



**Fundação Educacional do Município de Assis
Instituto Municipal de Ensino Superior de Assis
Campus "José Santilli Sobrinho"**

RENAN POLLO BENELLI

TECHLINGO: UM SISTEMA DE GESTÃO DE APRENDIZAGEM

**Assis/SP
2024**



Fundação Educacional do Município de Assis
Instituto Municipal de Ensino Superior de Assis
Campus "José Santilli Sobrinho"

RENAN POLLO BENELLI

TECHLINGO: UM SISTEMA DE GESTÃO DE APRENDIZAGEM

Projeto de pesquisa apresentado ao curso de Análise e Desenvolvimento de Sistemas do Instituto Municipal de Ensino Superior de Assis – IMESA e a Fundação Educacional do Município de Assis – FEMA, como requisito parcial à obtenção do Certificado de Conclusão.

Orientando(a): Renan Pollo Benelli.
Orientador(a): Luiz Carlos Begosso.

Assis/SP
2024

TECHLINGO: UM SISTEMA DE GESTÃO DE APRENDIZAGEM.

RENAN POLLO BENELLI

Trabalho de Conclusão de Curso apresentado ao Instituto Municipal de Ensino Superior de Assis, como requisito do Curso de Graduação, avaliado pela seguinte comissão examinadora:

Orientador: _____ Luiz Carlos Begosso

Examinador: _____ Luiz Ricardo Begosso

Assis/SP
2024

Benelli, Renan Pollo

B465t TechLingo: um sistema de gestão de aprendizagem / Renan Pollo Benelli. -- Assis, 2024.

42p.

Trabalho de Conclusão de Curso (Graduação em Análise e Desenvolvimento de Sistemas) -- Fundação Educacional do Município de Assis (FEMA), Instituto Municipal de Ensino Superior de Assis (IMESA), 2024.

Orientador: Prof. Dr. Luiz Carlos Begosso.

1. Tecnologia educacional. 2. Ensino e aprendizagem. 3. Ludificação da aprendizagem. I Begosso, Luiz Carlos. II Título.

CDD 372.358

DEDICATÓRIA

Dedico este trabalho aos meus queridos pais, Valdecir Paulo Benelli e Maria Therezinha Pollo Benelli, que, com muita paciência e esforço, sempre me incentivaram a continuar e me deram suporte, ajudando-me a superar todas as barreiras.

AGRADECIMENTOS

Agradeço primeiramente a Deus, pelas oportunidades e condições que me foram concedidas.

Aos meus pais que sempre me apoiaram nos estudos, me dando força e compreendendo os momentos em que estive ausente por causa dos estudos.

Aos amigos do curso de Análise e Desenvolvimento de Sistemas, que estiveram comigo dividindo cada momento e etapas concluídas.

Ao professor orientador Luiz Carlos Begosso pelo apoio e contribuição, bem como a todos os professores que de uma forma ou outra, me ajudaram a ultrapassar esta etapa, obrigado pela riqueza de suas palavras e pela aprendizagem proporcionada.

RESUMO

O presente trabalho refere-se ao desenvolvimento de um sistema de gestão de aprendizagem denominado TechLingo, com foco no ensino de tecnologias. O objetivo principal é oferecer aos estudantes de tecnologia uma plataforma interativa com características de gamificação para aprimorar o aprendizado de tecnologias e torná-lo mais interessante. Para alcançar esse objetivo, foi desenvolvido um sistema web que proporciona uma oportunidade de aprendizado de novas tecnologias, utilizando elementos de gamificação para torná-lo mais interativo. O sistema foi projetado para suprir a necessidade de ferramentas para o ensino de tecnologia, auxiliando os estudantes a aprenderem melhor o conteúdo. Para o desenvolvimento do software, foram utilizadas as seguintes tecnologias: Java, Spring Boot, Angular, TypeScript, Bootstrap e MySQL, com o auxílio da IDE IntelliJ.

Palavras-chave: Programação; Tecnologia; TechLingo; Gestão de Aprendizagem; Gamificação.

ABSTRACT

The present work refers to the development of a learning management system called 'TechLingo', focusing on teaching technologies. The main objective is to offer technology students an interactive platform with gamification touches to improve technology learning and make it more interesting. To achieve this objective, a web system was developed that provides an opportunity to learn new technologies, using gamification elements to make it more interactive. The system was designed to meet the need for tools for teaching technology, helping students learn the content better. To develop the software, the following technologies were used: Java, Spring Boot, Angular, TypeScript, Bootstrap and MySQL, with the help of the IntelliJ IDE.

Keywords: Programming; Technology; TechLingo; Learning Management; Gamification

LISTA DE ILUSTRAÇÕES

Figura 1 - Esquema Metodológico	16
Figura 2 – Cronograma para o desenvolvimento do TechLingo	18
Figura 3 – Mapa Mental TechLingo	22
Figura 4 - Diagrama de Caso de Uso	23
Figura 5 - Diagrama de Classes	32
Figura 6 - Diagrama Entidade Relacionamento (DER)	33
Figura 7 – Página de controle do administrador.	34
Figura 8 - Página de cadastro de curso.....	35
Figura 9 - Página de alteração de Curso.	36
Figura 10 - Página Principal.....	37
Figura 11 - Página de seleção de curso.	38
Figura 12 - Página de Visão Geral do Curso	39
Figura 13 - Página de Aprendizagem	40

LISTA DE ABREVIATURAS E SIGLAS

API - Application Programming Interface (Interface de Programação de Aplicativos)

DER – Diagrama entidade Relacionamento

HTTP - Hypertext Transfer Protocol (Protocolo de Transferência de Hipertexto)

IDE – Integrated Development Environment (Ambiente de desenvolvimento integrado)

JVM - Java Virtual Machine (Máquina Virtual Java)

ORM - Object-Relational Mapping (Mapeamento Objeto-Relacional)

SQL - Structured Query Language (Linguagem de Consulta Estruturada)

TCC – Trabalho de Conclusão de curso

UML - Unified Modeling Language (Linguagem de Modelagem Unificada)

UC – Caso de Uso.

DB – Data Base (Banco de Dados).

SUMÁRIO

1. INTRODUÇÃO	12
1.1. OBJETIVOS GERAIS	13
1.2. OBJETIVOS ESPECÍFICOS.....	14
1.3. JUSTIFICATIVA.....	14
1.4. MOTIVAÇÃO	15
1.5. PERSPECTIVAS DE CONTRIBUIÇÃO.....	15
1.6. METODOLOGIA	15
1.7. GAMIFICAÇÃO.....	16
1.7.1. Características de Gamificação no TechLingo.....	17
1.8. CRONOGRAMA	18
2. TECNOLOGIAS UTILIZADAS PARA O DESENVOLVIMENTO DO PROJETO	19
2.1. JAVA.....	19
2.2. INTELLIJ IDEA	19
2.3. SPRING BOOT	20
2.4. ANGULAR	20
2.5. TYPESCRIPT.....	20
2.6. MYSQL.....	21
2.7. VISUAL STUDIO CODE	21
3. MODELAGEM DO SOFTWARE.....	22
3.1. MAPA MENTAL.....	22
3.2. DIAGRAMA DE CASO DE USO	23
3.3. NARRATIVAS DE CASO DE USO.....	24
3.4. DIAGRAMA DE CLASSES.....	32
3.5. DIAGRAMA ENTIDADE RELACIONAMENTO.....	33
4. DESCRIÇÃO DO SOFTWARE IMPLEMENTADO	34
4.1. PÁGINAS ADMINISTRATIVAS.....	34
4.1.1. Página controle do Administrador.....	34
4.1.2. Página de cadastro	35
4.1.3. Página de alteração.....	36

4.2.	PÁGINA INICIAL (HOME PAGE).....	37
4.3.	PÁGINA DE SELEÇÃO DE CURSO	38
4.4.	PÁGINA DE VISÃO GERAL DO CURSO	39
4.5.	PÁGINA DE APRENDIZAGEM	40
5.	CONCLUSÃO.....	41
5.1.	TRABALHOS FUTUROS	41
	REFERÊNCIAS	42

1. INTRODUÇÃO

Nos últimos anos, a área da tecnologia tem passado por um crescimento exponencial, desempenhando um papel fundamental na transformação digital em diferentes setores da sociedade. A demanda por profissionais qualificados em tecnologia nunca foi tão alta, e essa tendência continua a se expandir. O conhecimento em tecnologia deixou de ser uma habilidade restrita a especialistas e tornou-se uma competência necessária para diversas profissões. No entanto, aprender novas tecnologias e adquirir habilidades práticas nesse campo em constante evolução pode ser um desafio para muitas pessoas. A complexidade das tecnologias, como as linguagens de programação, a falta de recursos adequados e uma abordagem de ensino eficiente são obstáculos frequentemente enfrentados por aqueles que buscam aprender a programar.

Há algumas plataformas e recursos de aprendizagem de tecnologia e softwares que utilizam gamificação, como o Duolingo (2012), SoloLearn (2017) e Mimo (2020).

O Duolingo é um aplicativo de aprendizado de idiomas que se destaca por sua abordagem interativa e gamificada. Ele oferece uma ampla variedade de idiomas para os usuários escolherem e utiliza lições em formato de jogos para praticar habilidades de vocabulário, gramática, compreensão auditiva e leitura. Além disso, o Duolingo acompanha o progresso do usuário, oferece recompensas virtuais e fornece exercícios adaptativos.

O SoloLearn, por sua vez, é um aplicativo de aprendizado de programação focado em fornecer recursos abrangentes para o ensino de várias linguagens de programação. Ele oferece lições interativas, desafios de codificação, quizzes e um ambiente de comunidade para interação entre os usuários. Com um editor de código embutido, os usuários podem praticar e escrever programas diretamente no aplicativo.

O Mimo é um aplicativo de aprendizado de programação que se concentra em ensinar diversas linguagens de programação, incluindo Python, JavaScript, Kotlin, entre outras. Ele oferece lições interativas, exemplos de código, desafios práticos e projetos para ajudar os usuários a desenvolverem suas habilidades de programação. Além disso, o Mimo possui um editor de código embutido, permitindo que os usuários escrevam e testem seu código diretamente no aplicativo.

Com a ausência de um aplicativo que combine a abordagem interativa e gamificada do Duolingo com o ensino de tecnologias, surge uma demanda não atendida no mercado. Enquanto o Duolingo se concentra no ensino de idiomas e o SoloLearn e o Mimo estão direcionados ao ensino de programação, não existe uma plataforma abrangente que explore plenamente os princípios da gamificação para tornar o aprendizado de tecnologias tão envolvente e prazeroso quanto o aprendizado de idiomas proporcionado pelo Duolingo. Diante dessa situação, surgiu a necessidade de criar um aplicativo abrangente que aproveite plenamente os princípios da gamificação, com o foco no ensino de tecnologia, de forma a cativar os estudantes e despertar genuinamente seu interesse por aprender novas tecnologias, tornando o aprendizado mais prazeroso e estimulante.

A ausência de um aplicativo que combine a abordagem interativa e gamificada do Duolingo com o ensino de tecnologias revela uma demanda não atendida no mercado. Enquanto o Duolingo se concentra no ensino de idiomas e plataformas como o SoloLearn e o Mimo são voltadas para o ensino de programação, há uma carência de uma plataforma que explore os princípios da gamificação para tornar o aprendizado de tecnologias tão envolvente e interessante quanto o aprendizado de idiomas no Duolingo. Diante desse cenário, surge a necessidade de criar um aplicativo que utilize os princípios da gamificação com foco no ensino de tecnologia, a fim de cativar os estudantes e despertar um genuíno interesse por aprender novas tecnologias, tornando o aprendizado mais prazeroso e estimulante.

1.1. OBJETIVOS GERAIS

O objetivo deste trabalho foi desenvolver o TechLingo, um sistema de gestão de aprendizagem interativo com elementos de gamificação para tornar o aprendizado de tecnologias mais interessante e auxiliar os estudantes a aprenderem melhor. A lacuna que este projeto buscou preencher está relacionada ao fato de que, apesar da existência de aplicativos de ensino de sucesso como Duolingo, Mimo e SoloLearn, ainda há uma insuficiência de soluções que integrem, de forma gamificada, a abordagem do Duolingo no ensino de tecnologia.

1.2. OBJETIVOS ESPECÍFICOS

Em relação aos objetivos específicos, o presente trabalho propôs um sistema de gestão de aprendizagem voltado para a área de tecnologia. A proposta consistiu na elaboração de uma interface intuitiva e de fácil navegação, que incluiu as seguintes funcionalidades: permitir que o usuário selecione o curso; oferecer lições interativas com respostas e feedback imediato e personalizado para ajudar a identificar onde ocorreu o erro; utilizar princípios de gamificação para que o usuário ganhe pontos ao finalizar as lições e perda vidas ao errar uma resposta; e fornecer um certificado ao concluir o curso.

1.3. JUSTIFICATIVA

A justificativa para a realização deste trabalho é sustentada pela crescente importância da tecnologia na sociedade moderna e pela necessidade de desenvolver habilidades tecnológicas essenciais nos estudantes. Dessa forma, foi proposto o desenvolvimento de um aplicativo de aprendizagem de tecnologia que integra a abordagem gamificada e a experiência de aprendizado do Duolingo, tornando o aprendizado de tecnologia mais acessível a estudantes de todos os níveis de conhecimento.

Atualmente, a tecnologia desempenha um papel central em diversas esferas da vida e do trabalho, o que aumenta a demanda por profissionais qualificados em áreas como programação, design de experiência do usuário e outras habilidades relacionadas. No entanto, a aprendizagem dessas competências pode ser desafiadora devido à complexidade das matérias, à abordagem do ensino e às condições dos usuários.

É nesse contexto que aplicativos de sucesso, como Duolingo (2012), SoloLearn (2017) e Mimo (2020), têm se destacado ao empregar a gamificação para tornar o aprendizado mais prazeroso e estimulante. A ausência de uma plataforma abrangente que adote plenamente os princípios da gamificação no ensino de tecnologia representa uma oportunidade de contribuir significativamente para o campo da educação em tecnologia.

O TechLingo tem como objetivo preencher essa lacuna, oferecendo aos estudantes e entusiastas da tecnologia uma ferramenta inovadora para adquirir habilidades de forma eficaz e motivadora. O aplicativo desenvolvido usa os princípios da gamificação para tornar o aprendizado de tecnologia mais acessível e atraente para um público diversificado, despertando um genuíno interesse pela área.

1.4. MOTIVAÇÃO

A motivação para este trabalho surge da crescente importância da tecnologia na sociedade moderna e da necessidade de desenvolver habilidades tecnológicas essenciais nos estudantes. Assim, surgiu a demanda por um sistema de gestão de aprendizagem que incentive os estudantes a continuarem estudando e integre os princípios da gamificação. Portanto, a motivação intrínseca do autor reside na convicção de que ao tornar o aprendizado de tecnologia mais acessível, envolvente e prazeroso, pode-se equipar os aprendizes com as habilidades necessárias para enfrentar os desafios de um mundo cada vez mais tecnológico e digital. Acredita-se que essa iniciativa pode fazer uma valiosa contribuição à educação em tecnologia e preparar os estudantes para um mercado de trabalho em constante evolução.

1.5. PERSPECTIVAS DE CONTRIBUIÇÃO

Após a conclusão deste trabalho, espera-se que ele possa contribuir para o campo da educação em tecnologia de diversas maneiras. Primeiramente, tornando o aprendizado de tecnologia mais envolvente e motivador por meio da utilização de princípios da gamificação, como lições, recompensas e o sistema de vidas, o que torna o processo de aprendizado mais prazeroso e estimulante, ajudando os alunos a se manterem engajados e motivados. Além disso, o aplicativo será projetado para atender às necessidades de alunos com diferentes níveis de conhecimento. Por fim, ao ser disponibilizado gratuitamente, o aplicativo se tornará acessível a um público mais amplo, promovendo o acesso à educação em tecnologia e incentivando a inclusão.

1.6. METODOLOGIA

Para alcançar os objetivos de construção do TechLingo, elaborou-se uma estratégia metodológica que consistiu em: inicialmente, conduziram-se pesquisas aprofundadas sobre o tema, seguidas por reuniões com o público-alvo, composto por estudantes de tecnologia e entusiastas interessados em aprender sobre tecnologia. Essas reuniões foram realizadas presencialmente, com o objetivo de promover uma interação eficaz e produtiva para coletar feedback dos participantes. Após a obtenção dessas opiniões, os resultados foram registrados e avaliados para identificar possíveis melhorias. Posteriormente, realizaram-se

estudos para identificar o processo específico do negócio, a partir dos quais os requisitos do sistema foram levantados.

Após essa primeira etapa, foi realizada a análise do software utilizando os requisitos levantados. Nesse momento, foram criados: mapa mental, diagramas de casos de uso, diagramas de atividades e diagramas de classes utilizando UML com o software Astah Community. Para a prototipagem, foi utilizado o Figma.

Quanto ao desenvolvimento do software, a linguagem adotada foi Java, em conjunto com o *framework* Spring Boot e a biblioteca Lombok. O banco de dados da aplicação foi implementado em MySQL, acompanhado do *framework* Hibernate, responsável pelo mapeamento objeto-relacional (ORM). O *front-end* da aplicação foi desenvolvido utilizando Angular, TypeScript e Bootstrap.

A seguir, na figura 1, apresenta-se o esquema metodológico do trabalho:

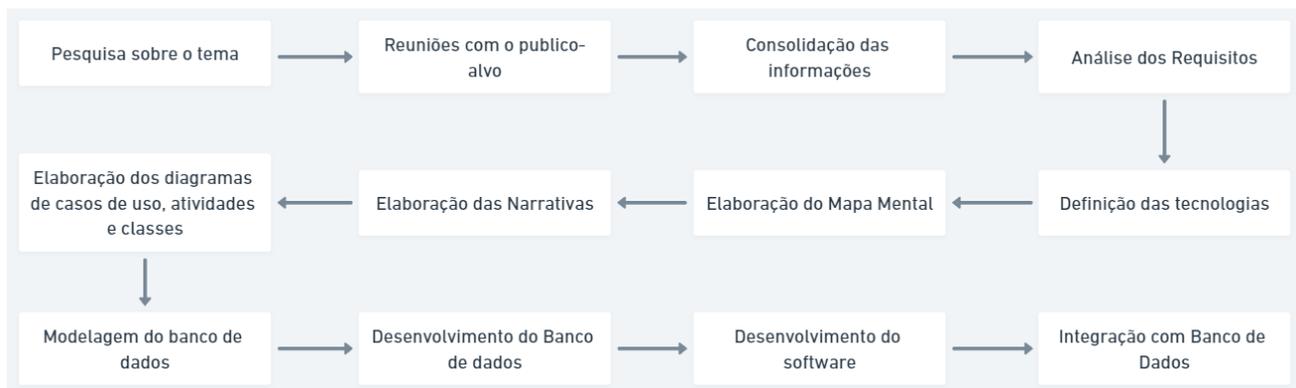


Figura 1 - Esquema Metodológico

1.7. GAMIFICAÇÃO

A gamificação é uma estratégia que utiliza elementos de jogos em contextos fora do entretenimento, como educação, trabalho e marketing. Seu objetivo é motivar e engajar as pessoas por meio de dinâmicas lúdicas, como desafios, recompensas, competição e pontuação. Isso é alcançado incorporando mecânicas como pontuação, níveis, vidas, recompensas, desafios, competições e rankings em atividades cotidianas, tornando-as mais atraentes e estimulantes. Esses elementos de jogo despertam o interesse dos usuários, ativando o desejo de atingir metas e progredir, promovendo maior envolvimento e engajamento.

Portanto, a gamificação oferece uma abordagem promissora para tornar o aprendizado de tecnologia mais atrativo e eficaz. Pesquisas indicam que a gamificação pode ser uma estratégia eficaz para tornar o aprendizado mais envolvente e motivador. Por exemplo, uma meta-análise conduzida por Xu, Liu e Ren (2021) integrou pesquisas empíricas e quantitativas sobre a gamificação em ambientes educacionais formais, revelando um impacto positivo nos resultados de aprendizagem dos alunos.

Além disso, um estudo conduzido por Jaques (2020) explorou os efeitos da gamificação no aprendizado, comportamento e engajamento dos alunos. Este estudo, que se concentrou em um ambiente de aprendizado de programação baseado na web, descobriu que a gamificação pode ser particularmente eficaz quando adaptada aos traços de personalidade dos alunos.

Esses estudos fornecem evidências convincentes de que a gamificação pode ser uma estratégia poderosa para tornar o aprendizado mais envolvente e motivador, especialmente no campo da tecnologia.

1.7.1. Características de Gamificação no TechLingo

No TechLingo, diversos elementos de gamificação foram incorporados para aprimorar a experiência de aprendizado. Os usuários acumulam pontos ao completar lições e etapas do curso, promovendo uma sensação de conquista e incentivando o progresso. Além disso, o sistema de vidas e penalidades, onde os usuários perdem vidas ao errar questões, cria um senso de urgência e responsabilidade, similar às penalidades em jogos.

O curso é estruturado em fases (unidades) que os usuários devem concluir para avançar e ganhar mais pontos, oferecendo uma progressão clara e recompensas para cada etapa. Ao finalizar o curso, os usuários recebem um certificado de conclusão, validando seu esforço e proporcionando uma sensação de sucesso.

A integração desses elementos de gamificação torna o aprendizado mais dinâmico e alinhado com as melhores práticas de gamificação, criando um ambiente mais envolvente e motivador para os usuários.

1.8. CRONOGRAMA

Para facilitar a visualização e o controle do progresso deste trabalho, foi estruturado um cronograma de atividades. De acordo com o Guia PMBOK (2017), o Cronograma é uma ferramenta de planejamento e controle das atividades a serem executadas durante o período estimado para um projeto. O cronograma detalha as atividades do projeto frente aos prazos de execução e permite que se tenha um controle sobre o andamento das atividades elencadas para ele.

A seguir, na figura 2, apresenta-se cronograma desenvolvido e executado para o presente projeto:

Atividades	Set	Out	Nov	Dez	Jan	Fev	Mar	Abr	Mai	Jun	Jul	Ago	Set
Estudo da Linguagem TypeScript	X												
Estudo Angular	X	X											
Estudo da Linguagem Java + Spring Boot		X											
Levantamento de Requisitos		X											
Elaboração das Narrativas			X										
Elaboração dos Diagramas de Classes			X										
Elaboração DER			X										
Elaboração do Protótipo				X									
Implementação do Banco de Dados				X	X								
Implementação do Menu Principal					X								
Escrita da Qualificação						X	X						
Exame da Qualificação								X					
Implementação do restante do Front-End								X	X				
Implementação Back-End									X	X			
Escrita Versão Final TCC										X	X	X	
Ajustes Finais no Sistema											X	X	X
Defesa													X

Figura 2 – Cronograma para o desenvolvimento do TechLingo

Como forma de organização deste presente trabalho apresenta-se a seguinte estrutura: o Capítulo 1 apresenta a introdução, contextualiza a área de aplicação, define os objetivos, a justificativa, a motivação do autor, as perspectivas de contribuição e a metodologia adotada. O Capítulo 2 descreve brevemente as tecnologias utilizadas. No Capítulo 3, são apresentados os requisitos e a arquitetura do sistema. O Capítulo 4 discute as principais características do sistema. Por fim, o Capítulo 5 apresenta as considerações finais e sugestões para trabalhos futuros.

2. TECNOLOGIAS UTILIZADAS PARA O DESENVOLVIMENTO DO PROJETO

Neste capítulo são tratados os recursos empregados para a concepção e implementação do *software* aqui proposto, exibindo ideias fundamentais para um entendimento mais completo da proposta de aplicação

2.1. JAVA

Java é uma linguagem de programação orientada a objetos que foi criada com o objetivo de permitir que os desenvolvedores escrevam programas uma vez e os executem em qualquer plataforma. Segundo Deitel e Deitel (2018), a linguagem Java foi projetada para ser segura, portátil, eficiente e independente de plataforma. Isso é possível graças à Máquina Virtual Java (JVM), que permite que o código Java seja executado em diferentes sistemas operacionais sem a necessidade de recompilar o código-fonte.

Além disso, a linguagem Java é amplamente utilizada em muitos setores, incluindo finanças, governo, saúde, entretenimento e telecomunicações. Sua popularidade é devida à sua portabilidade, desempenho, segurança e facilidade de uso

2.2. INTELLIJ IDEA

IntelliJ IDEA é um ambiente de desenvolvimento integrado (IDE) escrito em Java para o desenvolvimento de software de computador escrito em Java, Kotlin, Groovy e outras linguagens baseadas em JVM. Ele é desenvolvido pela JetBrains (anteriormente conhecido como IntelliJ). Segundo Akemi (2022), o IntelliJ IDEA é um ambiente de desenvolvimento integrado e com reconhecimento de contexto para trabalhar com Java e outras linguagens que rodam na JVM, como Kotlin, Scala e Groovy. Ele oferece facilidades na hora de declarar variáveis, estrutura de controle de fluxo e até mesmo nas próprias estruturas de código. Além disso, fornece dicas e truques que podem ser usados no cotidiano como pessoas desenvolvedoras, aumentando a produtividade no trabalho em projetos Java

2.3. SPRING BOOT

Spring Boot é um framework baseado em Java que simplifica a configuração e implantação de aplicativos. Ele nasceu a partir do *Spring Framework* com o objetivo de facilitar as configurações iniciais de um projeto. O Spring Boot permite que você crie aplicações robustas e altamente configuráveis com um mínimo de esforço. Segundo José (2022), o Spring Boot reduz drasticamente o tempo de desenvolvimento e aumenta a produtividade da equipe, configurando todos os componentes para um aplicativo de nível de produção e que ele vem com servidores (HTTP) embutidos como o Tomcat, assim podendo testar páginas web. Ele também menciona que o Spring Boot é muito popular na comunidade de desenvolvedores Java, o que facilita encontrar materiais de aprendizagem ou tirar dúvidas rapidamente.

2.4. ANGULAR

Angular é um framework de código aberto, desenvolvido pelo Google, para a criação de aplicativos da web dinâmicos e interativos. Ele se baseia em TypeScript, uma linguagem superset do JavaScript, e utiliza uma abordagem orientada a componentes para a construção de interfaces. O Angular oferece uma estrutura sólida para desenvolvimento, incluindo suporte a gerenciamento de estado, roteamento, validação de formulários e muito mais. Segundo Batista (2023), o Angular é o framework escolhido por muitas empresas e pessoas desenvolvedoras devido à sua robustez e conjunto de recursos abrangente. Sua estrutura modularizada promove a reutilização de código e a manutenção mais fácil.

2.5. TYPESCRIPT

TypeScript é um *superset* da linguagem JavaScript criado pela Microsoft que permite a escrita de *scripts* com a utilização de tipagem estática e orientação a objetos, facilitando a escrita de código com uma sintaxe de fácil compreensão. Ele adiciona recursos avançados ao JavaScript, como a tipagem estática e interfaces, tornando mais fácil detectar e prevenir erros durante a fase de desenvolvimento. Ao compilar um código TypeScript, é gerado um código JavaScript, e esse código é o que será executado no navegador. De acordo com Neves (2024), o TypeScript não apenas melhora a legibilidade e a manutenibilidade do código, mas também antecipa erros que poderiam ser custosos se identificados apenas em

tempo de execução. Esse sistema de tipos ajuda desenvolvedoras e desenvolvedores a construir aplicações mais complexas e seguras, fornecendo uma documentação implícita e facilitando as refatorações. Além disso, TypeScript vem com ferramentas poderosas de análise de código que integram-se perfeitamente a vários editores de texto e ambientes de desenvolvimento, proporcionando uma experiência de desenvolvimento mais produtiva e agradável.

2.6. MYSQL

O MySQL é um sistema gerenciador de banco de dados relacional de código aberto usado na maioria das aplicações gratuitas para gerir suas bases de dados. O serviço utiliza a linguagem SQL (*Structure Query Language*) como interface, que é a linguagem mais popular para inserir, acessar e gerenciar o conteúdo armazenado num banco de dados. De acordo com DuBois (2013), o MySQL é um sistema de gerenciamento de banco de dados relacional de código aberto que se destaca por sua velocidade e facilidade de uso. Ele é particularmente adequado para o desenvolvimento de sites e aplicações com suporte a banco de dados. DuBois descreve tudo, desde os fundamentos de inserir informações em um banco de dados e formular consultas, até usar o MySQL com PHP ou Perl para gerar páginas da web dinâmicas, escrever seus próprios programas que acessam bancos de dados MySQL e administrar servidores MySQL.

2.7. VISUAL STUDIO CODE

O Visual Studio Code é um editor de código-fonte desenvolvido pela Microsoft para Windows, Linux e macOS. Ele inclui suporte para depuração, controle de versionamento Git incorporado, realce de sintaxe, complementação inteligente de código, *snippets* e refatoração de código. Ele é customizável, permitindo que os usuários possam mudar o tema do editor, teclas de atalho e preferências. Ele é um software livre e de código aberto, apesar do download oficial estar sob uma licença proprietária.

3. MODELAGEM DO SOFTWARE

3.1. MAPA MENTAL

Mapa mental, é um tipo de diagrama, sistematizado pelo psicólogo inglês Tony Buzan, voltado para a gestão de informações, de conhecimento e de capital intelectual; para a compreensão e solução de problemas; na memorização e aprendizado; na criação de manuais, livros e palestras; como ferramenta de *brainstorming* (tempestade de ideias); e no auxílio da gestão estratégica de uma empresa ou negócio.

Os mapas mentais procuram representar, com o máximo de detalhes possíveis, o relacionamento conceitual existente entre informações que normalmente estão fragmentadas, difusas e pulverizadas no ambiente operacional ou corporativo. Trata-se de uma ferramenta para ilustrar ideias e conceitos, dar-lhes forma e contexto, traçar os relacionamentos de causa, efeito, simetria e/ou similaridade que existem entre elas e torná-las mais palpáveis e mensuráveis, sobre os quais se possa planejar ações e estratégias para alcançar objetivos específicos. De acordo com Galante (2023) mapas mentais são ferramentas que estimulam uma nova percepção na maneira de ensinar e aprender, criando um ambiente propício para uma melhor compreensão e interpretação das ideias.

A seguir, na figura 3, apresenta-se o Mapa Mental do Projeto:

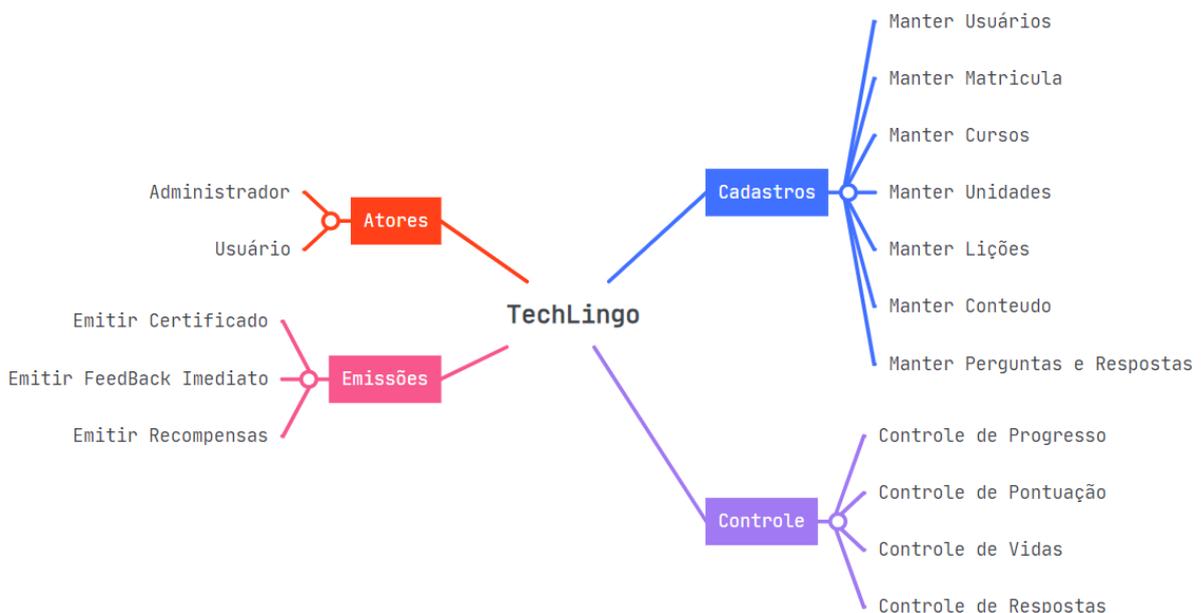


Figura 3 – Mapa Mental TechLingo

3.2. DIAGRAMA DE CASO DE USO

O diagrama de casos de uso é uma das principais técnicas de modelagem utilizadas na engenharia de requisitos para representar os requisitos funcionais de um sistema. Segundo Kotonya e Sommerville (1998, p. 60), "um diagrama de casos de uso é uma descrição gráfica de uma coleção de atores, casos de uso e seus relacionamentos".

Os autores Kotonya e Sommerville (1998) ressaltam que o diagrama de casos de uso é uma ferramenta útil para a comunicação entre os membros da equipe de desenvolvimento e os usuários do sistema, pois permite a visualização das principais funcionalidades e interações do sistema de uma forma simples e intuitiva. Eles ainda destacam que o diagrama de casos de uso pode ser utilizado para a validação dos requisitos do sistema, ajudando a garantir que todas as funcionalidades importantes sejam consideradas.

Abaixo, na figura 4, apresenta-se o diagrama de Caso de Uso para o TechLingo:

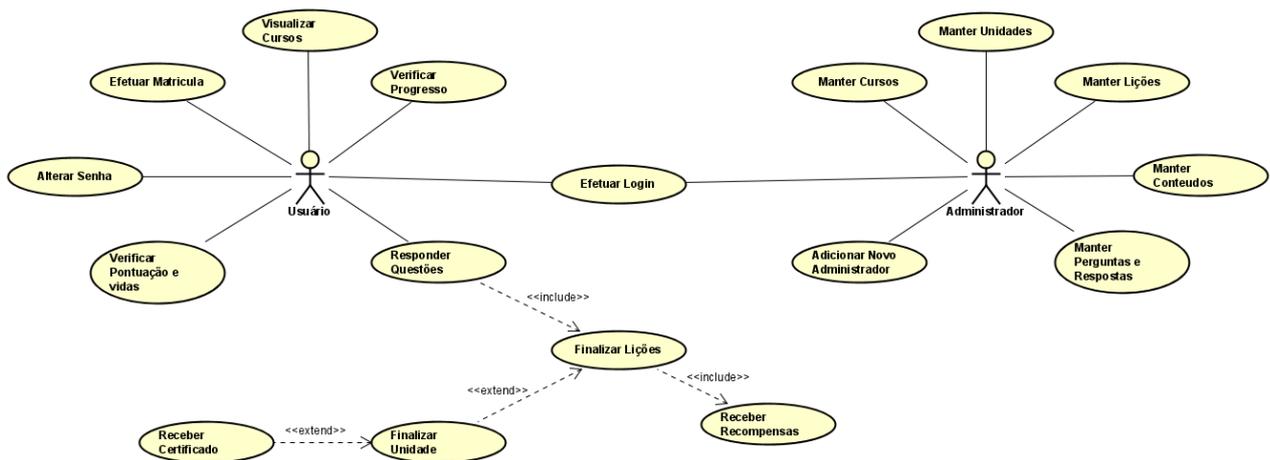


Figura 4 - Diagrama de Caso de Uso

3.3. NARRATIVAS DE CASO DE USO

O diagrama de caso de uso descreve a funcionalidade proposta para um novo sistema que será projetado, é uma excelente ferramenta para o levantamento dos requisitos funcionais do sistema.

UC 01 – Manter Cursos

1- Finalidade/Objetivo	Permitir que o ator faça o cadastro, alteração, consulta e exclusão de Cursos.
2- Atores	Administrador
3- Pré-condições	O ator precisa estar logado no sistema.
4- Evento Inicial	O ator começa o caso de uso clicando na Opção “Cursos”
5- Fluxo Principal	<p>a) O sistema apresenta na tela as opções “Cadastrar Curso” e “Buscar Curso”.</p> <p>b) O ator seleciona a opção “Buscar Curso”; [A1]</p> <p>c) O Sistema apresenta todos os cursos existentes [E1];</p> <p>d) O ator visualiza os dados do curso [A2] [A3];</p> <p>e) O Ator clica no botão fechar;</p> <p>f) Caso de uso é encerrado.</p>
6- Fluxo Alternativo	<p>A1 – Incluir novo curso</p> <p>a) O ator clica na opção “Cadastrar”;</p> <p>b) O sistema apresenta os campos para digitação dos dados do Curso;</p> <p>c) O ator informa os dados e clica no botão Salvar;</p> <p>d) Os dados são salvos;</p>

	<p>e) volta ao passo a do fluxo principal;</p> <p>A2 – Alterar Curso</p> <p>a) O ator clica na opção de “Alterar”;</p> <p>b) O sistema exibe uma página com os campos preenchidos com os dados atuais do curso, permitindo que o ator os edite.</p> <p>c) O ator informa as novas informações e clica no botão Salvar;</p> <p>d) Os dados são salvos;</p> <p>e) volta passo “a” do fluxo principal.</p> <p>A3 – Excluir Curso</p> <p>a) O ator clica na opção de “Excluir”;</p> <p>b) O sistema exige confirmação de exclusão[E2]</p> <p>e) O Sistema remove dados do Curso do DB.</p> <p>c) Volta ao passo “a” do fluxo principal</p>
7- Fluxo de Exceção	<p>E1 – Cursos não encontrados</p> <p>a) Nenhum curso foi encontrado. O sistema carrega a página, mas não lista os cursos.</p> <p>b) Volta ao passo “b” do fluxo principal;</p> <p>E2- Desiste da exclusão</p> <p>a) O ator clica em cancelar</p> <p>b) O sistema saí do modo de exclusão</p> <p>c) Volta ao passo “a” do fluxo principal.</p>

UC 02 – Efetuar Matrícula

1- Finalidade/Objetivo	Permitir que o ator realize sua matrícula.
2- Atores	Usuário (Jogador).
3- Pré-condições	O ator precisa estar logado no sistema.
4- Evento Inicial	O ator começa o caso de uso clicando na opção “Aprender”
5- Fluxo Principal	<p>a) O sistema apresenta uma tela com opções de Cursos;</p> <p>b) O ator seleciona a opção [T1];</p> <p>c) O aluno é cadastrado no Curso;</p> <p>d) Caso de uso é encerrado.</p>
6- Fluxo Alternativo	<p>A1 – Jogador já está matriculado no curso.</p> <p>a) O jogador já está matriculado no curso. O sistema carrega os dados do curso e o progresso do usuário;</p> <p>b) Retorna ao passo "d" do fluxo principal.</p>
7- Casos de Teste	<p>T1 – Verificar Matrícula</p> <p>a) O sistema verifica se o jogador já está matriculado no curso;</p> <p>b) Caso o jogador já esteja matriculado, vai para o passo “a” do Fluxo Alternativo [A1];</p> <p>c) Caso não esteja matriculado, vai para o passo “c” do fluxo principal</p>

UC 03 – Receber Certificado

1- Finalidade/Objetivo	Permitir que o ator receba seu certificado.
2- Atores	Usuário (Jogador).
3- Pré-condições	O ator precisa ter finalizado a unidade.
4- Evento Inicial	O ator começa o caso de uso clicando na Opção “Receber Certificado”.
5- Fluxo Principal	<p>a) O sistema apresenta uma tela parabenizando o ator por ter concluído a unidade;</p> <p>b) O ator aperta em “Continuar”;</p> <p>c) O Sistema gera um certificado contendo o título, a descrição da Unidade e a quantidade de horas aproximadas e mostra na tela;</p> <p>d) O Ator visualiza o certificado e aperta em fechar [A1];</p> <p>e) Caso de uso é encerrado</p>
6- Fluxo Alternativo	<p>A1 – Imprimir da Certificado;</p> <p>a) O ator “Clica” em imprimir;</p> <p>b) O sistema entra em modo de impressão e imprime o certificado;</p> <p>c) Volta ao passo “e” do fluxo principal</p>

UC 04 – Responder Questões

1- Finalidade/Objetivo	Permitir que o usuário Responda Questões
2- Atores	Usuário (Jogador)
3- Pré-condições	Ter se matriculado no Curso
4- Evento Inicial	O ator começa o caso de uso clicando na Opção “Começar”
5- Fluxo Principal	<p>a) O sistema apresenta uma tela que pode conter informações de algum conteúdo, ou alguma pergunta;</p> <p>b) O ator clica em alguma opção de resposta;</p> <p>c) O ator aperta no botão “Verificar” [T1];</p> <p>d) O Sistema emite o feedback com a resposta da pergunta;</p> <p>e) O ator clica em “Continuar”;</p> <p>f) O sistema vai para a próxima pergunta e entra em um loop até responder todas as perguntas;</p> <p>g) Ao responder todas as perguntas o ator aperta em “Concluir”;</p> <p>h) Caso de uso é encerrado.</p>
6- Fluxo Alternativo	<p>A1 – Resposta Incorreta</p> <p>a) O sistema verificou a resposta e a resposta está incorreta</p> <p>b) O sistema salva a questão e a coloca no final da fila de questões para que o jogador possa respondê-la novamente quando terminar de responder as questões</p> <p>c) Volta ao passo “d” do fluxo principal</p>
7- Casos de Teste	<p>T1 – Verificar Resposta</p> <p>a) O sistema verifica a resposta do ator;</p>

	<p>b) Caso esteja correta, vai para o passo “d” do fluxo principal;</p> <p>c) Caso esteja incorreta, vai para o passo “a” do fluxo alternativo [A1];</p>
--	--

UC 05 – Manter Perguntas e Respostas

1- Finalidade/Objetivo	Permitir que o ator faça inclusão, alteração, consulta e exclusão de “Perguntas e respostas”.
2- Atores	Administrador
3- Pré-condições	O ator precisa estar logado no sistema.
4- Evento Inicial	O ator começa o caso de uso clicando na Opção “Perguntas e Respostas”.
5- Fluxo Principal	<p>a) O sistema apresenta na tela as opções “Cadastrar Perguntas e Respostas” e “Buscar Perguntas e Respostas”;</p> <p>b) O ator seleciona a opção “Buscar Perguntas e Respostas” [A1];</p> <p>c) O Sistema apresenta todas as perguntas e respostas existentes [E1];</p> <p>d) O ator visualiza os dados das perguntas e as respostas[A2] [A3];</p> <p>e) O Ator clica no botão fechar;</p> <p>f) Caso de uso é encerrado.</p>
6- Fluxo Alternativo	<p>A1 – Incluir nova Pergunta e Respostas</p> <p>a) O ator clica na opção “Cadastrar”;</p>

	<p>b) O sistema apresenta os campos para digitação da pergunta e, logo abaixo, um botão de “Adicionar Resposta”</p> <p>c) O ator preenche todos os campos e clica no botão “Adicionar Resposta”;</p> <p>d) O ator informa as informações da resposta;</p> <p>e) O ator clica no botão “Salvar”;</p> <p>f) Os dados são salvos;</p> <p>g) Volta ao passo “a” do fluxo principal.</p> <p>A2 – Alterar Perguntas e Respostas</p> <p>a) O ator clica na opção “Alterar”;</p> <p>b) O sistema exibe uma página com os campos preenchidos com os dados atuais da pergunta e suas respostas, permitindo que o ator os edite;</p> <p>c) O ator informa as novas informações e clica no botão “Salvar”;</p> <p>d) Os dados são salvos;</p> <p>e) Volta passo “a” do fluxo principal.</p> <p>A3 – Excluir Perguntas e Respostas</p> <p>a) O ator clica na opção “Excluir”;</p> <p>b) O sistema exige confirmação de exclusão[E2]</p> <p>e) O sistema remove os dados da pergunta e suas respostas do DB;</p> <p>c) Volta ao passo “a” do fluxo principal</p>
7- Fluxo de Exceção	E1 – Perguntas não encontradas

	<p>a) Nenhuma pergunta foi encontrada. O sistema carrega a página, mas não lista as perguntas;</p> <p>b) Volta ao passo “b” do fluxo principal.</p> <p>E2- Desiste da exclusão</p> <p>a) O ator clica em cancelar</p> <p>b) O sistema sai do modo de exclusão</p> <p>c) Volta ao passo “a” do fluxo principal.</p>
--	---

UC 06 – Receber Recompensas

1- Finalidade/Objetivo	Permitir que o usuário receba as recompensas
2- Atores	Jogador (Aluno)
3- Pré-condições	Ter Finalizado uma lição
4- Evento Inicial	O ator começa o caso de uso clicando na Opção “Receber Recompensas”
5- Fluxo Principal	<p>a) O sistema apresenta uma tela parabenizando o jogador e mostrando sua vida atual, a quantidade de pontos a serem recebidos e o tempo que o jogador demorou para finalizar a lição.</p> <p>b) O ator clica no botão “Receber”</p> <p>c) O sistema acrescenta a quantidade de pontos a serem recebidos.</p> <p>d) O caso de uso é encerrado.</p>

3.4. DIAGRAMA DE CLASSES

Um diagrama de classes é uma representação visual da estrutura e das relações entre as classes em um sistema orientado a objetos. Ele é uma parte fundamental da modelagem de dados e é amplamente utilizado na engenharia de software durante a fase de design. Os diagramas de classes descrevem as classes, seus atributos, métodos e as relações entre elas. Pressman (2016) destaca que o diagrama de classes é uma das principais ferramentas utilizadas na UML (*Unified Modeling Language*), que é uma linguagem padrão para modelagem de sistemas orientados a objetos. Ele ressalta que o diagrama de classes é importante para documentar a arquitetura do sistema e para comunicar as informações de maneira clara e precisa entre os membros da equipe.

A seguir, na figura 5, apresenta o diagrama de classes mencionado.

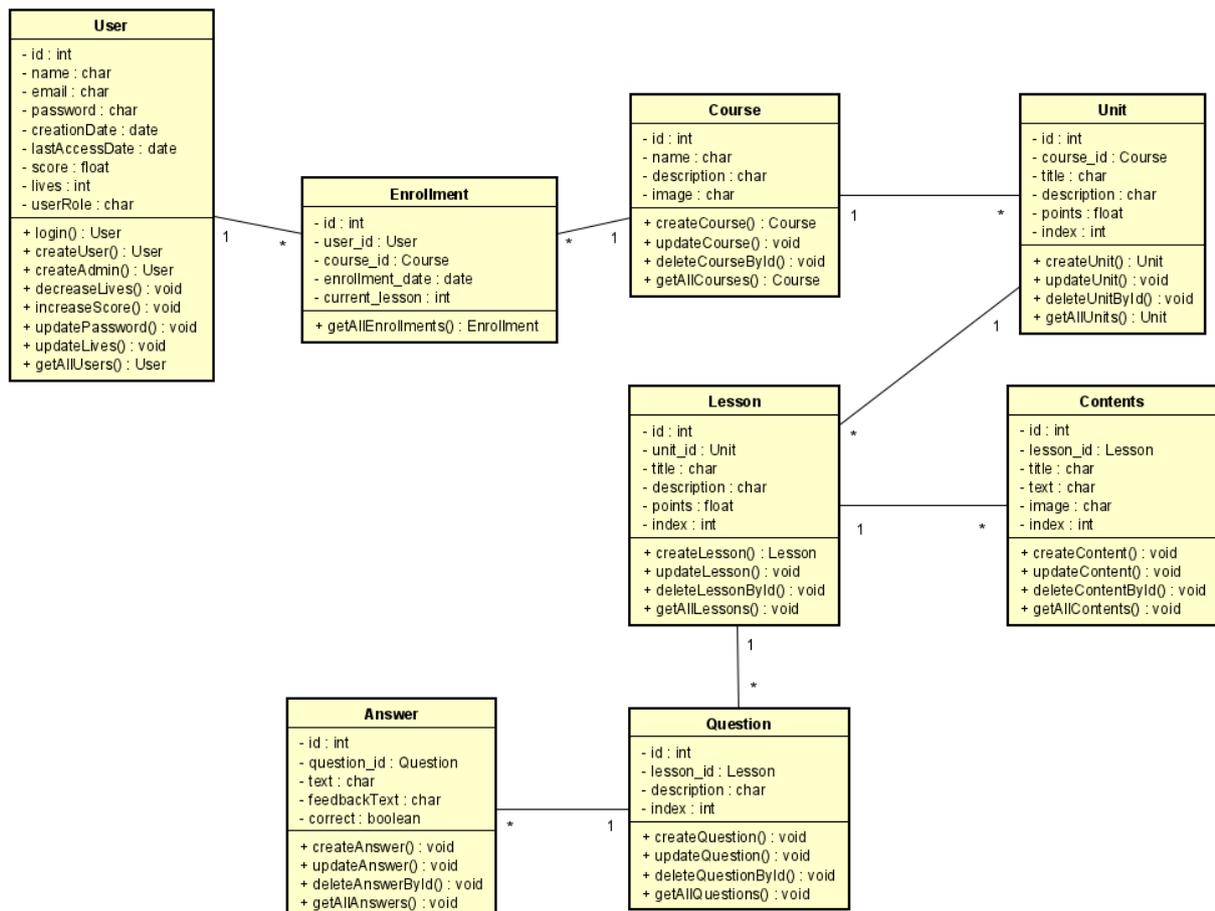


Figura 5 - Diagrama de Classes

3.5. DIAGRAMA ENTIDADE RELACIONAMENTO

Diagrama de Entidade e Relacionamento (DER) é uma representação gráfica do Modelo de Entidade e Relacionamento e evita excesso de abstração, trazendo para a realidade informações pertinentes de uma forma mais visual e, conseqüentemente, mais intuitiva.

De acordo com Nogueira (1988), um diagrama entidade-relacionamento é um tipo de fluxograma que ilustra como entidades, pessoas, objetos ou conceitos se relacionam entre si dentro de um sistema. Diagramas entidade-relacionamento são mais utilizados para projetar ou depurar bancos de dados relacionais nas áreas de Engenharia de Software, sistemas de informações empresariais, educação e pesquisa.

A Figura 6 ilustra o diagrama entidade relacionamento do TechLingo.

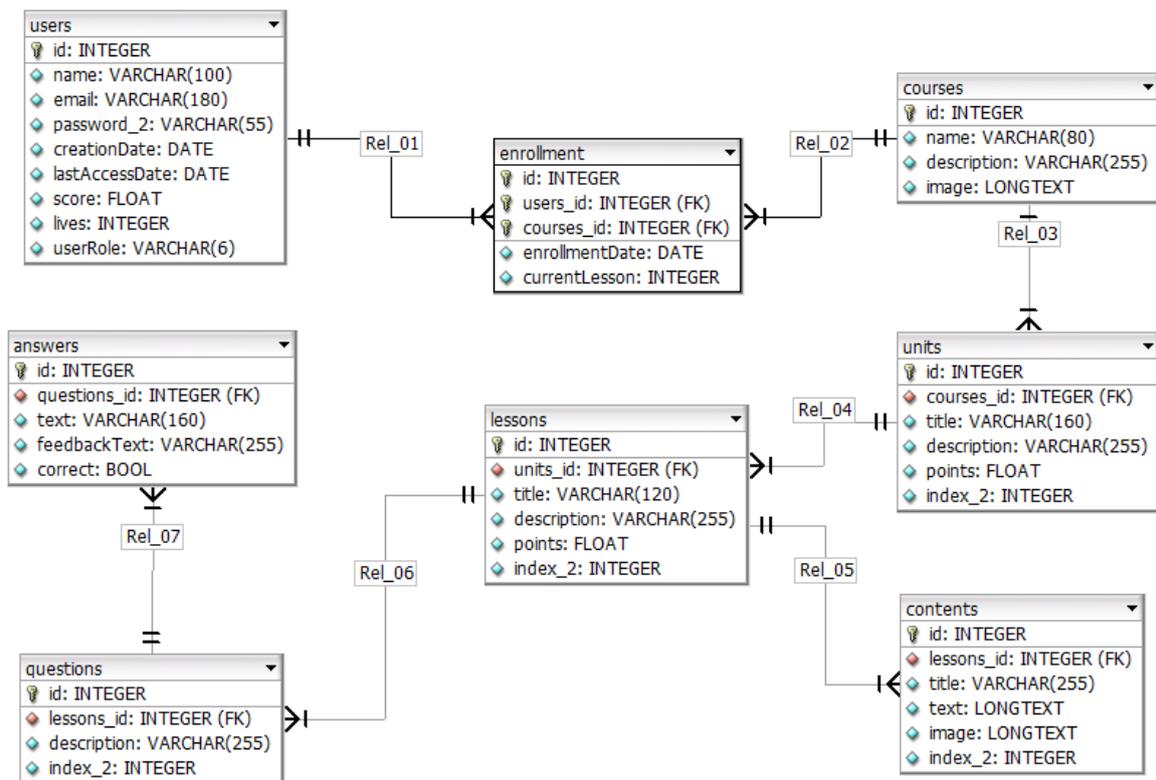


Figura 6 - Diagrama Entidade Relacionamento (DER)

4. DESCRIÇÃO DO SOFTWARE IMPLEMENTADO

Este capítulo tem o objetivo de detalhar as principais funcionalidades do software, oferecendo uma maior compreensão do sistema. Este capítulo está dividido em cinco seções: Página Inicial, Páginas Administrativas, Página de Seleção de Curso, Página de Visão Geral do Curso e Página de Aprendizagem.

4.1. PÁGINAS ADMINISTRATIVAS

A seguir, estão alguns exemplos das principais páginas de administrador do sistema.

4.1.1. Página controle do Administrador

A página de controle do administrador é um componente essencial para o sistema TechLingo, pois é onde o administrador pode cadastrar cursos, unidades, lições, conteúdos, perguntas e respostas, além de realizar ações de alteração, exclusão e listagem, como a apresentada na figura 7. Seu principal objetivo é permitir a gestão do sistema, fornecendo e atualizando informações.

A seguir, na figura 7, apresenta-se a Página de controle do administrador.

The screenshot shows the 'Consulta de Cursos' page in the TechLingo system. On the left is a sidebar with navigation options: Cursos (Cadastrar, Buscar Curso), Unidades, Lições, Conteúdos, Perguntas e Respostas, and Conta (Novo, Sair). The main content area displays a table with the following data:

Código	Nome do Curso	Descrição	Ações
1	Curso de Java + Spring boot	O melhor curso de java do brasil	Editar Excluir
2	Curso de python	Para iniciar sua jornada na programação, é crucial d...	Editar Excluir

Figura 7 – Página de controle do administrador.

4.1.2. Página de cadastro

As páginas de cadastro são componentes que permitem a adição de novas informações ao sistema e mantêm um padrão, já que todas as telas de cadastro seguem o mesmo layout. A página de cadastro de curso, apresentada na figura 8, assim como as outras telas de cadastro, possuem dois botões: um para realizar o cadastro e outro para voltar, caso o usuário desista da operação. Para cadastrar um curso, o administrador deve preencher todos os campos indicados na tela e, nos campos de imagem, fornecer a URL correspondente à imagem.

A seguir, na figura 8, apresenta-se a Página de cadastro de curso.

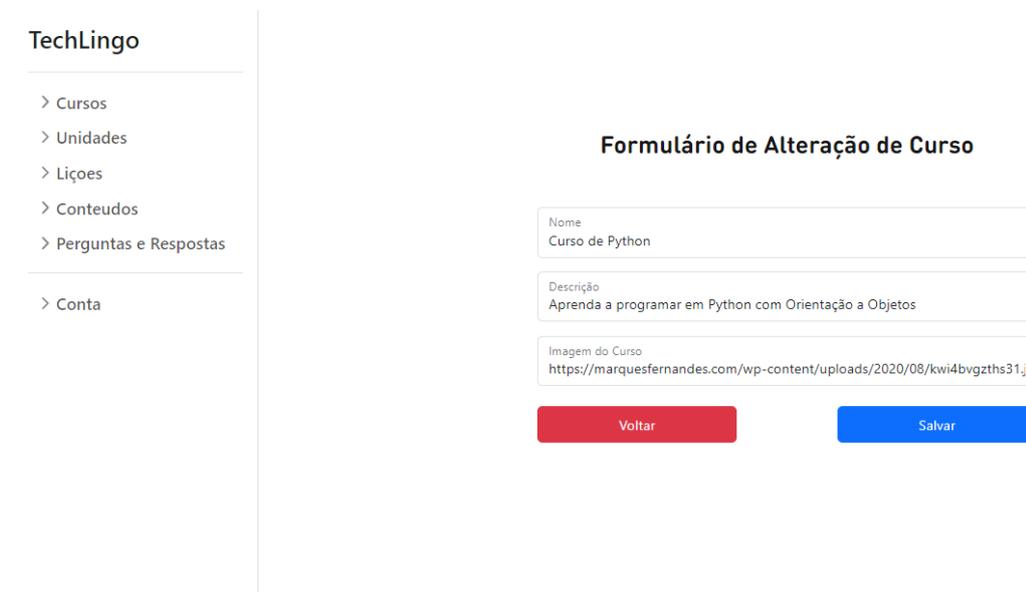
The image shows a web interface for a course registration form. On the left is a sidebar menu for 'TechLingo' with a 'Cursos' section expanded to show 'Cadastrar' and 'Buscar Curso'. The main content area is titled 'Formulário de Cadastro de Curso' and contains three text input fields labeled 'Nome', 'Descrição', and 'Imagem do Curso'. At the bottom of the form are two buttons: a red 'Voltar' button and a blue 'Cadastrar' button.

Figura 8 - Página de cadastro de curso.

4.1.3. Página de alteração

As páginas de alteração são componentes que permitem a modificação de informações no sistema e mantêm um padrão, já que todas as telas de alteração seguem o mesmo layout. A página de alteração de curso, apresentada na Figura 9, assim como as outras telas de alteração, possuem dois botões: um para salvar e outro para voltar, caso o usuário desista da operação. Para alterar um curso, o administrador deve modificar os campos desejados e clicar em salvar.

A seguir, na figura 9, apresenta-se a Página de alteração de Curso.



TechLingo

- > Cursos
- > Unidades
- > Lições
- > Conteudos
- > Perguntas e Respostas

> Conta

Formulário de Alteração de Curso

Nome
Curso de Python

Descrição
Aprenda a programar em Python com Orientação a Objetos

Imagem do Curso
<https://marquesfernandes.com/wp-content/uploads/2020/08/kwi4bvgzths31.j>

Voltar Salvar

Figura 9 - Página de alteração de Curso.

4.2. PÁGINA INICIAL (HOME PAGE)

A página inicial do sistema (*Home Page*) é o primeiro componente do sistema e também a página principal. Ao acessar o site, o usuário se deparará com ela, que contém um texto de apresentação e dois botões: um para realizar o login e outro para o cadastro. Sua principal função é autenticar o usuário.

A seguir, na figura 10, apresenta-se a Página principal do sistema.

TechLingo



O jeito grátis, divertido e eficaz de aprender uma tecnologia!

COMECE AGORA

JÁ TENHO UMA CONTA

Figura 10 - Página Principal.

4.3. PÁGINA DE SELEÇÃO DE CURSO

A página de seleção de curso é um componente do software, cuja principal função é permitir que o usuário selecione o curso que deseja acessar ou no qual deseja se matricular. Após o usuário realizar o login ou o cadastro, ele será redirecionado para essa página.

A seguir, na figura 11, apresenta-se a Página de seleção de curso.

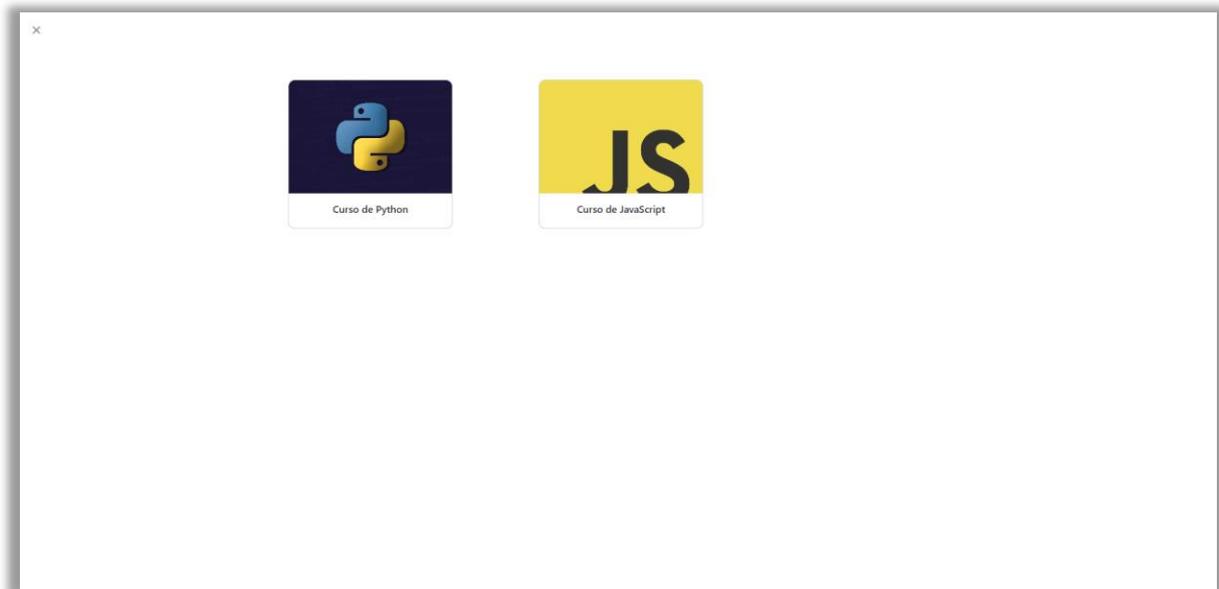


Figura 11 - Página de seleção de curso.

4.4. PÁGINA DE VISÃO GERAL DO CURSO

A página de visão geral do curso é um componente do software e apresenta diversas opções. Ela possui um menu lateral com as opções Aprender, Perfil e Mais. A opção 'Aprender' redirecionará o usuário para a página de seleção de curso. A opção 'Perfil' exibirá informações sobre o jogador, como nome, vidas, pontuação e progresso. A opção 'Mais' permitirá ao usuário alterar a senha e sair do sistema. Na parte central da página, encontram-se as unidades e as lições do curso selecionado. À direita, são exibidas informações do usuário, como pontuação, vidas e o curso selecionado.

A seguir, na figura 12, apresenta-se a Página de Visão Geral do Curso

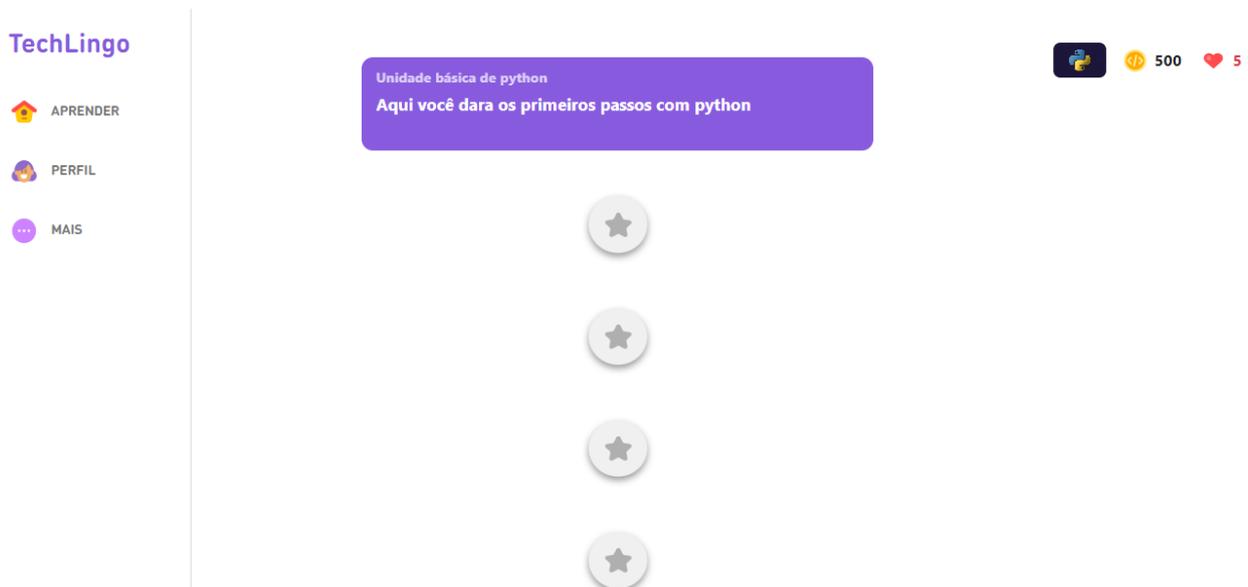


Figura 12 - Página de Visão Geral do Curso

4.5. PÁGINA DE APRENDIZAGEM

A página de aprendizagem é onde o usuário poderá ler o conteúdo e responder às perguntas. Nessa página, existe uma barra de progresso no topo. No lado esquerdo da barra, há um botão de fechar, caso o usuário desista de responder à lição, e do outro lado está a 'vida do usuário', que é diminuída a cada resposta errada. A página pode conter conteúdos explicando algum tópico da tecnologia e, em seguida, uma pergunta com opções de resposta. Para responder a uma pergunta, basta selecionar a resposta e clicar em 'Continuar', e o sistema realizará a verificação da resposta e emitirá o feedback, após isso irá para o próximo conteúdo ou pergunta. Também há um botão de 'Voltar', caso o usuário queira revisar algo.

A seguir, na figura 13, apresenta-se a Página principal do sistema.

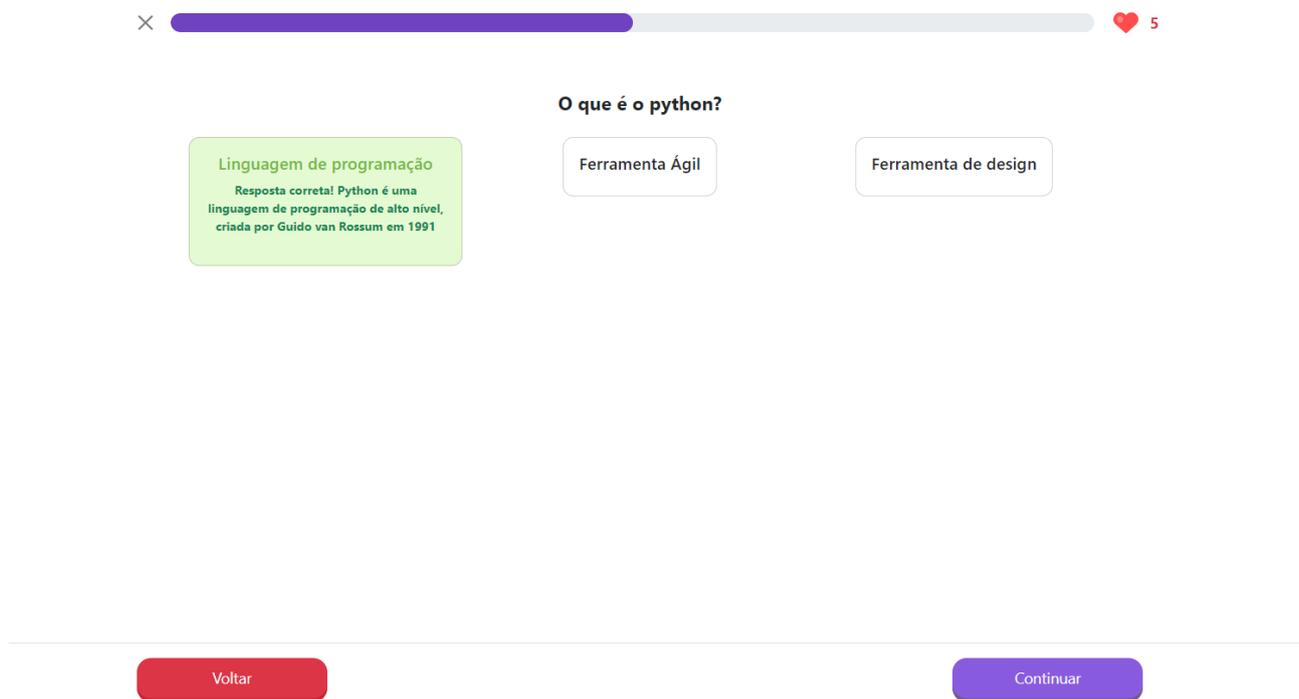


Figura 13 - Página de Aprendizagem

5. CONCLUSÃO

Este trabalho apresentou uma abordagem gamificada com o intuito de desenvolver o software TechLingo, focado em integrar os princípios da gamificação ao ensino de tecnologia. O principal objetivo foi criar um software capaz de auxiliar estudantes e entusiastas da área de tecnologia a aprender de forma eficaz, interativa e motivadora.

A escolha do TechLingo foi impulsionada pela oportunidade de inovar o processo de aprendizagem tecnológico, aplicando elementos de gamificação para incentivar o progresso dos usuários. Esses elementos foram cuidadosamente integrados ao design do sistema para promover uma maior motivação e engajamento, com a intenção de tornar o processo de aprendizagem mais leve e envolvente. Além disso, a plataforma foi pensada para criar um ambiente que estimule o aprendizado contínuo, encorajando o usuário a avançar em seu próprio ritmo.

Embora o TechLingo ainda não tenha sido testado com usuários reais, seu desenvolvimento foi fundamentado em aplicativos similares e pesquisas. Isso possibilitou a criação de um design centrado no usuário, buscando oferecer uma experiência intuitiva e envolvente, alinhada às melhores práticas.

5.1. TRABALHOS FUTUROS

Ao longo do desenvolvimento deste trabalho, foram identificadas possíveis oportunidades para aprimorar o software e atender ainda melhor às necessidades dos usuários. Embora o desenvolvimento do software tenha sido baseado em aplicativos similares e pesquisas, a validação com usuários reais se mostrou fundamental para identificar pontos fortes, problemas de usabilidade e obter insights sobre a experiência do usuário no dia a dia.

A avaliação com usuários reais permitirá compreender como a plataforma é utilizada, quais as dificuldades encontradas e quais os aspectos positivos que podem ser otimizados. Além disso, identificamos diversas funcionalidades que podem ser incorporadas em futuras versões do sistema, como a integração de um terminal e compilador para que os usuários possam digitar e executar códigos, e a implementação de um sistema de pontos para aquisição de itens adicionais.

REFERÊNCIAS

AKEMI, A. (2022). IntelliJ IDEA: dicas e truques para usar no dia a dia. Alura. Disponível em <https://www.alura.com.br/artigos/intellij-idea-dicas-truques-usar-no-dia-a-dia>. Acesso em: 04 jan. 2024.

BATISTA, Nyanne. (2023). Angular: o que é, para que serve e um Guia para iniciar no framework JavaScript. Disponível em: <https://www.alura.com.br/artigos/angular-js>. Acesso em 07 jan. 2024.

DEITEL, P. J. e Deitel, H. M. (2018). Java: Como programar. São Paulo: Pearson.

DUBOIS, Paul, **MySQL**, 5. Ed, São Paulo, Novatec, 2013.

DUOLINGO. Aplicativo Duolingo. Disponível em: <https://www.duolingo.com/>. Acesso em: 14 jun. 2023.

GALANTE, Carlos Eduardo da Silva. O uso de mapas conceituais e de mapas mentais como ferramentas pedagógicas no do ensino superior. In: CONGRESSO DE ENSINO SUPERIOR, 1, 2023, São Paulo, Brasil. Anais do Congresso de Ensino Superior, vol. 1, jan. 2023, p. 1-10.

LUDOS PRO. O que é gamificação? Disponível em: <https://www.ludospro.com.br/blog/o-que-e-gamificacao>. Acesso em: 07 set. 2024.

Jaques, Patrícia; Coelho, Jorge Arthur Peçanha de Miranda; Marques, Leonardo B.; Smiderle, Rodrigo. "The impact of gamification on students' learning, engagement and behavior based on their personality traits." Smart Learning Environments, v. 7, n. 1, dez. 2020.

JOSÉ, Eduardo. (2022). *Porque você deve usar Spring Boot*. Disponível em: <https://medium.com/@duduxss3/porque-voc%C3%AA-deve-usar-spring-boot-a22f75da0955>. Acesso em 07 jan. 2024.

MAPA MENTAL. Wikipédia, a enciclopédia livre. Disponível em https://pt.wikipedia.org/wiki/Mapa_mental. Acesso em 01 set. 2024.

MIMO. Aplicativo Mimo. Disponível em <https://mimo.org/>. Acesso em 17 jun. 2023.

NEVES, Vinicius (2024) TypeScript: o que é, diferenças para o JS e como começar a aprender. Alura. Disponível em: [TypeScript: o que é, diferenças para o JS e como aprender | Alura](#). Acesso em 29 mar. 2024.

NOGUEIRA, D. L. Ferramentas automatizadas para apoio ao projeto estruturado: uma aplicação do diagrama de entidade-relacionamento. 1998. Tese (Doutorado em Ciências em Engenharia de Sistemas e Computação) – Universidade Federal do Rio de Janeiro, Rio de Janeiro, RS.

PRESSMAN, R. S. Engenharia de Software: uma abordagem profissional. 8. ed. Porto Alegre: AMGH, 2016.

Project Management Institute. (2013). [A Guide to the Project Management Body of Knowledge \(PMBOK® Guide\) – Fifth Edition1](#).

Project Management Institute. **A Guide to the Project Management Body of Knowledge (PMBOK® Guide)**, 6º. Guia PMBOK®, PA: Project Management Institute, 2017.

SOLOLEARN. Aplicativo SoloLearn. Disponível em: <https://www.sololearn.com/>. Acesso em: 13 jun. 2023.

SOMMERVILLE, I. Engenharia de Software. 9. ed. São Paulo: Pearson Addison Wesley, 2011.

Visual Studio Code. Wikipédia, a enciclopédia livre. Disponível em https://pt.wikipedia.org/wiki/Visual_Studio_Code. Acesso em 19 fev. 2024.

XU, Wei; Liu, Ziqing; Ren, Jiaopin; The Impact of Educational Games on Learning Outcomes: Evidence From a Meta-Analysis, International Journal of Game-Based Learning, v. 14, n. 1, jan. 2021, p. 1-25.