

BitBeak: Gamificação e Mangás como Apoio ao Ensino de Programação

Gustavo Henrique Alves
Escola Politécnica, PUCPR
Curitiba, Brasil
a.henrique8@pucpr.edu.br

Marcos Vinicius Alves de Souza
Escola Politécnica, PUCPR
Curitiba, Brasil
a.vinicius1@pucpr.edu.br

Tiago Adelino Navarro
Escola Politécnica, PUCPR
Curitiba, Brasil
tiago.navarro@pucpr.br

RESUMO

O ensino de programação enfrenta desafios relacionados ao engajamento e à retenção de estudantes, especialmente em disciplinas introdutórias que frequentemente apresentam altas taxas de evasão. Diversos fatores — como a complexidade dos conceitos iniciais em lógica de programação, estruturas de controle, manipulação de variáveis, desenvolvimento de algoritmos, uso de funções, arrays e listas, e estruturas de dados básicas como pilhas e filas — aliados à falta de motivação, podem dificultar a aprendizagem e o progresso dos alunos na área. Este artigo apresenta o BitBeak, uma ferramenta educacional gamificada que utiliza narrativas em estilo mangá para tornar o ensino de programação mais acessível e envolvente. O impacto dessa abordagem foi avaliado por meio do modelo MEEGA+, analisando usabilidade, experiência do jogador e percepção de aprendizado. O estudo envolveu estudantes universitários que interagiram com a plataforma, com aplicação de métricas quantitativas e qualitativas para mensurar engajamento e retenção de conhecimento. Os resultados indicam que a combinação entre gamificação e mangás aumenta a motivação e a compreensão conceitual dos alunos, apoiando os processos de ensino e aprendizagem em programação.

PALAVRAS-CHAVE

Gamificação, mangás, ensino de programação, aprendizagem ativa, MEEGA+

1 Introdução

A aprendizagem de programação desempenha um papel fundamental na formação de profissionais de tecnologia. No entanto, disciplinas introdutórias frequentemente apresentam altos índices de evasão, um problema recorrente em cursos de computação [4]. A complexidade dos conceitos iniciais, aliada à ausência de contextos práticos motivadores, torna o processo de ensino-aprendizagem desafiador para muitos estudantes.

Diante desse cenário, diversas estratégias educacionais inovadoras têm sido exploradas com o objetivo de tornar o ensino de programação mais acessível e envolvente. Entre essas abordagens, destacam-se a *gamificação* e o uso de *mangás*. A gamificação aplica elementos lúdicos — como desafios, rankings e recompensas — para estimular o engajamento dos alunos [12]. Já os mangás auxiliam na visualização e contextualização de conceitos abstratos, como estruturas de dados, lógica condicional e orientação a objetos, tornando o aprendizado mais intuitivo e imersivo [11].

Este artigo apresenta o *BitBeak*, uma ferramenta educacional que combina gamificação e mangás para apoiar o ensino introdutório de programação. A proposta é investigada por meio de um estudo empírico com aplicação do modelo MEEGA+, que avalia

dimensões como usabilidade, experiência do jogador e percepção de aprendizado [8]. As principais contribuições deste trabalho são:

- **Ferramenta interativa:** Desenvolvimento de uma solução que integra gamificação e mangás no ensino de programação.
- **Avaliação empírica:** Análise do impacto dessa abordagem no engajamento e aprendizagem dos estudantes.
- **Retenção de alunos:** Estudo de dados quantitativos e qualitativos para compreender o potencial das técnicas aplicadas.

2 Fundamentação Teórica

2.1 Gamificação no Ensino de Programação

A gamificação tem se consolidado como uma estratégia pedagógica eficaz para aumentar a motivação, a persistência e o engajamento de estudantes em diferentes níveis de ensino, especialmente em cursos introdutórios de programação [4, 12]. Em disciplinas como Lógica de Programação, Algoritmos e Estruturas de Dados, os alunos frequentemente enfrentam dificuldades para compreender conceitos abstratos, como operadores lógicos, fluxos de controle, recursão e estruturas de dados dinâmicas. A ausência de feedback imediato e a descontextualização desses conteúdos contribuem para altos índices de frustração e evasão. Abordagens gamificadas têm sido utilizadas para mitigar esses desafios ao proporcionar progressão clara, feedback contínuo e experiências de aprendizagem mais envolventes [7, 12].

Pesquisas mostram que a motivação dos alunos pode ser intensificada por meio da gamificação ao atingir tanto aspectos extrínsecos (como recompensas e reconhecimento) quanto intrínsecos (como autonomia, propósito e senso de competência) [9]. Frameworks como o *Octalysis*, de Chou [2], e o modelo de *gameful design* de Deterding et al. [3] são amplamente utilizados para estruturar experiências mais ricas em plataformas educacionais, alinhando mecânicas de jogo com objetivos de aprendizagem.

Além disso, há evidências empíricas de que a gamificação pode reduzir significativamente as taxas de evasão em cursos de computação, ao promover senso de progresso e conquista entre os alunos [7]. Ferramentas como *CodeCombat*, *Kahoot!*, *Grasshopper* e *Classcraft* exploram essas técnicas com diferentes níveis de sucesso.

No entanto, muitas dessas plataformas não integram elementos narrativos significativos — componentes fundamentais de uma história, como enredo, personagens, ambiente, tempo, narrador, conflito e tema — que são essenciais para criar uma experiência envolvente e contextualizada. A ausência desses elementos limita o potencial imersivo da experiência educacional.

O *BitBeak* busca justamente preencher essa lacuna, oferecendo uma experiência gamificada alinhada a uma progressão pedagógica

baseada em histórias em mangá, nas quais os conceitos de programação são apresentados de forma integrada à narrativa.

2.2 Mangás no Ensino de Programação

O uso de histórias visuais — como quadrinhos e, especialmente, mangás — tem sido explorado como ferramenta pedagógica para facilitar a compreensão de conteúdos complexos. Esse formato permite representar situações realistas ou fictícias, nas quais os conceitos são apresentados em contexto, promovendo identificação emocional e engajamento cognitivo com o conteúdo [11].

No ensino de programação, essa abordagem pode ser particularmente útil para ajudar os estudantes a visualizarem problemas computacionais dentro de cenários narrativos coerentes. A construção de enredos que envolvem personagens e desafios progressivos contribui para um senso de continuidade, propósito e progressão — características essenciais em ambientes de aprendizagem significativos. Pesquisas mostram que a identificação com personagens e o envolvimento emocional com a narrativa aumentam a retenção de conteúdo e a persistência em tarefas difíceis [6].

Ferramentas como o *CodeToon*, que transforma código-fonte em histórias em quadrinhos, ou abordagens educacionais baseadas em *storytelling*, como o projeto *Quest to Learn*, reforçam o potencial do uso de narrativas visuais no contexto educacional [10]. No entanto, essas soluções geralmente não integram gamificação de maneira estruturada, nem utilizam o mangá como parte fundamental da experiência de aprendizagem.

A proposta do *BitBeak* se diferencia nesse ponto ao incorporar histórias em estilo mangá como parte essencial do ciclo pedagógico: os alunos não apenas leem uma história, mas vivenciam esse enredo enquanto resolvem problemas relacionados ao conteúdo. Essa construção narrativa está entrelaçada aos desafios gamificados de cada trilha, promovendo engajamento contínuo com os temas de lógica, algoritmos e estruturas de dados.

2.3 Modelos de Avaliação para Jogos Educacionais

A avaliação de ferramentas educacionais gamificadas exige abordagens que vão além da simples aquisição de conhecimento, considerando também fatores como usabilidade da plataforma, motivação do usuário e experiência ao longo do processo de aprendizagem. Modelos tradicionais de avaliação de software, como o *System Usability Scale* (SUS), embora úteis, não contemplam aspectos pedagógicos nem dimensões emocionais diretamente relacionadas ao engajamento do aluno — como frustração, entusiasmo, senso de conquista ou pertencimento — que são fundamentais em jogos educacionais [1].

Nesse contexto, o modelo *MEEGA+* tem se destacado por oferecer uma abordagem mais abrangente, alinhada às particularidades do ensino de computação mediado por jogos. Desenvolvido com base em validações empíricas, o *MEEGA+* permite avaliar três dimensões centrais: usabilidade, experiência do jogador e percepção de aprendizado [8]. Esses elementos fornecem uma perspectiva mais concreta sobre o impacto que uma ferramenta pode ter sobre o envolvimento do aluno com o conteúdo e sobre sua permanência na trilha de aprendizagem.

Estudos anteriores demonstraram a eficácia do *MEEGA+* em diferentes contextos educacionais, validando seu uso em disciplinas como lógica de programação, estruturas de dados e engenharia de software [8]. Sua aplicação neste trabalho permite compreender o impacto do *BitBeak* não apenas sob a ótica do aprendizado de conceitos, mas também quanto à forma como os estudantes se relacionam com a plataforma, sua narrativa e os desafios propostos.

3 BitBeak: A Ferramenta Desenvolvida

A concepção do *BitBeak* combinou práticas de desenvolvimento iterativo com diretrizes pedagógicas centradas na experiência do usuário. O processo envolveu definição de personas, revisão bibliográfica de estratégias educacionais visuais e gamificadas, e observação empírica em contextos de ensino introdutório de programação.

A construção das trilhas, da narrativa visual e dos desafios seguiu uma abordagem tangencial, que serve como complemento ao ensino tradicional, e não como substituto. A aprendizagem tangencial (ou tangential learning) consiste em proporcionar ao aluno um ambiente de interação no qual o conhecimento é adquirido de maneira indireta, como um efeito indireto de uma experiência significativa, lúdica e envolvente [5].

Em vez de apresentar o conteúdo de forma direta e instrucional, a abordagem tangencial cria oportunidades para que o estudante aprofunde temas com os quais já teve contato em sala de aula, aplicando-os em novos contextos por meio de curiosidade e exploração ativa. Ao invés de ensinar “o que” e “como” explicitamente, a ferramenta provoca o “por que” e “quando” por meio da prática, narrativa e resolução de problemas.

No contexto do *BitBeak*, essa abordagem se manifesta por meio de trilhas temáticas organizadas em níveis, que remetem a conceitos previamente introduzidos nas disciplinas formais, como lógica de programação, algoritmos e estruturas de dados. A resolução de desafios ocorre dentro de uma narrativa visual em estilo mangá, que contextualiza os problemas, apresenta personagens e motivações, e mantém o aluno engajado por meio de progressão simbólica, recompensas e competição.

Com isso, o *BitBeak* não apenas reforça conteúdos já abordados em aula, mas o faz de forma imersiva, promovendo um ciclo de revisão, prática e descoberta contínua, que favorece a retenção, reduz a ansiedade frente ao erro e estimula o protagonismo no processo de aprendizagem.

As interfaces foram inicialmente prototipadas no Figma e implementadas com *React* (front-end), *.NET* (back-end) e *SQL Server* (banco de dados).

3.1 Estrutura Geral e Objetivos

O *BitBeak* é uma ferramenta educacional gamificada desenvolvida para apoiar o ensino de programação, com foco na linguagem JavaScript, voltada a estudantes de cursos como Engenharia de Software. Seu principal objetivo é oferecer uma experiência de aprendizado mais envolvente, progressiva e significativa, especialmente em disciplinas introdutórias como Lógica de Programação e Algoritmos.

A plataforma foi desenvolvida como um aplicativo mobile com *front-end* em *React*, *back-end* em C# com *.NET*, e banco de dados

SQL Server hospedado na nuvem. O design inicial da interface foi realizado no *Figma*, com foco em usabilidade e clareza na progressão do aluno. Todo o conteúdo está em português, com aplicação voltada ao contexto educacional brasileiro.

A jornada do usuário é construída em torno da resolução de questões organizadas em trilhas temáticas. A cada atividade completada, o estudante ganha experiência e penas, que funcionam como recompensas internas e alimentam o sistema de progressão. O *BitBeak* também conta com um ranking de desempenho, atualizado quinzenalmente, incentivando o engajamento contínuo.

3.2 Trilhas e Progressão

A jornada do usuário é construída em torno da realização de atividades organizadas em trilhas temáticas, cada uma subdividida em cinco níveis sequenciais. Esses níveis abordam progressivamente os principais tópicos de programação, com objetivos de aprendizagem específicos que se alinham à narrativa e à complexidade crescente das atividades propostas. A progressão é baseada na resolução gradual de problemas, respeitando a lógica da construção em espiral do conhecimento.

As trilhas e seus focos são:

- Trilha I – Lógica de Programação: Introduz os fundamentos da lógica computacional e das estruturas de controle. Exemplos de temas por nível: operadores lógicos e aritméticos, estruturas condicionais, laços de repetição, decomposição de problemas e fluxogramas.
- Trilha II – Algoritmos: Explora técnicas de resolução de problemas com foco em busca, ordenação, recursão e análise de complexidade. Exemplos de temas por nível: algoritmos de busca linear e binária, ordenações simples, estruturas iterativas, recursão e refinamento de algoritmos.
- Trilha III – Estruturas de Dados: Foca na organização e manipulação de dados usando estruturas básicas e dinâmicas. Exemplos de temas por nível: *arrays*, listas encadeadas, pilhas, filas e árvores binárias de busca.
- Trilha IV – Programação Orientada a Objetos (POO): Introduz os conceitos da POO aplicados à modelagem de sistemas. Exemplos de temas por nível: classes e objetos, encapsulamento, herança, polimorfismo e sobrecarga.
- Trilha V – Algoritmos Avançados: Aborda algoritmos de maior complexidade com aplicação em problemas do mundo real. Exemplos de temas por nível: programação dinâmica, grafos e caminhos mínimos, roteamento, heurísticas e otimização.

Cada trilha exige que o aluno conclua todas as atividades de um nível para desbloquear o próximo, garantindo o domínio progressivo dos conceitos. Essa estrutura permite que o estudante avance de forma autônoma, sempre inserido em um contexto narrativo que reforça a motivação e a retenção do conteúdo.

3.3 Narrativa e Contextualização das Atividades

Cada trilha do *BitBeak* é acompanhada por uma narrativa original em formato de mangá, organizada em episódios que correspondem aos níveis da trilha. Esses episódios servem como introdução ao conteúdo da trilha e apresentam personagens, conflitos e situações que contextualizam os conceitos de programação abordados.



Figura 1: Interface da trilha “Lógica de Programação”.

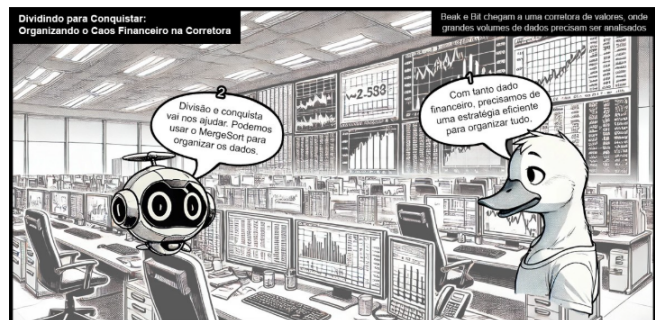


Figura 4: Mangá do BitBeak: organização de dados financeiros com MergeSort.

Após a leitura da história, o aluno acessa a interface das atividades, onde deve resolver os problemas apresentados. A resolução correta desbloqueia o próximo episódio da narrativa e dá continuidade ao progresso na trilha.



Figura 2: Mapa da trilha “Lógica de Programação” no BitBeak.



Figura 3: Mangá do BitBeak: sistema de inventário com arrays.

Cada trilha apresenta uma história alinhada aos conceitos trabalhados. A seguir, são descritos os enredos utilizados:

- Trilha I – Um Dia na Lanchonete: Beak e Bit ajudam um atendente a organizar pedidos, calcular troco e otimizar o atendimento, aplicando operadores lógicos, estruturas condicionais e comandos de repetição.
- Trilha II – Organizando o Inventário: Os personagens catalogam e localizam produtos em um armazém, utilizando algoritmos de busca, ordenação e estratégias de refinamento.
- Trilha III – Labirinto de Dados: Em uma exploração por um centro de dados, os alunos aplicam estruturas como árvores, pilhas e filas para otimizar o fluxo de informações.
- Trilha IV – Sistema para a Lanchonete: Os desafios envolvem a modelagem de um sistema para gerenciar pedidos e funcionários, explorando conceitos de classes, herança e encapsulamento.
- Trilha V – Cidade Inteligente: Os alunos enfrentam o desafio de otimizar rotas de entrega em um cenário urbano, aplicando algoritmos de grafos, caminhos mínimos e heurísticas.

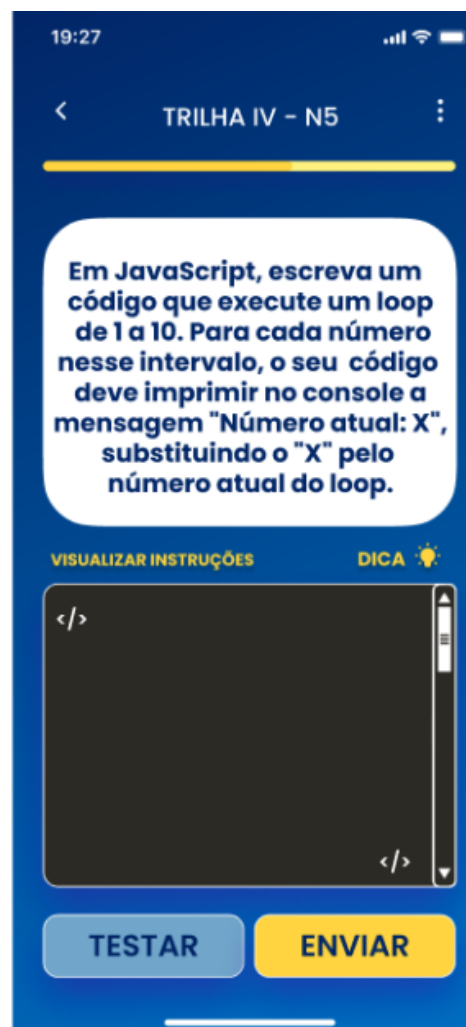


Figura 5: Atividade prática da Trilha IV sobre laços em JavaScript.

Essa separação clara entre narração e atividade reforça o caráter pedagógico da plataforma. O aluno não apenas resolve problemas técnicos, mas o faz dentro de uma experiência narrativa coesa, que motiva, contextualiza e dá significado ao conteúdo.

3.4 Funcionalidades Complementares

Além das trilhas temáticas e das narrativas em mangá, o *BitBeak* oferece funcionalidades complementares que tornam a experiência mais envolvente e dinâmica. Um dos principais recursos é o ranking, que exhibe periodicamente a posição dos usuários com base nas penas acumuladas. A pontuação é obtida ao concluir desafios, realizar atividades e avançar nas trilhas. O ranking cria um ambiente saudável de competição, incentivando o engajamento contínuo na plataforma.



Figura 6: Ranking do BitBeak com avatares e pontuação quinzenal.

Outro elemento motivador são as missões, que funcionam como objetivos passivos. Elas são registradas automaticamente conforme o aluno interage com a plataforma — por exemplo, ao concluir episódios de uma trilha ou resolver desafios consecutivos sem erros.

Ao atingir a meta, o sistema recompensa o aluno com penas adicionais. Esse tipo de gamificação passiva reforça o engajamento ao não exigir ações explícitas por parte do usuário, criando momentos de surpresa e reforço positivo.

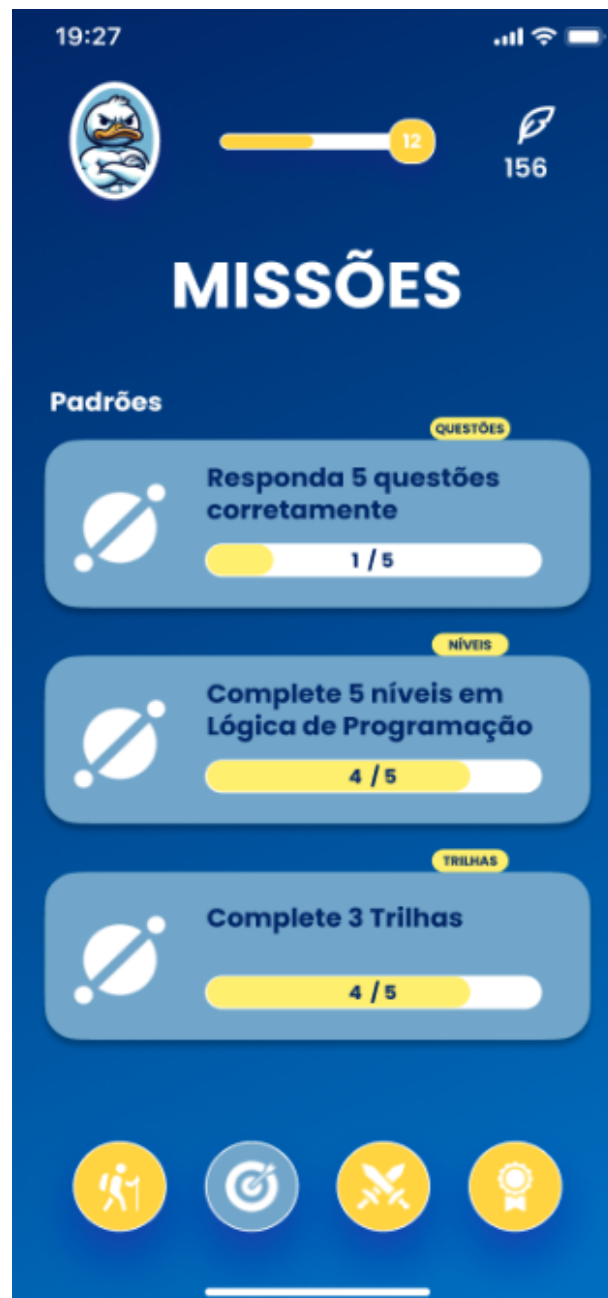


Figura 7: Tela de missões do BitBeak com objetivos paralelos.

Por fim, o *BitBeak* conta com um modo de desafios entre usuários, no qual dois estudantes competem para resolver problemas. A partida termina quando um deles comete um erro, criando um ambiente de tensão e foco semelhante ao de competições de programação.

Essa funcionalidade promove a aplicação rápida do conhecimento e estimula o domínio conceitual por meio da prática recorrente.

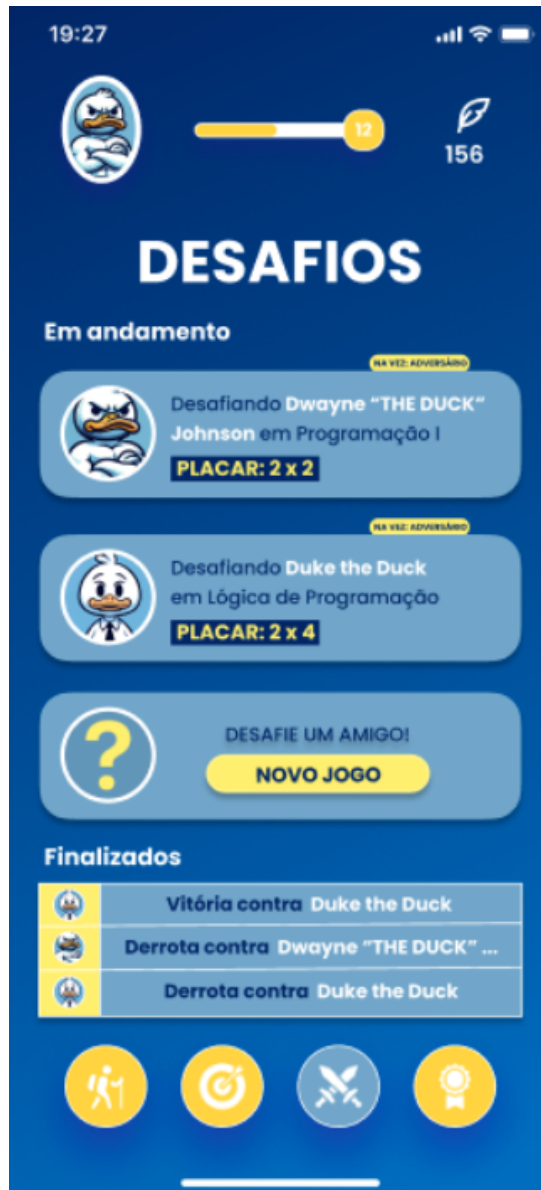


Figura 8: Modo Desafios no BitBeak com placares e histórico de partidas.

4 Metodologia

Esta pesquisa teve como objetivo avaliar a experiência dos alunos com a plataforma *BitBeak*, considerando fatores como usabilidade, engajamento e percepção de aprendizado. Para isso, foi realizada uma aplicação prática com estudantes de graduação da área de computação, seguida da coleta de dados utilizando um questionário baseado no modelo *MEEGA+*.

A avaliação foi conduzida em sala de aula com a participação de 20 estudantes, todos com diferentes níveis de conhecimento

prévio em programação. Durante a atividade, os alunos acessaram a plataforma em seus próprios dispositivos (celulares), navegando pelas trilhas, lendo as narrativas e resolvendo os desafios propostos.

Após a experiência, os participantes preencheram um formulário baseado no modelo *MEEGA+*, que avalia aspectos como percepção de aprendizado, experiência do jogador, engajamento e usabilidade. O questionário foi levemente adaptado para o contexto do *BitBeak*, com ajustes em termos originalmente voltados para jogos de tabuleiro e a inclusão de perguntas específicas sobre mecânicas da plataforma, como a utilização de mangás e o sistema de ranking.

As respostas foram coletadas de forma anônima e analisadas quantitativamente. Os dados foram agrupados por dimensão e tratados por meio de estatísticas descritivas, com destaque para a mediana e a distribuição das respostas. Também foram consideradas as observações dos participantes nos campos abertos do formulário, permitindo uma análise complementar qualitativa da experiência com a ferramenta.

5 Resultados

A análise dos dados coletados com o questionário *MEEGA+* foi organizada em três eixos principais: **Percepção de Aprendizado**, **Experiência do Jogador** e **Usabilidade**. Cada dimensão foi avaliada por um conjunto de afirmações, com respostas em escala de concordância. Os resultados são apresentados a seguir, com base na mediana das respostas e nas distribuições obtidas entre os 20 participantes.

5.1 Percepção de Aprendizado

Esta dimensão busca identificar se os alunos perceberam melhora em seu entendimento dos conteúdos abordados. Os dados indicam que a maioria dos participantes concordou que o *BitBeak* contribuiu para a compreensão dos conceitos de programação, sobretudo pela contextualização das atividades proporcionada pela leitura dos mangás.

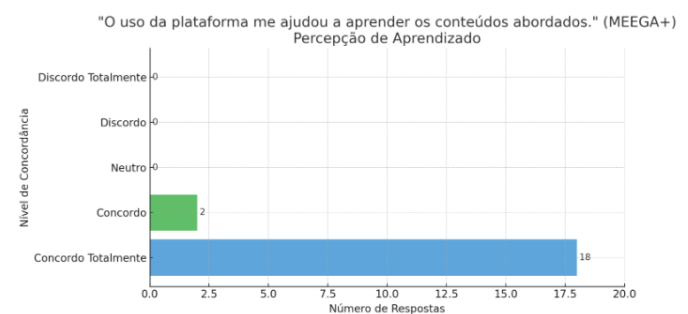


Figura 9: Percepção de aprendizado com o uso da plataforma BitBeak.

5.2 Experiência do Jogador

Esta dimensão contempla aspectos como imersão, clareza de objetivos, diversão e satisfação geral com a plataforma. Os resultados sugerem que a experiência com o *BitBeak* foi amplamente positiva: os participantes relataram altos níveis de envolvimento durante o

uso, consideraram os objetivos claros e apontaram a proposta como inovadora em comparação a outras atividades de programação.

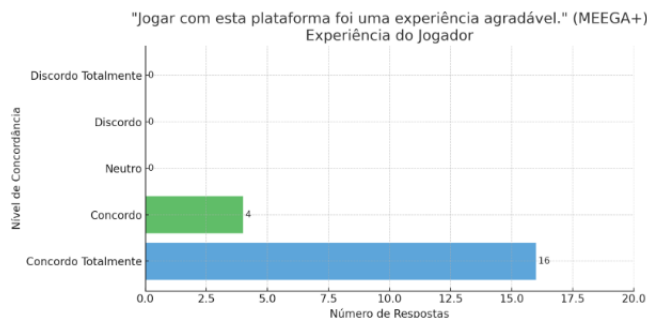


Figura 10: Avaliação da experiência do jogador com a plataforma BitBeak.

5.3 Usabilidade

A dimensão de usabilidade avalia a estética, navegabilidade e facilidade de uso da plataforma. Os resultados mostram que a interface do *BitBeak* foi bem recebida, com avaliações positivas sobre a clareza visual, acessibilidade e a fluidez da interação. Não foram relatadas dificuldades técnicas relevantes durante o uso.

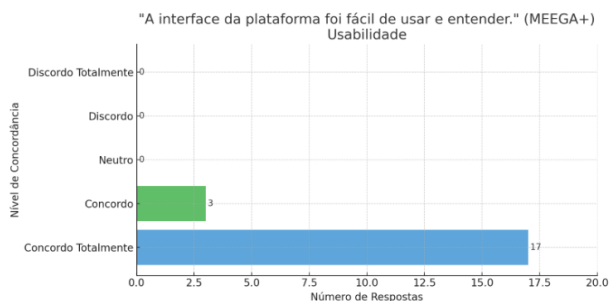


Figura 11: Avaliação da usabilidade da plataforma BitBeak.

5.4 Comentários Abertos

Além das respostas quantitativas, os participantes deixaram comentários que complementam a análise da experiência com o *BitBeak*. Muitos destacaram a proposta como criativa e diferente do habitual no ensino de programação, especialmente pela presença dos mangás e personagens. Também foram feitas sugestões para aprimorar funcionalidades e expandir o conteúdo disponível.

A seguir, trechos de comentários fornecidos por participantes durante a avaliação da plataforma:

“A proposta com mangás foi muito criativa, nunca tinha visto algo assim para ensinar programação.”

“Gostei da interface, simples de usar e bonita. Os personagens deixaram mais divertido.”

“Seria legal se as penas acumuladas pudessem ser usadas de alguma forma no jogo.”

6 Discussão dos Resultados

6.1 Interpretação

Os resultados obtidos com a aplicação do modelo MEEGA+ indicam que o *BitBeak* foi bem recebido pelos participantes em todas as dimensões avaliadas. A dimensão *Percepção de Aprendizado* apresentou respostas majoritariamente positivas, sugerindo que a contextualização dos conteúdos por meio de narrativas em mangá, aliada à resolução de atividades práticas, contribuiu para a compreensão dos conceitos de programação.

Na dimensão *Experiência do Jogador*, os participantes relataram níveis elevados de concentração, diversão e satisfação durante o uso da plataforma. A clareza dos objetivos e o estilo narrativo foram citados como elementos que tornaram a experiência mais envolvente do que métodos tradicionais. Essa resposta positiva está alinhada com os princípios da gamificação, ao proporcionar recompensas simbólicas (como penas e rankings) e elementos estéticos que favorecem o engajamento.

Quanto à *Usabilidade*, os estudantes avaliaram positivamente a interface do *BitBeak*, apontando facilidade de uso, boa estética e fluidez na navegação. Não foram identificadas barreiras significativas ao uso da plataforma, o que reforça sua viabilidade como ferramenta de apoio didático.

As observações abertas reforçaram os dados quantitativos, com comentários que destacaram a originalidade da proposta, a qualidade visual das histórias e a motivação gerada pelo formato não convencional. Algumas sugestões apontaram oportunidades de expansão, como a inclusão de novas trilhas e o aproveitamento prático das penas acumuladas.

6.2 Limitações do Estudo

Apesar dos resultados positivos, esta avaliação apresenta algumas limitações que devem ser consideradas. O número de participantes foi relativamente reduzido ($n = 20$), o que limita a generalização dos achados para outros contextos educacionais. Além disso, a aplicação ocorreu em uma única sessão, o que dificulta a análise de impacto a longo prazo ou da retenção do conhecimento adquirido.

O instrumento de avaliação, embora baseado no modelo validado MEEGA+, foi levemente adaptado, o que pode afetar a comparação direta com outros estudos. Algumas perguntas foram ajustadas para refletir as características específicas do *BitBeak* e outras foram adicionadas para captar elementos que o modelo original não contemplava (como o uso de mangás ou o sistema de ranking).

Também se destacam ameaças à validade do estudo. A validade interna é impactada pela ausência de grupo controle e pela possibilidade de viés na seleção dos participantes. A validade externa é limitada pela aplicação em um único contexto institucional. Em relação à validade de construto, as adaptações no modelo MEEGA+ podem influenciar sua comparação com outros trabalhos. Para mitigar esses aspectos, a aplicação foi conduzida em ambiente controlado, com condições padronizadas e coleta anônima de dados. O uso do MEEGA+, por ser um instrumento validado e amplamente adotado, contribuiu para a confiabilidade das dimensões avaliadas — usabilidade, motivação e percepção de aprendizado.

Por fim, os dados foram analisados de forma descritiva, sem aplicação de testes estatísticos mais robustos. Embora adequados ao escopo da pesquisa, análises futuras poderão empregar métodos

inferenciais para identificar padrões mais aprofundados ou relações entre variáveis.

6.3 Lições Aprendidas

A aplicação da ferramenta proporcionou aprendizados relevantes sobre o impacto de abordagens visuais e gamificadas no ensino de programação. Os participantes demonstraram alto nível de engajamento e destacaram a clareza dos objetivos, a sensação de progresso e a motivação gerada pelas trilhas temáticas e narrativas em mangá.

Entre os pontos observados, destacou-se a importância da linguagem acessível nos enunciados dos desafios — alguns estudantes relataram dificuldades com termos técnicos. Essa devolutiva motivou ajustes no vocabulário e reforço da contextualização sem comprometer a precisão conceitual. Além disso, os comentários dos participantes sugerem que o uso de histórias ilustradas e desafios interativos contribui para tornar o conteúdo mais intuitivo, especialmente para iniciantes.

Planeja-se, em fases futuras, realizar comparações sistemáticas com outras metodologias e ferramentas, usando o *MEEGA+* e instrumentos complementares para aprofundar a análise do impacto na aprendizagem e retenção.

7 Conclusão e Trabalhos Futuros

Este trabalho apresentou o *BitBeak*, uma ferramenta educacional gamificada desenvolvida para apoiar o ensino de programação por meio da combinação de narrativas em mangá e mecânicas de jogo. A plataforma foi estruturada em trilhas progressivas com atividades contextualizadas, buscando tornar o processo de aprendizagem mais acessível, envolvente e significativo para estudantes de cursos de computação.

A avaliação realizada com 20 alunos, utilizando o modelo *MEEGA+*, indicou resultados positivos em todas as dimensões analisadas. Os participantes relataram aumento na motivação, boa compreensão dos conteúdos abordados e uma experiência de uso agradável e intuitiva. A originalidade da proposta, aliada ao foco pedagógico, demonstra o potencial do *BitBeak* como recurso complementar no ensino introdutório de programação.

Como contribuições principais, destacam-se o desenvolvimento de uma plataforma com mangás integrados, a estrutura pedagógica baseada em progressão por trilhas, e a aplicação de um modelo de avaliação validado no contexto da computação. Apesar dos resultados promissores, reconhece-se a limitação do número reduzido de participantes e da aplicação pontual.

Como trabalhos futuros, pretende-se expandir a plataforma com novas trilhas, aumentar a personalização da experiência do usuário e implementar funcionalidades ainda não ativadas, como o uso das penas acumuladas. Também se propõe a realização de avaliações longitudinalmente, com amostras maiores e comparações entre grupos, a fim de aprofundar a análise do impacto do *BitBeak* no desempenho e na retenção dos estudantes.

Além disso, a próxima versão da plataforma priorizará o acesso via navegadores web, visando melhor aproveitamento visual e compatibilidade com diferentes dispositivos. A interface mobile será mantida como alternativa. A coleta de dados será integrada ao fluxo de uso por meio de formulários nativos, permitindo maior fluidez na análise de usabilidade, motivação e aprendizado.

Pretende-se também implementar personalização baseada no perfil do usuário — incluindo idade, familiaridade com jogos e conhecimento prévio em programação — coletados no momento do cadastro. Essa personalização poderá influenciar positivamente a forma como os conteúdos são apresentados, respeitando diferentes contextos e níveis de familiaridade.

A plataforma será disponibilizada para uso gratuito em contextos educacionais, embora o código-fonte permaneça privado devido ao planejamento de lançamento comercial. Artefatos como imagens, documentação e amostras de uso poderão ser fornecidos sob solicitação, com o objetivo de apoiar colaborações acadêmicas e futuras pesquisas na área.

DISPONIBILIDADE DE ARTEFATO

Os artefatos desenvolvidos neste estudo, incluindo o código-fonte do jogo *BitBeak*, imagens de tela e documentação associada, estão disponíveis para consulta mediante solicitação durante o processo de revisão. O projeto encontra-se em fase de desenvolvimento e preparação para lançamento comercial. A plataforma será disponibilizada para uso gratuito em ambientes educacionais, mas o código-fonte permanecerá fechado.

AGRADECIMENTOS

Agradecemos aos estudantes que participaram voluntariamente das avaliações, bem como à Pontifícia Universidade Católica do Paraná pelo apoio institucional. Também reconhecemos as contribuições dos colegas que auxiliaram em testes, revisões e discussões ao longo do desenvolvimento da plataforma.

REFERÊNCIAS

- [1] John Brooke. 1996. SUS: A Quick and Dirty Usability Scale. In *Usability Evaluation in Industry*, Patrick W. Jordan, Bruce Thomas, Ian L. McClelland, and Bernard Weerdmeester (Eds.). Taylor & Francis, London, 189–194.
- [2] Yu-kai Chou. 2015. *Actionable Gamification: Beyond Points, Badges, and Leaderboards*. CreateSpace Independent Publishing Platform, Fremont, CA. Trade paperback, 506p.
- [3] S. Deterding, D. Dixon, R. Khaled, and L. Nacke. 2011. From game design elements to gameness: Defining “gamification”. In *15th International Academic MindTrek Conference*. 9–15. doi:10.1145/2181037.2181040
- [4] Marcin Fojcik, Martyna K. Fojcik, Sven-Olavi Høyland, and Jon Øivind Hoem. 2022. Reflections on Didactical Challenges in Teaching Computer Programming. In *Education and New Developments 2022*, M. Carmo (Ed.). inScience Press, Lisbon, Portugal, Chapter 42, 517–529. <https://press.insciencypress.org/index.php/press/catalog/view/18/34/509-1>
- [5] James Paul Gee. 2003. *What Video Games Have to Teach Us About Learning and Literacy*. Palgrave Macmillan, New York, NY.
- [6] Melanie C. Green and Timothy C. Brock. 2000. The role of transportation in the persuasiveness of public narratives. *Journal of Personality and Social Psychology* 79, 5 (2000), 701–721. doi:10.1037/0022-3514.79.5.701
- [7] Juho Hamari, Jonna Koivisto, and Harri Sarsa. 2014. Does gamification work? – A literature review of empirical studies on gamification. In *47th Hawaii International Conference on System Sciences (HICSS)*. 3025–3034. doi:10.1109/HICSS.2014.377
- [8] Gustavo Petri, Claudia Gresse von Wangenheim, and Adilson Luiz de Oliveira Borgatto. 2018. *MEEGA+: A Method for the Evaluation of Educational Games for Computing Education*. Technical Report INCoD/GQS.05.2018.E. Universidade Federal de Santa Catarina (UFSC), INCoD/GQS Research Group, Florianópolis, Brazil. <https://www.researchgate.net/publication/326722665>
- [9] R. M. Ryan and E. L. Deci. 2000. Intrinsic and extrinsic motivations: Classic definitions and new directions. *Contemporary Educational Psychology* 25, 1 (2000), 54–67. doi:10.1006/ceps.1999.1020
- [10] Katie Salen, Robert Torres, Loretta Wolozin, Rebecca Rufo-Tepner, and Arana Shapiro. 2011. *Quest to Learn: Developing the School for Digital Kids*. The MIT Press, Cambridge, MA. <https://clalliance.org/wp-content/uploads/2020/03/Quest-To-Learn-Developing-School-for-Digital-Kids.pdf>
- [11] Sangho Suh, Celine Latulipe, Ken Jen Lee, Bernadette Cheng, and Edith Law. 2021. Using Comics to Introduce and Reinforce Programming Concepts in CS1. In

Proceedings of the 52nd ACM Technical Symposium on Computer Science Education (SIGCSE '21). Association for Computing Machinery, New York, NY, USA, 852–858. doi:10.1145/3408877.3432465

[12] Jakub Swacha and Justyna Szydłowska. 2023. Does Gamification Make a Difference in Programming Education? Evaluating FGPE-Supported Learning Outcomes. *Education Sciences* 13, 10 (2023), 984. doi:10.3390/educsci13100984