

DESENVOLVENDO PLATAFORMA DE ENSINO GAMIFICADO

Danilo Nogueira Silva
Graduando em Engenharia de Software – Uni-FACEF
danilo.nogueira1802@gmail.com

Igor Pereira Campos
Graduando em Engenharia de Software – Uni-FACEF
igorpcampos2004@gmail.com

Márcio Maestrelo Funes
Docente do Departamento de Computação do Uni-FACEF
marciofunes@facef.br

Resumo

A crescente integração de tecnologias digitais no ambiente educacional tem impulsionado a busca por metodologias de ensino mais dinâmicas e motivadoras. Nesse contexto, a gamificação surge como uma estratégia eficaz para aumentar o engajamento e a participação dos alunos, aplicando elementos de jogos em atividades de aprendizagem. Este trabalho apresenta o processo de desenvolvimento de uma plataforma de ensino gamificada, projetada para permitir que professores criem planos de aula interativos em formato de mapas, com exercícios e sistemas de progressão. A arquitetura do sistema foi construída utilizando uma abordagem de microserviços com NestJS no *back-end* e uma interface reativa com React e Next.js no *front-end*, garantindo escalabilidade e uma experiência de usuário fluida. O projeto resultou em uma aplicação funcional que demonstra o potencial da gamificação como ferramenta pedagógica, oferecendo um ambiente de aprendizado mais imersivo e estimulante, alinhado às novas demandas da educação.

Palavras-chave: Gamificação, Educação, Plataforma de Ensino, Tecnologia Educacional, Desenvolvimento de Software

Abstract

The increasing integration of digital technologies into the educational environment has driven the search for more dynamic and motivating teaching methodologies. In this context, gamification emerges as an effective strategy to increase student engagement and participation by applying game elements to learning activities. This work presents the development process of a gamified learning platform, designed to allow teachers to create interactive lesson plans in the format of maps, complete with exercises and progression systems. The system architecture was built using a microservices approach with NestJS for the back-end and a reactive interface with React and Next.js for the front-end, ensuring scalability and a fluid user experience. The project resulted in a functional application that demonstrates the potential of gamification as a pedagogical tool, offering a more immersive and stimulating learning environment aligned with the new demands of education.

Keywords: Gamification, Education, Learning Platform, Educational Technology, Software Development

1 Introdução

A educação contemporânea enfrenta o desafio constante de manter os alunos engajados em um mundo cada vez mais digital e interativo. As metodologias tradicionais de ensino, embora fundamentais, nem sempre conseguem capturar a atenção e a motivação necessárias para um aprendizado eficaz. Em resposta a essa demanda, a integração de tecnologias digitais tornou-se uma realidade indispensável nas salas de aula, abrindo portas para novas abordagens pedagógicas. Dentre elas, a gamificação se destaca como uma poderosa estratégia para transformar a experiência de aprendizagem.

A gamificação consiste na aplicação de mecânicas e dinâmicas de jogos — como pontos, níveis, recompensas e desafios — em contextos não lúdicos, com o objetivo de aumentar o engajamento, a motivação e a participação dos usuários. No cenário educacional, essa abordagem tem o potencial de converter o processo de aquisição de conhecimento em uma jornada mais envolvente e gratificante, estimulando a autonomia e a resolução de problemas.

Diante desse potencial, este trabalho tem como objetivo principal apresentar o planejamento, a arquitetura e o desenvolvimento de uma plataforma de ensino gamificada. A solução proposta permite que educadores estruturem seus planos de aula de forma visual e interativa, como mapas de fases, onde cada etapa representa um conteúdo ou uma atividade a ser concluída pelos alunos. A plataforma visa não apenas modernizar a entrega de conteúdo, mas também fornecer ferramentas que tornem o aprendizado uma experiência mais estimulante e mensurável. Assim, este trabalho tem como objetivo geral desenvolver uma plataforma de ensino gamificada que proporcione uma experiência interativa, capaz de aumentar o engajamento dos alunos e otimizar o planejamento pedagógico dos professores. Os objetivos específicos incluem: (i) oferecer uma ferramenta intuitiva para criação e gestão de planos de aula; (ii) aplicar elementos de gamificação para estimular a motivação dos estudantes; (iii) facilitar o acompanhamento do desempenho acadêmico; e (iv) promover um ambiente de aprendizagem mais autônomo e envolvente.

Para alcançar este objetivo, o trabalho está estruturado da seguinte forma: a primeira seção apresenta o referencial teórico que embasa os conceitos de tecnologias na educação e gamificação; em seguida, a seção de materiais e métodos detalha as tecnologias escolhidas e a metodologia de desenvolvimento adotada; por fim, são apresentados os resultados do desenvolvimento e as conclusões sobre o projeto, incluindo suas limitações e possibilidades de trabalhos futuros.

2 Referencial Teórico

Nessa seção estão apresentados os principais conceitos e estudos que fundamentam o desenvolvimento da plataforma gamificada, com foco na aplicação de tecnologias digitais na educação e nas estratégias de gamificação como recurso pedagógico. Ele é composto por um conjunto de teorias, conceitos, estudos e pesquisas já realizadas sobre o tema abordado. O objetivo principal do referencial teórico é situar a pesquisa no contexto do conhecimento existente, mostrando o que já foi discutido sobre o tema e como sua pesquisa se conecta a esses estudos prévios.

Além de fornecer uma base sólida para a pesquisa, o referencial teórico ajuda a definir os principais conceitos, identificar lacunas no conhecimento, justificar a relevância do estudo e indicar como a pesquisa contribui para o avanço do conhecimento na área.

2.1 Educação e Tecnologias

A utilização de tecnologias digitais na educação tem se intensificado ao longo dos anos e se consolidado como um dos maiores impulsionadores da transformação educacional. Conforme mencionado por Alves e Lopes (2024), a incorporação das tecnologias digitais nas escolas promove uma mudança substancial nos modelos pedagógicos, possibilitando novas formas de interação e personalização do ensino. Essa transição, no entanto, não ocorre de maneira simples. Ela exige uma adaptação tanto nas metodologias de ensino quanto na capacitação contínua dos educadores. Casagrande et al. (2023) destacam que, com a pandemia de COVID-19, as escolas enfrentam o desafio de migrar rapidamente para o ensino remoto, o que evidenciou limitações na infraestrutura tecnológica e na capacitação docente para o uso de plataformas digitais.

Além disso, a tecnologia permite que os alunos tenham acesso a uma maior variedade de conteúdos e recursos, como vídeos, animações e quizzes, facilitando um aprendizado mais ativo e autônomo. Porém, conforme apontado por Silva e Cavalcante (2024), essa transformação não deve se limitar apenas ao uso de novas ferramentas. A adoção dessas tecnologias exige uma mudança significativa na abordagem pedagógica, com foco em metodologias colaborativas e dinâmicas que envolvam mais o aluno no processo de aprendizagem. A desigualdade no acesso à tecnologia é uma preocupação constante, e como destacado por Sancinetti e Xavier (2023), é preciso buscar estratégias que garantam o acesso universal às ferramentas digitais para que todos os estudantes, independentemente da sua localização ou classe social, possam se beneficiar dessas inovações.

2.2 Gamificação na Educação

A gamificação consiste na incorporação de elementos característicos dos jogos — como pontos, recompensas, rankings e níveis — em ambientes não lúdicos, com o intuito de tornar atividades mais atrativas e engajadoras. No contexto educacional, essa abordagem transforma o processo de aprendizagem em uma experiência interativa, promovendo maior envolvimento dos estudantes. Silva e Castro (2019) destacam que a gamificação é eficaz para estimular a participação ativa dos alunos, ao criar um ambiente competitivo e gratificante que favorece o alcance de metas educacionais. Isso é alcançado por meio da criação de um ambiente de aprendizado mais competitivo e gratificante, o que naturalmente incentiva os alunos a se esforçarem para atingir metas e avançar no conteúdo.

Contudo, é importante destacar que a gamificação não deve ser confundida com o aprendizado baseado em jogos. Embora ambas as abordagens utilizem elementos de jogos, a principal diferença reside no objetivo e na forma como o conteúdo é estruturado. Silva e Cavalcante (2024) explicam que, enquanto a gamificação busca aumentar o engajamento com o conteúdo educacional por meio de mecânicas de jogo, o aprendizado baseado em jogos envolve o uso de jogos como uma ferramenta central para o ensino.

2.2.1 Benefícios da Gamificação

Os benefícios da gamificação no ensino são amplamente discutidos na literatura. Diversos estudos apontam que a gamificação contribui significativamente para o aumento do engajamento e da motivação dos alunos, refletindo diretamente na realização de tarefas e no cumprimento de objetivos educacionais. Além disso, o envolvimento ativo proporcionado por essa abordagem favorece a retenção de conhecimento, ao estimular a memorização e a compreensão dos conteúdos por meio de experiências práticas e interativas.

A mesma fonte destaca que a gamificação também tem efeitos positivos no desenvolvimento de habilidades socioemocionais, como trabalho em equipe, colaboração e resolução de problemas, uma vez que muitas plataformas gamificadas incentivam a interação entre os alunos.

Sancinetti e Xavier (2023) vão além, afirmando que a gamificação pode ser um fator importante para tornar o processo educacional mais inclusivo, ao possibilitar uma personalização da aprendizagem. Isso é particularmente relevante para alunos com dificuldades de aprendizagem, que podem se beneficiar de desafios adaptados ao seu ritmo e estilo de aprendizado, além do feedback constante que a gamificação oferece.

2.2.2 Elementos da Gamificação no Ensino

Os principais elementos da gamificação aplicados ao ensino incluem recompensas, feedback imediato, progressão por níveis e desafios. Recompensas como pontos, medalhas e rankings funcionam como estímulos que reforçam o comportamento positivo e incentivam o alcance de metas. O feedback imediato permite ao aluno identificar seus erros e acertos com rapidez, promovendo ajustes contínuos e fortalecendo sua autoconfiança. A progressão por níveis, quando bem planejada, mantém o interesse dos estudantes ao oferecer desafios compatíveis com seu nível de habilidade, criando uma jornada de aprendizado estruturada e motivadora.

Silva e Cavalcante (2024) também destacam que o uso de recompensas e feedback pode ajudar a fortalecer a autoconfiança do aluno, criando uma sensação de conquista cada vez que ele atinge uma meta. Isso contribui para um aprendizado mais eficaz e contínuo, pois o aluno percebe que suas ações têm resultados concretos.

Outro aspecto fundamental da gamificação é a progressão e os desafios. A gamificação permite que os alunos avancem por diferentes níveis, enfrentando desafios cada vez mais complexos à medida que progredem. Isso mantém o interesse do aluno ao longo do tempo, pois ele se sente constantemente desafiado, mas sem ser sobrecarregado. Segundo Silva e Cavalcante (2024), essa progressão deve ser cuidadosamente planejada para garantir que os desafios sejam adaptados ao nível de habilidade de cada aluno.

Sancinetti e Xavier (2023) ressaltam que a sensação de conquista ao superar desafios e avançar para o próximo nível pode aumentar significativamente a motivação do aluno. Além disso, o modelo de progressão gradual oferece uma forma

clara e estruturada de acompanhar o aprendizado, o que é essencial para manter o engajamento dos estudantes.

3 Metodologia

A plataforma foi desenvolvida com foco em proporcionar uma experiência fluida e interativa, utilizando tecnologias atuais que favorecem escalabilidade, modularidade e desempenho. O backend foi construído com a arquitetura de microserviços utilizando o framework NestJS, que permite uma escalabilidade modular e facilita a manutenção contínua. O front-end foi desenvolvido com React e Next.js, oferecendo uma interface dinâmica e responsiva que se adapta a diferentes dispositivos. Para a comunicação entre os serviços e a atualização em tempo real das interações dos usuários, foi utilizado RabbitMQ para mensageria assíncrona, permitindo o processamento eficiente de dados sem comprometer o desempenho.

3.1 Métodos

O projeto adotou uma abordagem baseada em metodologias ágeis, com ciclos curtos de desenvolvimento (sprints) e validação contínua das funcionalidades implementadas. Todo o processo foi organizado de forma a garantir eficiência, qualidade e alinhamento com os objetivos propostos

3.1.1 Levantamento de Requisitos

A primeira etapa do desenvolvimento foi centrada na definição dos objetivos principais da plataforma, partindo da visão da equipe sobre os desafios do processo de ensino-aprendizagem. A concepção do projeto baseou-se em um brainstorm interno para estruturar uma solução gamificada que agregasse valor à experiência educacional. Como resultado, definiu-se um conjunto de requisitos essenciais, entre os quais se destacam:

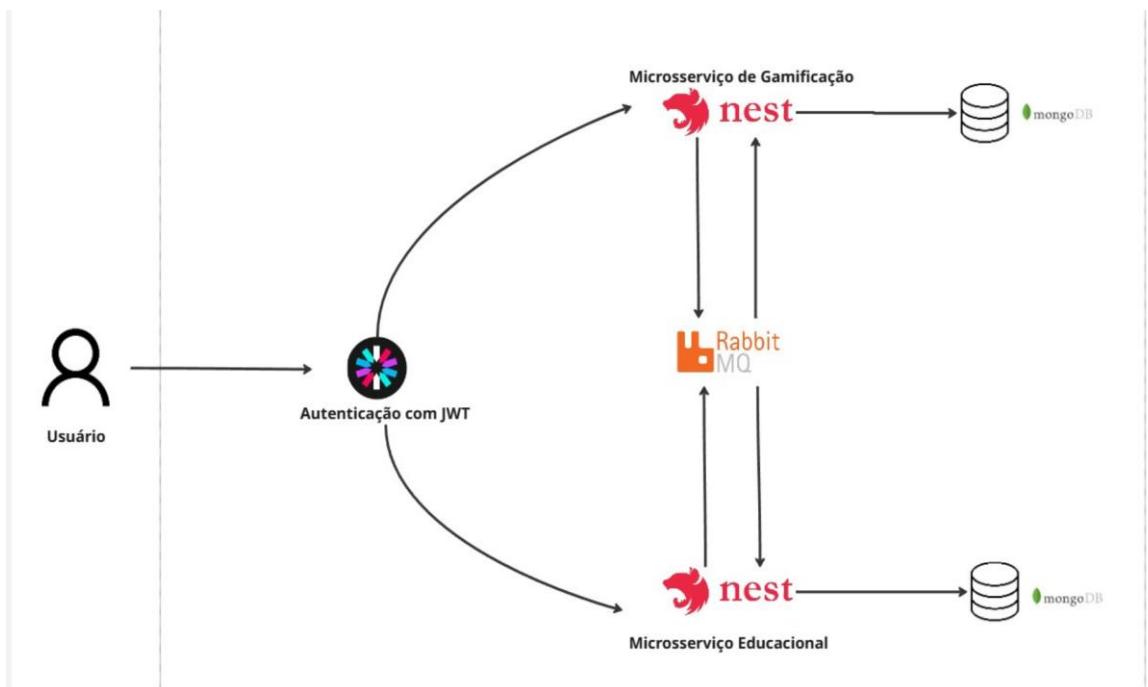
- Criação de planos de aula, com possibilidade de enviar convites para os alunos participarem dele;
- Cadastro de exercícios que funcionam como fases do mapa;
- Sistema de pontuação e ranking dos alunos;
- Dashboard para monitoramento do desempenho;
- Possibilidade de incluir recursos audiovisuais nas aulas.

3.1.2 Modelagem e Arquitetura do Sistema

Com os requisitos em mãos, foi realizada a modelagem do sistema, utilizando diagramas UML para representar a estrutura das entidades, os fluxos de dados e as interações entre os diferentes componentes.

A arquitetura adotada seguiu o modelo Client-Server, com separação entre as camadas de apresentação e lógica de negócios. A comunicação foi realizada por meio de APIs REST, assegurando segurança, desempenho e escalabilidade. Para o armazenamento dos dados, optou-se pela utilização do MongoDB, cuja estrutura flexível permitiu a modelagem dinâmica das informações necessárias.

Figura 1 - Arquitetura do software



Fonte: Os autores.

3.1.3 Desenvolvimento

O desenvolvimento do projeto foi organizado em sprints semanais, cada uma focada na entrega de uma funcionalidade específica ou na melhoria de um módulo já existente, seguindo práticas de metodologias ágeis que valorizam ciclos curtos de desenvolvimento e feedback contínuo. Foram incorporadas práticas de metodologias ágeis, como scrum e kanban, além de testes unitários e integração contínua, com o objetivo de aumentar a confiabilidade e a qualidade do código, como afirma Beck, K., & Andres, C. (2005).

Além disso, a integração contínua e a entrega contínua (CI/CD) foram adotadas como pilares do processo de desenvolvimento, permitindo que cada nova versão fosse validada e disponibilizada de maneira ágil e estável. Esse modelo, segundo Shahin, Babar & Zhu (2017), representa um avanço nas práticas de engenharia de software, pois reduz o tempo de entrega, melhora a detecção de erros e fortalece a colaboração entre os membros da equipe. Dessa forma, as boas práticas implementadas no projeto — como versionamento em GitHub, revisões de código e workflows automatizados — alinharam-se às recomendações encontradas na literatura, assegurando qualidade, segurança e eficiência no ciclo de vida do software.

3.1.4 Entrega e Documentação

Concluído o desenvolvimento, todas as etapas do projeto foram devidamente documentadas. A documentação contemplou o código-fonte, diagramas

de arquitetura, instruções de instalação e um manual de uso voltado para professores e alunos, visando facilitar a adoção e evolução da plataforma.

Essa documentação teve como objetivo não apenas registrar o processo de desenvolvimento, mas também facilitar a continuidade e evolução do projeto, seja para fins acadêmicos ou para possíveis aplicações futuras.

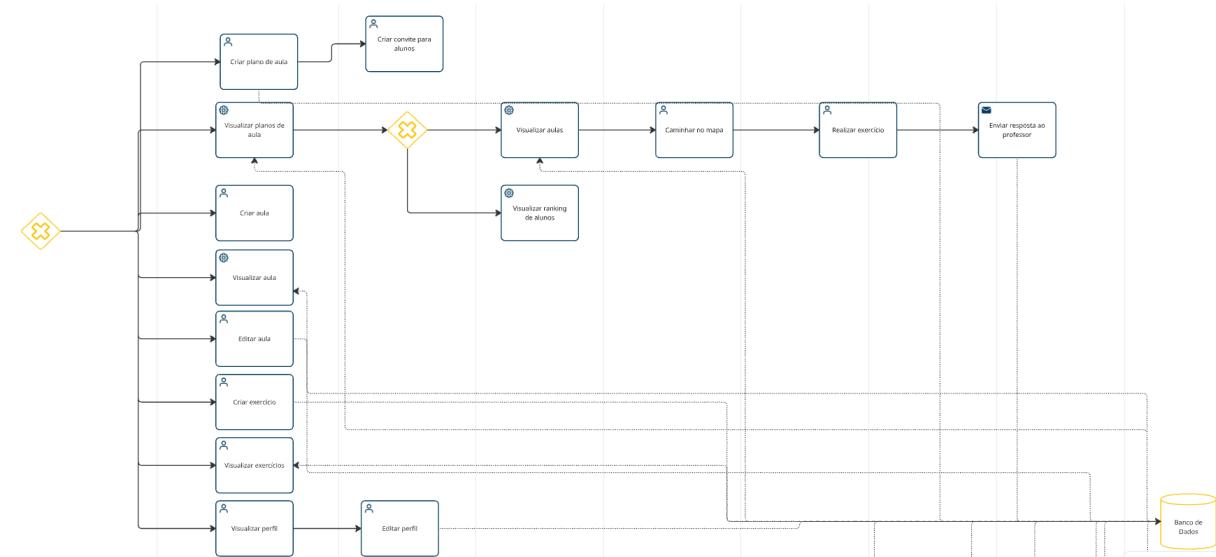
3.2 Diagramas

Os diagramas utilizados foram fundamentais para representar a estrutura do sistema, os fluxos de dados e as interações entre os componentes. Foram aplicados modelos de processo, classes e sequência, que contribuíram para a documentação técnica e para o alinhamento da equipe com os requisitos do projeto.

3.2.1 BPMN

O diagrama de processo modelado em Business Process Model and Notation (Figura 2) descreve o fluxo de atividades realizadas pelo professor dentro da plataforma. Entre as ações representadas estão: criar plano de aula, adicionar exercícios, associar aulas ao plano e publicar o conteúdo. O fluxo também contempla decisões e interações com o sistema, permitindo visualizar como as etapas se encadeiam até a conclusão do processo pedagógico. Esse modelo favorece o entendimento do funcionamento da plataforma do ponto de vista do usuário.

Figura 2 - Diagrama de Processo (BPMN)



Fonte: Os autores.

3.2.2 Diagrama de entidades

O diagrama de entidades (Figura 3) representa a estrutura do banco de dados da plataforma. Nele estão modeladas as principais entidades do sistema, como User (usuário), LessonPlan (plano de aula), Lesson (aula) e Exercise (exercício). O

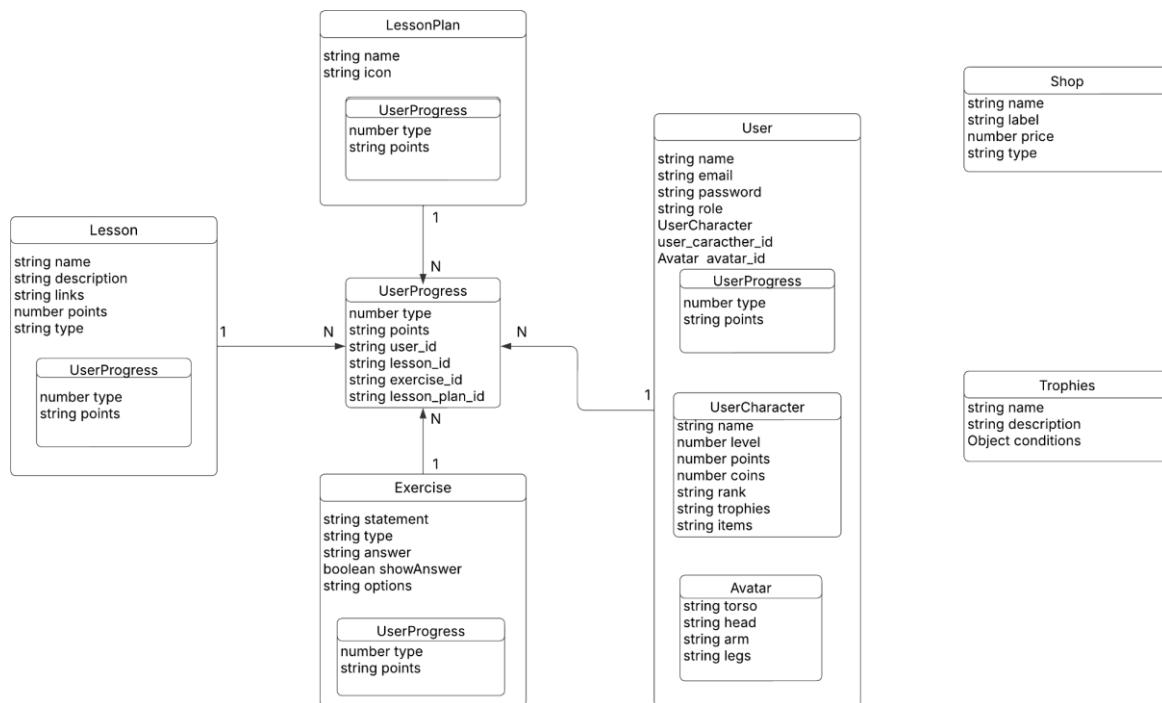
modelo também detalha entidades de gamificação, como UserCharacter, Avatar, Shop e Trophies.

A estrutura de aprendizado é hierárquica: um LessonPlan contém várias Lessons, e essas aulas são compostas por Exercises. A entidade User possui um campo role (papel), que provavelmente distingue alunos de professores.

Para o acompanhamento do desempenho, a estrutura centraliza os dados em uma única entidade chamada UserProgress. Diferente do mencionado no texto original, não existem tabelas separadas como user_class_progress ou user_map_progress. A entidade UserProgress conecta o User (user_id) ao conteúdo que ele consome, registrando referências ao LessonPlan (lesson_plan_id), Lesson (lesson_id) e Exercise (exercise_id). Isso permite um acompanhamento granular do desempenho do aluno em todas as etapas.

As relações entre as entidades são gerenciadas por referências (como user_id e lesson_id na entidade UserProgress), o que permite a conexão entre os diferentes documentos e facilita consultas complexas.

Figura 3 - Diagrama de Entidades do Banco



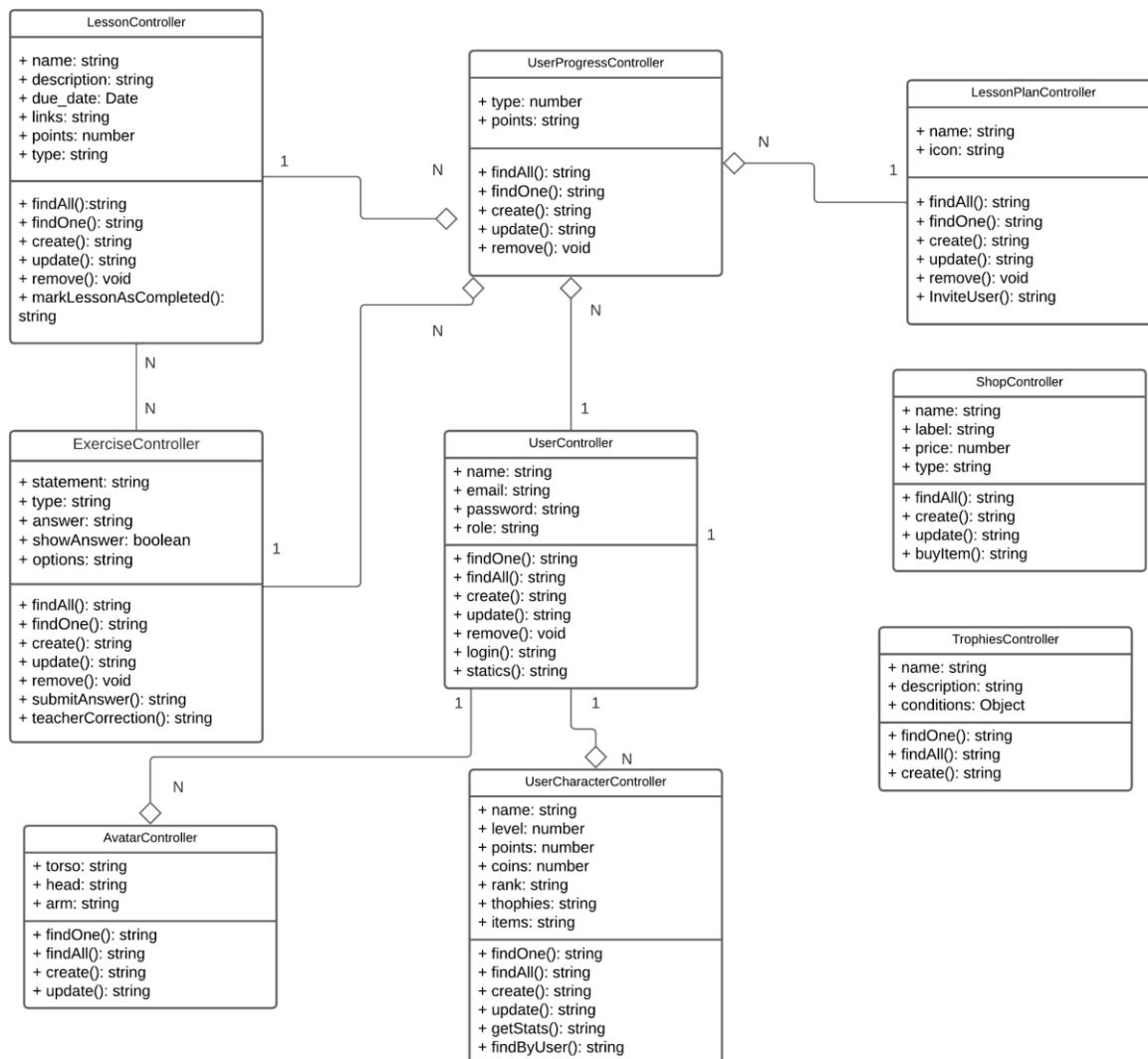
Fonte: Os autores.

3.2.3 Classes

O diagrama de classes (Figura 4) mostra como os controladores da aplicação foram organizados, seguindo o padrão de arquitetura comum em frameworks como NestJS. Cada controlador é responsável por uma entidade do sistema, implementando operações de criação, leitura, atualização e exclusão (CRUD). Por exemplo, o UserController gerencia os dados dos usuários, enquanto o

ClassController lida com as aulas, e o ExerciseController cuida dos exercícios. Essa divisão garante coesão e facilita a manutenção da aplicação.

Figura 4 - Diagrama de Classes

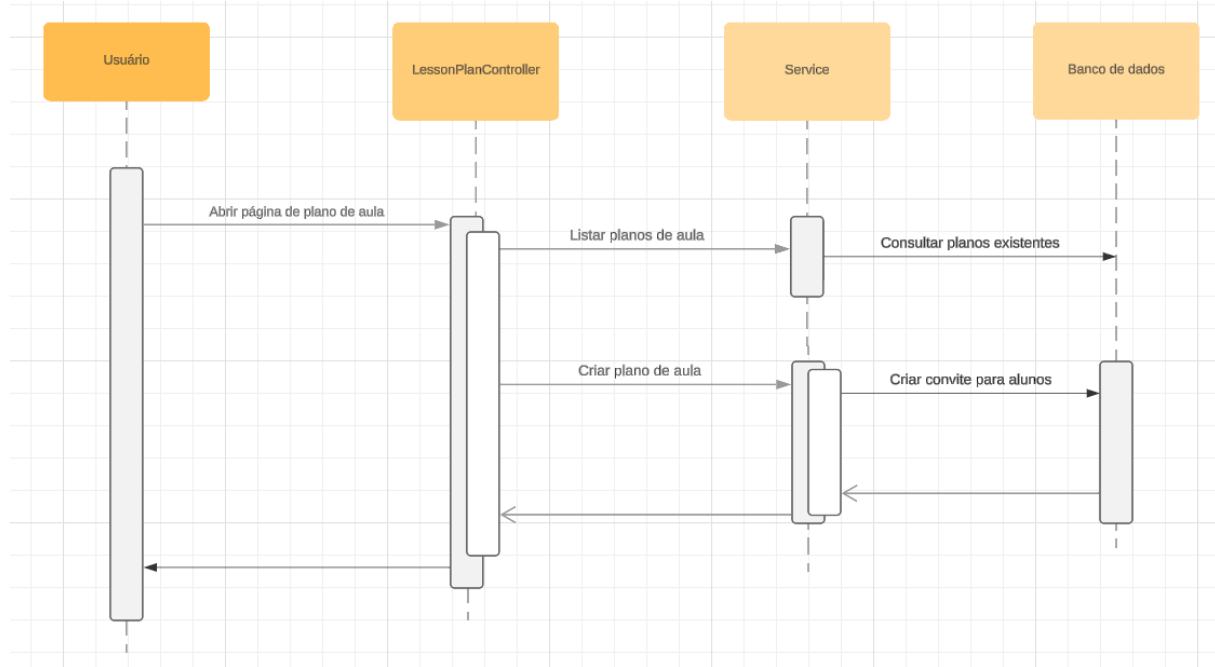


Fonte: Os autores.

3.2.4 Sequência

O diagrama de sequência representa o fluxo de interação entre o professor e o sistema durante a criação de um plano de aula. Ele descreve de forma cronológica como as ações são iniciadas e como o sistema responde a cada uma delas. Como pode ser observado na figura abaixo (Figura 5), o professor acessa a listagem de planos, cria um novo plano e, em seguida, gera um convite para os alunos. Essa modelagem é útil para entender a ordem e o comportamento dos componentes em tempo de execução, auxiliando na validação da lógica da aplicação.

Figura 5 - Diagrama de Sequência



Fonte: Os autores.

3.3 Tecnologias

O desenvolvimento da plataforma gamificada contou com um conjunto de tecnologias selecionadas estrategicamente para garantir desempenho, escalabilidade e interatividade. A seleção dos materiais utilizados não foi feita de forma aleatória; cada escolha foi fundamentada nas necessidades do projeto e no potencial que cada tecnologia poderia agregar à proposta. A seguir, apresento os principais recursos adotados, junto às razões que justificam cada decisão.

3.3.1 Node.js

O Node.js foi escolhido por sua alta performance e capacidade de gerenciar múltiplas conexões simultâneas de forma eficiente, característica essencial para aplicações com grande volume de interação entre usuários. Como o projeto envolve a interação constante entre professores e alunos, era fundamental utilizar uma tecnologia escalável e eficiente, baseada em seu modelo de I/O não bloqueante (Node.js Foundation, s.d.).

Além do desempenho, outro fator determinante foi o fato do Node.js ser baseado em JavaScript. Isso facilitou bastante o desenvolvimento, pois possibilitou que toda a aplicação — tanto no front-end quanto no back-end — fosse construída utilizando a mesma linguagem, o que diminuiu a complexidade e aumentou a produtividade da equipe. A vasta comunidade e a grande quantidade de bibliotecas também pesaram na escolha, já que garantem suporte constante e atualização frequente.

3.3.2 Nest.JS

Na camada de back-end, foi utilizado o NestJS, um framework que se destaca por sua estrutura organizada e modular. A escolha se deve, principalmente, ao fato de o NestJS ser construído sobre o Node.js, mas trazer consigo uma arquitetura robusta inspirada em conceitos como Inversão de Dependência e Domain Driven Design (DDD) (NestJS Documentation, s.d.).

A integração nativa com TypeScript é outro diferencial do NestJS, pois aumenta a segurança no desenvolvimento ao minimizar erros de tipagem e facilitar a manutenção do código (NestJS Documentation, s.d.). Além disso, sua estrutura clara e baseada em módulos facilitou bastante a manutenção e evolução do código, pontos essenciais para um projeto acadêmico de médio a longo prazo.

3.3.3 React.Js

No front-end, a tecnologia escolhida foi o ReactJS. Essa decisão foi motivada pela ampla adoção dessa biblioteca na indústria e pela sua proposta de desenvolvimento baseado em componentes reutilizáveis, o que favoreceu a construção de uma interface intuitiva, fluida e escalável (React Documentation, s.d.).

A capacidade do React de atualizar a interface de forma rápida, através do seu DOM Virtual, e sem recarregamento de página foi essencial para garantir uma experiência fluida e responsiva, especialmente em um ambiente gamificado. Além disso, sua compatibilidade com ferramentas modernas como Tailwind CSS e React Query contribuiu diretamente para o desempenho e qualidade visual do sistema.

3.3.4 Next.Js

Para complementar o uso do React, optou-se pelo framework Next.js. O Next.js foi adotado por oferecer recursos como Server Side Rendering (SSR) e Static Site Generation (SSG), que contribuem para o desempenho da aplicação e sua otimização para mecanismos de busca (SEO) (Vercel, s.d.). Outro aspecto que pesou na decisão foi a facilidade de integração do Next.js com APIs, além de permitir a construção de aplicações fullstack. Com ele, foi possível equilibrar bem as páginas estáticas, como a tela inicial da plataforma, e as páginas dinâmicas, como as fases e atividades do mapa gamificado.

3.3.5 MongoDB

No armazenamento dos dados, foi adotado o banco MongoDB. A decisão por um banco de dados NoSQL se deu devido à flexibilidade que ele oferece na estruturação das informações. A escolha pelo MongoDB se deve à sua estrutura flexível baseada em documentos (BSON/JSON), que facilita o armazenamento de dados heterogêneos, como perfis de usuários, progresso acadêmico e conteúdos personalizados (MongoDB, Inc., s.d.).

Além disso, o MongoDB possui alta escalabilidade horizontal e desempenho satisfatório mesmo com grandes volumes de dados, o que foi determinante para garantir que a plataforma pudesse crescer sem comprometer sua eficiência.

3.3.6 Trello

O Trello, baseado na metodologia Kanban, foi utilizado para gerenciar o fluxo de trabalho, permitindo o acompanhamento das tarefas e o cumprimento dos prazos estabelecidos (Atlassian, s.d.). A utilização do Trello também facilitou a visualização do progresso do projeto, dividindo as tarefas em categorias como "A Fazer", "Em Andamento" e "Concluído", o que contribuiu diretamente para uma organização mais clara e produtiva.

3.3.7 Visual Studio Code

O editor de código escolhido foi o Visual Studio Code (VSCode), principalmente por ser um ambiente leve, personalizável e compatível com todas as tecnologias utilizadas no projeto. Além disso, conta com uma grande variedade de extensões que auxiliaram no desenvolvimento, como Prettier, ESLint e suporte para Docker e MongoDB. A integração nativa com ferramentas de versionamento, como o GitHub, tornou o controle do código-fonte mais ágil e seguro durante o desenvolvimento (Microsoft, s.d.-b).

3.3.8 TypeScript

Uma das decisões técnicas mais relevantes foi a utilização do TypeScript em todo o projeto. O TypeScript foi escolhido por oferecer recursos que vão além do JavaScript tradicional, como tipagem estática, inferência de tipos e melhor suporte para desenvolvimento em larga escala (Microsoft, s.d.-a).

O uso do TypeScript aumentou a segurança do projeto ao evitar erros de tipagem, além de melhorar a legibilidade e facilitar o compartilhamento de tipos entre as camadas front-end e back-end.

3.3.9 RabbitMQ

Para a comunicação entre os diferentes serviços de back-end, foi adotado o RabbitMQ. A escolha por um message broker (intermediador de mensagens) foi estratégica para garantir a comunicação assíncrona e o desacoplamento da arquitetura. O RabbitMQ permite que os serviços operem de forma independente, publicando mensagens em filas sem a necessidade de esperar uma resposta imediata (RabbitMQ, s.d.). Isso garante que, mesmo se um microsserviço estiver temporariamente indisponível, as solicitações não sejam perdidas, aumentando a resiliência e a escalabilidade geral do sistema.

4 Desenvolvimento

O processo de desenvolvimento da plataforma seguiu um conjunto estruturado de atividades, conforme definido por Pressman (2016), envolvendo planejamento, implementação, testes e evolução contínua do sistema.

Nesta seção, é descrito o processo de construção da aplicação web para a plataforma de ensino gamificada, destacando-se tanto os aspectos pedagógicos quanto os elementos de gamificação. A proposta buscou integrar a criação de planos de aula interativos, a produção de conteúdos diversificados, a atribuição de níveis e troféus como forma de incentivo e, por fim, a implementação de uma loja de cosméticos que valoriza a personalização da experiência do aluno.

4.1 Implementação dos Planos de Aula

Os planos de aula constituem o núcleo pedagógico da plataforma, organizados como sequências de atividades que estruturam a jornada de aprendizagem dos estudantes. O professor, ao criar um plano, pode definir objetivos de aprendizagem, estabelecer a ordem de execução das atividades e disponibilizar recursos complementares que enriquecem o processo educacional. A funcionalidade de envio de convites permite ao professor gerenciar o acesso dos alunos aos planos de aula, promovendo uma organização eficiente das turmas. Além disso, a plataforma disponibiliza um sistema de ranking que posiciona os alunos conforme seu desempenho nas atividades, fomentando um ambiente de competitividade saudável e de incentivo à participação. Outro ponto de destaque é o suporte à correção e atribuição de notas pelo professor, que pode avaliar individualmente as respostas dos estudantes, registrar feedbacks e atribuir pontuações, fortalecendo a interação entre docente e discente e oferecendo maior transparência no acompanhamento do aprendizado.

4.2 Criação de aulas, exercícios e trabalhos

A aplicação também contempla recursos robustos voltados para a criação de conteúdos didáticos, permitindo que os professores elaborem aulas, listas de exercícios e trabalhos em diferentes formatos. As aulas permitem a integração de textos explicativos, vídeos, imagens e outros recursos multimídia, com o objetivo de tornar o conteúdo mais dinâmico e acessível ao aluno. Já os exercícios foram projetados para atender a diferentes metodologias pedagógicas, contemplando questões de múltipla escolha, verdadeiro ou falso e questões abertas, nas quais o estudante pode elaborar uma resposta dissertativa. Essa diversidade possibilita que os docentes se adaptem às atividades de acordo com os objetivos específicos de cada disciplina, promovendo maior flexibilidade no processo de ensino-aprendizagem. Além disso, a plataforma oferece a funcionalidade de criação de listas de exercícios, que agrupam diversas questões em blocos avaliativos, garantindo um acompanhamento mais consistente do desempenho dos alunos. Para trabalhos, o sistema permite que os estudantes realizem o upload de arquivos, ampliando o escopo das atividades e possibilitando a entrega de relatórios, apresentações e documentos diversos. Esse conjunto de funcionalidades proporciona ao professor meios variados de avaliar o conhecimento, ao mesmo tempo em que assegura ao aluno diferentes formas de expressão e participação.

4.3 Nível e troféus

A gamificação foi incorporada à plataforma como uma estratégia central para aumentar o engajamento e a motivação dos estudantes. Cada atividade concluída gera pontos de experiência (XP), cuja quantidade varia de acordo com a dificuldade da tarefa e o desempenho do aluno na resolução. A soma desses pontos permite que o estudante progrida dentro do sistema, alcançando novos níveis que representam a evolução de sua jornada educacional. A progressão por níveis foi projetada para reforçar o senso de evolução do aluno, estimulando sua dedicação por meio de metas claras e recompensas simbólicas. Além disso, o sistema de recompensas foi enriquecido pela implementação de troféus, concedidos quando o

usuário atinge metas específicas, como a realização de determinado número de atividades, a obtenção de notas máximas em exercícios ou a conclusão de um conjunto de aulas. Esses troféus funcionam como símbolos de conquista e reconhecimento, gerando um sentimento de realização e incentivando os estudantes a superarem novos desafios. Dessa forma, a lógica de pontos, níveis e troféus não apenas reforça a motivação individual, mas também cria um ambiente competitivo e colaborativo, no qual os alunos se sentem desafiados a progredir continuamente.

4.4 Loja de cosmético

Complementando a mecânica de gamificação, a plataforma introduziu uma economia virtual na qual os alunos, ao realizarem atividades, recebem moedas digitais que podem ser utilizadas na loja de cosméticos. Esse espaço foi concebido para oferecer itens de personalização do avatar do usuário, permitindo que cada estudante configure sua identidade visual dentro da plataforma. Os itens disponíveis possuem diferentes graus de raridade, classificados em comum, incomum, raro, épico e lendário, de modo a estimular o desejo de conquista e a valorização dos esforços realizados. Essa funcionalidade cria uma camada adicional de engajamento, pois vincula o desempenho acadêmico à possibilidade de personalização estética, tornando o aprendizado mais atrativo. Além de reforçar a motivação, a loja de cosméticos contribui para o senso de pertencimento e identidade dos alunos, que podem exibir suas conquistas não apenas em termos acadêmicos, mas também na forma de diferenciação visual de seus personagens. Ao unir progresso pedagógico e recompensas virtuais, o sistema consolida a experiência gamificada, transformando o aprendizado em uma jornada envolvente e personalizada.

5 Aplicação Web

A aplicação web desenvolvida concretiza os requisitos pedagógicos e de gamificação definidos nas etapas iniciais do projeto. Diferentemente da fase de prototipação, voltada à validação de fluxos e usabilidade, esta seção apresenta as telas finais da plataforma, já em funcionamento. O foco esteve na criação de um ambiente visualmente atrativo, responsivo e intuitivo, capaz de atender tanto professores quanto alunos em suas necessidades educacionais.

5.1 Tela de Login

A tela de login (Figura 6) é a porta de entrada da aplicação e foi projetada para garantir segurança e simplicidade no processo de autenticação. O usuário insere seu e-mail e senha, podendo ainda criar uma nova conta caso seja seu primeiro acesso. O layout prioriza a clareza e a objetividade, reduzindo barreiras de usabilidade e facilitando o início da experiência dentro da plataforma.

Figura 6 - Login

Fazer login na plataforma

Insira seu email e senha para entrar!

Email *

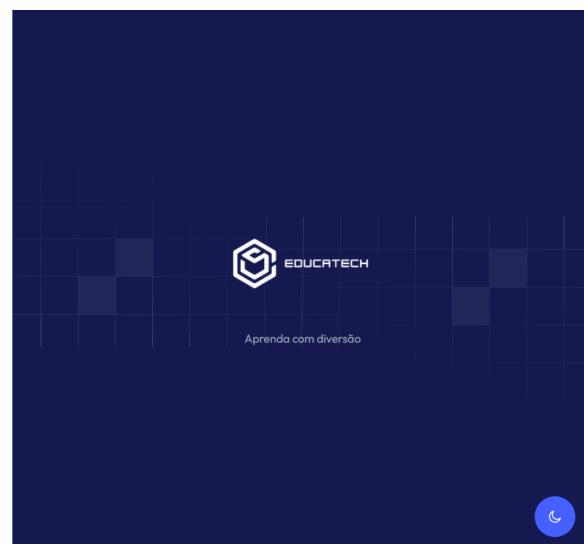
info@gmail.com

Senha *

Insira sua senha

Entrar

Não possui uma conta? [Criar conta](#)



Fonte: Os autores.

5.2 Tela de Planos de Aula

A tela de planos de aula (Figura 7) concentra a principal funcionalidade docente da aplicação. Cada plano é exibido em formato de cartão, contendo informações sobre disciplina, autor e barra de progresso. O professor pode criar novos planos, gerenciá-los e acompanhar o desempenho dos alunos vinculados a cada um. A interface facilita a organização do trabalho pedagógico, permitindo a visualização clara dos planos de aula e o envio de convites aos estudantes para participação nas atividades.

Figura 7 - Plano de aula

The screenshot shows the Educatech platform's interface for teachers. On the left, there is a sidebar with a logo and navigation links: 'Planos de Aula' (highlighted in blue), 'Exercícios', 'Aulas', 'Loja', and 'Perfil'. Below this, under 'Professor', there is a link. The main area displays five cards representing lesson plans: 'História 8º Ano A' (professor, 100% completion), 'Física' (professor, 100%), 'Português' (professor, 100%), 'Química' (professor, 100%), and 'Geografia' (professor, 100%). Each card has a three-dot menu icon and a progress bar at the bottom. At the bottom of the main area, there is a blue button labeled 'Adicionar plano de aula' with a file icon.

Fonte: Os autores.

5.3 Tela de Exercícios e Correção

Na tela de exercícios (Figura 8), os conteúdos cadastrados são apresentados em formato de tabela, exibindo dados como título, tipo de questão, nota, plano de aula relacionado e dificuldade. A visualização em formato de tabela amplia o controle do professor sobre os exercícios cadastrados. A tela de correção (Figura 9) permite a avaliação das respostas dos alunos, com atribuição de notas e registro de feedbacks. Além disso, o processo de criação de exercícios é totalmente integrado ao sistema, permitindo selecionar o tipo de questão (múltipla escolha, verdadeiro ou falso ou aberta), cadastrar alternativas e configurar critérios de avaliação.

Figura 8 - Exercícios

Enunciado	Nota	Dificuldade	Tipo de Questão	Plano de Aula	Ações
Exercício Teste	0	Fácil	Aberta	História 8º Ano A, Geografia	
Qual é a capital do Estado de São Paulo?	2	Médio	Aberta	História 8º Ano A	
Antiope Poesia, não será esse o sentido em que ainda te escrevo: flor! (Te escrevo: flor! Não uma flor, nem aquela flor-Virtude - em disfarçados urinôis). Flor é a palavra flor; verso inscrito no verso, como os manhãs no tempo. Flor é o salto da ave para o voo: o... saltofaria do sono quando seu tecido se rompe; é uma explosão posta a funcionar.	4	Difícil	Múltipla Escolha	História 8º Ano A	

Fonte: Os autores.

Figura 9 - Correção de exercícios

Fonte: Os autores.

5.4 Tela de Aulas

A tela de aulas (Figura 10) organiza os conteúdos disponibilizados pelos professores, permitindo anexar materiais multimídia, como vídeos, links e documentos. Os conteúdos são apresentados em formato de lista, com informações como nome, tipo, análise e vínculo com o plano de aula. Essa estrutura favorece a flexibilidade pedagógica e contribui para um processo de ensino mais dinâmico, ao permitir a integração de diversos recursos educacionais.

Figura 10 -Aulas

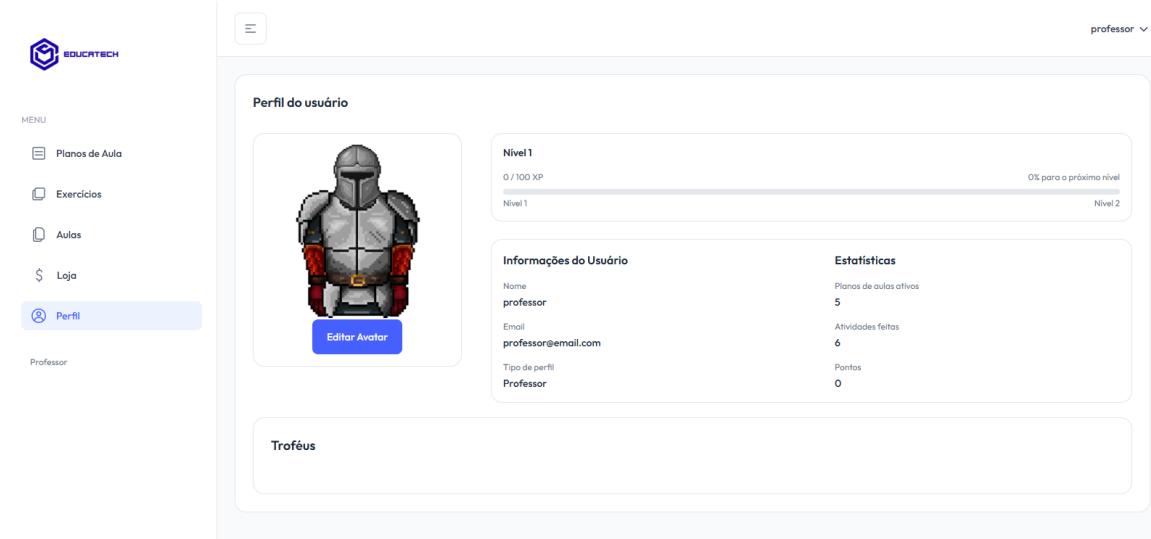
Nome	Nota	Arquivo	Links	Tipo	Plano de aula	Ações
Segunda Guerra mundial	2	Redes Neurais.docx	https://link.com	Aula	História 8º Ano A	
Primeira guerra mundial	Sem nota atribuída		https://www.youtube.com/watch?v=EOYAN2CWwM, https://www.youtube.com/watch?v=EOYAN2CWwM	Aula	História 8º Ano A	
Entrega de Trabalho	Sem nota atribuída			Trabalho	História 8º Ano A, História 8º Ano A, Física	

Fonte: Os autores.

5.5 Tela de Perfil do Usuário

O perfil do usuário (Figura 11) reúne informações acadêmicas e gamificadas em um só espaço. Além de dados pessoais como nome, e-mail e tipo de perfil, a tela apresenta estatísticas de desempenho, como planos de aula ativos, atividades realizadas e pontos acumulados. O avatar do estudante pode ser personalizado e exibe os troféus conquistados, reforçando o aspecto motivacional. A tela de perfil reforça o senso de identidade e progressão do aluno, ao reunir dados acadêmicos e elementos gamificados em um único espaço, permitindo o acompanhamento visual da evolução individual.

Figura 11 - Perfil

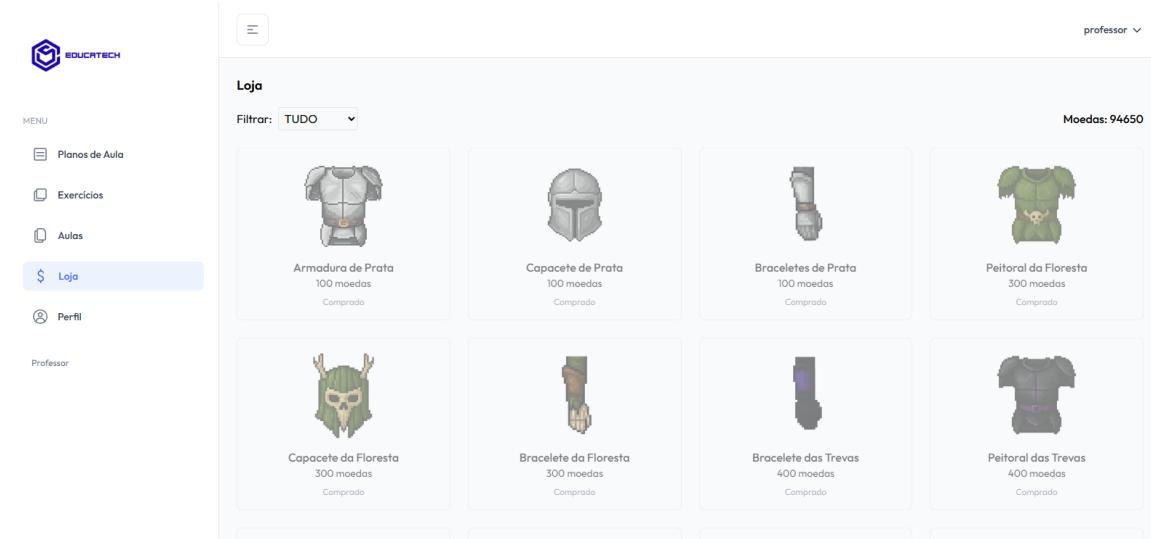


Fonte: Os autores.

5.6 Tela da Loja

A tela da loja (Figura 12) apresenta os itens cosméticos disponíveis para compra com as moedas virtuais adquiridas ao longo das atividades. Os itens variam de acordo com sua raridade, sendo classificados em comuns, raros, épicos e lendários. A loja fortalece a economia gamificada da plataforma ao permitir que os alunos utilizem moedas virtuais, obtidas por desempenho acadêmico, para personalizar seus avatares, reforçando o engajamento e a identidade visual no ambiente educacional.

Figura 12 - Loja



Fonte: Os autores.

6 Conclusão e Trabalhos Futuros

O desenvolvimento desta plataforma de ensino gamificada buscou atender ao desafio de engajar alunos em um ambiente educacional cada vez mais digital, culminando na entrega de uma aplicação robusta e funcional. O uso da arquitetura de microsserviços e de tecnologias modernas não só viabilizou a criação de uma experiência interativa, como também se mostrou um diferencial estratégico, facilitando a análise de métricas de desempenho dos alunos e permitindo a futura integração de novos módulos e funcionalidades de forma ágil e escalável.

A implementação de recursos possibilitou a mensuração do desempenho dos estudantes e a organização do processo de ensino-aprendizagem. Foram construídos módulos específicos para exercícios, listas, aulas e trabalhos, possibilitando o acompanhamento do progresso discente de forma contínua. Essa estrutura oferece aos educadores indicadores valiosos, permitindo identificar avanços, dificuldades e o nível de engajamento dos alunos. Além disso, a utilização de uma arquitetura modular garante que novos conteúdos possam ser integrados sem comprometer a estabilidade do sistema.

A plataforma possibilita, ainda, alinhar os objetivos pedagógicos com a experiência gamificada, por meio de pontos, níveis e troféus, reforçando a motivação e incentivando a participação ativa dos estudantes. A flexibilidade da arquitetura baseada em microsserviços favorece a reutilização de componentes e a integração de novas funcionalidades, assegurando evolução constante do sistema. Dessa forma, o projeto se consolida como uma solução escalável e adaptável, que pode ser ampliada e aplicada em diferentes contextos educacionais.

Como implementação futura, destaca-se o desenvolvimento de um aplicativo mobile nativo, que complementa a versão web já responsiva, oferecendo uma experiência ainda mais integrada aos dispositivos móveis. A mobilidade de um aplicativo permitirá que os alunos acessem os conteúdos e interajam com a plataforma de maneira mais ágil e acessível, aproveitando funcionalidades como notificações push e maior personalização. A proposta de um aplicativo mobile nativo representa uma evolução natural do projeto, ampliando sua acessibilidade e integração com o cotidiano dos estudantes. Essa expansão poderá facilitar o acompanhamento em tempo real por parte dos educadores, além de oferecer novas possibilidades de personalização e engajamento.

Referências

- ALVES, L., & Lopes, D. (Eds.). (2024). Educação e plataformas digitais: Popularizando saberes, potencialidades e controvérsias. EDUFBA;
- Atlassian. (s.d.). O que é Trello?. Trello. Disponível em: <https://trello.com/guide>
- BECK, K., & Andres, C. (2005). Extreme programming explained: Embrace change (2nd ed.). Addison-Wesley Professional.
- CASAGRANDE, A. L., Maieski, A., & Alonso, K. M. (2023). Tecnologias digitais na educação pós-pandemia e educação híbrida: efeitos, lições e possibilidades. EmRede - Revista de Educação a Distância, 10. <https://doi.org/10.53627/remed.v10i.1099>

Microsoft. (s.d.-a). TypeScript for JavaScript Programmers. TypeScript. Disponível em: <https://www.typescriptlang.org/docs/handbook/typescript-from-scratch.html>

Microsoft. (s.d.-b). Why Visual Studio Code?. Visual Studio Code. Disponível em: <https://code.visualstudio.com/docs/editor/why-vscode>

MongoDB, Inc. (s.d.). Why MongoDB?. MongoDB. Disponível em: <https://www.mongodb.com/why-mongodb>

NestJS Documentation. (s.d.). Introduction. NestJS - A progressive Node.js framework. Disponível em: <https://docs.nestjs.com/>

Node.js Foundation. (s.d.). About Node.js. Nodejs.org. Disponível em: <https://nodejs.org/en/about/>

PRESSMAN, R. S. (2016). Engenharia de software: Uma abordagem profissional (8a ed.). AMGH Editora;

RabbitMQ. (s.d.). What is RabbitMQ?. RabbitMQ. Disponível em: <https://www.rabbitmq.com/>

React Documentation. (s.d.). React: A JavaScript library for building user interfaces. React.dev. Disponível em: <https://react.dev/>

SANCINETTI, G. P., & Xavier, A. R. C. (2023). Gamificação como Metodologia Ativa de Avaliação: Relato de uma Experiência no Ensino Superior. Revista de Graduação USP, 7(1), 58–67. <https://doi.org/10.11606/issn.2525-376X.gradmais.2023.216350>;

SHAHIN, M., Babar, M. A., & Zhu, L. (2017). Continuous integration, delivery and deployment: A systematic review on approaches, tools, challenges and practices. IEEE Access, 5, 3909-3943. <https://doi.org/10.1109/ACCESS.2017.2685629>;

SILVA, C. L. da, Santos, C. da S. M. dos, Silva, L. M. O. da, Sousa, L. H. de, Gurgel, M. R. de F., Gurgel, R. F., Castro, R. C. F. G. de, Freire, T. G. de O., & Cavalcante, L. D. (2024). GAMIFICAÇÃO NA EDUCAÇÃO: BENEFÍCIOS, DESAFIOS E INOVAÇÕES TECNOLÓGICAS. Revista FT, 28(139);

SILVA, J. B. da, Sales, G. L., & Castro, J. B. de. (2019). Gamificação como estratégia de aprendizagem ativa no ensino de Física. Revista Brasileira de Ensino de Física, 41(4), e20180309. <https://doi.org/10.1590/1806-9126-RBEF-2018-0309>;

Vercel. (s.d.). Rendering: Server-Side Rendering (SSR). Next.js by Vercel. Disponível em: <https://nextjs.org/docs/pages/building-your-application/rendering/server-side-rendering>