

Gamificação na educação:

Matheus Arnaud Macambira Guedes



CENTRO DE INFORMÁTICA
UNIVERSIDADE FEDERAL DA PARAÍBA

João Pessoa, 2024

Matheus Arnaud Macambira Guedes

Gamificação na educação

Monografia apresentada ao curso
Engenharia da Computação
do Centro de Informática, da Universidade Federal da Paraíba,
como requisito para a obtenção do grau de Bacharel

Orientador: Alisson Vasconcelos De Brito

Maio de 2024

Catalogação na publicação
Seção de Catalogação e Classificação

G924g Guedes, Matheus Arnaud Macambira.
Gamificação na educação / Matheus Arnaud Macambira
Guedes. - João Pessoa, 2024.
43 f. : il.

Orientação: Alisson Brito.
TCC (Graduação) - UFPB/CI.

1. Gamificação. 2. Educação. 3. Aplicativo. 4.
Aprendizagem. 5. Engajamento. I. Brito, Alisson. II.
Título.

UFPB/CI

CDU 004.9(37)



CENTRO DE INFORMÁTICA
UNIVERSIDADE FEDERAL DA PARAÍBA

Trabalho de Conclusão de Curso de Engenharia da Computação intitulado ***Gamificação na educação*** de autoria de Matheus Arnaud Macambira Guedes, aprovada pela banca examinadora constituída pelos seguintes professores:

Prof. Dr. Alisson de Vasconcelos Brito
Universidade Federal da Paraíba

Altair Jussadir da Silva Pinto
Universidade Federal da Paraíba

Prof. Maelso Pacheco
Universidade Federal da Paraíba

João Pessoa, 20 de maio de 2024

AGRADECIMENTOS

Gostaria de expressar minha sincera gratidão a todos que contribuíram para a realização deste trabalho.

Em primeiro lugar, agradeço a Deus e a Nossa Senhora pela orientação e proteção fundamentais ao longo desta jornada. Sua presença constante me proporcionou a serenidade necessária para enfrentar os desafios e tomar as decisões corretas.

Agradeço também à minha noiva, Gabriela, por seu apoio inabalável e por acreditar em mim nos momentos em que até mesmo eu duvidava de mim mesmo.

Não posso deixar de expressar minha profunda gratidão ao meu pai, Junior, e à minha mãe, Flavia, pelo apoio incondicional e pelos sacrifícios que fizeram para me ajudar a alcançar meus objetivos. Suas palavras de encorajamento e amor foram essenciais para minha jornada.

Por fim, mas não menos importante, meu sincero agradecimento ao professor Alisson, cuja orientação e apoio foram fundamentais para o desenvolvimento deste trabalho. E a todos os meus amigos, que tornaram esta jornada a mais engraçada e divertida de todas, saibam que vocês têm um lugar especial em meu coração e em minhas memórias.

RESUMO

Este trabalho apresenta uma revisão de literatura sobre o uso da gamificação na educação, considerando o crescente interesse por essa abordagem, especialmente impulsionado pelo aumento do mercado de jogos, notavelmente durante a pandemia do COVID-19. Diante desse contexto, foi desenvolvido um aplicativo como proposta de utilização da gamificação no ambiente educacional, com o objetivo de promover a aprendizagem sem recorrer a ranqueamentos. A metodologia incluiu uma revisão detalhada de artigos acadêmicos e produtos disponíveis no mercado. O aplicativo resultante representa uma solução completa para a integração da gamificação no ensino, oferecendo uma alternativa que se baseia em elementos de premiação e motivação sem depender de sistemas de ranqueamentos.

Palavras-chave: <gamificação>, <educação>, <aplicativo>, <aprendizagem>, <engajamento>.

ABSTRACT

This work presents a literature review on the use of gamification in education, considering the growing interest in this approach, particularly driven by the increasing gaming market, notably during the COVID-19 pandemic. In this context, an application was developed as a proposal for using gamification in the educational environment, aiming to promote learning without resorting to rankings. The methodology included a detailed review of academic articles and products available in the market. The resulting application represents a comprehensive solution for integrating gamification into teaching, offering an alternative based on reward and motivation elements without relying on ranking systems.

Keywords: gamification, education, application, learning, engagement.

LISTA DE FIGURAS

1	Fluxo do Professor	15
2	Exemplo de desenvolvimento em trunk based	18
3	Visao geral do trello	20
4	Modelagem de dados	27
5	Fluxo de usuário para escola	29
6	Tela de login	30
7	<i>Dashboard</i> de usuário escola	31
8	Formulário de Cadastro de Aluno	31
9	Formulário de Cadastro de Professor	32
10	Formulário de Cadastro de Prêmios	33
11	Ver estudantes	33
12	Fluxo do Professor	34
13	<i>Dashboard</i> do Professor	35
14	Cadastro de Atividade	36
15	Visão do fluxo de atividades	37
16	Ver estudantes	37
17	Fluxo do Aluno	38
18	Visão da tela inicial de aluno	39
19	Ver atividades abertas	39
20	Visão da tela de presentes	40

LISTA DE TABELAS

1	Analise de elemento por efeito negativo	15
2	Comparação entre métodos ágeis	19

LISTA DE ABREVIATURAS

SIGLA – NOME COMPLETO

LUMO – Laboratório de computação Móvel e Ubíqua

UbiComp – Computação Ubíqua

API - Interface de programação de aplicativos

SUMÁRIO

1	INTRODUÇÃO	13
1.1	Definição do Problema	13
1.1.1	Objetivo geral	13
1.1.2	Objetivos específicos	13
2	Referencial Teórico	14
2.1	Gamificação	14
2.1.1	Pontos Negativos	14
2.1.2	Pontos Positivos	16
2.1.3	Considerações	16
2.2	Interface de programação de aplicativos	17
2.3	Gerenciamento de código fonte	17
2.4	Banco de Dados	18
2.5	Modelo de Dados	18
2.6	Metodologia ágil	19
3	Trabalhos Relacionados	22
3.1	kahoot	22
3.2	Gamificação além de emblemas	22
3.3	Swift Playground	23
3.4	Aprendizado de introdução a programação através de jogos	23
3.5	Comparativo	24
4	METODOLOGIA	25
4.1	Gamificação sem ranking	25
4.2	Levantamento de requisitos	25
4.3	Modelagem de dados	26
4.4	Proposta de solução	27
5	APRESENTAÇÃO E ANÁLISE DOS RESULTADOS	28

5.1	Desenvolvimento do <i>back-end</i>	28
5.2	Desenvolvimento <i>front-end</i>	28
5.2.1	Fluxo de Escola	29
5.2.2	Fluxo do Professor	33
5.2.3	Fluxo de Aluno	38
6	CONCLUSÕES E TRABALHOS FUTUROS	41
6.1	Dificuldades e Melhorias	41
6.2	Trabalhos Futuros	41
	REFERÊNCIA	41

1 INTRODUÇÃO

1.1 Definição do Problema

A conjuntura educacional, acelerada pela pandemia de Covid-19, tem presenciado um aumento expressivo no emprego de tecnologias, destacando-se a gamificação como uma abordagem comum para engajar e recompensar alunos durante suas atividades acadêmicas. Contudo, a implementação de sistemas de rankings nesse contexto levanta preocupações substanciais conforme mostrado por (ALMEIDA et al., 2023). Diante desse cenário, emerge a necessidade premente de reavaliar estratégias gamificadas, buscando não apenas a promoção da participação, mas também a construção de um ambiente educacional inclusivo e motivador, onde todos os alunos se sintam encorajados a contribuir ativamente para o processo de aprendizado.

1.1.1 Objetivo geral

Este trabalho propõe criar um aplicativo de gamificação sem o uso de sistemas de ranqueamento mas com um sistema de recompensa que motive o estudante em escolas de educação de ensino fundamental I e fundamental II a cumprir os objetivos e assim buscar melhorar o processo educacional.

1.1.2 Objetivos específicos

Abaixo serão listados os objetivos específicos para resolução do trabalho:

- Revisar as práticas de desenvolvimento de software do mercado
- Construir um sistema back-end capaz de receber qualquer sistema que queira usar o estilo de gamificação proposto
- Construir um aplicativo para utilizar o back-end desenvolvido
- Propor um sistema de gamificação sem o uso de rankeamentos
- Fazer uma revisão da literatura existente sobre o assunto

2 Referencial Teórico

Nesta seção são apresentados conceitos gerais para o entendimento do trabalho proposto. O capítulo é dividido em: Gameficação, *Apis*, Interface de programação de aplicativos, Gerenciamento de código fonte, Banco de dados, Modelagem de dados e Metodologia ágil.

2.1 Gamificação

O termo "gamificação", como discutido por (BOTRA; RERSELMAN; FORD, 2014) e (DETERDING et al., 2011), teve sua origem na indústria de mídia digital. O primeiro registro oficial remonta a 2008, quando Brett Tahil o definiu em um post de blog como "a aplicação de mecânicas de jogos em contextos da web". No entanto, foi apenas por volta de 2010 que o termo começou a ganhar popularidade. Mesmo dentro da própria indústria, surgiram controvérsias a respeito do seu uso, incluindo algumas opiniões negativas que argumentam que a definição original já não é mais adequada, uma vez que o processo de gamificação evoluiu para além do simples engajamento.

De acordo com (DETERDING et al., 2011), a gamificação é definida como o "uso de elementos de design de jogos em ambientes não relacionados a jogos", o que implica a incorporação de características típicas de jogos em contextos que normalmente não seriam considerados jogáveis, com o objetivo de tornar esses ambientes mais agradáveis e estimulantes para os usuários reais. As seções abaixo buscam abordar os pontos negativos e positivos dessa abordagem.

2.1.1 Pontos Negativos

A abordagem de gamificação tem sofrido diversas críticas ao longo dos anos, Ian Bogost que é um premiado autor, professor e *game design*, por exemplo, critica em (BOGOST, 2015) o uso de gamificação e a vê apenas como uma estratégia criada por *marketing* visando explorar uma oportunidade, efeitos negativos também foram evidenciados por (ALMEIDA et al., 2023), onde foi feito um mapeamento das principais críticas ao método de gamificação utilizado, e por (ZAINUDDIN et al., 2020), que encontrou evidências de que o uso de elementos como *badges*, pontos, troféus, avatares e presentes falhou em aumentar o valor percebido do processo de aprendizado. Pesquisas recentes destacaram ainda que a atividade gamificada não deve ser obrigatória, pois corre o risco de perder seu valor de entretenimento e pode levar ao colapso do ambiente de trabalho (HAMMEDI et al., 2021).

Foram feitas diversas revisões de literatura sobre o tema, em uma delas (ALMEIDA et al., 2023) fez um gráfico mostrado na figura 1 onde mostra os efeitos negativos causados

ao usuário mencionado por pelo menos dois artigos e o número de vezes que esses artigos referenciaram esses efeitos negativos, onde é possível observar que ranqueamentos é um dos líderes em efeitos negativos.

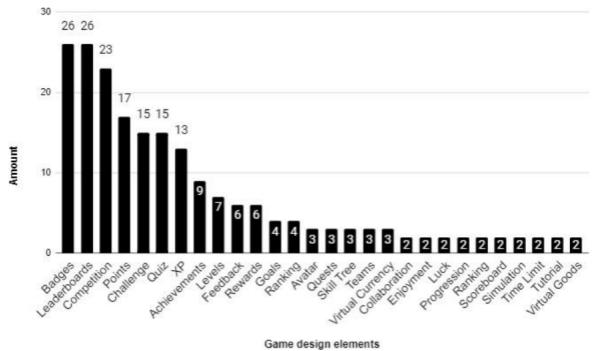


Figura 1: Fluxo do Professor

Fonte: Grafico retirado de (ALMEIDA et al., 2023)

Como é observável, a escolha da abordagem é tão importante quanto a execução em si. A tabela 1 mostra algumas estratégias de gamificação e os efeitos negativos associados a elas, mapeados por (ALMEIDA et al., 2023).

Elemento	Efeito negativo
Achievements	Irrelevância, Piora de performance
Badges	Bugs, Irrelevância, Falta de efeito, Falta de melhorias, Falta de entendimento, Perda de motivação, Piora de performance
Challenges	Não gostar de gamificação, Falta de entendimento, Piora de performance
Competições	Bugs, Trapaças, Irrelevância, Pressão pelo tempo, Piora de performance
Feedback	Não gostar de gamificação, Falta de efeito, Piora de performance
Objetivos	Falta de efeito
Rankeamentos	Bugs, Falta de motivação, Desencorajamento, Não gostar de ranqueamento, Não gostar de gamificação, Esforço humano extra, Jogando pelo sistema, Falta de melhorias, Falta de motivação, Falta de entendimento, Piora de performance
Níveis	Piora de performance
Pontos	Falta de efeito, Pressão pelo tempo, Piora de performance
Quizzes	Falta de efeito, Ansiedade
Rankings	Falta de efeito
Recompensas	Falta de motivação
Moedas Virtuais	Falta de efeito
Experiência	Não gostar de competição, Falta de motivação, Piora de performance

Tabela 1: Analise de elemento por efeito negativo

Fonte: Autoria própria baseada em tabela feita por (ALMEIDA et al., 2023)

2.1.2 Pontos Positivos

Apesar de apresentar potenciais efeitos negativos, a gamificação demonstra diversos usos adequados que resultam em impactos positivos em diferentes contextos educacionais. Por exemplo, estudos conduzidos por (PUTZ; HOFBAUER; TREIBLMAIER, 2020) revelaram que atividades gamificadas podem ter efeitos significativos na retenção de conhecimento entre estudantes de educação secundária e terciária. Ao analisar 617 alunos ao longo de dois anos, divididos em grupos que participaram de workshops gamificados e tradicionais, observou-se que os alunos expostos a atividades gamificadas apresentaram níveis mais altos de retenção de conhecimento a curto prazo e melhor desempenho em questionários realizados semanas após as atividades, independentemente de idade e gênero.

Além disso, uma revisão sistemática realizada por (KLOCK et al., 2018) também constatou resultados predominantemente positivos do uso da gamificação na educação. Os estudos mapeados destacaram não apenas melhorias de desempenho, mas também um aumento significativo no engajamento dos estudantes em sala de aula. Esta constatação ressalta a eficácia da gamificação como uma ferramenta motivacional e de aprendizagem.

Ampliando o escopo para uma perspectiva mais abrangente, (JOHNSON et al., 2016) examinou o uso da gamificação na promoção do bem-estar, especialmente na área da saúde. Sua revisão de literatura revelou que a maioria dos estudos analisados relatava efeitos positivos do emprego da gamificação. Essas descobertas reforçam a versatilidade e os benefícios potenciais da gamificação em diversos campos, desde a educação até a saúde, fornecendo *insights* valiosos para sua aplicação prática.

2.1.3 Considerações

Analizando o contexto, chegou-se à conclusão de que a melhor abordagem seria a implementação de um sistema de recompensas opcional na gamificação educacional. Nessa perspectiva, os alunos não seriam obrigados a utilizar a ferramenta, mas os prêmios oferecidos pela escola seriam atrativos o suficiente para incentivá-los a participar de forma espontânea. Cada escola teria autonomia para selecionar prêmios adequados à sua realidade, enquanto os professores teriam liberdade para criar atividades que valessem pontos. Uma vez que os alunos completassem as atividades, os professores poderiam atribuir pontos correspondentes, e os estudantes teriam a opção de resgatar esses pontos por prêmios ou acumulá-los até encontrarem algo que despertasse seu interesse.

Dessa forma, o presente projeto busca implementar um ambiente gamificado que envolve diversos elementos da escola, desde diretores até alunos, visando criar uma experiência mais envolvente e motivadora para todos os envolvidos no processo educacional.

Ao adotar uma abordagem baseada em recompensas opcionais, o objetivo é minimizar os potenciais efeitos negativos associados à gamificação, ao mesmo tempo em que se promove uma participação mais ativa e voluntária dos alunos.

2.2 Interface de programação de aplicativos

Também conhecida como API, é um conjunto de regras e protocolos que possibilitam a comunicação entre diferentes softwares. Ela define os métodos e estruturas de dados que os desenvolvedores podem utilizar para interagir com sistemas ou serviços externos. As APIs desempenham um papel essencial na integração de sistemas, permitindo que aplicativos compartilhem dados e funcionalidades de maneira eficiente.

Uma das arquiteturas mais amplamente utilizadas para a implementação de APIs é a arquitetura REST (Representational State Transfer). Proposta pela primeira vez por Roy Fielding em sua tese de doutorado (FIELDING, 2000), a arquitetura REST estabelece um conjunto de princípios e restrições que promovem a escalabilidade, a simplicidade e a eficiência na comunicação entre sistemas distribuídos.

2.3 Gerenciamento de código fonte

O gerenciamento de código-fonte é uma prática essencial adotada em todos os projetos de software. Ele desempenha um papel fundamental na organização e controle da manipulação do código-fonte, permitindo rastrear alterações, resolver conflitos e gerenciar as versões do projeto.

Uma ferramenta amplamente utilizada para esse propósito é o Git através da ferramenta Github¹, uma plataforma online de propriedade da Microsoft. O GitHub é construído sobre o sistema de controle de versão Git, um projeto de código aberto desenvolvido por Linus Torvalds em 2005. O Git foi criado para permitir que desenvolvedores colaborem de forma segura, flexível e controlada no mesmo código-fonte.

O Git oferece diversos fluxos de trabalho, sendo os mais comuns o Gitflow e o *Trunk-Based Development*. Na figura abaixo, apresentamos um exemplo de desenvolvimento baseado no Trunk-Based Development.

¹<https://github.com/>

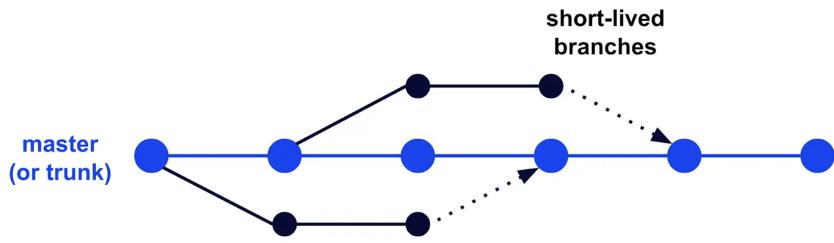


Figura 2: Exemplo de desenvolvimento em trunk based

Nesse modelo, o projeto possui apenas um ramo de longa duração, conhecido como "branch mestra" ou "trunk." Os desenvolvedores criam ramos de curta duração para trabalhar em novas funcionalidades ou correções de erros. Após a conclusão, eles mesclam essas alterações no ramo mestre, que geralmente contém as ferramentas de implantação. Essa estratégia foi escolhida para o desenvolvimento deste projeto.

2.4 Banco de Dados

Um banco de dados, no contexto da ciência da computação e da gestão da informação, é uma coleção organizada de dados inter-relacionados, geralmente armazenados de forma eletrônica, que são projetados para serem eficientes de consulta, recuperação e manipulação. Os dados em um banco de dados são estruturados de acordo com um modelo de dados específico, que define as tabelas, campos, relacionamentos e restrições que governam a representação dos dados. O objetivo fundamental de um banco de dados é fornecer um meio seguro e confiável para armazenar, gerenciar e acessar informações de maneira consistente e escalável, a fim de atender às necessidades de uma organização ou aplicação.

Dentre os mais diversos tipos de banco de dados foi escolhido o *PostGresql*² devido a necessidade de um banco relacional e por ser dentre os bancos relacionais a escolha mais comum do mercado.

2.5 Modelo de Dados

Segundo (KIM, 1990) um modelo de dados é uma organização lógica de objetos do mundo real, contendo suas restrições e relacionamentos entre os objetos, ou seja, ele descreve o conjunto de regras que governam como os dados são armazenados, acessados e manipulados. Os modelos de dados desempenham um papel crucial na definição da

²<https://www.postgresql.org/>

estrutura de um banco de dados, fornecendo um meio sistemático para representar informações de maneira coerente. Existem diversos tipos de modelos de dados, incluindo modelos conceituais, modelos lógicos e modelos físicos, cada um com um foco diferente na representação dos dados em diferentes níveis de abstração. Modelos de dados desempenham um papel essencial no projeto e na implementação de sistemas de gerenciamento de bancos de dados, ajudando a garantir a integridade dos dados e a eficiência das operações de consulta e atualização.

Dentre as diversas ferramentas usadas para a modelagem de dados, foi escolhida o *DbDiagram*³, devido ao fato que o mesmo gera uma visualização muito parecida ao que será implementado no banco de dados, gerando assim uma visualização maior e o adiantamento de parte do processo.

2.6 Metodologia ágil

Kent Beck, juntamente com os coautores do Manifesto Ágil, estabeleceu uma definição fundamental para a metodologia ágil de desenvolvimento de software. Essa abordagem é orientada por princípios que destacam a adaptabilidade às mudanças nos requisitos, a entrega frequente de valor em ciclos curtos e a manutenção da simplicidade, concentrando-se no que é essencial, de acordo com as necessidades do cliente.

Dentro desse contexto delineado por Beck e seus colegas, duas metodologias de gestão de projetos ganharam notoriedade no mercado: o Scrum e o Kanban. A tabela a seguir proporciona uma compreensão clara das diferenças entre esses dois métodos de desenvolvimento.

Tabela 2: Comparação entre métodos ágeis

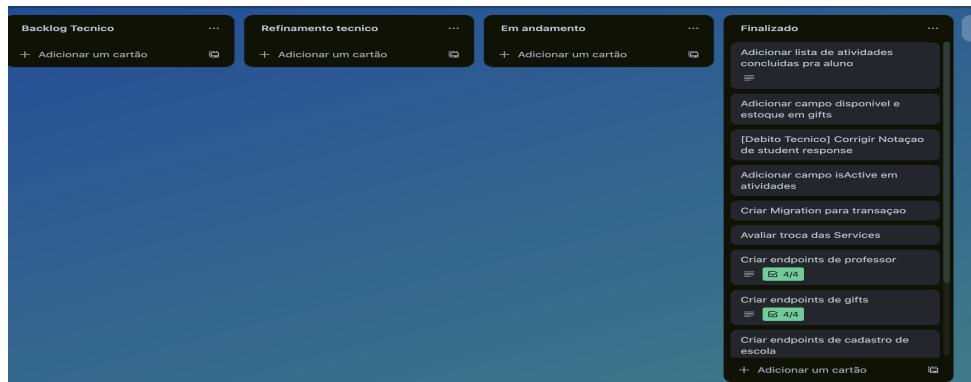
	Scrum	Kanban
Origem	Desenvolvimento de software	Fabricação lean
Ideologia	Aprenda por meio de experiência, auto-organização e priorização e reflita sobre os ganhos e as perdas para sempre melhorar.	Use recursos visuais para melhorar o trabalho em andamento
Ritmo	Sprints regulares, de duração fixa (ex.: duas semanas)	Fluxo contínuo
Continua na próxima página		

³<https://dbdiagram.io/d>

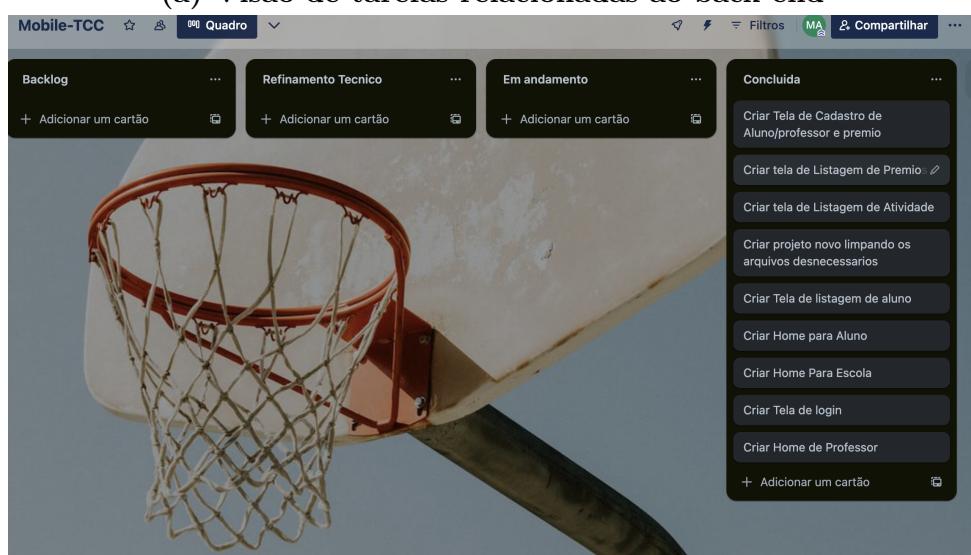
Tabela 2 – Continuação da página anterior

	Scrum	Kanban
Práticas	Planejamento de sprint, sprint, scrum diário, revisão de sprint, retrospectiva do sprint	Visualize o fluxo do trabalho, limite o trabalho em andamento, gerencie e fluxo e incorpore loops de feedback
Funções	Proprietário do produto, mestre scrum, equipe de desenvolvimento	Sem funções obrigatórias

A partir da comparação mostrada foram criados quadros de gestão conforme mostram as figuras abaixo.



(a) Visão de tarefas relacionadas ao back end



(b) Visão de tarefas relacionadas ao front end

Figura 3: Visão geral do trello

Fonte: Autoria Própria

Foram criados dois quadros diferentes, um para o *back-end* e outro para o *front-*

end, devido às diferenças significativas no escopo de desenvolvimento entre as duas áreas. Essa abordagem facilitou a gestão das tarefas que precisavam ser desenvolvidas. Sobre as colunas dos quadros, elas foram definidas da seguinte forma:

- *Backlog*: Representa as tarefas que precisam ser executadas.
- Refinamento Técnico: Refere-se a escrever no cartão as especificações necessárias para a tarefa e os critérios de aceitação
- Em Andamento: Indica que a tarefa está atualmente em desenvolvimento. Concluído: Sinaliza que a tarefa foi finalizada.
- Concluído: Sinaliza que a tarefa foi finalizada.

Essa organização e uso do *Trello* proporcionaram uma gestão eficiente das atividades de desenvolvimento no projeto.

3 Trabalhos Relacionados

3.1 kahoot

O *Kahoot* é um aplicativo norueguês de aprendizado baseado em jogos, desenvolvido em 2012 (KAHOOOT...,) por Morten Versvik, Johan Brand e Jamie Brooker, com base na tecnologia utilizada no mestrado de Morten Versvik na época. A plataforma permite a criação de sessões virtuais por qualquer usuário, onde um indivíduo é responsável por criar a sessão e elaborar perguntas, enquanto os demais participantes respondem às questões, acumulando pontos a cada resposta correta. Cada sessão culmina com a identificação de um vencedor, o participante com o maior número de pontos ao longo da atividade.

O *Kahoot* destaca-se por oferecer uma abordagem interativa e envolvente no processo de aprendizado, utilizando elementos lúdicos para tornar a experiência educacional mais dinâmica. Essa dinâmica de perguntas e respostas competitivas não apenas estimula a participação ativa dos alunos, mas também promove um ambiente de aprendizado colaborativo e divertido. A simplicidade na organização de sessões virtuais torna o *Kahoot* uma ferramenta versátil, aplicável em diversas situações educacionais para melhorar a retenção de informações e o engajamento dos participantes.

3.2 Gamificação além de emblemas

O projeto ICT for Rural Education Development (ICT4RED) faz parte do programa de pesquisa TECH4RED, iniciado pelo Departamento Sul-Africano de Ciência e Tecnologia (DST), em colaboração com o Departamento Sul-Africano de Educação Básica (DBE), o Departamento de Educação da Província do Cabo Oriental (ECDoE) e o Departamento Sul-Africano de Desenvolvimento Rural e Reforma Agrária (DRDLR). O TECH4RED visa contribuir para o aprimoramento da educação em áreas rurais por meio de inovações impulsionadas pela tecnologia.

Um dos componentes essenciais desse programa, conforme descrito em (BOTRA; RERSELMAN; FORD, 2014), é um aplicativo que concede distintivos ao usuário à medida que ele progride, representando suas conquistas educacionais. Essa abordagem visa estimular os alunos a se envolverem ainda mais no processo de aprendizado. O projeto foi implementado em salas de aula, recebendo feedbacks extremamente positivos. Contudo, é crucial ressaltar que a integração efetiva do aplicativo requer atividades de colaboração com os professores. Essa parceria é essencial para associar o uso do aplicativo ao contexto educacional específico de cada sala de aula, maximizando assim seu impacto e eficácia no processo de ensino-aprendizagem.

3.3 Swift Playground

O *Swift Playground* é um aplicativo desenvolvido pela *Apple* com o objetivo de promover o aprendizado da linguagem *Swift* por meio da gamificação. Este aplicativo foi incorporado por (CHENG; CHEN, 2021) como parte do programa de curso de pensamento computacional em Taiwan, destinado a estudantes de 12 anos. A abordagem do aplicativo é centrada na utilização de recursos visuais, em que vários cenários visuais são criados. No contexto de cada cenário, os alunos devem desenvolver códigos para resolver problemas enfrentados pelo personagem *Bytes*. Uma vez que o código é executado corretamente, o personagem realiza os movimentos programados.

O estudo revelou que tanto professores quanto alunos consideraram o aplicativo ideal para o ensino de pensamento computacional e raciocínio lógico. Notavelmente, oitenta e seis por cento dos estudantes consideraram as atividades de adicionar comandos fáceis, enquanto trinta e sete por cento acharam os exercícios que envolviam *loops* desafiadores. Esse *feedback* resultou na reformulação do calendário de aulas, visando torná-lo mais longo para proporcionar uma melhor adaptação dos estudantes.

3.4 Aprendizado de introdução a programação através de jogos

Um estudo conduzido por (MASKELIŪNAS et al., 2020) resultou no desenvolvimento de um jogo destinado a auxiliar alunos de graduação no aprendizado de programação, especialmente no contexto de algoritmos. O jogo utiliza um mapa dinâmico que representa uma cidade, onde cada comando é avaliado e visualizado em um *loop*, permitindo que um caminhão execute o último comando mostrado. Para os exercícios iniciais, são gerados mapas simples, enquanto para exercícios mais complexos, são gerados mapas randômicos, incentivando os estudantes a produzirem seus próprios algoritmos. Um aspecto fundamental desta abordagem é o reforço visual, no qual os estudantes podem observar os algoritmos que eles mesmos construíram sendo executados pelo caminhão, possibilitando uma compreensão mais profunda do processo e identificação de possíveis erros.

Este método de ensino combina elementos visuais com conceitos fundamentais de programação, proporcionando uma experiência de aprendizado envolvente e prática. Ao visualizarem diretamente os resultados de seus esforços, os alunos têm a oportunidade de corrigir erros e aprimorar suas habilidades de resolução de problemas de forma mais eficaz. Essa abordagem foi avaliada através da aplicação de testes antes e depois da disciplina em dois grupos: um de controle e outro que utilizou o jogo, onde o grupo que utilizou o jogo apresentou resultado melhor no teste pós disciplina.

3.5 Comparativo

O presente projeto compartilha algumas semelhanças com os trabalhos mencionados anteriormente. Assim como o *Kahoot*, visa o cumprimento de atividades, e, de maneira similar ao estudo conduzido por (BOTRA; RERSELMAN; FORD, 2014), utiliza um sistema de recompensas. No entanto, apresenta distinções fundamentais.

Enquanto o *Kahoot* concentra-se em sessões curtas e rápidas, ideal para uma abordagem mais pontual e específica, o presente estudo busca criar um aplicativo de visualização contínua. Essa escolha estratégica permite a observação do acúmulo de pontos ao longo de várias sessões, proporcionando aos participantes uma experiência de engajamento constante e progressivo, independentemente da duração específica de uma atividade.

Quando comparado ao *Swift Playground* e ao jogo desenvolvido por (MASKE-LIŪNAS et al., 2020), percebe-se uma clara diferença de abordagens. Enquanto ambos valorizam o estímulo visual, refletindo as ações do usuário na tela, o sistema proposto neste projeto foca na ideia de que a conclusão de atividades gera pontos que podem ser utilizados para adquirir recompensas.

Outro diferencial encontra-se na liberdade concedida aos professores neste projeto. Ao contrário do *Kahoot*, onde as atividades geralmente se limitam a questionários, o presente estudo permite que os educadores criem uma variedade de atividades que se adaptem melhor às suas metodologias de ensino específicas. Essa flexibilidade reconhece que diferentes disciplinas e abordagens pedagógicas demandam formatos diversos de engajamento.

Além disso, a distinção do projeto se acentua ao incorporar prêmios reais, proporcionando uma dimensão tangível ao processo de gamificação. Os prêmios, definidos pela escola, acrescentam uma camada adicional de motivação aos alunos, criando um ambiente onde a conquista não é apenas virtual, mas também reflete-se em recompensas palpáveis. Essa abordagem busca não apenas envolver os alunos nas atividades propostas, mas também estimulá-los através de incentivos tangíveis que reforçam positivamente o processo educacional.

4 METODOLOGIA

Nesse capítulo ocorrerá a discussão da metodologia escolhida para a resolução do problema em questão, essa foi dividida nas seguintes etapas: Levantamento de requisitos, modelagem de dados e proposta de solução.

4.1 Gamificação sem ranking

Apesar de ranking ser elemento efetivo na criação de sistemas de gamificação, como demonstrado por (CODISH; RAVID, 2014), e ser considerado um ingrediente-chave no design de jogos (REEVES; READ, 2009), (JIA et al., 2017) propõe uma análise considerando a personalidade do aluno. O fator extroversão está intrinsecamente ligado ao sucesso de um sistema de gamificação com ranking, destacando a importância de levar em conta as características individuais dos alunos ao implementar elementos de jogos. Então como a intenção é a implementação de um aplicativo que seja utilizado de forma geral, sem ter como reduzir o número de usuários ao ponto de sabermos a personalidade da maioria, foi proposta a estratégia de gamificação sem o uso de rankings.

4.2 Levantamento de requisitos

Requisitos de software são definidos por (SOMMERVILLE; SAWYER, 1997) como descrições de como um sistema a ser construído deve funcionar e quais os seus comportamentos esperados. Para desenvolvimento do /Pensar num nome para o app/ foram levantadas comparações com sistemas de gamificação já existente, bem como feitas hipóteses de funcionalidades que poderiam ser úteis aos diferentes tipos de usuário, chegando nos seguintes requisitos:

- Escolas devem poder cadastrar e/ou remover alunos
- Escolas devem poder cadastrar/remover professores
- Escolas devem poder cadastrar/remover brindes
- Alunos só podem ver a própria pontuação
- Alunos podem gastar os pontos em brindes
- Professores podem cadastrar atividades
- Professores podem ver a pontuação dos alunos
- Professores podem atualizar atribuir uma atividade a um aluno

- Brindes tem uma descrição e um custo em pontos associadas a eles.
- Brindes são criados pela escola
- Atividades precisam ter pontuação e descrição
- Atividades precisam estar associadas a um professor

4.3 Modelagem de dados

Com base nos requisitos definidos para o desenvolvimento do aplicativo, o próximo passo foi a modelagem de dados, um processo crucial para criar uma estrutura que atendesse de forma eficiente e abrangente às necessidades do sistema.

A modelagem de dados é uma representação abstrata e organizada das informações que o sistema irá gerenciar. Ela visa definir como os dados serão armazenados, acessados e relacionados entre si. Nesse contexto, a modelagem de dados para o aplicativo proposto envolveu a criação de entidades e seus relacionamentos, garantindo uma representação fiel e coerente dos requisitos levantados.

As entidades identificadas no processo de modelagem incluem:

- Escola: Responsável por cadastrar e remover alunos, professores e brindes. Também é responsável pela criação dos brindes.
- Aluno: Pode visualizar apenas sua pontuação e possui a capacidade de gastar os pontos em brindes.
- Professor: Encarregado de cadastrar atividades, visualizar a pontuação dos alunos e atribuir atividades a alunos.
- Brinde: Criado pela escola, cada brinde possui uma descrição e um custo em pontos associado a ele.
- Atividade: Cada atividade cadastrada pelos professores deve conter uma pontuação e uma descrição. Está associada a um professor.

Ao definir essas entidades e seus relacionamentos, busca-se estabelecer uma base sólida para o armazenamento e manipulação eficientes dos dados. Além disso, a modelagem considera a integridade dos dados e as relações necessárias para garantir o correto funcionamento do sistema, isso foi feito com auxílio do dbdiagram uma ferramenta que permite uma visualização do sistema modelado, com seus relacionamentos e sua cardinalidade conforme mostra a figura abaixo.

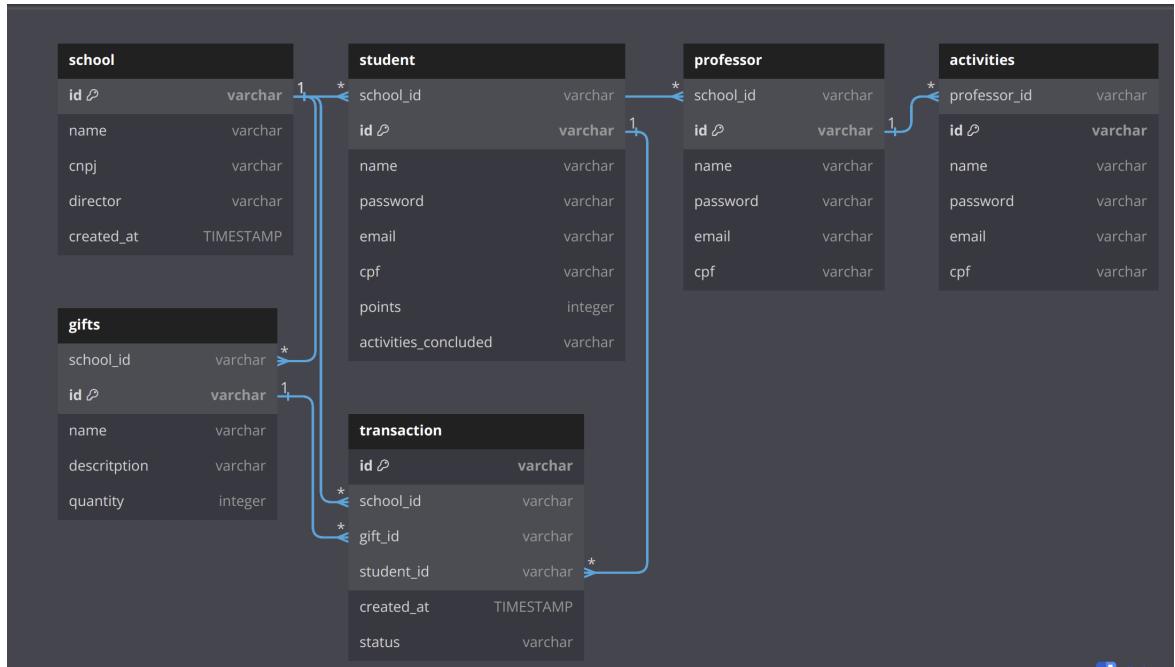


Figura 4: Modelagem de dados

Fonte: Autoria Própria

4.4 Proposta de solução

Com base nos requisitos levantados e na modelagem de dados realizada, foi desenvolvido um aplicativo completo, abrangendo tanto a parte de front-end quanto de back-end. Essa solução integrada visa atender de maneira eficaz e abrangente às demandas identificadas, proporcionando uma ferramenta funcional e coerente com as necessidades educacionais e de gamificação propostas.

5 APRESENTAÇÃO E ANÁLISE DOS RESULTADOS

5.1 Desenvolvimento do *back-end*

No processo de desenvolvimento do projeto, a primeira etapa consistiu na criação de um *back-end* robusto, capaz de atender a todos os requisitos inicialmente delineados. Para esta finalidade, optou-se pela linguagem de programação *Kotlin*. Essa escolha foi motivada pelo fato de que o *Kotlin* é um *framework* da linguagem Java e, portanto, oferece acesso ao renomado *Spring Boot*, amplamente reconhecido e utilizado no mercado para o desenvolvimento de sistemas web no lado do servidor.

Com essa base sólida estabelecida, a equipe de desenvolvimento concentrou-se inicialmente na criação das rotas que permitem o cadastro dos três tipos de usuários do sistema. Em seguida, a atenção se voltou para a implementação de funcionalidades mais complexas, como a troca de pontos por brindes e o registro de atividades concluídas pelos alunos.

Para facilitar a implementação do *front-end* de maneira intuitiva, a biblioteca *Swagger* foi empregada para a documentação das rotas. O uso do *Swagger* se mostrou vantajoso, pois reúne em um único local informações essenciais, como detalhes de teste e a visualização dos parâmetros necessários para cada rota. Essa abordagem proporcionou uma experiência de desenvolvimento mais eficiente e coesa.

5.2 Desenvolvimento *front-end*

Dada a perspectiva inicial de que o sistema seria amplamente utilizado em ambientes escolares, onde o acesso a computadores muitas vezes é limitado, a decisão foi tomar o *front-end* da aplicação na forma de um aplicativo móvel. Essa escolha visava proporcionar maior facilidade de uso para o público em geral. O aplicativo foi desenvolvido utilizando a plataforma *Expo* e programado em *Typescript*.

A escolha do *Typescript* oferece diversas vantagens no desenvolvimento de aplicações *React-Native*. Ele ajuda a prevenir erros comuns durante o processo de desenvolvimento, oferecendo ferramentas avançadas para autocompletar código e detecção de erros. Após a instalação e configuração inicial do aplicativo, procedeu-se à definição dos temas de cores da aplicação, garantindo uma consistência visual em todo o projeto.

Com os temas da aplicação estabelecidos, o próximo passo envolveu a configuração das rotas da aplicação. Para esse propósito, optou-se pelo modelo de navegação em pilha, também conhecido como *Stack Navigation*, no qual as telas são sobrepostas sequencialmente à medida que o usuário avança na aplicação.

O uso de rotas possibilita a criação de diversas telas e componentes, em que cada

tela possui sua própria rota específica. O conceito de componentização do *React* foi aplicado, o que significa que elementos visuais compartilhados entre diversas telas foram componentizados para evitar a repetição de código. Exemplos de componentes incluem os botões exibidos na tela inicial de cada usuário e o cabeçalho, entre outros.

Para uma análise mais detalhada do fluxo dentro do aplicativo, examinaremos a experiência de uso sob a perspectiva de três tipos de usuários possíveis: Aluno, Escola e Professor.

Para o desenvolvimento deste aplicativo foi usado de base um projeto desenvolvido como parte do curso de *react-native* criada pela Rocketseat, foram reaproveitados alguns componentes e a paleta de cores.

5.2.1 Fluxo de Escola

Para primeiro explicar o fluxo de quando o usuário for um representante de escola, foi criado o diagrama abaixo que demonstra as possibilidades do usuário dentro do aplicativo.

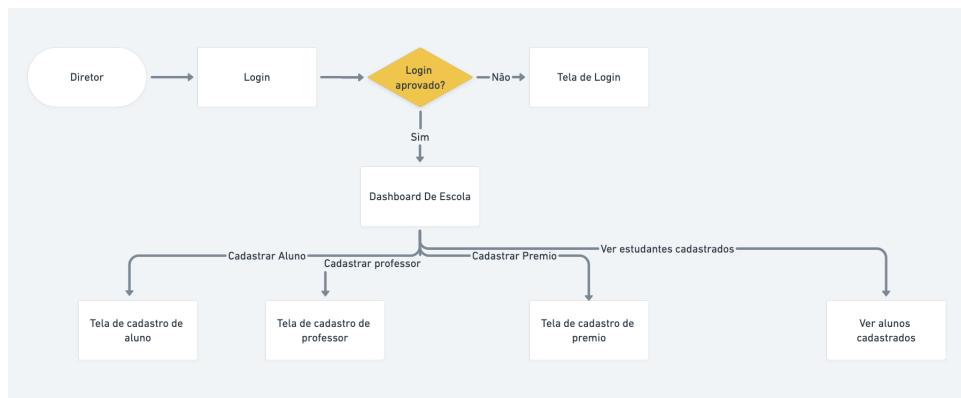


Figura 5: Fluxo de usuário para escola

Fonte: Autoria Própria

Conforme demonstrado no diagrama de escola na figura 5, quando um usuário do tipo escola acessa o sistema, ele tem acesso a um conjunto específico de funcionalidades, as quais serão detalhadas a seguir.

A primeira tela à qual o usuário tem acesso é a tela de login mostrada na figura 6. Nessa etapa, o usuário deve fornecer seu endereço de *e-mail* e senha. Ao clicar em 'Enviar', as informações são encaminhadas ao *back-end*, que é responsável por validar a autenticidade do usuário e verificar se ele corresponde ao tipo de usuário declarado.



Figura 6: Tela de login
Fonte: Autoria Própria

Com base na resposta recebida pelo aplicativo, existem duas possibilidades:

1. Caso alguma das informações fornecidas pelo usuário não corresponda aos dados armazenados no banco de dados, o usuário permanecerá na tela de login. Nesse cenário, uma mensagem é exibida informando que as informações fornecidas são inválidas, e o usuário é orientado a revisar os dados inseridos.
2. Se todas as informações estiverem corretas, o usuário será automaticamente direcionado para a próxima tela, onde terá acesso às funcionalidades exclusivas destinadas ao tipo de usuário 'escola'.

A tela subsequente, denominada '*dashboard*', disponibiliza ao usuário funcionalidades específicas associadas ao seu tipo. No caso de um usuário do tipo escola, serão apresentadas opções para cadastrar alunos, professores, prêmios e visualizar detalhes dos estudantes registrados alem de ja ter a visão da quantidade total de estudantes cadastrados pela escola. É relevante destacar que a construção dessa tela foi orientada pelo princípio de componentização do React, possibilitando que grande parte dela fosse reutilizada em diferentes tipos de usuários.



Figura 7: Dashboard de usuário escola
Fonte: Autoria Própria

Quando o usuário acessa a tela principal, suas ações no aplicativo dependem de sua intenção. Se desejar cadastrar um novo aluno, basta clicar no botão de cadastro correspondente, levando-o a um formulário personalizado, conforme ilustrado na Figura 7.

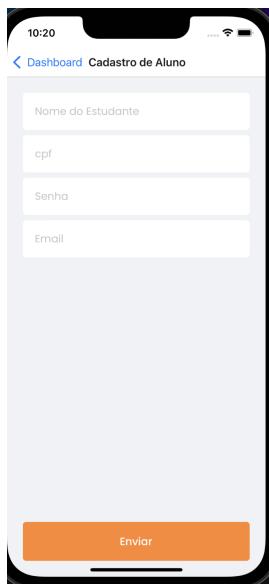


Figura 8: Formulário de Cadastro de Aluno
Fonte: Autoria Própria

Após preencher todas as informações do aluno, desde que estejam de acordo com os critérios de validação — lembrando que implementamos um validador para assegurar a aceitação apenas de dados válidos —, ao pressionar o botão 'Enviar', o aplicativo encaminhará esses dados ao *back-end*. Nesta etapa, é responsabilidade do *back-end* verificar a existência, ou não, de um estudante com o mesmo número de CPF. O *back-end* retornará um status indicando o sucesso ou erro no processo de criação do estudante. Com a con-

clusão bem-sucedida, o estudante terá a habilidade de acessar o sistema usando o *e-mail* e senha atribuídos, começando com um saldo inicial de 0 pontos.

Por outro lado, se o usuário desejar cadastrar um professor, será direcionado para o formulário correspondente a essa funcionalidade, conforme ilustrado na Figura 8.

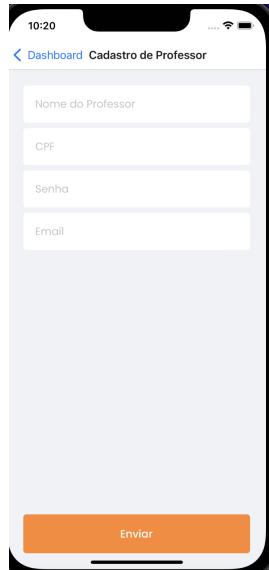


Figura 9: Formulário de Cadastro de Professor

Fonte: Autoria Própria

Apesar de as informações, a princípio, serem completamente semelhantes entre alunos e professores, optamos por criar formulários separados. Isso permite uma maior flexibilidade para futuros requisitos que não foram mapeados neste protótipo, tornando mais simples atender às necessidades específicas de cada tipo de usuário.

Além dos cadastros mencionados acima, os representantes da escola podem desejar cadastrar prêmios que os estudantes podem resgatar em troca de pontos, uma vez que isso é um elemento central do protótipo. Essa funcionalidade foi implementada de maneira simples, permitindo aos representantes cadastrar prêmios que melhor atendam à realidade da escola. As opções variam desde itens maiores, como kits de materiais escolares ou livros, até opções mais simples, como lanches da cantina da escola, como ilustrado na Figura ??.

Formulário de Cadastro de Prêmios

O formulário contém campos para Nome, Descrição e Pontos, com um botão 'Enviar' no final.

Figura 10: Formulário de Cadastro de Prêmios

Fonte: Autoria Própria

Ao invés de realizar um cadastro, o usuário pode querer simplesmente ver quem são os estudantes cadastrados, será nessa tela onde futuramente será implementada a funcionalidade de editar e deletar estudantes, mas pra esse protótipo foi definido que o central era conseguir listar e ver os estudantes cadastrados.

Dashboard do Professor StudentView

Matheus
Pontos: 0
596.483.620-84

Thiago
Pontos: 120
719.426.890-63

Figura 11: Ver estudantes

Fonte: Autoria Própria

5.2.2 Fluxo do Professor

Para uma compreensão mais clara das funcionalidades disponíveis para um professor dentro do sistema, apresentamos o fluxo ilustrado na Figura 12.

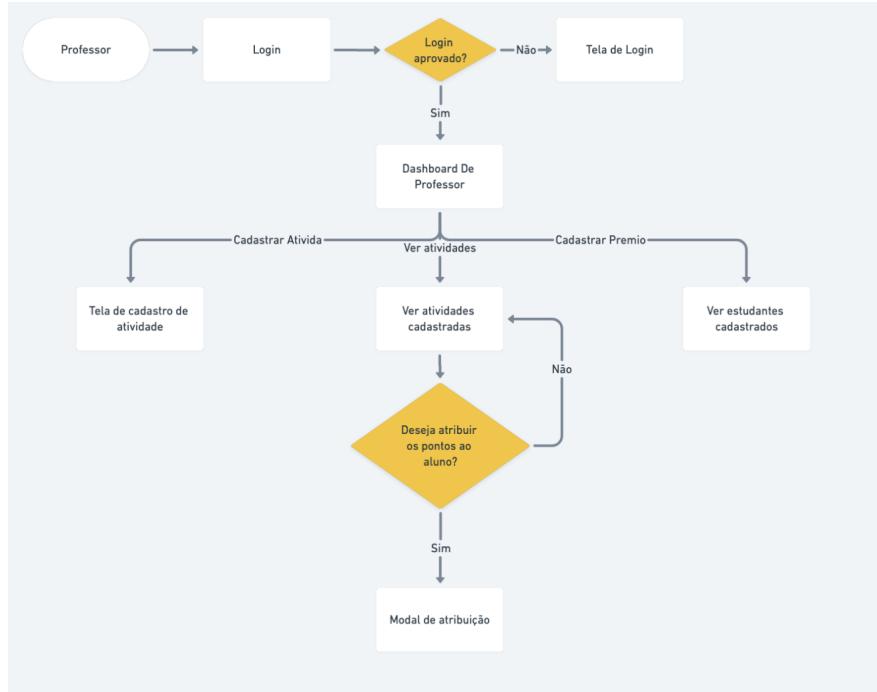


Figura 12: Fluxo do Professor

Fonte: Autoria Própria

O fluxo de um professor inicia da mesma maneira que o fluxo de uma escola, na tela de login. O professor inserirá seu *e-mail* e senha, e o *back-end* ficará encarregado de validar as informações fornecidas. Se os dados forem válidos, o professor será direcionado para a tela de *dashboard* com funcionalidades específicas para professores, conforme ilustra a Figura 13.



Figura 13: *Dashboard do Professor*

Fonte: Autoria Própria

O caminho que o professor seguirá dependerá de suas intenções. Se desejar cadastrar uma nova atividade, ele será encaminhado para a tela contendo um formulário semelhante ao apresentado no fluxo da escola, conforme demonstra a Figura 14.

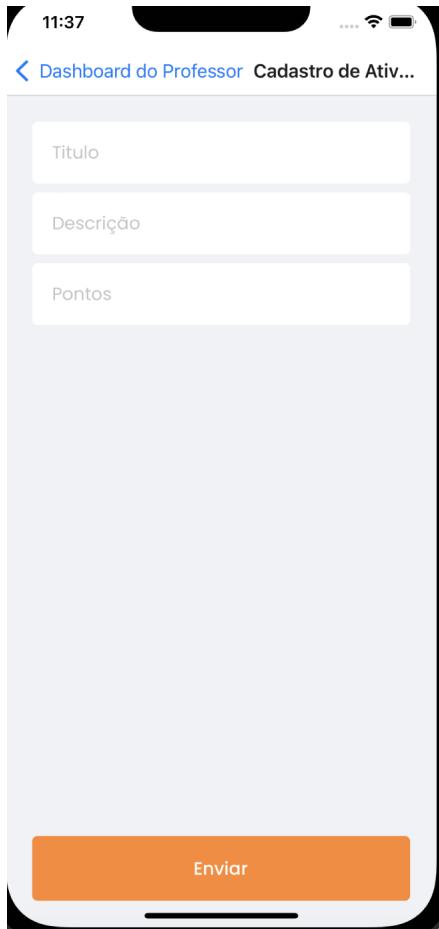


Figura 14: Cadastro de Atividade

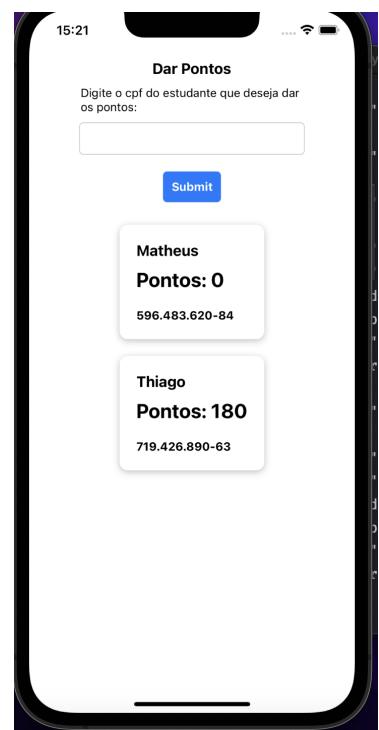
Fonte: Autoria Própria

Nesse formulário, poucas validações são necessárias, pois o professor tem a liberdade de definir uma ampla variedade de atividades e suas respectivas pontuações. É importante ressaltar que o professor pode até mesmo atribuir pontuações negativas, permitindo que crie atividades como "alunos que chegaram atrasados" e deduza pontos, se desejar. Isso proporciona máxima flexibilidade na condução das atividades.

Caso deseje visualizar as atividades cadastradas, o professor pode acessar a tela de 'Atividades Cadastradas', conforme demonstrado na Figura 15. Nessa tela, ele pode atribuir pontos às atividades de alunos que as tenham concluído, simplesmente clicando no botão 'Atribuir Pontos' no cartão que representa cada atividade. Um modal se abrirá, permitindo ao professor inserir o nome do aluno ao qual deseja atribuir os pontos da atividade.



(a) Listagem de atividades



(b) Escolher qual aluno receberá os pontos

Figura 15: Visão do fluxo de atividades

Fonte: Autoria Própria

Caso deseje o professor também consegue ver os estudantes cadastrados e futuramente também poderá editar o cadastro de estudantes como mostrado na figura 16

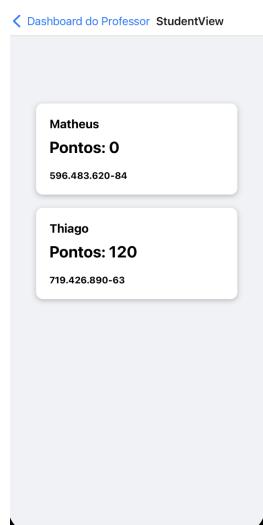


Figura 16: Ver estudantes

Fonte: Autoria Própria

5.2.3 Fluxo de Aluno

Para uma compreensão mais clara das funcionalidades disponíveis para um aluno dentro do sistema, apresentamos o fluxo ilustrado na Figura 17.

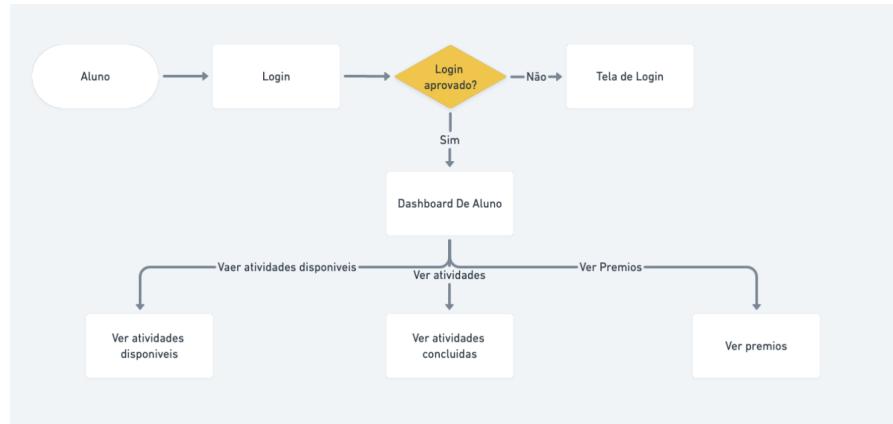
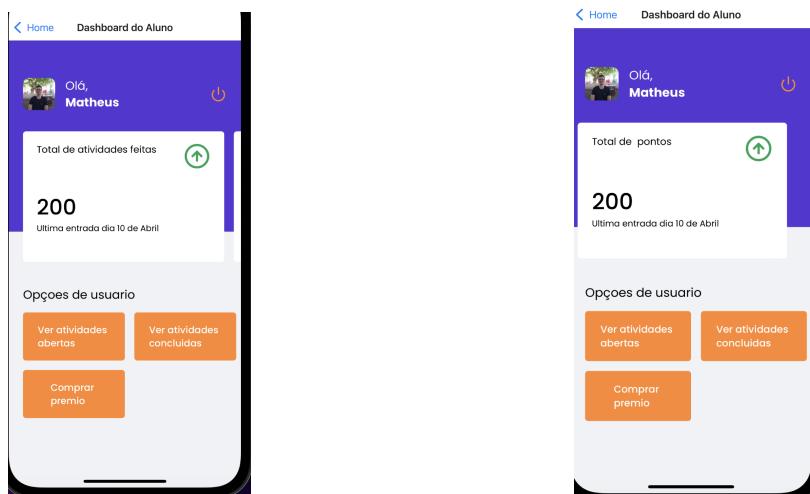


Figura 17: Fluxo do Aluno

Fonte: Autoria Própria

Assim como mencionado anteriormente, o fluxo do aluno também tem início na tela de login, onde ele insere seu *e-mail* e senha e envia as informações para o back-end. Uma vez que as informações são enviadas com sucesso, ele é direcionado para o *dashboard* do aluno, onde é apresentado às suas opções.

Na tela do *dashboard*, existem algumas diferenças entre o estudante e os demais usuários. Primeiro, ele possui dois cartões de dados, um mostrando a quantidade de atividades que ele concluiu e o outro mostrando quantos pontos ele possui, como ilustrado na figura 18.



(a) Fluxo do Aluno primeiro cartão

(b) Fluxo do Aluno com cartão de pontos

Figura 18: Visão da tela inicial de aluno

Fonte: Autoria Própria

A segunda diferença é que o aluno não preenche nenhum formulário. Ele pode apenas ver as atividades que ainda estão abertas por um professor, ver as atividades que ele já concluiu e ver os prêmios disponíveis.

Se ele desejar abrir as atividades ainda em andamento, será direcionado para uma tela que mostrará a ele a listagem de atividades ainda abertas.



Figura 19: Ver atividades abertas

Fonte: Autoria Própria

Se ele desejar comprar um prêmio, será direcionado para a tela com a lista de prêmios onde, ao clicar em comprar, receberá o prêmio desejado como mostrado pela figura abaixo.

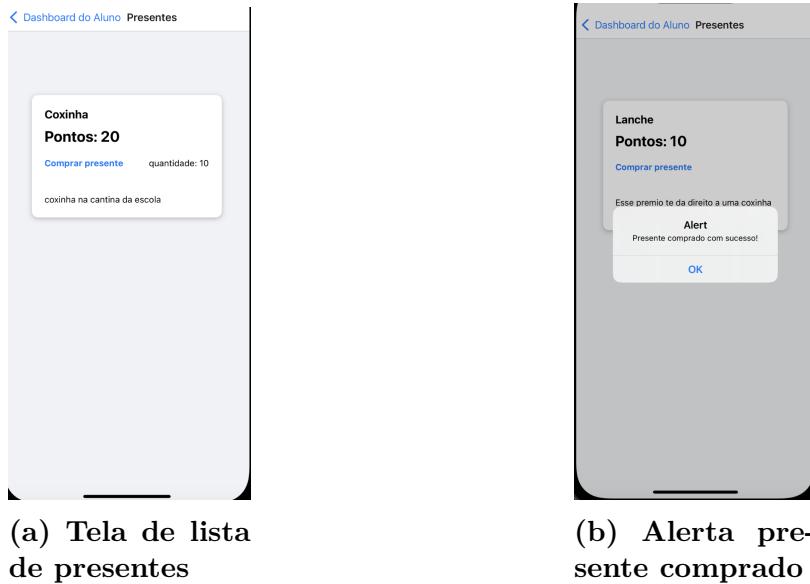


Figura 20: Visão da tela de presentes

Fonte: Autoria Própria

6 CONCLUSÕES E TRABALHOS FUTUROS

O sistema desenvolvido neste trabalho destaca-se como um exemplo claro de como a integração da tecnologia e da gamificação pode desempenhar um papel crucial na transformação da educação. Ao proporcionar uma ferramenta que não apenas auxilia os professores e as escolas no cumprimento de seus objetivos educacionais, mas também motiva os alunos a progredirem e se engajarem ativamente no processo de aprendizado.

A sinergia entre a tecnologia e a gamificação revela-se na forma como os alunos são incentivados não apenas a participar das atividades propostas, mas também a buscar constantemente o aprimoramento e a excelência. Ao criar um possível ambiente educacional dinâmico e estimulante, o sistema oferece não apenas uma solução prática para os desafios educacionais atuais, mas também busca ser um incentivo para o progresso acadêmico dos estudantes.

6.1 Dificuldades e Melhorias

Devido ao caráter experimental do projeto, uma dificuldade encontrada foi a falta de experiência no quesito design e acessibilidade, bem como no desenvolvimento de padrões de cores autorais para o projeto. Além disso, a participação de usuários professores para validação de requisitos também se apresentou como um ponto de melhoria uma vez que o projeto foi todo baseado na revisão da literatura encontrada sobre o assunto.

6.2 Trabalhos Futuros

Os próximos passos para o projeto incluem a contratação de um designer para o desenvolvimento de cores e componentes autorais, validação de acessibilidade e criação de um ambiente de teste e produção no backend. A melhoria da arquitetura do aplicativo, alinhando-a com as melhores práticas possíveis no desenvolvimento mobile, também é essencial. Além disso, a ampliação do uso em salas de aula será fundamental para validar a escalabilidade do aplicativo em diferentes contextos educacionais.

Além disso, propõe-se a adição de um sistema de emblemas em conjunto com um sistema de níveis, juntamente com avatares personalizáveis, visando atrair principalmente estudantes mais jovens. Esses elementos adicionais podem enriquecer ainda mais a experiência dos usuários, tornando o ambiente educacional mais atraente e motivador, bem como uma adição a atividades que o professor pode criar, incluindo *quizzes* e avaliações.

REFERÊNCIAS

- ALMEIDA, C.; KALINOWSKI, M.; UCHÔA, A.; FEIJÓ, B. Negative effects of gamification in education software: Systematic mapping and practitioner perceptions. **Information and Software Technology**, Elsevier, v. 156, p. 107142, 2023.
- BOGOST, I. Why Gamification is Bullshit. In: **The Gameful World: Approaches, Issues, Applications**. The MIT Press, 2015. ISBN 9780262325714. Disponível em: <<https://doi.org/10.7551/mitpress/9788.003.0005>>.
- BOTRA, A.; RERSELMAN, M.; FORD, M. Gamification beyond badges. In: **IEEE. 2014 IST-Africa Conference Proceedings**. [S.l.], 2014. p. 1–10.
- CHENG, G.-M.; CHEN, C.-P. Processing analysis of swift playgrounds in a children's computational thinking course to learn programming. **Computers**, MDPI, v. 10, n. 5, p. 68, 2021.
- CODISH, D.; RAVID, G. Personality based gamification-educational gamification for extroverts and introverts. In: **THE OPEN UNIVERSITY OF ISRAEL RA'ANANA. Proceedings of the 9th CHAIS Conference for the Study of Innovation and Learning Technologies: Learning in the Technological Era**. [S.l.], 2014. v. 1, p. 36–44.
- DETERDING, S.; DIXON, D.; KHALED, R.; NACKE, L. From game design elements to gamefulness: defining "gamification". In: **Proceedings of the 15th international academic MindTrek conference: Envisioning future media environments**. [S.l.: s.n.], 2011. p. 9–15.
- FIELDING, R. T. **Architectural styles and the design of network-based software architectures**. [S.l.]: University of California, Irvine, 2000.
- HAMMEDI, W.; LECLERCQ, T.; PONCIN, I.; ALKIRE, L. Uncovering the dark side of gamification at work: Impacts on engagement and well-being. **Journal of Business Research**, Elsevier, v. 122, p. 256–269, 2021.
- JIA, Y.; LIU, Y.; YU, X.; VOIDA, S. Designing leaderboards for gamification: Perceived differences based on user ranking, application domain, and personality traits. In: **Proceedings of the 2017 CHI conference on human factors in computing systems**. [S.l.: s.n.], 2017. p. 1949–1960.
- JOHNSON, D.; DETERDING, S.; KUHN, K.-A.; STANEVA, A.; STOYANOV, S.; HIDES, L. Gamification for health and wellbeing: A systematic review of the literature. **Internet interventions**, Elsevier, v. 6, p. 89–106, 2016.
- KAHoot about us. <<https://kahoot.com/company/>>. Accessed: 2023-11-09.
- KIM, W. Object-oriented databases: Definition and research directions. **IEEE Transactions on Knowledge & Data Engineering**, IEEE Computer Society, v. 2, n. 03, p. 327–341, 1990.

KLOCK, A. C. T.; OGAWA, A. N.; GASPARINI, I.; PIMENTA, M. S. Does gamification matter? a systematic mapping about the evaluation of gamification in educational environments. In: **Proceedings of the 33rd Annual ACM Symposium on Applied Computing**. [S.l.: s.n.], 2018. p. 2006–2012.

MASKELIŪNAS, R.; KULIKAJEVAS, A.; BLAŽAUSKAS, T.; DAMAŠEVIČIUS, R.; SWACHA, J. An interactive serious mobile game for supporting the learning of programming in javascript in the context of eco-friendly city management. **Computers**, MDPI, v. 9, n. 4, p. 102, 2020.

PUTZ, L.-M.; HOFBAUER, F.; TREIBLMAIER, H. Can gamification help to improve education? findings from a longitudinal study. **Computers in Human Behavior**, Elsevier, v. 110, p. 106392, 2020.

REEVES, B.; READ, J. L. **Total engagement: How games and virtual worlds are changing the way people work and businesses compete**. [S.l.]: Harvard Business Press, 2009.

SOMMERVILLE, I.; SAWYER, P. **Requirements engineering: a good practice guide**. [S.l.]: John Wiley & Sons, Inc., 1997.

ZAINUDDIN, Z.; CHU, S. K. W.; SHUJAHAT, M.; PERERA, C. J. The impact of gamification on learning and instruction: A systematic review of empirical evidence. **Educational research review**, Elsevier, v. 30, p. 100326, 2020.