

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ  
DEPARTAMENTO ACADÊMICO DE COMPUTAÇÃO  
CURSO DE BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

**RENAN KODAMA RODRIGUES**

**ESTUDO DO USO DA LINGUAGEM DE BLOCOS  
SCRATCH NO ENSINO DO PENSAMENTO  
COMPUTACIONAL**

MONOGRAFIA

**CAMPO MOURÃO**

**2019**

**RENAN KODAMA RODRIGUES**

**ESTUDO DO USO DA LINGUAGEM DE BLOCOS  
SCRATCH NO ENSINO DO PENSAMENTO  
COMPUTACIONAL**

Trabalho de Conclusão de Curso de Graduação apresentado à disciplina de Trabalho de Conclusão de Curso 2, do Curso de Bacharelado em Ciência da Computação do Departamento Acadêmico de Computação da Universidade Tecnológica Federal do Paraná, como requisito parcial para obtenção do título de Bacharel em Ciência da Computação.

Orientador: Prof. Dr. Igor Scaliante Wiese

Coorientador: Prof. Dr. Marco Aurélio Graciotto Silva

**CAMPO MOURÃO**

**2019**



## ATA DE DEFESA DO TRABALHO DE CONCLUSÃO DE CURSO

Às **16:00** do dia **28 de junho de 2019** foi realizada na sala **E007** da UTFPR-CM a sessão pública da defesa do Trabalho de Conclusão do Curso de Bacharelado em Ciência da Computação do(a) acadêmico(a) **Renan Kodama Rodrigues** com o título **ESTUDO DO USO DA LINGUAGEM DE BLOCOS SCRATCH NO ENSINO DO PENSAMENTO COMPUTACIONAL**. Estavam presentes, além do(a) acadêmico(a), os membros da banca examinadora composta por: **Prof. Dr. Igor Scaliante Wiese** (orientador), **Prof. Dr. Marco Aurélio Graciotto Silva**, **Prof. Dr. André Luis Schwerz** e **Prof. Dr. Rafael Liberato Roberto**. Inicialmente, o(a) acadêmico(a) fez a apresentação do seu trabalho, sendo, em seguida, arguido(a) pela banca examinadora. Após as arguições, sem a presença do(a) acadêmico(a), a banca examinadora o(a) considerou \_\_\_\_\_ na disciplina de Trabalho de Conclusão de Curso **2** e atribuiu, em consenso, a nota \_\_\_\_\_ (\_\_\_\_\_). Este resultado foi comunicado ao(à) acadêmico(a) e aos presentes na sessão pública. A banca examinadora também comunicou ao acadêmico(a) que este resultado fica condicionado à entrega da versão final dentro dos padrões e da documentação exigida pela UTFPR ao professor Responsável do TCC no prazo de **onze dias**. Em seguida foi encerrada a sessão e, para constar, foi lavrada a presente Ata que segue assinada pelos membros da banca examinadora, após lida e considerada conforme.

Observações: \_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

Campo Mourão, **28 de junho de 2019**

\_\_\_\_\_  
**Prof. Dr. Marco Aurélio Graciotto  
Silva**  
Membro 1

\_\_\_\_\_  
**Prof. Dr. André Luis Schwerz**  
Membro 2

\_\_\_\_\_  
**Prof. Dr. Rafael Liberato Roberto**  
Membro 3

\_\_\_\_\_  
**Prof. Dr. Igor Scaliante Wiese**  
Orientador

**A ata de defesa assinada encontra-se na coordenação do curso.**

# Agradecimentos

A minha família e namorada por darem apoio para superar meus obstáculos.

A esta universidade, seu corpo docente, direção e administração.

Ao meu orientador Prof. Dr. Igor Scaliante Wiese e ao meu coorientador Prof. Dr. Marco Aurélio Graciotto Silva.

Aos professores e professoras, Prof<sup>a</sup>. Dr<sup>a</sup>. Adriana da Silva Fontes, Prof<sup>a</sup>. Ana Claudia Maciel, Prof. Dr. Marcos Silvano Orita Almeida, Prof. Dr. Ivanilton Polato, Prof. Dr. Diego Bertolini Goncalves, Prof. Dr. Narci Nogueira da Silva e Prof. Yuri Feitosa, por terem cedido a oportunidade de aplicar o estudo em suas turmas.

E a todos que participaram diretamente ou indiretamente deste estudo, o meu muito obrigado.

# Resumo

---

Rodrigues, Renan Kodama. Estudo do uso da Linguagem de Blocos Scratch no Ensino do Pensamento Computacional. 2019. 61. f. Monografia (Curso de Bacharelado em Ciência da Computação), Universidade Tecnológica Federal do Paraná. Campo Mourão, 2019.

Devido ao grande avanço e a constante integração da sociedade nas áreas de tecnologia e desenvolvimento, a computação está presente em quase todos os seguimentos de trabalho principalmente no que refere-se à programação. Entretanto, mesmo os dispositivos tecnológicos estando presentes no cotidiano de uma parte considerável da população, ela não apresenta habilidades para desenvolver aplicativos, aperfeiçoá-los ou criá-los. Logo, para desenvolver as habilidades que permitam ao estudante aperfeiçoar ou criar novos aplicativos e funcionalidades, se faz necessário integrar os educandos às técnicas do pensamento computacional. Sendo assim, esta pesquisa apresenta um estudo da linguagem de programação em blocos com Scratch para promover o ensino das técnicas do pensamento computacional, aplicada através de um curso semipresencial guiada por meio de videoaulas. Para isso, foi elaborado um material educativo separado em seis módulos, contendo em cada módulo videoaulas e questionários. Para a realização deste estudo, foram aplicadas atividades para avaliar o desempenho e questionários de motivações para analisar os estímulos causados nos participantes em cada módulo do experimento. O estudo foi aplicado em 5 diferentes turmas, sendo todas do 1º período e integrantes da Universidade Tecnológica Federal do Paraná (UTFPR), no câmpus Campo Mourão. Com o intuito de avaliar o desempenho dos estudantes em relação aos conceitos do pensamento computacional que serão apresentados em videoaulas, utilizando como ferramenta a plataforma Scratch para o desenvolvimento das atividades avaliadas. Os resultados mostrará que o desempenho dos participantes apresentaram resultados conceituados como satisfatórios tanto nas atividades das semanas quanto nos conceitos apresentados, enquanto também, permaneciam motivados durante as fases do experimento. Podendo este indicar um forte indício positivo da influência da programação em blocos com Scratch, no ensino/aprendizado dos conceitos de programação e da Engenharia de Software.

**Palavras-chaves:** Pensamento Computacional. Scratch. Educação. Programação em Blocos.

# Abstract

---

Rodrigues, Renan Kodama. Study of the Use of Language from Scratch Blocks in Teaching Computational Thinking. 2019. 61. f. Monograph (Undergraduate Program in Computer Science), Federal University of Technology – Paraná. Campo Mourão, PR, Brazil, 2019.

Because of the great advancement and constant integration of society in the areas of technology and development, computing is present in almost all work segments, especially when it comes to programming. However, even the technological devices being present in the daily lives of a considerable part of the population, it does not have the skills to develop applications, to perfect them or to create them. Therefore, in order to develop the skills that allow the student to improve or create new applications and functionalities, it is necessary to integrate the learners with the techniques of computational thinking. Thus, this research presents a study of the block programming language with Scratch to promote the teaching of computational thinking techniques, applied through a semi - classroom course guided through video lessons. For this, a separate educational material was elaborated in six modules, containing in each module videos and questionnaires. For the accomplishment of this study, activities were applied to evaluate the performance and questionnaires of motivations to analyze the stimuli caused in the participants in each module of the experiment. The study was applied in 5 different classes, all of them from the 1st period and members of the Federal University of Paraná (UTFPR), in Campo Mourão campus. In order to evaluate the performance of students in relation to the concepts of computational thinking that will be presented in video lessons, using as a tool the Scratch platform for the development of the evaluated activities. The results showed that the performance of the participants presented results that were considered satisfactory both in the activities of the weeks and in the presented concepts, while also remaining motivated during the phases of the experiment. This may indicate a strong positive indication of the influence of programming in blocks with Scratch, in teaching / learning programming concepts and Software Engineering.

**Keywords:** Computational Thinking. Scratch. Education. Block Programming.

# Lista de figuras

---

4.1	Apresentação do fluxograma das atividades desenvolvidas. . . . .	23
4.2	Exemplo da atividade elaborada na 1ª semana, labirinto na qual o jogador deve atravessar. . . . .	26
4.3	Exemplo da atividade elaborada na 2ª semana, cenário da caça e caçador. . .	27
4.4	Exemplo da atividade elaborada na 3ª semana, retratando o jogo no estilo perguntas e respostas. . . . .	27
4.5	Exemplo da atividade elaborada na 4ª semana, apresentado um jogo no estilo atravesse a rua. . . . .	28
4.6	Exemplo da atividade elaborada na 5ª semana similar ao jogo Flappy Bird. .	28
4.7	Exemplo da atividade elaborada na 6ª semana, baseando-se em jogos no estilo plataforma. . . . .	29
4.8	Fragmento do modelo estrutural da primeira semana. . . . .	30
5.1	Gráfico de linhas representando a quantidade de participantes ativos por semana.	34
5.2	Gráfico de colunas representando os níveis de satisfação dos participantes por semana. . . . .	35
5.3	Gráfico de combinação representando o desempenho dos participantes do Grupo 2 na Semana 01. . . . .	36
5.4	Gráfico de combinação representando o desempenho dos participantes do Grupo 2 na Semana 02. . . . .	37
5.5	Gráfico de combinação representando o desempenho dos participantes do Grupo 2 na Semana 03. . . . .	38
5.6	Gráfico de combinação representando o desempenho dos participantes do Grupo 2 na Semana 04. . . . .	39
5.7	Gráfico de combinação representando o desempenho dos participantes do Grupo 2 na Semana 05. . . . .	40
5.8	Gráfico de combinação representando o desempenho dos participantes do Grupo 2 na Semana 06. . . . .	41
5.9	Gráfico de linhas representando o desempenho geral do Grupo 2 nos questionários.	42
5.10	Gráfico de combinação. Desempenho geral do Grupo 2 nos conceitos dos questionários. . . . .	43

5.11	Gráfico de colunas empilhadas representando o desempenho geral do Grupo 2.	44
5.12	Gráfico de linhas representando o fluxo do desempenho geral do Grupo 2 nos conceitos apresentados. . . . .	45
5.13	Gráfico de linhas representando o fluxo do desempenho geral do Grupo 2 nos conceitos apresentados. . . . .	46
5.14	Gráfico de colunas representando a motivação do Grupo 2 nas semanas de 1 à 5.	48
5.15	Gráfico de setores representando a proporção dos comentários positivos, neutros e negativos. . . . .	49
5.16	Imagem gerada através da técnica de <i>Word Clouds</i> , destacando as principais palavras de maior ocorrência. . . . .	50



# Lista de tabelas

---

A.1	Afirmações para a análise da motivação de acordo com modelo IMMS. . . . .	53
A.2	Continuação da tabela de afirmações para a análise da motivação pelo modelo IMMS. . . . .	54
B.1	Mapeamento dos conteúdos abordados por tópicos e apresentação da forma de aplicação. . . . .	55
B.2	Continuação dos mapeamento dos conteúdos abordados por tópicos. . . . .	56
C.1	Mapeamento dos conceitos abordados em tópicos referentes às 1 <sup>o</sup> e 2 <sup>o</sup> semanas.	57
C.2	Continuação dos mapeamentos dos conteúdos, sendo apresentadas as 3 <sup>o</sup> e 4 <sup>o</sup> semanas. . . . .	58
C.3	Mapeamento final dos conteúdos, sendo apresentadas as 5 <sup>o</sup> e 6 <sup>o</sup> semanas. . .	59

# Sumário

---

<b>1</b>	<b>Introdução</b>	<b>10</b>
<b>2</b>	<b>Fundamentação Teórica</b>	<b>14</b>
2.1	Pensamento Computacional . . . . .	14
2.2	Programação em Blocos . . . . .	16
2.2.1	Scratch . . . . .	17
2.3	Análise Motivacional . . . . .	17
2.3.1	Modelo RIMMS . . . . .	18
2.4	Dr. Scratch . . . . .	19
<b>3</b>	<b>Trabalhos Relacionados</b>	<b>20</b>
3.1	Considerações Finais . . . . .	21
<b>4</b>	<b>Metodologia</b>	<b>23</b>
4.1	Busca e Leitura de Artigos Relacionados . . . . .	24
4.2	Planejamento de Aula . . . . .	24
4.3	Elaboração dos Exercícios . . . . .	25
4.3.1	Avaliações Motivacionais . . . . .	25
4.3.2	Desafios Propostos . . . . .	26
4.4	Elaboração do Curso na Plataforma Moodle de Ensino . . . . .	29
4.5	Aplicação . . . . .	30
4.5.1	Ciclo das Aulas . . . . .	31
4.6	Coleta e Análise dos Dados . . . . .	31
<b>5</b>	<b>Resultados e Discussões</b>	<b>33</b>
5.1	Dados Descartados . . . . .	33
5.2	Análise do Desempenho nos Questionários . . . . .	35
5.3	Análise do Desempenho nos Projetos . . . . .	43
5.4	Análise do Desempenho Motivacional . . . . .	47
<b>6</b>	<b>Conclusão</b>	<b>51</b>
	<b>Apêndices</b>	<b>52</b>

A	Lista de afirmações do modelo IMMS	53
B	Mapeamento das avaliações e Abordagens	55
C	Mapemanto dos Conceitos Lecionados	57
	Referências	60

---

## Introdução

---

Com o decorrer da constante integração da sociedade em meios tecnológicos, o pensamento computacional é tão importante para a vida e gestão em uma sociedade quanto os conhecimentos básicos de física, história, matemática, dentre outras disciplinas lecionadas no contexto atual da educação (BRACKMANN, 2017). Sendo assim, podemos entender o pensamento computacional como sendo um tipo de pensamento analítico e criterioso, que combina técnicas do pensamento lógico e matemático em geral. A essência do pensamento computacional é o conceito de abstração. Na computação são abstraídas noções além das dimensões físicas do tempo e espaço, sendo estas abstrações extremamente genéricas, uma vez que são simbólicas e tendem a ser mais ricas e complexas do que a matemática e ciências físicas (WING, 2008).

A computação está presente no cotidiano e em quase todos os seguimentos de trabalho, inclusive nas áreas de pesquisa, como é mencionado no artigo *A Model Curriculum for K-12 Computer Science* (TUCKER, 2003), no qual é relatado que a grande parte das profissões existentes no século 21 exigem uma compreensão na área da ciência da computação, principalmente no que refere-se à programação.

Entretanto, mesmo a tecnologia estando presente em boa parte do dia-a-dia, é notável que grande parte das pessoas não apresentam habilidades para desenvolver aplicações, aperfeiçoá-las ou criá-las, apesar da maioria demonstrar domínio no manuseio ou utilização de tais dispositivos, como descrito por Christian Brackmann em *Desenvolvimento do Pensamento Computacional Através de Atividades Desplugadas na Educação Básica*, “seria como se eles conseguissem ler, mas não conseguissem escrever com as novas tecnologias” (BRACKMANN, 2017, p.19).

A abordagem das tecnologias é contemplada nas Diretrizes Curriculares Nacionais Gerais para a Educação Básica, ao incluir dentre os objetivos da formação básica “a compreensão do ambiente natural e social, do sistema político, da economia, da tecnologia,

das artes, da cultura, dos direitos humanos e dos valores em que se fundamenta a sociedade”. Nesta perspectiva, as novas mídias e tecnologias educacionais serão utilizadas como “processo de dinamização dos ambientes de aprendizagem”. Por fim, ter acesso à diversas fontes de pesquisas para buscar, analisar e reproduzir novas referências e novos conhecimentos, podem fornecer habilidades necessárias para a utilização adequada de novas tecnologias da informação e da comunicação, assim como, realizar procedimentos básicos de busca e produções nas áreas científicas (BRASIL, 2013).

Sendo assim, para ensinar conceitos do pensamento computacional são utilizadas linguagens de programação, promovendo um processo de ensino-aprendizagem mais inclusivo e motivador, pois estimula habilidades e competências que extrapolam o conhecimento curricular. A interação dos alunos para a construção de soluções em diferentes tipos de problemas por meio do uso de recursos tecnológicos, desenvolve o pensamento lógico, a decomposição, a associação, a abstração e o reconhecimento de padrões (EGUCHI, 2010). Logo, a educação por meio de técnicas computacionais é uma ferramenta única que oferece atividades atrativas e uma aprendizagem mais envolvente, despertando interesses e curiosidades nos estudantes, gerando experiências de aprendizagem mais construtivas. Entretanto, o uso de linguagens de programação não é trivial, dado a sua complexidade de compreensão e uso (ALIMISIS, 2013).

Por fim, podemos concluir que compreender o pensamento computacional é de grande importância nos dias atuais, contudo seu aprendizado não é trivial. Sendo assim, visando introduzir e desenvolver técnicas voltadas ao pensamento computacional, a linguagem em blocos é utilizada para atrair e lecionar conceitos computacionais para pessoas inexperientes em programação. Em geral, tais ambientes que usam a programação em blocos, utilizam metáforas ou abstrações primitivas de programação, na qual, são fornecidas dicas visuais sobre como e onde os comandos podem ser usados. Em resumo, a programação para este tipo de linguagem consiste basicamente em arrastar os elementos para uma área de desenvolvimento e juntá-los para compor a construção lógica da solução do problema (WERNTROP; WILENSKY, 2017).

A utilização de linguagens de programação baseadas em técnicas visuais e de multimídia, como a plataforma Scratch, proporciona diversos benefícios quanto à construção do conhecimento sobre o assunto. Logo, a linguagem em blocos permite maior abstração quando comparada à outras linguagens de baixo nível (SILVA et al., 2015).

O Scratch é uma linguagem de programação visual baseada em blocos, desenvolvida e sustentada pelo grupo Lifelong Kindergarten do Instituto de Tecnologia de Massachusetts (MIT). Criada em 2007, foi inicialmente idealizada para atender um público com idades entre 8 e 16 anos. Porém, é utilizada hoje por pessoas em qualquer faixa etária (REZENDE; BISPO, 2018). Como descrito por Mitchel Resnick, o objetivo desta linguagem é permitir que o estudante possa programar ou desenvolver sua história, colaborando assim no desenvolvimento da criatividade e dos processos para a resolução de problemas (RESNICK et al., 2009).

Neste contexto, a busca frequente por novos métodos de ensino tem atraído o esforço

de educadores em todo o mundo, com o propósito de buscar soluções para aprimorar a forma de ensino tradicional que é empregada no ambiente escolar, de modo, que estimule a construção de novas estruturas mentais. A questão principal na qual este estudo foi conduzido, é apresentado a seguir:

- Q1: Os participantes dos grupos de aprendizagem apresentaram bons desempenhos nas atividades envolvendo conceitos do pensamento computacional utilizando a linguagem Scratch?

A partir do exposto, o objetivo deste trabalho é promover o ensino-aprendizado do pensamento computacional em diferentes grupos de participantes por meio da linguagem de blocos Scratch. Para isso, foi elaborado um material educativo separado em seis módulos, contendo em cada módulo videoaulas e questionários.

Nestes módulos, o estudante deverá assistir os vídeos e responder questionários de múltiplas escolhas, seguida da submissões de projetos em Scratch. O conteúdo dos vídeos elaborados nos módulos abordam o desenvolvimento de simples jogos computacionais, como é descrito na Seção 4.3.2.

Tal estudo foi aplicado em 5 diferentes grupos de diferentes cursos, sendo todos integrantes da Universidade Tecnológica Federal do Paraná do câmpus Campo Mourão. O Grupo 1 é composto pelos professores de física na disciplina de mestrado Atividades Computacionais para o Ensino Médio e Fundamental. O Grupo 2 é formado pelos estudantes do 1º ano do Ensino Médio com Técnico Integrado em Informática da UTFPR-CM, na disciplina de algoritmos. O Grupo 3 é composto pelos estudantes do 1º período dos cursos de Engenharia de Alimentos e Engenharia Química na disciplina de programação. O Grupo 4 é formado pelos estudantes do 1º período do curso de Bacharelado em Ciência da Computação na disciplina de algoritmos. Por fim, o Grupo 5 é composto pelos estudantes do 1º período do curso de Engenharia Eletrônica na disciplina de algoritmos, como apresentado na Seção 4.5.

Após a coleta dos dados, como apresentado na Seção 5, as análises ocorreram sobre o Grupo 2, composto pelos estudantes do 1º ano do Ensino Médio com Técnico Integrado em Informática da UTFPR-CM. Os demais grupos foram descartados, pois muitos participantes não realizaram as atividades resultando em dados insuficientes para as análises.

É possível afirmar, que os participantes apresentaram desempenhos considerados como satisfatórios tanto nos questionários de múltipla escolha, quanto nos projetos em Scratch, ao longo das semanas. Também, foi possível afirmar que os participantes mantiveram-se motivados durante o experimento, através das análises dos questionários motivacionais.

Sendo assim, os resultados de desempenho dos participantes nos questionários e projetos foram considerados como satisfatórios, enquanto também, os participantes mantiveram-se motivados durante o estudo, sendo confirmadas pelas análises motivacionais. Logo, podemos considerar, que o Scratch pode trazer benefícios no ensino/aprendizagem dos

conceitos de programação e Engenharia de Software. Na qual, este método de ensino, tem o potencial de se tornar um instrumento para a aprimoração dos modelos atuais.

Este trabalho está estruturado da seguinte forma. No Capítulo 2 é apresentada a fundamentação teórica utilizada para fomentar os principais conceitos científicos utilizados neste trabalho. No Capítulo 3 são apresentados os trabalhos utilizados no desenvolvimento das atividades para o curso. No Capítulo 4 é apresentada a metodologia aqui desenvolvida. No Capítulo 5 é apresentado os dados obtidos e as análises realizadas sobre as turmas concluintes. Por fim, no Capítulo 6 é apresentado as conclusões obtidas acerca do estudo realizado.

---

## Fundamentação Teórica

---

Nas sessões seguintes, serão apresentados as principais fundamentações para a compreensão deste trabalho, sendo assim, na Seção 2.1, é apresentado a importância e explicação do que é o pensamento computacional, sendo o ponto chave deste estudo. Com o objetivo de apresentar o pensamento computacional por meio da programação em blocos, a Seção 2.2, aborda alguns dos relatos sobre o uso da programação em blocos, seus resultados, assim como a explicação da ferramenta Scratch. Na Seção 2.3.1 é apresentado o modelo de avaliação motivacional de materiais didáticos, sendo utilizado para analisar se os participantes estiveram interessados nas atividades. Na Seção 2.4 é apresentado uma ferramenta para a avaliação dos projetos em Scratch, avaliando alguns dos conceitos do pensamento computacional sendo eles: abstração, paralelismo, pensamento lógico, sincronização, controle de fluxo, interatividade com o usuário e representação de dados.

### 2.1. Pensamento Computacional

O pensamento computacional é uma habilidade fundamental recomendada para todos, não somente para cientistas, pois colabora no desenvolvimento de competências tais como: ler, escrever e aperfeiçoamento do pensamento lógico. Tal concepção envolve também outros tipos de habilidades tais como resolução de problemas, construção de sistemas e entendimento do comportamento humano (WING, 2006).

Através de uma resposta obtida a partir de um problema é possível saber se a resolução aproximada satisfaz as condições, logo tal entendimento computacional está reestruturando pensamentos difíceis em pensamentos mais simples para o desenvolvimento de uma solução, seja por meio de reduções, incorporações, transformações ou simulações. O pensamento computacional usa técnicas de abstração e decomposição para soluções em sistemas ou tarefas complexas, procurando a melhor representação e convertendo problemas por meio de soluções



mais tangíveis (WING, 2006).

Uma vez apresentado um problema, por meio de técnicas desta concepção, ele será dividido em problemas menores, logo tais fragmentos podem ser analisados de forma individual, focando em detalhes mais relevantes e ignorando outros que não agregam na solução deste fragmento, facilitando assim o processo de identificação de problemas e reconhecimento de comportamento. Sendo assim, tais técnicas desenvolvidas no pensamento computacional são identificadas como: decomposição, reconhecimento de padrões, abstração e algoritmos. O pensamento computacional deveria ser uma habilidade básica para qualquer pessoa ou área, pois no instante que temos o conhecimento da problemática é possível definir o que é necessário para atingir os objetivos estabelecidos, assim como é possível também saber os passos que não devem serem seguidos (BRACKMANN, 2017).

O pensamento computacional está atuando na pesquisa em muitas disciplinas, tanto nas ciências quanto nas humanidades, por exemplo, a computação que permitiu aos pesquisadores realizarem perguntas sobre um grande conjunto de dados e conseqüentemente analisar os resultados obtidos através de representações visuais. O pensamento computacional colaborou e colabora nas mais diversas áreas tais como geologia, psicologia, biologia, astrologia, física aplicada, quântica, medicina, entre muitas áreas. Tal capacidade fornecida pela computação para a investigação de novas questões, possibilitou a infiltração dos conceitos computacionais e teorias em outras disciplinas. Através da influência do cálculo raramente pode-se prever de forma instantânea os resultados, pois em geral é mais sutil e mais profunda tal solução (BUNDY, 2007).

Como descrito pelo artigo *Computational thinking and thinking about computing*, o pensamento computacional é um tipo de pensamento analítico e criterioso, que compartilha o pensamento lógico e matemático em geral. Resultando assim em variadas formas de abordagens para a resolução correta de um problema, sendo cada uma dessas resoluções reflexo do raciocínio lógico e domínio do problema de cada indivíduo. Sendo assim, tal tipo de pensamento é análogo ao pensamento científico em geral, podendo compreender nas abordagens da computabilidade tais como: a inteligência, a mente e o comportamento humano.

A essência do pensamento computacional é a abstração, na computação são abstraídas noções além das dimensões físicas do tempo e espaço, sendo estas abstrações extremamente genéricas, uma vez que são simbólicas e tendem a ser mais ricas e complexas do que a matemática e ciências físicas. A reflexão computacional irá influenciar todas as áreas e campos de atuação, uma vez que a sociedade encontra-se cada vez mais dependente da tecnologia (WING, 2008).

## 2.2. Programação em Blocos

A programação nos dias atuais é reconhecida como uma nova habilidade do Século 21. Sendo assim, as instituições de ensino estão rapidamente integrando novos cursos de ciência da computação e pensamentos computacionais em programas que tradicionalmente não utilizavam a computação. A técnica de desenvolvimento em blocos está se tornando um método cada vez mais fácil e eficaz de introduzir novos alunos aos conceitos de programação, com uma variedade de ferramentas para o desenvolvimento como exemplo as plataformas Scratch e Alice. Em geral, a programação em blocos consiste em um desenvolvimento mais interativo e visual do que a programação tradicional, na qual, os blocos representam instruções computacionais, podendo estes serem encaixados de diferentes formas, gerando assim diferentes respostas (WERNTROP; WILENSKY, 2015).

Para a primeira análise, os autores David Werntrop e Uri Wilensky discutiram como as características de exibição visual e dinâmica de interação em técnicas de programação em blocos são compreendidas pelos estudantes em comparação com técnicas usuais de programação nas quais não utilizam elementos visuais. Como relatado, os estudantes apresentaram um melhor desempenho em questões que utilizavam elementos visuais de programação, entretanto os autores descrevem dificuldades para mapear quais elementos facilitam na compreensão dos trechos em blocos comparados à outras formas textuais de programação, uma vez que existem inúmeros fatores que podem influenciar no entendimento de trechos descritos através dos blocos (WERNTROP; WILENSKY, 2015).

O artigo *Comparing Block-Based and Text-Based Programming in High School Computer Science Classrooms*, apresenta um estudo comparativo e sistemático em sala de aula, abordando como a modalidade de programação impacta os alunos em questões como: atitudes, percepções e aprendizado conceitual. Como resultado foi apresentado que tal modalidade de ensino tem um grande impacto positivo nas experiências dos educandos com a programação. A primeira contribuição do estudo relatado é de que os alunos que usaram técnicas baseadas em programação em blocos, apresentaram ganhos de aprendizado significativos, após cinco semanas de aulas assistidas em sala. Constatou-se um padrão positivo consistente de estudantes na condição de ensino por conceitos de blocos superando os educandos baseados em ensino por linguagens textuais.

O desempenho dos educandos pela qualidade das respostas foi analisado por meio de perguntas que abordam temas de programação, onde demonstrou-se que os alunos na condição de aprendizagem por blocos apresentaram melhor desempenho. É importante compreender a relação entre a modalidade e a aprendizagem, pois é consequente no que diz respeito a decidir quais ferramentas usar em sala de aula e informar os ambientes de programação introdutória (WERNTROP; WILENSKY, 2017).

### 2.2.1. Scratch

O Scratch é uma linguagem de programação visual, desenvolvida e mantida pelo grupo Lifelong Kindergarten do Instituto de Tecnologia de Massachusetts (MIT). Criada em 2007, foi inicialmente idealizada para atender um público com idade entre 8 e 16 anos. Porém, é utilizada hoje por pessoas em qualquer faixa etária. A linguagem utiliza elementos visuais e de multimídia possibilitando a criação de sequências por meio de animações, colaborando de forma positiva na aprendizagem de programação de forma simples e eficiente por meio de "blocos" que representam funcionalidades específicas no personagem. Nele o usuário pode arrastar blocos que representam comandos de programação, sendo possível criar diversos jogos utilizando aspectos de animação por meio da interação entre os objetos do cenário (REZENDE; BISPO, 2018).

Assim como é descrito por Mitchel Resnick, o Scratch é uma linguagem de programação de comunidade aberta, que permite criar jogos interativos, histórias e animações utilizando como desenvolvimento a programação em blocos. Podendo também estes projetos serem compartilhados na comunidade. Quando as crianças estão desenvolvendo os projetos de programação, elas simultaneamente estão aprendendo a pensar criativamente aperfeiçoando o raciocínio lógico e trabalhando em equipe, enquanto também aprendem conceitos importantes das áreas da computação.

Por tanto, a linguagem Scratch oferece um ambiente no qual os educandos possam programar suas histórias. A intenção da linguagem é oferecer ao aluno um ambiente em que ele mesmo possa interagir e resolver os desafios de acordo com seus conhecimentos. Isto exige o desenvolvimento de sua criatividade no processo de resolução de problemas. Esse desenvolvimento está aliado à lógica de programação de uma maneira mais intuitiva e agradável, pois esta linguagem proporciona o encapsulamento de detalhes sobre a sintaxe de linguagens de programação textuais. Como a programação com o Scratch é feita em blocos, o usuário organiza estes blocos de acordo com o seu raciocínio, a fim de prover uma sequência lógica na história (RESNICK et al., 2009).

## 2.3. Análise Motivacional

Na área da ciência educacional, os modelos de IMMS (Pesquisa de Motivação de Materiais Didáticos do inglês *Instructional Materials Motivation Survey*) desenvolvidos por Keller no ano de 2010, têm sido utilizados para aplicar estratégias motivacionais em materiais didáticos e analisar seus efeitos sobre os educandos (LOORBACH et al., 2015). Originalmente o modelo foi criado para analisar a motivação dos educandos em sala de aula, por meio de aulas ministradas de forma presencial, ou seja, estudantes e orientadores estariam inseridos no mesmo ambiente de aprendizagem. Entretanto, atualmente o modelo também é aplicado para analisar questões motivacionais em outros contextos, como por exemplo, a instrução

assistida utilizando computadores ou aparelhos celulares.

Keller apresentou o modelo IMMS em *Motivational design for learning and performance: the ARCS model approach* no ano de 2010, no qual é apresentada uma medida situacional de trinta e seis itens do comportamento relacional das pessoas, focando nos constructos de atenção, relevância, confiança e satisfação, resultado assim em uma pontuação geral de motivação, podendo este modelo ser observado através das Tabelas A.1 e A.2 referenciadas no Apêndice A. O modelo IMMS foi projetado para estar em correspondência com a fundamentação teórica dos conceitos e teorias motivacionais que compõem o modelo ARCS (Atenção, Relevância, Confiança e Satisfação, do inglês *Attention, Relevance, Confidence and Satisfaction*). Tal pesquisa foi desenvolvida para medir reações em materiais instrucionais auto-dirigidos, sendo um auto relato de situação específica, podendo ser usada para estimar o comportamento motivacional dos educandos em qualquer contexto de ensino (LOORBACH et al., 2015).

### 2.3.1. Modelo RIMMS

Como mencionado na pesquisa realizado por Loorbach em *Validation of the Instructional Materials Motivation Survey (IMMS) in a self-directed instructional setting aimed at working with technology*, o RIMMS (Pesquisa Reduzida de Motivação de Materiais Didáticos do inglês *Reduced Instructional Materials Motivation Survey IMMS*) é um modelo de pesquisa motivacional que mensura os quatro construtores que compõe o modelo ARCS sendo eles: atenção, relevância, confiança e satisfação.

O modelo RIMMS é composto por 12 afirmações, sendo uma versão reduzida da versão original IMMS que contém 36 afirmações, podendo estas serem observadas no Apêndice A, na qual, identificadores marcados com asterisco compõem o modelo RIMMS. As afirmações do modelo podem ser relacionadas com os quatro construtores de atenção, relevância, confiança e satisfação, sendo identificadas pelas iniciais de cada palavra dos indicadores. Portanto, o modelo contém 4 grupos de construtores, na qual, cada grupo contém 3 afirmações e cada afirmação é avaliada em uma escala Likert de resposta de 1 à 5, onde, 1 representa que o participante discordou totalmente ou que está totalmente desmotivado, e 5 que ele concorda totalmente ou que está totalmente motivado.

Para o construtor de atenção é verificado se o participante está curioso e/ou interessado pelas aulas. O construtor de relevância analisa se as aulas estão relacionadas a objetivos ou motivos pessoais importantes e se o participante está conectado ao contexto. No construtor de confiança é analisado se os participantes estão motivados devido a pouca ou muita confiança ou expectativas de sucesso. No construtor de satisfação é verificado se há sentimentos de satisfação com o processo ou com os resultados da experiência de aprendizado (LOORBACH et al., 2015).

O estudo realizado por Loorbach mostrou que em um ambiente instrucional auto-dirigido visando motivar os participantes a aprender e a executar tarefas, os resultados obtidos através do modelo RIMMS se ajustam aos quatro construtores de atenção, relevância, confiança e satisfação melhor do que o modelo original IMMS (LOORBACH et al., 2015).

## 2.4. Dr. Scratch

De acordo com o trabalho apresentado por Jesus Moreno e Robles em *Comparing computational thinking development assessment scores with software complexity metrics*, Dr. Scratch é uma ferramenta livre de código aberto, que analisa projetos em Scratch, avaliando tais projeto de acordo com conceitos do pensamento computacional. Os conceitos avaliados pelo Dr. Scratch são: abstração, paralelismo, pensamento lógico, sincronização, controle de fluxo, interatividade com o usuário e representação de dados. A nota para cada conceito consiste em números de 0 à 3. Somando o total das notas em cada conceito é obtido a nota final do projeto, na qual, projetos com notas de até 7 pontos são considerados simples, entre 8 à 14 pontos são considerados como satisfatórios e acima de 15 pontos são considerados como ótimos (MORENO-LEON et al., 2016).

A ferramenta Dr. Scratch permite que os estudantes avaliem seus próprios projetos, através de uma pontuação recebida que mensura seu raciocínio computacional. Possibilitando assim, o estudante a descobrir seu próprio grau de desenvolvimento para essa habilidade (MORENO-LEON et al., 2016). Também é relatado pelo autor que a ferramenta fornece relatórios com sugestões e dicas para melhorar o código desenvolvido pelo estudante. Os relatórios fornecidos pela ferramenta visam incentivar os estudantes a continuarem desenvolvendo seus códigos e aperfeiçoá-los.

---

## Trabalhos Relacionados

---

O relato mencionado pelas autoras Felienne e Efthimia, registra uma experiência de ensino de programação em blocos para crianças de até doze anos de idade, conduzido na Holanda, o qual tinha por objetivo analisar o comportamento e desempenho dos estudantes em relação às boas práticas de programação.

Em resumo, o curso elaborado pelas autoras foi organizado em módulos semanais somando no total seis semanas. Introduzindo de forma incremental as novas funcionalidades presentes na plataforma Scratch, elementos e conceitos da programação e engenharia de software. A coleta e análises dos resultados envolvendo os testes avaliativos, tais como submissões e questionários, permitiu às autoras concluir que, os educandos não apresentaram dificuldades em compreender qualquer dos conceitos seja ele da engenharia de software ou programação. Para esta conclusão, foram analisadas as relações do desempenho nos conceitos de engenharia de software e programação, tais como: uso de laços de repetição e condição, paralelização, código morto, nomeação, método longo, entre outros mencionados no artigo (HERMANS; AIVALOGLOU, 2017).

Logo, a pesquisa feita por Felienne e Efthimia sugere que, ambos os conceitos podem ser ensinados simultaneamente à um dado grupo separado por faixas etárias, no qual o grupo selecionado abrangia crianças de até 12 anos de idade. Em outra análise o artigo também apresenta que existe uma relação entre a idade e a forma de desenvolvimento da atividade, principalmente nos conceitos que envolvem a utilização de operadores e procedimentos para o desenvolvimento do código (HERMANS; AIVALOGLOU, 2017).

Em um outro trabalho elaborado pelos autores Priscila S. et. al., é apresentado uma revisão sistemática em iniciativas acadêmicas para desenvolver o pensamento computacional e programação. Utilizando como contexto a educação brasileira nos níveis fundamental e médio, na qual, foram selecionados 4108 artigos, entretanto apenas 2018 artigos foram selecionados para a análise (SANTOS et al., 2018).

Como resultado das metodologias aplicadas, descobriram que, abordagens utilizadas para promover o pensamento computacional e programação empregavam com mais frequência metodologias baseadas no uso ou desenvolvimento de jogos, seguida de abordagens utilizando robótica. Em uma análise realizada nas abordagens utilizando jogos, constaram que as ferramentas mais utilizadas para promover tal método de ensino foram as seguintes em ordem de popularidade: Scratch, Lego, Kits de Robótica e Arduinos.

Analisando a linguagem mais popular dentre as pesquisas coletadas, a linguagem baseada em blocos foi a linguagem que mais se destacou para promover o pensamento computacional, seguidas pelas linguagens C, pseudocódigo e java. Para as metodologias baseadas em Scratch estas estavam mais ligadas à aplicação em turmas do ensino fundamental, conclui-se que o uso dessa ferramenta tem sido mais frequente entre os alunos de 6 a 14 anos de idade usado provavelmente devido à sua ludicidade. Entretanto, o uso de ferramentas baseadas em Lego, kits de robóticas ou Arduinos estas estavam mais presentes em turmas do ensino médio. Entretanto o foco para ambos os níveis de ensino tem se tornado divergentes, onde foi constatado que a partir do ano de 2015 houve uma grande redução de experimentos voltados ao nível de ensino médio e um aumento do número de trabalhos direcionados ao ensino fundamental, onde grande parte dos trabalhos desenvolvidos no ensino médio foram motivados com o intuito de despertar interesses na área da computação, em contraste, os trabalhos que envolviam estudantes de nível fundamental, estes estavam mais interessados em desenvolver habilidades e pensamentos computacionais (SANTOS et al., 2018).

Outro exemplo de trabalho relacionado, a pesquisa conduzida pelos autores, Dayanne Oliveira et. al., é relatado uma experiência de oficina aplicada com programação em blocos. Utilizando a ferramenta App Inventor e Scratch, com o intuito de despertar jovens talentos para os cursos da área de computação. A oficina foi aplicada em turmas do segundo e terceiro ano do ensino médio, de quatro escolas distintas da região norte do Paraná (GARCIA et al., 2016). Em resumo, a oficina foi elaborada com duração de cinco semanas, cada qual, conteúdos de algoritmos e processos, fluxogramas, Scratch e AppInventor foram ministrados durante esse período. Como resultado concluíram que aproximadamente 63% alunos obtiveram aproveitamento superior à 70% em relação às atividades, enquanto 10% dos estudantes obtiveram um aproveitamento abaixo de 50% nas atividades. Por fim, após uma pesquisa de interesse em ingressar em cursos na área de computação, os pesquisadores relataram que 67,2% dos estudantes apresentaram interesses por cursos nesta área (GARCIA et al., 2016).

### **3.1. Considerações Finais**

Uma vez compreendido os fundamentos e aplicações dos trabalhos relacionados mencionados anteriormente, é válido concluir que tais trabalhos colaboraram em diversos aspectos do desenvolvimento e fundamento para esta pesquisa.

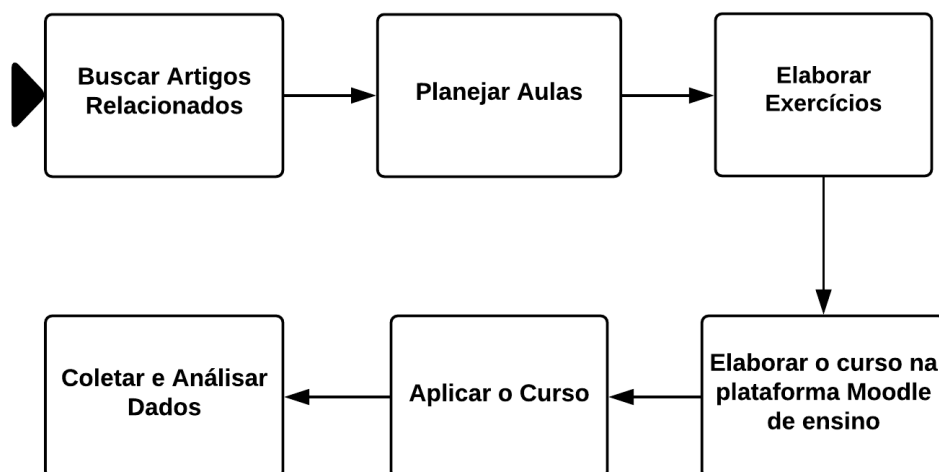
Por fim, este trabalho se difere no estado da arte por apresentar uma combinação das técnicas de aplicações utilizando conceitos do ensino à distância como vídeos aulas mas dando suporte aos participantes de forma presencial. Como também, se difere no que diz respeito às avaliações dos participantes, uma vez que foram combinados questionários e submissões para analisar o desempenho e motivação dos participantes no decorrer do curso.



## Metodologia

Neste tópico serão apresentados os passos estabelecidos para a elaboração e execução desta pesquisa. Na Seção 4.1 são apresentados os procedimentos realizados para a busca e seleção de artigos que compõe o material bibliográfico. Na Seção 4.2 é apresentado o planejamento de aula. Na Seção 4.3 é apresentada a elaboração dos exercícios e desafios. Na Seção 4.4 é apresentada a elaboração do curso na plataforma Moodle. Na Seção 4.5 é apresentada as informações para a aplicação desta pesquisa. Por fim, na Seção 4.6 é apresentada métodos para a análise dos dados.

O seguinte fluxograma apresenta os passos realizados para a elaboração e execução desta pesquisa apresentado pela Figura 4.1.



**Figura 4.1.** Apresentação do fluxograma das atividades desenvolvidas.

Uma vez apresentado o diagrama dos procedimentos, abaixo é descrito cada processo a ser realizado.

## 4.1. Busca e Leitura de Artigos Relacionados

Para a coleta e análise dos artigos relacionados, foram utilizadas as seguintes bases de dados: IEEE xplora, DBLP Computer Science Bibliography e Google Scholar. Para a expressão de busca foi utilizado a seguinte sentença: *((“Block Programming” OR “Unplugged Computing” OR “Block-Based Programming” OR “Scratch”) AND (K12 OR “K-12” OR Children OR Kids OR Novices OR student ) AND (“Software Engineering” OR “Computational Thinking”))*.

Tendo os artigos obtidos como resposta da expressão de pesquisa em mãos, os artigos selecionados seguem os critérios citados abaixo:

- Artigos que façam alusão aos métodos de programação em blocos visando promover o ensino/aprendizagem dos conceitos computacionais;
- Estudos que coletaram resultados de forma empírica ou seja, artigos que tenham aplicado o experimento em alguma turma de estudantes;
- Trabalhos que realizaram experimentos na forma presencial ou por meio de aulas assistidas pelo computador como no ensino à distância.

A pesquisa resultou em aproximadamente 770 artigos, os quais em sua maioria atendiam aos critérios de seleção. Sendo assim, muitos desses trabalhos foram utilizados como auxílio para o tópico de referencial teórico, de trabalhos relacionados e para o desenvolvimento das atividades.

## 4.2. Planejamento de Aula

O planejamento das videoaulas seguiu com base no artigo *Teaching Software Engineering Principles to K-12 Students: A MOOC on Scratch* (HERMANS; AIVALOGLOU, 2017). Em cada semana os educandos devem assistir em média três horas de videoaulas e responder à um total de vinte questionários de no máximo quatro questões.

Foi preparada uma sequência de 33 videoaulas, divididas em 6 semanas, cada uma tendo como objetivo, desenvolver um jogo no ambiente Scratch. Paralelamente o educando também estará desenvolvendo os conceitos de programação tais como: variáveis, laços de repetição e de condição, paralelização, operadores, coordenação e procedimentos. Foram também apresentados os conceitos de Engenharia de Software tais como, depuração, duplicação, métodos longos e nomeação de variáveis.

As Tabelas C.1, C.2 e C.3 no Apêndice C apresentam os conceitos que serão ministrados aos educandos no decorrer do curso. Dentre as quais, a Tabela C.1 refere-se aos conceitos apresentados durante as 1<sup>a</sup> e 2<sup>a</sup> semanas. A Tabela C.2 é referente aos conceitos apresentados no decorrer das 3<sup>a</sup> e 4<sup>a</sup> semanas. Por fim, a Tabela C.3 refere-se às 5<sup>a</sup> e 6<sup>a</sup> semanas. Por fim, na 6<sup>a</sup> semana, são revisados todos os conceitos apresentados nas semanas anteriores.

Para a elaboração das atividades apresentadas nos vídeos, foi escolhida a plataforma de desenvolvimento Scratch, por ser um ambiente virtual simples e atrativo. Além de ser indicada para iniciantes em programação, por conter elementos auto-descritivos e interações na forma de animação.

### 4.3. Elaboração dos Exercícios

As atividades avaliativas tais como, questionários e projetos, foram distribuídas após a conclusão de cada videoaula. Ao todo foram elaborados vinte e seis questionários os quais eram formados por quatro questões. As questões com o passar dos módulos, tem o seu grau de dificuldade aumentado, se tornando mais complexas e trabalhosas a cada semana.

As Tabelas B.1 e B.2 no Apêndice B apresentam o mapeamento dos conceitos tais como: condicionais, variáveis, laços, operadores, paralelização, procedimentos, coordenação, depuração, duplicação, método longo e nomeação. Ainda, nessas tabelas são apontados as questões que avaliam cada conceito.

Para as atividades de submissões, é requerida a entrega dos projetos elaborados na plataforma Scratch. Como auxílio aos estudantes, foram elaborados alguns modelos para as submissões das atividades. Estes modelos são apenas conjuntos de blocos prontos onde os estudantes devem desenvolvê-los para resolver o problema proposto. Também como forma de submissão é solicitado que os estudantes enviem suas respostas na forma textual, em geral, utilizadas para avaliar o conceitos de nomeação.

#### 4.3.1. Avaliações Motivacionais

O modelo utilizado para analisar o engajamento dos educandos no decorrer dos seis módulos semanais, foi o RIMMS (Pesquisa de Motivação de Materiais Reduzido, do inglês *Reduced Instructional Materials Motivation Survey*), que trata de uma versão reduzida do modelo original IMMS, sendo a forma reduzida contemplada com apenas doze afirmações, diferente do modelo original que contém trinta e seis afirmações.

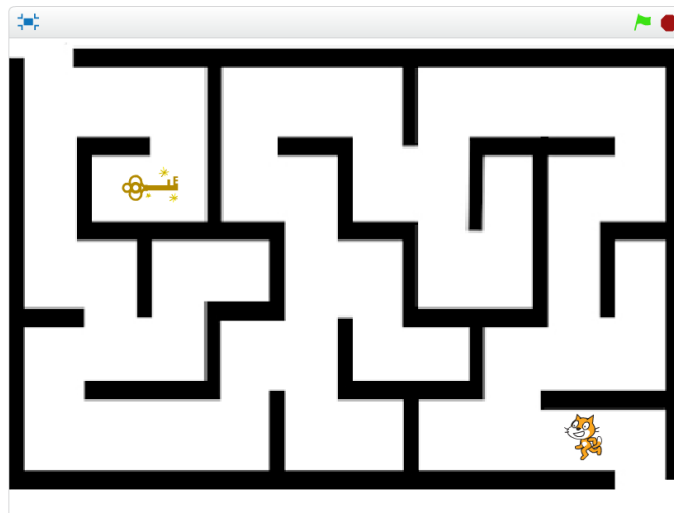
Sendo assim, este modelo foi disponibilizados ao final de cada módulo, permitindo o educando avaliar separadamente cada conjunto de aulas. Logo, o modelo IMMS pode ser encontrado no Apêndice A. Para compor o modelo RIMMS foram selecionadas as questões com os respectivos identificadores: 11A03, 17A06, 28A10, 06R01, 23R06, 33R09, 13C05, 25C07, 35C09, 14S02, 21S03 e 36S06, conforme apresentadas nas Tabelas A.1 e A.2.

Outra ferramenta utilizada foi o diário individual. O diário é apresentado como um campo textual, no qual cada estudante pode realizar comentários pessoais acerca das dificuldades encontradas. Em geral, o estudante poderá realizar qualquer comentário a respeito do seu envolvimento com os conteúdos lecionados nas aulas. Dessa forma, essa

ferramenta será utilizada para realizar observações quanto à motivação dos participantes pela análise de seus comentários, durante a elaboração das atividades em cada semana.

### 4.3.2. Desafios Propostos

Para a primeira semana, a atividade sugerida é a criação de um jogo onde um personagem tenta percorrer um labirinto sendo limitado pelas paredes. O objetivo do jogo é alcançar uma chave posicionada no cenário e após pegá-la o jogo é finalizado, tornando o personagem invisível para simbolizar que ele saiu do labirinto. A seguir, a Figura 4.2 apresenta os aspectos visuais do jogo a ser desenvolvido.



**Figura 4.2.** Exemplo da atividade elaborada na 1ª semana, labirinto na qual o jogador deve atravessar.

Na segunda semana, a atividade proposta é criar um jogo onde existem três atores representados por peixes. O primeiro peixe é controlado pelo jogador, o segundo tem movimentos independentes e representa a presa, e o terceiro peixe também tem seus movimentos independentes do jogador, mas agora representa o predador.

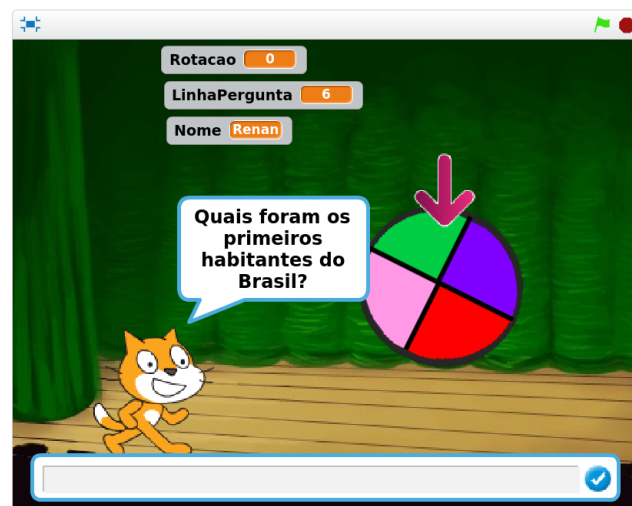
O objetivo do jogo é fugir do predador e comer o peixe menor. Quando isto acontecer o tamanho do peixe que o comeu é incrementado e o peixe comido tem seu tamanho decrementado. O jogo é finalizado quando o jogador conseguir comer o predador por apresentar um tamanho maior que o dele. A seguir, a Figura 4.3 apresenta os aspectos visuais do jogo desenvolvido.

Para a terceira semana foi elaborada uma atividade com o intuito de criar um jogo onde o jogador responderá perguntas aleatórias sobre um determinado assunto escolhido de acordo com a sua preferência. Após o jogador girar uma roleta disposta no cenário, é apresentado uma pergunta escolhida de forma aleatória de dentro de uma lista de perguntas. Após o jogador inserir sua resposta é necessário relacioná-la com uma lista de respostas, se o jogador acertar então a pergunta feita não deverá ser repetida. O jogo é finalizado quando



**Figura 4.3.** Exemplo da atividade elaborada na 2ª semana, cenário da caça e caçador.

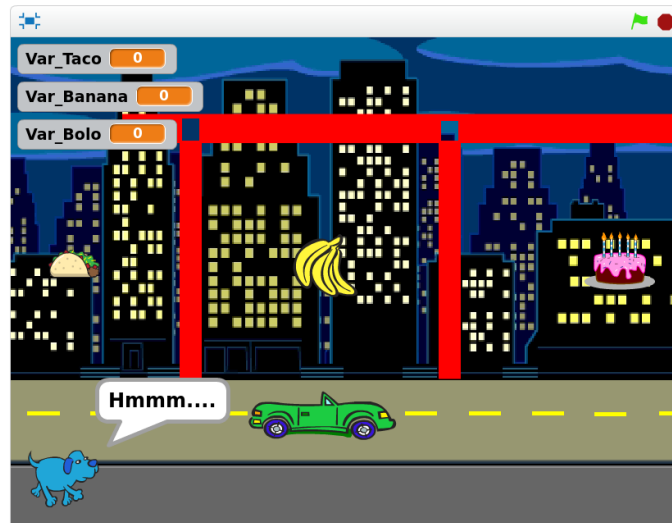
todas as perguntas da lista forem respondidas corretamente. A Figura 4.4 retrata o jogo solicitado como desafio.



**Figura 4.4.** Exemplo da atividade elaborada na 3ª semana, retratando o jogo no estilo perguntas e respostas.

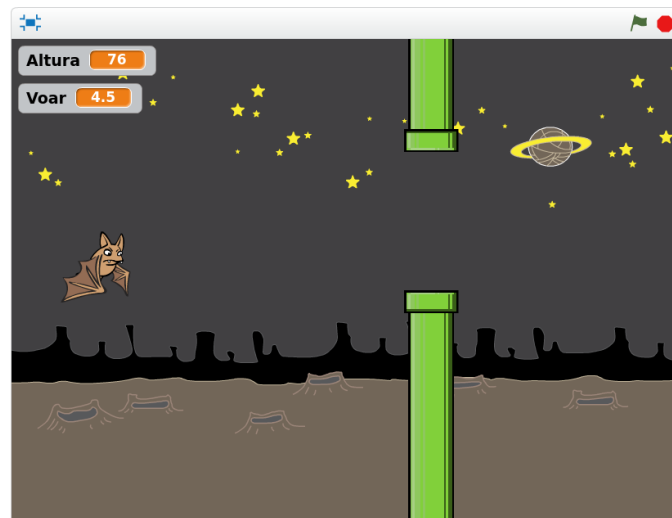
Na quarta semana foi elaborada uma atividade para a implementação de jogo onde o jogador irá controlar um cachorro para atravessar a rua. O jogador deve coletar os itens que estão posicionados no cenário. Como obstáculo foi utilizado um carro que tem movimentos repetitivos indo do canto esquerdo da tela para o canto direito, quando o cachorro é atingido pelo carro ele volta a ser posicionado no ponto inicial do cenário. O jogo é finalizado quando o jogador coleta todos os itens do cenário. Vale destacar que o jogo deve avisar o momento exato para atravessar a rua com base na distância do carro em relação com a distância do cachorro. A seguir, a Figura 4.5 retrata o jogo solicitado como desafio para a quarta semana.

Para a quinta semana foi elaborado um desafio para os alunos de desenvolver um jogo que assemelha-se ao jogo popular Flappy Bird, onde basicamente no cenário o jogador irá tentar desviar dos obstáculos que estarão em movimento contínuo em sua direção. Através da



**Figura 4.5.** Exemplo da atividade elaborada na 4ª semana, apresentado um jogo no estilo atravesse a rua.

tecla espaço o jogador poderá desviar dos obstáculos. Nesta semana também é apresentado aos alunos uma outra forma de controle de movimento do personagem, podendo também o personagem ser controlado pela câmera do computador. A seguir, a Figura 4.6 retrata o jogo solicitado como desafio para a quinta semana.



**Figura 4.6.** Exemplo da atividade elaborada na 5ª semana similar ao jogo Flappy Bird.

Na sexta e última semana é sugerido aos alunos o desenvolvimento de um jogo baseado no estilo aventura de plataforma. Assim como abordado pelo artigo publicado pela Felienne foi decidido um jogo simples, onde o jogador poderá se mover para frente e pular sobre obstáculos e inimigos, se por acaso o jogador tocar em um personagem inimigo e ele estiver em uma posição mais abaixo dele então o personagem sofrerá algum dano, caso contrário ele irá causar dano ao personagem inimigo. A seguir, a Figura 4.7 retrata o jogo a ser desenvolvido.



**Figura 4.7.** Exemplo da atividade elaborada na 6<sup>a</sup> semana, baseando-se em jogos no estilo plataforma.

## 4.4. Elaboração do Curso na Plataforma Moodle de Ensino

Após a elaboração dos vídeos e questionários, inicialmente a plataforma Udemy <sup>1</sup> foi utilizada para hospedar todo conteúdo desenvolvido, atingindo assim uma variedade de educandos. Entretanto, foi constatado que a plataforma Udemy não fornecia ferramentas para a análise do desempenho dos estudantes nos questionários elaborados, como também não permitia criar seções para a submissão dos desafios propostos.

Sendo assim, foi decidido que os vídeos e atividades, fossem submetidos à plataforma Moodle <sup>2</sup> de ensino. Pois esta plataforma fornece maior controle no gerenciamento dos conteúdos além de fornecer métodos para a submissão das atividades. Um fragmento do modelo elaborado para a primeira semana no Moodle é apresentado pela Figura 4.8. Os vídeos elaborados estão disponíveis para a visualização e podem ser acessados através da plataforma Google Drive <sup>3</sup>.

<sup>1</sup> Portal Udemy: <https://www.udemy.com/programacao-em-blocos-com-scratch>. Acesso em: 11 fev. 2019.

<sup>2</sup> Portal Moodle: <http://moodle.utfrpr.edu.br/course/view.php?id=9252>. Acesso em: 11 fev. 2019.

<sup>3</sup> Google Drive: [https://drive.google.com/open?id=1hqL6\\_aEszIbuyM\\_eG-gapSQcFLO0X-q9](https://drive.google.com/open?id=1hqL6_aEszIbuyM_eG-gapSQcFLO0X-q9). Acesso em: 11 fev. 2019

## 1º Semana

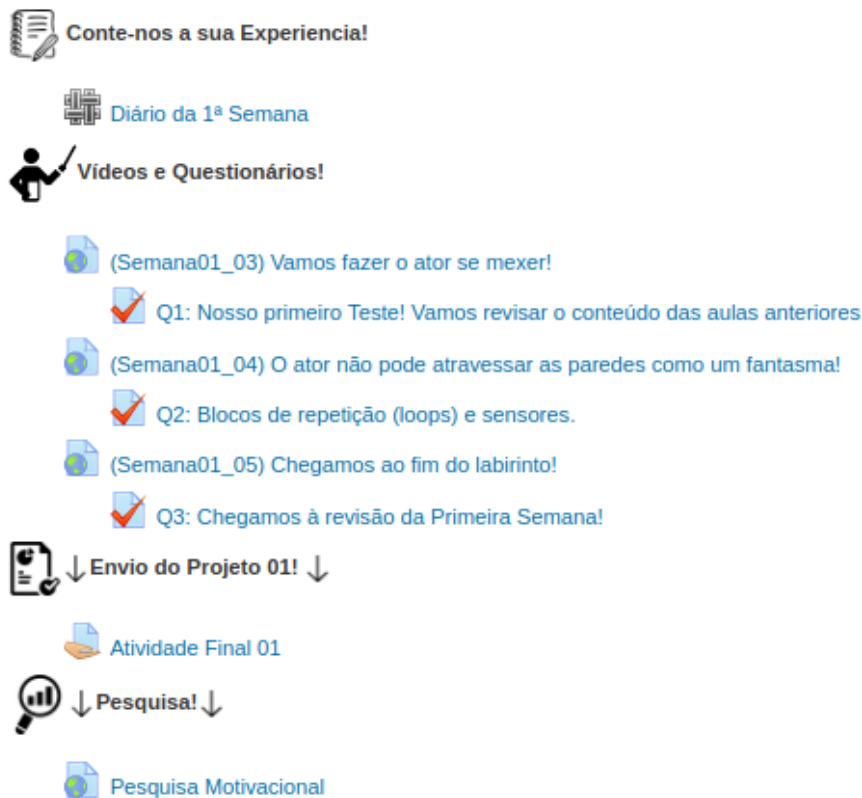


Figura 4.8. Fragmento do modelo estrutural da primeira semana.

## 4.5. Aplicação

As aplicações deste trabalho foram realizadas nos seguintes grupos, em suas respectivas disciplinas:

O Grupo 1 é composto pelos professores de física na disciplina de mestrado Atividades Computacionais para o Ensino Médio e Fundamental da UTFPR-CM (Universidade Federal do Paraná campus Campo Mourão), com o propósito de apresentar novos dispositivos tecnológicos que colaboram no processo de ensino dos conceitos envolvendo física em geral.

O Grupo 2 é formado pelos os estudantes do 1º ano do ensino médio com Técnico Integrado em Informática da UTFPR-CM, na disciplina de algoritmos. O Grupo 3 é composto pelos os estudantes do 1º período dos cursos de Engenharia de Alimentos e Engenharia Química na disciplina de programação da UTFPR-CM.

O Grupo 4 é composto pelos estudantes do 1º período do curso de Bacharelado em Ciência da Computação na disciplina de algoritmos da UTFPR-CM (Universidade Federal do Paraná campus Campo Mourão). Por fim, o Grupo 5 é formado pelos estudantes do curso de Engenharia Eletrônica na disciplina de algoritmos da UTFPR-CM.



### 4.5.1. Ciclo das Aulas

No primeiro encontro com os participantes, é apresentados à eles, uma breve introdução ao curso, explicando o objetivo, a estrutura das aulas e as atividades propostas. Sobre as aulas, é informado que o conteúdo será ministrado através de vídeos e que as atividades deverão ser submetidas no portal Moodle de ensino até a data estimada.

Como o curso é apresentado por meio de videoaulas, é comunicado aos participantes que, em caso de dúvidas, eles podem contatar o pesquisador em qualquer momento das atividades, seja por meio das redes sociais ou mensagens via e-mail. Sobre os próximos encontros, estes só ocorreriam se muitos participantes apresentassem as mesmas dúvidas. Caso apresentassem, um encontro seria agendado com o professor da disciplina, para posteriormente poder sanar as dúvidas da turma. Uma vez compreendidas as atividades e de comum acordo com os termos de consentimento, os participantes poderiam iniciar as atividades a qualquer momento.

## 4.6. Coleta e Análise dos Dados

Nesta fase, serão coletados os resultados obtidos através dos questionários e submissões enviados pelos participantes. Para responder a seguinte questão primária:

- Q1: Os participantes dos grupos de aprendizagem apresentaram bons desempenhos nas atividades envolvendo técnicas do pensamento computacional utilizando a linguagem Scratch?

A questão Q1 pode ser decomposta em outras três subquestões auxiliares, apresentadas a seguir:

- Q1.1: Os estudantes apresentaram resultados satisfatórios quanto aos conceitos de programação tais como: condicionais, variáveis, loops, operadores, paralelização, procedimentos e coordenação?
- Q1.2: Os estudantes apresentaram resultados satisfatórios quanto aos conceitos de engenharia de software tais como: depuração, duplicação, método longo e nomeação?
- Q1.3: Os estudantes mostraram-se motivados durante os módulos do curso?

Sendo assim, respondendo tais subquestões será respondido também a questão primária. Após as análises dos dados coletados, as subquestões Q1.1 e Q1.2, foram respondidas através das análises das respostas nos questionários e nas avaliações dos projetos. Cada questão foi mapeada para um conceito do pensamento computacional, conforme apresentado na metodologia.

Para a avaliação dos projetos foi utilizado a ferramenta Dr. Scratch <sup>4</sup> conforme

---

<sup>4</sup> Dr. Ssrcatch: <http://www.drscratch.org/>. Acesso em 12 de jul. de 2019

apresentado no trabalho *Dr. Scratch: a Web Tool to Automatically Evaluate Scratch Projects* (MORENO-LEÓN; ROBLES, 2015). A subquestão Q1.3 foi respondida através das análises dos dados obtidos em cada semana, por meio das avaliações motivacionais. Permitindo notar em qual semana os participantes ficaram mais ou menos motivados. Como no modelo RIMMS existem 12 afirmações separadas em 4 grupos, na qual cada grupo contém 3 afirmações, avaliadas em uma escala de 1 à 5, que representam os construtores do modelo ARCS, as avaliações motivacionais foram calculadas através da média das respostas em cada grupo.

Assim como também para responder a questão Q1.3, foram analisados os diários pessoais, na qual os participantes eram incentivados à postarem suas experiências e dificuldades. Buscando analisar o conteúdo dos comentários, foi utilizado a técnica de *World Clouds*, já que a natureza dos dados são na forma textual. Utilizando esta técnica para avaliar os diários, as palavras chaves mais utilizadas serão destacadas, podendo estas palavras apresentarem sentimentos relacionados à satisfação ou insatisfação no aprendizado.

Para a análise dos dados são consideradas as turmas que tiveram pelo menos 40% de concluintes em relação ao número de candidatos inscritos.

Como forma de incentivo, grande parte das turmas tiveram a oportunidade de obter recompensas como notas em suas disciplinas, por participarem do experimento, desde que respondessem à todas as atividades. Apenas o Grupo 1, composto pelos professores de física, não receberam nenhuma forma ou tipo de recompensa por participarem do experimento. Como uma outra forma de incentivo, para todos os questionários de múltipla escolha, os participantes receberam duas tentativas para respondê-las e entre as tentativas respondidas, a maior nota é considerada.

Por fim, as respostas dos diferentes grupos serão coletadas, estruturadas e analisadas para depois serem disponibilizados na plataforma Github <sup>5</sup>. As análises serão apresentadas por meio de visualizações gráficas, destacando a motivação e o desempenho dos participantes nos questionários e atividades entre as semanas. Para a visualização dos dados, serão utilizados os gráficos de colunas empilhadas, de combinação e de linhas, para indicar a distribuição das médias entre as notas.

---

<sup>5</sup> GitHub: [https://github.com/RenanKodama/Prog\\_Scratch](https://github.com/RenanKodama/Prog_Scratch). Acesso em: 18 jun. 2019

---

## Resultados e Discussões

---

Neste tópico serão apresentados os resultados obtidos após a aplicação deste experimento. Sendo assim, na Seção 5.1 é apresentado o processo de descarte dos resultados. Na qual, para as análises, foram considerados os participantes do Grupo 2 formado pelos estudantes do 1º ano do Ensino Médio com Técnico Integrado em Informática da UTFPR-CM, na disciplina de algoritmos. Na Seção 5.2 é apresentado o desempenho dos participantes do Grupo 2 nos questionários de múltiplas escolhas. Na Seção 5.3 é apresentado o desempenho do Grupo 2 em relação aos projetos desenvolvidos em Scratch, sendo avaliados pela ferramenta Dr. Scratch. Na Seção 5.4 é apresentado a motivação dos participantes durante as fases do experimento.

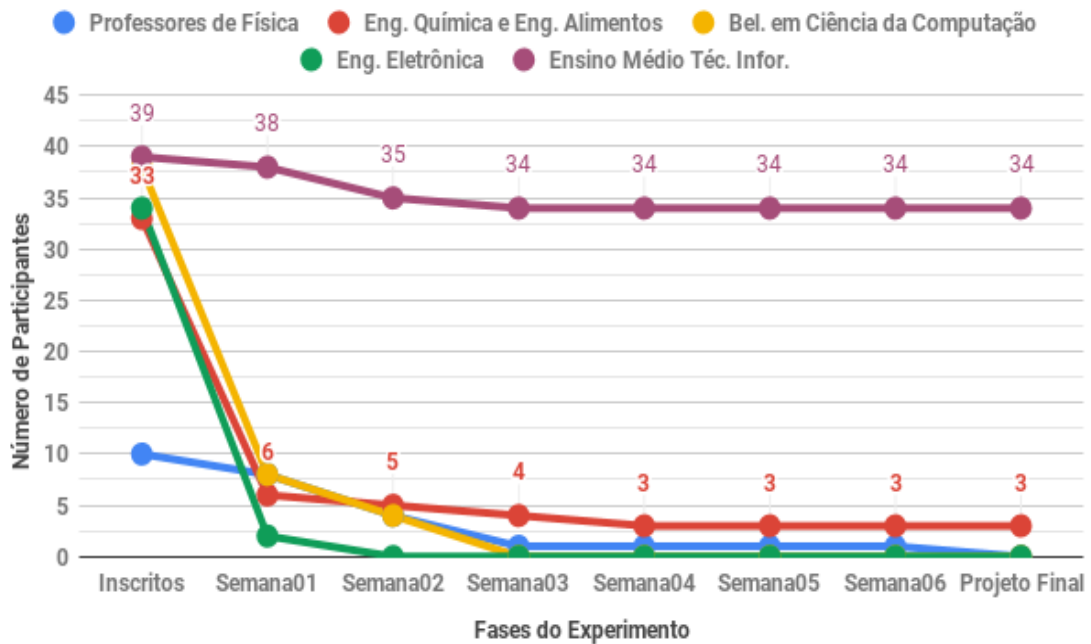
### 5.1. Dados Descartados

Considerando apenas as turmas que apresentaram 40% de concluintes em relação ao número de candidatos inscritos, como descrito na Seção 4.6, o Grupo 2 foi considerado para as análises. Sendo assim, os grupos 1, 3, 4 e 5, sendo respectivamente os grupos compostos pelos professores de física, estudantes do curso de Engenharia de Alimentos e Engenharia Química, estudantes do curso de Bacharelado em Ciência da Computação e estudantes do curso de Engenharia Eletrônica, foram descartados, pois não houveram resultados suficientes para a análise.

O gráfico de linhas, apresentado pela Figura 5.1, representa a quantidade de participantes ativos em cada semana, tendo como referência, o total de participantes inscritos. Considerando os grupos que foram descartados, muitos dos participantes não iniciaram as atividades da primeira semana, na qual o número de participantes ativos foi reduzido para uma faixa entre 2 à 8 participantes em cada turma.

No final, entre os grupos descartados, apenas um deles contendo 3 participantes haviam concluído todas as atividades.

## Fluxo dos Participantes



**Figura 5.1.** Gráfico de linhas representando a quantidade de participantes ativos por semana.

Sobre os mesmos grupos que foram descartados, também foram analisados os dados referentes à motivação dos participantes no experimento pelos formulários RIMMS (Pesquisa Reduzida de Motivação de Materiais Didáticos do inglês *Reduced Instructional Materials Motivation Survey IMMS*), buscando identificar se as desistências foram ocasionadas pela baixa motivação dos participantes durante o estudo ou não.

Sendo assim, o gráfico de colunas, apresentado pela Figura 5.2, representa a média da motivação entre os participantes dos grupos descartados em cada semana. Os construtores para o modelo IMMS podem ser encontrados na legenda do gráfico, sendo eles os construtores de atenção, relevância, confiança e satisfação. Para a legenda identificada como média, esta representa o valor médio entre os valores dos construtores, significando assim o grau geral de motivação.

Como observado pelo gráfico de colunas na Figura 5.2, os participantes dos grupos descartados, apresentaram um bom valor médio de satisfação nas semanas 1, 2, 4 e 5, aproximando-se do valor 4. Entretanto, nas semanas 3 e 6, o valor médio decaiu, aproximando-se do valor 3. Em uma escala Likert de resposta de 1 até 5, nas semanas 1, 2, 4 e 5 os participantes mantiveram-se parcialmente motivados, enquanto nas demais semanas os participantes estavam imparciais. Analisando a média dos 4 construtores entre as semanas, é observado que a proporção aproximada entre eles se mantém até a quinta semana.

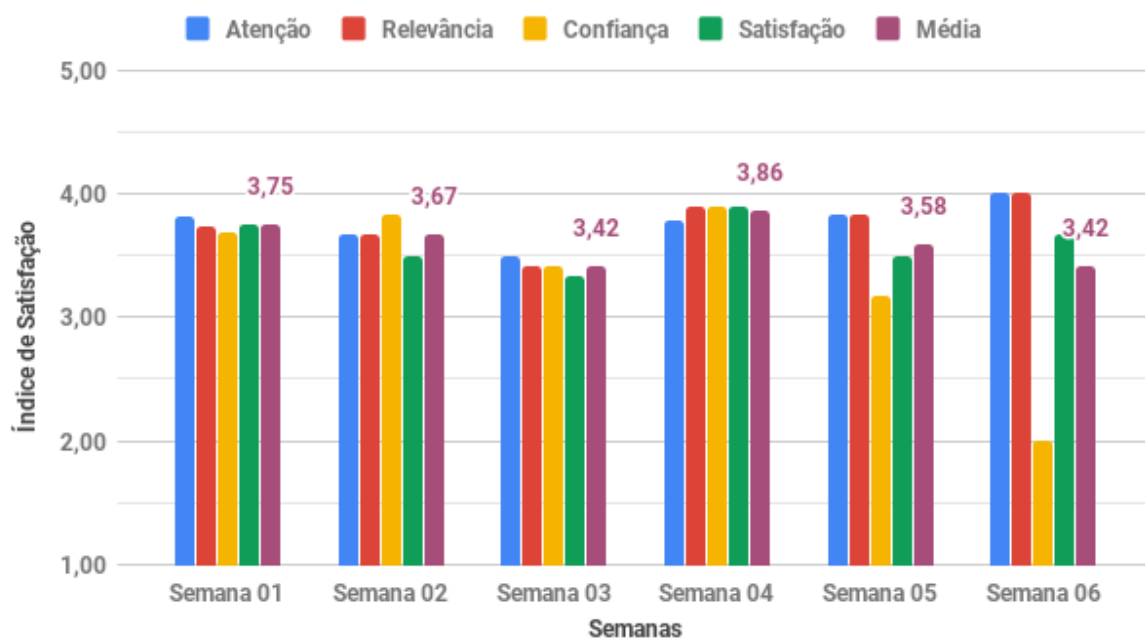
Na sexta semana, o valor médio do atributo de confiança atingiu o valor igual à 2, identificando que os participantes que realizaram as atividades da sexta semana, estavam parcialmente inseguros em terem aprendido ou para replicar as atividades.

Observando os valores das médias no gráfico da Figura 5.2 e o fluxo dos participantes dos grupos descartados no gráfico da Figura 5.1, é percebido que na semana anterior as desistências dos participantes, os valores médios de motivação, são conceituados como imparciais ou parcialmente motivados. Indicando que os participantes que desistiram, não estavam desmotivados antes de desistirem.

Logo, os questionários RIMMS, não detectaram que os participantes dos grupos descartados, ficaram totalmente desmotivados ao longo do experimento, podendo considerar as desistências relacionadas a outros fatores, que poderiam ser o acúmulo de atividades que os participantes teriam das disciplinas na qual frequentavam regularmente em seus cursos. Um dos indícios que reforça esta suposição, é que muitos participantes relataram se a data das entregas das atividades poderiam ser adiadas, pois haviam outras tarefas a serem realizadas.

Considerando os conteúdos abordados por meio de jogos, talvez esta forma de abordagem não tenha sido atrativa para os estudantes dos grupos do ensino superior que foram descartados, cuja a idade média destes grupos é maior quando comparada com a idade média dos estudantes do Ensino Médio com Técnico Integrado em Informática.

### Motivação dos Grupos Descartados



**Figura 5.2.** Gráfico de colunas representando os níveis de satisfação dos participantes por semana.

## 5.2. Análise do Desempenho nos Questionários

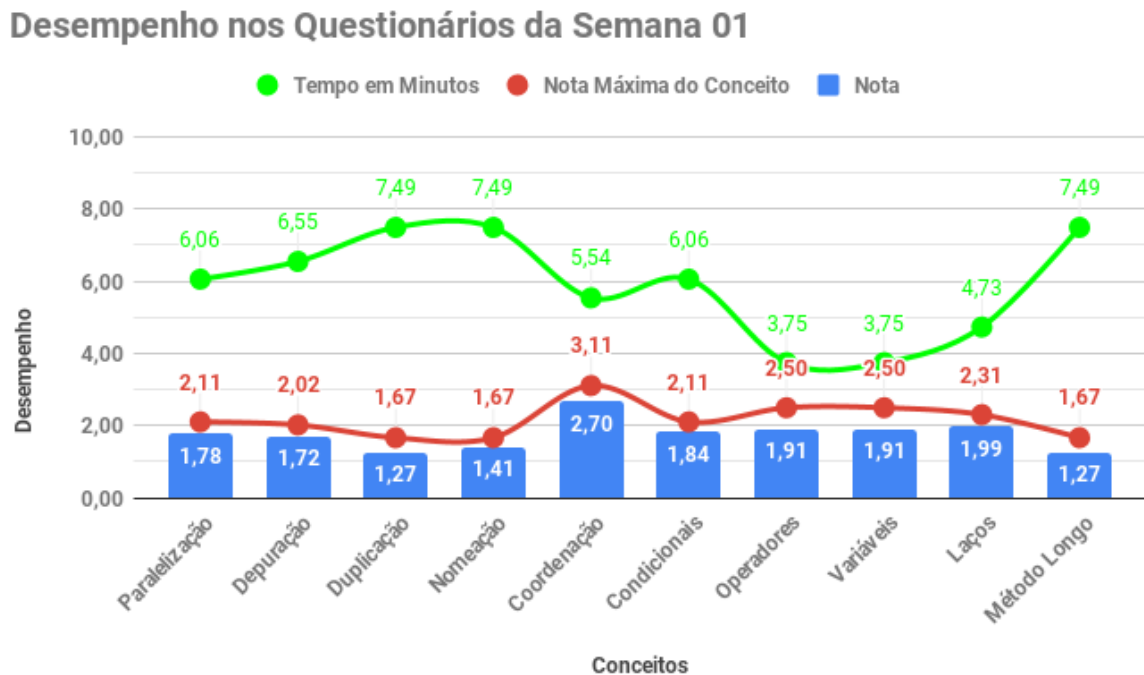
Após o processo de descarte dos dados, foram realizadas as análises de desempenho sobre o Grupo 2, em relação aos questionários de múltiplas escolhas. Para uma análise mais

profunda, foram consideradas as análises de cada semana separadamente. Para a geração das visualizações, foi utilizado a combinação do gráfico de colunas com o de linhas, como apresentado pelas Figuras 5.3, 5.4, 5.5, 5.6, 5.7, 5.8, 5.9 e 5.10.

Para estes gráficos, as abscissas representam os conceitos da Engenharia de Software e de programação, abordados nesta pesquisa. O eixo das ordenadas representam o valor médio dos identificadores. Os identificadores são compostos por tempo em minutos, nota máxima do conceito e nota.

O identificador tempo em minutos representa a média de tempo em minutos que os participantes levaram para responder às questões. O identificador nota máxima do conceito representa o valor máximo que é atribuído à cada questão relacionada a um conceito. O identificador nota representa a média entre as notas recebidas pelos participantes por conceito.

A seguir o gráfico apresentado na Figura 5.3, representa o desempenho do Grupo 2 nos questionários da primeira semana.



**Figura 5.3.** Gráfico de combinação representando o desempenho dos participantes do Grupo 2 na Semana 01.

Levando em consideração os identificadores de nota e nota máxima do conceito, pelo gráfico da Figura 5.3, o valores das notas aproximaram-se dos valores máximos das questões. Os valores máximos dos conceitos apresentaram variações, pois a quantidade de questões que avaliam cada conceito não mantiveram as mesmas proporções nos questionários nesta semana.

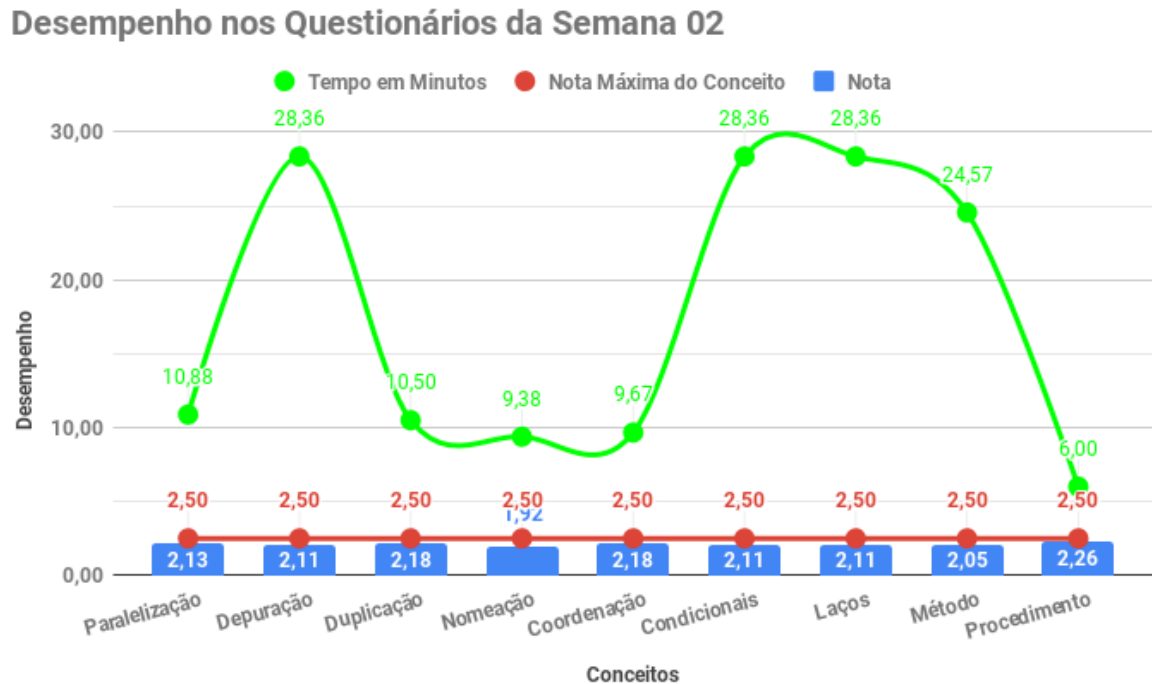
A diferença entre a nota e a nota máxima dos conceitos, obteve um valor médio de

0,40. Isto indica que os participantes receberam notas próximas aos valores máximos das questões. Observando a linha referente ao tempo em minutos, é observado que os participantes levaram mais tempo para resolver questões relacionadas aos conceitos de duplicação, nomeação e método longo. No total os participantes levaram em média 6 minutos para resolver os conceitos apresentados.

As análises do desempenho nos questionários da segunda semana é apresentado pelo gráfica da Figura 5.4. Como apresentado neste gráfico os participantes apresentaram uma melhora em relação entre a nota e a nota máxima dos conceitos. A diferença entre a nota e a nota máxima, obteve um valor médio de 0,38, sendo uma diferença um pouco menor do que apresentado na primeira semana.

Entretanto, para os valores de tempo gasto, os participantes levaram mais tempo comparados à primeira semana, onde o tempo médio foi de 17 minutos para resolver os exercícios apresentados. Como também é possível notar, os participantes levaram mais tempo para responder questões relacionados aos conceitos de depuração, de condicionais, de laços e de método longo.

Podendo ser explicado pois o nível de dificuldade das questões da segunda semana, eram maiores quando comparadas com as questões da semana anterior. É válido também considerarmos, que os participantes ainda estavam se habituando a plataforma e aos blocos de programação.



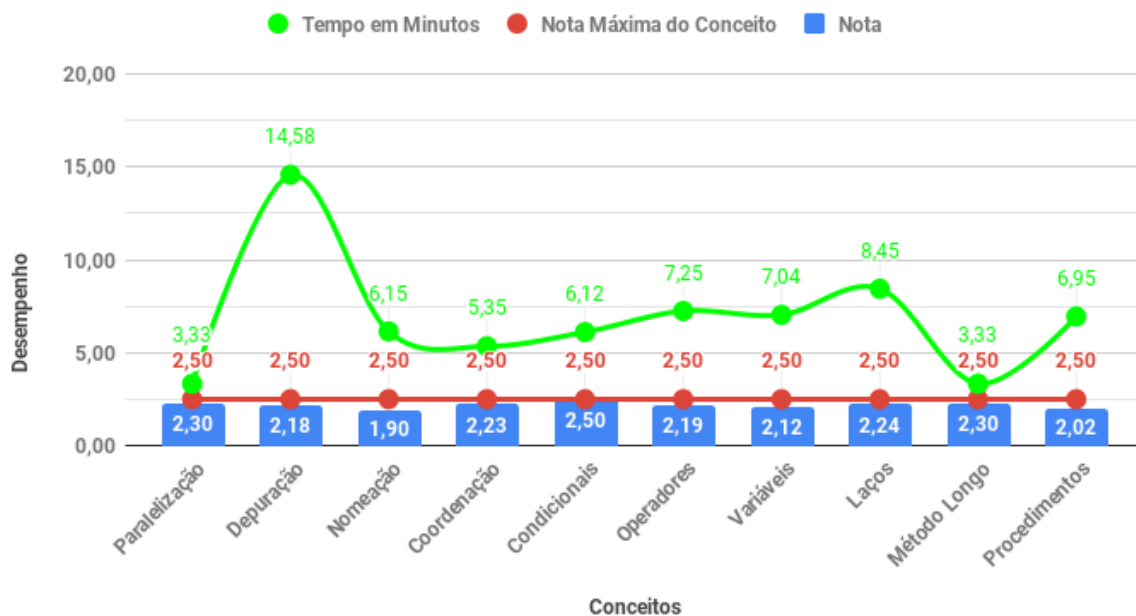
**Figura 5.4.** Gráfico de combinação representando o desempenho dos participantes do Grupo 2 na Semana 02.

O gráfico apresentado pela Figura 5.5 representa o desempenho do Grupo 2 nos questionários da terceira semana. Comparando as notas com as notas máximas, também é possível afirmar, que os participantes apresentaram notas satisfatórias em todos os conceitos, obtendo uma média das diferenças entre as notas e notas máximas de 0,30, sendo melhores que nas semanas anteriores.

Analisando a linha referente ao tempo gasto, os participantes levaram em média 7 minutos para resolver os questionários, sendo um tempo melhor do que apresentado na segunda semana. É percebido também que os participantes gastaram mais tempo resolvendo questões envolvendo conceitos de depuração, gastando em média 14,6 minutos.

Podendo ser explicado pois as atividades se tornaram mais complexas, principalmente no conceito envolvendo depuração. Na qual, os participantes passaram a analisar maiores trechos de códigos e a identificarem seu funcionamento.

### Desempenho nos Questionários da Semana 03



**Figura 5.5.** Gráfico de combinação representando o desempenho dos participantes do Grupo 2 na Semana 03.

A Figura 5.6 apresenta o gráfico de desempenho do Grupo 2 nos questionários da quarta semana. Observa-se que os participantes também apresentaram notas satisfatórias, onde a média das diferenças entre as notas e notas máximas dos conceitos, atingiu o valor de 0,35, se distanciando do valor da nota máxima quando comparado com a terceira semana.

Entretanto, na análise de tempo gasto, os participantes levaram em média 5,6 minutos para resolverem os questionários, sendo este, melhor do que apresentado na terceira semana.

Como observado, os participantes gastaram mais tempo para solucionar as

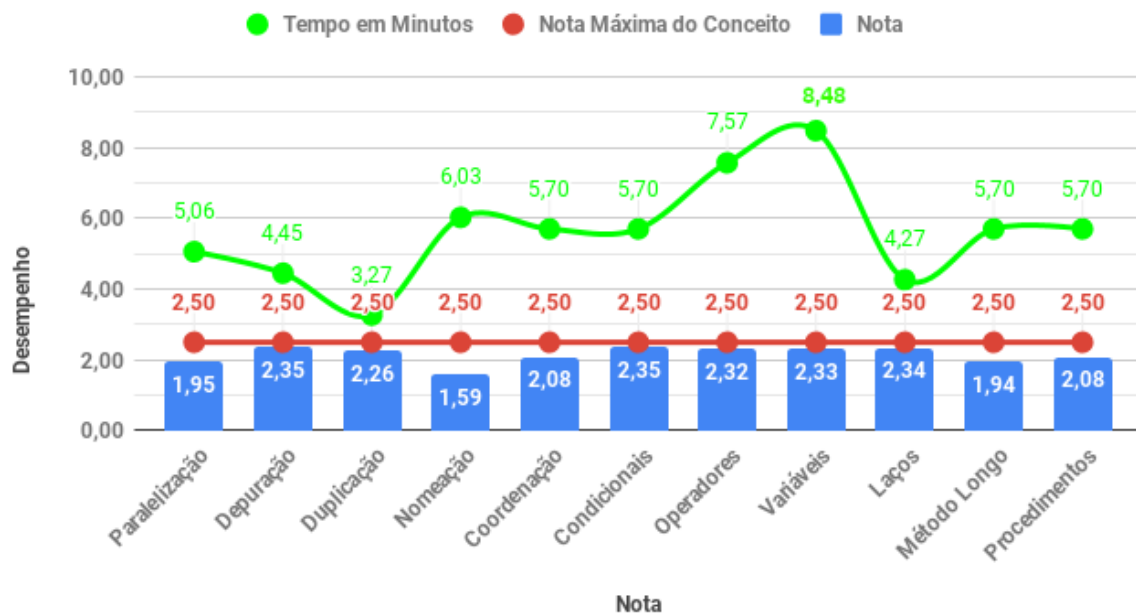


atividades envolvendo o conceito de variáveis. Podendo ser explicado pois, na quarta semana, as questões envolvendo este conceito, se tornaram mais complexas.

Na qual, era sugerido aos participantes que mencionassem pelo menos três possíveis nomes para as variáveis utilizadas nos trechos de códigos apresentados e que elas deveriam remeter à sua funcionalidade ou propósito no código.

Mesmo que os participantes tenham gasto um dos maiores tempos quando comparado a outros conceitos, a média das notas ainda assim foi conceituada como satisfatória.

### Desempenho nos Questionários da Semana 04



**Figura 5.6.** Gráfico de combinação representando o desempenho dos participantes do Grupo 2 na Semana 04.

O gráfico apresentado pela Figura 5.7, apresenta o desempenho do Grupo 2 nos questionários da quinta semana de experimento.

Assim como na primeira semana, os valores máximos dos conceitos apresentaram variações, pois, a quantidade de questões que avaliam cada conceito, não mantiveram as mesmas proporções nos questionários nesta semana.

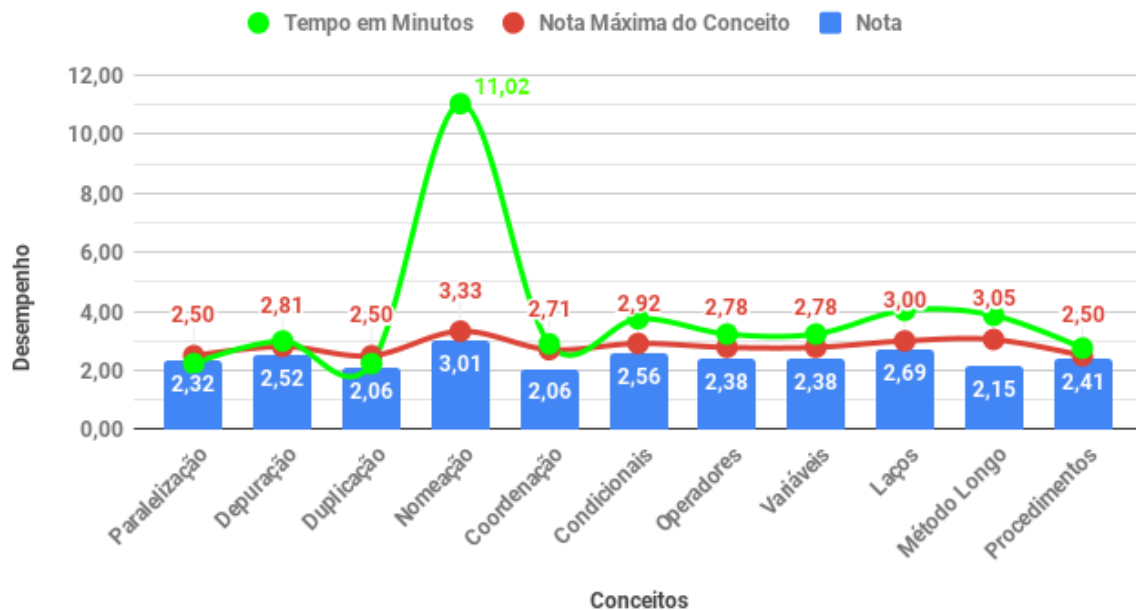
Como observado, os participantes apresentaram notas próximas aos limites máximos representado pelo identificador nota máxima do conceito. A média das diferenças entre as notas e as notas máximas se distanciaram, com o valor de 0,39, quando comparado com a quarta semana.

Entretanto, a média de tempo destacado pelo identificador tempo em minutos, obteve o valor médio de 3,78 minutos, sendo menor do que apresentado na quarta semana. Como observado, o valor de tempo que mais se destacou, é relacionado ao conceito de variáveis,

atingindo o valor de 11 minutos.

Podendo ser explicado, pois nesta semana, as atividades envolvendo este conceito, se tornaram mais complexas quando comparado com as semanas anteriores. Na qual, era solicitado que os participantes não só sugerissem nomes para as variáveis, mas como também, diferentes valores que as variáveis poderiam receber.

### Desempenho nos Questionários da Semana 05



**Figura 5.7.** Gráfico de combinação representando o desempenho dos participantes do Grupo 2 na Semana 05.

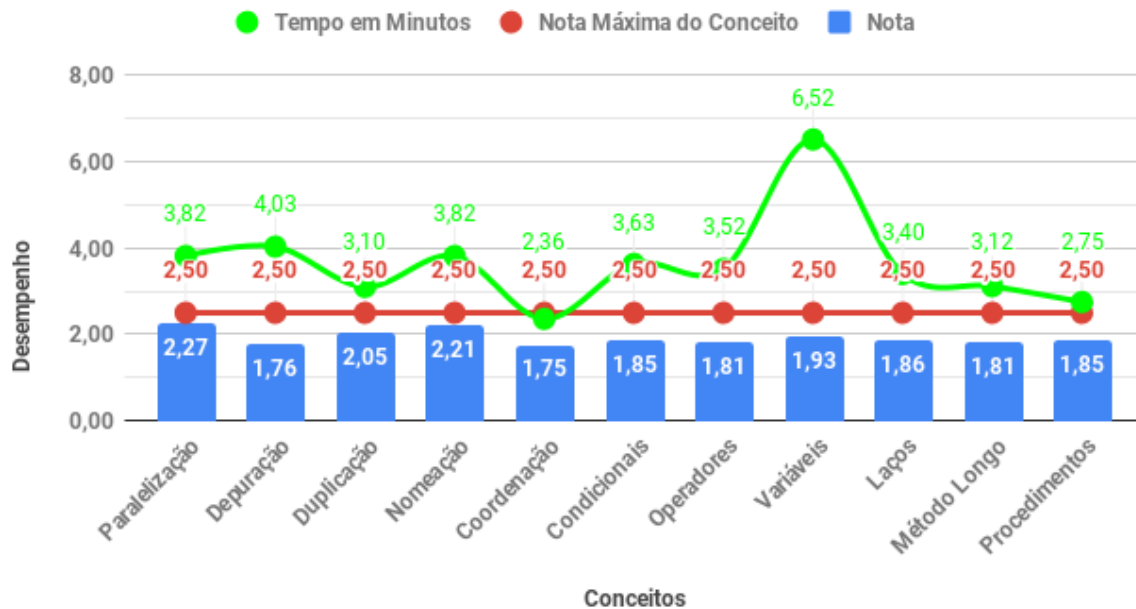
O desempenho do Grupo 2 nos questionários da sexta e última semana é apresentado pelo gráfico da Figura 5.8. Como observado neste gráfico, os participantes continuaram com as notas próximas as notas máximas em cada conceito. Entretanto, a média das diferenças entre as notas e notas máximas obteve o valor de 0,58, sendo a maior diferença entre todas as semanas. Podendo ser explicado pois, na última semana, são abordados todos os conceitos apresentados, entretanto com o grau de dificuldade aumentado.

Mesmo sendo a maior diferença entre as semanas, é possível conceituar o desempenho dos participantes como satisfatório. Se analisado o tempo médio gasto nos conceitos, os participantes levaram 3,6 minutos para responderem aos questionários, sendo o menor tempo gasto entre as semanas. É notado que os participantes gastaram mais tempo para resolver as questões envolvendo o conceito de variáveis, obtendo o valor de 6,52 minutos.

Mesmo que a dificuldade das questões tenha aumentado em relação a semana anterior, era esperado que o tempo de desenvolvimento das atividades para o conceito de variáveis, atingisse as mesmas proporções de tempo que outros conceitos obtiveram nesta semana. É

válido considerarmos, que os participantes também apresentaram valores altos de tempo nas semanas anteriores, indicando que eles tiveram certas dificuldades neste conceito. Talvez, o material elaborado para as videoaulas abordando este conceito, não tenha dado suporte adequado para a mesmo.

### Desempenho nos Questionários da Semana 06



**Figura 5.8.** Gráfico de combinação representando o desempenho dos participantes do Grupo 2 na Semana 06.

Para uma representação geral do desempenho do Grupo 2 nas semanas e nos conceitos dos questionários, foram elaborados os gráficos presentes nas Figuras 5.9 e 5.10.

Como observado no gráfico da Figura 5.9, os participantes apresentaram notas consideradas como satisfatórias durante o experimento, onde a média da diferença entre a nota e a nota máxima dos conceitos obteve o valor de 0,39, podendo ser conceituado como um desempenho satisfatório nas atividades.

Considerando os valores de tempo gasto, é possível observar que, no decorrer das semanas, os participantes passaram a gastar menos tempo para realizar as atividades, mesmo que a dificuldade dos questionários tenha aumentado no decorrer das semanas. Podendo deduzir, que os participantes passaram a ter maior facilidade para resolverem as atividades em relação às semanas anteriores.

Através do gráfico apresentado na Figura 5.10, é possível observar o desempenho geral do Grupo 02 nos questionários em relação à todos os conceitos abordados nesta pesquisa. A média da diferença entre as notas e as nota máximas obteve o valor de 0,41, representando, que os resultados dos participantes foram satisfatórios em todos os conceitos apresentados.

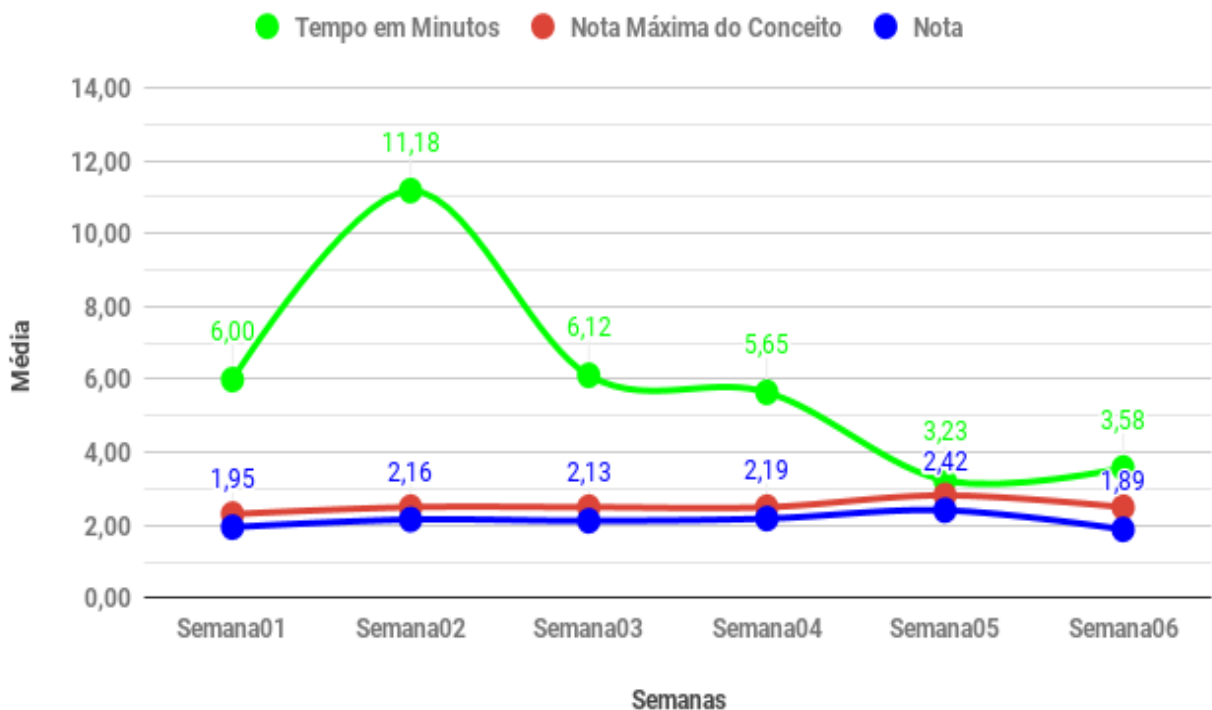
A linha referente ao tempo gasto, no gráfico da Figura 5.10, apresenta uma média de 5,2 minutos, sendo esta, a média de tempo gasto na resolução das questões em cada conceito.

Logo, os conceitos que apresentaram o valor de tempo acima da média, foram os conceitos de método longo, de nomeação, de operadores, de procedimentos e de variáveis. Na qual, representam os conceitos onde os participantes mais gastaram tempo, podendo ser explicado pela dificuldade de elaboração do problema ou entendimento da questão.

Observando os valores das notas pelo gráfico da Figura 5.10, é notado que as menores notas são apresentadas nos conceitos de nomeação e método longo. Indicando que os participantes apresentaram dificuldades em tais conceitos.

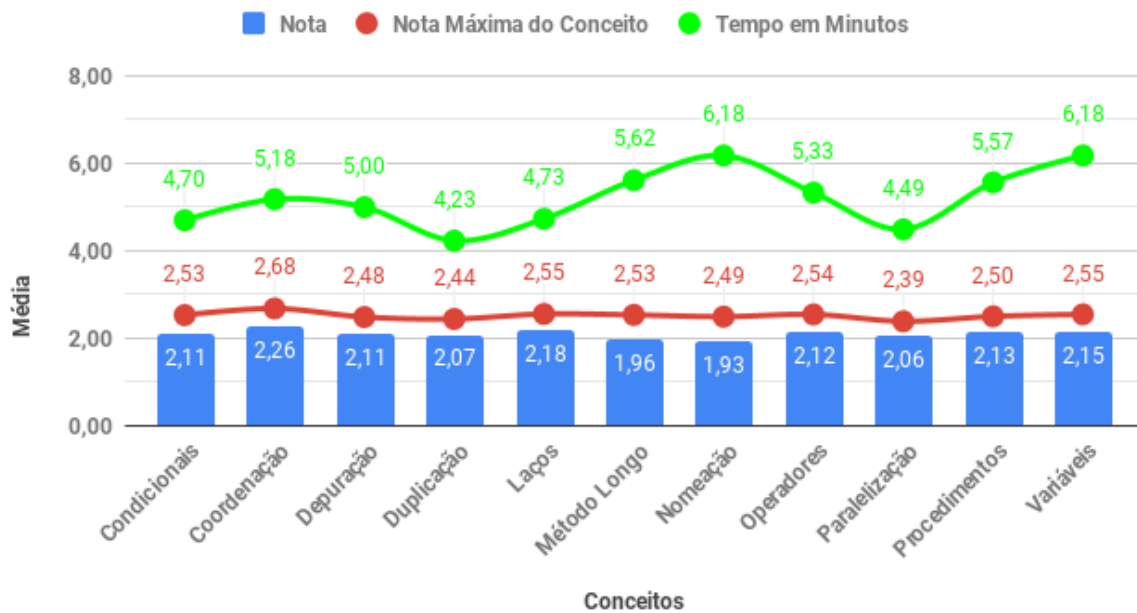
Em especial, para o conceito de nomeação, os participantes levaram mais tempo para resolver as atividades e obtiveram uma das menores notas quando comparada com os demais conceitos. Reforçando a suposição de que talvez o material elaborado, não tenha abordado de forma adequada este conceito.

## Desempenho Geral nos Questionários



**Figura 5.9.** Gráfico de linhas representando o desempenho geral do Grupo 2 nos questionários.

## Desempenho Geral nos Conceitos dos Questionários



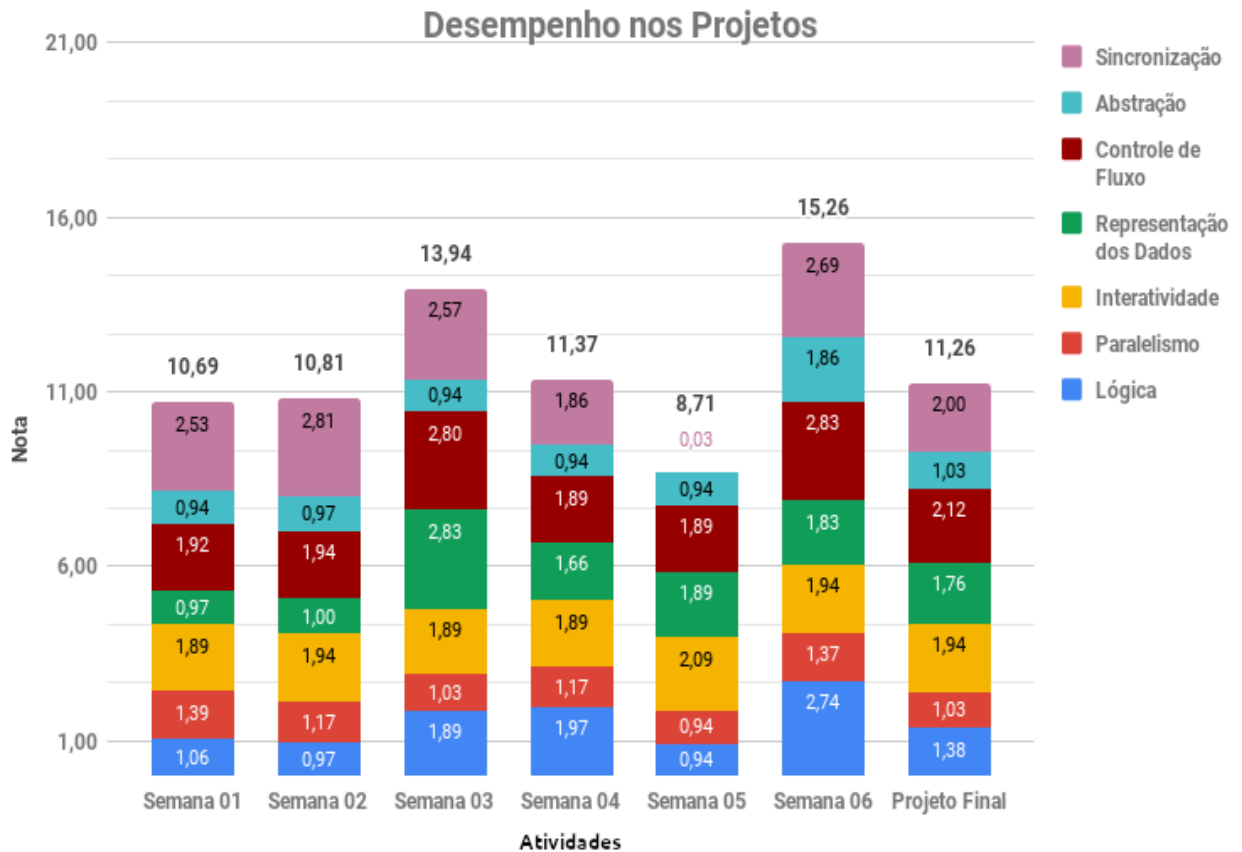
**Figura 5.10.** Gráfico de combinação. Desempenho geral do Grupo 2 nos conceitos dos questionários.

### 5.3. Análise do Desempenho nos Projetos

Para a análise dos projetos em Scratch, foi utilizado a ferramenta Dr. Scratch. Como mencionado na pesquisa realizada pelos autores Jesus Leon et. el., o Dr. Scratch analisa os conceitos básicos da Engenharia de Software, sendo eles, abstração, paralelismo, pensamento lógico, sincronização, controle de fluxo, interatividade com o usuário e representação de dados (MORENO-LEON et al., 2016). Para as avaliações dos projetos, são atribuídas notas em uma escala de 0 à 21, na qual projetos com notas de até 7 pontos são considerados simples, entre 8 à 14 pontos são considerados como satisfatórios e acima de 15 pontos são considerados como ótimos.

Para as avaliações dos conceitos, são atribuídas notas em uma escala de 0 à 3 a soma total deles representam a nota final do projeto, podendo atingir no máximo 21 pontos. Sendo assim, o gráfico de colunas empilhadas da Figura 5.11 representa o desempenho dos participantes em todos os projetos elaborados durante o estudo.

Para este gráfico, o eixo das abscissas representam os projetos elaborados em cada fase do experimento. O eixo das ordenadas representam as médias das notas que os participante receberam nos projetos em uma escala de 0 à 21 pontos. Para as legendas, os valores no topo de cada coluna, representam a nota total dos projetos compostos pela soma total de seus valores internos.



**Figura 5.11.** Gráfico de colunas empilhadas representando o desempenho geral do Grupo 2.

Como observado pelo gráfico da Figura 5.11, a média das notas nos projetos foram conceituadas como satisfatórias. Em especial na sexta semana, na qual, as notas dos participantes foram consideradas como ótimas.

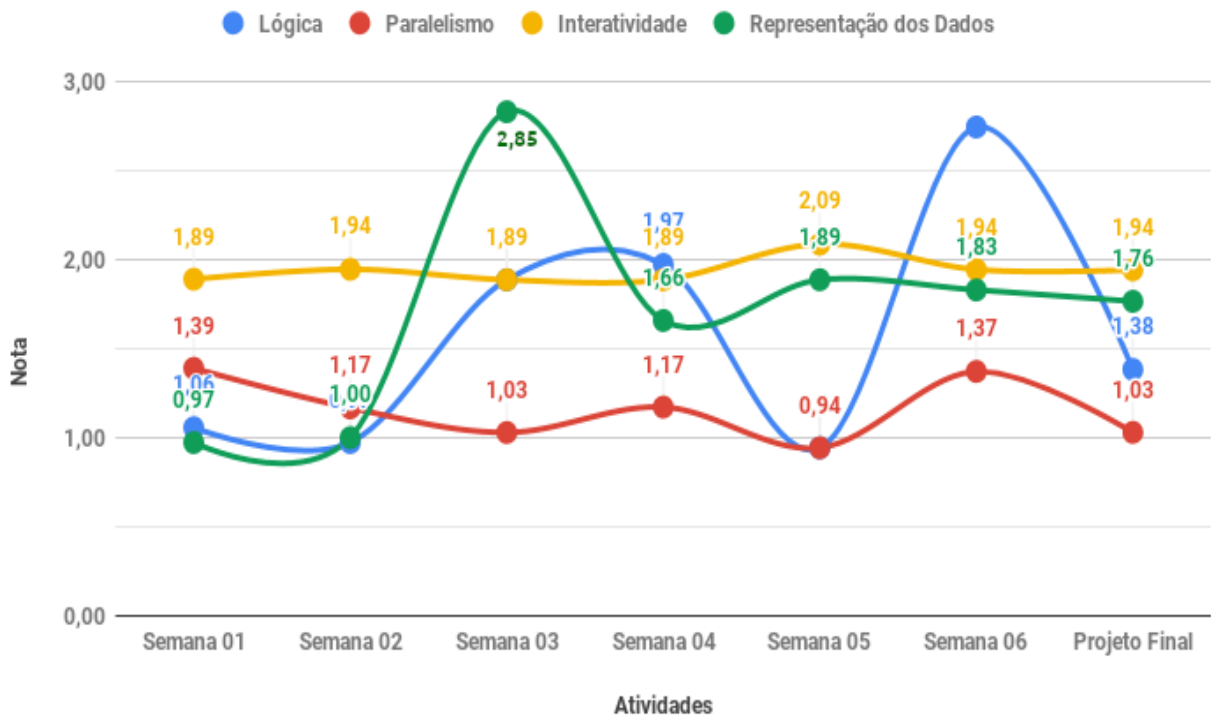
Para a visualização dos resultados referentes ao desempenho nos conceitos, foram gerados os gráficos de linhas nas Figuras 5.12 e 5.13. Juntos, os gráficos contém todos os conceitos avaliados pela ferramenta Dr. Scratch, entretanto, foram divididas em dois gráficos para evitar a sobreposição das linhas.

As abscissas, representam as fases do experimento onde ocorreram as submissões dos projetos. O eixo das ordenadas, representam a média das notas por conceito, na qual, pode-se considerar, que notas com o valor 0, são conceituadas como insatisfatórias, notas próximas à 1 são conceituadas como regulares, notas próximas à 2 são conceituadas satisfatórias e notas próximas à 3 são conceituadas como ótimas.

O gráfico da Figura 5.12 apresenta o desempenho dos participantes nos conceitos de lógica, paralelismo, interatividade e representação dos dados.

Observando o conceito de lógica, é notado, que as notas apresentaram grandes variações no decorrer das semanas, variando de 0,94 à 2,74 com uma diferença de 1,8. A média das notas neste conceito, obteve o valor de 1,56, indicando que os participantes apresentaram um desempenho regular.

## Fluxo de Desempenho nos Conceitos dos Projetos



**Figura 5.12.** Gráfico de linhas representando o fluxo do desempenho geral do Grupo 2 nos conceitos apresentados.

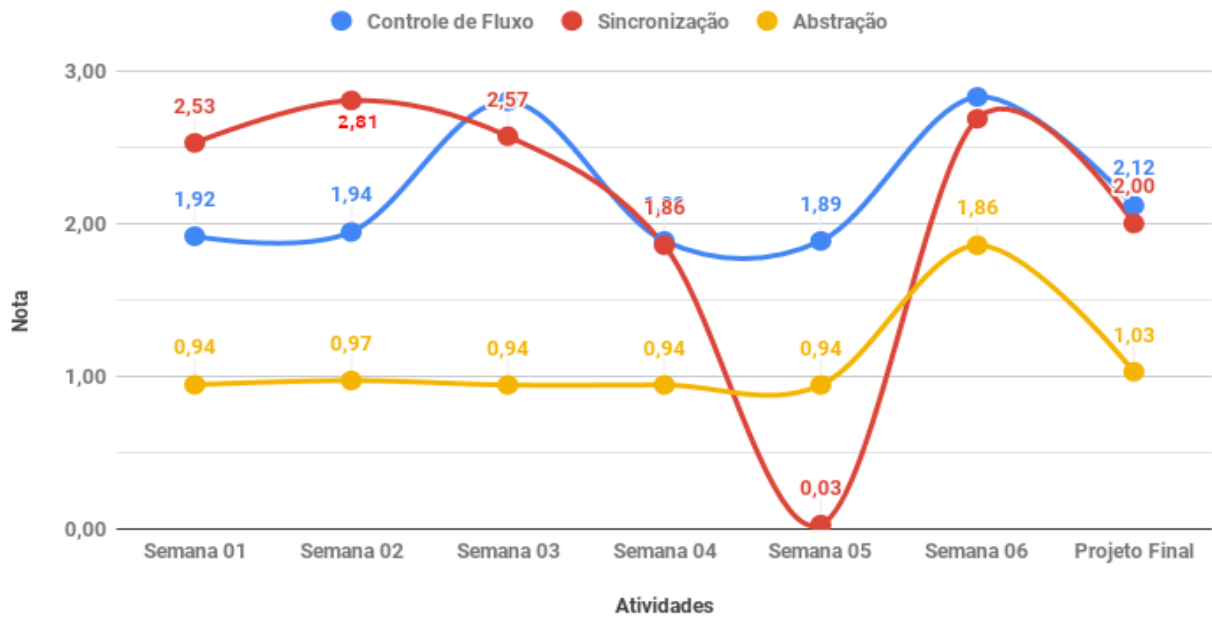
No conceito de paralelismo, é observado, que as notas não apresentaram grandes variações, variando de 0,94 à 1,39, com uma diferença de 0,45. A média das notas neste conceito, obteve o valor de 1,16, significando que os participantes apresentaram um desempenho regular.

Para o conceito de interatividade, é notado, que os participantes também não apresentaram grandes variações nas notas, variando de 1,89 à 2,09, com uma diferença de 0,2. A média de notas neste conceito obteve o valor de 1,94, podendo o desempenho, ser considerado como satisfatório.

No conceito de representação dos dados, os participantes apresentaram maiores notas na terceira semana, onde a média atingiu 2,83. Na qual, a variação ficou entre 0,97 à 2,83 com uma diferença de 1,86. A média entre as notas, obteve o valor de 1,7, indicando que os participantes apresentaram um desempenho satisfatório neste conceito.

Através do gráfico da Figura 5.13, é possível observar o desempenho dos participantes nos conceitos restantes, sendo eles, o controle de fluxo, a sincronização e a abstração.

### Fluxo de Desempenho nos Conceitos dos Projetos



**Figura 5.13.** Gráfico de linhas representando o fluxo do desempenho geral do Grupo 2 nos conceitos apresentados.

Analisando o conceito de controle de fluxo, pelo gráfico da Figura 5.13, é notado, que os participantes apresentaram desempenhos considerados como ótimos na terceira e sexta semana, obtendo a nota de 2,82. Nas demais semanas os participantes apresentaram desempenhos considerados como satisfatórios. As notas variaram entre 1,89 à 2,82 com uma diferença de 0,93. A média de nota para este conceito obteve o valor de 2,19, podendo considerar o desempenho dos participantes como satisfatório.

No conceito de sincronização, é percebido que os participantes apresentaram resultados considerados como insatisfatórios apenas na quinta semana, onde a média das notas atingiu o valor de 0,03. Enquanto nas demais semanas, os resultados apresentaram-se como ótimos ou satisfatórios. O intervalo de notas variou entre 2,8 à 0,03 com uma diferença de 2,77. O valor médio neste conceito obteve o valor de 2,07, sendo considerado no geral, o desempenho dos participantes como satisfatório. O baixo valor apresentado na quinta semana, pode ser explicado pois o projeto elaborado, poderia ser resolvido sem a utilização dos blocos de sincronização, resultando assim em uma nota inferior.

Para o conceito de abstração, é constatado que os participantes apresentaram resultados considerados como regulares nas semanas, exceto na sexta semana, onde os participantes apresentaram resultados considerados como satisfatórios. A variação das notas ficou entre 0,94 à 1,86 com uma diferença de 0,92. A média geral neste conceito, obteve o valor de 1,09, sendo o desempenho dos participantes considerados como regulares.

Logo, podemos concluir, que a média de desempenho dos participantes nos conceitos



avaliados, variaram entre regular e satisfatório. Na qual, para os conceitos avaliados como regular, estão os conceitos de abstração, de lógica e de paralelismo.

Para os conceitos avaliados como satisfatório estão os conceitos de sincronização, de controle de fluxo, de representação dos dados e de interatividade com o usuário. Se considerarmos as notas finais de cada projeto, é notável, que predominantemente, o desempenho dos participantes foi classificado como satisfatório. Podendo assim, conceituarmos que os participantes apresentaram um desempenho satisfatório nos projetos.

## 5.4. Análise do Desempenho Motivacional

A análise motivacional do Grupo 2 é apresentado pelo gráfico de colunas da Figura 5.14. Para a geração deste gráfico, foram utilizados os dados referentes aos questionários RIMMS (Pesquisa Reduzida de Motivação de Materiais Didáticos do inglês *Reduced Instructional Materials Motivation Survey IMMS*).

Neste gráfico, o eixo das abscissas representam as semanas onde ocorreram as avaliações RIMMS. O eixo das ordenadas representam a média de satisfação dos participantes.

Utilizando a escala Likert para interpretar os valores das resposta de 1 até 5, temos , o valor 1 que representa os participantes totalmente desmotivados e o valor 5 que representa os participantes totalmente motivados.

Para as legendas estão os construtores do modelo RIMMS. A legenda identificada como média, representa o valor médio entre os construtores do modelo RIMMS, na qual, simboliza o grau de motivação geral.

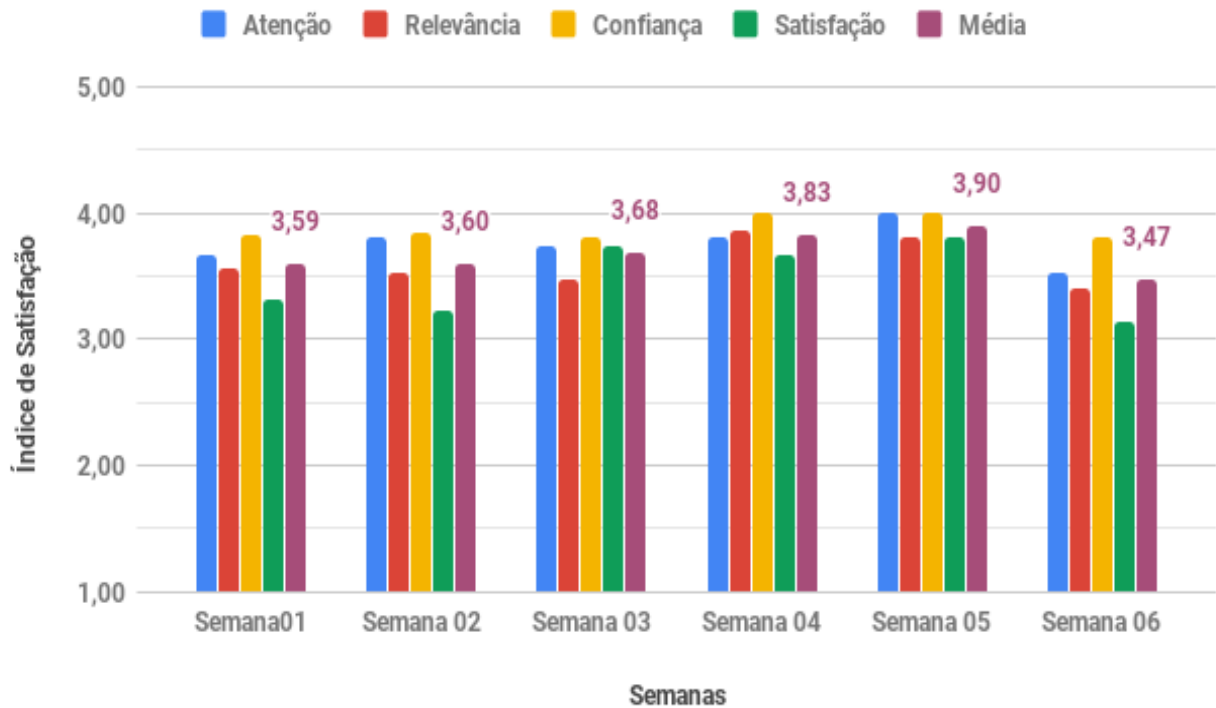
Como observado no gráfico da Figura 5.14, a média dos valores aproximaram-se de 4. Na qual, pode-se considerar que, os participantes estavam parcialmente motivados na atividade das semanas de 1 à 5. Entretanto na última semana, o valor médio de motivação, ficou próximo à 3, indicando que os participantes, estavam imparciais.

Analisando os valores dos construtores RIMMS, também é notável, que eles mantiveram proporções crescentes até a quinta semana. Enquanto na sexta sexta, os valores dos construtores decaíram em comparação com a semana anterior, na qual a diferença foi de 0,43.

Diferente dos grupos descartados, o atributo de confiança se destacou na sexta semana, sendo o valor mais alto entre os outros atributos, representando que os participantes, estavam confiantes em aprender ou replicar as atividades.

Entretanto para o atributo de satisfação, os participantes, demonstraram-se indiferentes em terem participado da lição ou se gostaram dela, tanto na primeira, quanto na segunda e sexta semana. A média geral de motivação entre as semanas, obteve o valor de 3,67, indicando que os participantes demonstraram-se parcialmente motivados durante o experimento.

## Motivação dos Participantes



**Figura 5.14.** Gráfico de colunas representando a motivação do Grupo 2 nas semanas de 1 à 5.

Para a análise do conteúdo nos diários, foram considerados os dados referentes aos relatos em cada semana. Para esta análise, os comentários foram avaliados manualmente, buscando classificá-los em comentários positivos, neutros ou negativos acerca da experiência de participação no estudo.

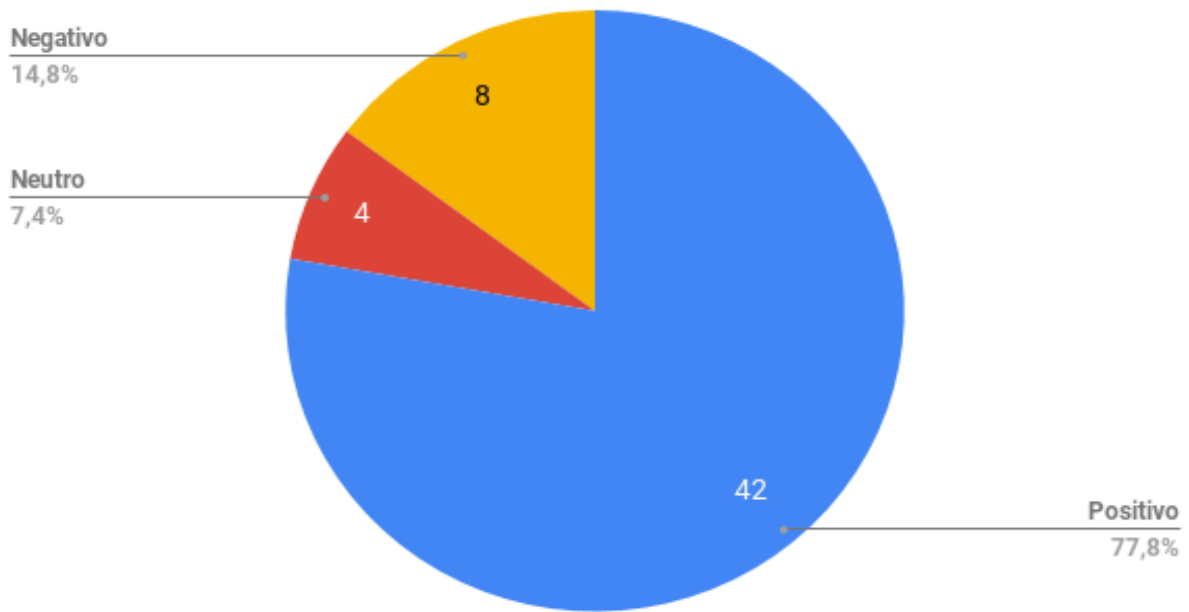
No gráfico de setores da Figura 5.15 é apresentado a proporção dos comentários classificados em positivos, neutros ou negativos. Como observado, ao todo foram realizados 54 comentários, sobre como os participantes se sentiram em cada semana.

Grande parte dos comentários foram conceituados como positivo, representando 77,8% de todos comentários realizados pelo Grupo 02. Reforçando a ideia, de que os participantes mantiveram-se motivados durante o experimento.

Entende-se por comentários positivos, aqueles que apresentaram qualquer sentimento relacionado à satisfação. Para os comentários considerados como neutros, estes, receberam esta classificação, por não ser identificado nenhum tipo de satisfação ou insatisfação.

Os comentários classificados como negativos, nestes, foram encontrados sentimentos relacionados à insatisfação.

## Classificação dos Relatos nos Diários



**Figura 5.15.** Gráfico de setores representando a proporção dos comentários positivos, neutros e negativos.

Buscando visualizar o conteúdo dos comentários, a imagem da Figura 5.16 foi gerada utilizando a técnica de *Word Clouds*. Esta técnica, consiste em apresentar as principais palavras que mais foram utilizadas nos comentários, sendo destacadas pelo tamanho. Em geral, as palavras em destaque podem representar sentimentos de satisfação ou insatisfação, e através de seu tamanho é possível analisar a proporção de ocorrências quando comparado à outras palavras.

Para uma geração mais precisa da imagem, os dados dos diários passaram por filtros, removendo palavras que não identificavam qualquer tipo de sentimento, como por exemplo, foram removidas as conjunções (e, nem, mas também, ora, logo, se, etc), para não se sobrepor a outras palavras de maior interesse.

Como observado na Figura 5.16, as palavras de interesse que mais se destacaram foram as palavras, divertido, aprendi, simples, fácil, consegui, ajudou, diferente e gostei. Indicando, que estes termos, apresentaram-se em maior quantidade nos comentários dos diários. Em geral, grande parte das palavras em destaque, apresentaram sentimentos relacionados à satisfação, durante as fases do experimento.

Também foi possível observar palavras que caracterizaram sentimentos de insatisfação, sendo as elas, difícil e dificultando. Entretanto, como o tamanho visual das palavras em comparação com as palavras de sentimentos positivos, é menor, podemos então afirmar que os sentimentos de satisfação e motivação prevaleceram durante este experimento. Confirmando assim as observações motivacionais anteriores, de que os participantes permaneceram motivados



---

## Conclusão

---

Após as análises dos resultados obtidos no Capítulo 5, pode-se constatar, que os participantes apresentaram resultados satisfatórios tanto nos projetos implementados quanto nos questionários respondidos. Na qual, as avaliações ocorreram sobre os principais conceitos da engenharia de software e programação, sendo eles, os conceitos de condicionais, de coordenação, de depuração, de duplicação, de laços, de método longo, de nomeação, de operadores, de paralelização, de procedimentos e de variáveis, totalizando em 11 conceitos.

Também foi possível constatar, que os participantes apresentaram-se motivados em toda a fase do experimento. Até mesmo os grupos que foram descartados, na qual era esperado que os valores dos questionários RIMMS (Pesquisa de Motivação de Materiais Reduzido, do inglês *Reduced Instructional Materials Motivation Survey*), para este grupo atingissem valores abaixo do nível de satisfação classificado como indiferente. Entretanto os valores obtidos, nos questionários apresentaram-se em grande maioria como parcialmente motivados, podendo indicar que as desistências ocorreram à outros fatores externos.

De uma forma geral, os participantes que realizaram as atividades apresentaram notas satisfatórias em todos os conceitos e permaneceram motivados durante as fases do experimento. Respondendo assim, a questão principal deste estudo. Como observado, a programação em blocos com Scratch pode trazer benefícios tanto no ensino quanto no aprendizado, para os conceitos de programação e Engenharia de Software. Podendo se tornar uma ferramenta importante para novos métodos de ensino, buscando a aprimoração dos modelos atuais. Entretanto, novos estudos devem ser realizados para comparar a programação em blocos com outras abordagens para melhor determinar a importância deste paradigma no ensino de conceitos de Engenharia de Software e programação.

# Apêndices

## Lista de afirmações do modelo IMMS

**Tabela A.1.** Afirmações para a análise da motivação de acordo com modelo IMMS.

Identificador	Afirmação
1C011	Quando olhei pela primeira vez para esta lição, tive a impressão de que seria fácil para mim.
02A01	Havia algo interessante no começo desta lição que chamou minha atenção.
03C02	Esse material era mais difícil de entender do que eu gostaria que fosse.
04C03	Depois de ler as informações introdutórias, senti-me confiante de que sabia o que deveria aprender com essa lição.
05S01	Completar os exercícios desta lição me deu um sentimento satisfatório de realização.
*06R01	É claro para mim como o conteúdo deste material está relacionado a coisas que eu já conheço.
07C04	Muitas das páginas tinham tantas informações que era difícil escolher e lembrar os pontos importantes.
08A02	Estes materiais são atraentes.
09R02	Havia histórias, fotos ou exemplos que me mostraram como esse material pode ser importante para algumas pessoas.
10R03	Completar esta lição com sucesso foi importante para mim.
*11A03	A qualidade da escrita ajudou a prender minha atenção.
12A04	Essa lição é tão abstrata que era difícil manter minha atenção nela.
*13C05	Enquanto trabalhava nesta lição, estava confiante de que poderia aprender o conteúdo.
*14S02	Gostei tanto desta lição que gostaria de saber mais sobre esse assunto.
15A05	As páginas desta lição parecem secas e desagradáveis.
16R04	O conteúdo deste material é relevante para os meus interesses.

**Tabela A.2.** Continuação da tabela de afirmações para a análise da motivação pelo modelo IMMS.

<b>Identificador</b>	<b>Afirmação</b>
*17A06	A maneira como as informações são organizadas nas páginas ajudou a manter minha atenção.
18R05	Existem explicações ou exemplos de como as pessoas usam o conhecimento nesta lição.
19C06	Os exercícios desta lição foram muito difíceis.
20A07	Esta lição tem coisas que estimularam minha curiosidade.
*21S03	Gostei muito de estudar esta lição.
22A08	A quantidade de repetições nesta lição me fez ficar entediado às vezes.
*23R06	O conteúdo e o estilo de escrever nesta lição transmitem a impressão de que vale a pena conhecer seu conteúdo.
24A09	Eu aprendi algumas coisas que foram surpreendentes ou inesperadas.
*25C07	Depois de trabalhar nesta lição por um tempo, eu estava confiante de que eu seria capaz de passar por um teste nela.
26R07	Esta lição não foi relevante para as minhas necessidades porque eu já sabia a maior parte dela.
27S04	A redação do feedback após os exercícios, ou de outros comentários nesta lição, ajudou-me a sentir-me recompensado pelo meu esforço.
*28A10	A variedade de passagens de leitura, exercícios, ilustrações, etc., ajudou a manter a atenção na aula.
29A11	O estilo de escrever é entediante.
30R08	Consegui relacionar o conteúdo desta lição com coisas que vi, fiz ou pensei em minha própria vida.
31A12	Há tantas palavras em cada página que é irritante.
32S05	Foi bom concluir com êxito esta lição.
*33R09	O conteúdo desta lição será útil para mim.
34C08	Eu não conseguia entender muito do material desta lição.
*35C09	A boa organização do conteúdo me ajudou a ter certeza de que aprenderia esse material.
*36S06	Foi um prazer trabalhar em uma aula tão bem projetada.



## Mapeamento das avaliações e Abordagens

**Tabela B.1.** Mapeamento dos conteúdos abordados por tópicos e apresentação da forma de aplicação.

Tópicos	Número do Questionário e Questão	Abordagem
Condicionais	2_3, 3_3, 6_1, 10_2, 11_1, 11_2, 12_2, 13_2, 17_1, 17_2, 18_1, 18_2, 19_1, 19_2, 20_1, 20_2, 21_1, 21_2, 22_1, 22_2, 23_1, 23_2.	Perguntamos aos estudantes qual bloco é utilizado para sabermos se determinada resposta está correta ou se alguma ação está acontecendo.
Variáveis	2_2, 7_1, 7_2, 8_2, 9_1, 9_2, 10_1, 10_3, 10_4, 11_1, 11_2, 13_2, 16_1, 18_1, 18_2, 21_2, 22_1, 22_2	Perguntamos aos educandos qual a diferença entre uma variável e uma lista. Também é pedido que criem scripts para guardar o nome do jogador e preencher as listas de perguntas e respostas.
Laços	2_1, 2_2, 2_3, 3_2, 6_1, 9_1, 9_2, 10_1, 11_1, 11_2, 12_1, 12_2, 13_2, 16_1, 17_1, 17_2, 18_1, 18_2, 19_1, 19_2, 20_1, 20_2, 22_1, 22_2, 23_1, 23_2	Verificamos se os educandos sabem a quantidade de vezes que o bloco irá executar e qual será o valor ao final da execução.
Operadores	2_1, 2_2, 2_3, 3_2, 6_1, 9_1, 9_2, 10_1, 11_1, 11_2, 12_1, 12_2, 13_2, 16_1, 17_1, 17_2, 18_1, 18_2, 19_1, 19_2, 20_1, 20_2, 22_1, 22_2, 23_1, 23_2	Apresentamos aos educandos operações matemáticas utilizando operadores, como também sequências de expressões concatenados pela conjunção lógica e perguntamos a validade da resposta.
Paralelização	2_1, 2_4, 3_4, 3_6, 5A_1, 5A_2, 8_3, 8_4, 11_3, 12_3, 13_4, 15_3, 18_4, 19_1, 20_3, 22_3	Perguntamos aos educandos qual bloco é utilizado para aguardar as execuções e sua funcionalidade. Também é perguntado à eles quais blocos irão funcionar juntos quando o jogo for iniciado.

**Tabela B.2.** Continuação dos mapeamento dos conteúdos abordados por tópicos.

<b>Tópicos</b>	<b>Número do Questionário e Questão</b>	<b>Abordagem</b>
Procedimentos	4_1, 4_2, 4_3, 5_1, 5_2, 5_3, 5A_4, 6_1, 6_3, 7_3, 7_4, 8_1, 9_4, 10_2, 10_4, 11_3, 11_4, 12_2, 13_4, 18_4, 19_1, 19_2, 20_2	Pedimos aos educandos que selecione o bloco mais adequado para a resolução do problema, assim como também pedimos à eles que desenvolvam uma atividade utilizando os blocos apresentados
Coordenação	1_1, 1_2, 3_1, 3_2, 3_6, 4_1, 5_1, 6_2, 6_4, 8_3, 8_4, 9_2, 9_4, 10_1, 11_3, 11_4, 12_2, 13_4, 15_3, 17_3, 18_1, 18_2, 23_1, 23_2	Apresentamos blocos que utilizam o envio e recebimento de mensagens e perguntamos quais blocos estão ativados e qual será o resultado
Depuração	2_2, 2_3, 3_1, 3_2, 3_3, 6_1, 7_1, 9_1, 9_2, 11_1, 11_2, 12_1, 12_2, 13_2, 14_2, 15_1, 15_2, 16_1, 17_1, 17_2, 18_1, 18_2, 18_4, 19_2, 20_1, 21_1, 21_2, 22_1, 22_2, 23_1, 23_2	Apresentamos aos estudantes blocos com funções elaboradas e perguntamos à eles o que este bloco está fazendo, assim como também, pedimos para que eles encontrem os erros no código, sendo tais erros inseridos de forma proposital
Duplicação	3_6, 5_4, 5A_3, 6_2, 11_2, 12_3, 12_4, 13_3, 15_3, 18_3, 19_1, 19_4, 21_4, 22_3, 23_3, 23_4	Pedimos para os educandos que identifiquem os blocos que estão realizando a mesma tarefa
Método Longo	3_6, 5A_4, 6_2, 6_3, 6_4, 8_3, 8_4, 11_3, 12_3, 11_4, 13_4, 15_4, 16_3, 17_3, 19_1, 19_2, 21_3, 22_3, 23_4	Apresentamos aos educandos blocos únicos com excesso de funções e pedimos para que eles submetam uma nova versão mais simplificada com os blocos separados
Nomeação	3_5, 4_4, 6_4, 7_3, 7_4, 8_3, 8_4, 9_3, 9_4, 10_3, 10_4, 13_4, 14_3, 14_4, 16_2, 19_3, 20_4, 22_4	Apresentamos aos estudantes trechos de código que utilizam identificadores imprecisos para mensagens, variáveis e listas. Pedimos para que eles submetam um texto com outros possíveis nomes mais interessantes para as variáveis

## Mapemanto dos Conceitos Lecionados

**Tabela C.1.** Mapeamento dos conceitos abordados em tópicos referentes às 1<sup>o</sup> e 2<sup>o</sup> semanas.

Tópicos	Material Apresentado	Nº do Vídeo
<b>Semana 01</b>		
Condicionais	Bloco se-então é apresentado aos estudantes, assim como seu funcionamento e aplicações	4, 5
Laços	Apresentação do Bloco sempre aos estudantes, e explicação de sua contínua execução	4, 5
Paralelização	Criação dos blocos de movimentação do personagem e a explicação de execuções simultâneas	5
Coordenação	Apresentação do conceito de mensagens e sua criação para a coordenação com outros blocos de programação	5
Depuração	Antes da execução dos blocos, é realizado a análise para prever o possível resultado dos blocos.	4, 5
<b>Semana 02</b>		
Condicionais	Revisão do funcionamento do Bloco se-então	4
Laços	Revisão sobre os conceitos do Bloco sempre	3, 4, 5
Paralelização	Criação dos movimentos independentes nos personagem inimigos e introdução do Bloco espere	2, 3
Coordenação	Coordenação por mensagens entre Blocos para esconder as Sprites que são tocadas por peixes maiores	4, 6
Depuração	Apresentação de erros propositalis na movimentação dos personagens inimigos, conduzindo assim o estudante à solução apropriada	4, 6
Duplicação	Apresentação de exemplos de códigos duplicados e mencionamos que queremos minimizá -los	4, 5
Método Longo	Apresentação de Blocos sobrecarregados por funções, explicando que não é uma boa prática e que queremos separá -los	6

**Tabela C.2.** Continuação dos mapeamentos dos conteúdos, sendo apresentadas as 3<sup>o</sup> e 4<sup>o</sup> semanas.

<b>Tópicos</b>	<b>Material Apresentado</b>	<b>Nº do Vídeo</b>
<b>Semana 03</b>		
Condicionais	Apresentação e explicação de uso do Bloco se-então-senão	4, 5
Variáveis	Introdução dos conceitos e diferenças entre variáveis e listas	2, 3, 4
Laços	Introdução e explicação do Bloco repita-até-que	2, 3
Operadores	Apresentação e explicação dos Blocos números aleatórios e operador de igualdade (=)	3
Coordenação	Coordenação entre Blocos por meio de mensagens criadas, minimizando métodos longos	3, 4
Nomeação	Apresentação de boas práticas de nomeação de variáveis, listas e mensagens aos estudantes	2, 3, 4
<b>Semana 04</b>		
Condicionais	Reforço dos conceitos e utilização dos blocos de controle se-então e se-então-senão	3, 4
Variáveis	Criação de variáveis para o controle do fluxo do jogo e para armazenar o nome do jogador melhorando assim aspectos de interação	4, 5
Laços	Revisão dos conceitos sobre laços e utilização do bloco de repetição sempre	2, 3, 4
Operadores	Apresentação do uso e funcionamentos dos operadores maior (>) e menor (<)	4, 5
Paralelização	Reforço dos conceitos de paralelização através da movimentação contínua dos obstáculos no cenário de forma independente	2, 3
Coordenação	Coordenação com outros blocos para esconder objetos do cenário quando ocorrer alguma ação através de mensagens enviadas	3, 4, 5
Depuração	Após a elaboração de cada conjunto de blocos é verificado se a lógica desenvolvida atende aos resultados antes de sua execução	4, 5
Método Longo	Apresentação dos conjuntos de blocos com muitos métodos e soluções para minimizá -los	4, 5
Nomeação	Revisão de boas práticas de nomeação de variáveis, listas e mensagens	4, 5

**Tabela C.3.** Mapeamento final dos conteúdos, sendo apresentadas as 5<sup>o</sup> e 6<sup>o</sup> semanas.

<b>Tópicos</b>	<b>Material Apresentado</b>	<b>Nº do Vídeo</b>
<b>Semana 05</b>		
Condicionais	Reforço dos conceitos e uso do bloco de controle se-então	4, 5
Variáveis	Apresentação de variáveis para controle de ações do personagem principal	2, 3
Laços	Revisão dos conceitos dos blocos de repetição repita-até-que e sempre	2, 3, 4
Operadores	Introdução do operador de conjunção lógica (AND) e revisão dos operadores maior (>), menor (<) e igual (=)	3, 4
Procedimentos	Apresentação e criação de blocos com funcionalidades específicas para a movimentação do personagem e ações de movimento controladas pela câmera	3, 5
Depuração	Inserção de problemas propositais que afetam a movimentação do personagem, guiando assim o aluno à solução adequada	3, 4, 5
Duplicação	Apresentação de formas para minimizar duplicações de blocos, por meio da coordenação por mensagens	4, 5
Método Longo	Apresentação de blocos sobrecarregados de métodos informando que estes não são uma boa prática	4, 5
<b>Semana 06</b>		
Condicionais	Revisão de todos os blocos condicionais apresentados tais como: se-então e se-então-senão	1, 2, 3
Variáveis	Novamente será apresentado os conceitos e utilização de variáveis para o controle da direção do personagem inimigo	2, 3, 4
Laços	Revisão de todos os blocos de repetição apresentados tais como: repita-até-que e sempre	1, 2, 3
Operadores	Revisão dos operadores apresentados tais como: maior (>), menor (<), igualdade (=) e conjunção lógica (AND)	2, 3, 4
Paralelização	Revisão dos conceitos de paralelização aplicados na contínua movimentação independente do cenário e dos inimigos	5, 6
Procedimentos	Novamente é apresentado aos educandos a criação de blocos próprios para minimizar a duplicação e métodos longos	1, 2
Coordenação	Revisão dos conceitos através da troca de mensagens entre blocos para determinar quando um personagem deve esconder ou atacar	3, 4, 5
Depuração	Novamente falhas são inseridas no desenvolvimento do jogo, conduzindo assim o educando à possíveis soluções	3, 6
Nomeação	Novamente apresentamos boas práticas para nomeação de variáveis, escolhendo o nome de acordo com sua funcionalidade	2, 3, 4, 6

# Referências

---

- ALIMISIS, Dimitris. *Themes in Science and Technology Education*. Trípoli, Greece: Athanasios Jimoyiannis e Tassos A. Mikropoulos, 2013. v. 6. 63-71 p.
- BRACKMANN, Christian Puhmann. Desenvolvimento do pensamento computacional através de atividades desplugadas na educação básica. p. 226, 2017. Disponível em: <<http://hdl.handle.net/10183/172208>>.
- BRASIL, Ministério da Educação. *Diretrizes Curriculares Nacionais da Educação Básica*. [S.l.: s.n.], 2013. 562 p. ISBN 978-857783-136-4.
- BUNDY, Alan. Computational Thinking Is Pervasive. *Journal of Scientific and Practical Computing Noted Reviews*, v. 1, n. 2, p. 67–69, 2007. Disponível em: <<http://www.inf.ed.ac.uk/research/programmes/comp-think/>>.
- EGUCHI, Amy. What is educational robotics? theories behind it and practical implementation. Association for the Advancement of Computing in Education (AACE), San Diego, CA, USA, p. 406–414, March 2010. Disponível em: <<https://www.learntechlib.org/p/34007>>.
- GARCIA, Victor Ronchi; SILVA, Dayanne Oliveira da; OLIVEIRA, Igor Goulart de; TRINDADE, Daniela De Freitas Guilhermino; SGARBI, Ederson Marcos; NASCIMENTO, Luiz Fernando Legore do. Despertando Jovens Talentos com o Conhecimento da Computação. n. Cbie, p. 583, 2016. ISSN 2316-6541. Disponível em: <<http://www.br-ie.org/pub/index.php/wie/article/view/6865>>.
- HERMANS, Feliene; AIVALOGLU, Efthimia. Teaching software engineering principles to K-12 students: A MOOC on scratch. In: *Proceedings - 2017 IEEE/ACM 39th International Conference on Software Engineering: Software Engineering and Education Track, ICSE-SEET 2017*. [s.n.], 2017. p. 13–22. ISBN 9781538626719. Disponível em: <<https://www.edx.org/course/scratch->>.
- LOORBACH, Nicole; PETERS, Oscar; KARREMAN, Joyce; STEEHOUDER, Michaël. Validation of the Instructional Materials Motivation Survey (IMMS) in a self-directed instructional setting aimed at working with technology. *British Journal of Educational Technology*, v. 46, n. 1, p. 204–218, 2015. ISSN 14678535.
- MORENO-LEÓN, Jesús; ROBLES, Gregorio. Dr. Scratch: a Web Tool to Automatically Evaluate Scratch Projects. *Proceedings of the Workshop in Primary and Secondary Computing Education on ZZZ - WiPSCE '15*, p. 132–133, 2015. Disponível em: <<http://dl.acm.org/citation.cfm?doid=2818314.2818338>>.
- MORENO-LEON, Jesus; ROBLES, Gregorio; ROMAN-GONZALEZ, Marcos. Comparing computational thinking development assessment scores with software complexity metrics.

*IEEE Global Engineering Education Conference, EDUCON*, v. 10-13-April-2016, n. April, p. 1040–1045, 2016. ISSN 21659567.

RESNICK, Mitchel; MALONEY, John; MONROY-HERNÁNDEZ, Andrés; RUSK, Natalie; EASTMOND, Evelyn; BRENNAN, Karen; MILLNER, Amon; ROSENBAUM, Eric; SILVER, J a Y; SILVERMAN, Brian; KAFAI, Yasmin. Scratch: Programming for All. *Communications of the ACM*, v. 52, p. 60–67, 2009. ISSN 00010782. Disponível em: <[REZENDE, Criscilla M.C.; BISPO, Esdras L. Comparison between the use of pseudocode and visual programming in programming teaching: An evaluation from scratch tool. v. 2018-June, p. 1–5, 2018. ISSN 21660735.](http://search.ebscohost.com/login.aspx?direct=true{\&}db=bth{\&}AN=45021156{\&}site=eds-live{\&}5Cnfiles/130/RESNICK et al. - 2009 - Scratch Programming > .</a></p>
</div>
<div data-bbox=)

SANTOS, Priscila S C; ARAUJO, Luis Gustavo J; BITTENCOURT, Roberto A. A Mapping Study of Computational Thinking and Programming in Brazilian K-12 Education. *FIE 2018 - 48th Annual Frontiers In Education Conference*, n. July, 2018.

SILVA, Thiago Reis; MEDEIROS, Tainá; MEDEIROS, Handerson; LOPES, Ranyer; ARANHA, Eduardo. Ensino-aprendizagem de programação: uma revisão sistemática da literatura. *Revista Brasileira de Informática na Educação*, v. 23, n. 01, p. 182, 2015. ISSN 1414-5685. Disponível em: <<http://www.br-ie.org/pub/index.php/rbie/article/view/2838>> .

TUCKER, Allen. A {Model} {C}urriculum for {K}–12 {C}omputer {S}cience: {F}inal {R}eport of the {ACM} {K}–12 {T}ask {F}orce {C}urriculum {C}ommittee. 2003.

WERNTROP, David; WILENSKY, Uri. The challenges of studying blocks-based programming environments. In: *Proceedings - 2015 IEEE Blocks and Beyond Workshop, Blocks and Beyond 2015*. [S.l.: s.n.], 2015. p. 5–7. ISBN 9781467383677.

WERNTROP, David; WILENSKY, Uri. Comparing Block-Based and Text-Based Programming in High School Computer Science Classrooms. *ACM Transactions on Computing Education*, v. 18, n. 1, p. 1–25, 2017. ISSN 19466226. Disponível em: <<http://dl.acm.org/citation.cfm?doid=3155324.3089799>> .

WING, Jeannette M. Computational thinking. *Communications of the ACM*, v. 49, n. 3, p. 33, 2006. ISSN 00010782. Disponível em: <<http://portal.acm.org/citation.cfm?doid=1118178.1118215>> .

WING, Jeannette M. Computational thinking and thinking about computing. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, v. 366, n. 1881, p. 3717–3725, 2008. ISSN 1364503X.