72. ISBN 1-59593-055-8. Disponível em: <<http://doi.acm.org/10.1145/1140124.1140145>>.
- Hamer et al. 2008 HAMER, J. et al. Contributing student pedagogy. *SIGCSE Bull.*, ACM, New York, NY, USA, v. 40, n. 4, p. 194–212, nov. 2008. ISSN 0097-8418. Disponível em: <<http://doi.acm.org/10.1145/1473195.1473242>>.
- Hamer et al. 2010 HAMER, J. et al. Tools for "contributing student learning". In: *Proceedings of the 2010 ITiCSE Working Group Reports*. New York, NY, USA: ACM, 2010. (ITiCSE-WGR '10), p. 1–14. ISBN 978-1-4503-0677-5. Disponível em: <<http://doi.acm.org/10.1145/1971681.1971683>>.
- Hamer et al. 2012 HAMER, J. et al. Contributing student pedagogy. *Computer Science Education*, Routledge, v. 22, n. 4, p. 315–318, 2012. Disponível em: <<https://doi.org/10.1080/08993408.2012.727709>>.
- Kahn 1990 KAHN, W. A. Psychological conditions of personal engagement and disengagement at work. *Academy of management journal*, Academy of Management Briarcliff Manor, NY 10510, v. 33, n. 4, p. 692–724, 1990.
- Kirschner, Sweller e Clark 2006 KIRSCHNER, P. A.; SWELLER, J.; CLARK, R. E. Why minimal guidance during instruction does not work: An analysis of the failure of constructivist, discovery, problem-based, experiential, and inquiry-based teaching. *Educational psychologist*, Taylor & Francis, v. 41, n. 2, p. 75–86, 2006.
- Kuh 2001 KUH, G. D. Assessing what really matters to student learning inside the national survey of student engagement. *Change: The Magazine of Higher Learning*, Taylor & Francis Group, v. 33, n. 3, p. 10–17, 2001.
- Kuh 2003 KUH, G. D. What we're learning about student engagement from nsse: Benchmarks for effective educational practices. *Change: The Magazine of Higher Learning*, Taylor & Francis, v. 35, n. 2, p. 24–32, 2003.
- Kuh 2007 KUH, G. D. What student engagement data tell us about college readiness. *Peer Review*, Citeseer, v. 9, n. 1, p. 4–8, 2007.

- Kuh 2009 KUH, G. D. The national survey of student engagement: Conceptual and empirical foundations. *New directions for institutional research*, Wiley Online Library, v. 2009, n. 141, p. 5–20, 2009.
- Linnenbrink e Pintrich 2003 LINNENBRINK, E. A.; PINTRICH, P. R. The role of self-efficacy beliefs in student engagement and learning in the classroom. *Reading & Writing Quarterly*, Taylor & Francis, v. 19, n. 2, p. 119–137, 2003.
- Luxton-Reilly et al. 2018 LUXTON-REILLY, A. et al. Introductory programming: a systematic literature review. In: ACM. *Proceedings Companion of the 23rd Annual ACM Conference on Innovation and Technology in Computer Science Education*. [S.l.], 2018. p. 55–106.
- Luxton-Reilly e Denny 2010 LUXTON-REILLY, A.; DENNY, P. Constructive evaluation: a pedagogy of student-contributed assessment. *Computer Science Education*, Taylor & Francis, v. 20, n. 2, p. 145–167, 2010.
- Luxton-Reilly et al. 2012 LUXTON-REILLY, A. et al. Activities, affordances and attitude: How student-generated questions assist learning. In: *Proceedings of the 17th ACM Annual Conference on Innovation and Technology in Computer Science Education*. New York, NY, USA: ACM, 2012. (ITiCSE '12), p. 4–9. ISBN 978-1-4503-1246-2. Disponível em: <<http://doi.acm.org/10.1145/2325296.2325302>>.
- Maia, Figueiredo e Serey 2019 MAIA, M. C.; FIGUEIREDO, J. C. A. de; SEREY, D. Online student engagement: A case study in teaching of programming. In: *Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação-SBIE)*. [S.l.: s.n.], 2019. v. 30, n. 1, p. 51.
- McWilliam, Trivette e Dunst 1985 MCWILLIAM, R.; TRIVETTE, C. M.; DUNST, C. J. Behavior engagement as a measure of the efficacy of early intervention. *Analysis and Intervention in Developmental Disabilities*, Elsevier, v. 5, n. 1-2, p. 59–71, 1985.
- Morán 2015 MORÁN, J. Mudando a educação com metodologias ativas. *Coleção Mídias Contemporâneas. Convergências Midiáticas, Educação e Cidadania: aproximações jovens*, v. 2, n. 1, p. 15–33, 2015.
- Morgan et al. 2018 MORGAN, M. et al. How cs academics view student engagement. In: *Proceedings of the 23rd Annual ACM Conference on Innovation and Technology in Computer Science Education*. New York, NY, USA: ACM, 2018. (ITiCSE 2018), p. 284–289. ISBN 978-1-4503-5707-4. Disponível em: <<http://doi.acm.org/10.1145/3197091.3197092>>.
- Morgan et al. 2018 MORGAN, M. et al. Understanding international benchmarks on student engagement: awareness and research alignment from a computer science perspective. In: *Proceedings of the 2017 ITiCSE Conference on Working Group Reports*. [S.l.: s.n.], 2018. p. 1–24.
- Parsons e Taylor 2011 PARSONS, J.; TAYLOR, L. Improving student engagement. *Current issues in education*, v. 14, n. 1, 2011.
- Pears 2010 PEARS, A. N. Enhancing student engagement in an introductory programming course. In: IEEE. *2010 IEEE Frontiers in Education Conference (FIE)*. [S.l.], 2010. p. F1E–1.

- Prince 2004 PRINCE, M. Does active learning work? a review of the research. *Journal of engineering education*, Wiley Online Library, v. 93, n. 3, p. 223–231, 2004.
- Rumberger e Lim 2008 RUMBERGER, R. W.; LIM, S. A. Why students drop out of school: A review of 25 years of research. *California Dropout Research Project*, Gevirtz Graduate School of Education Santa Barbara, v. 15, p. 1–3, 2008.
- Scaico 2018 SCAICO, P. D. *Um Estudo Sobre o Desenvolvimento de Interesse Pela Aprendizagem de Programação*. Tese (Doutorado) — Universidade Federal de Pernambuco, 4 2018.
- Scherer 2005 SCHERER, K. R. What are emotions? and how can they be measured? *Social science information*, Sage Publications Sage CA: Thousand Oaks, CA, v. 44, n. 4, p. 695–729, 2005.
- Sheard et al. 2009 SHEARD, J. et al. Analysis of research into the teaching and learning of programming. In: *Proceedings of the fifth international workshop on Computing education research workshop*. [S.l.: s.n.], 2009. p. 93–104.
- Silva 2011 SILVA, C. d. *Design emocional: afetos positivos e negativos nas interações com o ambiente web. 2011*. Tese (Doutorado) — Tese (Doutorado)-Universidade Federal de Santa Catarina, Florianópolis, 2011.
- Sinclair et al. 2015 SINCLAIR, J. et al. Measures of student engagement in computer science. In: *Proceedings of the 2015 ACM Conference on Innovation and Technology in Computer Science Education*. New York, NY, USA: ACM, 2015. (ITiCSE '15), p. 242–247. ISBN 978-1-4503-3440-2. Disponível em: <<http://doi.acm.org/10.1145/2729094.2742586>>.
- Skinner e Belmont 1993 SKINNER, E. A.; BELMONT, M. J. Motivation in the classroom: Reciprocal effects of teacher behavior and student engagement across the school year. *Journal of educational psychology*, American Psychological Association, v. 85, n. 4, p. 571, 1993.
- Technologies 2019 TECHNOLOGIES, S. *Slack*. 2019. Disponível em: <<https://slack.com/intl/pt-br/features>>. Acesso em: Setembro de 2019.
- Trowler 2010 TROWLER, V. Student engagement literature review. *The higher education academy*, v. 11, p. 1–15, 2010.
- Umer et al. 2018 UMER, R. et al. A learning analytics approach: Using online weekly student engagement data to make predictions on student performance. In: IEEE. *2018 International Conference on Computing, Electronic and Electrical Engineering (ICE Cube)*. [S.l.], 2018. p. 1–5.
- Vihavainen, Airaksinen e Watson 2014 VIHAVAINEN, A.; AIRAKSINEN, J.; WATSON, C. A systematic review of approaches for teaching introductory programming and their influence on success. In: *Proceedings of the Tenth Annual Conference on International Computing Education Research*. New York, NY, USA: ACM, 2014. (ICER '14), p. 19–26. ISBN 978-1-4503-2755-8. Disponível em: <<http://doi.acm.org/10.1145/2632320.2632349>>.

Zhang e Meng 2016 ZHANG, X.; MENG, Y. Students' engagement in collaborative learning group supported by communication tools: an empirical study. In: *PACIS*. [S.l.: s.n.], 2016. p. 55.

Apêndice A

Parecer do Comitê de Ética

PARECER CONSUBSTANCIADO DO CEP

DADOS DA EMENDA

Título da Pesquisa: Uma Abordagem Colaborativa para Ensino de Introdução à Programação

Pesquisador: Mirna Carelli Oliveira Maia

Área Temática:

Versão: 3

CAAE: 69427916.4.0000.5182

Instituição Proponente: UNIVERSIDADE FEDERAL DE CAMPINA GRANDE

Patrocinador Principal: Financiamento Próprio

DADOS DO PARECER

Número do Parecer: 3.080.140

Apresentação do Projeto:

Trata-se de projeto que tem como instituição proponente a Universidade Federal de Campina Grande, sem instituições Coparticipantes. É um estudo experimental que engloba uma serie de estudos que visam explorar estratégias do "cooperative learning" para auxiliar os alunos a aprender a programar.

Os sujeitos do estudo serao alunos universitários vinculados à disciplina de Introdução à Programação do curso de Ciência da Computação na UFCG, sejam alunos matriculados, sejam alunos monitores ou alunos em estágio docência. Os estudos englobam surveys e estudos longitudinais onde os alunos participam através de criação de conteúdo e criação colaborativa de conteúdo.

Objetivo da Pesquisa:

Objetivo primario

Investigar os efeitos do uso de estratégias de aprendizagem colaborativa no contexto de um curso de programacao introdutoria.

Objetivos secundarios

1. Adaptar estratégias de ensino cooperativo e colaborativo para o contexto de introducao a programacao;

Endereço: Rua: Dr. Carlos Chagas, s/ n

Bairro: São José

CEP: 58.107-670

UF: PB **Município:** CAMPINA GRANDE

Telefone: (83)2101-5545

Fax: (83)2101-5523

E-mail: cep@huac.ufcg.edu.br

Continuação do Parecer: 3.080.140

2. Avaliar o efeito da abordagem cooperativa e colaborativa no ensino de programação;
3. Avaliar o efeito dessas estratégias no comportamento dos alunos;

Avaliação dos Riscos e Benefícios:

Riscos: Os participantes da pesquisa podem se sentir intimidados ou desconfortáveis em realizar alguma etapa desta pesquisa. Neste caso, eles podem desistir de participar a qualquer momento sem nenhum onus e receberão apoio necessário a questão apresentada.

Benefícios: A pesquisa apresenta benefícios diretos para os participantes, já que para a coleta de dados, serão realizados estudos que lhes farão refletir sobre a sua participação no contexto da aprendizagem de programação, algo que poderá influenciar o estabelecimento de novos hábitos e atitudes em relação ao hábito de estudo.

Comentários e Considerações sobre a Pesquisa:

Trata-se de um trabalho bastante interessante que visa a desenvolver estratégias de aprendizagem colaborativa no contexto de um curso de programação introdutória para a melhoria da disciplina e, por consequência, do curso.

Considerações sobre os Termos de apresentação obrigatória:

Toda a documentação obrigatória foi apresentada, trata-se de uma emenda de um projeto já aprovado.

Conclusões ou Pendências e Lista de Inadequações:

A pesquisadora solicita a prorrogação do prazo para a conclusão deste projeto, pois será necessário realizar mais três experimentos durante o semestre 2019.1 da graduação. Trata-se de um projeto de pesquisa já aprovado pelo CEP-HUAC, em conformidade com a Resolução no 466/ 2012 em seus aspectos éticos.

Considerações Finais a critério do CEP:

Este parecer foi elaborado baseado nos documentos abaixo relacionados:

Tipo Documento	Arquivo	Postagem	Autor	Situação
Informações Básicas do Projeto	PB_INFORMAÇÕES_BASICAS_1214805_E2.pdf	22/10/2018 21:21:43		Aceito
Outros	FormulariosGoogle.PDF	22/10/2018 21:20:45	Mirna Carelli Oliveira Maia	Aceito
Projeto Detalhado / Brochura	UmaAbordagemColaborativaparaEnsino deIntroducaoProgramacao.PDF	22/10/2018 21:20:01	Mirna Carelli Oliveira Maia	Aceito

Endereço: Rua: Dr. Carlos Chagas, s/ n
Bairro: São José **CEP:** 58.107-670
UF: PB **Município:** CAMPINA GRANDE
Telefone: (83)2101-5545 **Fax:** (83)2101-5523 **E-mail:** cep@huac.ufcg.edu.br

UFCG - HOSPITAL
UNIVERSITÁRIO ALCIDES
CARNEIRO DA UNIVERSIDADE



Continuação do Parecer: 3.080.140

Investigador	UmaAbordagemColaborativaparaEnsino deIntroducaoProgramacao.PDF	22/10/2018 21:20:01	Mirna Carelli Oliveira Maia	Aceito
TCLE / Termos de Assentimento / Justificativa de Ausência	TCLE.pdf	22/10/2018 21:15:13	Mirna Carelli Oliveira Maia	Aceito
Folha de Rosto	folhaDeRostoAssinada.pdf	25/05/2017 10:46:25	Mirna Carelli Oliveira Maia	Aceito
Outros	CartaDeAnuenciaAssinada.pdf	25/05/2017 08:40:17	Mirna Carelli Oliveira Maia	Aceito
Outros	DeclaracaoparaDivulgacaoAssinado.pdf	22/05/2017 11:17:26	Mirna Carelli Oliveira Maia	Aceito
Declaração de Pesquisadores	TermoDeCompromissoAssinado.pdf	22/05/2017 11:04:55	Mirna Carelli Oliveira Maia	Aceito

Situação do Parecer:

Aprovado

Necessita Apreciação da CONEP:

Não

CAMPINA GRANDE, 13 de Dezembro de 2018

Assinado por:
Andréia Oliveira Barros Sousa
(Coordenador(a))

Endereço: Rua: Dr. Carlos Chagas, s/ n
Bairro: São José **CEP:** 58.107-670
UF: PB **Município:** CAMPINA GRANDE
Telefone: (83)2101-5545 **Fax:** (83)2101-5523 **E-mail:** cep@huac.ufcg.edu.br

Apêndice B

Online Student Engagement: A Case Study in Teaching of Programming

Online Student Engagement: A Case Study in Teaching of Programming

Mirna Carelli O. Maia¹, Jorge C. A. de Figueiredo², Dalton Dario Serey Guerrero²

¹Instituto Federal de Educação, Ciência e Tecnologia da Paraíba (IFPB)
Campina Grande - Paraíba - Brazil

²Universidade Federal de Campina Grande (UFCG)
Campina Grande - Paraíba - Brazil

mirna.carelli@academico.ifpb.edu.br

(abrantes,dalton)@computacao.ufcg.edu.br

Abstract. *Student engagement is a key indicator of student academic success, especially in the online environment, where students can study anywhere, anytime. In this paper, we analyze track records of the novice programmers' interaction with an online learning environment. The results show indicators of poor and good engagement and can guide teachers in choosing the most appropriate teaching strategy according to the needs of each individual.*

Resumo. *O envolvimento do aluno é um indicador fundamental para o sucesso acadêmico do aluno, principalmente no ambiente on-line, onde os alunos podem estudar em qualquer lugar e a qualquer momento. Neste trabalho, analisamos registros da interação do aluno iniciante em programação com um ambiente de aprendizado on-line. Os resultados apresentam indicadores de fraco e bom engajamento, e podem guiar os professores na escolha de estratégia de ensino mais adequadas de acordo com a necessidade de cada indivíduo.*

1. Introduction

Several factors influence student learning experience. The method of teaching, students' background, and engagement are an example of these factors. Recently, researchers focused their attention on student engagement because there is a robust correlation with positive outcomes of student success [Fredricks and McColskey 2012]

Student engagement is concerned with "the interaction between the time, effort and other relevant resources invested by both students and their institutions intended to optimize the student experience and enhance the learning outcomes and development of students and the performance, and reputation of the institution" [Trowler 2010, Fredricks and McColskey 2012].

Some academic behaviors can signal student engagement. They include regularly attending class, paying attention, participating in instructional activities and class discussions, devoting out-of-school time to studying and completing homework, among others [Farrington et al. 2012].

Creating an enjoyable educational environment and encouraging student engagement is a big challenge [Harris 2008]. A way to improve the learning experience is the

usage of technology to support education. The technological support allows students to stay connected to the learning environment anywhere and anytime, changing the learning experience [Coates 2007].

Learning Management System (LMS) aims to support students in dealing with difficulties and improving their performance. It provides tools for course administration and pedagogical functions for designing, building, and delivering scalable learning environments. They help the resource delivery, communication, keep track of activities and assessments, collaborative work, and student management [Dixson 2015].

This study explores the online student engagement of a programming course. We developed a study case in an introductory programming course in which instructors adopted a blended strategy of teaching: face-to-face and online. In a face-to-face class, teachers expose lectures and clarify student doubts. Whereas in practice classes, students answer quizzes and submit the answer of programming assignments through an online system.

In programming education, students develop several programming assignments using LMS. In this study, we tracked the students' interaction with LMS in order to outline their online engagement. So, we intend to profile the novice programmer's behavior in order to help teachers to know who can be on the right learning track and those who can need additional help.

To achieve this objective, we propose a set of indicators of student engagement and correlate them to academic performance. Results show that these metrics can signal how learner progress in the course and allow teachers for monitoring the development of learning activities and define suitable teaching strategies tailored to the needs of each student.

2. Background

2.1. Student Engagement

Student engagement is has been seen as a valid indicator of academic success [Fredricks and McColskey 2012]. It is defined as the interaction between the time, effort, and other relevant resources invested by both students intended to optimize the student experience and enhance the learning outcomes [Trowler 2010, Fredricks and McColskey 2012].

There are three components of student engagement. Behavioral involvement is evidenced by participation in academic, social, or extracurricular activities. Emotional engagement encompasses reactions to relationships with teachers, peers, and institutions. Also, cognitive engagement represents the effort required to understand complex ideas and develop difficult skills [Fredricks and McColskey 2012].

There are some examples of methods to measure engagement. Self-report, interview, and teacher rating are the most common because they are often the most practical and easy to administer in classroom settings. Self- report measures and interviews depend on the answers of students, while the others of the teacher's observation.

2.2. Learning Management System

Learning management systems (LMS) is an Internet-based platform which holds an essential position in higher education. It integrates teaching, learning, and administrative

tools for students and teachers. It includes learning platforms, content management systems, and virtual learning environments [Coates 2007].

An LMS intends to enable administrators and mentors to manage the learning process. For this reason, its tools provide online access to course content like lecture notes, readings and quizzes; tools for communication and collaboration; and administrative tools [Coates 2007].

Because of the advantages of online environments, LMS allows students to take courses anywhere and anytime. So, it is widely applied in a face to face class and for the distance course [Dixson 2015].

3. Related Works

There are many studies on student engagement in online learning environments. In general, their focus is on user interaction through applied learning and high-quality course material design. We list in this section researches which give particular attention to student engagement in computing education as follows. Rahila [Umer et al. 2018] aims to predict students who are at risk of failing the course. The paper presents an analysis of assignments related information and engagement data. In that study, the engagement was measured by actions data like viewing course modules, reading forum posts, and submission of assignments. Results show that assignment scores are more discriminative than engagement data.

Gray [Gray and DiLoreto 2016] examines the factors that impact student learning outcomes and student satisfaction in asynchronous online learning courses. The study concludes that learner interaction did not have a significant impact on student satisfaction. On the contrary, both course structure and instructor presence had a substantial direct effect.

Dixson [Dixson 2015] tried to discover what activities might be expected to lead to more highly engaged students. A disappointing result was found: there is no significant difference in student engagement levels between those reporting active vs. passive activities. However, instructors should create opportunities for students to interact with each other not just with the content, and to require they do so.

The main difference among these papers and our research is the indicators used to measure the student engagement [Umer et al. 2018, Gray and DiLoreto 2016, Dixson 2015]. In our research, we suggest the usage of the duration of assignments and pace learning as an indicator of engagement, while these papers only use frequency of interaction.

4. Method and Materials

This research aims to profile the novice programmer's behavior and understand how students spend their time in learning activities. To profile the student behavior can help teachers to know who can be on the right learning track and those who can need additional help.

Our study was followed for these research questions:

- **RQ1:** How many time students spend on learning activities?

- **RQ2:** How do students progress in the course?
- **RQ3:** Is there any indicator of the online engagement related to student performance?

RQ1 aims to provide information about the participation of students in programming assignments during the course period. To achieve this objective, we used metrics related to the participation in learning activities in and out-class-time and the participation in extracurricular activities (simulators and marathons).

RQ2 aims to understand how students prograde in the learning of course content through the metric of **learning pace**. Finally, with RQ3, we intend to find a metric as a good indicator of student performance. For this reason, the data were divided into two groups based on students final performance on the programming course. Group A was formed by students those who had a final grade lower than 7.0 and group B who had a degree a greater than or equal. The student data were only included in the analysis if he/she did not give up before the end of the course.

A research ethics committee authorized the development of this study with CAAE number CAAE: 69427916.4.0000.5182.

To answer the abovementioned research questions, we analyzed records from the online LMS used in "Programação 1" and "Laboratório de Programação 1" of the undergraduate course Computer Science at the Federal University of Campina Grande.

The program content, LMS, teachers, and students of both classes are the same. The difference between them is the place where classes occur. "Programação 1" is in a traditional classroom, while "Laboratório de Programação 1" occurs at informatics lab.

The program is organized in small units content:

- Units 1 and 2 include basic programming elements;
- Units 3, 4, and 5 address conditional statements and loops;
- Manipulation of functions is seen in unit 6;
- data structures in units 7, 8, 9 and 10

The teaching styles used in both courses are based on an active method which includes self-paced, mastery learning, and flipped classroom. With self-paced and mastery learning, instructors break the whole class into a sequence of smaller learning units, each covering about two weeks' worth of material. Students stay in the current unit while hasn't mastery in the content. To advance to the next unit, learners have to demonstrate mastery standard in the assessment.

With the flipped classroom, the content delivery to learners occurs outside and in-class time. As homework, learners are told to study instructional materials, video, and books. In both at home and practice class, they should make assignments and answer quizzes. There are still facultative activities to emulate the assessment environment like a programming marathon and fake test.

The evaluation occurs weekly and has two questions for each content unit. Learners are approved if submits the correct answer to two exercises from the current unit. They can correctly answer one question in test 1 and another in test 2, and even so, be approved in the unit. When an individual is approved, he/she progress to the next unit.

4.1. TST

TST is a Learning Management System developed by a Computing Educational Group from the Federal University of Campina Grande. It has been used to support the teaching process of the Programming Course since 2010. In this online system, teachers can create, update, and remove programming exercises and exams, besides monitor class' progress. Furthermore, TST allows learners to read and solve activities, submit source code, answer a quiz, and track their progress, among others.

TST presents a database of programming exercises grouped by the content unit, which helps the delivery process. Teachers are responsible for planning exercises delivery based on student' progress. They are also responsible for managing the time limit for students to submit their answers according to the learning activity. Exams, simulators, and marathons have a time out to submission, opposite to other modes of activities (quiz, programming assignments).

Students can interact with TST in many ways. They can log in on the system, request new learning activities, submit answers, check if the solution is correct, close activity, among others.

We are mainly interested in a subset of data stored by TST, which includes information about the events of the request and close assignment. An event represents when a user read (open) or close a learning activity.

The trace record used in this study is similar to a list of events. For each one, we extracted the following data from the track record:

- Student id;
- Date and time;
- Learning Activity: content unit, status (opened or closed), mode (exercise, simulator or exam);

5. Indicators of Online Student Engagement

In this study, we believe that student is studying while she/he is interacting with LMS. So, we considered the period and how many times learner is contacting TST as a good indicator of engagement.

We proposed some metrics to understand the student's behavior in an online environment. We use the term **study session** to represent a period in which a student interacts with TST. In our case, the interaction occurs when a user opens or close learning activity.

We divided study time into two moments: *thinking-time* and *break-time*. In our study, the first moment represents the time in which student is focused on to study, thinking, or solving an exercise. While *break-time* is when the student is not engaged in the study doing activities outside of TST. In general, a student spends more time in *break-time* than in *thinking-time*, that is *break-time* is longer period than *thinking-time*.

Figure 1(a) presents time intervals between all events. In 1(a)). In Figure 1 (b), we searched for the point which split small ones from long intervals like as in [de Araujo et al. 2013] and highlight where the range of 23 minutes. After 23 minutes, time interval values increase in more significant proportions.

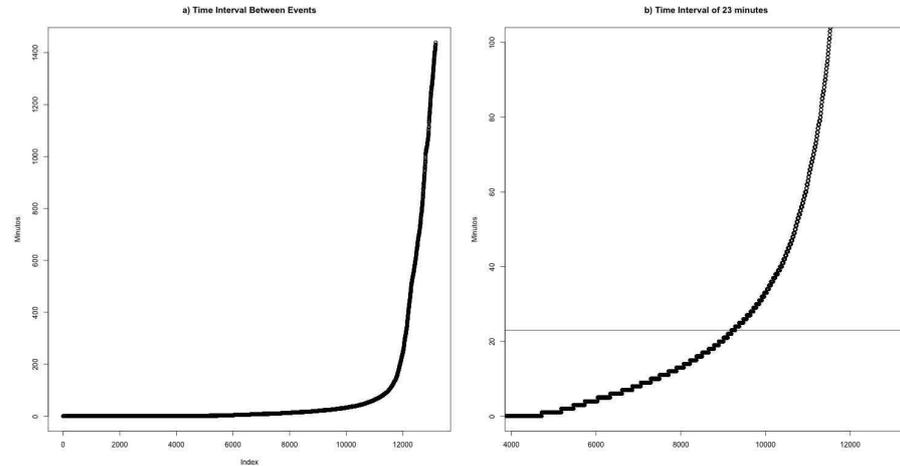


Figure 1. Time Interval between Events

Based on the intervals, we proposed a metric called study session, which includes all events that occurred for a period of fewer than 23 minutes. For example, Figure 2 shows examples of five events on 10/10/2017. At ten o'clock, a student reads an assignment (E1), and solve it after twenty-two minutes (E2). An hour later, she/he reads three activities resulting in events E4, E5, and E6. The time interval between E1 and E2 is less than 23, so both are in Session 1. Unlike between E2 and E3, the range is higher than 23. In this example, we recorded two events in Session 1 and three events in Session 2.



Figure 2. Study Session

We also proposed an indicator named **learning pace**. It suggests how learner progress in the stage of the course, and can help teachers to know who can be on the right learning track and those who can need additional help.

We consider the relationship between the evaluation result and the total of exams as ongoing progress. For example, if in the first test, a learner has been solved two questions of unit 1 and two questions of unit 2, his learning pace is **2** units per test.

The development of homework and participation of extra-class activities are good indicators of student engagement. We registered assignments made out of class time as homework. And the participation in optional activities, like programming marathon and fake test, as indicators of extra-class activities.

6. Results and Discussion

During the semester 2017.2, there were 100 students enrolled in theory and practice programming courses. However, about 16 of them drop out or failed the course. So, we

analyzed the data of the remaining 84 students who solved 450 questions during 151 days and generated a track record with 14320 events.

6.1. RQ1: How many time students spend on learning activities?

The study sessions of learners varied between 4h51m and 44h29m, with an average of 18h19m. This shows that there is a difference of more than 900 % between the two extremes, which may mean that some students study hard, while others have studied very little. During the sessions, a total of events ranging from 143 to 718 were recorded, with a mean of 341.4.

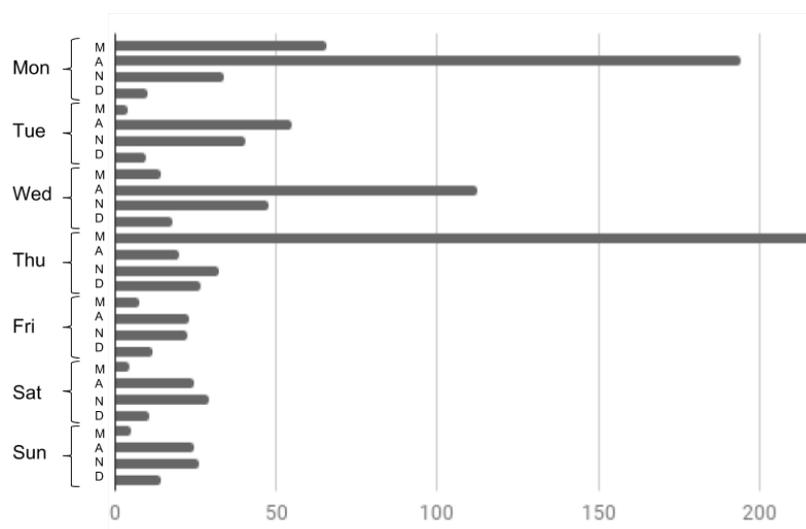


Figure 3. Study Routine

In analyzed semester, classes have occurred on Monday, Wednesday, and Thursday in the morning. And we divided days in four shift: Dawn, Morning, Afternoon and Night. Figure 3 presents the study routine and how it occurs over the week. It shows a study peak in exam time on Thursday in the morning. There are other study peaks on Monday in the afternoon and Wednesday in the afternoon. It also suggests that students were dedicated to homework. The first case occurs after a theoretical lesson, while the second occurs on the eve of the assessment. So, we realized that students study more when they are next to class days, and the majority of assignments is made as homework.

The total number of completed questions ranged from 49 to 334, with an average of 147 activities. The number of assignments made outside of class time ranged from 43 to 321, with an average of 140. Comparing the data, we realized that about 97.5 % of the questions are resolved outside of class time. The number of activities solved during the simulation ranged from 0 to 41, with a mean of 21 questions.

6.2. RQ2: How do students progress in the course?

On average, class achieves 1.6 units per test as learning pace. About 54 % of the students had 0.5 unit per mini-test or one question per exam while 2.3 % of the students did five

exercises per test.

On average, class achieves 1.6 units per test as learning pace. About 54 % of the students had 0.5 unit per mini-test or one question per exam while 2.3 % of the students did five exercises per test.

Intending to profile student's behavior, we grouped students into two groups based on students final performance on the programming course. Group A was formed by students those who had a final grade lower than 7.0 and group B who had a degree higher than or equal.

Figure 4 presents the difference among values of learning pace from group A and B. The lowest learning rate of group A is 0.5, and the highest is 1.42. Already in group B, the value ranges from 1.08 to 5.

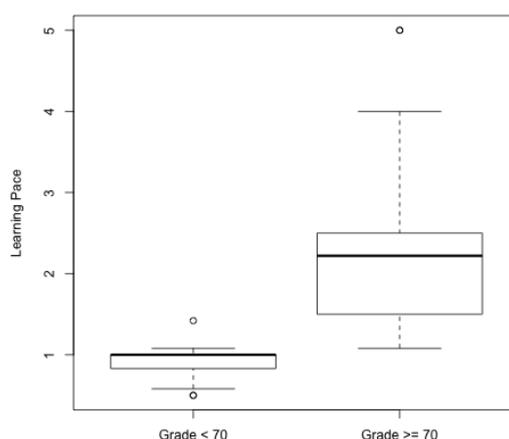


Figure 4. Boxplot graph of Learning Pace: Group A x Group B

Regarding the duration of sessions, group A presented between 291 and 1721, with an average of 833.4. Group B, on the other hand, had meetings between 456 and 2669, and on average of 1299. Concerning the number of events during the sessions, group A is lower than in group B.

Concerning participation in optional learning activities, group A did between 4 and 29 questions, with an average of 19.08. Group B was between 0 and 41, with a mean of 23.5. Using the test t and with a *p-value* = 0.00171, we conclude that group B made more questions in simulations.

With values of learning pace, study session, and participation in optional activities, we can profile a student's behavior of group A. They study during a short time, made fewer assignments, and have less participation in extra-class activities.

Table 1. Correlation between Indicators of Student Engagement

Metrics	Assign. Made	Sim.	at Home	Pace	Session	Event
Sim.	0.34					
at Home	0.93*	0.32				
Pace	0.25	0.25	0.27			
Session	0.80*	0.31	0.79	0.31		
Event	0.91*	0.35	0.88*	0.20	0.76	
Perform.	0.27	0.25	0.28	0.82	0.33	0.22

Table 2. Correlation With Student Performance

Cor.	Act. made	Sim.	Study Sessions	Events	Pace	Out. class
kendall (τ)	0.2736	0.2519	0.332	0.2221	0.8238	0.2867
spearman (ρ)	0.3862	0.3561	0.4727	0.3099	0.9415	0.4097

6.3. RQ3: Is there any indicator of the online engagement related to student performance?

To answer RQ3, we test the normality of metrics using *Shapiro-Wilk* method. We note that the data does not follow a normal distribution because all values of *p-value* were less than 0.05. For this reason, we chose the *Kendall* method to calculate the τ value and estimate the correlation among these metrics.

Table 1 presents the τ value as well as the correlation between the metrics. We highlight the *strong* correlation between the *learning pace* and the *student performance*. We marked with an asterisk (*) the correlation among assignments made and duration of sessions, the number of events and assignments made, and quantity of assignments made at home and quantity of events. All of them have a strong correlation. The strongest correlation is registered among assignments made and homework revealing how much more the student engages in the home study, the more tasks he or she does. The relationship among homework vs. number of events ($\tau = 0.88$) reinforces this conclusion. The correlations registered among the assignments made and the number of events ($\tau = 0.91$) is explained by to make an assignment, the system generates at least three events in the execution trail (read, submit and close). The correlation among the session duration vs. assignments made ($\tau = 0.80$) can indicate how much more the student make assignments, the more time he or she maintain interacting with TST.

Table 2 shows values resulted of the correlation between indicators and student performance. We chose *Kendall* and *Spearman* methods to calculate the τ and ρ values and estimate the correlation among these metrics. We highlighted in grey τ and ρ values of learning pace describing correlation with student performance.

With the analysis of these values, we can profile a students' behavior of high academic performance. They prograde faster, spend more time on homework, make more

assignments, and involve in extra class activities.

7. Conclusion

In this study, we aimed to profile an online student engagement in an introductory programming course. As well as the general feeling of teachers, we realized that students study more on the eve of the evaluation. However, there is an increase in study time in the shift after the theoretical lesson. We corroborated that learners who prograde faster, spend more time on homework, make more assignments, and involve in extra class activities tends to have good academic performance.

The main weakness of this study was the assumption that the student is studying if she/he is interacting with LMS.

References

- Coates, H. (2007). A model of online and general campus-based student engagement. *Assessment & Evaluation in Higher Education*, 32(2):121–141.
- de Araujo, E. C., Gaudencio, M., Menezes, A., Ferreira, I., Ribeiro, I., Fagner, A., Ponciano, L., Morais, F., Guerrero, D. S., and Figueiredo, J. A. (2013). O papel do hábito de estudo no desempenho do aluno de programação.
- Dixson, M. D. (2015). Measuring student engagement in the online course: The online student engagement scale (ose). *Online Learning*, 19(4):n4.
- Farrington, C. A., Roderick, M., Allensworth, E., Nagaoka, J., Keyes, T. S., Johnson, D. W., and Beechum, N. O. (2012). *Teaching Adolescents to Become Learners: The Role of Noncognitive Factors in Shaping School Performance—A Critical Literature Review*. ERIC.
- Fredricks, J. A. and McColskey, W. (2012). The measurement of student engagement: A comparative analysis of various methods and student self-report instruments. In *Handbook of research on student engagement*, pages 763–782. Springer.
- Gray, J. A. and DiLoreto, M. (2016). The effects of student engagement, student satisfaction, and perceived learning in online learning environments. *International Journal of Educational Leadership Preparation*, 11(1):n1.
- Harris, L. R. (2008). A phenomenographic investigation of teacher conceptions of student engagement in learning. *The Australian Educational Researcher*, 35(1):57–79.
- Trowler, V. (2010). Student engagement literature review. *The higher education academy*, 11(1):1–15.
- Umer, R., Susnjak, T., Mathrani, A., and Suriadi, S. (2018). A learning analytics approach: Using online weekly student engagement data to make predictions on student performance. In *2018 International Conference on Computing, Electronic and Electrical Engineering (ICE Cube)*, pages 1–5. IEEE.

Apêndice C

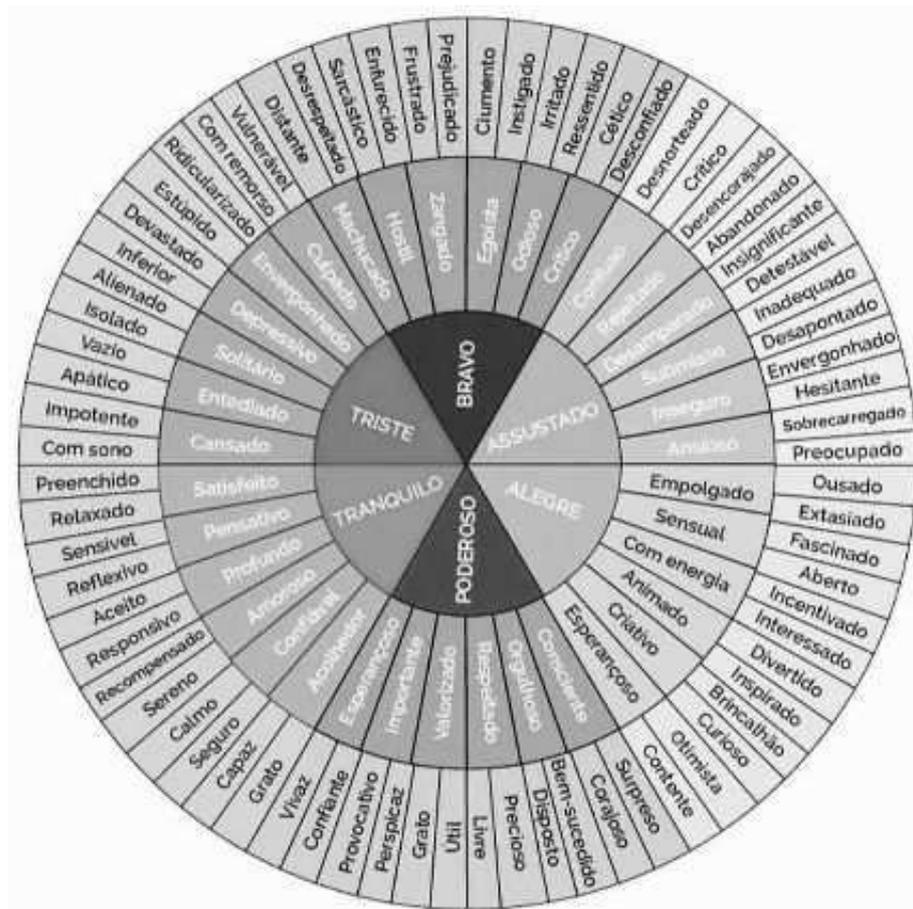
Questionário do Grupo Controle

Apêndice D

Questionário do Grupo Experimental

Simulado

1. Para aprender a programar, você deve realizar muitas atividades práticas, tais como: resolver problemas de programação; analisar uma solução para identificar bugs e sugerir melhorias; e criar seus próprios exercícios para estimular o raciocínio e pensamento crítico.
Use o computador no **ambiente de prova** para fazer a atividade que está no endereço **<http://bit.do/exercicioE>**
2. **Ao concluir o simulado**, analise a figura a seguir e assinale as 3 emoções que você experimentou com maior intensidade na aula de hoje.



3. O que você achou da prática de hoje? Apresente aspecto(s) positivo(s) e negativo(s).

Apêndice E

Atividades Personalizadas: Grupo Controle

Turma Controle

Conceitos elementares de programação; Condições, alternativas e funções; Laços definidos e Laços indefinidos

Para aprender a programar, você deve realizar muitas atividades práticas, tais como: resolver problemas de programação; analisar uma solução para identificar bugs e sugerir melhorias; e criar seus próprios exercícios para estimular o raciocínio e pensamento crítico.

Nesta lista, transformamos as dúvidas mais comuns dos alunos no início do curso em exercícios com estilos diferentes dos que você está habituado a fazer na disciplina. Assim, você terá outras formas de estudar e praticar programação.

1. Crie um exercício de programação que ajude a praticar o arredondamento de valores float.
2. Implemente um programa para responder o exercício que você criou na questão 1.
3. Divisores: Escreva um programa que leia um número inteiro (N), e mostre os seus divisores, onde $0 < N \leq 100$.
4. Mostre o que há de errado na solução para o exercício "Divisores" apresentada a seguir.

```
#coding: utf-8
n = int(raw_input())
divisores =
    range(1,100,1)
for e in divisores:
if e >= n:
    break
elif n % e == 0:
    print e
```

5. Apresente exemplo(s) de testes (com entrada e saída válidos) que mostre(m) o(s) erro(s) do código da questão 4.
6. Crie um exercício de programação que ajude a praticar o cálculo de porcentagem.
7. Implemente um programa para responder o exercício que você criou na questão 6.
8. Média: Crie um programa que recebe números até que o caractere "#" seja digitado. Seu programa deve imprimir quantos números foram lidos e a média deles com 3 casas decimais.
9. Mostre o que há de errado na proposta de solução para o exercício "Média" apresentada a seguir.

```

# coding: utf-8
contador = 1
soma = 0
while True:
    num = raw_input()
    if num == "#":
        break
    else:
        soma += float(num)
        contador += 1
media = soma / contador
print "A média dos números lidos é: %f" % media
print "%i números foram lidos." % contador

```

10. Apresente exemplo(s) de testes (com entrada e saída válidos) que mostre(m) o(s) erro(s) do código da questão 9.
11. Crie um exercício de programação que ajude a praticar o uso das funções do tipo String.
12. Implemente um programa para responder o exercício que você criou na questão 11.
13. Rodízio de Veículos: Em cidades com tráfego intenso é comum existir sistemas de rodízio de veículos. Esse sistema consiste, basicamente, na restrição de circulação de veículos nos horários de pico de acordo com o final da placa e dia da semana. Seu programa deve retornar o número de veículos restritos e as placas que terão restrição de circulação no dia da semana requerido.

O programa deve receber como entrada o dia da semana, um número $N > 1$ de placas de carro e em seguida as N placas no formato "XXX-YYYY", sendo "X" letra e "Y" número.

O programa deve imprimir o número de veículos restritos e as placas que identificam esses veículos, conforme o exemplo de execução.

****Importante:**** Caso não haja placas restritas ou se o dia da semana não for um dia útil, o usuário deve receber a mensagem de que "Não há restrições!".*

Dia	Seg	Ter	Qua	Qui	Sex
Final da placa	1 ou 2	3 ou 4	5 ou 6	7 ou 8	9 ou 0

Exemplo de Execução

```
python rodizio.py
terça
4
MNO-2343
GDI-7964
KJH-2281
LPO-5658
Número de veículos
restritos: 2
Placa(s): MNO-2343,
GDI-7964.
```

14. Mostre o que há de errado na proposta de solução para o exercício “Rodízio de Veículos” apresentada a seguir.

```
#coding: utf-8
dia = raw_input()
diasSemana = ["segunda" ,"terça" ,"quarta" ,"quinta","sexta"]
if(dia in diasSemana):
    quantidade = int(raw_input())
    restritas = []
    contador = 0
    while(contador <= quantidade):
        placa = raw_input()
        contador += 1
        if(dia == "segunda" and placa[7] in "12"):
            restritas.append(placa)
        elif(dia == "terça" and placa[7] in "34"):
            restritas.append(placa)
        elif(dia == "quarta" and placa[7] in "56"):
            restritas.append(placa)
        elif(dia == "quinta" and placa[7] in "78"):
            restritas.append(placa)
        elif(dia == "sexta" and placa[7] in "90"):
            restritas.append(placa)
    if(len(restritas) >= 0):
        print "Número de veiculos restritos: %d" % len(restritas)
        print "Placa(s):"
        for i in restritas:
            print i
    else:
        print "Não há restrições!"
else:
```

```
print "Não há restrições!"
```

15. Apresente exemplo(s) de testes (com entrada e saída válidos) que mostre(m) o(s) erro(s) do código da questão 14.

Apêndice F

Atividades Personalizadas: Grupo Experimental

Turma Experimental

Conceitos elementares de programação; Condições, alternativas e funções; Laços definidos e Laços indefinidos

Para aprender a programar, você deve realizar muitas atividades práticas, tais como: resolver problemas de programação; analisar uma solução para identificar *bugs* e sugerir melhorias; e criar seus próprios exercícios para estimular o raciocínio e pensamento crítico.

Nesta lista, transformamos as dúvidas mais comuns dos alunos no início do curso em exercícios com estilos diferentes dos que você está habituado a fazer na disciplina. Assim, você terá outras formas de estudar, praticar programação e **ajudará quem está com dificuldade**.

1. Em uma postagem no Slack, <estudante1> (estudante1@email) perguntou "Alguém lembra como arredondo para cima? Tipo 297.10 para 298.00?". Crie um exercício de programação que ajude Cristiano a praticar o arredondamento de valores float.
2. Implemente um programa para responder o exercício que você criou na questão 1.
3. Responda a questão "Divisores" criada pelo aluno <estudante2> (estudante2@email): Escreva um programa que leia um número inteiro(N), e mostre os seus divisores, onde $0 < N \leq 100$.
4. A solução desenvolvida pelo aluno <estudante3> (estudante3@email) para a questão "Divisores" exibida a seguir está errada. Mostre o que há de errado no código que enviaremos sua dica de melhoria para Gabriel.

```
#coding: utf-8
#Author: estudante3
n = int(raw_input())
divisores =
range(1,100,1)
for e in divisores:
    if e >= n:
        break
    elif n % e == 0:
        print e
```

5. Apresente exemplo(s) de testes (com entrada e saída válidos) que pode(m) ajudar <estudante4> (estudante4@email) a encontrar erro(s) no código da questão 3.
6. Em uma postagem no Slack, <estudante5> (estudante5@email) disse "tô me atrapalhando em calcular porcentagem, às vezes dá um número muito acima e às vezes dá 0". Crie um exercício de programação que ajude Paloma a praticar o cálculo de porcentagem.
7. Implemente um programa para responder o exercício que você criou na questão 6.
8. Responda a questão "Média" criada pelo aluno <estudante6> (estudante6@email): Crie um programa que recebe números até que o caractere "#" seja digitado. Seu programa deve imprimir quantos números foram lidos e a média deles com 3 casas decimais.
9. A solução de <estudante7> (estudante7@email) exibida a seguir para a questão "Média" está errada. Mostre o que há de errado no código que enviaremos sua dica de melhoria para Felipe.

```
# coding: utf-8
# author: estudante7
contador = 1
soma = 0
while True:
    num = raw_input()
    if num == "#":
        break
    else:
        soma += float(num)
        contador += 1
media = soma / contador
print "A média dos números lidos é: %f" % media
print "%i números foram lidos." % contador
```

10. Apresente exemplo(s) de testes (com entrada e saída válidos) que pode(m) ajudar <estudante8> (estudante8@email) a encontrar erro(s) no código da questão 9.
11. Em uma postagem do slack <estudante9> (estudante9@email) pediu "dicas de como criar uma função que retorne a quantidade de caracteres digitados". Crie um exercício de programação que ajude <estudante9> a praticar o uso das funções do tipo String.
12. Implemente um programa para responder o exercício que você criou na questão 11.

13. Responda a questão “Rodízio de veículos” criada pela aluna <estudante10> (estudante10@email): Em cidades com tráfego intenso é comum existir sistemas de rodízio de veículos. Esse sistema consiste, basicamente, na restrição de circulação de veículos nos horários de pico de acordo com o final da placa e dia da semana. Seu programa deve retornar o número de veículos restritos e as placas que terão restrição de circulação no dia da semana requerido.

O programa deve receber como entrada o dia da semana, um número $N > 1$ de placas de carro e em seguida as N placas no formato “XXX-YYYY”, sendo “X” letra e “Y” número.

O programa deve imprimir o número de veículos restritos e as placas que identificam esses veículos, conforme o exemplo de execução.

Importante: Caso não haja placas restritas ou se o dia da semana não for um dia útil, o usuário deve receber a mensagem de que “Não há restrições!”.

Dia	Seg	Ter	Qua	Qui	Sex
Final da placa	1 ou 2	3 ou 4	5 ou 6	7 ou 8	9 ou 0

Exemplo de Execução

```
python rodizio.py
terça
4
MNO-2343
GDI-7964
KJH-2281
LPO-5658
Número de veículos restritos:
2
Placa(s): MNO-2343, GDI-7964.
```

14. A solução de <estudante11> (estudante11@email) exibida a seguir está errada. Mostre o que há de errado no código que enviaremos sua dica de melhoria para Ana.

```
#coding: utf-8
# Author: estudante11
dia = raw_input()
diasSemana = ["segunda" ,"terça" ,"quarta" ,"quinta","sexta"]
if(dia in diasSemana):
    quantidade = int(raw_input())
    restritas = []
    contador = 0
    while(contador <= quantidade):
        placa = raw_input()
        contador += 1
        if(dia == "segunda" and placa[7] in "12"):
            restritas.append(placa)
        elif(dia == "terça" and placa[7] in "34"):
            restritas.append(placa)
        elif(dia == "quarta" and placa[7] in "56"):
            restritas.append(placa)
        elif(dia == "quinta" and placa[7] in "78"):
            restritas.append(placa)
        elif(dia == "sexta" and placa[7] in "90"):
            restritas.append(placa)
    if(len(restritas) >= 0):
        print "Número de veículos restritos: %d" % len(restritas)
        print "Placa(s):"
        for i in restritas:
            print i
    else:
        print "Não há restrições!"
else:
    print "Não há restrições!"
```

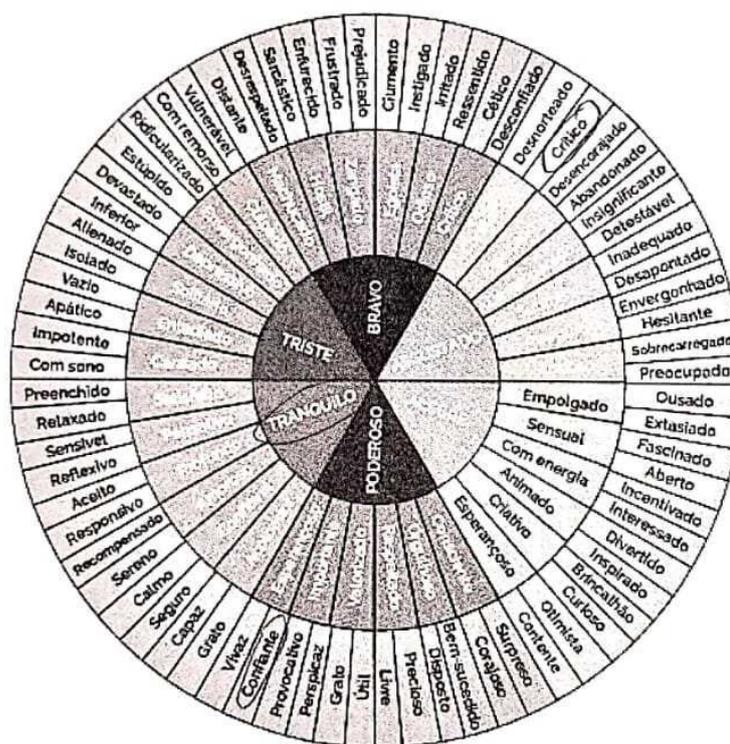
15. Apresente exemplo(s) de testes (com entrada e saída válidos) que pode(m) ajudar <estudante12> (estudante12@email) a encontrar erro(s) em seu código.

Apêndice G

Feedback dos Estudantes

Simulado: 20/05/2019

- Para aprender a programar, você deve realizar muitas atividades práticas, tais como: resolver problemas de programação; analisar uma solução para identificar bugs e sugerir melhorias; e criar seus próprios exercícios para estimular o raciocínio e pensamento crítico.
Use o computador no ambiente de prova para fazer a atividade que está no endereço <http://bit.do/exercicioE>
- Ao concluir o simulado, analise a figura a seguir e assinale as 3 emoções que você experimentou com maior intensidade na aula de hoje.



- O que você achou da prática de hoje? Apresente aspecto(s) positivo(s) e negativo(s).

achei a atividade interessante, com um bom de construção de conhecimentos para ajudar os alunos e a disciplina o evoluiram. O aspecto negativo foi o tamanho da atividade para um simulado e a exposição dos mails de colegas.



Scanned with
CamScanner

Figura G.3: Opinião Sobre Atividades Personalizadas (III)

Apêndice H

Learning Styles in Programming

Education: a Systematic Mapping Study

Learning Styles in Programming Education: A Systematic Mapping Study

Mirna Carelli Oliveira Maia
Federal Institute of Paraiba
Paraiba, Brazil
Email: mirna.maia@ifpb.edu.br

Dalton Serey
Federal University of Campina Grande
Paraiba, Brazil
Email: dalton@dsc.ufcg.edu.br

Jorge Figueiredo
Federal University of Campina Grande
Paraiba, Brazil
Email: abrantess@computacao.ufcg.edu.br

Abstract—The interest on effective strategies to improve the learning process increased over the years. The usual practice is to know the characteristics of a student and how those can affect the learning style. The style describes how learners collect, select, interpret, organize and store information. This paper overviews the application of learning styles and how those affect the teaching of introductory programming. We performed a systematic mapping study to improve our knowledge about student characteristics in the teaching of programming. Twenty-one papers were relevant for our research. The main findings are: i) the Soloman-Felder Index is the most used model of learning style; ii) studies focus on the effects of learning styles on student performance; iii) instructors apply active approaches to motivate a different kind of learners. We conclude that learning style does affect a student's ability to learn. For this reason, teachers should incorporate those styles into their pedagogical material to cater to the majority of students needs.

I. INTRODUCTION

The interest on effective strategies to improve the learning process increased over the years. The usual practice is to know the characteristics of a student and how those can affect the teaching and learning process. The learning styles describe how learners collect, select, interpret, organize and store information[1]. They serve as relatively stable indicators of how students respond to the learning environment[2].

In introductory programming courses (IPC), there is a continuous effort to reduce the freshmen students difficulties as well as to find ways to decrease the failure rates. Understand how learning styles can contribute to IPC is the focus of this paper. We performed a study to improve our knowledge about student characteristics, difficulties, and abilities in the programming education. This study describes a systematic mapping study to overview the use of learning styles in IPC in the last ten years. The following research questions guided our study:

- **RQ1:** What are the models of learning styles used in the introductory programming education?
- **RQ2:** How are used the learning styles in the introductory programming courses?

These research questions guided the steps of our study. We searched for relevant papers at six scientific databases. The searching process retrieved a total of 4321 documents. After, we analyzed and filtered twenty-one as relevant papers for our research.

Our main findings are the following: i) the Soloman-Felder Index is the most used model of learning style; ii) studies focus on the effects of learning styles on student performance; iii) instructors applied active approaches to cater a different kind of learners. The results point out that learning style affects the ability to learn, however, it can not be considered a success or a failure indicator. We concluded that teachers should incorporate these styles into their pedagogical material to cater to the majority of students needs.

The remainder of this paper is organized as follows: Section II presents the related works. Section III details the employed research methodology, followed by the presentation and discussion of the results in Section IV. The validity threats are discussed in Section V. Finally, Section VI presents the conclusions and future works.

II. RELATED WORK

In this section, we summarize three relevant literature reviews we found on learning styles.

- Vicki [3] investigated the implications of learning-styles-based teaching and concluded that learning style can not be considered a successful indicator. She suggested that teachers evaluate learners' success by their performance and attitudes, not only by their characteristics;
- Coffield et al. [4] present an evaluation of the major models of learning style. They aim to understand the models' merits and deficiencies better. So, they investigated if there is empirical evidence that these models have an impact on the teaching and learning processes. They concluded that knowing how people learn can be used as a tool to enhance the learning. In that study, the Dunn model was mentioned as the most used learning style. Although, in our study, this model was not cited in any paper;
- Workman [5] reported the beneficial and controversial facts in accommodating the teaching method for the style of each learner. According to him, there are quantitative facts to disagree in adapting the teaching process to the learning style;

Literature reviews mentioned above are not focused on a particular area. We did also not find any literature review that related learning style and programming education. For this

reason, we aimed to explore the relationships among both of these topics.

III. RESEARCH METHOD

In this section, we detail the steps performed in systematic mapping study. In this method, researchers provide a classification scheme and structure a field of interest [6]. Petersen suggested a process which has five main steps: (i) definition of research questions; (ii) conducting the search; (iii) screening of papers; (iv) key-wording of abstracts and (v) the data extraction. These steps are detailed below.

A. Definition of Research Questions

We defined two research questions to achieve the goals of this study.

- **RQ1:** What are the models of learning styles used in the introductory programming education?
- **RQ2:** How are the learning styles used in the introductory programming courses?

RQ1 aims to describe the most used models and how to measure the learning styles of novice programmers. The answer of RQ2 reports how they can influence the teaching in the programming education context.

B. Conduct Search

In this section, we set up the strategy to design a suitable search string. We were guided by the following suggestions of Kitchenham et al. [6] with the purpose to decrease bias:

- Identifying of relevant concepts in research questions;
- Finding out an alternative spelling, abbreviations, and synonyms for these keywords;
- Checking subject headings in relevant journals and conference proceedings;
- Using boolean operators OR to connect the synonyms;
- Using boolean operators AND to connect the other terms;

We selected the keywords *learning style*, *introduction* and *programming* to perform some exploratory searches. Next, we find out synonyms for these keywords. Finally, we established the following final string to be used in the automatic search:

(learning style AND programming) AND (beginner OR novice OR introductory OR freshmen)

We chose scientific databases that provide tool or mechanism to perform automatic searching and have relevant sources in the computing education area. We selected the following scientific databases: ACM Digital Library[7], IEEEExplore[8], Scopus[9], EdITLib[10], Science Direct[11] and Springer[12]. After, we used the automatic searching tool to conduct the search. Table I shows the number of papers retrieved by each database.

This step was done from November 15th, 2016 to February 1st, 2017 and a total of 4321 documents were found.

Table I
AMOUNT OF PAPERS RETRIEVED BY SEARCH

Database	Retrieved	Relevant Papers
ACM Digital Library	17	10
IEEEExplore	13	0
Scopus	100	7
EdITLib Digital Library	341	0
Science Direct	436	4
Springer	1366	0
Total	4321	21

C. Screening of Papers

The definition of inclusion and exclusion criteria are a typical strategy to perform these screening. In our research, six criteria were created, three inclusion criteria and three exclusion criteria. They are listed below:

Inclusion criteria:

- **F1:** The title, abstract or keywords should explicitly mention the terms *learning style* **and** *programming*;
- **F2:** Study should be related to Computer Science;
- **F3:** The paper should be published not before 2006;

Exclusion criteria:

- **F4:** Work in progress;
- **F5:** Papers that are not reported in workshops, conferences or journals;
- **F6:** Documents whose full text is not available for free;

The documents previously found were used as input for this step. We applied criteria mentioned above to filter the relevant studies. We kept only one version of the document. So, duplicated papers were removed if indexed by more than one database. As a result, a total of 21 were considered as relevant papers.

Table I also shows the result of this screening process by each scientific database. There is no relevant papers from the databases IEEEExplore, EdITLib and Springer. IEEEExplore retrieved duplicate documents, or they did not mention the words in F1. Some of the documents in EdITLib are not available for free download (F6) whereas the majority of Springer papers is not related to computer science (F2). The database that had the best retrieval rate was ACM with 58.8% of relevant papers.

D. Keywording

According to Petersen et al. [6], keywording reduces the time needed in developing the classification scheme. When a set of keywords has been chosen, studies can be clustered,

Table II
RESEARCH FACET TYPES

Category	Description
Experience Paper	It explains what and how something has been done in practice. It has to be the personal experience of the author and usually includes a lesson-learned section
Evaluation Research	Techniques are implemented in practice and an evaluation of the technique is conducted
Opinion Paper	It express the personal opinion of somebody about a certain technique
Solution Proposal	When a solution for a problem is proposed, the potential benefits and the applicability is shown by a small example or a good line of argumentation
Philosophical Papers	sketch a new way of looking at existing things by structuring the field in the form of a taxonomy or conceptual framework
Validation Research	Techniques have not yet been implemented in practice or the work was done in the lab

used to form the categories for the map and can help answer the research questions. In this step, we identified keywords that reflect the concepts and contribution of the paper and combined them to classify and understand the contribution of the research. So, we used the following criteria for the studies categorization:

- **Research Facet Types:** The research facet reflects the research approach used in the papers. This category are proposed by Petersen et al. [6] and are described in Table II;
- **Model of the learning style:** This category represents the model of learning style used to classify students. It can help us to answer RQ1;
- **Tool:** It represents the questionnaire used to measure the learning styles;
- **Teaching method:** This one represents the pedagogical strategies applied in the introductory programming courses.

E. Data Extraction

The data extraction helped us to answer the research questions. We read the filtered documents and, then, we retrieved a lot of information. After that, we stored the title, year of

Table III
YEAR WISE DISTRIBUTION

Year	Studies
2007	[13]
2008	[14]
2009	[15], [16], [17]
2010	[17], [18], [19], [20], [21]
2011	[22], [23]
2012	[24]
2013	[25], [26], [27], [28]
2014	[29]
2015	[30], [31]
2016	[32]

publication, venue (forum or conference proceedings), author's name and location. Each one was assigned to a set of keywords and classified according to categories as mentioned above. In this work, none of them were assigned to more than one category from the same criterion.

IV. RESULTS AND DISCUSSION

This section shows the results produced by the extraction of data from the primary studies according to the process described in Section III. To provide the simplest way to view the results, those were separated into two groups: general information and the research questions. Firstly, we present the overview of relevant studies through the general results. After that, the answers to the research questions are presented.

A. General Results

In this section, we present general information about the relevant studies. We considered as general information the year of publication, the venue where they were published and authors information. The year wise distribution of the relevant studies is presented in Table III. The following patterns were observed in the result: the number of publications varies from 1 to 5 papers; 2010 and 2013 are the most productive years with 5 and 4 publications, respectively; 2009 had 3 studies; 2011 and 2015 with two and the years 2007, 2008, 2012 and 2016 with only one paper each.

The majority of the studies (81% - 17 papers) were published in conference proceedings, while 19% (4 studies) were journal papers[13], [15], [24], [32]. *Conference on Innovation and Technology in Computer Science Education (ITiCSE)* accepted the majority of the publications with three papers.

The countries of the researchers are displayed in Table IV. Results show that Malaysia and Australia are the leading countries with 4 and three studies each one. USA, United Kingdom, and Portugal had two studies and the other countries with one

Table IV
COUNTRY WISE DISTRIBUTION

Author Location	Paper
Malaysia	[18], [24], [25], [26]
Australia	[22], [23], [33]
USA	[16], [32]
United Kingdom	[27], [31]
Portugal	[19], [20]
Argentina	[28]
Sweden	[30]
Finland	[17]
Italy	[14]
Hong Kong	[15]
New Zealand	[29]
Thailand	[21]
United Arab Emirates	[13]

Research Type Facet

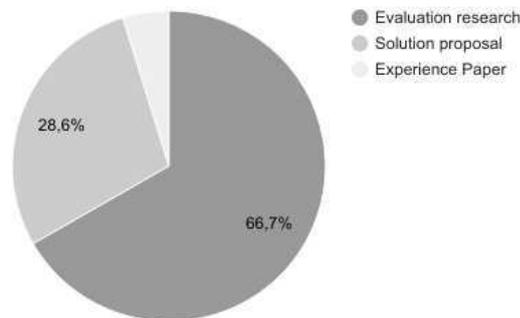


Figure 1. The Research Facet Type of Selected Studies

study. Anabela Jesus Gomes, Antonio José Mendes [19], [20] and Bedour Alshaigy [27], [31] are the most prolific authors with 2 papers each.

Figure 1 shows the papers distribution according to the classification scheme described in Table II. 66.7% of studies were classified as evaluation research, and 28.6% of them were classified as solution proposal. The rest of them (4.7%) was categorized as experience paper. Philosophical papers, opinion paper, and validation research papers were not found in the selected studies.

B. Answers to Research Questions

In this section, we present the results grouped by the answer to research questions.

• RQ1: What are the models of learning styles used in the introductory programming education?

The literature reports 71 types of learning styles. Some of them have been used whereas others have fallen into obscurity. Here, we outline the models of learning styles most used in programming education.

There are 11 papers that used **Soloman-Felder Index** as the model of style [13], [16], [17], [18], [19], [20], [21], [24], [29], [31]. This model categorizes four different dimensions of an individuals learning style: active/reflective, sensing/intuitive, visual/verbal, and sequential/global. *Active* learners prefer an environment that enables them to learn to do something active like applying or discussing it with their peers. *Reflective* learners prefer an environment that enables them to think about something quietly first. *Sensing* learners prefer learning facts and concepts, *intuitive* learners prefer learning by discovering possibilities, applications, and relationships. *Visual* learners prefer learning from material they can see, they prefer the use of pictures, diagrams, flowcharts, and demonstrations. *Verbal* learners prefer words, either spoken or written. *Sequential* learners follow material in a step-by-step sequence and gain understanding in linear steps. *Global* learners tend to learn in large jumps [34].

VARK Modalities is an acronym for Visual (V), Aural (A), Read/Write (R), and Kinaesthetic (K). This model defines four learning styles: *visual* is a learner that learns best by seeing, similar to visual Learner previous mentioned; *Aural* is a student that learns best through listening to lectures, and it is similar to verbal learner; *Read/Write* is a student that prefers to learn reading textbooks and then summarizing or writing what they have understood; *Kinaesthetic* students learn most effectively through experience by undertaking experiments, and carrying out case studies, practical sessions, etc. [35], [36]. This model was used in two papers [22], [26].

There were two models mentioned by one paper: **Perkins' model** and **Gregorc Style Delineator** (GSD). Perkins suggests the classification of novice programmers' as stoppers, non-starters, movers, and tinkers. *Stoppers* are students who tend to stop or give up when they can not immediately know the solution to a problem. *Non-starters* are a type of stoppers who makes some progress in exercises but do not submit their programs for assessment. *Movers* try different approaches when faced with a problem. *Tinkers* make changes, but using a random approach. Cardell [23] applied this model to categorize novice programmers.

In the GSD model, there are two mediation abilities with two qualities which result in four styles: concrete sequential, concrete random, abstract sequential and abstract random [37]. *Concrete sequential* learners tend to perceive reality through their physical senses and to think sequentially. *Concrete random* learners like to think intuitively, instinctively, impulsively and independently. *Abstract sequential* learners are logical

Table V
MODELS OF LEARNING STYLES

Model	Studies
Soloman-Felder	[13], [16], [17], [18], [19],
Index of Learning Style	[20], [21], [24], [28], [29]
VARCK	[22], [26]
Modalities	
Perkin's	[23]
Gregorc Style	[15]
Delineator	
No mention	[14], [25], [27], [30], [32] [33]

and analytical individuals who have a preference for mentally stimulating tasks and environments. *Abstract random* learners tend to think in a nonlinear and emotional manner. This model was mentioned in [15].

Learning styles are measured based on the answers of the learners to a questionnaire. In general, they have to choose responses suitable for particular occasions. The style is identified according to the student responses. Each model has a specific inventory. The Index of Learning Styles Questionnaire was formulated by Richard M. Felder and Barbara A. Soloman to measure the styles according to Solomon-Felder Index [38]. This questionnaire has 44 questions whose answers can be submitted, and the results are automatically sent via the Web [39]. The VARK Questionnaire intends to report the learner's experience to measure their learning styles [35]. It is a free questionnaire and available online.

Five papers did not mention which model of learning style was used.

The answer of RQ1 is schematized in Figure 3. It shows the relation between research facet types and the models. As aforementioned, the majority of the papers were categorized as evaluation research. These studies reported an application of a teaching strategy and evaluated those effect on the learning styles. A solution was proposed in six of them, three using the Soloman-Felder Index and others three papers did not mention any model.

• **RQ2: How are used the learning styles in IPC?**

In general, the majority of studies explored if learning style affects the student performance in introductory programming courses. However, there is no consensus about this question. According to [20], there is no correlation and no pattern

Degree Education Level

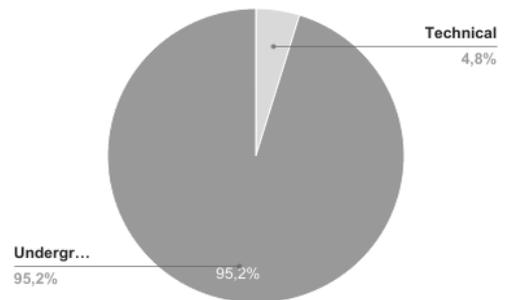


Figure 2. Educational Stage

of the students that have better or worse academic results. Conversely, studies [17], [22] suggest that the reflective students were better in programming than active. Additionally, Raadt [22] also reports that verbal learner outperformed visual learner. Other study compared the profile of programming and mathematics students [16]. We concluded that the learning style can not be used as an instrument to predict student success.

On the teaching strategies, some of the studies reported the beneficial for adopting pair activities. Student improves learning when they verbalize their thoughts to someone. It also happens when they were pairing to a learner with a different style. For this reason, the majority of the studies applied strategies based on an active method. The most used are pair programming, peer discussion, collaborative and cooperative learning. Some researchers [18], [24], [25] reported the use of pair programming and peer discussion as strategies of teaching to encourage students to be more engaged. Moreover, [16] suggested the incorporation of more active, sensing, and visual learning style practices into the teaching process.

Raadt and Simon [22] compared the relation between the learning style of students with the learning style of their instructor. They report that students with profiles that matched those of the instructors tend to perform better than others. Authors also suggested that is unwise to assume that students share the same learning preference as their instructors. Some instructors are teaching according to their preferences. If the learning materials are designed to cover both read/write, and kinaesthetic profiles will suit all students. So, when dealing with it, it should be a good textual material to suit read/write modality and hands-on experience to suit the kinaesthetic.

Finally, Figure 2 shows that about 95 % of the studies target to undergraduate courses. While only one was driven to technical course [28]. Other educational stages were not mentioned in the studies.

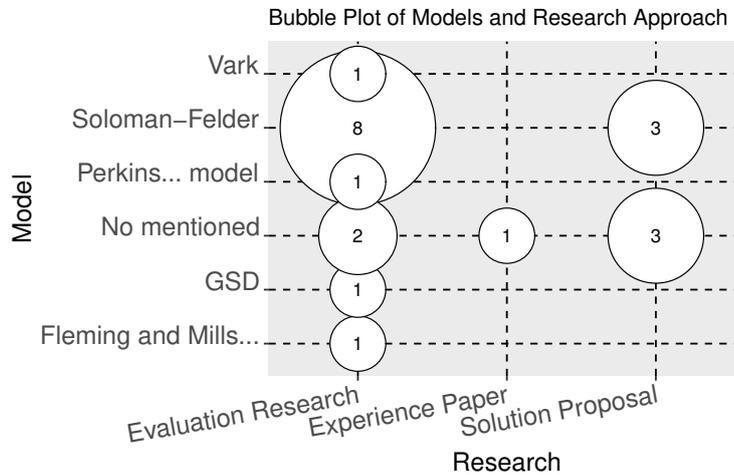


Figure 3. Visualization of a Systematic Map in the Form of a Bubble Plot

V. THREATS

Although the systematic mapping is more accurate than other literature review approaches, it is still possible to enumerate some threats to its validity. This study does not have any statistical assumption what can mean a threat internal validity. The rate of retrieved papers can indicate that the terms used in the search process are suitable. Only 0.48% of the studies extracted were considered as a relevant paper. It happens because the term "learning style" is also applied to other research areas. Moreover, the data collected are limited by the automated searching engine from databases selected in this work. Therefore, the results can not be generalized.

VI. CONCLUSIONS

This paper presents a systematic mapping study to overview the use of learning styles in introductory programming teaching. We summarized papers that were published from 2006 to 2017. For this propose, we defined research questions and search strings to guide the automatic search on six databases. After running the search, 4321 articles were selected. We filtered them based on the inclusion and exclusion criteria resulting in 21 studies. We defined classification criteria to help us answer the research questions. The majority were conferences proceedings papers. Furthermore, Malaysia is the leading country.

The main findings are the following: i) the Soloman-Felder is the most used model; ii) researchers mainly investigate the effects of LS on student performance; iii) and in most cases, instructors apply an active method in order to encourage a variety of learners. The results point out that learning style does affect a student's ability to learn. Nonetheless, it can not be considered as a unique factor of success or a failure.

Our interpretation of the study results has allowed us to identify some possible research opportunities:

- Propose, apply and validate active pedagogy concepts for undergraduate students in IPC;
- Evaluate the effects of active learning on student's motivation and interests in IPC;

ACKNOWLEDGMENT

This work was partially sponsored by the agreement N^o 08200.315131/2016-10 between UFCG and ePol/DPF.

REFERENCES

- [1] J. Vermunt, "Learning styles and guidance of learning processes in higher education," Amsterdam: Lisse Swets & Zeitlinger, 1992.
- [2] D. Gill, "Meeting differing learning styles of non-traditional students in the second language classroom," *Journal of College Teaching & Learning (TLC)*, vol. 2, no. 8, 2011. [Online]. Available: <https://cluteinstitute.com/ojs/index.php/TLC/article/view/1848>
- [3] V. A. Wilson, "Learning How They Learn: A Review of the Literature on Learning Styles." *ERIC*, vol. ED427017, 1998. [Online]. Available: <http://search.epnet.com/login.aspx?direct=true&\#38;db=eric\&\#38;an=ED427017>
- [4] F. Coffield, D. Moseley, E. Hall, and K. Ecclestone, "Learning styles and pedagogy in post-16 learning A systematic and critical review (application/pdf Object)," [www.LSRC.ac.uk](http://www.lsc.ac.uk), 2004. [Online]. Available: <http://www.lsc.ac.uk/files/PDF/1543.pdf>
- [5] G. F. Workman, "Learning styles: Fact or fiction, a literature review of learning styles," 2012.
- [6] K. Petersen, R. Feldt, S. Mujtaba, and M. Mattsson, "Systematic mapping studies in software engineering," in *Proceedings of the 12th International Conference on Evaluation and Assessment in Software Engineering*, ser. EASE'08. Swindon, UK: BCS Learning & Development Ltd., 2008, pp. 68–77. [Online]. Available: <http://dl.acm.org/citation.cfm?id=2227115.2227123>
- [7] "Acm digital library," <http://dl.acm.org/>, december 2015.
- [8] "Ieee xplore digital library," <http://ieeexplore.ieee.org/>, december 2015.
- [9] "Scopus," <http://www.scopus.com/>, december 2015.
- [10] "Editlib digital library," <http://www.editlib.org/>, december 2015.
- [11] "Science direct," <http://www.sciencedirect.com/>, december 2015.
- [12] "Springer," <http://www.springer.com.br/>, december 2015.

- [13] I. A. Zualkernan, "Using soloman-felder learning style index to evaluate pedagogical resources for introductory programming classes," in *29th International Conference on Software Engineering (ICSE'07)*, May 2007, pp. 723–726.
- [14] A. Riccioni, E. Denti, and R. Laschi, "An experimental environment for teaching java security," in *Proceedings of the 6th International Symposium on Principles and Practice of Programming in Java*, ser. PPPJ '08. New York, NY, USA: ACM, 2008, pp. 13–22. [Online]. Available: <http://doi.acm.org/10.1145/1411732.1411735>
- [15] W. W. Lau and A. H. Yuen, "Promoting conceptual change of learning sorting algorithm through the diagnosis of mental models: The effects of gender and learning styles," *Computers & Education*, vol. 54, no. 1, pp. 275 – 288, 2010. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0360131509002061>
- [16] C. Zander, L. Thomas, B. Simon, L. Murphy, R. McCauley, B. Hanks, and S. Fitzgerald, "Learning styles: Novices decide," *SIGCSE Bull.*, vol. 41, no. 3, pp. 223–227, Jul. 2009. [Online]. Available: <http://doi.acm.org/10.1145/1595496.1562948>
- [17] S. Alaoutinen, "Effects of learning style and student background on self-assessment and course performance," in *Proceedings of the 10th Koli Calling International Conference on Computing Education Research*, ser. Koli Calling '10. New York, NY, USA: ACM, 2010, pp. 5–12. [Online]. Available: <http://doi.acm.org/10.1145/1930464.1930465>
- [18] H. Tie and I. N. Umar, "The impact of learning styles and instructional methods on students recall and retention in programming education," in *Proceedings of the 18th International Conference on Computers in Education*. Putrajaya, Malaysia: Asia-Pacific Society for Computer in Education, 2010.
- [19] A. Gomes and A. J. Mendes, "Studies and proposals about initial programming learning," in *2010 IEEE Frontiers in Education Conference (FIE)*, Oct 2010, pp. S3F–1–S3F–6.
- [20] A. J. Gomes and A. J. Mendes, "A study on student performance in first year cs courses," in *Proceedings of the Fifteenth Annual Conference on Innovation and Technology in Computer Science Education*, ser. ITICSE '10. New York, NY, USA: ACM, 2010, pp. 113–117. [Online]. Available: <http://doi.acm.org/10.1145/1822090.1822123>
- [21] D. W. P. L. Sasithorn Chookaew, Patcharin Panjaburee, "A personalized e-learning environment/tool to promote student's conceptual learning on basic computer programming," *Procedia - Social and Behavioral Sciences*, 2010.
- [22] M. de Raadt and Simon, "My students don't learn the way i do," in *Proceedings of the Thirteenth Australasian Computing Education Conference - Volume 114*, ser. ACE '11. Darlinghurst, Australia, Australia: Australian Computer Society, Inc., 2011, pp. 105–112. [Online]. Available: <http://dl.acm.org/citation.cfm?id=2459936.2459949>
- [23] R. Cardell-Oliver, "How can software metrics help novice programmers?" in *Proceedings of the Thirteenth Australasian Computing Education Conference - Volume 114*, ser. ACE '11. Darlinghurst, Australia, Australia: Australian Computer Society, Inc., 2011, pp. 55–62. [Online]. Available: <http://dl.acm.org/citation.cfm?id=2459936.2459943>
- [24] I. N. Umar and T. H. Hui, "Learning style, metaphor and pair programming: Do they influence performance?" *Procedia - Social and Behavioral Sciences*, vol. 46, pp. 5603 – 5609, 2012. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S1877042812022185>
- [25] M. Othman, M. Othman, and F. M. Hussain, "Designing prototype model of an online collaborative learning system for introductory computer programming course," *Procedia - Social and Behavioral Sciences*, vol. 90, pp. 293 – 302, 2013. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S1877042813019666>
- [26] T. L. Hun, C. K. Loy, and R. M. K. Hansaram, "A study on predicting undergraduate's improvement of academic performances based on their characteristics of learning and approaches at a private higher educational institution," *Procedia - Social and Behavioral Sciences*, vol. 93, pp. 1957 – 1965, 2013. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S1877042813035933>
- [27] B. Alshaigy, "Development of an interactive learning tool to teach python programming language," in *Proceedings of the 18th ACM Conference on Innovation and Technology in Computer Science Education*, ser. ITICSE '13. New York, NY, USA: ACM, 2013, pp. 344–344. [Online]. Available: <http://doi.acm.org/10.1145/2462476.2465601>
- [28] L. Reynoso, E. Grosclaude, L. Snchez, and M. Ivarez, "Technicians and their learning styles preferences and cognitive processes of formal inferences," in *2013 IEEE 12th International Conference on Cognitive Informatics and Cognitive Computing*, July 2013, pp. 51–60.
- [29] C. J. Burrell, "Learning object oriented programming: Unique visualizations of individuals learning styles, activities and the programs produced," in *Proceedings of the 13th Annual Conference on Innovation and Technology in Computer Science Education*, ser. ITICSE '08. New York, NY, USA: ACM, 2008, pp. 339–339. [Online]. Available: <http://doi.acm.org/10.1145/1384271.1384381>
- [30] P. Mozelius and M. Olsson, "Putting the programming hut online self learning for the net-generation," in *ECEL 2015, 14th European Conference on e-Learning ECEL-2015 Hatfield, UK* . Academic Conferences Publishing, 2015.
- [31] B. Alshaigy, S. Kamal, F. Mitchell, C. Martin, and A. Aldea, "Pilet: An interactive learning tool to teach python," in *Proceedings of the Workshop in Primary and Secondary Computing Education*, ser. WiPSCE '15. New York, NY, USA: ACM, 2015, pp. 76–79. [Online]. Available: <http://doi.acm.org/10.1145/2818314.2818319>
- [32] W. T. Tarimo, F. A. Deeb, and T. J. Hickey, "Early detection of at-risk students in cs1 using teachback/spinoza," *J. Comput. Sci. Coll.*, vol. 31, no. 6, pp. 105–111, Jun. 2016. [Online]. Available: <http://dl.acm.org/citation.cfm?id=2904446.2904471>
- [33] K. Falkner and E. Palmer, "Developing authentic problem solving skills in introductory computing classes," *SIGCSE Bull.*, vol. 41, no. 1, pp. 4–8, Mar. 2009. [Online]. Available: <http://doi.acm.org/10.1145/1539024.1508871>
- [34] R. M. Felder and R. Brent, "Understanding student differences," *Journal of Engineering Education*, vol. 94, pp. 57–72, 2005.
- [35] N. Fleming, "Learning Styles Again: VARKing up the right tree!" *Educational developments*, 2006. [Online]. Available: <http://www.vark-learn.com/documents/Educational%20Developments.pdf>
- [36] N. D. Fleming, "Vark: a guide to learning styles." [Online]. Available: <http://vark-learn.com/introduction-to-vark/the-vark-modalities/>
- [37] A. Gregorc, *Gregorc Style Delineator: Developmental Technical and Administration Manual*, ser. Gregorc Style Delineator: Development, Technical and Administration Manual. Gregorc Associates Incorporated, 1984. [Online]. Available: <https://books.google.com.br/books?id=up11AAAACAAJ>
- [38] S. L. K. Felder, Richard M., "Learning and teaching styles in engineering education. engineering education," 1988.
- [39] B. A. Soloman and R. M. Felder, "Index of learning styles questionnaire," 2008.