



DEPARTAMENTO DE CIÊNCIAS E TECNOLOGIAS
MESTRADO EM ENGENHARIA INFORMÁTICA E DE TELECOMUNICAÇÕES
UNIVERSIDADE AUTÓNOMA DE LISBOA
“LUÍS DE CAMÕES”

**DESENVOLVIMENTO DE UMA APLICAÇÃO WEB DE GESTÃO DE
CURSOS ONLINE**

Dissertação para a obtenção do grau de Mestre em Engenharia Informática e
Telecomunicações

Autora: Angélica Ribeiro da Silva

Orientador: Doutor Laercio Cruvinel

Número da candidata: 30003699

Setembro de 2024

Lisboa

Agradecimentos

Em primeiro lugar, quero agradecer ao Omnipresente Deus pela sua infinita bondade. Aos meus pais, pelo carinho e pelo apoio constante durante o meu percurso acadêmico, sem os quais este curso não seria realizado.

Agradeço ao meu orientador Doutor Laercio Cruvinel pela sua dedicação e apoio ao longo deste projeto.

A todos que, direta ou indiretamente, fizeram parte da minha formação, o meu muito obrigado.

Resumo

Este trabalho apresenta o desenvolvimento da ***LearnMug***, uma aplicação *Web* para a gestão de cursos online com ênfase em *Web Responsivo*. Em resposta ao aumento da demanda por educação à distância, a *LearnMug* visa oferecer uma plataforma acessível e adaptável a diferentes dispositivos, proporcionando uma experiência de aprendizagem otimizada. A aplicação inclui funcionalidades robustas, como um sistema de recomendação baseado em preferências dos utilizadores, que utiliza técnicas de aprendizagem de máquina para sugerir cursos personalizados. Além disso, a plataforma integra uma interface intuitiva e recursos interativos, garantindo uma navegação fluida em qualquer dispositivo. A relevância desta ferramenta para o contexto educacional online é destacada, com especial atenção à sua contribuição para a acessibilidade e à necessidade de atualizações futuras para aprimorar a experiência de aprendizagem.

Palavras-chave: *Web Responsivo*, Educação Online, Gestão de Curso, Sistema de Recomendações.

Abstract

This work presents the development of ***LearnMug***, a *Web* application for the management of online courses with an emphasis on *Responsive Web*. In response to the increased demand for distance education, *LearnMug* aims to offer a platform that is accessible and adaptable to different devices, providing an optimized learning experience. The app includes robust features, such as a recommendation system based on user preferences, which uses machine learning techniques to suggest personalized courses. In addition, the platform integrates an intuitive interface and interactive features, ensuring fluid navigation on any device. The relevance of this tool to the online educational context is highlighted, with special attention to its contribution to accessibility and the need for future updates to enhance the learning experience.

Keywords: Responsive Web, Online Education, Course Management, Recommendation System.

Índice geral

1	Introdução	9
1.1	Contextualização	9
1.2	Motivação	10
1.3	Objetivos	10
1.4	Metodologia	10
1.5	Organização do documento.....	11
2	Revisão da Literatura	12
2.1	Breve Histórico do Desenvolvimento do <i>Web Responsivo</i>	12
2.2	Benefícios do <i>Web</i> responsivo para aplicações <i>Web</i>	13
2.3	Modelos e processo de desenvolvimento de <i>software</i>	14
2.4	Avaliação das soluções atuais e identificação de desafios	15
2.5	Abordagens Inovadoras para o Aprimoramento da Experiência de Aprendizagem	16
3	Sistemas de Recomendações	18
3.1	Sistema de Recomendação com filtragem Colaborativa	19
3.2	Sistemas de Recomendação baseados em conteúdo.....	21
4	Recursos de aprendizagem <i>online</i>	23
4.1	Modelos de aprendizagem <i>online</i>	23
4.2	Ferramentas de Aprendizagem <i>Online</i>	24
5	Conceitualização e Contextualização do Projeto.....	27
5.1	<i>Frameworks front-end</i>	27
5.2	<i>Frameworks back-end</i>	28
5.3	Base de Dados	32
5.4	<i>Firebase</i>	33
6	Implementação do projeto.....	34

6.1 Requisitos do sistema	35
6.2 Diagrama de Interação.....	36
6.3 Desenvolvimento.....	46
6.4 Interfaces do Sistema.....	48
6.4.1 Registo de Utilizadores.....	50
6.4.2 Categorias e cursos	53
6.4.3 Ecrã Principal	54
6.5 Recomendações	58
6.6 Medidas de segurança na aplicação.....	59
6.7 Proteção de dados sensíveis dos utilizadores	62
6.8 Desafios e soluções no desenvolvimento da aplicação <i>LearnMug</i>	63
6.9 Análise do <i>Feedback</i> dos Alunos	64
7 Conclusões	66
7.1 Limitações do projeto	66
7.2 Trabalho futuro.....	67
7.3 Considerações Finais	68
Referências	70

Índice de figuras

Figura 1: Ciclo de vida de <i>software</i>	14
Figura 2: Processo de recomendação	19
Figura 3: Sistema de recomendações filtragem colaborativa	19
Figura 4: Matriz binária frequência x termo	21
Figura 5: Modelo tradicional de gestão de projetos	35
Figura 6: Diagrama caso de uso do <i>site</i>	38
Figura 7: Diagrama caso de uso estudante autenticado	41
Figura 8: Diagrama caso de uso administrador.....	41
Figura 9: Método <i>Atomic Web</i>	48
Figura 10: API do projeto	50
Figura 11: Registo e <i>login</i>	51
Figura 12: Formulário do docente	52
Figura 13: Página dos cursos	53
Figura 14: Área do estudante	53
Figura 15: Área administrativa das categorias e cursos	54
Figura 16: Ecrã principal	55
Figura 17: Resultado da pesquisa não encontrado	56
Figura 18: Resultado da pesquisa da procura por categoria ou curso	56
Figura 19: Página fórum	57
Figura 20: Página <i>Webinars</i>	58
Figura 21: Cursos recomendados	59
Figura 22: Análise do feedback dos alunos	65

Índice de tabelas

Tabela 1: Vantagens e desvantagens dos sistemas de recomendação baseados na filtragem colaborativa.....	22
Tabela 2: Vantagens e desvantagens dos sistemas de recomendação baseada em conteúdo ...	23
Tabela 3: Modelos de aprendizagem <i>online</i>	25
Tabela 4: Requisitos funcionais	36
Tabela 5: Requisitos não funcionais.....	37
Tabela 6: Descrição caso de uso “Registo”	40
Tabela 7: Descrição caso de uso “Registo docente”	41
Tabela 8: Descrição caso de uso “ <i>Login</i> ”	42
Tabela 9: Descrição caso de uso “Gerir Categorias e Cursos”	43
Tabela 10: Descrição caso de uso “Acesso aos materiais do curso”	44
Tabela 11: Descrição caso de uso “Criar <i>Flashcards</i> ”	45
Tabela 12: Descrição caso de uso “Criar fórum”	46
Tabela 13: Descrição caso de uso “Gerir eventos”	47
Tabela 14: Implementação de medidas de segurança na aplicação <i>Web</i>	60

1 Introdução

O avanço da tecnologia digital transformou profundamente o cenário educacional, promovendo novas formas de interação e acesso ao conhecimento. Neste contexto, o desenvolvimento de plataformas de gestão de cursos online se tornou vital para atender à crescente demanda por flexibilidade e acessibilidade na aprendizagem a distância. Este capítulo tem como objetivo contextualizar o desenvolvimento da aplicação *LearnMug*, uma plataforma *Web* voltada para a gestão de cursos com ênfase em *Web Responsivo*. Ao longo desta seção, serão apresentados os antecedentes que justificam o desenvolvimento do projeto, a sua motivação e os principais objetivos, oferecendo uma visão clara dos desafios e soluções propostos para otimizar a experiência de aprendizagem online. Além disso, será destacada a relevância da tecnologia educacional no atual cenário digital.

1.1 Contextualização

Embora a transformação digital seja um tema amplamente debatido atualmente, os conceitos fundamentais de produtos e serviços digitais, assim como o papel das redes sociais, já eram compreendidos desde as décadas de 1990 e 2000. No setor comercial, por exemplo, a publicidade e as campanhas nos meios de comunicação de massa eram consideradas estratégias digitais essenciais para alcançar os consumidores, ainda que as compras continuassem a ser realizadas predominantemente em lojas físicas [1].

Entre 2000 e 2015, o crescimento de dispositivos inteligentes e o surgimento das redes sociais provocaram uma mudança significativa na forma como os consumidores se relacionavam com as empresas [2]. As organizações passaram a reconhecer a possibilidade de se conectar digitalmente com seus clientes de maneira personalizada e em tempo real, alterando radicalmente o panorama das interações comerciais. [3].

Nos últimos anos, a revolução digital transformou de forma profunda a maneira como interagimos, aprendemos e conduzimos negócios. Nesse contexto dinâmico e em constante evolução, a tecnologia da informação tornou-se uma força motriz que impacta diversos setores da sociedade. Entre os campos mais afetados, destaca-se o setor educacional, onde a integração de plataformas online tem criado amplas oportunidades para o ensino a distância, ampliando o acesso ao conhecimento e facilitando a aprendizagem remota.

1.2 Motivação

Embora Angola conte com professores altamente qualificados, ainda persiste uma certa resistência em relação aos cursos online ministrados por docentes angolanos. Diante desse cenário, o presente trabalho propõe o desenvolvimento da aplicação *Web LearnMug*, voltada especificamente para o contexto educacional angolano. A *LearnMug* busca oferecer aos professores locais uma plataforma para divulgar seu conhecimento, ao mesmo tempo que possibilita aos estudantes com limitações financeiras o acesso gratuito a cursos de qualidade.

No atual panorama educacional, a flexibilidade e a acessibilidade tornaram-se fatores essenciais na formulação e implementação de programas de ensino. O desenvolvimento de uma aplicação *Web* para a gestão de cursos online surge, assim, como uma solução prática para esse desafio, além de representar um avanço significativo no uso da tecnologia educacional no país.

1.3 Objetivos

O objetivo geral deste trabalho é desenvolver uma aplicação *Web* voltada para a educação em Angola, que permita aos professores angolanos compartilhar seus conhecimentos, demonstrar seu potencial e oferecer oportunidades de aprendizagem gratuita. Os objetivos específicos são:

- Desenvolver uma aplicação *Web* responsiva, funcional, eficiente e intuitiva;
- Proporcionar uma plataforma que facilite a divulgação do conhecimento pelos professores angolanos;
- Oferecer uma solução acessível para estudantes com limitações financeiras, permitindo o acesso gratuito a cursos;
- Contribuir para o desenvolvimento teórico e prático do ensino em Angola, utilizando tecnologia moderna para aprimorar a experiência educacional.

1.4 Metodologia

A metodologia adotada neste trabalho é composta por três abordagens principais: bibliográfica, descritiva e de campo. A pesquisa bibliográfica envolve a análise aprofundada da literatura, com o objetivo de fundamentar o estudo com materiais relevantes da área. A abordagem de campo foca na compreensão das tecnologias utilizadas, especificamente na construção de um *Web* responsivo e na implementação do sistema de recomendação. A integração dessas duas metodologias proporciona uma visão abrangente sobre o tema,

permitindo avaliar de forma eficaz como os sistemas de recomendação podem auxiliar na personalização da seleção de cursos, aprimorando a experiência do utilizador.

1.5 Organização do documento

Este documento está organizado em sete capítulos, que abordam de forma estruturada e progressiva os diferentes aspectos do desenvolvimento da aplicação *Web LearnMug*.

O primeiro capítulo, intitulado Introdução, apresenta o contexto do trabalho, a motivação para a sua realização, os objetivos gerais e específicos, assim como a metodologia utilizada e a estrutura do documento.

O segundo capítulo oferece uma Revisão da Literatura, analisando os principais conceitos relacionados com o *Web* responsivo, plataformas de ensino online e sistemas de recomendação. Este capítulo fundamenta teoricamente o desenvolvimento da aplicação.

No terceiro capítulo, é feita uma Descrição dos Sistemas de Recomendação, com enfoque nas técnicas utilizadas para personalizar a experiência do utilizador e melhorar a seleção de cursos.

O quarto capítulo explora os Recursos de Aprendizagem Online, abordando as ferramentas e modelos pedagógicos integrados na aplicação, como o *e-learning* e a gamificação, que são cruciais para o sucesso da plataforma.

O quinto capítulo trata da Conceituação e Contextualização do Projeto, descrevendo as tecnologias e *frameworks* utilizados no desenvolvimento da aplicação *Web*, incluindo tanto as ferramentas de *front-end* como de *back-end*.

No sexto capítulo, é detalhado o Processo de Implementação da aplicação, desde a definição dos requisitos até à implementação das funcionalidades principais e à integração do sistema de recomendação.

Por fim, o sétimo capítulo apresenta a Conclusão, onde são discutidos os resultados alcançados, as limitações encontradas e as possíveis direções para trabalhos futuros.

2 Revisão da Literatura

Este capítulo apresenta uma análise abrangente da literatura relacionada aos principais temas que fundamentam o desenvolvimento da aplicação *LearnMug*. Iniciamos com uma revisão dos conceitos fundamentais de *Web* responsivo e sua importância na criação de interfaces *Web* acessíveis em múltiplos dispositivos. Em seguida, exploramos os modelos de aprendizagem online e as ferramentas que têm sido utilizadas para enriquecer a experiência educacional em ambientes virtuais. Adicionalmente, discutimos os sistemas de recomendação e suas aplicações no contexto educacional, bem como as práticas de segurança e proteção de dados essenciais para garantir a privacidade dos utilizadores em plataformas de ensino a distância. Esta revisão fornece um quadro teórico e prático que embasa o desenvolvimento do *LearnMug*, destacando as tendências, desafios e oportunidades existentes no campo da educação online.

2.1 Breve Histórico do Desenvolvimento do Web Responsivo

Desde o início da internet, a necessidade de adaptar interfaces a diferentes dispositivos tem sido uma preocupação constante dos desenvolvedores *Web*. Contudo, foi apenas com o surgimento do conceito de *Web* responsivo que esta questão passou a ser abordada de forma mais abrangente e sistemática. O *Web* responsivo emergiu como uma solução para a crescente diversidade de dispositivos e tamanhos de ecrã utilizados para aceder à *Web*. A discussão em torno deste tema ganhou relevância em 2010, com a publicação do artigo seminal de Ethan Marcotte, intitulado *Responsive Web*. Neste trabalho, Marcotte introduziu uma abordagem inovadora, propondo o desenvolvimento de interfaces *Web* capazes de se adaptar dinamicamente ao tamanho do ecrã do utilizador. Esta proposta revolucionária marcou o início de uma nova era no *Web* de interfaces digitais, onde a flexibilidade e a adaptabilidade se tornaram elementos cruciais no desenvolvimento *Web* [4].

O *Web* responsivo não foi apenas impulsionado pela necessidade de suportar uma variedade crescente de dispositivos, mas também pela alteração nos padrões de consumo de conteúdo online. Com o aumento do uso de dispositivos móveis para aceder à internet, tornou-se fundamental garantir uma experiência consistente e de alta qualidade em todas as plataformas. Esta necessidade de uma abordagem mais inclusiva e acessível ao *Web* de interfaces deu origem a uma série de práticas e técnicas inovadoras no desenvolvimento *Web*. Ao longo do tempo, o *Web* responsivo evoluiu, acompanhando os avanços tecnológicos e as mudanças nas preferências dos utilizadores. Novas técnicas e ferramentas foram criadas para

facilitar a construção de layouts flexíveis e adaptáveis, assegurando uma experiência otimizada em qualquer dispositivo [4].

Atualmente, o *Web* responsivo é amplamente reconhecido como uma prática essencial no desenvolvimento *Web*, e a sua importância tende a aumentar à medida que a diversidade de dispositivos continua a expandir-se. Desde a sua conceção até à sua adoção como padrão da indústria, o *Web* responsivo tem desempenhado um papel fundamental na criação de experiências de utilizador mais acessíveis, consistentes e satisfatórias em todo o espectro de dispositivos digitais [4].

2.2 Benefícios do *Web* responsivo para aplicações *Web*

O *Web* responsivo é uma abordagem essencial no desenvolvimento de aplicações *Web*, especialmente considerando a ampla diversidade de dispositivos e tamanhos de ecrã utilizados pelos utilizadores nos dias de hoje. A comunicação e a mobilidade estão cada vez mais interligadas, o que ressalta a necessidade de adaptar os conteúdos para dispositivos móveis. Nesse contexto, o *Web* responsivo surge como uma solução eficaz para garantir uma experiência de utilizador consistente e de qualidade, independentemente do dispositivo utilizado [4].

Um dos principais benefícios do *Web* responsivo é a melhoria da acessibilidade, ao ajustar o *layout* e os elementos da interface com base no tamanho e nas capacidades de cada dispositivo, o *Web* responsivo torna as aplicações *Web* mais inclusivas, permitindo que mais utilizadores possam aceder ao conteúdo de forma eficaz [5].

Além da acessibilidade, o *Web* responsivo também contribui significativamente para a usabilidade e para a experiência global do utilizador. Ao fornecer uma interface otimizada para vários dispositivos, permite que os utilizadores naveguem pela aplicação com facilidade e eficiência, independentemente do equipamento que estão a utilizar. Esta consistência na experiência melhora a satisfação do utilizador e fomenta a fidelidade à aplicação [4], enquanto reduz a taxa de rejeição e aumenta o envolvimento.

Outro benefício importante é a economia de tempo e recursos proporcionada por esta abordagem. Em vez de criar e manter várias versões da mesma aplicação para diferentes dispositivos, o *Web* responsivo possibilita o desenvolvimento de uma única base de código que se adapta dinamicamente a diferentes contextos de visualização. Esta solução simplifica o processo de desenvolvimento e manutenção, reduzindo custos e esforços desnecessários [4].

Em resumo, o *Web* responsivo oferece uma gama de benefícios significativos para o desenvolvimento de aplicações *Web*, desde a melhoria da acessibilidade e usabilidade até à otimização de tempo e recursos. Ao adotar esta abordagem, desenvolvedores e empresas conseguem oferecer uma experiência de utilizador consistente e de alta qualidade em múltiplos dispositivos, atendendo às necessidades e expectativas de um público cada vez mais diversificado e exigente.

2.3 Modelos e processo de desenvolvimento de *software*

O desenvolvimento de *software* para aplicações *Web*, particularmente no contexto da gestão de cursos online com *Web* responsivo, exige a adoção de modelos e processos específicos que garantam a eficiência, qualidade e acessibilidade da solução. A aplicação de metodologias adequadas é, portanto, crucial para o sucesso do projeto, assegurando que todas as fases do desenvolvimento sejam conduzidas de forma sistemática e organizada.

No caso do desenvolvimento da plataforma ***LearnMug***, foi adotado o ciclo de vida de desenvolvimento de *software*, conforme ilustrado na Figura 1. Este modelo, de natureza sistemática e disciplinada, abrange várias etapas essenciais à construção de uma aplicação robusta [6].

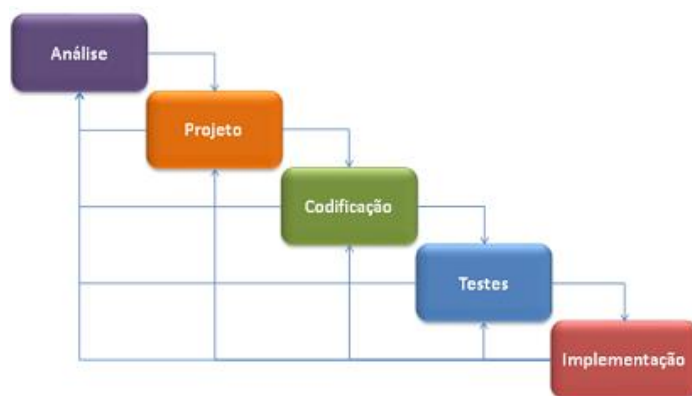


Figura 1: Ciclo de vida de software

Fonte: Pressman (2011)

O processo iniciou-se com uma análise detalhada dos requisitos, que visou identificar as necessidades dos utilizadores e definir as funcionalidades essenciais da aplicação. Com base nessa análise, avançou-se para a fase de projeto, onde foram delineadas a arquitetura da aplicação e as especificações técnicas necessárias para a sua implementação. A fase de implementação envolveu a codificação das funcionalidades planeadas, garantindo que estas

fossem integradas de forma eficaz. Posteriormente, realizaram-se testes rigorosos para assegurar a qualidade e o correto funcionamento do sistema. Por fim, a fase de manutenção foi concebida para garantir que a aplicação se mantém adaptada a eventuais mudanças de requisitos, assegurando o seu desempenho e funcionalidade a longo prazo.

Para alcançar os objetivos de desenvolvimento da aplicação *Web LearnMug*, é fundamental considerar as características e os requisitos específicos do projeto, compreender tanto os paradigmas clássicos como os orientados a objetos no desenvolvimento de *software*, uma vez que estes influenciam directamente as decisões de projecto e implementação, com impacto significativo na arquitectura e na manutenção do sistema. A adopção de uma abordagem orientada a objetos permite aos desenvolvedores estruturar a aplicação de forma modular e escalável, o que facilita a sua manutenção e evolução ao longo do tempo. Esta estrutura modular promove a reutilização de componentes, a redução de complexidade e a adaptação eficiente a novos requisitos, assegurando assim a sustentabilidade do sistema no futuro [7].

2.4 Avaliação das soluções atuais e identificação de desafios

Para compreender melhor as necessidades dos estudantes angolanos e desenvolver uma plataforma que responda adequadamente a essas exigências, foi realizada uma análise das soluções atualmente disponíveis no mercado. Nesse sentido, foram estudadas duas plataformas de cursos online amplamente utilizadas em Angola: ***Kubeta*** e ***Longuka***.

A ***Kubeta*** é uma plataforma de ensino online que tem como público-alvo os estudantes angolanos, oferecendo cursos em diversas áreas do conhecimento. Embora a sua interface seja intuitiva e fácil de utilizar, a plataforma apresenta um desafio significativo: a falta de opções de personalização da aprendizagem. A ***Kubeta*** não possui um sistema de recomendações robusto que sugira cursos ou conteúdos com base nas preferências e no histórico de interação dos utilizadores. Essa ausência de funcionalidades pode resultar numa experiência de aprendizagem menos personalizada e comprometida, limitando a capacidade dos estudantes de descobrir novos conteúdos relevantes e de adaptar a sua formação às suas necessidades individuais [8].

Por outro lado, a ***Longuka***, uma das principais plataformas de cursos online gratuitos em Angola, foi criada em 2019 e viu o seu uso expandido durante a pandemia de *COVID-19*. Embora ofereça uma ampla gama de cursos em diferentes áreas, a plataforma enfrenta desafios relacionados à disponibilização de materiais atualizados. A quantidade de cursos atualizados e a adequação do conteúdo às necessidades específicas dos utilizadores angolanos são

insuficientes. Esta falta de personalização e atualização afeta negativamente a qualidade da experiência de aprendizagem, desmotivando os utilizadores que procuram recursos mais relevantes e adaptados ao seu contexto [9].

O *LearnMug* pretende enfrentar esses desafios ao desenvolver uma aplicação *Web* que se destaca pela personalização do conteúdo, oferecendo uma solução adaptada às necessidades específicas dos professores e estudantes angolanos. A plataforma visa proporcionar uma experiência de aprendizagem mais envolvente e dinâmica, através de recursos como um sistema de recomendação de cursos, que sugere conteúdos com base nas preferências e no histórico de interação dos utilizadores. Combinando esse sistema com um *Web* responsivo, o *LearnMug* busca superar as limitações das soluções atuais, garantindo uma navegação fluida e eficiente para todos os utilizadores, independentemente do dispositivo que utilizem para aceder à plataforma.

2.5 Abordagens Inovadoras para o Aprimoramento da Experiência de Aprendizagem

No desenvolvimento do *LearnMug*, a proposta de aprimorar a experiência de aprendizagem *online* está centrada em várias abordagens inovadoras, adaptadas às necessidades dos estudantes angolanos. Uma das principais funcionalidades da plataforma é a inclusão de *flashcards* personalizados, que permitem aos estudantes criar, editar e rever materiais de estudo de forma eficaz. Os *flashcards* são uma ferramenta poderosa para a memorização e revisão de conceitos essenciais, e a sua integração no *LearnMug* promove uma aprendizagem mais ativa e adaptada às necessidades individuais de cada estudante.

A **responsividade** da plataforma é outro elemento crucial, garantindo uma experiência de aprendizagem consistente em todos os dispositivos. O *Web* responsivo do *LearnMug* assegura que a plataforma funcione de forma otimizada em computadores, *tablets* e *smartphones*. Esta abordagem é especialmente relevante no contexto angolano, onde o acesso a dispositivos móveis é predominante. A capacidade de aceder ao conteúdo educacional de maneira fluída, independentemente do dispositivo, contribui significativamente para a **acessibilidade** e **inclusão digital**, permitindo que um maior número de estudantes beneficie dos recursos da plataforma.

Além disso, o *LearnMug* integra um sistema de recomendações baseado nas preferências dos utilizadores, sugerindo cursos que melhor correspondam aos interesses de cada estudante. Este sistema personaliza a experiência de aprendizagem, melhorando a relevância

dos conteúdos apresentados e facilitando o acesso a recursos ajustados às necessidades específicas dos estudantes

Para otimizar o desempenho e a gestão de dados, o ***LearnMug*** utiliza o ***Firebase*** como solução para armazenamento de vídeos e artigos. O *Firebase* proporciona uma infraestrutura de *back-end* eficiente e escalável, permitindo à plataforma gerir grandes volumes de dados sem comprometer o desempenho. Ao utilizar o *Firebase* para armazenar conteúdos de grande porte, como vídeos, o ***LearnMug*** minimiza a carga sobre o servidor principal, garantindo que o *site* se mantenha leve e ágil. Esta abordagem não só melhora a velocidade de carregamento das páginas, como também proporciona uma experiência de utilização mais fluida, contribuindo para o desempenho geral otimizado da aplicação [10].

3 Sistemas de Recomendações

A introdução dos sistemas de recomendação teve suas raízes com a publicação do trabalho "*Intelligent Information-Sharing System*", que identificou o problema da sobrecarga de informações gerada pela troca de e-mails entre colaboradores de uma empresa. Esse estudo deu origem à primeira divisão dos sistemas de filtragem, organizados em três categorias: cognitiva, social e económica [11].

A **filtragem cognitiva** estava centrada no conteúdo das mensagens e nas necessidades dos utilizadores, sendo usada para filtrar informações relevantes. A **abordagem social** complementava a cognitiva, ao focar-se exclusivamente na forma como os indivíduos se organizavam em comunidade, com base nas suas relações interpessoais. Finalmente, o **filtro económico** avaliava o custo-benefício de cada item, considerando fatores como o tamanho da mensagem e o número de vezes em que o autor era mencionado [12].

Posteriormente, surgiram as **listas de discussão**, que filtravam automaticamente os conteúdos recebidos pelos utilizadores, de acordo com os seus interesses. Esse método evoluiu, permitindo que os utilizadores marcassem determinados tópicos como importantes ou não, conforme os autores, o que formou a base do conceito de **filtragem colaborativa** [13].

O termo "*sistemas de recomendação*" foi formalizado no artigo "*Recommender System*" de Resnick e Varian, onde se definiu o objetivo central desses sistemas: surpreender os utilizadores com novos itens que pudessem despertar o seu interesse. Nesses sistemas, os utilizadores não se conheciam entre si, o que resultava numa colaboração implícita [14].

Os **sistemas de recomendação** têm como objetivo auxiliar os utilizadores na tomada de decisões, sugerindo itens como livros, músicas, artistas, cursos ou outros objetos semelhantes a um conteúdo previamente pesquisado ou adquirido [12]. O funcionamento desses sistemas será abordado ao longo deste capítulo, conforme exemplificado na Figura 2.

Esses sistemas fundamentam-se nos princípios da **personalização** e da **recuperação da informação**, agregando valor à experiência do utilizador, independentemente da técnica utilizada. Neste capítulo, serão discutidos dois tipos principais de sistemas de recomendação: o **sistema baseado em conteúdo** e o **sistema de filtragem colaborativa** [12].

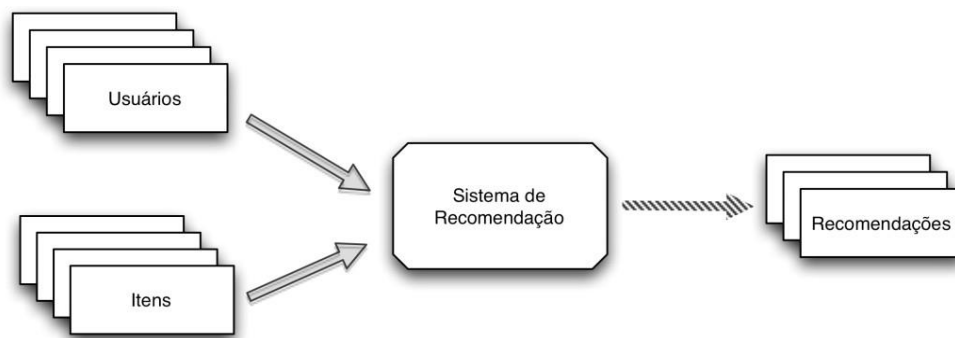


Figura 2: Processo de recomendação.

Fonte: Santana, 2018

3.1 Sistema de Recomendação com filtragem Colaborativa

A **filtragem colaborativa** é uma das técnicas mais comuns em sistemas de recomendação, na qual são utilizadas informações sobre o comportamento dos utilizadores para estimar quais cursos ou conteúdos podem ser de interesse para outros utilizadores com perfis semelhantes. Esse sistema baseia-se na análise de dados como histórico de navegação, avaliações de cursos e preferências de conteúdos, entre outros.

O princípio central deste tipo de recomendação é o relacionamento entre utilizadores e itens. Utilizadores que demonstraram interesse em determinados cursos são agrupados com outros que apresentam comportamentos semelhantes. Dessa forma, as escolhas feitas por utilizadores com perfis similares são recomendadas a outros, que podem ter interesses e preferências equivalentes, conforme ilustrado na Figura 3 [15].

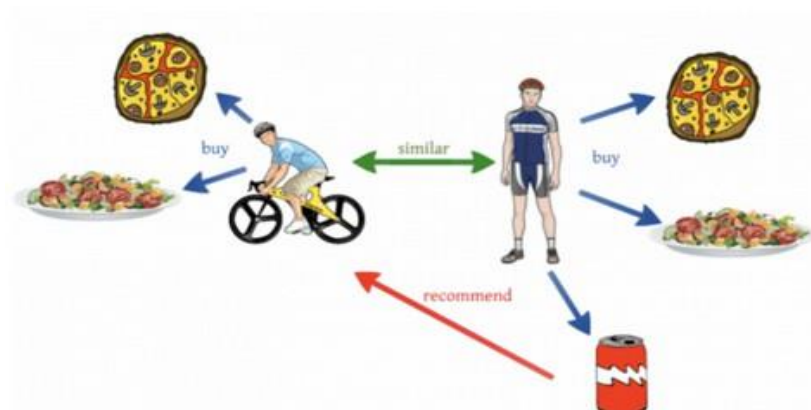


Figura 3: Sistema de recomendação de filtragem colaborativa.

Fonte: Bonin, 2018.

O funcionamento deste tipo de sistema baseia-se numa **matriz de utilizadores x itens**, que armazena dados sobre o comportamento dos utilizadores e a sua interação com cada item. A partir dessa matriz, é feita uma estimativa de quais itens podem ser do interesse de um determinado utilizador. Com base nessas informações, geralmente é gerada uma lista de recomendações de itens relevantes para cada utilizador. Os métodos de funcionamento da filtragem colaborativa podem ser divididos em dois tipos principais: **baseados em vizinhança** e **baseados em modelos** [15].

O **método baseado em vizinhança** utiliza abordagens focadas em utilizadores ou itens, identificando os "vizinhos" mais próximos através da medição de coeficientes de correlação ou distâncias entre utilizadores. No caso dos coeficientes de correlação, são aplicadas métricas como a identificação de utilizadores que realizaram compras ou avaliações semelhantes. Para as distâncias, são usadas medidas como o cosseno ou o coeficiente de Pearson, particularmente nas avaliações de itens. Assim, se dois ou mais utilizadores demonstraram gostos semelhantes no passado, é provável que, no futuro, continuem a ter preferências parecidas.

Já o **método baseado em modelos** envolve a criação de modelos preditivos com base nos comportamentos e preferências dos utilizadores. Estes modelos são desenvolvidos através de algoritmos de aprendizagem automática, tornando a fase de treino mais complexa. Nesses casos, as recomendações são construídas antes mesmo de o utilizador iniciar uma pesquisa, com base nas predições do modelo [15].

Exemplos práticos de sistemas que utilizam filtragem colaborativa incluem o **Facebook**, que sugere amigos, páginas e grupos com base em interações anteriores, o **Spotify**, que cria semanalmente playlists personalizadas com base nas últimas músicas ouvidas, e a **Amazon**, que recomenda produtos semelhantes às compras anteriores do utilizador [15].

Na Tabela 1, são apresentadas de forma resumida as principais vantagens e desvantagens deste tipo de sistema de recomendação.

Tabela 1: Vantagens e desvantagens dos sistemas de recomendação baseados na filtragem colaborativa. Fonte: da Autora.

Vantagens	Desvantagens
Não depende de análise de conteúdo.	Lidar com a grande dimensão da matriz utilizadores x itens e, consequentemente, com a identificação do seu vizinho mais próximo.
Capacidade de filtrar todo o tipo de conteúdo como livros, artistas, notícias, artigos, filmes, entre outros.	
Não se preocupa com o idioma ou o sentido ambíguo das palavras.	Dificuldade de recomendação para utilizadores que possuem poucas ou nenhuma avaliação.

Possível recomendar itens para os utilizadores que não possuam características semelhantes com seu perfil.	Novos itens no sistema que não possuem recomendações dos utilizadores e, por isso, acabam por não ser recomendados.
	Escalabilidade do sistema pois possuem mais utilizadores do que itens.

3.2 Sistemas de Recomendação baseados em conteúdo

Os sistemas de recomendação baseados em conteúdo têm como objetivo analisar as características dos itens que os utilizadores já avaliaram e compará-las com outros itens disponíveis no sistema, a fim de gerar recomendações personalizadas. Nesses sistemas, o primeiro passo é criar uma representação estruturada ou semiestruturada do conteúdo dos itens.

Os **dados estruturados** podem incluir informações como a categoria dos cursos, a data de lançamento, e outras propriedades organizadas em formatos predefinidos. Já os **dados semiestruturados** referem-se a dados textuais, como o nome dos cursos, a descrição ou palavras-chave associadas. Para mapear esse conteúdo, é comum utilizar uma matriz de termos/palavras, onde cada termo associado ao conteúdo é representado numericamente. Por exemplo, uma **matriz binária**, como ilustrado na Figura 4, atribui o valor 1 quando uma palavra está presente num item e o valor 0 quando não está.

Doc 1 - Gatos pretos dão mais sorte que gatos brancos.

Doc 2 - Meus gatos dão muito trabalho.

gatos	pretos	dão	mais	sorte	que	brancos	doce	muito
1	1	1	1	1	1	1	0	0
1	0	1	0	0	0	0	0	1

Figura 4: Matriz binária frequência x termo.

Fonte: Bonin, 2018.

Outra forma de criar essa matriz é atribuindo um peso a cada palavra com base na sua frequência dentro do documento, utilizando métodos como **Document Frequency (DF)** ou **Term Frequency (TF)**. Esses métodos permitem identificar a relevância de cada termo no conjunto de dados, tornando o processo de recomendação mais preciso.

Após a construção da matriz, a próxima etapa é a **redução da dimensionalidade**, uma vez que este tipo de mapeamento pode gerar matrizes com milhares de dimensões, dificultando a análise eficiente. Para lidar com isso, são aplicados métodos de **recuperação de informação**, que reduzem a quantidade de dados a serem processados, sem perder informações relevantes.

Em seguida, algoritmos de **classificação** e **agrupamento** são utilizados para associar itens com conteúdos semelhantes.

Algoritmos de classificação, como o *Naive Bayes*, são especialmente úteis quando há necessidade de associar um item a categorias pré-definidas. À medida que o utilizador fornece mais informações sobre as suas preferências, o sistema torna-se mais preciso. Além disso, esses algoritmos são responsáveis por classificar os itens e determinar o grau de relevância de cada curso, permitindo recomendações mais personalizadas [16].

Por fim, após identificar os interesses dos utilizadores, as recomendações são geradas de forma semelhante a um **filtro de informações**, que é constantemente atualizado e ordenado. Esse filtro contém os principais itens a serem sugeridos ao utilizador, com base numa quantidade predefinida [17]. Na Tabela 2, são apresentadas as principais vantagens e desvantagens deste tipo de sistema de recomendação.

As necessidades específicas da aplicação de gestão de cursos online são amplas e interdependentes. Entre os aspetos cruciais a serem considerados durante o desenvolvimento estão a **acessibilidade**, a **adaptação a diferentes tamanhos de ecrã** e a **otimização de desempenho**. Esses elementos são essenciais para garantir que a aplicação ofereça uma experiência de utilizador fluida e inclusiva, independentemente do dispositivo utilizado. Ao integrar as recomendações de especialistas, é possível desenvolver uma plataforma que responda de forma eficaz às necessidades de uma audiência diversificada, proporcionando uma navegação eficiente e acessível a todos os utilizadores [17].

Tabela 2: Vantagens e desvantagens dos sistemas de recomendação baseada em conteúdo.

Vantagens	Desvantagens
Sensível a mudanças de interesses dos utilizadores.	Algoritmo não consegue aprender detalhes que extrapolam a similaridade textual, por exemplo, Guerra das Estrelas e Jornada nas Estrelas não são, necessariamente, similares.
Necessário conhecer somente o próprio utilizador e não os demais indivíduos.	
Quanto mais o utilizador interage com o sistema, mais o sistema aprende sobre ele, e mais precisos ficam os resultados.	Não é capaz de surpreender o utilizador, pois esta técnica procura, apenas, itens similares e que estejam dentro dos gostos do utilizador.
Um novo item inserido no sistema pode ser recomendado, visto que, não há necessidade de registos de reações de outros utilizadores a eles.	Formas de escrita diferentes podem gerar 2 classificações diferentes.
Como, em geral, há mais utilizadores do que itens, a identificação de itens similares tem um custo computacional menor que a identificação de utilizadores semelhantes.	Alta dimensionalidade nas matrizes - o que pode gerar dificuldade em aplicações comerciais.

Fonte: da Autora

4 Recursos de aprendizagem online

Neste capítulo, serão abordados os principais **recursos de aprendizagem online** utilizados na plataforma *LearnMug*, com foco na sua relevância e aplicação no contexto educacional angolano. Inicialmente, serão analisadas as ferramentas pedagógicas integradas, como o e-learning, a gamificação e os *flashcards*, destacando as suas vantagens na facilitação do processo de ensino-aprendizagem. Em seguida, será discutido o papel da tecnologia na criação de experiências de aprendizagem interativas e personalizadas, enfatizando como esses recursos contribuem para o engajamento dos estudantes e a melhoria da qualidade do ensino a distância. O capítulo também abordará as melhores práticas para a implementação de recursos digitais e os desafios enfrentados na adaptação de materiais de ensino para ambientes online. Por fim, será feita uma análise crítica dos benefícios e limitações desses recursos, com vista a explorar as suas contribuições para a melhoria da experiência de aprendizagem no *LearnMug*.

4.1 Modelos de aprendizagem online

A evolução tecnológica tem provocado mudanças profundas na forma como as pessoas adquirem conhecimento, especialmente no contexto educacional. Nesse cenário, os **modelos de aprendizagem online** ganharam destaque, oferecendo maior flexibilidade e acessibilidade aos estudantes. O desenvolvimento de aplicações Web para a gestão de cursos online, como a *LearnMug*, responde diretamente à necessidade de adaptação a um mundo cada vez mais digital e interconectado. Esses modelos de ensino permitem que os estudantes acessem ao conteúdo de forma autónoma, independentemente de restrições geográficas ou temporais, ampliando as oportunidades de aprendizagem e democratizando o acesso à educação [18].

Conforme ilustrado na Tabela 3, observa-se uma tendência importante nos modelos de aprendizagem online: a capacidade de adaptar o conteúdo e a metodologia de ensino às necessidades individuais dos estudantes. Essa personalização não só melhora a eficácia do processo de aprendizagem, como também aumenta o nível de engajamento dos alunos. Para garantir uma experiência de aprendizagem mais rica e personalizada, a aplicação deve incorporar recursos que permitam a customização do ambiente de ensino, oferecendo aos utilizadores uma experiência ajustada às suas preferências e necessidades específicas.

Tabela 3: Modelos de aprendizagem online

Modelo de Aprendizagem Online	Descrição	Exemplos
Aprendizagem Assíncrona	Os alunos têm acesso ao conteúdo do curso no seu próprio ritmo e conveniência, sem a necessidade de participação em tempo real.	- Fóruns de discussão - Vídeo-aulas gravadas
Aprendizagem Síncrona	Os alunos participam em atividades de aprendizagem em tempo real, interagindo com instrutores e colegas de aula.	- <i>Webinars</i> (aulas ao vivo) - Salas de conversação (<i>chat</i>) em tempo real
Aprendizagem Híbrida	Combina elementos de aprendizagem <i>online</i> e presencial, oferecendo flexibilidade e interação pessoal.	- Sessões presenciais ocasionais combinadas com conteúdo <i>online</i> - Trabalhos em grupo mistos (<i>online</i> e <i>offline</i>)
Aprendizagem Adaptativa	Personaliza o processo de aprendizagem com base nas necessidades individuais do aluno, utilizando tecnologias inteligentes.	- Plataformas que ajustam o conteúdo de acordo com o desempenho do aluno - Sistemas de recomendação de conteúdo personalizado

Fonte: adaptado de Rathfux et al. (2018)

Os modelos de aprendizagem online estão em constante evolução, impulsionados pela crescente convergência entre educação e tecnologia. A aplicação proposta neste trabalho busca incorporar as melhores práticas pedagógicas aliadas às mais recentes inovações tecnológicas, com o objetivo de proporcionar uma experiência de aprendizagem online mais acessível, personalizada e eficaz. Ao integrar essas abordagens, pretende-se atender às necessidades de uma ampla diversidade de utilizadores, garantindo um ambiente educacional que seja inclusivo, flexível e orientado para o sucesso dos estudantes.

4.2 Ferramentas de Aprendizagem Online

As **ferramentas de aprendizagem online** desempenham um papel crucial no desenvolvimento de plataformas de cursos, servindo como pilares para a construção de um ambiente que responda eficazmente às necessidades dos estudantes. No contexto do **LearnMug**, três ferramentas foram especialmente importantes: **e-learning**, **gamificação** e **vídeos/aulas online**. Essas ferramentas não apenas orientam o Web pedagógico da plataforma, mas também enriquecem a experiência de aprendizagem, tornando-a mais envolvente, dinâmica e acessível para um público diversificado.

E-learning: O **e-learning** refere-se à aprendizagem distribuída de forma online, oferecendo um ambiente interativo baseado na Web, que possibilita o envolvimento e a

interação entre professores e alunos em plataformas digitais [19]. Esse modelo de ensino é sustentado por três principais tipos de interação: **aluno-aluno**, **aluno-instrutor** e **aluno-conteúdo**, que são essenciais para a eficácia do processo de ensino. Além disso, o *e-learning* tem sido amplamente adotado globalmente como um método inovador de instrução [19].

O *e-learning* é uma abordagem inovadora que oferece um ambiente de aprendizagem bem projetado, centrado no aluno, interativo e acessível, permitindo que qualquer pessoa, em qualquer lugar e a qualquer momento, tenha acesso à educação. Esta metodologia aproveita os recursos e atributos das tecnologias digitais, em combinação com materiais de aprendizagem adequados, para criar um ambiente aberto, flexível e distribuído. [19].

Gamificação: A **gamificação** tem emergido como uma estratégia promissora para envolver e motivar os utilizadores em diversos contextos, incluindo a educação online. A gamificação é definida como a aplicação de elementos e mecânicas de jogos em ambientes não lúdicos, com o objetivo de promover comprometimento, motivação e aprendizagem. A integração de elementos de jogos numa aplicação de cursos online, como no caso de um curso de **Web Responsivo**, pode amplificar a experiência do utilizador e aumentar a eficácia do processo de aprendizagem. [20].

Um dos maiores desafios enfrentados pelos cursos online é manter os alunos comprometidos e motivados ao longo de todo o processo educativo. A gamificação pode ajudar a superar esse obstáculo, oferecendo recompensas, desafios e *feedback* imediato – fatores essenciais para sustentar o interesse e o engajamento dos alunos. Ao integrar elementos de jogos como pontuações, rankings e conquistas, a aplicação Web pode tornar a experiência de aprendizagem mais dinâmica e interativa, motivando os estudantes a progredirem nos seus estudos [20].

Outro aspeto relevante da gamificação é a sua capacidade de promover a colaboração e a competição saudável entre os alunos, permite ressaltar a interação social é um dos principais fatores que contribuem para o sucesso da gamificação. Ao incorporar elementos sociais, como fóruns de discussão, grupos de estudo e desafios em equipa, a aplicação pode incentivar a troca de conhecimentos e o trabalho em grupo, criando uma comunidade de aprendizagem colaborativa e interativa [20].

Assim, a gamificação surge como uma estratégia eficaz para melhorar uma aplicação Web educacional. Ao integrar elementos e mecânicas de jogos, é possível aumentar o comprometimento, a motivação e a eficácia do processo de aprendizagem, proporcionando aos alunos uma experiência mais envolvente e interativa.

Vídeos e Aulas Online: A integração de vídeos e aulas *online* tornou-se uma prática essencial no desenvolvimento de aplicações *Web*, especialmente no contexto do **Web Responsivo**. A utilização de vídeos como recurso pedagógico é uma estratégia eficaz para transmitir informações de forma visual e interativa. Aliada às aulas *online*, essa abordagem proporciona uma experiência de aprendizagem mais dinâmica e participativa, o que contribui significativamente para a melhor assimilação dos conteúdos pelos alunos. No entanto, é fundamental que sejam considerados aspetos de **acessibilidade**, garantindo que os materiais sejam acessíveis para todos os utilizadores, independentemente das suas capacidades ou condições de acesso [20].

A criação e disponibilização de vídeos e aulas *online* requerem uma infraestrutura tecnológica robusta. Isso inclui a escolha adequada de plataformas de alojamento e reprodução de vídeos, bem como a implementação de ferramentas de interatividade e acompanhamento do progresso dos alunos. Além disso, é crucial otimizar esses recursos para **dispositivos móveis**, visto que um número crescente de utilizadores acede a conteúdos educacionais através de smartphones e tablets [20].

No entanto, a integração de vídeos e aulas *online* traz consigo desafios tanto técnicos quanto pedagógicos. Do ponto de vista técnico, é necessário garantir a **compatibilidade** com diferentes navegadores e dispositivos, bem como assegurar a **qualidade do streaming** e a **segurança dos dados dos utilizadores**. Já no âmbito pedagógico, é essencial desenvolver conteúdos que mantenham os alunos envolvidos e promovam uma **aprendizagem ativa**, utilizando recursos como *quizzes*, fóruns de discussão e atividades práticas para estimular a participação e o entendimento [21].

5 Conceitualização e Contextualização do Projeto

Neste capítulo, serão detalhadas as principais **ferramentas e tecnologias** utilizadas no desenvolvimento da plataforma *LearnMug*, com ênfase nos **frameworks** aplicados tanto no *front-end* como no *back-end*. Serão discutidas a base de dados escolhida para o armazenamento de informações e a utilização do **Swagger** para a criação da API do *site*, proporcionando uma visão clara dos componentes técnicos que sustentam o projeto. Além disso, será apresentada a **IDE** selecionada para o desenvolvimento, o **Visual Studio**, uma ferramenta integrada da Microsoft que suporta diversas linguagens de programação e oferece um ambiente robusto para a construção da aplicação.

5.1 Frameworks front-end

Os **frameworks front-end** desempenham um papel crucial no desenvolvimento de aplicações *Web* modernas, fornecendo estruturas e ferramentas que simplificam a criação de interfaces responsivas e dinâmicas. Essas ferramentas são essenciais para garantir a eficiência, escalabilidade e manutenibilidade do código-fonte, permitindo aos desenvolvedores concentrarem-se mais na lógica de negócios e na experiência do utilizador.

Next.js: Um dos **frameworks** mais populares para o desenvolvimento *front-end* é o **Next.js**, criado pela Vercel. O **Next.js** é uma biblioteca *JavaScript* de código aberto, baseada no *React*, que facilita a construção de aplicações *Web* escaláveis e de alto desempenho. Ele permite a renderização do lado do servidor e a geração de *sites* estáticos, características que tornam as aplicações mais rápidas e otimizadas para *SEO* [22].

Jotai: Para lidar com o estado da aplicação, utilizamos a biblioteca **Jotai**, que adota uma abordagem simples e eficiente para a gestão de estados. A sua filosofia de "menos verbosidade" resulta num código mais limpo e fácil de compreender. A simplicidade do **Jotai**, aliada à ausência de *middlewares* e de conceitos complexos comuns noutras soluções de gestão de estado, proporciona um desempenho superior, especialmente em aplicações *Web* responsivas. Isso é fundamental para garantir uma experiência de utilizador fluida, particularmente em dispositivos móveis e em ambientes com conexões de internet mais lentas [22].

SASS: Para a **estilização** da aplicação, utilizámos o **SASS** (*Syntactically Awesome Stylesheets*), um pré-processador de *CSS* amplamente utilizado no desenvolvimento *Web* moderno [23]. Com uma sintaxe intuitiva e recursos avançados, o **SASS** oferece uma maneira mais eficiente e organizada de escrever folhas de estilo, facilitando a criação de aplicações *Web* responsivas. A abordagem modular que o **SASS** permite, com o uso de variáveis e funções, é

essencial para projetos de grande escala, como sistemas de gestão de conteúdos *online*. Esses recursos ajudam a reduzir a repetição de código, simplificam a manutenção e garantem a consistência dos estilos ao longo do tempo. [24].

5.2 Frameworks back-end

No desenvolvimento de uma aplicação *Web*, o **back-end** é responsável por operar no servidor e processar a lógica da aplicação. No projeto **LearnMug**, utilizámos a plataforma **Node.js**, uma tecnologia de desenvolvimento em *JavaScript* que permite a execução de código do lado do servidor, baseada no motor V8 do *Google Chrome*. A arquitetura assíncrona e orientada a eventos do *Node.js* torna-o especialmente adequado para aplicações *Web* escaláveis e de alto desempenho [25].

Uma das principais vantagens do **Node.js** é o seu extenso ecossistema de pacotes, gerido pelo **NPM (Node Package Manager)**. O **NPM** permite aos desenvolvedores integrar facilmente bibliotecas e módulos externos nos seus projetos, o que aumenta a eficiência no desenvolvimento e facilita a adição de funcionalidades complexas [25]. No contexto da aplicação *LearnMug*, o **NPM** oferece uma vasta gama de módulos que simplificam tarefas como a autenticação de utilizadores, a manipulação de ficheiros e a comunicação com a base de dados, contribuindo para a robustez e funcionalidade da aplicação.

Além disso, o **Node.js** é amplamente utilizado na implementação de **APIs RESTful**, que desempenham um papel fundamental na comunicação entre o *front-end* e o *back-end* de uma aplicação. As **APIs RESTful** permitem que os clientes solicitem e manipulem recursos utilizando operações *HTTP* padrão, como *GET*, *POST*, *PUT* e *DELETE*. No caso da gestão de cursos *online* em Web Responsivo, as **APIs RESTful** desenvolvidas com *Node.js* facilitam a integração entre os diferentes componentes da plataforma, garantindo uma interação fluida e eficiente entre o utilizador e a aplicação [26].

Express: Desenvolvido por TJ Holowaychuk, o **Express** é amplamente reconhecido pela sua simplicidade e poder. A sua arquitetura minimalista permite aos desenvolvedores criar aplicações *Web* de forma rápida e eficiente, sem sacrificar a flexibilidade ou escalabilidade do sistema [27]. Baseado numa abordagem de **middleware**, o **Express** oferece um mecanismo robusto para o tratamento de requisições *HTTP*, permitindo a adição de funcionalidades extras através da integração de *middleware* de terceiros ou personalizados [27].

No contexto de uma aplicação *Web* como a **LearnMug**, o **Express** disponibiliza ferramentas e recursos fundamentais para lidar com requisitos dinâmicos de conteúdo e interação. A sua capacidade de definir rotas específicas para diferentes tipos de requisições

HTTP permite a implementação de funcionalidades como o registo de utilizadores, gestão de cursos, e interações entre alunos e professores [27]. Além disso, a integração com **template engines** como o *EJS (Embedded JavaScript)* ou *Pug* facilita a renderização de páginas *Web* dinâmicas, adaptáveis a diferentes dispositivos e tamanhos de ecrã, assegurando uma experiência de utilizador consistente e responsiva

Axios: é uma biblioteca *JavaScript* amplamente utilizada para realizar requisições *HTTP* tanto em navegadores como no *Node.js*. Ela oferece uma interface simplificada e intuitiva para realizar operações assíncronas, como recuperar dados de *APIs Web* e enviar informações do cliente para o servidor. Desenvolvida por Matt Zabriskie, o **Axios** tornou-se uma escolha popular devido à sua facilidade de uso e aos seus recursos robustos de manipulação de requisições e respostas [28].

Além da simplicidade, o *Axios* suporta várias funcionalidades avançadas, como **cancelamento de requisições**, **autenticação HTTP**, **manipulação de erros** e a **conversão automática de dados JSON**, o que facilita o desenvolvimento de aplicações *Web* mais robustas e seguras, recursos são especialmente relevantes em contextos onde a confiabilidade e a segurança das comunicações de rede são essenciais, como em sistemas de gestão de cursos *online*, nos quais a integridade dos dados dos utilizadores é crucial para o bom funcionamento da plataforma [29].

Dompurify: No desenvolvimento de aplicações *Web* responsivas, a **segurança dos dados** e a **integridade do sistema** são aspetos fundamentais. Para mitigar vulnerabilidades, é essencial utilizar ferramentas adequadas que protejam as aplicações contra ameaças comuns. Uma dessas ferramentas é o **Dompurify**, uma biblioteca *JavaScript* projetada para proteger contra-ataques **XSS (Cross-Site Scripting)**, uma das principais ameaças à segurança na *Web* moderna [30].

Embora o **Dompurify** seja altamente eficaz na prevenção de ataques *XSS*, é importante ressaltar que ele não oferece uma solução completa para todos os desafios de segurança enfrentados por aplicações *Web*. Outras vulnerabilidades, como **injeção de SQL** e **CSRF (Cross-Site Request Forgery)**, também exigem medidas de proteção adicionais. Por isso, o uso do **Dompurify** deve ser complementado por práticas de **codificação segura**, testes de segurança regulares e a adoção de outras ferramentas e técnicas recomendadas no desenvolvimento de aplicações seguras.

JSON Web Tokens: *JSON Web Tokens (JWT)* é um método compacto e seguro para transmitir informações entre partes, representado como um objeto *JSON*, e é amplamente utilizado na autenticação de aplicações *Web*. Um *token JWT* é composto por três partes: o

cabeçalho (header), o **payload** e a **assinatura**. O cabeçalho especifica o tipo de *token* e o algoritmo de assinatura; o *payload* contém os dados do utilizador ou outras informações necessárias para a autenticação, enquanto a assinatura garante a **integridade** e **autenticidade** do *token* [30].

A utilização de *JWT* oferece várias vantagens. Uma delas é a **autossuficiência dos tokens**, o que elimina a necessidade de consultar o servidor de autenticação a cada requisição, resultando num desempenho mais eficiente e reduzindo a carga sobre o servidor. Além disso, os *tokens JWT* são **compactos** e podem ser facilmente transmitidos através de *URLs*, cabeçalhos *HTTP* ou como parte do corpo de uma requisição, tornando-os ideais para a comunicação entre cliente e servidor em aplicações *Web* [30].

No contexto de uma aplicação *Web*, os *JWTs* melhoram a experiência do utilizador, permitindo **sessões prolongadas** sem a necessidade de múltiplos logins, e protegem informações confidenciais dos utilizadores, como dados pessoais e registos de progresso nos cursos. Isso garante que apenas utilizadores autorizados tenham acesso a essas informações sensíveis, aumentando a segurança da plataforma [30].

Cookie-Parser: O *cookie-parser* é um *middleware* amplamente utilizado em aplicações Node.js, responsável por analisar os cookies enviados pelo cliente ao servidor e disponibilizá-los no objeto de requisição de forma conveniente para manipulação. A sua integração é essencial em sistemas que requerem **gestão de sessões** e **autenticação**, fornecendo uma maneira eficiente de lidar com informações armazenadas localmente no navegador do utilizador [30].

Com o *cookie-parser*, é possível extrair facilmente dados dos cookies, como *tokens* de autenticação ou preferências do utilizador, e utilizá-los no fluxo da aplicação. Este processo simplificado de manipulação de cookies aumenta a eficiência no desenvolvimento de aplicações *Web*, especialmente em cenários que exigem **gestão de estado** e **personalização da experiência do utilizador**. Além disso, o *cookie-parser* oferece opções avançadas para configurar cookies, como **tempo de expiração**, **domínio** e **segurança**, proporcionando um controlo preciso sobre o comportamento e a persistência desses dados no navegador do cliente [30].

CORS: *Cross-Origin Resource Sharing (CORS)* é uma política de segurança implementada pelos navegadores para controlar as interações entre diferentes origens de recursos numa aplicação *Web*. Compreender o *CORS* é essencial no desenvolvimento de aplicações *Web*, especialmente quando há interações entre diferentes servidores e domínios, para garantir tanto a **integridade** quanto a **segurança** da aplicação.

As **origens de recursos** referem-se aos locais de onde são carregados elementos como scripts, estilos e dados. A uma origem de recursos é definida pelo protocolo, domínio e porta de onde um recurso é servido. O *CORS* impõe restrições sobre como um navegador permite que um documento de uma origem acesse recursos de outra, sendo uma medida crucial para mitigar vulnerabilidades como ataques de **Cross-Site Scripting (XSS)**. A implementação do *CORS* requer a configuração correta do servidor para incluir os cabeçalhos *CORS* apropriados nas respostas *HTTP* [30].

A importância de configurar corretamente os principais **cabeçalhos CORS**, como "*Access-Control-Allow-Origin*", "*Access-Control-Allow-Methods*" e "*Access-Control-Allow-Headers*", para controlar quais origens, métodos *HTTP* e cabeçalhos são permitidos nas solicitações de recursos cruzados. Configurações inadequadas podem expor a aplicação a riscos de segurança, enquanto uma implementação correta garante que apenas origens autorizadas possam interagir com os recursos [30].

Swagger: No desenvolvimento de uma aplicação *Web* de gestão de cursos *online* com **Web Responsivo**, a documentação e os testes das *APIs* desempenham um papel fundamental. Para facilitar esse processo, uma das ferramentas mais destacadas é o **Swagger**, uma estrutura que permite a descrição, documentação e visualização de *APIs RESTful*, proporcionando uma maneira eficiente de interagir com esses serviços [23].

Uma abordagem comum para integrar o *Swagger* num projeto **Node.js** é utilizar pacotes como *swagger-autogen*, *swagger-jsdoc* e *swagger-ui-express*. O *swagger-autogen* simplifica a geração automática de especificações *Swagger* a partir do código-fonte, eliminando a necessidade de criar manualmente arquivos *YAML* ou *JSON* de especificação. Com ele, é possível gerar rapidamente uma documentação *Swagger* baseada nas definições das rotas de um aplicativo *Node.js*.

Por outro lado, o *swagger-jsdoc* permite que os desenvolvedores documentem a *API* diretamente nos comentários do código-fonte, usando a sintaxe **JSDoc**. Essa abordagem integrada facilita a manutenção da documentação, já que as alterações nas rotas são automaticamente refletidas na especificação *Swagger* [23]. Essa integração direta com o código simplifica o processo de manter a documentação sempre atualizada e precisa.

Além disso, o *swagger-ui-express* é uma biblioteca que incorpora o **Swagger UI** num aplicativo *Node.js*, fornecendo uma interface interativa para explorar e testar as *APIs* documentadas. Essa interface intuitiva facilita a compreensão das funcionalidades da *API* e acelera o processo de teste, permitindo que os desenvolvedores verifiquem rapidamente a integração e o comportamento dos *endpoints*.

A importância de uma documentação acessível e precisa para garantir a usabilidade de aplicações *Web*, especialmente para utilizadores com deficiências visuais. A integração do *Swagger* no desenvolvimento de uma aplicação *Web* de gestão de cursos *online* com *Web* Responsivo contribui significativamente para esse objetivo, fornecendo uma documentação clara, acessível e fácil de usar [23].

5.3 Base de Dados

A escolha da plataforma de gestão de dados é um fator crítico no desenvolvimento de uma aplicação *Web*. No projeto *LearnMug*, optou-se por utilizar o *HeidiSQL* como ferramenta de administração de bases de dados *SQL*. Esta solução de código aberto é amplamente utilizada por desenvolvedores *Web* para gerir bases de dados *MySQL*, *MariaDB*, *PostgreSQL* e *Microsoft SQL Server*. Desenvolvido por Ansgar Becker, o *HeidiSQL* oferece uma interface gráfica intuitiva e robusta que facilita a execução de consultas, a visualização e modificação de dados, bem como a gestão da estrutura do banco de dados [23].

O *MySQL* é uma das ferramentas mais utilizadas no desenvolvimento de aplicações *Web*, sendo um sistema de gestão de base de dados relacional confiável e de código aberto. A sua popularidade deve-se à sua eficiência, confiabilidade e ampla compatibilidade com várias linguagens de programação e sistemas operacionais. A integração de sistemas distribuídos pode beneficiar da escalabilidade e confiabilidade que o *MySQL* oferece. dados [31].

Uma das principais características do *MySQL* é a sua capacidade de lidar com grandes volumes de dados de maneira eficiente, garantindo a integridade e consistência das informações armazenadas. Essa capacidade é essencial em aplicações de gestão de cursos *online*, onde é necessário armazenar e gerir uma grande quantidade de dados, como informações de cursos, estudantes, conteúdos e interações [31].

No contexto de uma aplicação *Web* de gestão de cursos *online* com *Web* responsivo, a flexibilidade do *MySQL* em termos de tipos de dados e consultas é um ponto crucial. A importância do *Web* responsivo na arquitetura de informação, e o *MySQL* permite aos desenvolvedores armazenar e manipular uma vasta gama de dados, desde textos simples até imagens e vídeos, de maneira eficiente e integrada [32].

Assim, o *MySQL* destaca-se como uma escolha sólida para o desenvolvimento de aplicações *Web*, oferecendo uma combinação de desempenho, confiabilidade e flexibilidade que satisfaz as exigências deste tipo de aplicação. A sua capacidade de lidar com grandes volumes de dados, suportar transações *ACID* e oferecer recursos avançados de otimização torna-o uma opção amplamente preferida pelos desenvolvedores de software.

5.4 *Firestore*

O *Firestore* é uma plataforma de desenvolvimento de aplicativos móveis e *Web*, criada pela Google, que oferece uma ampla gama de serviços, incluindo armazenamento na nuvem, base de dados em tempo real, autenticação, entre outros. Uma de suas ferramentas principais, o *Firestore Storage*, permite o armazenamento eficiente e seguro de arquivos grandes, como vídeos. Isso facilita a gestão de conteúdos multimídia e oferece uma solução escalável para projetos que envolvem grandes volumes de dados [10].

No projeto *LearnMug*, optámos por utilizar o *Firestore* para aliviar a carga da base de dados principal, mantendo-a mais leve e eficiente. Com o *Firestore Storage*, os vídeos e arquivos enviados pelos docentes são armazenados diretamente na nuvem, o que contribui para o melhor desempenho da aplicação. Além disso, o *Firestore* oferece funcionalidades integradas de **segurança e controlo de acesso**, garantindo que apenas utilizadores autorizados possam aceder aos conteúdos.

Essa abordagem não só melhora a **experiência do utilizador**, proporcionando uma navegação mais fluida e eficiente, como também facilita a **escalabilidade** da aplicação, à medida que o número de vídeos e utilizadores cresce [10].

6 Implementação do projeto

Este capítulo apresenta a implementação da solução proposta, que consiste no desenvolvimento do site **LearnMug**, uma plataforma de cursos *online* onde os cursos são disponibilizados gratuitamente. Os utilizadores registados receberão recomendações de cursos com base nas suas preferências e interações na plataforma. Por outro lado, os docentes interessados em partilhar os seus cursos poderão manifestar interesse através de um formulário específico.

O processo de adição de cursos é gerido pelo administrador, que, após avaliar o conteúdo submetido pelos docentes, decide pela inclusão do curso no *site*. Esta abordagem centralizada garante a **qualidade** e **relevância** dos conteúdos oferecidos aos utilizadores, assegurando que todos os cursos passam por uma análise criteriosa antes de serem disponibilizados.

O desenvolvimento da aplicação seguiu o modelo tradicional de **gestão de projetos**, conforme ilustrado na Figura 5, em que cada etapa é realizada sequencialmente. Inicialmente, são definidos os requisitos e o escopo do projeto. Em seguida, realiza-se a análise dos requisitos, a criação da documentação e o mapeamento da base de dados. Com base nessas definições, a terceira etapa envolve a construção do sistema, que, posteriormente, passa por testes rigorosos antes da sua implementação final.

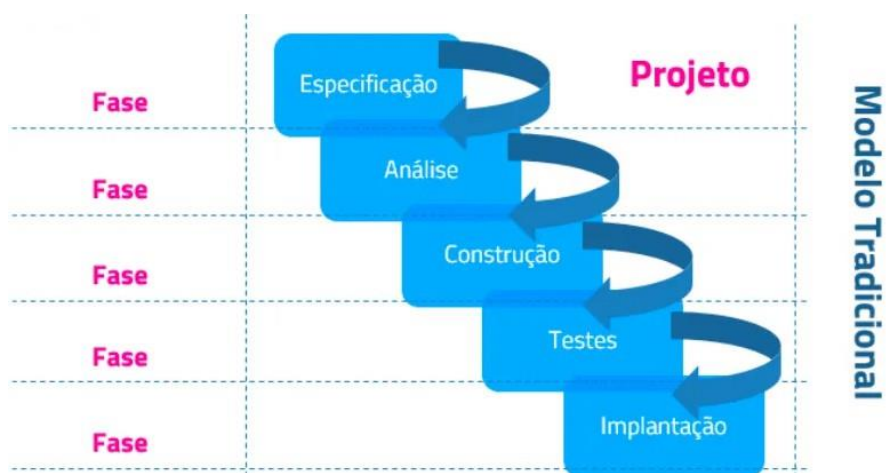


Figura 5: Modelo tradicional de gestão de projetos.

Fonte: Diário da Informática, s.d.

6.1 Requisitos do sistema

O primeiro passo no desenvolvimento de qualquer sistema é a **definição dos requisitos**, que podem ser classificados em **requisitos funcionais** e **requisitos não funcionais**. Os requisitos funcionais, apresentados na Tabela 4, descrevem as funcionalidades específicas que o sistema deve desempenhar, ou seja, as ações e operações essenciais para o seu funcionamento. Por outro lado, os requisitos não funcionais, listados na Tabela 5, referem-se às características de desempenho e qualidade do sistema, como segurança, escalabilidade e eficiência.

Tabela 4: Requisitos funcionais.

ID	Descrição	Prioridade
RF01	As categorias de cursos só podem ser registadas por utilizadores do tipo administrador.	Alta
RF02	Os utilizadores podem ser do tipo administrador ou estudante.	Alta
RF03	Todos os cursos disponibilizados precisam de ser aprovados pelo administrador.	Alta
RF04	Estudantes podem gerir o seu perfil e os seus dados pessoais.	Média
RF05	Estudantes receberão recomendações de cursos baseadas nas suas preferências.	Alta
RF06	Estudantes podem procurar cursos por categoria e por nome.	Alta
RF07	Para ter acesso aos cursos, é necessário <i>login</i> e senha válidos.	Alta
RF08	Utilizadores novos podem registar-se utilizando dados válidos e que ainda não tenham sido inseridos no sistema.	Alta
RF09	Os cursos mais procurados devem ser mostrados ao utilizador.	Média
RF10	Ao fazer <i>login</i> , o utilizador deve ser direcionado para a sua área interna.	Alta
RF11	Docentes podem enviar um formulário de interesse para oferecer os seus cursos.	Alta
RF12	O conteúdo dos cursos deve ser disponibilizado sem custo para os estudantes.	Alta
RF13	Estudantes podem criar um <i>flashcard</i> para revisão e auxiliar na aprendizagem.	Alta

Fonte: da Autora.

Tabela 5: Requisitos não funcionais.

ID	Descrição	Prioridade
RNF01	A senha do utilizador deve ser armazenada e criptografada na base de dados.	Alta
RNF02	O sistema deve ser intuitivo e de fácil uso.	Alta
RNF03	As cores do <i>site</i> devem ser agradáveis e harmoniosas.	Alta
RNF04	O sistema deve ter alta disponibilidade e desempenho para suportar múltiplos acessos simultâneos.	Alta
RNF05	O <i>site</i> deve ser responsivo e acessível em diferentes dispositivos e tamanhos de ecrã.	Alta
RNF06	A API deve ser documentada utilizando <i>Swagger</i> .	Alta
RNF07	O sistema deve garantir a proteção dos dados pessoais dos utilizadores.	Alta

Fonte: da Autora.

6.2 Diagrama de Interação

Com base nos requisitos definidos, é possível identificar dois principais tipos de utilizadores que interagem com o sistema: o **administrador** e o **estudante**. A interação de cada um desses utilizadores com o sistema é ilustrada nos diagramas de caso de uso, apresentados nas Figuras 6, 7 e 8, que descrevem as diferentes funcionalidades e ações disponíveis para cada perfil, de acordo com os cenários de uso estabelecidos.

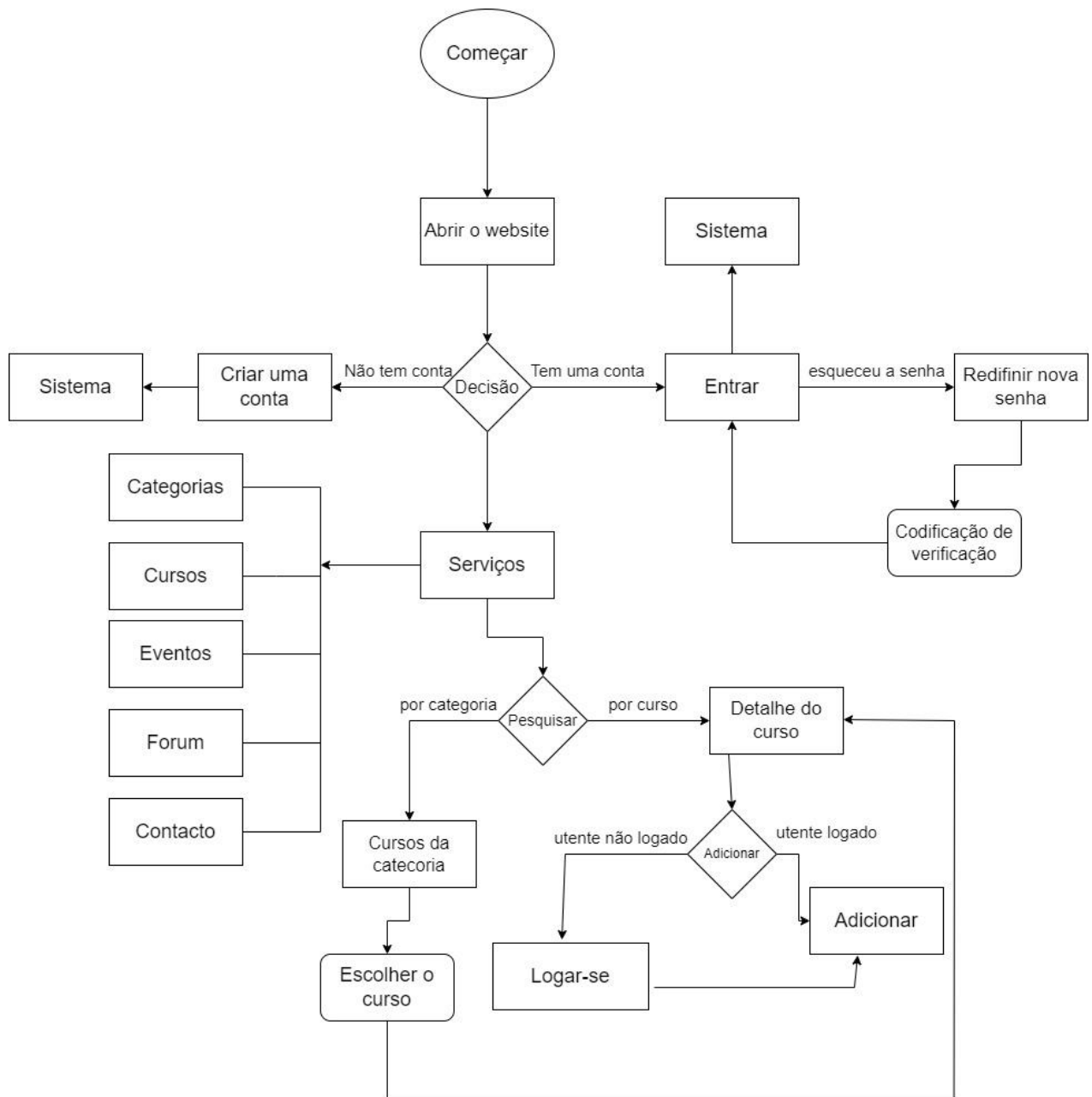


Figura 6: Diagrama de Interação do site

Fonte: Da Autora

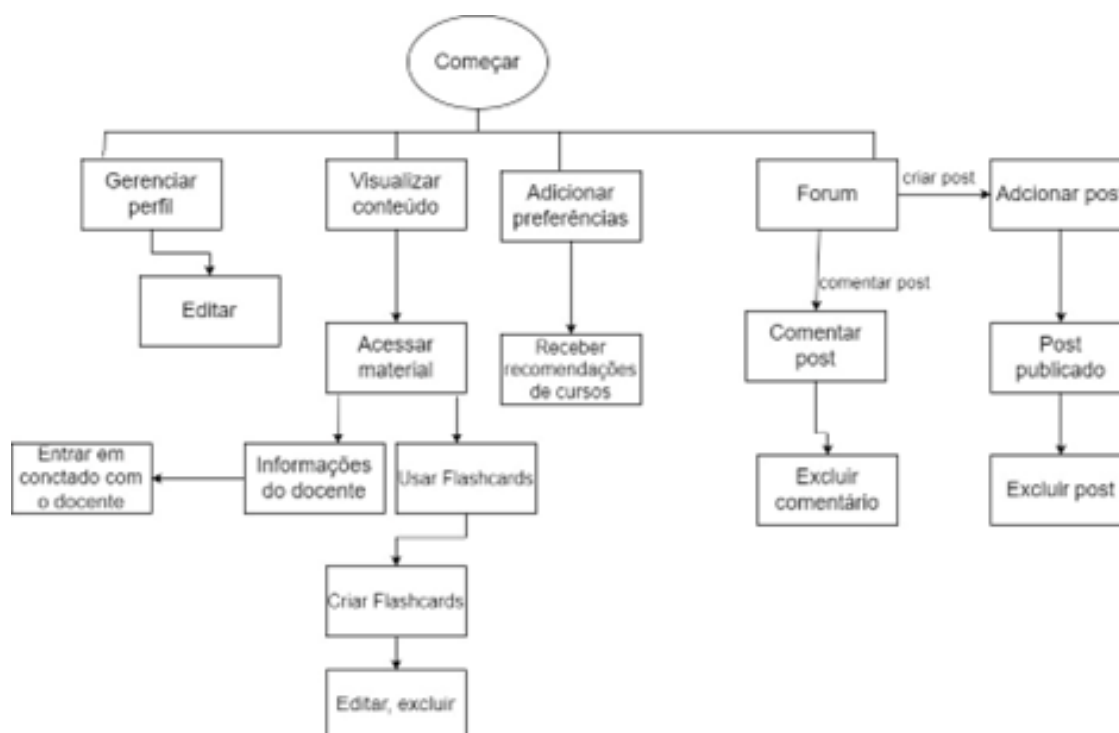


Figura 7: Diagrama de Interação para estudante autenticado.

Fonte: da Autora

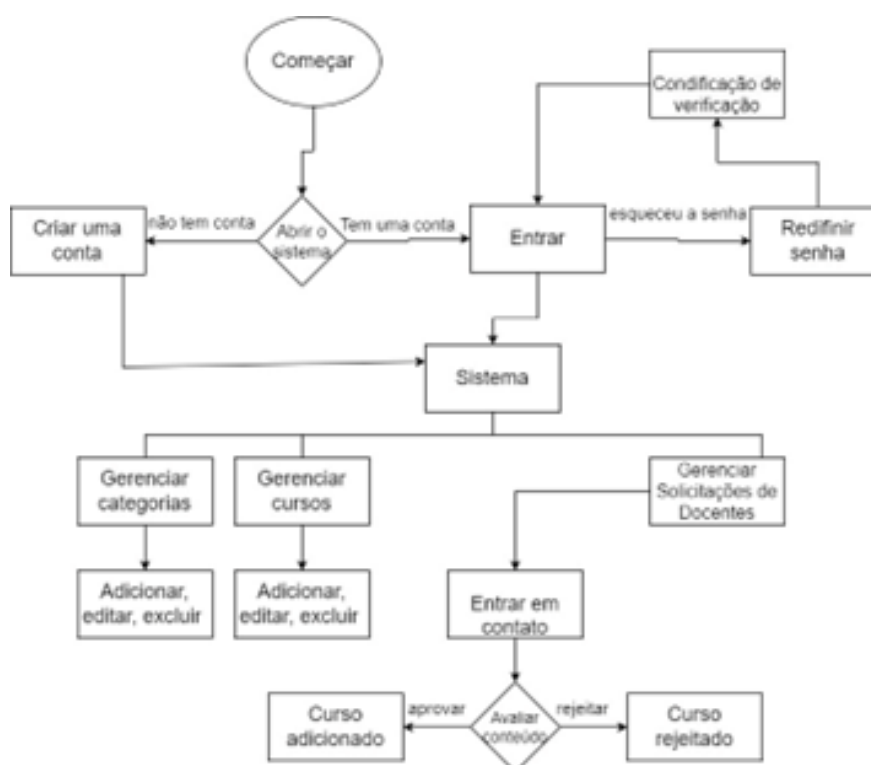


Figura 8: Diagrama de Interação do administrador.

Fonte: da Autora

A seguir apresentamos a descrição de cada caso de uso, detalhando o seu fluxo principal e secundário, bem como as validações e quais são os atores que estão vinculados.

A Tabela 6 descreve o caso de uso "Registro", detalhando o processo pelo qual um utilizador novo se regista no sistema. O caso de uso especifica as interações necessárias entre o utilizador e o sistema, incluindo a inserção de dados pessoais e de autenticação, a validação das informações submetidas e a criação de um novo perfil na plataforma. Este processo é essencial para permitir o acesso personalizado aos cursos e às funcionalidades da aplicação.

Tabela 6: Descrição caso de uso “Registro”

Nome do Caso de Uso	Registro
Caso de uso geral	Registro de novos utilizadores
Ator principal	Administrador e Estudante
Atores secundários	
Resumo	Descreve a etapa de registo para acesso ao sistema
Pré-condições	Utilizador não registado no sistema
Pós-condições	Acesso a cursos e a outras funcionalidades conforme o perfil de utilizador
Fluxo principal	
Ações do Ator	Ações do sistema
1. Informar nome, telefone, <i>e-mail</i> e senha	1. Validar <i>e-mail</i> para ver se já consta na base de dados do sistema
	1.Utilizador é registado no sistema
Restrições/ Validações	1. Todos os dados precisam de ser informados. 2. Os dados informados precisam de estar corretos. 3. O <i>e-mail</i> tem de ser válido e não pode estar em uso por outro utilizador.
Fluxo Exceção	
Ações do Ator	Ações do Sistema
	1. Comunicar que o <i>e-mail</i> já se encontra no sistema. 2. Solicitar os dados novamente.

Fonte: da Autora.

A Tabela 7 descreve o caso de uso "Registo Docente", detalhando o processo pelo qual um docente se regista na plataforma para submeter e partilhar cursos. Este caso de uso envolve a inserção de informações adicionais relevantes ao perfil do docente, bem como a validação dessas informações pelo sistema. Após o registo, o docente poderá manifestar o interesse em disponibilizar cursos, sujeito à aprovação do administrador.

Tabela 7: Descrição caso de uso “Registo docente”

Nome do Caso de Uso	Registar o professor
Caso de uso geral	Registo de novos docentes
Ator principal	Docentes
Atores secundários	Administrador
Resumo	Descreve a etapa de registo para acesso ao sistema
Pré-condições	Utilizador (docente) não registado no sistema
Pós-condições	Acesso às funcionalidades do sistema
Fluxo principal	
Ações do Ator	Ações do sistema
1. Informar nome, telefone, <i>e-mail</i> , senha e mensagem.	1. Validar <i>e-mail</i> para ver se já consta na base de dados do sistema. 2. Professor é registado no sistema.
Restrições/ Validações	1. Todos os dados precisam de ser informados. 2. Os dados informados precisam de estar corretos. 3. O <i>e-mail</i> tem de ser válido e não pode estar em uso por outro utilizador.
Fluxo Exceção	
Ações do Ator	Ações do Sistema
	1. Comunicar erro no envio do formulário. 2. Solicitar os dados novamente.

Fonte: da Autora.

A Tabela 8 descreve o caso de uso "Login", detalhando o processo de autenticação de utilizadores no sistema. Este caso de uso inclui a inserção das credenciais de acesso (nome de utilizador e palavra-passe), a verificação dessas credenciais pelo sistema e, caso sejam válidas, o acesso à conta do utilizador. Este processo é fundamental para garantir o acesso seguro e personalizado às funcionalidades da plataforma, tanto para estudantes como para docentes.

Tabela 8: Descrição caso de uso "Login"

Nome do Caso de Uso	<i>Login</i>
Caso de uso geral	Autenticação no sistema
Ator principal	Administrador, docente e estudante
Atores secundários	
Resumo	Descreve a etapa de autenticação no sistema
Pré-condições	Utilizador registado no sistema
Pós-condições	Acesso a cursos e a outras funcionalidades conforme o perfil de utilizador
Fluxo principal	
Ações do Ator	Ações do sistema
1. Informar <i>e-mail</i> e senha	1. Validar <i>e-mail</i> e senha 2. Acesso ao sistema
Restrições/ Validações	1. Todos os dados precisam de ser informados (nome, senha). 2. Os dados informados precisam de estar corretos. 3. O <i>e-mail</i> tem de ser válido e não pode estar em uso por outro utilizador.
Fluxo Exceção	
Ações do Ator	Ações do Sistema
	1. Comunicar que o <i>e-mail</i> e a senha são válidos. 2. Comunicar que o <i>e-mail</i> já se encontra registado no sistema.

Fonte: da Autora

A Tabela 9 descreve o caso de uso "Gerir Categorias e Cursos", que detalha as ações realizadas pelo administrador para criar, editar ou remover categorias e cursos na plataforma. Este caso de uso inclui a gestão do conteúdo educacional, garantindo que os cursos sejam organizados de forma adequada e acessível aos utilizadores. O administrador tem a capacidade de assegurar a qualidade e a relevância dos cursos disponíveis, contribuindo para a manutenção e atualização contínua da plataforma.

Tabela 9: Descrição caso de uso “Gerir Categorias e Cursos”

Nome do Caso de Uso	Gerir categorias e cursos
Caso de uso geral	Adição, atualização e exclusão de categorias e cursos.
Ator principal	Administrador
Atores secundários	
Resumo	Descreve as ações do administrador para gerir categorias e cursos
Pré-condições	Administrador autenticado no sistema
Pós-condições	Categorias e cursos adicionados, editados, atualizados ou excluídos
Fluxo principal	
Ações do Ator	Ações do sistema
1. Adicionar, atualizar ou excluir uma categoria ou curso.	1. Validar a ação do administrador 2. Adicionar, atualizar ou excluir categoria ou curso no sistema.
Restrições/ Validações	1. As informações da categoria ou curso precisam de ser informadas. 2. Apenas administradores podem gerir categorias e cursos.
Fluxo Exceção	
Ações do Ator	Ações do Sistema
	1. Comunicar erro na adição, atualização ou exclusão da categoria ou curso. 2. Solicitar dados novamente

Fonte: da Autora

A Tabela 10 descreve o caso de uso "Acesso aos Materiais do Curso", que detalha o processo pelo qual os estudantes podem visualizar e interagir com os materiais dos cursos disponíveis na plataforma. Este caso de uso inclui a navegação pelos conteúdos do curso, como vídeos, documentos e outros recursos educacionais, permitindo que os estudantes acedam ao material de forma organizada e fácil de utilizar. Essa funcionalidade é essencial para garantir que os estudantes tenham acesso contínuo e eficiente ao conteúdo necessário para o seu aprendizado.

Tabela 10: Descrição caso de uso “Acesso aos materiais do curso”

Nome do Caso de Uso	Acesso aos materiais do curso
Caso de uso geral	Acesso aos materiais dos cursos, informações do docente e recomendações de cursos
Ator principal	Estudante
Atores secundários	Docente
Resumo	Descreve como o estudante autenticado pode aceder aos materiais do curso, obter informações do docente e receber recomendações de cursos.
Pré-condições	Estudante autenticado no sistema e com cursos adicionados à sua conta
Pós-condições	Acesso aos materiais do curso
Fluxo principal	
Ações do Ator	Ações do sistema
1. Selecionar um curso na lista de cursos adicionados 2. Aceder ao material do curso 3. Selecionar a opção para visualizar informações do docente 4. Visualizar recomendações de cursos com base nas suas preferências	1. Exibir materiais do curso selecionado 2. Validar acesso aos materiais e exibir conteúdo 3. Exibir informações do docente 4. Analisar preferências do estudante e exibir cursos 5. Analisar preferências do estudante e exibir cursos recomendados
Restrições/ Validações	
1. O estudante deve estar autenticado para aceder às matérias do curso 2. As informações do docente só são exibidas para os cursos em que o estudante está matriculado. 3. Recomendações de cursos são baseadas nas preferências do estudante.	
Fluxo Exceção	
Ações do Ator	Ações do Sistema
	1. Comunicar erro no acesso aos materiais do curso ou informações do docente.

Fonte: da Autora

A Tabela 11 descreve o caso de uso "Criar *Flashcards*", que detalha o processo pelo qual os estudantes podem criar e personalizar *flashcards* como ferramenta de estudo. Este caso de uso envolve a criação de cartões de memória com perguntas e respostas ou conceitos-chave, permitindo que os estudantes organizem e reforcem os conhecimentos adquiridos ao longo dos cursos. A funcionalidade é projetada para oferecer uma abordagem interativa e eficaz de revisão e aprendizagem, adaptada às necessidades individuais dos utilizadores.

Tabela 11: Descrição caso de uso “Criar Flashcards”

Nome do Caso de Uso	Criar <i>flashcards</i>
Caso de uso geral	Criação de <i>flashcards</i> para revisão e aprendizagem
Ator principal	Estudante
Atores secundários	
Resumo	Descreve a etapa de criação de <i>flashcards</i> para revisão e auxílio na aprendizagem
Pré-condições	Utilizador autenticado no sistema
Pós-condições	<i>Flashcards</i> disponíveis para revisão do utilizador
Fluxo principal	
Ações do Ator	Ações do sistema
1. Informar perguntas e respostas para os <i>flashcards</i>	1. Validar dados inseridos 2. Armazenar <i>flashcards</i> no perfil do utilizador
Restrições/ Validações	1. Perguntas e respostas precisam de ser informadas
Fluxo Exceção	
Ações do Ator	Ações do Sistema
	1. Comunicar erro na criação dos <i>flashcards</i> 2. Solicitar dados novamente

Fonte: da Autora

A Tabela 12 descreve o caso de uso "Criar Fórum", detalhando o processo pelo qual o administrador ou docente pode criar um fórum de discussão na plataforma. Este caso de uso envolve a definição de tópicos e a configuração do espaço de interação entre os estudantes, permitindo que estes partilhem conhecimentos, discutam temas relacionados ao curso e troquem experiências. A funcionalidade de criação de fóruns visa promover a colaboração e o envolvimento dos utilizadores, enriquecendo a experiência de aprendizagem através da interação entre os membros da comunidade.

Tabela 12: Descrição caso de uso “Criar fórum”

Nome do Caso de Uso	Criar fórum
Caso de uso geral	Interação em fóruns
Ator principal	Estudante
Atores secundários	
Resumo	Descreve as ações de publicação, resposta e exclusão de <i>posts</i> no fórum
Pré-condições	Estudante autenticado no sistema
Pós-condições	<i>Post</i> publicado, respondido ou excluído
Fluxo principal	
Ações do Ator	Ações do sistema
1. Selecionar a opção de publicar, responder ou excluir um <i>post</i>	1. Validar ação do estudante 2. Publicar ou responder <i>post</i> 3. Excluir <i>post</i>
Restrições/ Validações	1. O conteúdo do <i>post</i> não pode estar vazio 2. Apenas o autor do <i>post</i> ou um administrador pode excluir um <i>post</i>
Fluxo Exceção	
Ações do Ator	Ações do Sistema
	1. Comunicar erro na publicação, resposta ou exclusão do <i>post</i> 2. Solicitar dados novamente

Fonte: da Autora

A Tabela 13 descreve o caso de uso "Gerir Eventos", que detalha as ações realizadas pelo administrador para criar, editar ou remover eventos na plataforma. Este caso de uso permite a gestão de atividades como workshops, seminários, ou sessões ao vivo relacionadas aos cursos. A funcionalidade inclui a definição de datas, horários e descrições dos eventos, garantindo que os estudantes tenham acesso a oportunidades adicionais de aprendizagem e interação. O gerenciamento eficaz de eventos contribui para o engajamento dos utilizadores e a diversificação das experiências educacionais oferecidas pela plataforma.

Tabela 13: Descrição caso de uso “Gerir eventos”

Nome do Caso de Uso	Gerir eventos
Caso de uso geral	Adição de vídeos e artigos por docentes
Ator principal	Docente
Atores secundários	Estudante, utilizador não autenticado
Resumo	Descreve as ações do docente para adicionar vídeos e artigos e o acesso a estes eventos pelos utilizadores
Pré-condições	Docente autenticado no sistema
Pós-condições	Vídeos e artigos adicionados ao sistema e disponíveis para visualização
Fluxo principal	
Ações do Ator	Ações do sistema
1. Selecionar a opção de adicionar vídeo ou artigo 2. Informar detalhes do vídeo ou artigo	1. Validar ação do docente 2. Adicionar vídeo ou artigo ao sistema 3. Tornar vídeo ou artigo disponível para utilizadores
Restrições/ Validações	1. Apenas docentes e administrador podem adicionar vídeos ou artigos 2. Eventos devem estar disponíveis para todos os utilizadores, autenticados ou não
Fluxo Exceção	
Ações do Ator	Ações do Sistema
	1. Comunicar erro no acesso ao vídeo ou artigo

Fonte: da Autora

6.3 Desenvolvimento

No desenvolvimento do projeto *LearnMug*, utilizámos o *framework Next.js*, que adota uma abordagem orientada a componentes. Esta abordagem permite dividir o código em módulos menores e reutilizáveis, otimizando o processo de desenvolvimento, melhorando a segurança da aplicação e facilitando a sua manutenção a longo prazo. Para organizar o *Web* e a estrutura dos componentes, seguimos o método *Atomic Web*, conforme ilustrado na Figura 9, o que nos permitiu criar uma hierarquia clara e consistente de elementos, promovendo a escalabilidade e a eficiência do projeto.

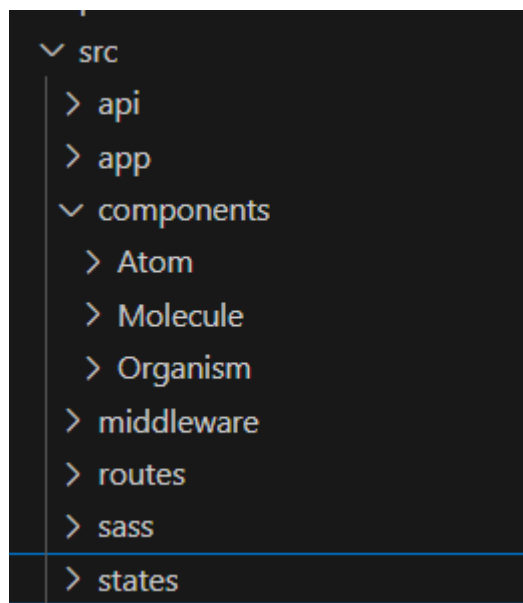


Figura 9: Método Atomic Web.

Fonte: da Autora

Nesta abordagem, o nosso projeto **LearnMug** está organizado de acordo com a seguinte estrutura de pastas, garantindo uma separação clara das responsabilidades e facilitando o desenvolvimento e a manutenção:

- **Api:** Contém os arquivos responsáveis por lidar com as requisições externas, incluindo a comunicação com *APIs*, integração de serviços e manipulação de dados recebidos de fontes externas.
- **App:** Esta pasta inclui a lógica principal da aplicação, responsável pelo fluxo de dados, operações de negócio e gestão de interações entre os componentes. É o núcleo da aplicação, onde a lógica central é implementada e coordenada.
- **Componentes:** Organizada com base no **Atomic Web**, esta pasta contém subpastas divididas em *atom*, *molecule* e *organism*, permitindo uma abordagem modular e escalável no desenvolvimento de interfaces:
 - **Atom:** Componentes básicos da interface e reutilizáveis, como botões, campos de entrada e ícones. São os blocos de construção fundamentais da aplicação.
 - **Molecule:** Componentes que combinam vários átomos para formar elementos de interface mais complexos, como formulários, grupos de botões ou campos de pesquisa. As moléculas encapsulam pequenos componentes interligados.

- **Organism:** Combinações de moléculas que formam partes significativas da interface, como cabeçalhos, rodapés ou seções completas de uma página. Esses organismos são utilizados para construir grandes porções da interface, proporcionando uma hierarquia clara.
- **Middleware:** Arquivos que implementam funcionalidades intermediárias e críticas, como a autenticação de utilizadores, gestão de sessões, e tratamento de permissões. O *middleware* é executado entre as requisições e as respostas, garantindo a segurança e o fluxo correto de dados.
- **Routes:** Define de forma clara e organizada as rotas de navegação da aplicação, especificando quais páginas ou componentes devem ser renderizados em resposta a cada URL. Esta pasta é crucial para garantir uma experiência de navegação fluida e eficiente dentro da aplicação.
- **SASS:** Contém os arquivos de estilo em **SASS**, que permitem a criação de folhas de estilo modulares e consistentes. Com o SASS, mantemos a coerência visual em toda a aplicação, utilizando variáveis, *mixins* e aninhamentos para organizar e otimizar os estilos.
- **States:** Responsável pela gestão e partilha de estados globais da aplicação. Para isso, utilizamos a biblioteca **Jotai**, que oferece uma forma simples e eficiente de gerir o estado em aplicações *React*. A *Jotai* facilita a partilha de estados entre componentes, garantindo uma comunicação fluida e eficaz, sem a necessidade de *middlewares* complexos.

6.4 Interfaces do Sistema

No projeto **LearnMug**, o sistema foi concebido com uma estrutura clara, dividida em três áreas distintas, cada uma com níveis de acesso e funcionalidades específicas:

1. **Área de Administrador:** Esta área é destinada aos administradores do sistema, que possuem o controlo total sobre a gestão dos cursos e utilizadores. Os administradores podem adicionar, editar ou remover cursos, gerir categorias, avaliar conteúdos submetidos pelos docentes e visualizar relatórios de desempenho da plataforma.
2. **Área de Estudantes e Docentes:** Esta secção é voltada para os utilizadores registados no sistema, sendo diferenciada entre estudantes e docentes:
 - **Estudantes:** Têm acesso aos cursos em que estão inscritos, bem como ao

seu perfil e progresso nas formações. Podem visualizar materiais de estudo, completar módulos e acompanhar o seu progresso ao longo do curso.

- **Docentes:** Podem ver os cursos que disponibilizaram, além de aceder aos dados relativos às inscrições e ao desempenho dos seus cursos. Os docentes também podem submeter novos cursos, que serão avaliados e aprovados pelo administrador.

3. **Área Comum:** Esta área é acessível a todos os utilizadores, mesmo aqueles que não estão registados. Inclui páginas públicas como a lista de cursos disponíveis, informações sobre a plataforma, formulários de registo e contacto.

A primeira parte do desenvolvimento focou-se na construção da **API**. Para definir e documentar as rotas e *endpoints*, utilizámos o **Swagger**, uma ferramenta que facilita a visualização e interação com as **APIs**, conforme ilustrado na Figura 10. A partir daí, criámos a base de dados e as respetivas tabelas, assegurando uma estrutura sólida para armazenar e gerir as informações dos cursos, utilizadores e interações.

Posteriormente, avançámos para o desenvolvimento do **back-end**, garantindo a integração eficiente entre a **API** e a base de dados, bem como a implementação de todas as funcionalidades necessárias para a gestão dos dados e fluxos da aplicação. Esta separação clara entre as diferentes áreas do sistema permite uma organização lógica e eficiente, facilitando a manutenção e futuras expansões da plataforma.



Figura 10: API do projeto.

Fonte: da Autora

6.4.1 Registo de Utilizadores

Após a criação e integração da API e do *back-end*, passámos ao desenvolvimento das interfaces de **registo** para os diferentes tipos de utilizadores: **administradores**, **docentes** e **estudantes**, conforme ilustrado na Figura 11. Cada ecrã de registo foi personalizado de acordo com as necessidades e funcionalidades de cada tipo de utilizador, garantindo uma experiência intuitiva e adaptada ao papel que cada um desempenha no sistema.

Após o **administrador** completar o processo de registo e autenticar-se com sucesso, ele tem acesso ao painel de controlo da aplicação, onde pode gerir todos os cursos disponíveis na plataforma. O administrador tem a capacidade de criar novos cursos, editar os existentes, remover cursos obsoletos, e supervisionar os cursos submetidos pelos docentes, assegurando a qualidade e relevância do conteúdo.

Da mesma forma, os **docentes** e **estudantes** seguem um processo de registo simples e claro, adaptado às suas funções dentro da plataforma. Os docentes, após o registo, podem submeter cursos para aprovação e acompanhar as estatísticas de desempenho dos mesmos. Já os estudantes, depois de se registarem, podem inscrever-se nos cursos disponíveis e aceder aos materiais de estudo.

The image displays two side-by-side web forms for user registration and login. Both forms feature a logo at the top center, consisting of two stylized, overlapping curved shapes in shades of green and blue.

Left Form (Registration):

- Name:** A text input field containing "Angelica Ribeiro da Silva".
- E-mail:** A text input field containing "angelica@sesam.com".
- Password:** A text input field with masked characters (dots).
- Telefone:** A text input field containing "931144217".
- REGISTER:** A dark gray button with white text.
- Login:** A small text label at the bottom right of the form.

Right Form (Login):

- E-mail:** A text input field containing "angelica@sesam.com".
- Password:** A text input field with masked characters (dots).
- LOGIN:** A dark gray button with white text.
- Register:** A small text label at the bottom right of the form.

Figura 11: Registo e login.

Fonte: da Autora

O **formulário de registo de docentes** permite que os professores interessados se registem no sistema para disponibilizar os seus cursos. No entanto, para garantir a qualidade dos conteúdos e a integridade da plataforma, o registo de cada docente precisa de ser **aprovado por um administrador** antes de o docente poder aceder a todas as funcionalidades, como a criação e gestão de cursos, conforme mostrado na Figura 12.

Este processo de aprovação garante que apenas docentes devidamente validados possam partilhar conteúdos, oferecendo aos estudantes a segurança de que os cursos são ministrados por profissionais qualificados. Após a submissão do registo, o administrador analisa as informações fornecidas pelo docente e, uma vez aprovado, o docente recebe um e-mail de confirmação e pode começar a utilizar as funcionalidades disponíveis na plataforma.

Essa etapa de validação é essencial para assegurar a **qualidade e confiabilidade** dos cursos oferecidos, além de manter o controle sobre os conteúdos que são disponibilizados aos estudantes, garantindo que os mesmos atendam aos critérios estabelecidos pela plataforma.

Torna-se um professor

Junta-se a equipa de professores do centro de formação da LearnMung. Aprender é uma arte. Ensinar é uma virtude.

Entre em Contacto

Preencha o formulário

Paulo Andre

paulo.silva@gmail.com

933666306

Deixe a sua mensagem

Sou Paulo André, formado em matemática aplicada, na universidade de Aveiro. |desejo partilhar os meus conhecimentos

Enviar

Figura 12: Formulário do docente

Fonte: da Autora

Para os **estudantes**, existe uma área dedicada onde podem realizar o seu **registo** e, posteriormente, **aceder à plataforma** através do login. Após autenticarem-se com sucesso, os estudantes podem explorar os cursos disponíveis e escolher aqueles que desejam realizar. Uma vez inscritos num curso, têm acesso a uma área específica onde são disponibilizados todos os **materiais** relacionados ao curso, como vídeos, documentos, *quizzes* e outros recursos de apoio, conforme ilustrado na Figura 13 e na Figura 14.

Esta interface foi desenhada para ser intuitiva e de fácil navegação, permitindo que os estudantes encontrem rapidamente os cursos em que estão inscritos e acessem aos conteúdos de forma organizada. A área de materiais do curso é atualizada automaticamente conforme o progresso do estudante, proporcionando uma experiência de aprendizagem estruturada e interativa.

Além disso, os estudantes podem acompanhar o seu progresso, ver notas ou *feedback* fornecido pelos docentes, e participar em atividades adicionais, como fóruns de discussão e eventos ao vivo, tudo dentro desta área dedicada ao estudo.

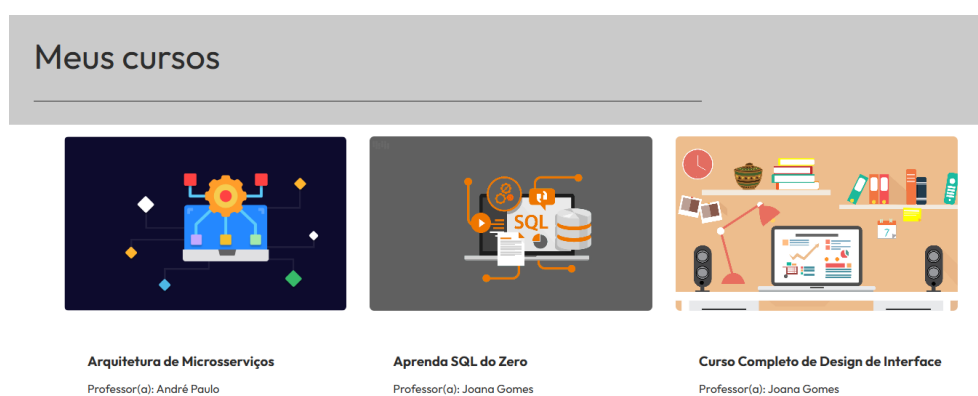


Figura 13: Página dos cursos.

Fonte: da Autora

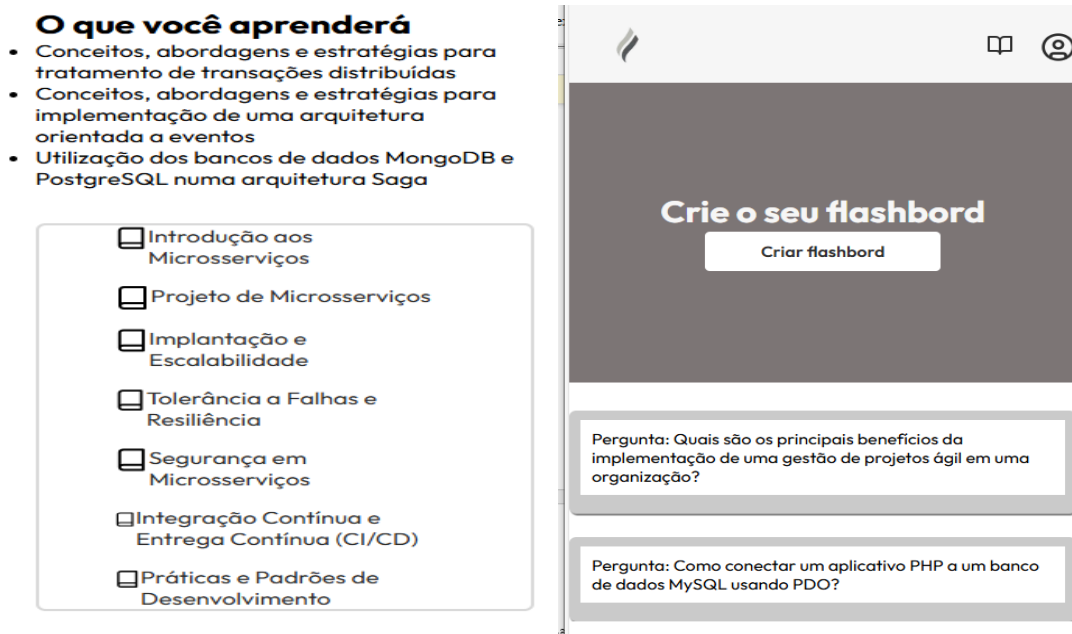


Figura 14: Área do estudante.

Fonte: da Autora

6.4.2 Categorias e cursos

As **categorias** e os **cursos** na plataforma **LearnMug** só podem ser registados e geridos pelos **administradores**, garantindo assim um controlo rigoroso sobre o conteúdo oferecido. Caso um docente deseje disponibilizar um curso, ele deve submeter uma solicitação ao administrador, que pode aprová-la ou recusá-la, dependendo da avaliação do conteúdo proposto. Esta abordagem assegura que apenas cursos relevantes e de qualidade são disponibilizados aos estudantes.

Para testar o sistema e demonstrar o seu funcionamento, inserimos algumas **categorias** e **cursos** de exemplo, conforme apresentado na Figura 15. Esses dados fictícios permitiram verificar a funcionalidade de criação e gestão de cursos, além de ilustrar como os cursos são organizados dentro das diferentes categorias, facilitando a navegação e a pesquisa por parte dos utilizadores.

Este processo de categorização e validação de cursos é essencial para manter a estrutura da plataforma organizada e acessível, ao mesmo tempo que garante a relevância do conteúdo educacional oferecido.

The image shows a web interface for adding a new course. At the top, there are two tabs: 'Cursos' and 'Categoria', with a user profile icon on the right. The main form is titled 'Inserir Cursos'. It contains several input fields: 'Name' with the value 'Marketing', 'Description' with the text 'Crie Sites usando a Inteligência Artificial e sem precisar ser Desenvolvedor e sem precisar aprender Códigos', 'Ratings' with the value '0', and 'Number of Ratings' with the value '0'. There is also a 'Course Syllabus' field with the text 'Conceitos, abordagens e estratégias para tratamento de transações distribuídas'. A blue button labeled 'Inserir Curso' is positioned at the bottom of the form.

Figura 15: Área administrativa das categorias e cursos.

Fonte: da Autora

6.4.3 Ecrã Principal

No **ecrã principal** da plataforma, representado na Figura 16, os utilizadores têm acesso a todas as funcionalidades principais do sistema de forma organizada e intuitiva. O ecrã oferece diversas opções, como a possibilidade de **filtrar e navegar** pelas **categorias** e **cursos**, permitindo que os utilizadores encontrem rapidamente o conteúdo que desejam explorar.

Além disso, o ecrã principal destaca os **cursos mais solicitados**, proporcionando uma visão clara dos cursos mais populares entre os utilizadores da plataforma. Também há uma **barra de pesquisa** que permite que os utilizadores insiram termos específicos para procurar cursos de forma eficiente.

No **rodapé do site**, encontram-se informações relevantes sobre a plataforma, como políticas de uso, contacto e outros detalhes institucionais, assegurando que os utilizadores tenham fácil acesso às informações necessárias.

Este ecrã principal foi projetado para facilitar a navegação e melhorar a experiência do utilizador, oferecendo um ponto de acesso centralizado para todas as funcionalidades e conteúdos disponíveis na plataforma.

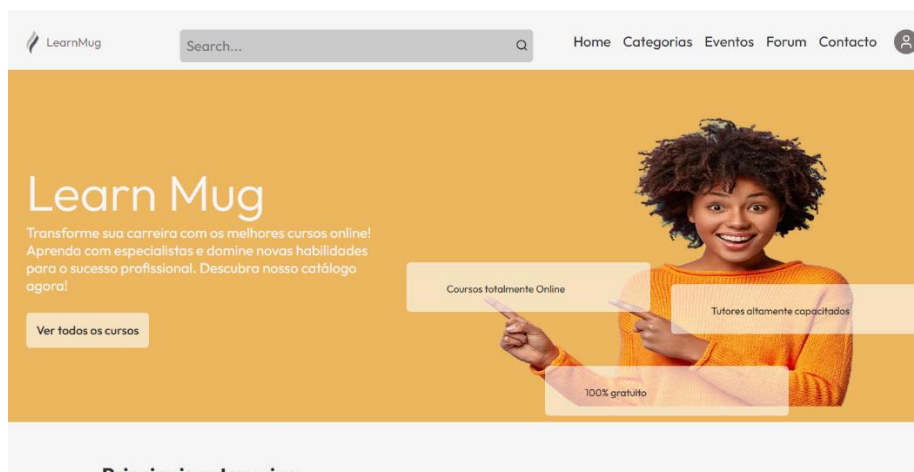


Figura 16: Ecrã principal.

Fonte: da Autora

A funcionalidade de **pesquisa por termos** permite aos utilizadores procurar cursos de forma rápida e eficiente através da **barra de navegação**. Quando um utilizador digita um termo de pesquisa, esse termo é armazenado na base de dados, o que permite à plataforma monitorar as pesquisas mais frequentes e, com base nesses registos, oferecer **sugestões de pesquisa** relevantes para os utilizadores.

Essa funcionalidade torna a navegação mais intuitiva e personalizada, permitindo que a plataforma sugira cursos ou termos semelhantes que possam interessar aos utilizadores, com base em pesquisas anteriores. No entanto, caso o curso procurado não exista na base de dados, a plataforma exibe uma mensagem de erro ou de ausência de resultados, conforme demonstrado na Figura 17.

Essa abordagem não só melhora a experiência do utilizador, como também permite à plataforma aprimorar continuamente a sua oferta de cursos, identificando possíveis lacunas e interesses dos utilizadores.

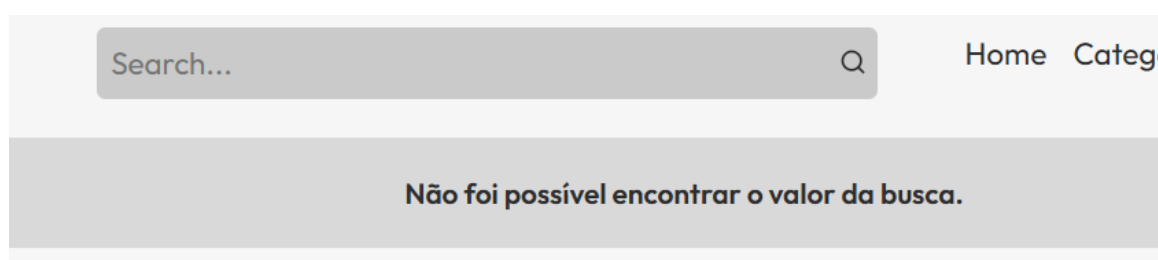


Figura 17: Resultado da pesquisa não encontrado.

Fonte: da Autora



Figura 18: Resultado da pesquisa por categoria ou curso

Fonte: da Autora

Quando o sistema localiza o curso correspondente, seja por meio da **pesquisa por categoria** ou por **termo específico**, o resultado é exibido de forma clara e organizada, conforme ilustrado na Figura 18. O ecrã de resultados apresenta os cursos relevantes que correspondem à pesquisa realizada, proporcionando aos utilizadores uma visão rápida e eficiente dos cursos disponíveis.

Cada curso encontrado é exibido com informações essenciais, como o **título**, a **descrição breve**, e, em alguns casos, a **classificação** ou **número de inscrições**, permitindo que os utilizadores avaliem rapidamente qual curso desejam explorar. Essa estrutura facilita a comparação entre os cursos e ajuda os utilizadores a tomar decisões informadas sobre qual curso seguir.

A navegação fluida entre as categorias e os resultados de pesquisa garante uma **experiência de utilizador** intuitiva, aumentando a satisfação e a eficiência ao procurar e seleccionar cursos na plataforma.

Na **página do fórum**, apresentada na Figura 19, os estudantes e docentes podem interagir, partilhando conhecimentos e discutindo temas relacionados aos cursos. No exemplo destacado, um estudante partilhou a sua visão sobre a **metodologia PMBOK**, promovendo uma discussão enriquecedora com outros membros da comunidade. O fórum é uma funcionalidade essencial para fomentar a troca de ideias, esclarecimento de dúvidas e a colaboração entre os utilizadores, criando uma comunidade de aprendizagem ativa.

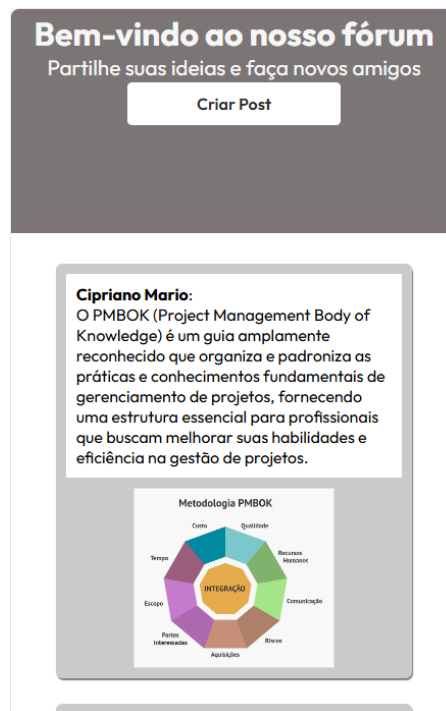


Figura 19: Página do fórum.

Fonte: do Autor

Além disso, a plataforma conta com uma **página de eventos**, onde são disponibilizados **artigos**, **Webinars** e outros recursos fornecidos pelos docentes para todos os utilizadores. Estes eventos, que podem incluir workshops e seminários *online*, oferecem oportunidades de aprendizagem adicionais, complementando os conteúdos dos cursos e proporcionando mais interatividade entre os participantes.

Essas funcionalidades contribuem para a criação de um ambiente educacional dinâmico e colaborativo, onde os utilizadores podem expandir os seus conhecimentos e participar de discussões relevantes, além de aceder a conteúdos complementares que enriquecem a sua experiência na plataforma.

Na página “Eventos” estão disponíveis artigos e *Webinares* para que os estudantes possam sempre atualizar os seus conhecimentos. A Figura 20 mostra alguns dos *Webinares* disponibilizados.



Figura 20: Página *Webnaries*.

Fonte: da Autora

6.5 Recomendações

Como o *LearnMug* está a ser lançado no mercado, ainda não possui registos suficientes das preferências dos utilizadores que permitiriam a implementação de um sistema de recomendações baseado em **filtragem colaborativa**. Para contornar essa limitação inicial, optámos por uma versão simplificada do sistema de recomendações, utilizando a **filtragem baseada em conteúdo**.

Neste sistema, os cursos são recomendados aos utilizadores com base nas **características e conteúdos** dos cursos que seleccionaram ou visualizaram previamente. Isso inclui fatores como a categoria do curso, tópicos abordados, e descrições detalhadas, permitindo que sejam sugeridos cursos com conteúdos semelhantes ou relacionados. Essa abordagem permite que o sistema ofereça sugestões relevantes e personalizadas, mesmo num cenário em que há pouca ou nenhuma interação anterior dos utilizadores.

A Figura 21 ilustra como os cursos recomendados são exibidos de acordo com o conteúdo do curso previamente escolhido pelo utilizador. Esta interface garante uma experiência de descoberta eficaz, ajudando os utilizadores a encontrar rapidamente cursos que correspondam aos seus interesses, desde o primeiro acesso à plataforma.

Com o aumento da interação dos utilizadores ao longo do tempo, o sistema de recomendações pode evoluir para incorporar **filtragem colaborativa**, permitindo sugestões ainda mais precisas com base nos perfis e preferências acumuladas dos utilizadores.

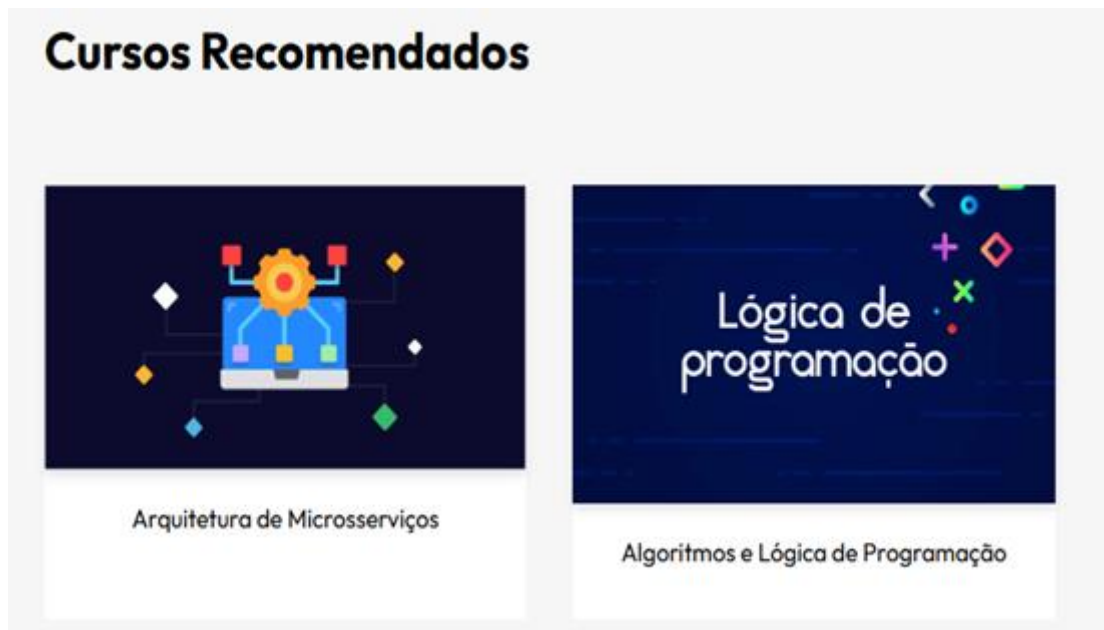


Figura 21: Cursos recomendados

Fonte: da Autora

6.6 Medidas de segurança na aplicação

A **integridade** e **confidencialidade** dos dados dos utilizadores, bem como a **robustez** da aplicação face a potenciais ameaças, são aspetos fundamentais para garantir o sucesso da plataforma **LearnMug** e manter a confiança dos seus utilizadores. Para atingir este objetivo, é essencial adotar abordagens eficazes de segurança, alinhadas com as **melhores práticas** do setor e as tendências mais recentes no campo da **cibersegurança**.

Neste contexto, foram implementadas várias medidas de segurança para proteger tanto os dados sensíveis dos utilizadores quanto o próprio sistema contra vulnerabilidades comuns. Essas medidas abrangem diversas áreas da segurança, como **autenticação**, **controlo de acesso**, **proteção contra ataques XSS (Cross-Site Scripting)** e **prevenção de injeção de SQL**, entre outras.

A Tabela 14 destaca as principais **categorias de segurança** e as respetivas **medidas adotadas** para fortalecer a segurança e a robustez do sistema segundo. Essas medidas incluem o uso de **autenticação segura** via *tokens* JWT (JSON Web Tokens), **criptografia** dos dados sensíveis, validação e sanitização das entradas de utilizadores, e a implementação de **firewalls** para prevenir acessos não autorizados [33].

Além disso, práticas como a realização de **testes de penetração** periódicos e a monitorização contínua do sistema foram adotadas para garantir que a plataforma esteja sempre

preparada para enfrentar novas ameaças. Essas abordagens visam proteger a aplicação contra ataques externos e assegurar que os dados dos utilizadores estejam sempre protegidos, promovendo assim um ambiente de aprendizagem seguro e confiável.

Tabela 14: Implementação de medidas de segurança na aplicação Web

Categoria de Segurança	Medida de Segurança	Justificativa
Comunicação Segura	Implementação de <i>HTTPS</i>	Protege a comunicação entre o cliente e o servidor, garantindo a integridade e a confidencialidade dos dados transmitidos.
Gestão de Sessões	Uso de tokens de sessão seguros	Reduz o risco de ataques de sessão, garantindo que apenas utilizadores autenticados tenham acesso à aplicação.
Proteção contra Injeção	Validação e sanitização de entradas	Evita ataques de injeção <i>SQL</i> e outros tipos de injeções, protegendo contra a manipulação maliciosa de dados na aplicação.
Atualizações e <i>Patching</i>	Política de atualizações regulares	Mantém a aplicação atualizada com as últimas correções de segurança, reduzindo a exposição a vulnerabilidades conhecidas.
<i>Backup</i> e Recuperação de Dados	Rotinas de <i>backup</i> automatizadas	Garante a disponibilidade e integridade dos dados, possibilitando a recuperação em caso de perda ou corrupção.

Fonte: adaptado de (SHOSTACK,2014).

A implementação de medidas de segurança numa aplicação *Web* é essencial para garantir a **integridade**, **confidencialidade** e **disponibilidade** dos dados, especialmente em contextos sensíveis, como a gestão de cursos *online*. Nesse sentido, uma abordagem abrangente de segurança deve cobrir todos os pontos vulneráveis, desde a comunicação segura até a gestão de acessos e a proteção contra ameaças externas.

A **comunicação segura** é assegurada através da implementação do protocolo ***HTTPS***, que protege os dados transmitidos entre o cliente e o servidor, impedindo interceptações maliciosas ou manipulações de dados. Esta é uma medida fundamental para proteger informações sensíveis, como credenciais de login e detalhes dos cursos.

A **segurança das sessões** é garantida pelo uso de ***tokens seguros*** e a aplicação de políticas de **expiração automática de sessões inativas**, minimizando os riscos de sequestro de sessões e outras vulnerabilidades relacionadas à persistência de sessão. Estas medidas reduzem a superfície de ataque e protegem os utilizadores contra acessos não autorizados.

A **proteção contra injeções** é alcançada por meio da **validação e sanitização rigorosa de entradas** de utilizadores, prevenindo ataques de injeção *SQL* e outras formas de injeções de

código. Isso garante que os dados manipulados na aplicação sejam confiáveis, mantendo a integridade do sistema.

Além disso, a **auditoria e monitorização contínua** são práticas cruciais para identificar e responder a atividades suspeitas. Através de **registros detalhados** e **alertas automáticos**, a aplicação pode detetar ameaças em tempo real, permitindo uma resposta proativa a potenciais brechas de segurança.

A **política de atualizações regulares** e a aplicação de **patches de segurança** garantem que a aplicação esteja sempre protegida contra vulnerabilidades conhecidas. A manutenção regular é uma prática essencial para reduzir a exposição a riscos, conforme evoluem as ameaças cibernéticas.

A **autenticação robusta** é a primeira linha de defesa essencial. A implementação de **autenticação de dois fatores (2FA)** proporciona uma camada adicional de proteção contra acessos não autorizados, garantindo que apenas utilizadores devidamente autenticados possam aceder à plataforma. As políticas de criação de **palavras-passe fortes** e a educação dos utilizadores sobre práticas de autenticação seguras são igualmente fundamentais [34].

A **autorização eficaz** é outro componente central da segurança. A aplicação de um sistema de **controlo de acesso baseado no princípio do menor privilégio** assegura que os utilizadores tenham apenas as permissões estritamente necessárias para desempenharem as suas funções. Isso reduz significativamente a possibilidade de acessos não autorizados a dados sensíveis [35].

Para garantir a **proteção contra ameaças externas**, a adoção de **práticas de codificação segura** é essencial. Segundo Rathfux *et al.* (2018), a implementação de **códigos robustos** e o uso de **frameworks de segurança** são medidas eficazes para prevenir vulnerabilidades como **injeção de código** e **Cross-Site Scripting (XSS)**. Além disso, é importante manter a aplicação sempre atualizada com **patches de segurança** para mitigar riscos associados a falhas conhecidas [36].

A realização de **testes de segurança periódicos** é uma prática recomendada, porque permite que a aplicação seja submetida a **testes de penetração** e **análises de vulnerabilidades** para identificar possíveis falhas de segurança antes que sejam exploradas por atacantes. Estas medidas permitem a correção proativa de problemas e aumentam a resiliência do sistema contra ataques [35].

A **segurança na transmissão de dados** também é um ponto crítico. A utilização de protocolos de segurança como o **HTTPS**, garante a **confidencialidade** dos dados em trânsito,

protegendo-os contra **interceptações indesejadas** e assegurando a integridade das comunicações entre cliente e servidor [35].

Em resumo, a implementação de **medidas de segurança** na aplicação *Web* de gestão de cursos *online* é uma etapa crucial do processo de desenvolvimento. A adoção de práticas robustas, como **autenticação forte**, **autorização eficaz**, **codificação segura**, **atualizações regulares** e **testes de segurança**, é fundamental para garantir a integridade, confidencialidade e disponibilidade dos dados, promovendo a **confiança dos utilizadores** e assegurando a conformidade com as normas de segurança em vigor.

6.7 Proteção de dados sensíveis dos utilizadores

Com a crescente digitalização do ensino, o desenvolvimento de aplicações *Web* para a gestão de cursos *online*, aliadas ao **Web responsivo**, tornou-se essencial para oferecer uma experiência otimizada em diversas plataformas. No entanto, à medida que essas plataformas coletam e processam **dados sensíveis** dos utilizadores, a **proteção da privacidade** surge como uma preocupação central. Implementar práticas eficazes de proteção de dados é crucial para garantir a confiança dos utilizadores e a conformidade com as regulamentações de privacidade e proteção de dados [35].

Um dos principais aspetos na proteção de dados sensíveis é a **consciencialização contínua dos desenvolvedores** sobre as melhores práticas de segurança [37]. A formação constante sobre ameaças emergentes e técnicas de mitigação é essencial para evitar vulnerabilidades no sistema. A adoção de **padrões de segurança** desde o início do ciclo de vida do desenvolvimento, contribui para a criação de uma base sólida de proteção de dados desde a concepção do sistema [35].

No contexto de **aplicações Web de gestão de cursos online**, a coleta e armazenamento de informações sensíveis, como **dados pessoais** e **históricos de desempenho académico**, exige medidas específicas de proteção. A utilização de **métodos robustos de criptografia** para proteger a transmissão e o armazenamento de dados confidenciais é uma prática indispensável [38]. Essa prática assegura que, mesmo em caso de acesso não autorizado, as informações permaneçam inacessíveis e ininteligíveis para terceiros.

Outro aspeto crucial é a **implementação de controlos de acesso granulares**, que restringem o acesso a informações sensíveis apenas a utilizadores autorizados. A aplicação do princípio de **menor privilégio** garante que cada utilizador aceda somente às informações necessárias para a sua função, reduzindo drasticamente o risco de exposição indevida de dados sensíveis e fortalecendo a segurança global do sistema [38].

Além das **medidas técnicas**, a **transparência com os utilizadores** é fundamental para a proteção de dados sensíveis. Fornecer **políticas de privacidade** claras e acessíveis, que expliquem de maneira transparente como os dados serão utilizados. Isso não só cumpre os requisitos legais, mas também fortalece a confiança dos utilizadores na plataforma, garantindo que estejam cientes de como suas informações estão a ser tratadas [38].

O desenvolvimento de aplicações *Web* de gestão de cursos *online* com **Web responsivo** requer uma abordagem proativa para a proteção de dados sensíveis dos utilizadores. A **integração de práticas de segurança** desde as fases iniciais do desenvolvimento, aliada à consciencialização contínua e à transparência, é essencial para criar uma plataforma confiável e em conformidade com as expectativas de privacidade dos utilizadores.

6.8 Desafios e soluções no desenvolvimento da aplicação *LearnMug*

Durante o desenvolvimento da aplicação *LearnMug*, enfrentámos diversos desafios técnicos que exigiram a implementação de soluções adequadas para garantir uma experiência de utilizador eficiente e contínua.

Um dos principais problemas encontrados foi relacionado à **renderização assíncrona de dados** em *React* e *Next.js*, que gerou alguns problemas de "hydration". Esse problema ocorre quando a renderização no lado do cliente (client-side rendering) não corresponde à renderização no lado do servidor (server-side rendering) durante a inicialização da aplicação. Em essência, o conteúdo renderizado pelo servidor pode diferir daquele gerado pelo cliente, o que resulta em **erros de hydration**. Isso acontece, geralmente, quando o *React* tenta reutilizar o HTML gerado no servidor e no cliente, mas encontra inconsistências entre os dois.

Para mitigar esse problema, decidimos utilizar a função **dynamic** do *Next.js*, que permite o carregamento dinâmico de componentes em tempo de execução. O uso de **dynamic** não apenas resolve o problema de hydration, mas também melhora o desempenho inicial da aplicação ao carregar os componentes de forma assíncrona.

Apesar de ser uma solução eficaz, a implementação do **dynamic** resultou numa **lentidão na renderização** de certos dados, o que afetava a apresentação imediata de algumas informações. Por exemplo, ao abrir o menu do utilizador pela primeira vez numa página, o nome e o e-mail não eram exibidos imediatamente, pois os dados ainda estavam a ser carregados. Para resolver essa situação, ajustámos a lógica de requisição ao servidor, movendo-a para o **componente-pai** de todas as páginas, e utilizámos um **gestor de estados** para garantir que as informações do utilizador fossem armazenadas e persistissem, evitando a necessidade de fazer requisições à base de dados constantemente. Esta abordagem melhorou

significativamente o desempenho e a fluidez da interface, garantindo que as informações dos utilizadores fossem carregadas de maneira mais eficiente e sem atrasos perceptíveis.

Outro desafio considerável foi relacionado ao **upload de imagens**. Devido às restrições do **servidor Vercel**, que não aceita o upload de imagens diretamente, foi necessário adotar uma solução alternativa. Para superar essa limitação, optámos por converter as imagens em **base64** e armazená-las diretamente na base de dados. Essa abordagem permitiu tanto o armazenamento quanto a apresentação das imagens na aplicação, sem a necessidade de modificações complexas no servidor. No entanto, reconhecemos que, embora funcional, o uso de imagens em base64 pode aumentar o tamanho dos dados armazenados e impactar o desempenho em grande escala, o que poderá ser otimizado em futuras iterações da aplicação.

Esses desafios exigiram a adoção de soluções práticas e eficientes, garantindo que o **LearnMug** se mantivesse funcional e responsivo, mesmo diante de limitações técnicas, proporcionando uma experiência contínua e de qualidade para os seus utilizadores.

6.9 Análise do *Feedback* dos Alunos

A **avaliação de cursos online** é uma prática essencial para o contínuo aprimoramento do ensino a distância, especialmente no contexto do desenvolvimento de uma aplicação *Web* de gestão de cursos *online* como o **LearnMug**. A **análise do feedback dos alunos** tem um papel fundamental para compreender a eficácia da plataforma, identificar áreas de melhoria e ajustar as estratégias pedagógicas para melhor atender às necessidades dos utilizadores.

Para medir a **eficácia** e **usabilidade** do **LearnMug**, foi conduzida uma pesquisa abrangente junto a 10 estudantes que utilizaram a plataforma durante alguns meses. Esta pesquisa foi realizada com o objetivo de recolher percepções detalhadas sobre a experiência dos alunos, incluindo aspectos relacionados à navegação, acessibilidade, apresentação de conteúdos e interatividade do sistema.

A recolha de **feedback** foi realizada através de **questionários online**, que incluíram perguntas de múltipla escolha e questões abertas para permitir uma análise mais completa das opiniões dos utilizadores. Os questionários abrangeram tópicos como a facilidade de utilização, a clareza na apresentação dos materiais, a qualidade das ferramentas interativas (como fóruns e *flashcards*) e a satisfação geral com o ambiente de aprendizagem. O **período de recolha de dados** ocorreu no mês de Junho de 2024, garantindo uma base de dados suficientemente ampla para avaliar o desempenho da plataforma.

A Figura 22 apresenta os resultados da pesquisa, fornecendo uma visão clara sobre o **feedback** dos estudantes. Esses dados são valiosos para identificar **pontos fortes** da plataforma,

como funcionalidades que melhor atenderam às expectativas dos utilizadores, e **áreas de melhoria**, que podem incluir aprimoramentos na navegação, Web responsivo ou funcionalidades adicionais solicitadas pelos estudantes.

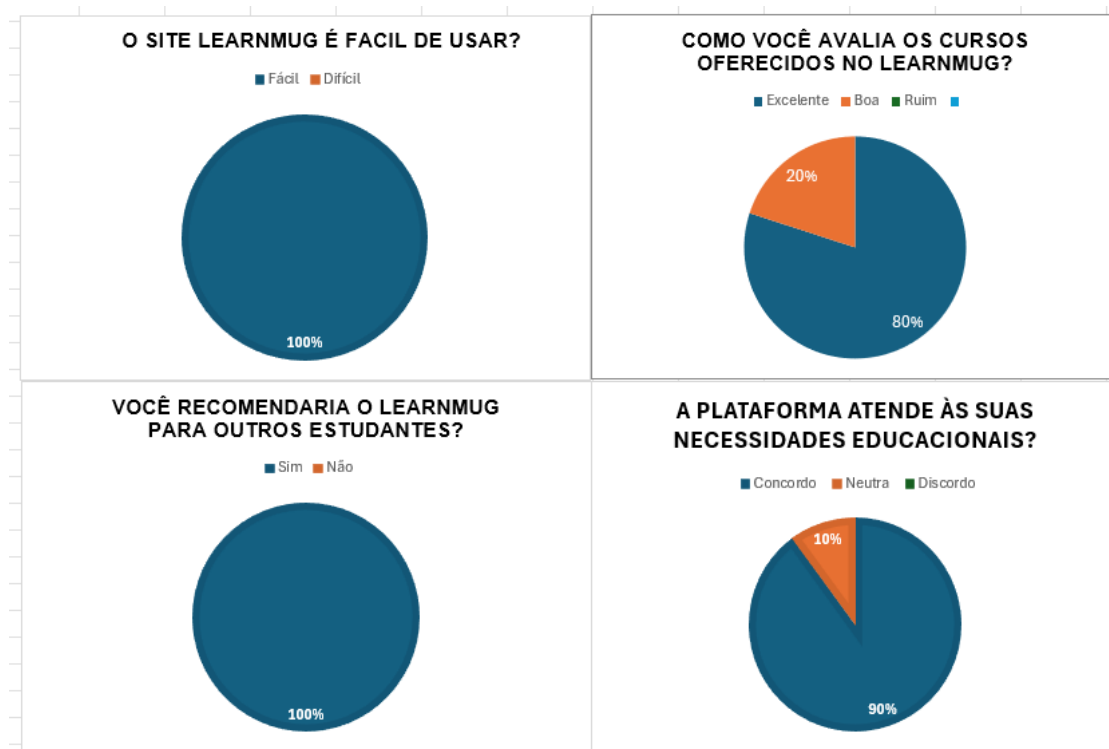


Figura 22: Análise do feedback dos alunos

Fonte: da Autora

A análise do *feedback* é uma parte crítica do processo de desenvolvimento contínuo da plataforma. Com base nos resultados, serão implementadas **ajustes e melhorias** tanto na interface do utilizador quanto nas funcionalidades da plataforma, garantindo que o **LearnMug** continue a evoluir e a proporcionar uma experiência de aprendizagem eficaz e satisfatória.

7 Conclusões

Neste capítulo, apresentamos uma reflexão final sobre o desenvolvimento e implementação da aplicação *LearnMug*, destacando as principais contribuições deste trabalho para o ensino a distância em Angola. Além disso, são discutidos os resultados alcançados, as lições aprendidas ao longo do processo e as soluções implementadas para superar os desafios encontrados. A partir da análise do *feedback* dos utilizadores e da avaliação das funcionalidades desenvolvidas, este capítulo também propõe direções futuras para aprimoramento da plataforma, garantindo a sua evolução contínua e a adaptação às novas necessidades educacionais.

7.1 Limitações do projeto

O desenvolvimento do *LearnMug* tem potencial para representar um avanço importante na democratização do acesso à educação digital em Angola, ao proporcionar uma plataforma acessível e eficiente para estudantes e docentes. Ao longo deste trabalho, foram explorados os desafios e as conquistas envolvidas nesse processo, com ênfase nos marcos alcançados e nas **limitações** que ainda precisam ser abordadas.

Entre as conquistas, destaca-se a implementação de um **Web responsivo**, que garante uma experiência de utilizador fluida e acessível em diferentes dispositivos, permitindo que os alunos acessem os conteúdos educacionais sem enfrentar barreiras técnicas. Além disso, a integração de funcionalidades de **gestão de cursos** simplificou significativamente a administração de conteúdos para instrutores e administradores, otimizando a organização e aumentando a eficiência operacional da plataforma.

No entanto, apesar desses avanços, o projeto enfrentou algumas **limitações** importantes que devem ser consideradas. Restrições de **tempo** e **recursos** influenciaram a extensão das funcionalidades implementadas e a profundidade da otimização do código. Embora as principais funcionalidades tenham sido entregues, algumas melhorias poderiam ter sido feitas, particularmente no que diz respeito à otimização da performance, à personalização de funcionalidades e à ampliação das ferramentas de análise de dados.

Além disso, desafios técnicos como a **compatibilidade entre navegadores** e a **escalabilidade** do sistema foram questões recorrentes ao longo do desenvolvimento. A variação de comportamentos entre diferentes navegadores exigiu adaptações adicionais para garantir uma experiência consistente para todos os utilizadores, independentemente da plataforma utilizada. Em termos de escalabilidade, embora o sistema tenha sido projetado para suportar um

número crescente de utilizadores, serão necessárias futuras otimizações para garantir que a plataforma mantenha um desempenho ideal à medida que o volume de dados e a base de utilizadores aumentem.

Outro ponto a considerar é a limitação na implementação de sistemas avançados de **recomendação**. A versão inicial utiliza uma filtragem baseada em conteúdo, que, embora eficiente, carece de personalização profunda que poderia ser alcançada com um sistema de **filtragem colaborativa** mais complexo, baseado nas interações dos utilizadores. Com mais dados e uma infraestrutura mais robusta, essa funcionalidade poderá ser explorada em versões futuras da plataforma.

Por fim, a dependência de serviços externos, como o uso de servidores para armazenamento de arquivos e a gestão de imagens, impõe algumas restrições ao desempenho e à flexibilidade do sistema. A necessidade de contornar essas limitações com soluções temporárias, como o armazenamento de imagens em **base64**, representa um compromisso que poderá ser aprimorado com recursos adicionais no futuro.

7.2 Trabalho futuro

Olhando para o futuro, existem várias áreas estratégicas que podem ser exploradas para aprimorar ainda mais a aplicação **LearnMug** e garantir que continue a evoluir de forma a atender às necessidades dos utilizadores. **Testes extensivos** e a recolha de **feedback contínuo** por parte dos utilizadores serão essenciais para identificar e resolver possíveis problemas de usabilidade ou desempenho que ainda possam existir. Com isso, será possível ajustar as funcionalidades de acordo com a experiência real dos utilizadores, garantindo uma plataforma ainda mais eficiente e intuitiva.

Além disso, a expansão das funcionalidades da plataforma representa uma oportunidade significativa de crescimento. A inclusão de **recursos de aprendizagem adaptativa** permitirá personalizar a experiência de cada aluno, oferecendo conteúdos e recomendações com base no seu progresso e preferências individuais. Da mesma forma, o **suporte a múltiplos idiomas** é uma evolução importante, que poderá ampliar o alcance da plataforma, permitindo que mais utilizadores de diferentes regiões tenham acesso ao conteúdo educacional.

Outro ponto crucial a ser considerado no desenvolvimento futuro é a **manutenção contínua da segurança e privacidade dos dados** dos utilizadores. Dado o crescente foco na **proteção de informações pessoais** em plataformas digitais, é fundamental implementar medidas robustas de **cibersegurança** e garantir a conformidade com as regulamentações de privacidade, como o **Regulamento Geral de Proteção de Dados (RGPD)**. Este compromisso

com a segurança reforçará a confiança dos utilizadores, assegurando que suas informações sejam tratadas de forma segura e responsável.

Considerando o ritmo acelerado do avanço tecnológico, é igualmente importante que a aplicação permaneça **atualizada** e adaptável às **novas tendências** do ambiente digital. Isso pode incluir a incorporação de tecnologias emergentes, como **inteligência artificial (IA)**, para melhorar os sistemas de recomendação e a personalização da aprendizagem, ou até o uso de **realidade virtual (VR)** e **realidade aumentada (AR)**, oferecendo novas formas interativas e imersivas de aprendizagem. Essas inovações poderiam transformar a forma como os cursos são apresentados e consumidos, enriquecendo ainda mais a experiência do utilizador.

Por fim, o desenvolvimento do **LearnMug** é apenas o início de uma jornada em direção à criação de uma plataforma acessível e eficaz para a **educação digital em Angola**. Embora desafios e limitações tenham sido enfrentados ao longo do processo, o compromisso com a **melhoria contínua** e a **inovação** impulsionará a plataforma rumo a um futuro onde o **acesso à educação online** seja verdadeiramente inclusivo e disponível para todos.

Este trabalho não representa apenas um ponto de partida, mas um **convite para uma jornada contínua de desenvolvimento e aperfeiçoamento**. Com **dedicação** e **colaboração**, a visão de uma educação digital inclusiva e acessível em Angola poderá ser plenamente realizada, trazendo impactos duradouros no campo da aprendizagem *online*.

7.3 Considerações Finais

Como já ressaltado, o desenvolvimento do **LearnMug** tem potencial para marcar um avanço significativo na promoção da **educação online em Angola**, refletindo um compromisso com a democratização do acesso ao conhecimento através de plataformas digitais. Ao longo deste trabalho, foram explorados os desafios e as oportunidades que emergem na criação de uma aplicação capaz de atender às necessidades de um público diversificado e em constante evolução.

Um dos pilares fundamentais do projeto foi o **Web responsivo**, que se destacou como um elemento crucial para garantir uma **experiência de utilizador fluida e consistente** em diferentes dispositivos, desde computadores a smartphones. Essa flexibilidade permite que os alunos acessem ao conteúdo educativo em qualquer lugar, sem enfrentar barreiras técnicas, promovendo assim uma **inclusão digital** mais ampla.

Outro ponto essencial do **LearnMug** é o **sistema de recomendação**, que desempenha um papel central no aumento do **comprometimento dos utilizadores**. Ao sugerir cursos com base nas preferências e interações dos alunos, o sistema de recomendação contribui

significativamente para uma experiência de aprendizagem personalizada, incentivando a continuidade dos estudos e melhorando a retenção dos utilizadores na plataforma.

Ao longo do desenvolvimento, foi dada atenção especial aos princípios de **usabilidade** e **acessibilidade**, com o objetivo de garantir que a aplicação seja **inclusiva** e acessível a uma ampla gama de utilizadores. Este foco inclui a consideração de **diferentes capacidades e níveis de familiaridade com a tecnologia**, tornando o *LearnMug* uma ferramenta eficiente tanto para iniciantes quanto para utilizadores mais experientes. A acessibilidade foi um dos valores centrais do projeto, assegurando que a plataforma possa ser usada por pessoas com **diversas habilidades e necessidades**.

A **segurança** da aplicação também foi uma prioridade ao longo de todo o processo de desenvolvimento, dado o caráter sensível das informações pessoais e académicas dos alunos. Foram implementadas **estratégias robustas de autenticação e proteção de dados**, garantindo a **privacidade** e a **confidencialidade** dos utilizadores. Essas medidas de segurança são essenciais para construir e manter a confiança dos utilizadores na plataforma.

Por fim, o **desenvolvimento contínuo** e a **manutenção da aplicação** serão fatores cruciais para garantir que o *LearnMug* permaneça relevante e eficaz ao longo do tempo. A rápida evolução da tecnologia, aliada às mudanças constantes nas necessidades e expectativas educacionais, exigirá atualizações regulares e a adição de **novas funcionalidades**. Para que a plataforma se mantenha competitiva e alinhada com as **tendências do mercado**, será necessário um esforço contínuo para incorporar inovações tecnológicas e melhorar a experiência do utilizador.

Em resumo, o *LearnMug* representa uma contribuição valiosa para o cenário da educação digital em Angola, proporcionando uma plataforma robusta e flexível que atende às exigências contemporâneas de **acessibilidade, usabilidade e segurança**. Este projeto, embora tenha atingido marcos importantes, é apenas o início de uma jornada de inovação contínua, que visa promover uma educação digital mais inclusiva, acessível e eficiente para todos.

Referências

- [1] E. von Kardorff, M. Castells, *The Rise of the Network society*. Cambridge, Mass. : Blackwell Publishers, 1996. ISBN: 1557866163, 1557866171.
- [2] D. Chaffet and P. R. Smith, *Emarketing excellence: Planning and optimizing your digital marketing*. London: Routledge, 2013. DOI: 10.4324/9780203082812.
- [3] D. Peppers and M. Rogers, *The one to one future: Building relationships one customer at a time*. Currency Doubleday, 1993. ISBN: 0385485662.
- [4] E. Marcotte, *Responsive Web*. Nova Iorque: Mandy Brown, 2011 ISBN: 978-0-9844425-77.
- [5] T.C. Nogueira, D.J. Ferreira, S.T. Carvalho, and L.O. Berreta, *Evaluating responsive Web Web's impact on blind users*. IEEE MultiMedia, vol.24 n° 2 pp.86–95. DOI:10.1109/MMUL.2017.21.
- [6] MAVINK, *Ciclo De Vida Del Desarrollo De Software*. Disponível em: <https://mavink.com/explore/Ciclo-De-Vida-Del-Desarrollo-De-Software>. [Acedido: 05-08-2024]
- [7] M. M. Paduelli, *Manutenção de Software: problemas típicos e diretrizes para uma disciplina específica*. Dissertação de Mestrado, Instituto de Ciências Matemáticas e de Computação. Universidade de São Paulo, São Carlos, SP 2007.
- [8] KUBETA. *Plataforma de e-learning Kubeta*. Disponível em: <https://www.kubeta.ao>. [Acedido: 05-06-2024]
- [9] LONGUKA (2019). *Plataforma de e-learning*. Disponível em: <https://www.verangola.net>. [Acedido: 05-06-2024]
- [10] N. Smyth, *Firestore essentials – Android Edition*. USA: CreateSpace Independent Publishing Platform, 2ª ed., 2017. ISBN: 9781979961608.
- [11] T. Malone, K. Grant, F. Turbak, S. Brobst and M. Cohen, *Intelligent information-sharing systems*. Association for Computing Machinery, vol. 30 n° 5, pp. 390–402. 1987. DOI: 10.1145/22899.22903.
- [12] R. G. N. Silva, *Sistema de recomendação baseado em conteúdo textual: Avaliação e comparação*. Dissertação de mestrado, Mestrado Institucional em Ciência da Computação, Universidade Federal da Bahia. Acedido em: https://repositorio.ufba.br/bitstream/ri/19281/1/dissertacao_mestrado_ciencia_computacao_rafael_glauber.pdf.

- [13] D. Goldberg, D. Nichols, B. M. Oki, and D. Terry, *Using collaborative filtering to weave an information tapestry*. Communications of the ACM, vol. 35, nº 12, pp. 61-70, 1992. DOI:10.1145/138859.138867.
- [14] P. Resnick and H. R. Varian, *Recommender systems*. Communications of the ACM vol 40 nº 3, pp. 56–58. DOI:10.1145/245108.245121.
- [15] D. Jannach, M. Zanker, A. Felfernig, and G. Friedrich, *Collaborative recommendation*, In: Recommender Systems: An Introduction, Cambridge University Press, 2010, pp.23-30. DOI:10.1017/CBO9780511763113.
- [16] G. Coulouris, J. Dollimore, and T. Kindbeg, *Sistemas distribuídos: Conceitos e projetos*. Porto Alegre: Bookman, 2013 ISBN:9788582600535.
- [17] M. Bornin, 2018. *O que são sistemas de recomendação?* Disponível em: <https://king.host/blog/2018/09/o-que-sao-sistemas-de-recomendacao>. [Acedido:12-03-2024]
- [18] M. G. Moore and G. Kearsley, *Distance education: A systems view of online learning*. Belmont, CA, Wadsworth Cengage Learning, 2012. ISBN: 978-1-111-52099-1.
- [19] D. R. Garrison, *E-learning in the 21st century: A framework for research and practice*. 1.^a ed., Routledge, 2003. DOI: 10.4324/9780203166093.
- [20] V.C Feijó and J. Baldessar, *Comunicação e mobilidade: A produção de conteúdo nas instituições de ensino superior catarinenses para dispositivos móveis*. Intercom – Sociedade Brasileira de Estudos Interdisciplinares da Comunicação XIV Congresso de Ciências da Comunicação na Região Sul – S. Cruz do Sul – RS, 2013.
- [21] T. Rathfux, J. Thoner, H. Kaindl, and R. Popp, RATHFUX, *Combining Design-time generation of Web-pages with responsive Design for improving low-vision accessibility*. In EICS '18: Proceedings of the ACM SIGCHI Symposium on Engineering Interactive Computing Systems. 2018. DOI: 10.1145/3220134.32201.
- [22] F. Zammetti, *Modern full-stack development: Using Typescript, React, Node.js, Webpack, and Docker*. New York: Apress, 2020. ISBN 9781484257388.
- [23] J. Duckett, *HTML & CSS Web and build Websites*. Indianapolis: John Wiley & Sons, Inc., 2011. ISBN: 9781118008188.
- [24] J. Spurllock, *Bootstrap: Responsive Web development*. Sebastopol: O'Reilly Media, Inc., 2013, ISBN 9781449343910.
- [25] S. R. Schach, *Engenharia de software: Os paradigmas clássicos e orientação a objetos*. 7.^a ed., 2008, ISBN: 9788577260454

- [26] J. Pires, *O que é API? REST e RESTful? Conheça as definições e diferenças!* Disponível em: <https://becode.com.br/o-que-e-api-rest-e-restful>.
- [27] J. Audy, *Desenvolvimento distribuído de software*. Rio de Janeiro: Elsevier Editora Ltda. 2012. eISBN: 9788535252385.
- [28] J. Lockhart, *Modern PHP*. Sebastopol, CA, O'Reilly Media, 2015. ISBN:1491905182.
- [29] B. Cherny, *ProgrammingTypeScript*. Sebastopol: O'Reilly Media, 2019. ISBN: 9781492037651.
- [30] M. Whitman and H. Mattord, *Principles of information security*. Boston: Cengage Learning., 2018. ISBN: 978-1-337-10206-3.
- [31] L. Agner, *Ergoweb e arquitetura de informação: Trabalhando com o utente*, 2.^a ed., Rio de Janeiro: Quartet, 2009.
- [32] E.Braude, *Projeto de software: Da programação à arquitetura - Uma abordagem baseada em Java*. Artmed.2004.
- [33] A. Shostack, *Threat modeling: Webing for security*. Nova Iorque: Wiley Publishing,2014. ISBN: 978-1-118-80999-0.
- [34] W. Stallings, *Computer security: Principles and practice*. Londres: Pearson,2020. ISBN 10: 1292220619.
- [35] R. S. Pressman, *Engenharia de software: Uma abordagem profissional* 7.^a ed., AMGH, 2011. ISBN :0073375977 / 9780073375977.
- [36] I. Sommerville, *Engenharia de software*. 9.^a ed., Pearson Education, 2011. ISBN: 978-85-7936-108-1.
- [37] R. Anderson, *Security engineering: A guide to building dependable distributed systems*. Nova Iorque: Wiley. ISBN-10: 1119642787.
- [38] S.Shirey, *RFC4949: Internet security glossary, version2*. Disponível em <https://datatracker.ietf.org/doc/html/rfc4>.