

UNIVERSIDADE FEDERAL DE SERGIPE  
CENTRO DE CIÊNCIAS EXATAS E TECNOLOGIA  
DEPARTAMENTO DE COMPUTAÇÃO

# **Gamificação no Processo de Aprendizagem: Desenvolvimento e Aplicação de uma Plataforma Lúdica Educacional**

Trabalho de Conclusão de Curso

Luan Kyrtinem de Oliveira Gomes



São Cristóvão – Sergipe

2025

UNIVERSIDADE FEDERAL DE SERGIPE  
CENTRO DE CIÊNCIAS EXATAS E TECNOLOGIA  
DEPARTAMENTO DE COMPUTAÇÃO

Luan Kyrtinem de Oliveira Gomes

## **Gamificação no Processo de Aprendizagem: Desenvolvimento e Aplicação de uma Plataforma Lúdica Educacional**

Trabalho de Conclusão de Curso submetido ao Departamento de Computação da Universidade Federal de Sergipe como requisito parcial para a obtenção do título de Bacharel em Ciência da Computação.

Orientador(a): Prof. Dr. Alberto Costa Neto

São Cristóvão – Sergipe

2025

# Agradecimentos

A Deus, pela vida, pela força e pela sabedoria concedida em cada etapa deste percurso, mesmo diante dos desafios mais difíceis.

À minha família, por todo o apoio, incentivo e compreensão ao longo da caminhada, sendo base essencial para que este trabalho pudesse se concretizar.

E a todos que, de alguma forma, contribuíram com palavras, gestos e estímulos durante este processo, tornando possível a realização desta conquista.

*Um fracasso não é razão para desistir, mas sim para se levantar mais forte.*  
*(Rock Lee, Naruto)*

# Resumo

Este trabalho apresenta o desenvolvimento de um aplicativo móvel educacional que utiliza a gamificação e minigames como estratégia pedagógica para apoiar o processo de ensino-aprendizagem. O aplicativo foi concebido com o objetivo central de tornar a aprendizagem mais acessível, estimulando o engajamento e a participação ativa dos estudantes por meio de abordagens lúdicas. Para fundamentar seu desenvolvimento, foi realizada uma análise de soluções existentes e de fundamentos teóricos que sustentam o uso de elementos lúdicos no contexto educacional. A partir desse embasamento, o aplicativo foi projetado e implementado como uma plataforma adaptável, na qual educadores podem criar seus próprios cursos. Como resultado, obteve-se uma solução viável que busca explorar o potencial da aplicação de mecânicas de gamificação no processo de ensino. Embora avanços importantes tenham sido alcançados, etapas como a validação com diferentes perfis de usuários permanecem como proposta para trabalhos futuros. Ainda assim, o sistema desenvolvido configura-se como um passo em direção à utilização de metodologias ativas apoiadas por tecnologia, oferecendo subsídios para novas investigações e aprimoramentos.

**Palavras-chave:** Gamificação. Aprendizagem Móvel. Metodologias Ativas. Tecnologia Educacional. Minigames.

# Abstract

This work presents the development of an educational mobile application that employs gamification and mini-games as a pedagogical strategy to support the teaching and learning process. The application was conceived with the central objective of making learning more accessible, fostering student engagement and active participation through playful approaches. To support its development, an analysis of existing solutions and theoretical foundations that sustain the use of playful elements in the educational context was conducted. Based on this foundation, the application was designed and implemented as an adaptable platform in which educators can create their own courses. As a result, a viable solution was obtained that seeks to explore the potential of applying gamification mechanics in the teaching process. Although significant advances have been made, stages such as validation with different user profiles remain as proposals for future work. Nevertheless, the developed system represents a step toward the adoption of active methodologies supported by technology, providing a basis for further investigations and improvements.

**Keywords:** Gamification. Mobile Learning. Active Methodologies. Educational Technology. Mini-games.

# Lista de ilustrações

|                                                                                           |    |
|-------------------------------------------------------------------------------------------|----|
| Figura 1 – Telas do aplicativo Mimo . . . . .                                             | 22 |
| Figura 2 – Telas do aplicativo Programming Hero . . . . .                                 | 23 |
| Figura 3 – Telas do aplicativo Scratch . . . . .                                          | 24 |
| Figura 4 – Telas do aplicativo Duolingo . . . . .                                         | 25 |
| Figura 5 – Telas do aplicativo Lightbot: Code Hour . . . . .                              | 26 |
| Figura 6 – Telas do aplicativo Quizizz: Play to learn . . . . .                           | 27 |
| Figura 7 – Telas do aplicativo Raptivity . . . . .                                        | 28 |
| Figura 8 – Telas do aplicativo Lumosity . . . . .                                         | 29 |
| Figura 9 – Diagrama de casos de uso . . . . .                                             | 36 |
| Figura 10 – Diagrama de classes . . . . .                                                 | 37 |
| Figura 11 – Diagrama de entidade relacionamento estendido . . . . .                       | 38 |
| Figura 12 – Diagrama da arquitetura . . . . .                                             | 41 |
| Figura 13 – Telas iniciais do aplicativo TikTec . . . . .                                 | 42 |
| Figura 14 – Tela de Detalhamento da Aula . . . . .                                        | 43 |
| Figura 15 – Telas de execução da aula . . . . .                                           | 44 |
| Figura 16 – Telas de cursos . . . . .                                                     | 45 |
| Figura 17 – Telas de turmas . . . . .                                                     | 46 |
| Figura 18 – Tela de Perfil . . . . .                                                      | 47 |
| Figura 19 – Estrutura de pastas da API . . . . .                                          | 48 |
| Figura 20 – Trecho do código turma.dto.ts . . . . .                                       | 49 |
| Figura 21 – Trecho do código turma.controller.ts . . . . .                                | 49 |
| Figura 22 – Trecho do código turma.service.ts . . . . .                                   | 50 |
| Figura 23 – Trecho do código turma.repository.ts . . . . .                                | 52 |
| Figura 24 – Trecho do arquivo schema.prisma . . . . .                                     | 53 |
| Figura 25 – Estrutura da pasta de migrações do Prisma . . . . .                           | 54 |
| Figura 26 – Estrutura de pastas do Projeto . . . . .                                      | 55 |
| Figura 27 – Exemplo de composição de tela a partir de componentes reutilizáveis . . . . . | 56 |
| Figura 28 – Configuração do Axios no arquivo api.ts . . . . .                             | 57 |
| Figura 29 – Exemplo de funções que consomem endpoints no aplicativo . . . . .             | 57 |

# Lista de quadros

|                                                                                                          |    |
|----------------------------------------------------------------------------------------------------------|----|
| Quadro 1 – Aplicativos e links de acesso . . . . .                                                       | 20 |
| Quadro 2 – Identificações e funcionalidades . . . . .                                                    | 21 |
| Quadro 3 – Comparação entre os aplicativos e as funcionalidades . . . . .                                | 22 |
| Quadro 4 – Termos de busca do INPI. . . . .                                                              | 29 |
| Quadro 5 – Resultados obtidos através da busca na base do INPI. . . . .                                  | 30 |
| Quadro 6 – Resumo dos estudos acadêmicos obtidos a partir dos registros de software encontrados. . . . . | 30 |
| Quadro 7 – Resumo dos resultados obtidos através da busca nas bases de pesquisa. . .                     | 31 |
| Quadro 8 – Comparação entre os aplicativos de trabalhos científicos e as funcionalidades                 | 31 |
| Quadro 9 – Requisitos Funcionais . . . . .                                                               | 34 |
| Quadro 10 – Requisitos Não Funcionais . . . . .                                                          | 35 |
| Quadro 11 – Requisitos Funcionais para Funcionalidades Futuras . . . . .                                 | 64 |

# Lista de abreviaturas e siglas

|          |                                                                                      |
|----------|--------------------------------------------------------------------------------------|
| APP      | Aplicativo                                                                           |
| API Rest | Interface de Programação de Aplicações para Transferência de Estado Representacional |
| MVP      | Mínimo Produto Viável                                                                |
| SGBD     | Sistema Gerenciador de Banco de Dados                                                |
| SO       | Sistema Operacional                                                                  |

# Sumário

|          |                                                                                     |           |
|----------|-------------------------------------------------------------------------------------|-----------|
| <b>1</b> | <b>Introdução</b>                                                                   | <b>11</b> |
| 1.1      | Objetivos                                                                           | 12        |
| 1.1.1    | Objetivo geral                                                                      | 12        |
| 1.1.2    | Objetivos específicos                                                               | 13        |
| 1.2      | Metodologia                                                                         | 13        |
| 1.3      | Estrutura do Documento                                                              | 14        |
| <b>2</b> | <b>Fundamentação Teórica</b>                                                        | <b>15</b> |
| 2.1      | Dificuldade na Aprendizagem de Lógica de Programação                                | 15        |
| 2.2      | Processo da Gamificação no Ensino                                                   | 16        |
| 2.3      | O Uso da Gamificação e de Softwares Educacionais como Ferramenta de Apoio ao Ensino | 17        |
| <b>3</b> | <b>Estudo de Soluções Relacionadas</b>                                              | <b>19</b> |
| 3.1      | Prospecção de Mercado                                                               | 19        |
| 3.1.1    | Mimo                                                                                | 21        |
| 3.1.2    | Programming Hero                                                                    | 23        |
| 3.1.3    | Scratch                                                                             | 23        |
| 3.1.4    | Duolingo                                                                            | 24        |
| 3.1.5    | Lighbot: Code Hour                                                                  | 25        |
| 3.1.6    | Quizizz: Play to learn                                                              | 26        |
| 3.1.7    | Raptivity                                                                           | 27        |
| 3.1.8    | Lumosity                                                                            | 28        |
| 3.2      | Base de Registros de Softwares                                                      | 29        |
| 3.3      | Resumo de Trabalhos Científicos                                                     | 30        |
| 3.4      | Considerações sobre os Trabalhos Relacionados                                       | 32        |
| <b>4</b> | <b>Desenvolvimento do Software</b>                                                  | <b>33</b> |
| 4.1      | Construção do Projeto de Software                                                   | 33        |
| 4.1.1    | Requisitos Funcionais e Não Funcionais                                              | 33        |
| 4.1.2    | Diagramas                                                                           | 35        |
| 4.1.2.1  | Casos de Uso                                                                        | 35        |
| 4.1.2.2  | Classes                                                                             | 36        |
| 4.1.2.3  | Entidade Relacionamento Estendido                                                   | 37        |
| 4.2      | Definição da Arquitetura                                                            | 38        |
| 4.2.1    | Tecnologias e Frameworks                                                            | 38        |

|          |                                                                   |           |
|----------|-------------------------------------------------------------------|-----------|
| 4.2.1.1  | React Native com TypeScript . . . . .                             | 39        |
| 4.2.1.2  | Expo . . . . .                                                    | 39        |
| 4.2.1.3  | API Rest . . . . .                                                | 39        |
| 4.2.1.4  | NestJs com TypeScript . . . . .                                   | 40        |
| 4.2.1.5  | PrismaIO . . . . .                                                | 40        |
| 4.2.1.6  | PostgreSQL . . . . .                                              | 40        |
| 4.2.2    | Modelo Arquitetural . . . . .                                     | 40        |
| 4.3      | Prototipagem das Telas . . . . .                                  | 41        |
| 4.3.1    | Telas Iniciais . . . . .                                          | 42        |
| 4.3.2    | Telas <i>Home</i> . . . . .                                       | 42        |
| 4.3.3    | Telas Cursos . . . . .                                            | 44        |
| 4.3.4    | Telas Turmas . . . . .                                            | 45        |
| 4.3.5    | Tela Perfil . . . . .                                             | 46        |
| 4.4      | Implementação da API . . . . .                                    | 47        |
| 4.4.1    | Camada de Controle (Controller) . . . . .                         | 48        |
| 4.4.2    | Camada de Serviço (Service) . . . . .                             | 50        |
| 4.4.3    | Camada de Repositório (Repository) . . . . .                      | 51        |
| 4.4.4    | Banco de Dados (BD) . . . . .                                     | 53        |
| 4.5      | Implementação do Aplicativo . . . . .                             | 54        |
| 4.5.1    | Arquitetura de Componentes . . . . .                              | 55        |
| 4.5.2    | Integração com a API . . . . .                                    | 56        |
| <b>5</b> | <b>Conclusões e Trabalhos Futuros . . . . .</b>                   | <b>58</b> |
|          | <b>Referências . . . . .</b>                                      | <b>60</b> |
|          | <b>APÊNDICE A Requisitos de Funcionalidades Futuras . . . . .</b> | <b>64</b> |

# 1

## Introdução

A sociedade atual é marcada por uma busca incessante por prazer imediato, impulsionada pela dopamina, uma das principais substâncias neuroquímicas relacionadas à recompensa. Essa busca pelo prazer rápido é amplamente explorada pelos meios de entretenimento mais populares da atualidade, onde a captura e retenção da atenção dos usuários são cruciais. Expressões como "arrasta pra cima", tornaram-se símbolos dessa metodologia de captação de atenção, popularizada por plataformas como TikTok e Kwai. Essa abordagem tem se mostrado eficaz em cativar e prender a atenção das pessoas por horas a fio, em tal nível que até as plataformas já consolidadas no ramo do entretenimento implementaram suas próprias versões dessa abordagem, como o YouTube com o Shorts e o Instagram com o Reels. No entanto, os efeitos adversos dessa distribuição rápida de dopamina podem ser prejudiciais para os aspectos psicossociais do indivíduo, principalmente na infância e adolescência, principais fases do desenvolvimento, como trabalhado por Saul Machado em sua monografia ([MACHADO, 2023](#)) e reforçado por Glau Gasparetto e Julia Moioli em sua matéria no Tilt UOL, canal de notícias relacionadas a tecnologia ([GASPARETTO; MOIÓLI, 2023](#)).

Diante desse panorama, surge a indagação: como utilizar de maneira construtiva essa abordagem de obtenção rápida de recompensa para um propósito benéfico no contexto educacional? A resposta a essa pergunta materializa-se no desenvolvimento de uma plataforma educacional inovadora. Um diferencial chave do projeto é seu caráter aberto e modelável, projetado para que educadores de diferentes áreas possam criar e estruturar seus próprios cursos. Embora este trabalho utilize o ensino de lógica de programação como estudo de caso inicial para demonstrar a eficácia do modelo. Essa flexibilidade possibilita sua aplicação em diversas outras áreas do conhecimento. A plataforma fundamenta-se na utilização de um layout vertical, apresentando módulos e submódulos de aprendizagem, onde, cada submódulo incorporará mini-games, proporcionando uma abordagem lúdica para o processo de aprendizagem, com potencial para envolver e engajar o estudante além de potencializar o processo de aprendizagem e assim ampliar significativamente a retenção do conhecimento, conforme sustentado em diversos estudos ([FARDO, 2013](#); [PIRES](#)

*et al.*, 2019; GONÇALVES; AZEVEDO; OLIVEIRA, 2022). A transição entre os módulos será efetuada através do já consagrado termo "arrasta pra cima", oferecendo aos usuários a oportunidade de adquirir conhecimento de maneira rápida, divertida e altamente memorável. Essa escolha de design não é meramente estética, mas sim uma decisão pedagógica que busca aliar a familiaridade das interfaces de redes sociais a um objetivo de aprendizagem, reduzindo a barreira cognitiva inicial e promovendo a adesão do estudante à ferramenta educacional.

O universo dos jogos serve como principal inspiração para a gamificação, atestando sua popularidade através do sucesso de empresas bilionárias nesse setor. Diante desse panorama, surge a indagação natural: se as mecânicas e os elementos que tornam os jogos capazes de cativar bilhões de pessoas globalmente são tão eficazes, por que não aplicá-los como ferramentas de aprendizagem? É nesse ponto que a gamificação, ao trazer esses aspectos para contextos educacionais, demonstra ser extremamente eficiente, já que potencializa o alto grau de fixação das informações (FARDO, 2013).

No cenário contemporâneo, a computação emerge como a área profissional de maior expansão global, destacando-se por sua complexidade e demanda crescente por profissionais qualificados, no Brasil essa insuficiência profissional é evidenciada através do panorama de talentos construído pelo Google for Startups (STARTUPS, 2023). A área da tecnologia da informação, através dos produtos de software, são aplicados às demais áreas do conhecimento (RAFALSKI *et al.*, 2022), visto que ainda estamos vivendo na era da 3<sup>o</sup> revolução industrial, a revolução informacional (OLIVEIRA, 2022). Contudo, as dificuldades encontradas em áreas de alto raciocínio abstrato, como a computação, apontam para a necessidade premente de estratégias inovadoras no processo de ensino e aprendizagem.

O cerne desta pesquisa parte da premissa de que os mecanismos de recompensa rápida, explorados por formatos de entretenimento efêmeros, podem ser canalizados para fins educacionais produtivos. Nesse contexto, o trabalho propõe uma plataforma lúdica e adaptável, que utiliza o potencial da gamificação e de seus elementos como catalisadores de um aprendizado inovador, sendo sua aplicação e viabilidade demonstradas através de um estudo de caso no ensino de lógica de programação.

## 1.1 Objetivos

### 1.1.1 Objetivo geral

O objetivo geral deste trabalho é desenvolver uma plataforma móvel lúdica e adaptável, baseada em gamificação, que sirva como ferramenta de apoio ao processo de ensino-aprendizagem. A plataforma deve ser projetada para permitir a criação e implementação de cursos por parte dos próprios usuários, viabilizando sua aplicação em diversas áreas do conhecimento.

### 1.1.2 Objetivos específicos

- Analisar o impacto de uma plataforma lúdica no processo de aprendizagem, por meio de estudos científicos existentes.
- Identificar aplicativos relacionados que aplicam a gamificação no processo de aprendizagem.
- Projetar a parte conceitual do aplicativo, definindo a arquitetura e especificando requisitos.
- Desenvolver o aplicativo para dispositivos móveis.

## 1.2 Metodologia

Este trabalho apresenta uma natureza exploratória. Para a sua elaboração e desenvolvimento, foi realizado um estudo acerca das aplicações da gamificação no processo de aprendizagem, além de uma revisão de mercado, para selecionar e analisar os aplicativos existentes que atuam no ensino, usando esse mesmo tipo de abordagem, onde foi possível identificar as funcionalidades individuais de cada um deles e reforçar a justificativa para o desenvolvimento de uma nova plataforma contendo tais funcionalidades, além de apresentar novos recursos e um estilo de funcionamento diferente dos encontrados atualmente. A justificativa para a abordagem exploratória deste projeto reside na necessidade de coletar dados de aplicações já existentes, para realizar a criação do aplicativo, elaborado com base em um levantamento de requisitos e na construção da conceituação e diagramação. A pesquisa pode ser considerada descritiva, visto que irá apresentar todas as etapas do desenvolvimento do produto de software, também pode ser caracterizada como aplicada, uma vez que se propôs a desenvolver um produto para resolver um problema específico. As etapas para o desenvolvimento do projeto foram as seguintes:

- Investigação aprofundada sobre os temas de ensino gamificado e aprendizagem lúdica para introduzir o assunto e estabelecer a fundamentação teórica.
- Análise de mercado e avaliação dos aplicativos disponíveis com propostas semelhantes ao projeto.
- Busca no INPI por programas de computador que possuem pedido de registro e que estejam relacionados ao tema do projeto.
- Revisão de trabalhos científicos por meio da pesquisa em bases de dados.
- Construção do modelo conceitual, levantamento dos requisitos do sistema, elaboração dos diagramas e a prototipagem das telas.
- Estudo das ferramentas de desenvolvimento utilizadas no processo de construção da plataforma.

## 1.3 Estrutura do Documento

A estruturação deste documento foi realizada através da divisão de capítulos, sendo eles:

- **Capítulo 1** - Introdução: o capítulo atual, que contextualiza o trabalho, apresenta os objetivos e a metodologia abordada.
- **Capítulo 2** - Fundamentação Teórica: aborda os conceitos importantes encontrados, que auxiliam no entendimento do que está sendo proposto para o desenvolvimento deste trabalho.
- **Capítulo 3** - Estudo de Soluções Relacionadas: realiza a descrição dos produtos obtidos através da prospecção de mercado, e relaciona as funcionalidades encontradas com o que é desejado no trabalho proposto.
- **Capítulo 4** - Desenvolvimento do Produto de Software: aborda toda a construção conceitual e prática do desenvolvimento do software, desde a definição da arquitetura, definição dos requisitos, elaboração dos diagramas, construção dos protótipos, e as ferramentas utilizadas para o desenvolvimento do sistema.
- **Capítulo 5** - Conclusões e Trabalhos Futuros: apresenta as considerações finais do trabalho, destacando os resultados alcançados, as limitações encontradas e as possibilidades de continuidade da pesquisa por meio de melhorias e validações futuras do aplicativo.

# 2

## Fundamentação Teórica

Este capítulo apresenta a fundamentação teórica necessária para entender a proposta deste trabalho. Nele são discutidas as principais dificuldades enfrentadas no processo de aprendizagem da lógica de programação, bem como o papel da gamificação no ensino e a utilização de jogos e softwares gamificados como ferramentas de apoio. A intenção é fornecer o embasamento conceitual que justifica a escolha da abordagem lúdica e reforça sua relevância como estratégia para facilitar a assimilação de conteúdos complexos.

### 2.1 Dificuldade na Aprendizagem de Lógica de Programação

O desafio mais premente enfrentado no ensino e aprendizagem da lógica de programação é a disparidade nos ritmos de aprendizado dos alunos, combinada com turmas numerosas e a falta de motivação. Em muitos casos, os alunos não reconhecem a relevância das atividades propostas e não se envolvem com elas (NATAL *et al.*, 2018). Um estudo aponta o desafio no processo de aprendizagem da programação: a dificuldade dos alunos em manter o mesmo ritmo estabelecido pelos professores nas aulas (GOMES; MENDES, 2007).

A compreensão dos conceitos envolvendo a lógica de programação é crucial para dominar qualquer linguagem de programação e, por essa razão, é introduzida nos estágios iniciais dos cursos da área de computação. Estudar lógica tem o potencial de aprimorar a capacidade de pensamento, formulação de questões e aplicação de soluções a problemas cada vez mais complexos, o que é possibilitado ao desenvolver o pensamento computacional, sendo este apoiado por quatro pilares fundamentais: A decomposição que envolve a divisão de problemas em partes menores, o reconhecimento de padrões que consiste em identificar regularidades em dados ou situações, a abstração que permite focar apenas nos aspectos essenciais de um problema, e os algoritmos que definem o conjunto de instruções necessárias para resolver um problema específico. Em essência, a lógica representa a arte de pensar corretamente (FRIEDRICH *et al.*,

2012).

Um fator que contribui para as dificuldades dos alunos nos cursos relacionados à computação é o fator de "ineditismo". Muitos alunos, especialmente aqueles provenientes do ensino fundamental, têm pouca ou nenhuma exposição prévia à área de computação. Isso faz com que a forma como o conteúdo é apresentado seja muitas vezes opressiva e cansativa, não permitindo tempo suficiente para a assimilação das novas ideias. Embora todos os cursos possuam seus próprios desafios, a familiaridade mínima com o assunto pode facilitar significativamente o processo de aprendizado. Por exemplo, na área da medicina, embora altamente complexa, a maioria das pessoas já teve algum contato com ela, seja através de consultas médicas ou aulas de biologia. No entanto, a lógica necessária aos cursos de computação é normalmente algo completamente novo para os alunos, o que pode contribuir para a evasão durante os estágios iniciais dos cursos de computação, bem como para o prolongamento do tempo médio de conclusão, o que já é uma característica proeminente na área, sendo as dificuldades conceituais e a falta de motivação, fatores frequentemente apontados como barreiras iniciais (SOUZA; BATISTA; BARBOSA, 2016).

Para contornar tais dificuldades, o campo da educação tem explorado cada vez mais as chamadas metodologias ativas. Em oposição ao modelo de ensino tradicional, no qual o aluno atua como um receptor passivo de informações, essas metodologias o posicionam como protagonista do seu próprio processo de aprendizagem. O objetivo é promover maior engajamento, autonomia e capacidade de resolução de problemas por meio de abordagens participativas. Nesse contexto, a gamificação, surge como uma proeminente estratégia alinhada a esses princípios.

## 2.2 Processo da Gamificação no Ensino

A gamificação pode ser definida como uma técnica que consiste em utilizar conceitos, mecânicas e dinâmicas de jogos para motivar os alunos no desempenho de ações não necessariamente relacionadas a jogos (JÚNIOR; BONIATI, 2015). Embora o termo tenha ganhado destaque a partir de 2010 com a adoção de sistemas de recompensas por empresas, sua origem remete a 2002, quando o desenvolvedor britânico Nick Pelling propôs a ideia de aplicar a lógica dos jogos para engajar pessoas na resolução de problemas (NATAL *et al.*, 2018).

Os elementos essenciais dos jogos presentes na gamificação podem ser categorizados em três grupos distintos: dinâmicas, mecânicas e componentes. As dinâmicas abrangem aspectos que proporcionam estrutura, regulamentação e significado à experiência, incorporando conceitos como emoções, narrativa, progressão e interações sociais. Mecânicas referem-se aos elementos que impulsionam a participação do usuário no jogo, incluindo desafios, cooperação, feedback e recompensas. Por fim, os componentes representam as partes específicas do jogo que compõem o sistema global, como conquistas do jogador, avatares, distintivos, coleções e progressão de níveis (NATAL *et al.*, 2018).

Uma solução para os desafios enfrentados na educação é a aplicação da gamificação, que envolve a incorporação desses elementos presentes em jogos, com o objetivo de engajar e motivar o público-alvo (SANTOS *et al.*, 2022). A abordagem da gamificação, nesse contexto, consiste em integrar tais características presentes nos jogos ao processo de ensino e aprendizagem. Dado que os jogos são uma parte comum da vida cotidiana para muitos indivíduos, desde a infância até a fase adulta, eles se tornam familiares e oferecem uma oportunidade de facilitar a aprendizagem por meio de elementos gamificados. A presente proposta se apropria desse conceito, utilizando a gamificação não como um fim em si mesma, mas como uma ponte para conectar os conteúdos educacionais ao universo do estudante, visando aumentar a motivação intrínseca e o tempo de engajamento com o material de estudo.

## 2.3 O Uso da Gamificação e de Softwares Educacionais como Ferramenta de Apoio ao Ensino

O uso de abordagens lúdicas e softwares gamificados como uma ferramenta complementar ao ensino tem ganhado destaque significativo. A gamificação, que consiste na integração de elementos, mecânicas e estruturas de jogos em contextos não próprios, tem sido reconhecida como uma forma eficaz de aumentar o engajamento, a produtividade, a criatividade e a aprendizagem autêntica dos alunos. No entanto, é importante notar que os jogos educativos podem incorrer em uma armadilha, na qual os jogadores se concentram apenas na lógica interna do jogo, em vez de internalizar os conceitos que estão sendo ensinados (KRANZ *et al.*, 2017).

Estudos mostraram que a aprendizagem nem sempre ocorre durante o jogo em si, mas sim quando os jogadores aplicam e internalizam o que aprenderam. Os jogos eletrônicos oferecem vantagens únicas, como liberdade de ação, expectativa de diversão, ambiente livre de riscos e interação com outros participantes. Essas características são especialmente úteis em disciplinas teóricas, como a lógica de programação, onde a dinamicidade proporcionada pelos jogos pode ajudar a superar as dificuldades intrínsecas (KRANZ *et al.*, 2017).

Pesquisas recentes têm demonstrado resultados positivos no uso de jogos e da gamificação para o ensino de programação. Ambientes como o "Tri-Logic" (NATAL *et al.*, 2018) e o "Kodesh" (CAMPOS; GARDIMAN; MADEIRA, 2015) foram desenvolvidos com o intuito de tornar o aprendizado mais interativo e motivador. Essas plataformas utilizam recursos gamificados, como pontos, rankings e medalhas. Os resultados apontados em estudos com essas ferramentas indicam um aumento na motivação e na confiança dos estudantes, além de uma correlação positiva com a diminuição das taxas de reprovação (NATAL *et al.*, 2018; CAMPOS; GARDIMAN; MADEIRA, 2015). Além disso, a competitividade gerada pela gamificação, especialmente através dos *leaderboards*, tem se mostrado eficaz em motivar os estudantes (SANTOS *et al.*, 2022).

Ao apresentar conteúdos de forma mais "casual", os jogos e softwares gamificados facilitam a adaptação e o desenvolvimento dos alunos. Isso permite que os alunos aprendam

no seu próprio ritmo, sem sentir a pressão psicológica imposta por metodologias tradicionais. Em suma, o uso de metodologias ativas, como a gamificação, está contribuindo para a inovação na educação, proporcionando uma forma mais motivadora e eficaz de adquirir conhecimento (CAMARGO; DAROS, 2018).

# 3

## Estudo de Soluções Relacionadas

Neste capítulo, serão apresentadas diversas abordagens e soluções existentes no mercado e na literatura científica que tratam do problema em questão. Na [seção 3.1](#), serão descritas algumas das aplicações disponíveis no mercado, enquanto na [seção 3.2](#) serão analisados os registros de software. A [seção 3.3](#) contém um resumo de trabalhos científicos relacionados. Finalmente, na [seção 3.4](#), serão feitas algumas considerações sobre os trabalhos relacionados.

### 3.1 Prospecção de Mercado

Nesta seção, serão apresentadas algumas aplicações já disponíveis. Para isso, foi realizada uma busca de anterioridade, assim como uma análise desses produtos identificados, a fim de entender quais soluções já estão disponíveis no mercado para abordar o problema em questão. Por meio de termos de busca, disponíveis na [Tabela 1](#), inseridos na Play Store, App Store e Google Chrome, foram identificados 44 aplicativos relevantes e relacionados ao tema de aprendizagem gamificada/lúdica. Com base nos resultados obtidos, escolhemos 8 aplicativos como referência para o desenvolvimento do nosso aplicativo, pois atendem a alguns critérios de seleção.

Os critérios de seleção estabelecidos no momento de realizar a filtragem dos aplicativos selecionados foram: permitir o acesso experimental gratuito sem necessidade de fidelização prévia, possuir aspectos de gamificação, disponibilizar a aplicação em português ou inglês, possuir foco no ensino específico de algo, possibilitar o ensino para quaisquer idades.

A partir desses critérios os aplicativos foram selecionados e, posteriormente, foram examinadas as funcionalidades presentes em cada um deles. Isso foi feito com o intuito de extrair inspiração para desenvolver o nosso aplicativo que apresentará recursos/abordagens inovadoras ou incomuns, para esse tipo de aplicação, centrada na aprendizagem.

Foram eliminados aplicativos que não atenderam aos critérios estabelecidos ou que adotaram a mesma abordagem de algum outro previamente selecionado. Esses aplicativos, embora

pertencentes ao mesmo tema e identificados através dos termos de busca definidos, revelaram-se redundantes ou divergentes da proposta do trabalho. Algumas características presentes nos aplicativos excluídos durante a seleção incluíam restrições ao uso apenas através de assinatura, a ausência de atributos exclusivos e o não atendimento aos critérios de seleção mencionados anteriormente.

Após a seleção dos aplicativos, foram reunidas as funcionalidades identificadas durante a revisão e, em seguida, procederemos com uma comparação das características de cada um deles.

O **Quadro 1** apresenta os aplicativos mencionados, juntamente com suas respectivas URLs de acesso.

Quadro 1 – Aplicativos e links de acesso

| <b>Aplicativo</b>      | <b>Link</b>                                                                                                                                                               |
|------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Mimo                   | < <a href="https://play.google.com/store/apps/details?id=com.getmimo">https://play.google.com/store/apps/details?id=com.getmimo</a> >                                     |
| Programming Hero       | < <a href="https://play.google.com/store/apps/details?id=com.learnprogramming.codecamp">https://play.google.com/store/apps/details?id=com.learnprogramming.codecamp</a> > |
| Scratch                | < <a href="https://play.google.com/store/apps/details?id=org.scratch">https://play.google.com/store/apps/details?id=org.scratch</a> >                                     |
| Duolingo               | < <a href="https://play.google.com/store/apps/details?id=com.duolingo">https://play.google.com/store/apps/details?id=com.duolingo</a> >                                   |
| Lightbot: Code Hour    | < <a href="https://apps.apple.com/br/app/lightbot-code-hour/id873943739">https://apps.apple.com/br/app/lightbot-code-hour/id873943739</a> >                               |
| Quizizz: Play to learn | < <a href="https://play.google.com/store/apps/details?id=com.quizizz_mobile">https://play.google.com/store/apps/details?id=com.quizizz_mobile</a> >                       |
| Raptivity              | < <a href="https://www.raptivity.com/">https://www.raptivity.com/</a> >                                                                                                   |
| Lumosity               | < <a href="https://play.google.com/store/apps/details?id=com.lumoslabs.lumosity">https://play.google.com/store/apps/details?id=com.lumoslabs.lumosity</a> >               |

Fonte: Autor (2024).

A **Tabela 1** exibe todos os termos de busca utilizados, juntamente com o número de resultados obtidos por meio de cada um deles. Em relação à contagem, apenas a primeira ocorrência foi considerada para os resultados que se repetiram em cada busca. Os termos estão dispostos na ordem em que foram utilizados.

A partir da análise dos aplicativos identificados, foi possível extrair algumas características principais encontradas neles que são desejadas para a aplicação aqui proposta, as quais estão destacadas no **Quadro 2**. Utilizando as informações obtidas nesse quadro, elaboramos uma tabela que estabelece a relação entre cada uma dessas características e os aplicativos examinados.

Com base na análise, notou-se que a maior parte dos aplicativos fazem uso de quizzes para a aprendizagem, e apenas alguns fazem uso de outros mini-games para esse fim, sendo que praticamente todos possuem acesso gratuito restrito a recursos específicos da aplicação, tendo acesso total apenas por meio de algum tipo de assinatura. Logo, com a grande variedade de

Tabela 1 – Termos de busca e o número de aplicativos encontrados.

| <b>Termo de Busca</b>            | <b>Resultados</b> |
|----------------------------------|-------------------|
| Educação Gamificada              | 2                 |
| Educação Lúdica                  | 4                 |
| Ensino Gamificado                | 3                 |
| Ensino Lúdico                    | 3                 |
| Programação                      | 4                 |
| Plataforma de Ensino Gamificado  | 2                 |
| Aprendizagem por Mini-games      | 0                 |
| Aplicativos Gamificação Educação | 26                |
| <b>Total</b>                     | <b>44</b>         |

Fonte: Autor (2024).

Quadro 2 – Identificações e funcionalidades

| <b>Identificação</b> | <b>Recurso</b>                                                        |
|----------------------|-----------------------------------------------------------------------|
| R1                   | Utiliza Quizzes / Perguntas e Respostas                               |
| R2                   | Utiliza outros tipos de minigames (ex: blocos lógicos, puzzles, etc.) |
| R3                   | Possibilita o ensino interativo para todas as idades                  |
| R4                   | Permite a criação/adição de "cursos"                                  |
| R5                   | Possibilita o ensino de diversas áreas do conhecimento                |
| R6                   | Separa os "cursos" em módulos mais simplificados                      |
| R7                   | Pontua a conclusão das atividades                                     |
| R8                   | Possui <i>leaderboards</i>                                            |
| R9                   | Disponibiliza versão compatível com Android e IOS                     |
| R10                  | Permite o acesso em mais de um idioma                                 |
| R11                  | Permite a inscrição em determinado "curso"                            |
| R12                  | Disponibiliza as informações de forma vertical                        |
| R13                  | Permite avaliar o "curso" após a conclusão                            |
| R14                  | Possui sistema de avaliação rápida                                    |
| R15                  | Disponibiliza acesso 100% gratuito                                    |

Fonte: Autor (2024).

recursos disponíveis observa-se a necessidade de extrair o que há de melhor nessas aplicações a fim de combina-las para apresentar um aplicativo mais completo e que faça uso de uma maior variedade de mini-games, além de adicionar uma nova forma de exibição do ensino. O [Quadro 3](#) relaciona os aplicativos estudados e as funcionalidades apontadas.

Nas subseções 3.1.1 a 3.1.8, são abordadas as funcionalidades de cada um dos 8 aplicativos analisados. Para elucidar essas funcionalidades, realizamos o download de cada aplicativo e navegamos por suas características.

### 3.1.1 Mimo

O Mimo é uma plataforma de cursos de programação, com os seus exercícios sendo realizados por questionários interativos e criação de blocos de código. Disponibilizam currículos de ensino, projetos guiados, áreas de código onde é possível praticar, certificados para Python,

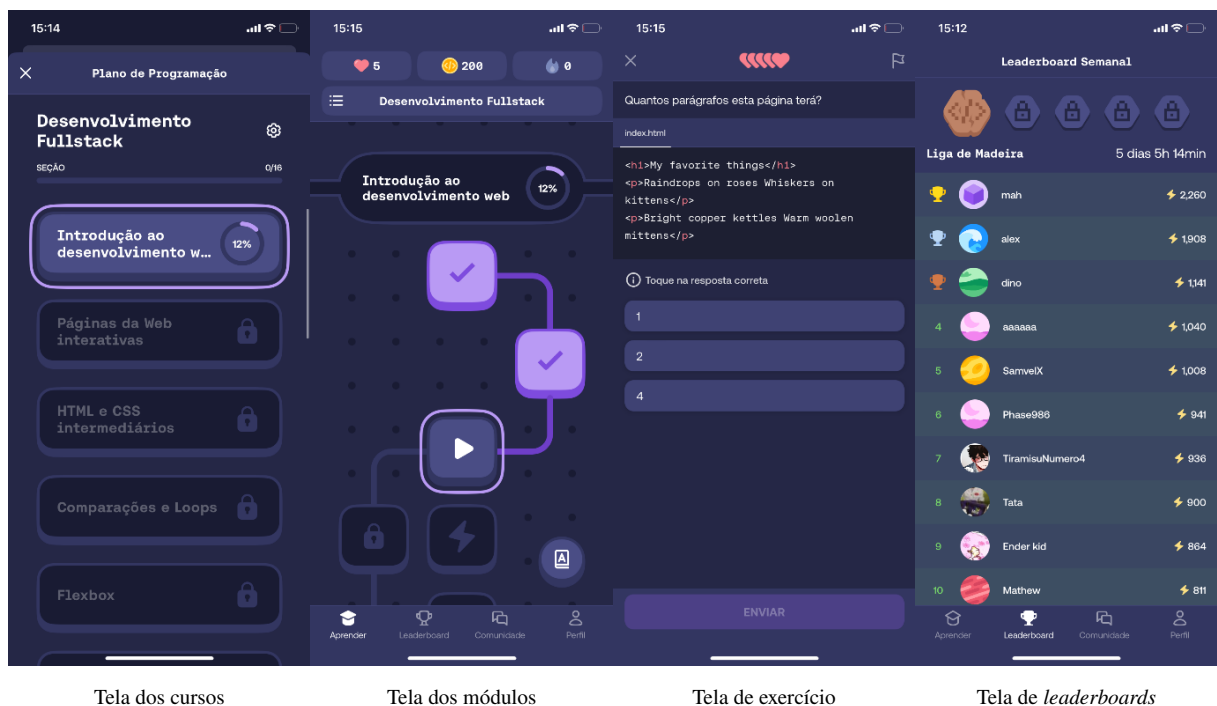
Quadro 3 – Comparação entre os aplicativos e as funcionalidades

| Aplicativo                   | R1 | R2 | R3 | R4 | R5 | R6 | R7 | R8 | R9 | R10 | R11 | R12 | R13 | R14 | R15 |
|------------------------------|----|----|----|----|----|----|----|----|----|-----|-----|-----|-----|-----|-----|
| Mimo                         | X  |    | X  |    | X  | X  | X  | X  | X  | X   | X   | X   |     | X   |     |
| Programming Hero             | X  | X  | X  |    | X  | X  |    |    | X  |     | X   | X   |     | X   |     |
| Scratch                      |    | X  | X  |    |    | X  |    |    |    | X   | X   |     |     | X   | X   |
| Duolingo                     | X  | X  | X  |    | X  | X  | X  | X  | X  | X   | X   | X   |     | X   | X   |
| Lightbot: Code Hour          |    | X  | X  |    |    | X  |    |    |    | X   |     |     |     | X   | X   |
| Quizizz: Play to learn       | X  |    | X  |    | X  |    | X  | X  | X  | X   |     | X   |     | X   | X   |
| Raptivity                    | X  |    | X  | X  | X  |    | X  |    | X  | X   |     |     |     | X   |     |
| Lumosity                     | X  | X  | X  |    | X  |    | X  |    | X  | X   |     | X   |     | X   |     |
| Aplicativo proposto (TikTec) | X  |    | X  | X  | X  | X  | X  | X  | X  | X   | X   | X   | X   | X   | X   |

Fonte: Autor (2024).

JavaScript, HTML, CSS e SQL. A metodologia de ensino consiste na divisão dos cursos em módulos, e esses módulos em submódulos, que possuem uma avaliação no final, onde são apontados os erros e acertos daquele respectivo submódulo. Também utilizam *leaderboards* para pontuar e comparar o nível dos usuários. As principais imagens de telas do aplicativo estão disponíveis na Figura 1. Após o cadastro são disponibilizados os planos de carreira que são compostos pelo ensino de diversas tecnologias, ou é possível também selecionar o ensino de apenas uma linguagem específica. Cada curso é disponibilizado em módulos e submódulos, contendo questionários interativos. <sup>1</sup>

Figura 1 – Telas do aplicativo Mimo



Tela dos cursos

Tela dos módulos

Tela de exercício

Tela de leaderboards

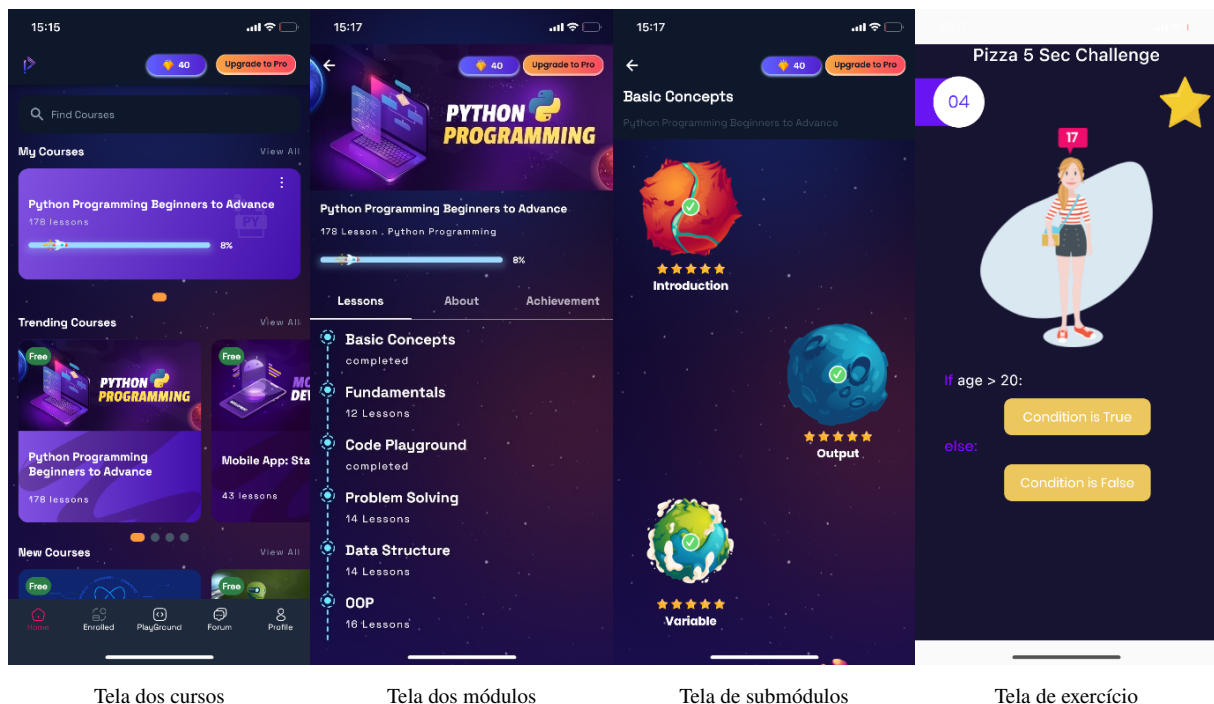
Fonte: Mimo GmbH (2024).

<sup>1</sup> Mimo: <https://play.google.com/store/apps/details?id=com.getmimo>

### 3.1.2 Programming Hero

O Programming Hero é uma plataforma de cursos de programação, com os seus exercícios sendo realizados por questionários interativos, possuindo alguns cursos disponíveis, como C, C++, Python e Java. O aplicativo divide os cursos em módulos e submódulos, possibilitando uma aprendizagem através de exercícios curtos e rápidos. Disponibilizam videoaulas para complementar o ensino. Utilizam também um sistema de *leaderboards* para pontuar e comparar o nível dos usuários, permitem a criação e exportação de projetos web e a emissão de certificados de conclusão dos respectivos cursos. As principais imagens de telas do aplicativo estão disponíveis na Figura 2. Após iniciar o aplicativo, é disponibilizado um playground de programação, além de uma série de cursos para conhecimentos específicos, onde cada curso é composto por módulos, os quais são divididos em lições que são ensinadas por meio de questionários interativos.<sup>2</sup>

Figura 2 – Telas do aplicativo Programming Hero



Fonte: Programming Hero (2024).

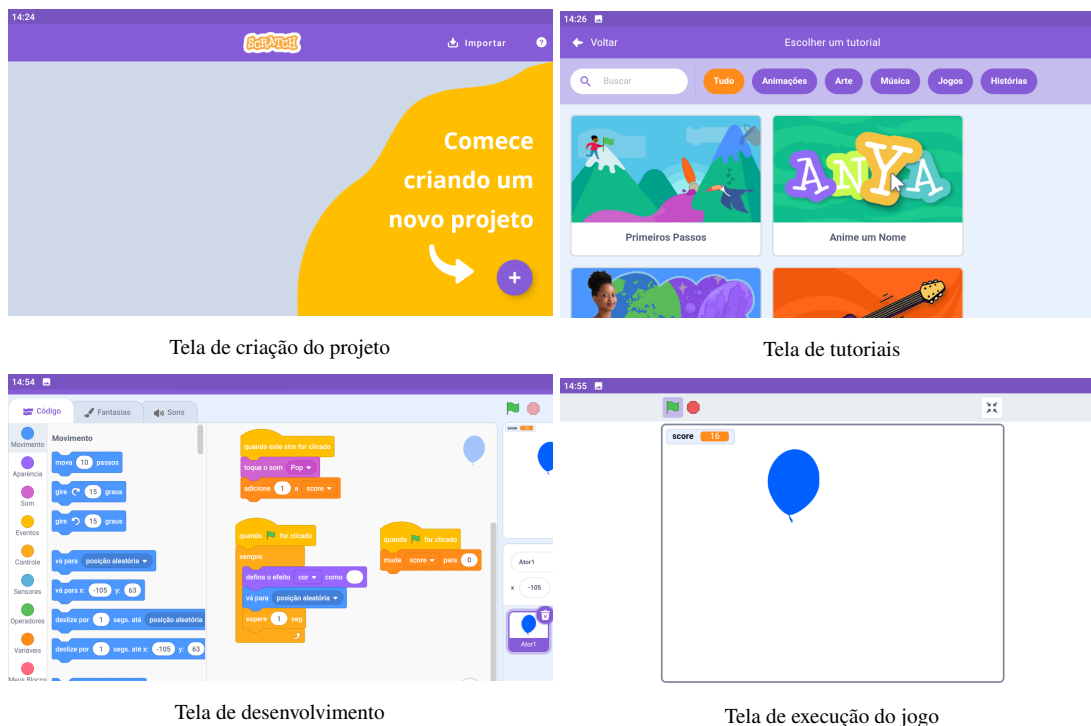
### 3.1.3 Scratch

O Scratch é uma plataforma de ensino originalmente no formato de um Website, mas que atualmente também possui um aplicativo para Android. Esse aplicativo disponibiliza a abordagem de programação/desenvolvimento por meio de blocos lógicos, onde é possível organiza-los para construir aplicativos e jogos de forma online e offline. Disponibiliza a criação e compartilhamento de projetos entre a comunidade de usuários, e a exportação dos projetos para salvamento local. Tem como objetivo ensinar a lógica de programação, tendo como seu público principal, crianças

<sup>2</sup> Programming Hero: <https://play.google.com/store/apps/details?id=com.learnprogramming.codecamp>

e adolescentes, por conta da metodologia utilizada. As principais imagens de telas do aplicativo estão disponíveis na Figura 3. Após iniciar o app, é disponibilizada a criação de projetos, onde será possível utilizar blocos, fantasias e sons para construir e compilar jogos, podendo ser este um projeto próprio, ou um dos tutoriais disponíveis no catálogo.<sup>3</sup>

Figura 3 – Telas do aplicativo Scratch



Fonte: Scratch Foundation (2024).

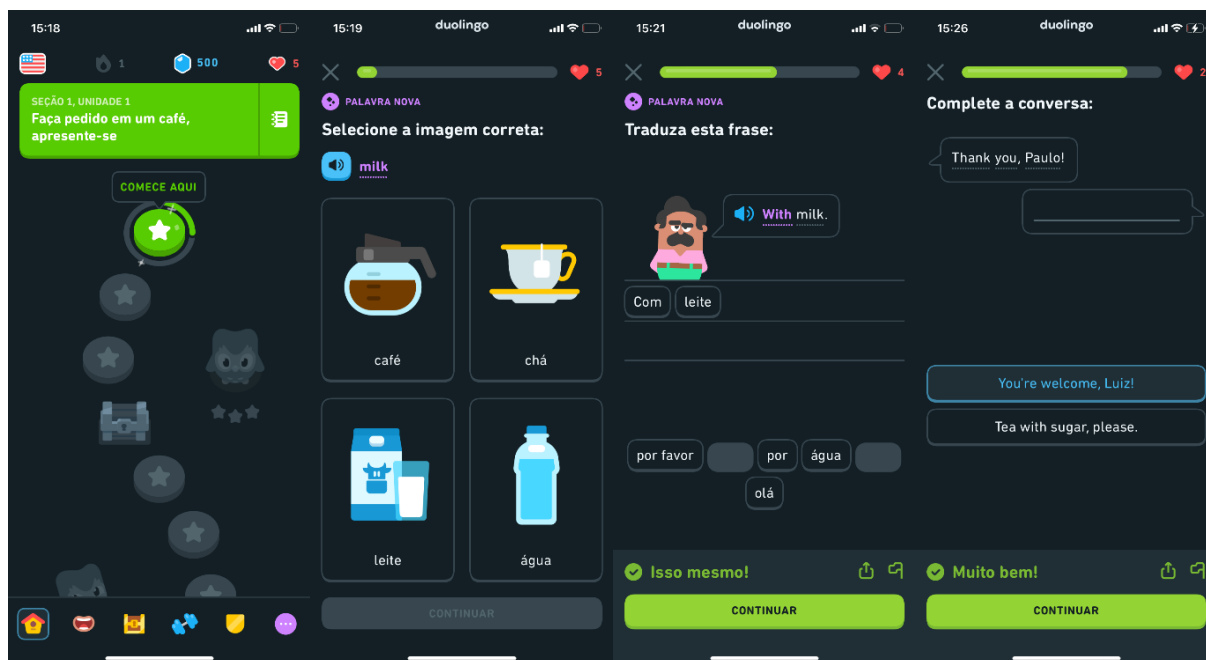
### 3.1.4 Duolingo

O Duolingo é uma plataforma de curso de idiomas gamificada, com alguns tipos de exercícios interativos para os cursos disponíveis. Através de lições rápidas e curtas que abrangem leitura, escrita, escuta e fala, o aplicativo divide a aprendizagem em pequenos módulos, que podem ser completados rapidamente, também disponibiliza ligas de ensino nas quais os alunos podem competir uns com os outros. Sua metodologia de ensino interativa visa a retenção do conteúdo na memória de longo prazo. As principais imagens de telas do aplicativo estão disponíveis na Figura 4. Após o cadastro e a escolha do idioma que será cursado, é disponibilizada a lista de módulos que o usuário deverá completar através de lições específicas, onde cada lição é composta por exercícios interativos.<sup>4</sup>

<sup>3</sup> Scratch: <https://play.google.com/store/apps/details?id=org.scratch>

<sup>4</sup> Duolingo: <https://play.google.com/store/apps/details?id=com.duolingo>

Figura 4 – Telas do aplicativo Duolingo



Tela dos módulos

Tela de exercício tipo 1

Tela de exercício tipo 2

Tela de exercício tipo 3

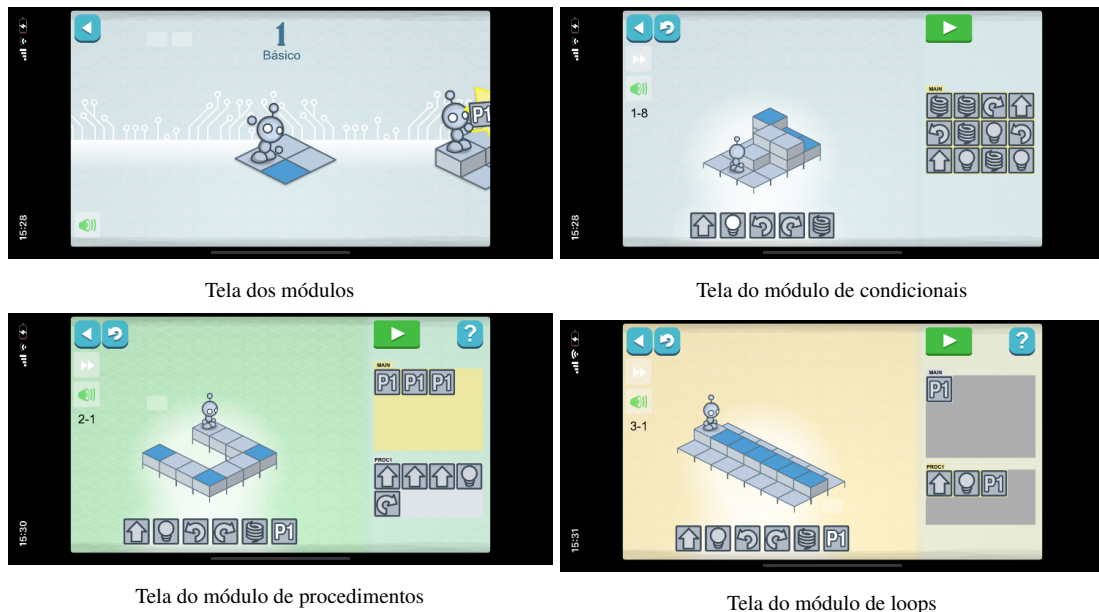
Fonte: Duolingo (2024).

### 3.1.5 Lightbot: Code Hour

O Lightbot: Code Hour é um jogo projetado para ensinar programação de forma lúdica. Através do jogo, os jogadores são desafiados a resolver diversos níveis aplicando lógica de programação. A aprendizagem dos conceitos reais de programação são dados ao guiar um robô para iluminar blocos usando comandos, onde são utilizados espaços para ordenar comandos simples, como andar, pular, virar e ascender. Nos espaços disponíveis, é possível definir procedimentos assim como usar loops. Embora esse aplicativo seja gratuito, possui apenas 3 módulos sendo eles de condicionais, procedimentos e loops. A versão completa, Lightbot: Programming Puzzles, disponibiliza mais módulos. As principais imagens de telas do aplicativo estão disponíveis na Figura 5. Após selecionar o módulo, é disponibilizada a lista de comandos possíveis para realizar com o robô, um bloco main, e um bloco para definição de processos, ambos utilizando esses comandos, além do robô e do trajeto a ser realizado. <sup>5</sup>

<sup>5</sup> Lightbot: Code Hour: <https://apps.apple.com/br/app/lightbot-code-hour/id873943739>

Figura 5 – Telas do aplicativo Lightbot: Code Hour



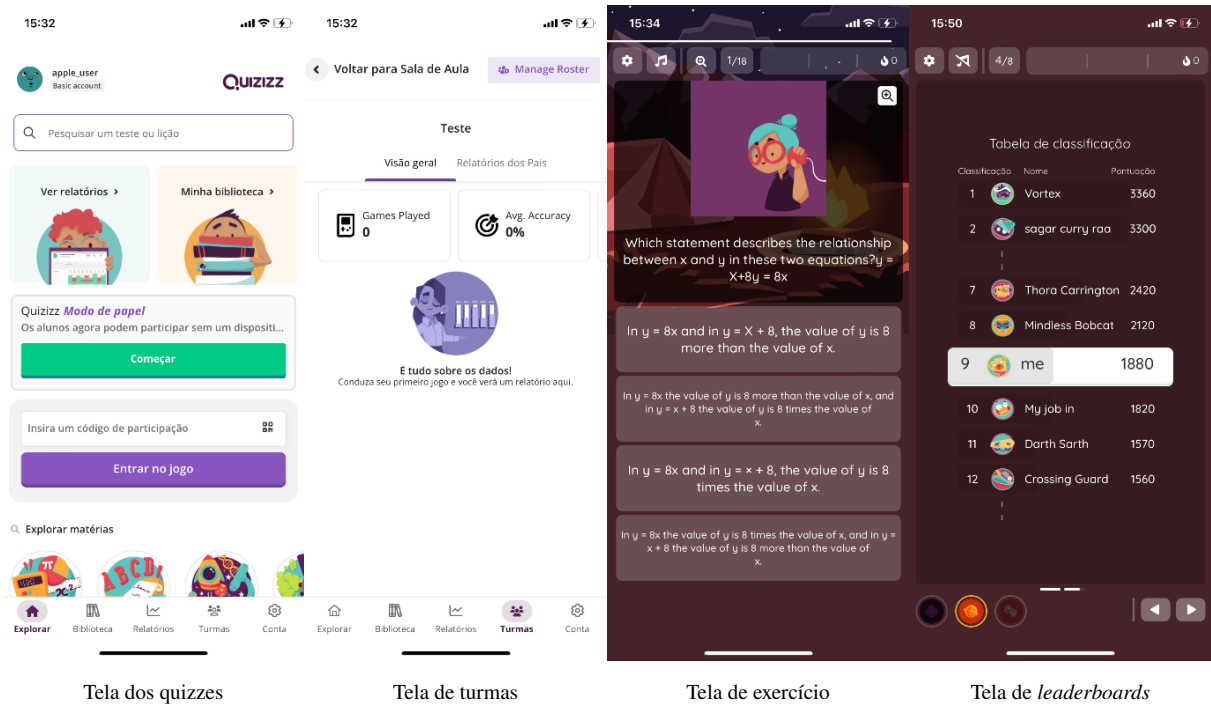
Fonte: LightBot Inc (2024).

### 3.1.6 Quizizz: Play to learn

O Quizizz é uma plataforma de quizzes interativos, voltada para a aquisição de conhecimento, seja de forma independente ou participando de questionários, tarefas e apresentações em grupo, no modo presencial ou remoto. São disponibilizadas a criação e a hospedagem de questionário de forma gratuita, e eles abrangem matemática, inglês, ciências, história, geografia, línguas e temas de conhecimento geral, também é possível criar turmas onde podem ser atribuídos diversos questionários, sejam eles desenvolvidos por você ou terceiros. Para o uso dos participantes, não é necessário baixar o aplicativo, pois é disponibilizado um URL ao vivo para aquela seção. As principais imagens de telas do aplicativo estão disponíveis na Figura 6. Após o cadastro, é disponibilizado o menu de opções de atividade, podendo ingressar em quizzes e lições através do seu respectivo código, selecionar um tópico e participar a quaisquer quiz e lição disponível, ou criar uma turma onde pode associar quizzes e lições a ela. <sup>6</sup>

<sup>6</sup> Quizizz: Play to learn: [https://play.google.com/store/apps/details?id=com.quizizz\\_mobile](https://play.google.com/store/apps/details?id=com.quizizz_mobile)

Figura 6 – Telas do aplicativo Quizizz: Play to learn



Fonte: Quizizz Inc (2024).

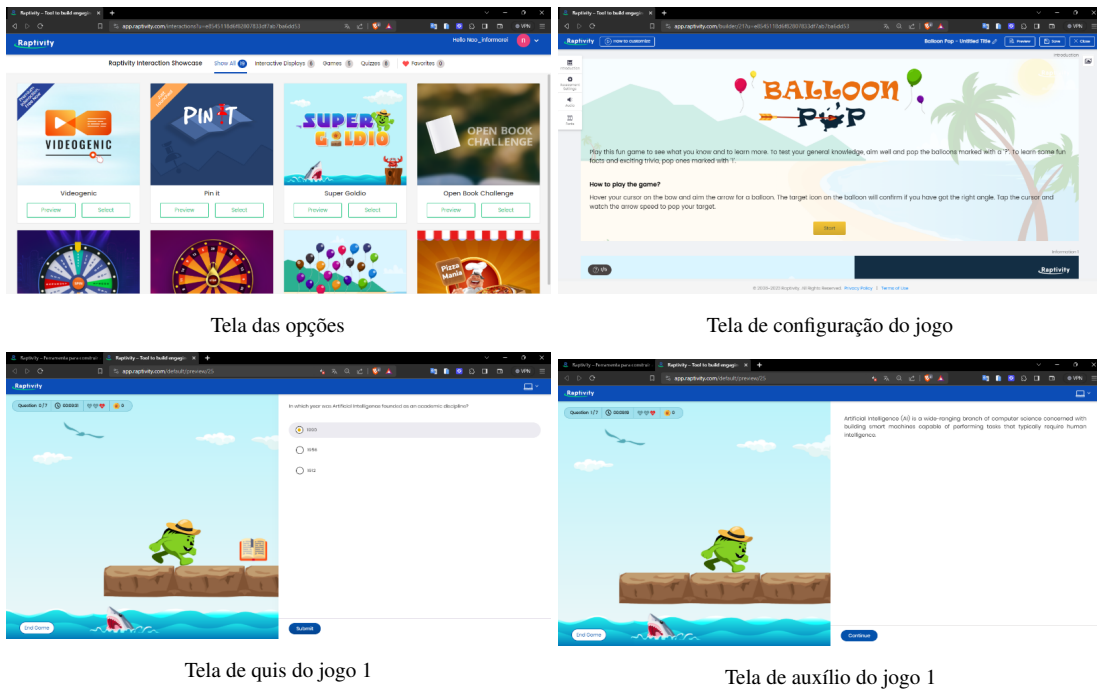
### 3.1.7 Raptivity

O Raptivity é uma plataforma web voltada para a construção de experiências interativas de aprendizagem online, com algumas funcionalidades que podem ser exploradas por professores e alunos. Embora seja uma plataforma web, após a criação dos "cursos" é possível executar os jogos e quizzes em quaisquer dispositivos. Os mini-games disponíveis nesta plataforma servem como um suporte para a distribuição da aprendizagem, pois não é através deles que ela é realizada, eles apenas tornam o processo de aprendizagem mais interessante, já que usam esses mini-games como um prêmio por ter respondido algo corretamente. As principais imagens de telas do aplicativo estão disponíveis na Figura 7. Após o cadastro, é disponibilizado o menu com as telas interativas, jogos e testes, onde o usuário pode selecionar o recurso desejado, para criar o seu próprio "curso" utilizando-o. Porém essa funcionalidade é disponibilizada apenas através de assinatura, tendo como único recurso gratuito a participação desses "cursos" após serem criados.

7

<sup>7</sup> Raptivity: <https://www.raptivity.com/>

Figura 7 – Telas do aplicativo Raptivity



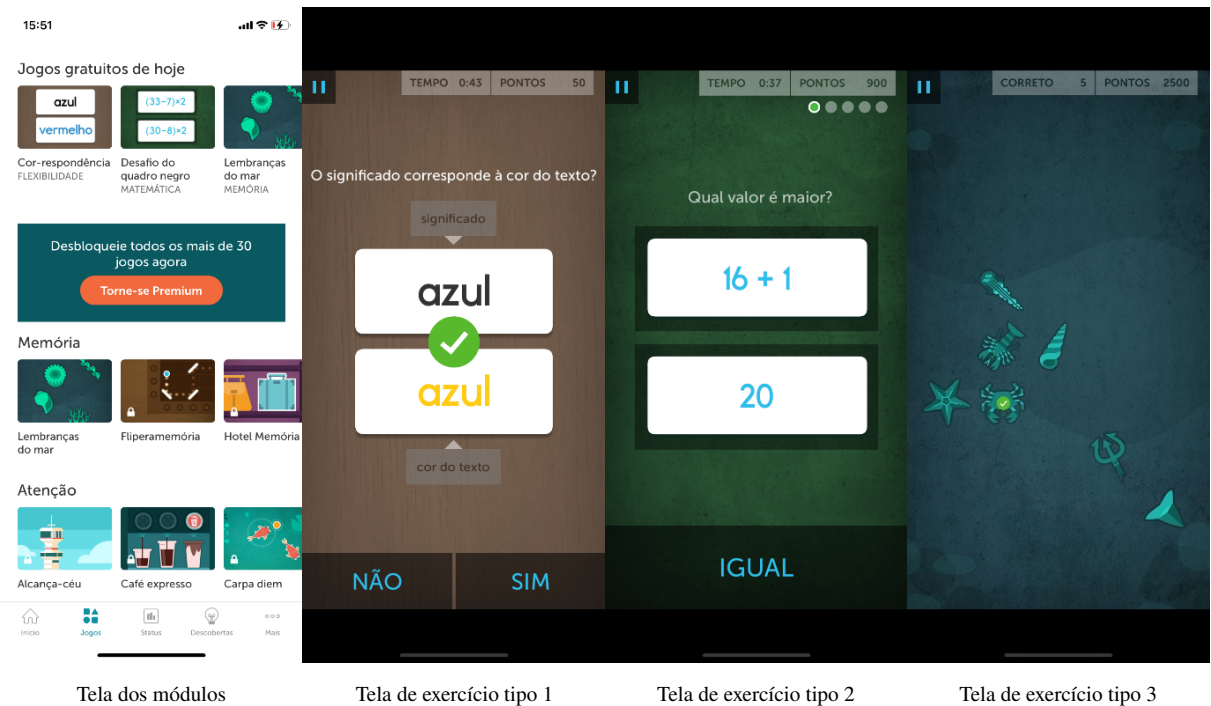
Fonte: Harbinger Group (2024).

### 3.1.8 Lumosity

O Lumosity é uma plataforma de treinamento cognitivo, voltada para o aprimoramento das habilidades mentais, abrangendo áreas como memória, velocidade, flexibilidade e resolução de problemas. Inicialmente o usuário preenche um fit test de 10 minutos para ajustar as pontuações individuais básicas e comparar esse desempenho com indivíduos da mesma faixa etária. Disponibilizam mini-games para cada uma dessas áreas, possibilitando através deles que a execução do treinamento não se torne repetitiva. Embora não seja uma aplicação voltada para o ensino de cursos específicos e não permitam a adição dos mesmos, ela fornece análises detalhadas, destacando pontos fortes e áreas para aprimoramento, oferecendo uma compreensão aprofundada dos seus padrões cognitivos à medida que avança nos jogos, e categoriza os tipos de aprendizagem que possuem, em módulos específicos. As principais imagens de telas do aplicativo estão disponíveis na [Figura 8](#). Após o fit test inicial, são disponibilizados diariamente 3 jogos gratuitos com áreas de foco cognitivo distintas para o processo de autodesenvolvimento, sendo que as outras funcionalidades são disponibilizadas apenas através de assinatura.<sup>8</sup>

<sup>8</sup> Lumosity: <https://play.google.com/store/apps/details?id=com.lumoslabs.lumosity>

Figura 8 – Telas do aplicativo Lumosity



Fonte: Lumos Labs, Inc (2024).

### 3.2 Base de Registros de Softwares

Foi realizada uma busca por registros de software relacionados ao tema do projeto no INPI. Uma síntese dos resultados obtidos são exibidos nessa seção. Para que a realização da busca fosse mais abrangente, foram utilizados termos mais amplos, tais quais, são exibidos no [Quadro 4](#). A pesquisa retornou 4 resultados relevantes, que contém artigos com informações detalhadas da aplicação, que são exibidos no [Quadro 5](#).

Quadro 4 – Termos de busca do INPI.

| <b>Termo de Busca</b>      |
|----------------------------|
| Lúdico                     |
| Lúdica                     |
| Gamificada                 |
| Gamificado                 |
| Gamificação                |
| Plataforma de Aprendizagem |
| Plataforma de Ensino       |
| Aprendizagem Jogo          |
| Ensino Jogo                |

Fonte: Autor (2024).

A partir da pesquisa por registro de software realizada no INPI, foi conduzida uma busca por estudos acadêmicos relacionados a esses respectivos registros, os quais podem estar ausentes

Quadro 5 – Resultados obtidos através da busca na base do INPI.

| Pedido              | Título                   |
|---------------------|--------------------------|
| BR 51 2022 000398 8 | Semio em jogo            |
| BR 51 2023 001059 6 | SQL Planet               |
| BR 51 2016 001119 0 | Jaspion                  |
| BR 51 2020 001395 3 | NiO (Nivelamento Online) |

Fonte: Autor (2024).

na pesquisa executada nas bases acadêmicas através da string de busca utilizada, pois, embora possuam relação com os tópicos principais do projeto, as suas palavras-chaves são de cunho restritivo ao tema abordado por eles. No [Quadro 6](#), estão resumidos os artigos dos registros de software.

Quadro 6 – Resumo dos estudos acadêmicos obtidos a partir dos registros de software encontrados.

| Título                   | Resumo                                                                                                                                                                                                                                                       | Referência                                       |
|--------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------|
| Semio em jogo            | O Aplicativo consiste em um site que tem como objetivo gamificar a aprendizagem de semiologia e semiotécnica, através do uso de Quizzes.                                                                                                                     | ( <a href="#">MATIAS et al., 2023</a> )          |
| SQL Planet               | O Aplicativo consiste em um jogo que tem como objetivo ensinar a linguagem SQL. Para progredir no jogo, o usuário deve controlar um personagem apenas por meio de comandos nessa linguagem.                                                                  | ( <a href="#">BATISTA et al., 2019</a> )         |
| Jaspion                  | O Aplicativo consiste em um jogo integrado ao mundo virtual OpenSim, que tem como objetivo o ensino de redes de computadores. Neste jogo, o usuário deve controlar um personagem que se aventura pelo mundo, a fim de completar missões e responder quizzes. | ( <a href="#">KRASSMANN; MEDINA, 2016</a> )      |
| NiO (Nivelamento Online) | O Aplicativo consiste em um jogo multiplataforma que tem como objetivo gamificar a aprendizagem de matemática, física e outras áreas do conhecimento, através do uso de Quizzes.                                                                             | ( <a href="#">SANDE; SANDE; CARVALHO, 2021</a> ) |

Fonte: Autor (2024).

### 3.3 Resumo de Trabalhos Científicos

Nesta seção é apresentado um resumo dos estudos acadêmicos identificados relacionados aos tópicos principais do projeto, aplicativos de aprendizagem gamificada. Para este fim, foram utilizadas as bases de pesquisa, Periódicos Capes, Google Acadêmico e Scielo, através da string de busca "(plataforma OR aplicativo OR app) AND (ensino OR aprendizagem) AND (gamificado OR gamificada OR gamificação OR ludico OR ludica OR jogo OR mini-game OR minigame OR mini game)". Ao realizar a pesquisa nas bases, foram selecionados um total de 7 artigos, pertinentes ao tema da pesquisa, os quais estão resumidos no [Quadro 7](#).

Usando as características observadas na análise dos aplicativos obtidos através da prospecção de mercado, podemos realizar um comparativo com os aplicativos encontrados na

Quadro 7 – Resumo dos resultados obtidos através da busca nas bases de pesquisa.

| Titulo                                                                                                                               | Resumo                                                                                                                                                                                                                                                                   | Referência                        |
|--------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------|
| Jogo na Educação Matemática: Desenvolvimento de um RPG para trabalhar o conceito de moeda no Ensino Fundamental                      | O Aplicativo consiste em um jogo feito por meio da plataforma RPG Maker VX. Tem como objetivo ensinar o conceito e os cálculos relacionados a moeda/dinheiro através de 25 perguntas, sendo divididas em questões de múltipla escolha e questões de verdadeiro ou falso. | (SCHIMIGUEL <i>et al.</i> , 2016) |
| Tecnologias Móveis e o Estudo da Tabuada: Reflexões sobre o Uso do Jogo Math Duel                                                    | O Aplicativo consiste em um jogo de duelo matemático, e tem como objetivo incentivar o ensino da tabuada por meio de disputas entre dois jogadores.                                                                                                                      | (ROSA; PRETTO; BULEGON, 2019)     |
| Criação de jogo matemático digital com crianças e jovens surdos: contribuições da pedagogia visual                                   | O Aplicativo consiste em um jogo feito por meio do Scratch, descrito na <a href="#">subseção 3.1.3</a> , e tem como objetivo ensinar frações matemáticas através de mini-games.                                                                                          | (ALMEIDA; SILVA; LINS, 2019)      |
| Q-LIBRAS: um jogo educacional para estimular alunos surdos à aprendizagem de Química                                                 | O Aplicativo consiste em um jogo que busca gamificar a aprendizagem por meio quizzes, e tem como objetivo ensinar a classificação dos elementos da tabela periódica, através de textos ou libras.                                                                        | (ROCHA <i>et al.</i> , 2019)      |
| O Jogo Digital Quiz PG nas Aulas de Matemática: possibilidades para o Ensino e Aprendizagem de Progressão Geométrica                 | O Aplicativo consiste em um jogo que busca gamificar a aprendizagem por meio quizzes, e tem como objetivo ensinar a progressão geométrica.                                                                                                                               | (OLIVEIRA; FERREIRA, 2021)        |
| Gamificação em plataformas educacionais: potencializando o ensino de física através do Nivelamento Online                            | O Aplicativo está descrito no <a href="#">Quadro 6</a> .                                                                                                                                                                                                                 | (NEVES; MIYAHARA; SANTOS, 2023)   |
| A utilização da plataforma Kahoot como ferramenta de gamificação: uma contribuição para o ensino e a aprendizagem na educação básica | O Aplicativo consiste em uma plataforma que busca gamificar a aprendizagem por meio quizzes dinâmicos, e tem como objetivo proporcionar a criação de quizzes personalizados.                                                                                             | (SILVA; FERREIRA, 2022)           |

Fonte: Autor (2024).

pesquisa conduzida nas bases. O [Quadro 8](#) relaciona os aplicativos de trabalhos científicos e as funcionalidades do aplicativo proposto.

Quadro 8 – Comparação entre os aplicativos de trabalhos científicos e as funcionalidades

| Aplicativo                   | R1 | R2 | R3 | R4 | R5 | R6 | R7 | R8 | R9 | R10 | R11 | R12 | R13 | R14 | R15 |
|------------------------------|----|----|----|----|----|----|----|----|----|-----|-----|-----|-----|-----|-----|
| Produto Final                | X  | X  |    |    |    |    | X  |    |    |     |     |     |     | X   |     |
| Math Duel                    | X  |    | X  |    |    |    |    |    | X  |     |     | X   |     | X   |     |
| Star Fraction Game           |    | X  |    |    |    |    | X  |    |    |     |     |     |     | X   |     |
| Q-LIBRAS                     | X  |    | X  |    |    |    | X  |    |    |     |     |     |     | X   |     |
| Quiz PG                      | X  |    | X  |    |    |    | X  |    |    |     |     | X   |     | X   |     |
| NiO (Nivelamento Online)     | X  |    | X  | X  | X  | X  | X  |    |    | X   | X   |     |     | X   |     |
| Kahoot                       | X  |    | X  |    | X  |    | X  | X  | X  | X   |     | X   |     | X   |     |
| Aplicativo proposto (TikTec) | X  |    | X  | X  | X  | X  | X  | X  | X  | X   | X   | X   | X   | X   | X   |

Fonte: Autor (2024).

Todos os aplicativos de artigos científicos encontrados foram analisados através da leitura dos mesmos, além de ter suas funcionalidades testadas, quando havia a possibilidade de acessar a aplicação. O [Quadro 8](#) apresenta esses aplicativos, distribuídos na mesma ordem que possui o

## Quadro 7.

### 3.4 Considerações sobre os Trabalhos Relacionados

A partir da prospecção de mercado, da busca por registros de software realizada na base de patentes do INPI e dos trabalhos científicos encontrados através da busca realizada nas bases de pesquisa, foram analisadas as aplicações encontradas, estudadas as suas funcionalidades e comparadas ao aplicativo desenvolvido. Com base em toda a análise realizada, conclui-se que o diferencial do aplicativo TikTec, em comparação com os aplicativos encontrados nas diferentes pesquisas realizadas, consiste em ser uma aplicação mais abrangente. Isso se deve à sua capacidade de ser utilizada em outros segmentos do ensino, permitindo aos usuários a adição de cursos próprios. Por meio dos mini-games a serem disponibilizados na biblioteca do aplicativo, os usuários poderão configurar seus respectivos cursos.

Além disso, o TikTec apresenta uma potencial inovação ao usar uma variedade de mini-games não apenas como um meio de recompensar os estudantes, mas como a própria forma de transmitir os conteúdos. Com o propósito de gerar familiaridade para o usuário, as informações e os mini-games são exibidos através da rolagem vertical, mantendo o gesto "arrasta pra cima", que já é familiar para os usuários.

# 4

## Desenvolvimento do Software

Neste capítulo envolverá a aplicação prática da base teórica adquirida durante o desenvolvimento deste trabalho, visando atender às necessidades identificadas. Para garantir um desenvolvimento eficiente, é crucial estabelecer claramente as tarefas.

### 4.1 Construção do Projeto de Software

Nesta fase, foi realizado o levantamento dos requisitos do software, incluindo os funcionais, não funcionais, bem como a definição de seus respectivos níveis de prioridade. Também serão apresentados os diagramas necessários para a delimitação do escopo do projeto e sua conceituação.

#### 4.1.1 Requisitos Funcionais e Não Funcionais

A definição de requisitos de um software é a forma de descrever as características necessárias para o sistema em questão, definir quais são as suas funcionalidades e as suas respectivas restrições. A classificação dos requisitos é normalmente realizada em:

- Funcionais, que são os responsáveis por discriminar os serviços que o sistema deve fazer, a sua reação para determinadas entradas de dados e o seu comportamento em certas situações, mas, essa definição é subjetiva para cada software, e os seus *stakeholders*;
- Não Funcionais, que são os responsáveis por discriminar as restrições do sistema, eles não se relacionam diretamente com os serviços, mas, sim, a características específicas do sistema, como usabilidade, confiabilidade e desempenho;
- Inversos, que são os responsáveis por definir o que não pode ocorrer no sistema, e é mais normalmente utilizado em sistemas críticos, como bancários e em tempo real.

Considerando o tempo disponível para o desenvolvimento do MVP, o escopo deste trabalho foi focado nas funcionalidades essenciais para o usuário que possui papel de "aluno". Requisitos importantes, mas não cruciais para a experiência inicial, como os painéis administrativos para gerenciamento de conteúdo por parte de professores e administradores, foram definidos e catalogados para futuras versões do projeto. A lista completa com estes requisitos pode ser consultada no [Apêndice A](#).

Nos [Quadro 9](#) e [Quadro 10](#) são listados os requisitos funcionais e não funcionais do software, respectivamente.

Quadro 9 – Requisitos Funcionais

| Código | Título                     | Nível de Prioridade | Descrição                                                                                                                    |
|--------|----------------------------|---------------------|------------------------------------------------------------------------------------------------------------------------------|
| RF01   | Cadastrar Aluno            | Alta                | O sistema deve permitir o cadastro de alunos.                                                                                |
| RF02   | Realizar Login             | Alta                | O sistema deve permitir o acesso dos alunos.                                                                                 |
| RF03   | Recuperar Acesso           | Alta                | O sistema deve permitir a redefinição de senha.                                                                              |
| RF04   | Listar Turmas              | Alta                | O sistema deve permitir a listagem das turmas matriculadas, turmas disponíveis, além de permitir a busca pelo nome da turma. |
| RF05   | Acessar Turmas             | Alta                | O sistema deve permitir o acesso das turmas, disponibilizando assim, todas as informações nela contidas.                     |
| RF06   | Realizar Matrícula         | Alta                | O sistema deve permitir a matrícula do aluno em uma turma.                                                                   |
| RF07   | Listar Cursos              | Alta                | O sistema deve permitir a listagem dos cursos iniciado e disponíveis, além de permitir a busca pelo nome do curso.           |
| RF08   | Acessar Cursos             | Alta                | O sistema deve permitir o acesso dos cursos, disponibilizando assim, todas as informações nele contidas.                     |
| RF09   | Iniciar Cursos             | Alta                | O sistema deve permitir ao aluno, iniciar os cursos.                                                                         |
| RF10   | Continuar Curso            | Alta                | O sistema deve permitir ao aluno, dar continuidade em um curso, que já tenha sido iniciado.                                  |
| RF11   | Visualizar Aula            | Alta                | O sistema deve permitir a visualização das informações de uma aula.                                                          |
| RF12   | Realizar Aula              | Alta                | O sistema deve permitir a realização de uma aula, desde o ensino até a execução dos minigames.                               |
| RF13   | Avançar Aula               | Alta                | O sistema deve permitir a rolagem da tela para baixo, avançando assim para a próxima aula.                                   |
| RF14   | Acessar <i>Leaderboard</i> | Média               | O sistema deve permitir a listagem da classificação dos alunos de uma turma, com suas respectivas pontuações e colocações.   |
| RF15   | Atualizar Dados Cadastrais | Média               | O sistema deve permitir a atualização de todos os dados mutáveis.                                                            |
| RF16   | Alterar Foto               | Sem                 | O sistema deve permitir a alteração da foto do usuário.                                                                      |

Fonte: Autor (2024).

Quadro 10 – Requisitos Não Funcionais

| Código | Título          | Nível de Prioridade | Descrição                                                                                 |
|--------|-----------------|---------------------|-------------------------------------------------------------------------------------------|
| RNF01  | Compatibilidade | Alta                | O sistema deverá ser compatível com Android e IOS.                                        |
| RNF02  | Comunicação     | Alta                | O sistema deverá integrar a comunicação entre a Aplicação, a API e o Banco de Dados.      |
| RNF03  | Responsividade  | Alta                | O sistema deverá se adaptar aos tamanhos de telas mais populares.                         |
| RNF04  | Segurança       | Alta                | O sistema deverá garantir autenticação segura, através do uso do <i>hashing</i> de senha. |

Fonte: Autor (2024).

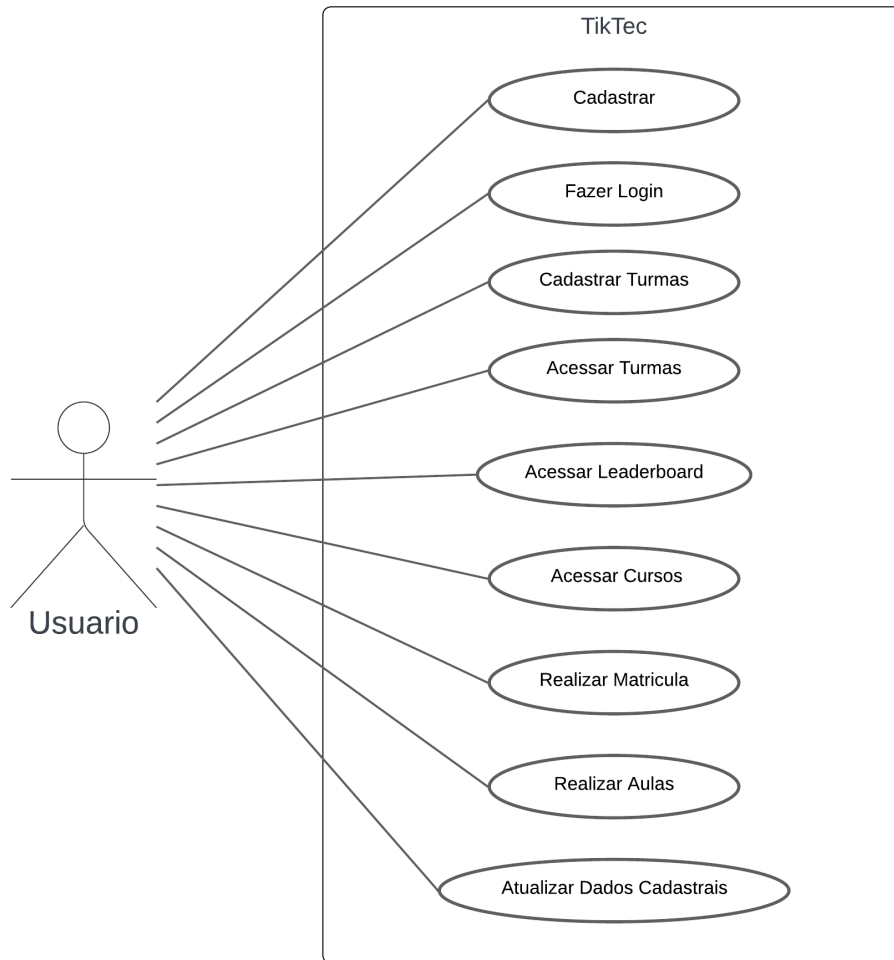
4.1.2 Diagramas

A elaboração de diagramas é fundamental no processo de desenvolvimento de um sistema, pois estes proporcionam uma representação visual clara e objetiva do contexto do problema. Adicionalmente, servem como um complemento essencial à documentação do projeto, oferecendo diversas perspectivas (visões), cada qual adequada a uma finalidade específica. De modo geral, pode-se dizer que eles facilitam a comunicação entre as equipes do projeto, reduzem a incidência de erros e auxiliam no planejamento.

4.1.2.1 Casos de Uso

Os casos de uso são uma forma de demonstrar as principais interações que um usuário pode realizar em um sistema, e também é uma forma visual de definir e descobrir os requisitos de um sistema. A ilustração está disponível na [Figura 9](#).

Figura 9 – Diagrama de casos de uso

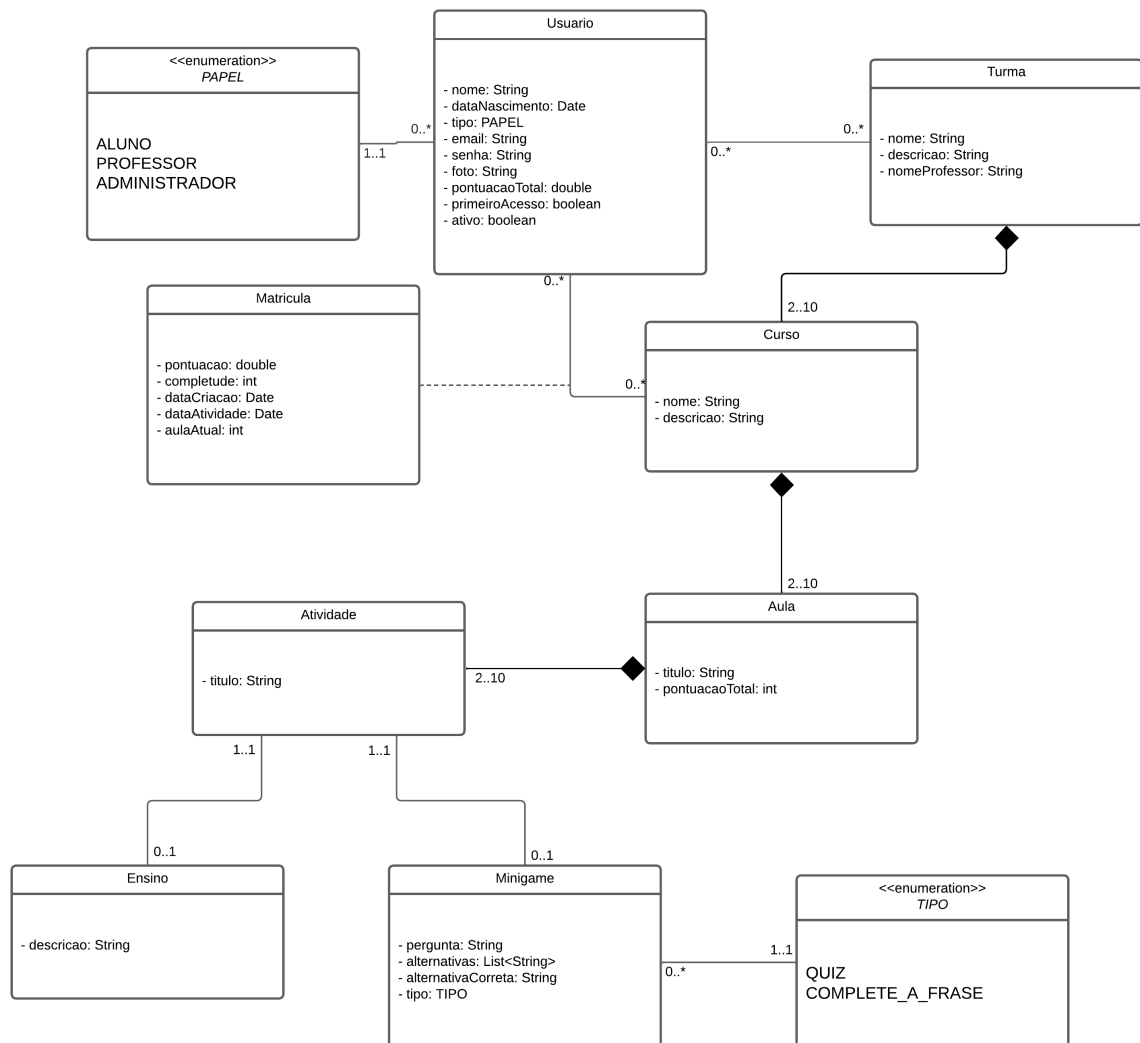


Fonte: Autor (2024).

#### 4.1.2.2 Classes

Os diagramas de classes, são usados principalmente em modelos de sistemas orientados a objetos, para que seja possível representa-las, e as suas respectivas associações. A ilustração está disponível na [Figura 10](#).

Figura 10 – Diagrama de classes

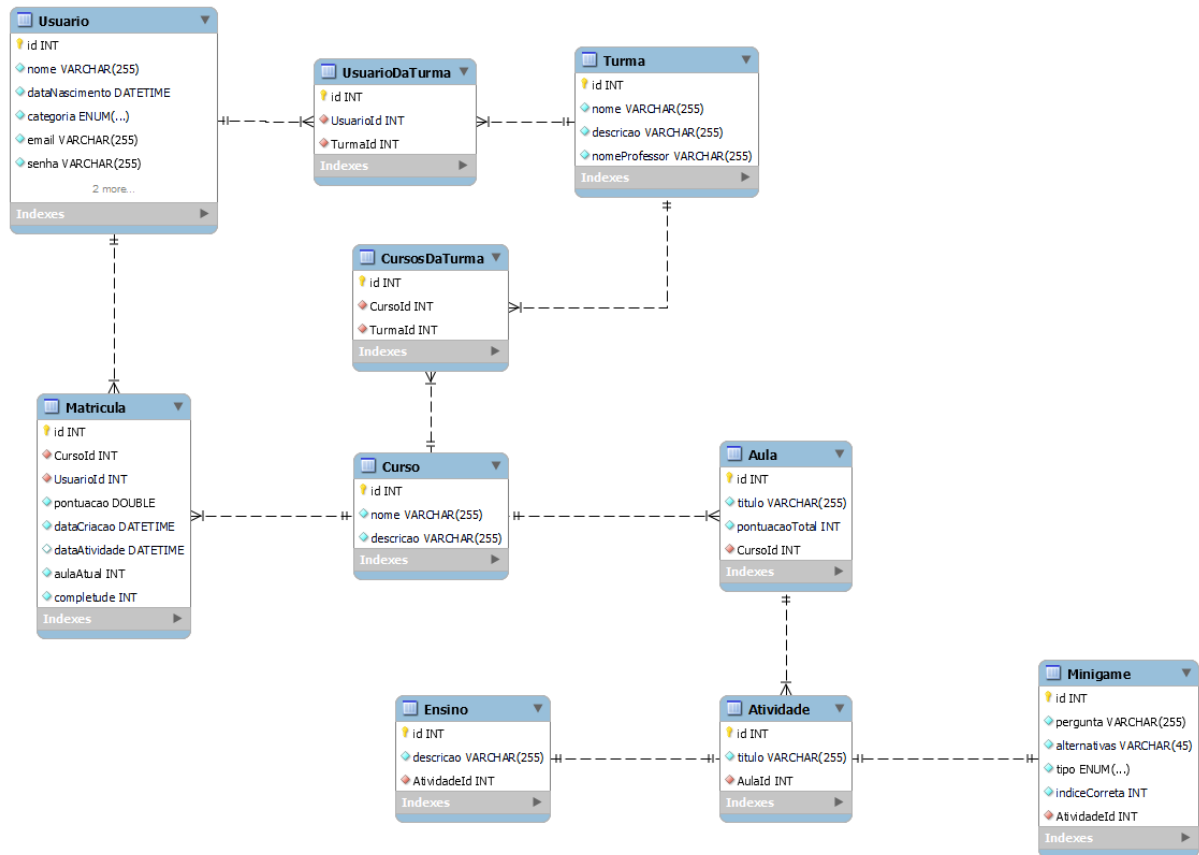


Fonte: Autor (2024).

#### 4.1.2.3 Entidade Relacionamento Estendido

Os diagramas de entidade e relacionamento estendido ou EER disponibilizam uma melhor visualização da construção da arquitetura do banco de dados, o que possibilita também a demonstração de possíveis requisitos mais complexos. A ilustração está disponível na [Figura 11](#).

Figura 11 – Diagrama de entidade relacionamento estendido



Fonte: Autor (2024).

## 4.2 Definição da Arquitetura

Nesta fase, foi realizada uma análise mais aprofundada das tecnologias anteriormente selecionadas, para que seja possível definir a arquitetura do sistema.

### 4.2.1 Tecnologias e Frameworks

Há uma variedade de ferramentas e tecnologias disponíveis para o desenvolvimento de aplicativos para dispositivos móveis. Algumas dessas tecnologias são empregadas no desenvolvimento de aplicativos nativos, enquanto outras são aplicadas no desenvolvimento de aplicativos híbridos.

O desenvolvimento nativo implica na criação de aplicativos para um sistema operacional específico. Isso possibilita um melhor aproveitamento dos recursos daquele sistema operacional, mas acaba por aumentar o tempo e o esforço no processo, visto que é necessário desenvolver uma aplicação para cada SO alvo. Por exemplo, para IOS, as linguagens Objective-C ou Swift são utilizadas; por outro lado, para Android, empregam-se as linguagens Java ou Kotlin.

No desenvolvimento híbrido, é possível criar um aplicativo que funciona em mais de

uma plataforma a partir de um único código-base. As principais tecnologias empregadas nessa abordagem são o Flutter e o React Native. Esses *frameworks* móveis multiplataforma permitem a criação de aplicativos compatíveis com diversos dispositivos, simplificando a portabilidade e reduzindo os custos de desenvolvimento.

Para o projeto em questão, foi escolhido o desenvolvimento híbrido, adotando o *framework* React Native como tecnologia para o desenvolvimento móvel multiplataforma. Essa decisão visa alcançar um público-alvo mais amplo, já que um único aplicativo pode ser direcionado para as plataformas IOS e Android, ampliando seu alcance. Além disso, buscando contemplar as tecnologias mais atuais e a possível portabilidade do sistema, optou-se por uma arquitetura que separa o desenvolvimento *back-end* (responsável por fornecer os serviços) do *front-end* (responsável pela interface do usuário).

Com base nessas definições, as tecnologias e *frameworks* selecionados para compor a solução são detalhados a seguir.

#### 4.2.1.1 React Native com TypeScript

O React Native permite o desenvolvimento de aplicações móveis para ambas as plataformas (IOS e Android) de forma simultânea, com a segurança e tipagem estática proporcionada pelo TypeScript, que é um superconjunto (superset) do JavaScript. Essa linguagem permite a verificação de tipos em tempo de compilação, o que mitiga o surgimento dos erros durante a execução. Além disso, sua adoção resulta em um desenvolvimento mais robusto, legível e de fácil manutenção. <sup>1 2</sup>

#### 4.2.1.2 Expo

O Expo foi escolhido por facilitar o desenvolvimento e os testes das aplicações, já que oferece uma grande variedade de ferramentas e bibliotecas prontas para uso. O Expo é um *framework open source* que gerencia o fluxo de trabalho e utiliza uma abordagem que abstrai as complexidades nativas de cada sistema operacional, focando o esforço no desenvolvimento do código, seja em JavaScript ou TypeScript. Além disso, permite a criação de *builds* tanto para Android quanto para IOS em seus próprios servidores, eliminando a necessidade de possuir um MacBook para depósitos na App Store. <sup>3</sup>

#### 4.2.1.3 API Rest

Construir uma API Rest permite que a mesma API possa ser utilizada tanto para a aplicação mobile quanto para uma futura aplicação web, promovendo reuso de código e padronização. Além de adicionar uma camada de segurança para as aplicações.

<sup>1</sup> React Native: <https://reactnative.dev/>

<sup>2</sup> TypeScript: <https://www.typescriptlang.org/>

<sup>3</sup> Expo: <https://expo.dev/>

#### 4.2.1.4 NestJs com TypeScript

Por ser um framework progressivo e versátil para a construção de APIs em Node.js, aproveitando ao máximo os recursos avançados do TypeScript, o NestJs promove uma arquitetura modular, facilitando a manutenção e escalabilidade do código, além de possuir suporte integrado para o framework PrismaIO. <sup>4</sup>

#### 4.2.1.5 PrismaIO

O PrismaIO é um framework que simplifica a interação com o banco de dados, proporcionando uma camada de ORM eficiente. <sup>5</sup>

#### 4.2.1.6 PostgreSQL

O PostgreSQL é um gerenciador de banco de dados relacionais altamente escalável e com vasta comunidade de suporte, oferecendo robustez, segurança e desempenho. <sup>6</sup>

### 4.2.2 Modelo Arquitetural

A arquitetura do sistema é um conceito que define a sua organização, os seus componentes e as suas respectivas interações. Na API será utilizado o *Layered Architecture* (Arquitetura em Camadas), pois organiza de forma a definir a responsabilidade individual de cada camada, promovendo o *Separation of Concerns* (separação de conceitos/preocupações). A ilustração está disponível na [Figura 12](#).

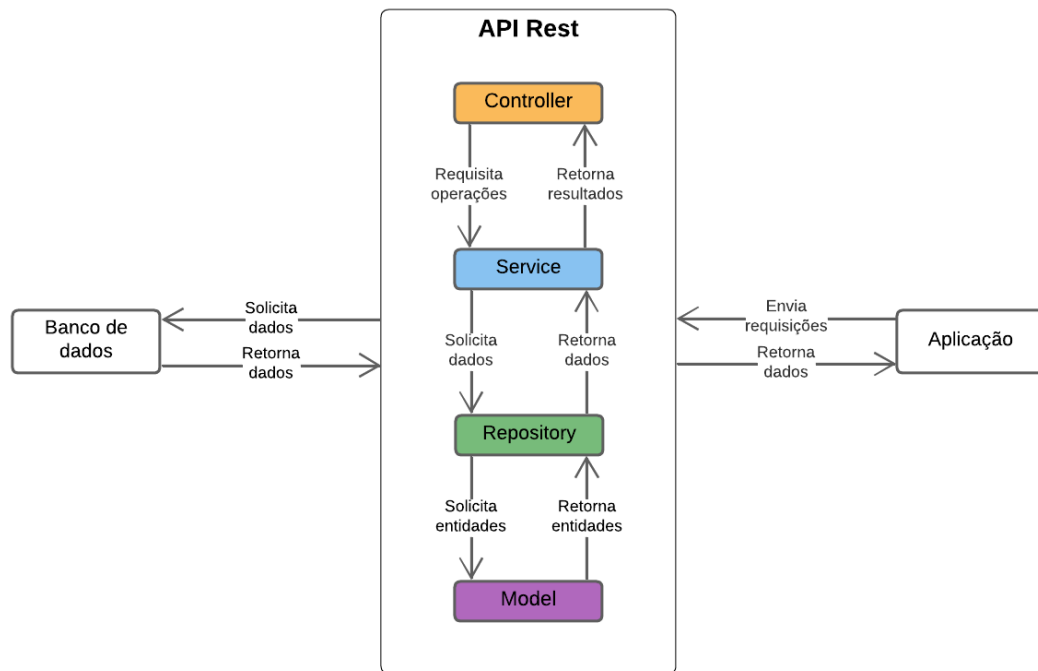
---

<sup>4</sup> NestJS: <https://nestjs.com/>

<sup>5</sup> PrismaIO: <https://www.prisma.io/>

<sup>6</sup> PostgreSQL: <https://www.postgresql.org/>

Figura 12 – Diagrama da arquitetura



Fonte: Autor (2024).

A camada *Controller* lida com as requisições e entradas e saídas do usuário, e fornece retorno o resultado desse processamento.

A camada *Service* lida com toda a regra de negócios do sistema.

A camada *Repository* lida com a comunicação com o banco de dados.

A camada *Model* lida com a representação das entidades e estruturação dos componentes do projeto.

## 4.3 Prototipagem das Telas

Nesta fase, foi realizada a prototipagem das telas, que serve como uma ferramenta essencial no processo de desenvolvimento. O seu propósito é funcionar como um alicerce para a orientação do que deve ser implementado e qual o resultado esperado, permitindo também o alinhamento entre a interpretação do desenvolvedor e a idealização dos *stakeholders*. Para esta modelagem, foi utilizado o software Figma, que é o editor gráfico de prototipagem mais popular na comunidade.

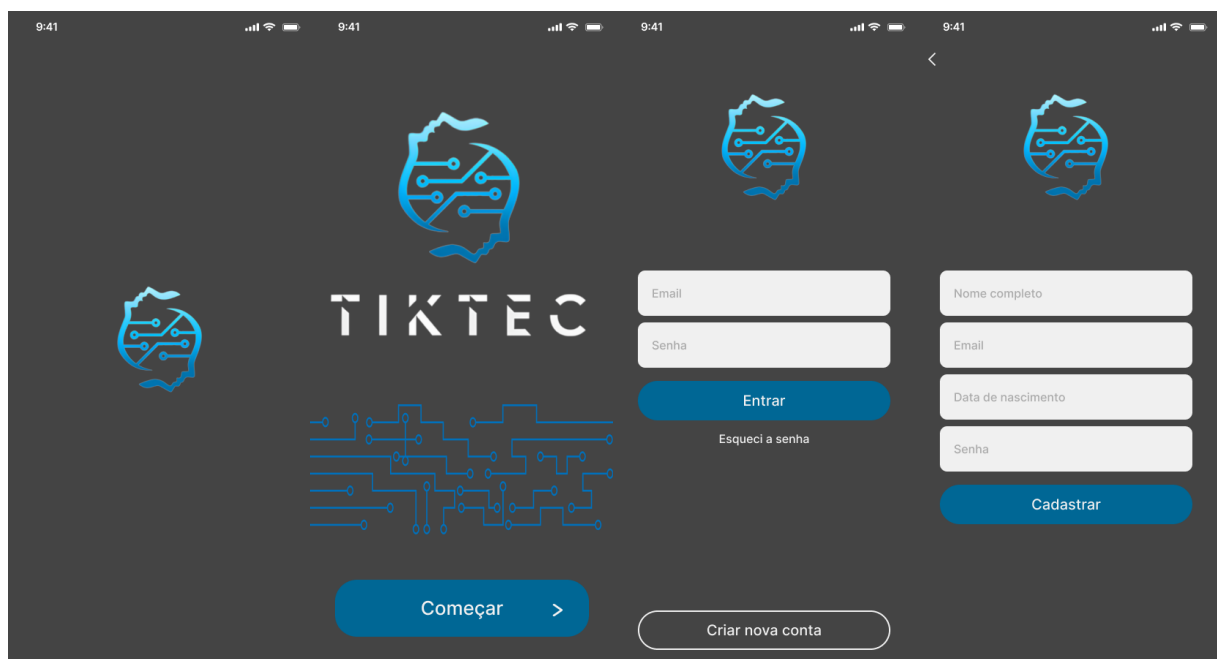
A seguir, são detalhadas as telas que compõem a aplicação. <sup>7</sup>

<sup>7</sup> Figma: <https://www.figma.com/pt-br/>

### 4.3.1 Telas Iniciais

Ao iniciar a execução da aplicação, o usuário visualizará a tela de *splash*, que é exibida até que o carregamento dos recursos iniciais seja finalizado. Em seguida, é disponibilizada a tela de acesso, indicando que a utilização do aplicativo está liberada. A partir dela, o usuário pode navegar para as telas de login, para realizar a autenticação, ou para a tela de cadastro, caso ainda não possua uma conta. As imagens das telas estão disponíveis na [Figura 13](#).

Figura 13 – Telas iniciais do aplicativo TikTec



Tela de *Splash*

Tela de Acesso

Tela de Login

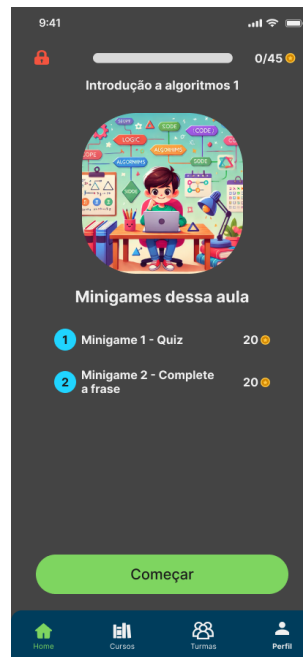
Tela de Cadastro

Fonte: Autor (2024).

### 4.3.2 Telas *Home*

Após a validação de suas credenciais, o usuário tem acesso à seção *home* da aplicação. Caso seja o seu primeiro acesso, será exibida uma mensagem para que se matricule em um curso ou turma. Para usuários já matriculados, a tela exibirá a aula mais recente do curso no qual a atividade do usuário seja a mais recente. Nesta tela, serão apresentadas as atividades com suas respectivas pontuações e um indicador de bloqueio, que informa a impossibilidade de avançar para a próxima aula até que a atual seja concluída. A imagem da tela está disponível na [Figura 14](#).

Figura 14 – Tela de Detalhamento da Aula



Fonte: Autor (2024).

Ao começar uma aula, o usuário terá acesso a atividades de ensino, com o objetivo de transmitir um conceito, e atividades de minigame (quiz ou complete a frase), que visam atestar o conhecimento adquirido. Se o usuário alcançar a pontuação determinada, poderá prosseguir para a próxima aula; caso contrário, poderá tentar novamente. Ao concluir com êxito, o cadeado é desbloqueado e a rolagem vertical é liberada para a próxima aula. As imagens das telas estão disponíveis na [Figura 15](#).

Figura 15 – Telas de execução da aula



Tela de Ensino

Tela de Minigame 1

Tela de Minigame 2

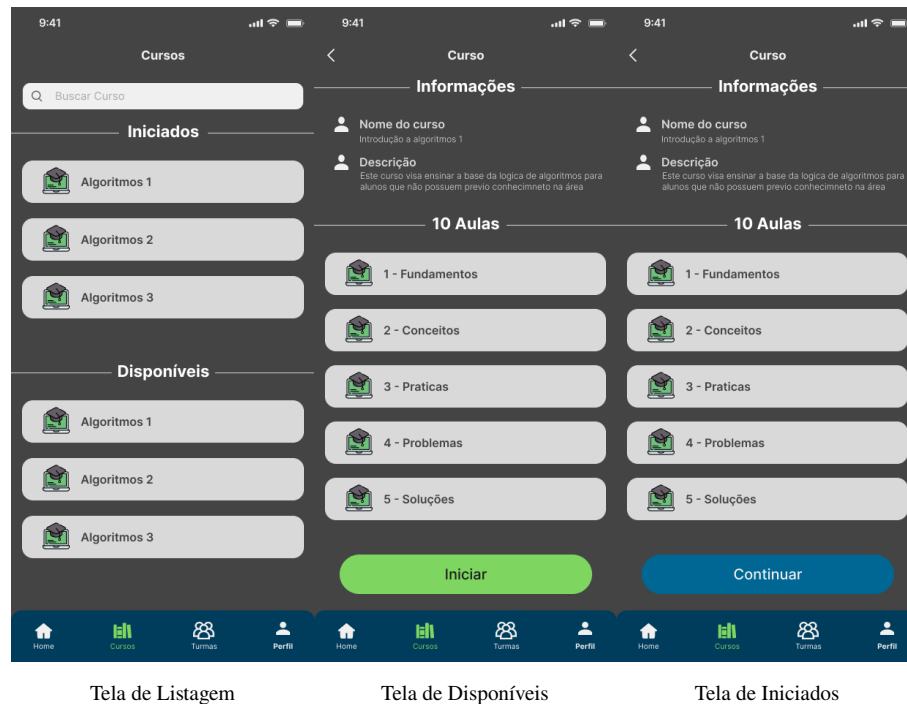
Tela de Parabenização

Fonte: Autor (2024).

### 4.3.3 Telas Cursos

A seção de cursos disponibiliza todos os cursos cadastrados, divididos entre os que o usuário iniciou e os que se encontram disponíveis. É possível realizar a busca pelo nome do curso ou acessá-lo diretamente. Ao fazer isso, são exibidas as informações descritivas do curso, suas aulas e a opção de iniciá-lo ou continuá-lo. As imagens das telas estão disponíveis na [Figura 16](#).

Figura 16 – Telas de cursos

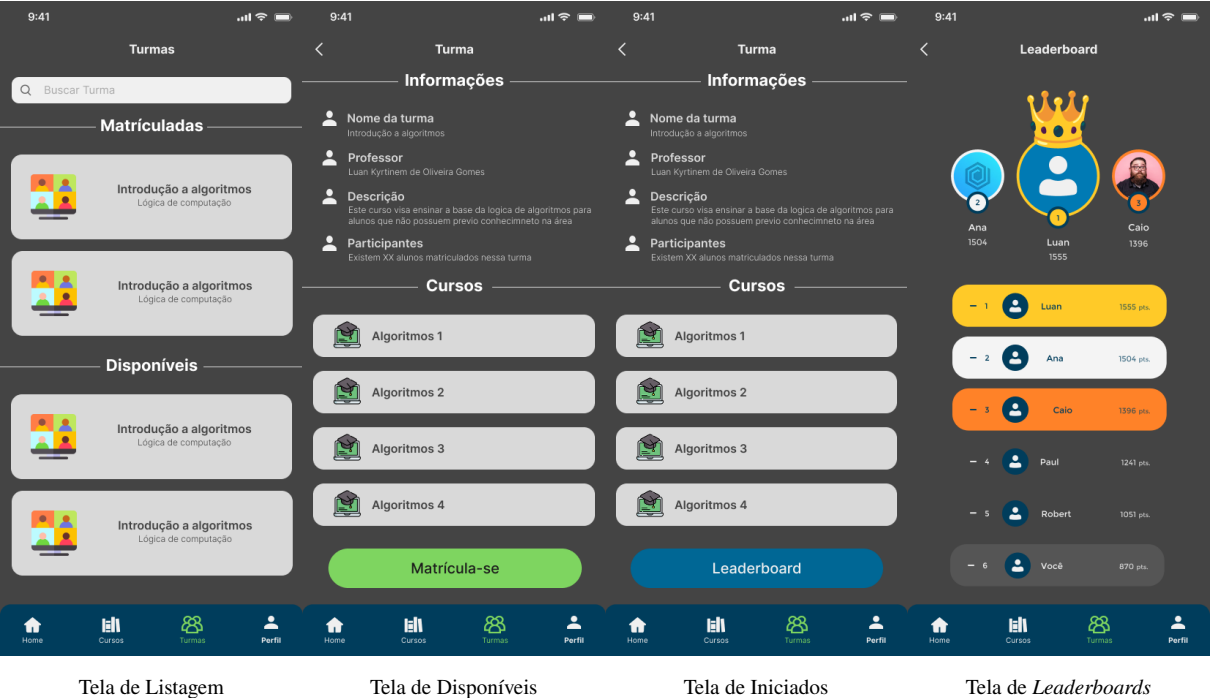


Fonte: Autor (2024).

#### 4.3.4 Telas Turmas

A seção de turmas apresenta todas as turmas cadastradas, divididas entre as quais o usuário se matriculou e as que estão disponíveis. É possível buscar pelo nome da turma ou acessá-la para visualizar suas informações descritivas e cursos respectivos. Ao se matricular em uma turma, o usuário inicia automaticamente todos os cursos que fazem parte dela. Para turmas em que já está matriculado, o usuário pode acessar o *leaderboard*. As imagens das telas estão disponíveis na [Figura 17](#).

Figura 17 – Telas de turmas

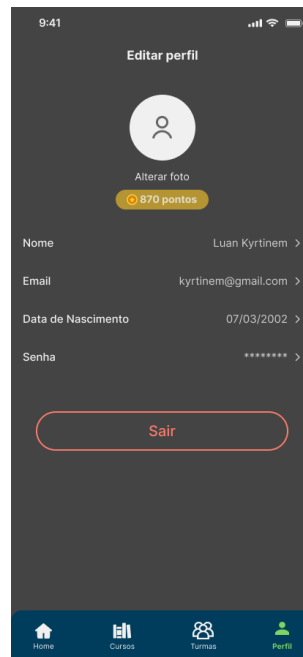


Fonte: Autor (2024).

### 4.3.5 Tela Perfil

A seção de perfil disponibiliza as informações específicas do usuário, que podem ser modificadas. A tela exibe também a sua pontuação total e o botão que finaliza a sessão de login. A imagem da tela está disponível na [Figura 18](#).

Figura 18 – Tela de Perfil

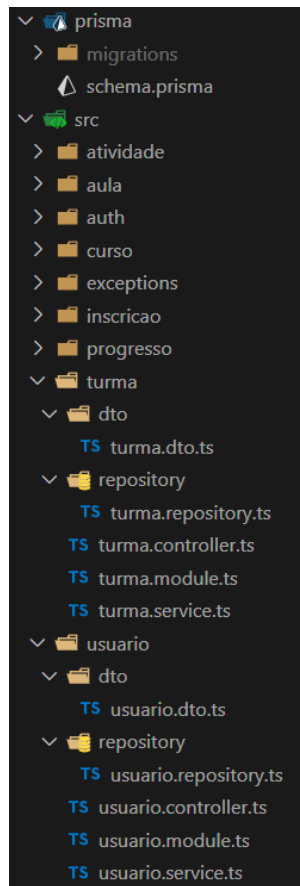


Fonte: Autor (2024).

## 4.4 Implementação da API

A API RESTful do projeto foi desenvolvida utilizando o framework NestJS com TypeScript, seguindo rigorosamente o modelo de arquitetura em camadas apresentado na [subseção 4.2.2](#). Essa abordagem favorece a separação de responsabilidades, a manutenção e a escalabilidade do sistema, permitindo que cada módulo possua um escopo bem definido, como por exemplo, os módulos turma e usuário. A [Figura 19](#) apresenta a estrutura de pastas do projeto, evidenciando a divisão modular adotada.

Figura 19 – Estrutura de pastas da API



Fonte: Autor (2025).

#### 4.4.1 Camada de Controle (Controller)

A camada de controle é a porta de entrada da aplicação, responsável por receber e direcionar as requisições HTTP. Na API, os controladores (\*.controller.ts) utilizam decorators do NestJS, como @Controller, @Get e @Post, para mapear as rotas e extrair dados das requisições.

Além de realizar validações superficiais por meio de DTOs (Data Transfer Objects), como exemplificado na [Figura 20](#), que apresenta um trecho do arquivo turma.dto.ts responsável por definir a estrutura dos dados recebidos, essa camada delega a execução da lógica de negócio para a respectiva camada de serviço. Essa divisão garante que a interface de comunicação com o cliente permaneça simples e organizada, enquanto a lógica mais complexa é tratada separadamente. Já a [Figura 21](#) apresenta um exemplo de implementação do arquivo turma.controller.ts, destacando as rotas utilizadas para a criação e listagem de turmas.

Figura 20 – Trecho do código turma.dto.ts

```
1 export class CreateTurmaDTO {
2   @IsString({ message: 'O nome da turma deve ser um texto.' })
3   @NotEmpty({ message: 'O nome da turma não pode estar vazio.' })
4   nome: string;
5
6   @IsString({ message: 'A descrição da turma deve ser um texto.' })
7   @NotEmpty({ message: 'A descrição da turma não pode estar vazia.' })
8   descricao: string;
9
10  @IsString({ message: 'O nome do professor deve ser um texto.' })
11  @NotEmpty({ message: 'O nome do professor não pode estar vazio.' })
12  nomeProfessor: string;
13
14  @IsArray({ message: 'A lista de cursos deve ser um array.' })
15  @Each({ each: true, message: 'Cada item na lista de cursos deve ser um ID (número inteiro).' })
16  @ArrayMinSize(2, { message: 'Uma turma deve ser associada a no mínimo 2 cursos.' })
17  @ArrayMaxSize(10, { message: 'Uma turma pode ser associada a no máximo 10 cursos.' })
18  cursosIds: number[];
19 }
```

Fonte: Autor (2025).

Figura 21 – Trecho do código turma.controller.ts

```
1 @Controller('turma')
2 @UseGuards(JwtAuthGuard)
3 export class TurmaController {
4   constructor(private readonly turmaService: TurmaService) { }
5
6   @Post('cadastrar')
7   async create(@Req() req: any, @Body() dto: CreateTurmaDTO) {
8     return await this.turmaService.create(dto, req.user as ValidarUsuarioDTO);
9   }
10
11  @Get()
12  async findAll(@Req() req: any) {
13    const usuarioLogado = req.user as ValidarUsuarioDTO;
14    return this.turmaService.findAll(usuarioLogado.id);
15  }
16 }
```

Fonte: Autor (2025).

### 4.4.2 Camada de Serviço (Service)

A camada de serviço (\*.service.ts) contém a lógica central de negócio. É nela que se concentram as regras, cálculos e validações necessárias antes da persistência ou recuperação de dados. Essa camada é injetada nos controladores, promovendo baixo acoplamento e alta coesão.

No contexto do módulo turma, por exemplo, o serviço é responsável por aplicar regras específicas antes de acionar a camada de repositório, como verificação de duplicidade de dados ou controle de acesso. Entre as linhas 9 e 24, observa-se o método create, responsável por validar a existência do usuário, checar se ele possui permissão adequada (administrador ou professor) e confirmar se os cursos informados são válidos antes de delegar a criação da turma ao repositório. Já entre as linhas 26 e 48, o método findAll organiza a recuperação das turmas vinculadas a um usuário, avaliando se ele está matriculado e se concluiu todos os cursos da turma para determinar seu status de finalização. A [Figura 22](#) ilustra um trecho do arquivo turma.service.ts.

Figura 22 – Trecho do código turma.service.ts

```
1  @Injectable()
2  export class TurmaService {
3    constructor(
4      private readonly turmaRepository: TurmaRepository,
5      private readonly usuarioRepository: UsuarioRepository,
6      private readonly cursoRepository: CursoRepository,
7    ) {}
8
9    async create(dto: CreateTurmaDTO, usuarioValido: ValidarUsuarioDTO) {
10      const usuario = await this.usuarioRepository.findUserByEmail(usuarioValido.email);
11      if (!usuario) throw new NotFoundException('Usuário', usuarioValido.email);
12
13      const isAllowed = usuario.papel === 'ADMINISTRADOR' || usuario.papel === 'PROFESSOR';
14      if (!isAllowed) {
15        throw new ForbiddenException('Você não tem permissão para criar turmas.');
```

Fonte: Autor (2025).

### 4.4.3 Camada de Repositório (Repository)

Para abstrair e centralizar a comunicação com o banco de dados, foi adotado o padrão Repository. Essa camada (\*.repository.ts) é a única responsável por interagir diretamente com o ORM Prisma, utilizando métodos como `findMany`, `create` e `update`.

Essa abordagem isola a lógica de persistência e permite que alterações no mecanismo de acesso aos dados sejam realizadas sem impactar as camadas superiores. No trecho de código, por exemplo, o método `create` entre as linhas 5 a 31 utiliza transações para registrar uma nova turma e associar os cursos correspondentes, garantindo consistência no processo. Já o método `findAllComStatusMatricula` entre as linhas 33 a 63 retorna as turmas de um usuário, trazendo também informações sobre matrículas e progresso nos cursos relacionados. A [Figura 23](#) apresenta um exemplo de implementação do arquivo `turma.repository.ts`.

Figura 23 – Trecho do código turma.repository.ts

```
1  @Injectable()
2  export class TurmaRepository {
3    private prisma = new PrismaClient();
4
5    async create(dto: CreateTurmaDTO) {
6      try {
7        return await this.prisma.$transaction(async (tx) => {
8          const novaTurma = await tx.turma.create({
9            data: {
10              nome: dto.nome,
11              descricao: dto.descricao,
12              nomeProfessor: dto.nomeProfessor,
13            },
14          });
15
16          const cursoTurmaData = dto.cursosIds.map((cursoId) => ({
17            TurmaId: novaTurma.id,
18            CursoId: cursoId,
19          }));
20
21          await tx.cursoTurma.createMany({
22            data: cursoTurmaData,
23          });
24
25          return novaTurma;
26        });
27      } catch (error) {
28        console.error(error);
29        throw new InternalServerErrorException('Ocorreu um erro ao criar a turma.');
```

Fonte: Autor (2025).

#### 4.4.4 Banco de Dados (BD)

A persistência de dados é realizada por meio do SGBD PostgreSQL, com a interação mediada pelo ORM PrismaIO. A definição da estrutura do banco encontra-se centralizada no arquivo `schema.prisma`, que atua como a fonte que estrutura todos os modelos de dados da aplicação.

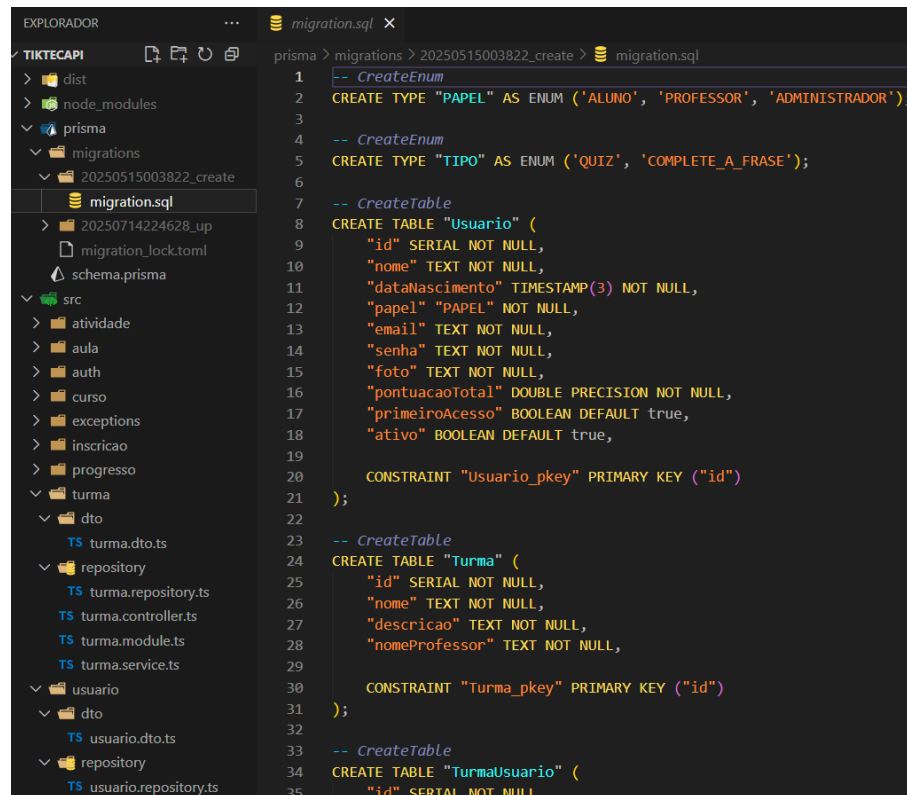
Para garantir um versionamento controlado e rastreável da estrutura do banco, foi empregada a ferramenta Prisma Migrate, que gera automaticamente scripts de migração a partir das alterações no `schema.prisma`. A [Figura 24](#) apresenta um exemplo de definição de modelo no `schema.prisma`, enquanto a [Figura 25](#) exibe a estrutura da pasta de migrações do projeto.

Figura 24 – Trecho do arquivo `schema.prisma`

```
1  enum PAPEL {
2    ALUNO
3    PROFESSOR
4    ADMINISTRADOR
5  }
6
7  model Usuario {
8    id          Int           @id @default(autoincrement())
9    nome        String
10   dataNascimento DateTime
11   papel        PAPEL
12   email        String       @unique
13   senha        String
14   foto         String       @default("")
15   pontuacaoTotal Float       @default(0)
16   primeiroAcesso Boolean?    @default(true)
17   ativo        Boolean?      @default(true)
18   TurmaUsuario TurmaUsuario[]
19   Matricula    Matricula[]
20 }
21
22 model Turma {
23   id          Int           @id @default(autoincrement())
24   nome        String
25   descricao   String
26   nomeProfessor String
27   TurmaUsuario TurmaUsuario[]
28   CursoTurma  CursoTurma[]
29 }
30
31 model TurmaUsuario {
32   id          Int           @id @default(autoincrement())
33   TurmaId     Int
34   UsuarioId   Int
35   turma       Turma        @relation(fields: [TurmaId], references: [id])
36   usuario     Usuario       @relation(fields: [UsuarioId], references: [id])
37 }
```

Fonte: Autor (2025).

Figura 25 – Estrutura da pasta de migrações do Prisma

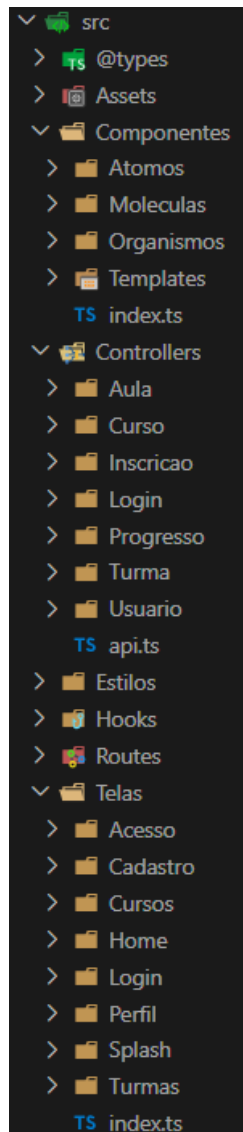


Fonte: Autor (2025).

## 4.5 Implementação do Aplicativo

A aplicação móvel foi desenvolvida com React Native e TypeScript, seguindo uma arquitetura voltada à reutilização de componentes e organização modular. Essa abordagem favorece a manutenção, a escalabilidade e a consistência visual entre diferentes telas da aplicação. A [Figura 26](#) apresenta a estrutura de pastas do projeto, evidenciando a separação entre componentes, telas e serviços.

Figura 26 – Estrutura de pastas do Projeto



Fonte: Autor (2025).

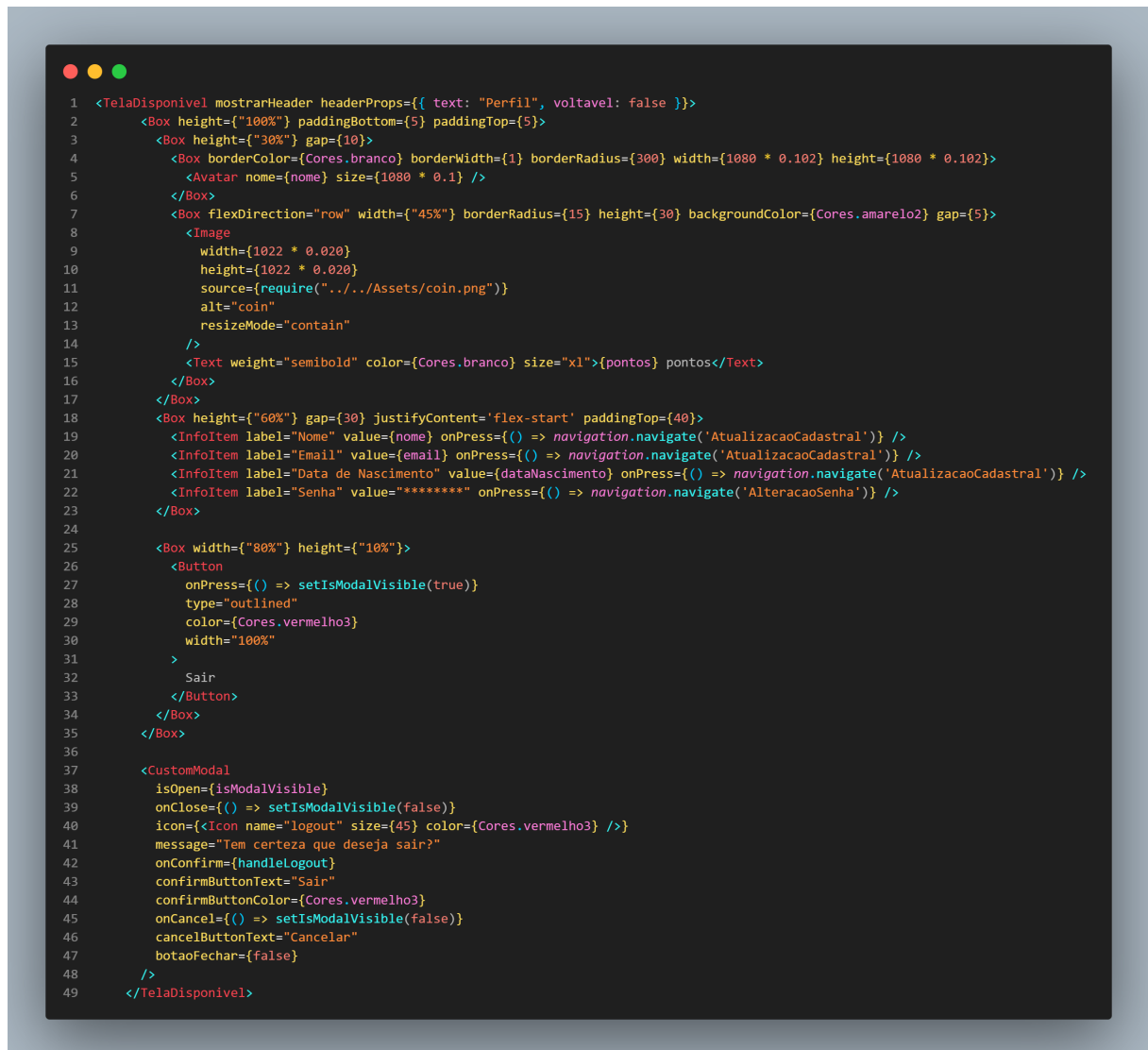
### 4.5.1 Arquitetura de Componentes

A construção da interface do usuário seguiu uma adaptação da metodologia Atomic Design, categorizando os componentes em níveis de complexidade: átomos, moléculas, organismos e templates.

Essa organização possibilita a criação de interfaces consistentes e facilita a evolução do projeto, permitindo que alterações em um componente de nível básico se propaguem automaticamente para as interfaces mais complexas que o utilizam. No exemplo, trata-se da tela de Perfil, onde entre as linhas 2 e 17 observa-se a estruturação da parte superior com avatar e pontuação do usuário. Entre as linhas 18 e 23, são exibidos dados cadastrais organizados em componentes reutilizáveis como InfoItem. Já as linhas 37 a 48 demonstram a utilização de um componente de modal personalizado (CustomModal), que permite confirmar ou cancelar a ação

de logout. A [Figura 27](#) ilustra um exemplo de tela composta por diferentes níveis de componentes.

Figura 27 – Exemplo de composição de tela a partir de componentes reutilizáveis



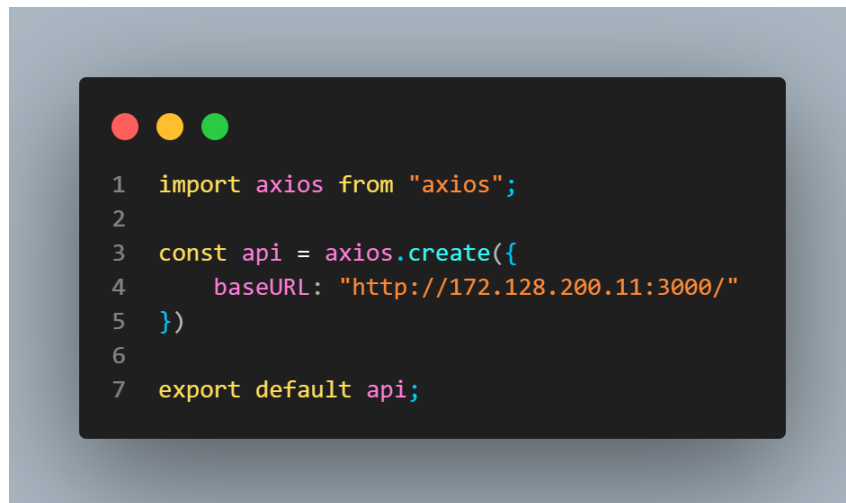
Fonte: Autor (2025).

## 4.5.2 Integração com a API

A comunicação entre a aplicação e a API é realizada por meio de requisições HTTP, utilizando a biblioteca Axios. Para centralizar e padronizar as configurações, foi criada uma instância dedicada (api.ts) que define parâmetros globais, como a URL base.

As chamadas para cada endpoint são encapsuladas em funções organizadas na pasta Controllers, atuando como uma camada de serviço no front-end e abstraindo a complexidade da comunicação com a API. A [Figura 28](#) apresenta o código de configuração do Axios, enquanto a [Figura 29](#) demonstra um exemplo de função responsável por consumir um endpoint.

Figura 28 – Configuração do Axios no arquivo api.ts

A screenshot of a code editor with a dark background and light-colored text. At the top left, there are three colored circles (red, yellow, green) representing window control buttons. The code is as follows:

```
1 import axios from "axios";
2
3 const api = axios.create({
4   baseUrl: "http://172.128.200.11:3000/"
5 })
6
7 export default api;
```

Fonte: Autor (2025).

Figura 29 – Exemplo de funções que consomem endpoints no aplicativo

A screenshot of a code editor with a dark background and light-colored text. At the top left, there are three colored circles (red, yellow, green) representing window control buttons. The code is as follows:

```
1 export async function CriarTurma(dto: CreateTurmaDTO, token: string) {
2   try {
3     const resultado = await api.post("/turma/cadastrar", dto, {
4       headers: { Authorization: `Bearer ${token}` }
5     });
6     return resultado.data;
7   } catch (error) {
8     console.log(error);
9     return null;
10  }
11 }
12
13 export async function ListarTurmas(token: string) {
14   try {
15     const resultado = await api.get("/turma", {
16       headers: { Authorization: `Bearer ${token}` }
17     });
18     return resultado.data as Turma[];
19   } catch (error) {
20     console.log(error);
21     return null;
22   }
23 }
```

Fonte: Autor (2025).

# 5

## Conclusões e Trabalhos Futuros

O desenvolvimento deste trabalho possibilitou a criação de uma plataforma móvel adaptável que utiliza a gamificação e os minigames como estratégia para apoiar o ensino em diversas áreas do conhecimento. A implementação do aplicativo resultou em um protótipo funcional, demonstrando a viabilidade técnica da proposta. Embora a validação prática não tenha sido realizada, o design da plataforma, embasado na literatura, foi estruturado para explorar o potencial de metodologias lúdicas no engajamento de estudantes em processos de aprendizagem tradicionalmente considerados desafiadores.

Entretanto, algumas etapas previstas inicialmente não puderam ser realizadas dentro do período de execução desta pesquisa. Em especial, destaca-se a validação prática do aplicativo com usuários de diferentes áreas de formação e faixas etárias, utilizando um curso-piloto, focado em lógica de algoritmos, como instrumento avaliativo inicial. Essa etapa é fundamental para aferir a usabilidade, o engajamento e o real impacto pedagógico da aplicação.

Como desdobramento natural, os trabalhos futuros incluem a condução de testes com grupos diversificados de participantes, de modo a verificar se a plataforma consegue atender satisfatoriamente a perfis heterogêneos de aprendizes. Além disso, sugere-se a realização de estudos comparativos entre a abordagem gamificada e metodologias tradicionais de ensino, a fim de mensurar a efetividade da solução proposta no desenvolvimento de habilidades e competências específicas de cada área.

Outras linhas de continuidade envolvem a criação de novos cursos em áreas distintas, a inclusão de recursos de acessibilidade que favoreçam um público ainda mais amplo e a integração de métricas de desempenho que permitam acompanhar a evolução dos usuários ao longo do curso. Soma-se a isso a possibilidade de implementação dos requisitos de gerenciamento listados no [Apêndice A](#), que apresentam funcionalidades planejadas para versões futuras e que podem tornar a plataforma mais robusta e completa.

Dessa forma, este trabalho se encerra como um primeiro passo rumo a uma proposta de

ensino mais dinâmica e interativa, mas abre espaço para pesquisas futuras que possam consolidar, validar e ampliar sua contribuição no cenário educacional.

## Referências

ALMEIDA, H. W. de; SILVA, J. C. da; LINS, H. A. de M. Criação de jogo matemático digital com crianças e jovens surdos: Contribuições da pedagogia visual. *Revista Cocar*, v. 13, n. 27, p. 399–422, 2019. Disponível em: <<https://periodicos.uepa.br/index.php/cocar/article/view/2845>>. Citado na página 31.

BATISTA, André Luiz França; FILHO, Aurélio Pajuaba Nehme; PIMENTEL, Daniel Ramos; MARTINS, Rodrigo Grassi. Sql planet: Jogo online para ensino de linguagem sql. In: SOCIEDADE BRASILEIRA DE COMPUTAÇÃO. *SBC – Proceedings of SBGames 2019 Education Track – Short Papers*. Rio de Janeiro, Brazil: Instituto Federal do Triângulo Mineiro, Departamento de Ciência da Computação, 2019. Disponível em: <<https://www.sbgames.org/sbgames2019/files/papers/EducacaoShort/197137.pdf>>. Citado na página 30.

CAMARGO, Fausto; DAROS, Thuinie. *A Sala de Aula Inovadora: estratégias pedagógicas para fomentar o aprendizado ativo*. 1. ed. Local de publicação: Penso e Uniamerica, 2018. 146 p. Citado na página 18.

CAMPOS, André; GARDIMAN, Renato; MADEIRA, Charles. Uma ferramenta gamificada de apoio à disciplina introdutória de programação. In: *WORKSHOP SOBRE EDUCAÇÃO EM COMPUTAÇÃO (WEI)*, 23. Recife: Sociedade Brasileira de Computação, 2015. p. 356–365. ISSN 2595-6175. Disponível em: <<https://sol.sbc.org.br/index.php/wei/article/view/10252>>. Citado na página 17.

COSTA, Stephany da Silva. *Quero Doar: um aplicativo móvel para conectar pessoas a instituições sociais que necessitam de doações ou trabalho voluntário*. Monografia de graduação, São Cristóvão, SE, 2023. Disponível em: <<https://ri.ufs.br/jspui/handle/riufs/17701>>. Nenhuma citação no texto.

FARDO, M. L. A gamificação aplicada em ambientes de aprendizagem. *Revista Novas Tecnologias na Educação*, v. 11, n. 1, 2013. Disponível em: <<https://seer.ufrgs.br/index.php/rnente/article/view/41629>>. Citado 2 vezes nas páginas 11 e 12.

FRIEDRICH, Ronaldo Vaz; SANTOS, Daniela Scherer dos; KELLER, Rodrigo dos Santos; PUNTEL, Marcio Daniel; BIASOLI, Daniel. Proposta metodologica para a inserção ao ensino de lógica de programação com logo e lego mindstorms. In: *XXIII Simpósio Brasileiro de Informática na Educação (SBIE)*. [s.n.], 2012. p. 10. Disponível em: <<http://milanesa.ime.usp.br/rbie/index.php/sbie/article/view/1762/1523>>. Citado na página 16.

GASPARETTO, Glau; MOIÓLI, Julia. *O que vídeos curtos e viciantes fazem no cérebro do seu filho*. 2023. <<https://www.uol.com.br/tilt/noticias/redacao/2023/08/21/videos-curtos-atencao-menor-ainda-o-cerebro-sob-os-efeitos-do-tiktok.htm>>. Citado na página 11.

GOMES, Anabela; MENDES, Antonio. Learning to program - difficulties and solutions. In: . [S.l.: s.n.], 2007. p. 283–287. Citado na página 15.

GONÇALVES, Inácia de Jesus Henriques; AZEVEDO, Breno Fabrício Terra; OLIVEIRA, Sergio Rafael Cortes de. Gamificação em um ambiente virtual de aprendizagem: relatos da aplicação de uma sequência didática gamificada sobre o conteúdo de funções em um curso superior. *Revista Novas Tecnologias na Educação*, v. 20, n. 1, p. 163–172, 2022. Disponível em: <<https://seer.ufrgs.br/index.php/renote/article/view/126631>>. Citado 2 vezes nas páginas 11 e 12.

JÚNIOR, Rogério Paulo Marcon; BONIATI, Bruno Batista. Logicblocks: Uma ferramenta para o ensino de lógica de programação. In: *Anais do Encontro Anual de Tecnologia da Informação (EATI)*. Frederico Westphalen, RS: EATI, 2015. p. 63–70. Disponível em: <<http://anais.eati.info:8080/index.php/2019/article/view/151>>. Citado na página 16.

KRANZ, André Carlos; JUNIOR, Eduardo Stumpf Lima; LIMA, Fabiana Alves de; TRISTÃO, Patrícia da Silva; DULLIUS, Vera Fátima. O minigame como objeto de aprendizagem para a metodologia da pesquisa. In: *SBC – Proceedings of SBGames*. [s.n.], 2017. p. 1024–1027. Disponível em: <<https://www.sbgames.org/sbgames2017/papers/CulturaShort/175203.pdf>>. Citado na página 17.

KRASSMANN, Aliane; MEDINA, Roseclea. Jogo sério ubíquo integrado a mundo virtual opensim para o ensino de redes de computadores (jaspion). In: *Anais do XV Simpósio Brasileiro de Jogos e Entretenimento Digital (SBGames)*. [s.n.], 2016. p. 50. Disponível em: <[https://repositorio.ufsm.br/bitstream/handle/1/17702/DIS\\_PPGCC\\_2016\\_KRASSMANN\\_ALIANE.pdf?sequence=1&isAllowed=y](https://repositorio.ufsm.br/bitstream/handle/1/17702/DIS_PPGCC_2016_KRASSMANN_ALIANE.pdf?sequence=1&isAllowed=y)>. Citado na página 30.

MACHADO, Saul Brito de Assis. A influência do tiktok no desenvolvimento psicossocial do adolescente. 2023. Disponível em: <<https://sis.unileao.edu.br/uploads/3/PSICOLOGIA/P1641.pdf>>. Citado na página 11.

MATIAS, Lucas David Maia; CELESTINO, Maria Nielly Santos; CARVALHO, Mariana Albernaz Pinheiro de; GOUVEIA, Bernadete de Lourdes André; GINÚ, Isabel Luiza do Nascimento; COSTA, Marta Miriam Lopes; ANDRADE, Lidiane Lima de. *SEMIO EM JOGO®*: Development and evaluation of a playful-educational technology for nursing education. *Texto & Contexto - Enfermagem*, Universidade Federal de Santa Catarina, Programa de Pós-Graduação em Enfermagem, v. 32, p. e20220324, 2023. ISSN 0104-0707. Disponível em: <<https://doi.org/10.1590/1980-265X-TCE-2022-0324en>>. Citado na página 30.

NATAL, Eduarda Contri; BARBOSA, Anghinoni; HERNANDES, Cardoso; MUCH, De Sousa; BIGOLIN, M.; SILVA, José Ribeiro da; SILVA, Balestro; CARVALHO, Filipe Batista de. Tri-logic: Um ambiente gamificado como ferramenta de auxílio ao ensino de aprendizagem de lógica de programação. *Revista Novas Tecnologias na Educação*, Porto Alegre, v. 16, n. 2, p. 41–50, 2018. Disponível em: <<https://seer.ufrgs.br/index.php/renote/article/view/89298>>. Citado 3 vezes nas páginas 15, 16 e 17.

NEVES, Marcos Antonio Soares das; MIYAHARA, Ricardo Yoshimitsu; SANTOS, Danilo Sande. Gamificação em plataformas educacionais: potencializando o ensino de física através do nivelamento online. *Revista BOEM*, v. 11, p. e0131, 2023. Disponível em: <<https://revistas.udesc.br/index.php/boem/article/view/24786>>. Citado na página 31.

OLIVEIRA, C. Alves de; FERREIRA, W. C. O jogo digital quiz pg nas aulas de matemática: Possibilidades para o ensino e aprendizagem de progressão geométrica. *Revista de Educação Matemática*, v. 18, p. e021015, 2021. Disponível em: <<https://www.revistasbemsp.com.br/index.php/REMat-SP/article/view/121>>. Citado na página 31.

OLIVEIRA, J. A. de. Reflexão sobre o uso das tecnologias nas escolas. *Revista Científica FESA*, v. 1, n. 19, p. 93–106, 2022. Disponível em: <<https://revistafesa.com/index.php/fesa/article/view/208>>. Citado na página 12.

PIRES, Glice; AO, Jeanne da Silva Barbosa Bulc AZEVEDO, Debora; MADEIRA, Charles. Gamificação no ensino de ciências: um relato de experiência. In: *Anais do Workshop de Informática na Escola (WIE)*, 25. Brasília: Sociedade Brasileira de Computação, 2019. p. 707–714. Disponível em: <<https://doi.org/10.5753/cbie.wie.2019.707>>. Citado 2 vezes nas páginas 11 e 12.

RAFALSKI, Jadson do Prado; SILVA, Maria Aparecida de Faria da; OLIVEIRA, Kenia Luíza Rabelo de; OLIVEIRA, Márcia Gonçalves de. Laboratório de pensamento computacional na perspectiva da educação científica para formação de professores. In: *Anais do Laboratório de Ideias - Simpósio Brasileiro de Educação em Computação (EDUCOMP)*, 2. Online: Sociedade Brasileira de Computação, 2022. p. 21–22. Disponível em: <[https://doi.org/10.5753/educomp\\_estendido.2022.19403](https://doi.org/10.5753/educomp_estendido.2022.19403)>. Citado na página 12.

ROCHA, K. N.; ALMEIDA, N. M.; SOARES, C. R. G.; SILVA, L. F. M. S. Q-LIBRAS: Um jogo educacional para estimular alunos surdos à aprendizagem de química. *Revista Educação Especial*, v. 32, p. e114/1–14, 2019. Disponível em: <<https://periodicos.ufsm.br/educacaoespecial/article/view/32977>>. Citado na página 31.

ROSA, Caroline Dalla; PRETTO, Vinicius; BULEGON, Ana Maria. Tecnologias móveis e o estudo da tabuada: Reflexões sobre o uso do jogo math duel. *Tear: Revista de Educação, Ciência e Tecnologia*, v. 8, n. 2, 2019. Disponível em: <<https://periodicos.ifrs.edu.br/index.php/tear/article/view/3467>>. Citado na página 31.

SANDE, D.; SANDE, D.; CARVALHO, A. A. Aprendizagem de física e engajamento através do jogo nivelamento online durante a pandemia da covid-19. *Revista Novas Tecnologias na Educação*, v. 19, n. 2, p. 61–70, 2021. Disponível em: <<https://seer.ufrgs.br/index.php/renote/article/view/121187>>. Citado na página 30.

SANTOS, Yuri D. C. de Faria; GARCIA, Daiane C.; SANTOS, Paulo C. dos; SAQUI, Diego. Codify: Uma plataforma gamificada para auxiliar e motivar estudantes no ensino de programação. In: *TRILHA DE EDUCAÇÃO – ARTIGOS CURTOS - SIMPÓSIO BRASILEIRO DE JOGOS E ENTRETENIMENTO DIGITAL (SBGAMES)*, 21. Natal/RN: Sociedade Brasileira de Computação, 2022. p. 1126–1130. Disponível em: <[https://sol.sbc.org.br/index.php/sbgames\\_estendido/article/view/23755](https://sol.sbc.org.br/index.php/sbgames_estendido/article/view/23755)>. Citado na página 17.

SCHIMIGUEL, Juliano; JR, Hélio Rosetti; AMARAL, Gustavo Perini do; MARTINS, Carlos Adriano; JR, Carlos Fernando Araújo. O jogo na educação matemática: Desenvolvimento de um rpg para trabalhar o conceito de moeda no ensino fundamental. *Holos*, v. 8, p. 113–121, 2016. Disponível em: <<https://www2.ifrn.edu.br/ojs/index.php/HOLOS/article/view/3607>>. Citado na página 31.

SILVA, Andréia Maciel da; FERREIRA, Débora Priscila Costa. A utilização da plataforma kahoot como ferramenta de gamificação: uma contribuição para o ensino e a aprendizagem na educação básica. *Revista EDaPECI - Educação a Distância e Práticas Educativas Comunicacionais e Interculturais*, v. 22, n. 2, 2022. Disponível em: <<https://periodicos.ufs.br/edapeci/article/view/17748/12974>>. Citado na página 31.

SOUZA, Draylson Micael de; BATISTA, Marisa Helena da Silva; BARBOSA, Ellen Francine. Problemas e dificuldades no ensino e na aprendizagem de programação: um mapeamento sistemático. *Revista Brasileira de Informática na Educação*, 2016. Citado na página 16.

STARTUPS, Google for. *A Escassez de Profissionais de Tecnologia no Brasil e Seu Consequente Impacto no Ecossistema de Startups*. [S.l.], 2023. Disponível em: <[https://drive.google.com/file/d/1HdlvsAmLvVHkPW46\\_7TqTcHnbn7\\_\\_gB0/view](https://drive.google.com/file/d/1HdlvsAmLvVHkPW46_7TqTcHnbn7__gB0/view)>. Citado na página 12.

# APÊNDICE A – Requisitos de Funcionalidades Futuras

Neste apêndice são detalhados os requisitos funcionais previstos para futuras implementações do sistema. Estas funcionalidades estão majoritariamente relacionadas aos painéis administrativos, que permitirão o gerenciamento do conteúdo da plataforma por parte de professores e administradores.

Quadro 11 – Requisitos Funcionais para Funcionalidades Futuras

| Código | Título                     | Nível de Prioridade | Descrição                                                                                                                                                                                                                                          |
|--------|----------------------------|---------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| RF17   | Cadastrar Minigame         | Alta                | O sistema deve permitir o cadastro de novos minigames, contendo uma pergunta, quatro alternativas e a indicação da resposta correta.                                                                                                               |
| RF18   | Cadastrar Ensino           | Alta                | O sistema deve permitir o cadastro do conteúdo de ensino descritivo para uma aula.                                                                                                                                                                 |
| RF19   | Cadastrar Aula             | Alta                | O sistema deve permitir o cadastro de aulas e a alocação de suas respectivas atividades (ensinos e minigames).                                                                                                                                     |
| RF20   | Cadastrar Curso            | Alta                | O sistema deve permitir o cadastro de cursos e a alocação de suas respectivas aulas.                                                                                                                                                               |
| RF21   | Cadastrar Turma            | Alta                | O sistema deve permitir o cadastro de turmas e a alocação de seus respectivos cursos.                                                                                                                                                              |
| RF22   | Reutilizar Aulas em Cursos | Alta                | O sistema deve permitir que uma aula já cadastrada possa ser associada a múltiplos cursos, evitando a necessidade de duplicação de conteúdo.                                                                                                       |
| RF23   | Duplicar Curso             | Alta                | O sistema deve oferecer uma funcionalidade para criar uma cópia de um curso existente, incluindo toda a sua estrutura de aulas e atividades. O curso duplicado será uma nova entidade independente, permitindo modificações sem afetar o original. |
| RF24   | Editar Curso               | Alta                | O sistema deve permitir a alteração dos dados de um curso previamente cadastrado, incluindo seu título, descrição e a reordenação ou substituição de suas aulas.                                                                                   |
| RF25   | Cadastrar Minigame Visual  | Alta                | O sistema deve suportar um novo tipo de minigame que permita a utilização de imagens tanto no enunciado da pergunta quanto nas alternativas de resposta, possibilitando a criação de atividades mais visuais e interativas.                        |

Fonte: Autor (2025).