

Faculdade de Tecnologia de Americana “Ministro Ralph Biasi”

Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas

André Luís Marciano Dias

Veruska Diniz Pereira Lima

UNews
WEBSITE DE NOTÍCIAS DA FACULDADE

Americana, SP

2023

Faculdade de Tecnologia de Americana “Ministro Ralph Biasi”
Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas

André Luís Marciano Dias

Veruska Diniz Pereira Lima

UNews
WEBSITE DE NOTÍCIAS DA FACULDADE

Trabalho de Conclusão de Curso desenvolvido em cumprimento à exigência curricular do Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas, sob a orientação do Prof. Antonio Alfredo Lacerda

Área de concentração: Desenvolvimento de Software

Americana, SP

2023

FICHA CATALOGRÁFICA – Biblioteca Fatec Americana Ministro Ralph Biasi- CEETEPS Dados Internacionais de Catalogação-na-fonte

DIAS, André Luís Marciano

Unews: website de notícias da faculdade. / André Luís
Marciano Dias, Veruska Diniz Pereira Lima – Americana, 2023.

33f.

Relatório técnico (Curso Superior de Tecnologia em Análise
e Desenvolvimento de Sistemas) - - Faculdade de Tecnologia de
Americana Ministro Ralph Biasi – Centro Estadual de Educação
Tecnológica Paula Souza

Orientador: Prof. Esp. Antonio Alfredo Lacerda

1. Desenvolvimento de software 2. Engenharia de software
3. UML – linguagem de programação. I. DIAS, André Luís Marciano,
II. LIMA, Veruska Diniz Pereira III. LACERDA, Antonio Alfredo IV.
Centro Estadual de Educação Tecnológica Paula Souza – Faculdade
de Tecnologia de Americana Ministro Ralph Biasi

CDU: 681.3.05

681.3.05

681.3.061UML

Elaborada pelo autor por meio de sistema automático gerador de
ficha catalográfica da Fatec de Americana Ministro Ralph Biasi.

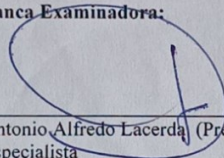
André Luis Marciano Dias
Veruska Diniz Pereira Lima

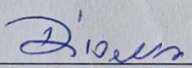
Unews
Website de notícias da faculdade


Trabalho de graduação apresentado como exigência parcial para obtenção do título de Tecnólogo em Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas pelo Centro Paula Souza – FATEC Faculdade de Tecnologia de Americana – Ralph Biasi.
Área de concentração: Sistemas de Informação

Americana, 28 de novembro de 2023

Banca Examinadora:


Antonio Alfredo Lacerda (Presidente)
Especialista
FATEC Faculdade de Tecnologia de Americana – Ralph Biasi


Diógenes De Oliveira (Membro)
Mestre
FATEC Faculdade de Tecnologia de Americana – Ralph Biasi


Clerivaldo José Roccia (Membro)
Mestre
FATEC Faculdade de Tecnologia de Americana – Ralph Biasi

RESUMO

O principal objetivo deste projeto é fornecer uma solução simples e eficaz para centralizar as notícias da Faculdade, tornando as informações institucionais mais acessíveis para a comunidade acadêmica. O site oferece uma experiência de usuário intuitiva e eficiente, facilitando a administração das notícias por meio de um painel de controle acessível, permitindo que a faculdade adicione, edite e remova notícias de maneira conveniente, garantindo que o conteúdo seja sempre atualizado e relevante. Além disso, o site inclui a capacidade de busca, o que permite aos leitores encontrar informações relevantes de forma rápida e fácil.

A escolha da arquitetura *MERN Stack* como base tecnológica para o projeto se deve à sua flexibilidade, que possibilita a criação de uma experiência eficaz tanto para os administradores quanto para os leitores. Além disso, foram integrados testes automatizados no processo, utilizando o *Robot Framework*, para garantir a qualidade e estabilidade do site, assegurando assim que a solução seja confiável e de alto desempenho.

.Palavras Chave: Notícias Institucionais, *Robot Framework*, *MERN Stack*

ABSTRACT

The main goal of this project is to provide a simple and effective solution for centralizing news from the College, making institutional information more accessible to the academic community. The website offers an intuitive and efficient user experience, making news management easy through an accessible control panel, allowing the college to add, edit, and remove news conveniently, ensuring that the content is always up-to-date and relevant. Additionally, the website includes a search capability, enabling readers to find relevant information quickly and easily.

The choice of the MERN Stack architecture as the technological foundation for the project is due to its flexibility, which allows for an effective experience for both administrators and readers. Furthermore, automated testing using the Robot Framework has been integrated into the process to ensure the quality and stability of the site, thereby ensuring that the solution is reliable and high-performing.

Keywords: Institutional News, Robot Framework, MERN Stack

SUMÁRIO

1 INTRODUÇÃO	1
2 ENGENHARIA DE SOFTWARE	2
2.1 REQUISITOS	2
2.2 REQUISITOS FUNCIONAIS	2
2.3 REQUISITOS NÃO FUNCIONAIS	3
2.4 MODELAGEM	4
2.5 DIAGRAMA DE CASO DE USO	4
2.6 DIAGRAMA DE CLASSES	8
2.7 DIAGRAMA DE ATIVIDADES	9
2.8 PLANO DE TESTES	11
3 BANCO DE DADOS	14
3.1 DIAGRAMA DE ENTIDADE E RELACIONAMENTO	14
3.2 DICIONÁRIO DE DADOS	14
4 DESENVOLVIMENTO	15
4.1 IDENTIFICAÇÃO	15
4.2 DESCRIÇÃO DA SOLUÇÃO PROPOSTA	16
4.3 FERRAMENTAS	17
4.3.1 MERN STACK	17
4.3.2 MONGO DB	18
4.3.3 EXPRESS JS	18
4.3.4 REACT JS	19
4.3.5 NODE JS	19
4.3.6 ROBOT FRAMEWORK	19
4.3.7 GIT	20
4.3.8 GITHUB	20
4.4 BACKEND	21
4.4.1 ROTAS	21
4.4.2 MIDDLEWARES	21
4.4.3 CONTROLLERS	21
4.4.4 SERVICES	22
4.5 FRONTEND	22
4.6 INTEGRAÇÃO	22
4.7 INTERFACES DE USUÁRIO E USABILIDADE	23
5 CONSIDERAÇÕES FINAIS	29
6 REFERÊNCIAS BIBLIOGRÁFICAS	31

LISTA DE FIGURAS

Figura 1 - Diagrama de Caso de Uso	5
Figura 2 – Diagrama de Classes.	9
Figura 3 – Diagrama de Atividades.	10
Figura 4 – Diagrama de Entidade e Relacionamento.	14
Figura 5 – Arquitetura da MERN Stack (Fonte: MongoDB).	18
Figura 6 - Responsividade do sistema	23
Figura 7 - Tela inicial para usuários não administradores	24
Figura 8 - Acesso ao conteúdo da notícia	25
Figura 9 - Tela inicial para usuários administradores	25
Figura 10 - Tela para edição de notícias	26
Figura 11 - Tela de acesso ao sistema	26
Figura 12 - Tela para recuperação de senha: inserção de email	27
Figura 13 - Tela para recuperação de senha: inserção de resposta	27
Figura 14 - Barra de navegação: usuário não autenticado	28
Figura 15 - Barra de navegação: usuário autenticado	28
Figura 16 - Modal com opções de filtros	28
Figura 17 - Tela para edição de perfil	29
Figura 18 – Acesso ao repositório do projeto	31

LISTA DE TABELAS

Tabela 1 - Lista de Requisitos Funcionais	3
Tabela 2 - Lista de Requisitos Não Funcionais	4
Tabela 3 – Caso de uso “Autenticação”.	6
Tabela 4 – Caso de uso “Adicionar Notícia”.	7
Tabela 5 – Caso de uso “Atualizar Notícia”.	7
Tabela 6 – Caso de uso “Remover Notícia”.	8
Tabela 7 – Caso de uso “Buscar Notícia”.	8
Tabela 8 – Caso de uso “Filtrar Notícia”.	9
Tabela 9 - Caso de teste “Login”	11
Tabela 10 - Caso de teste “Adicionar Notícia”	12
Tabela 11 - Caso de teste “Atualizar Notícia”	12
Tabela 12 - Caso de teste “Remover Notícia”	12
Tabela 13 - Caso de teste “Visualizar Notícia”	13
Tabela 14 - Caso de teste “Buscar Notícia”	13
Tabela 15 - Caso de teste “Logout”	13
Tabela 16 – Dicionário de Dados da entidade Faculdade.	14
Tabela 17 - Dicionário de Dados da entidade Notícias.	15

1 INTRODUÇÃO

O Projeto tem como seu objetivo o desenvolvimento de um website voltado para atender às necessidades da comunidade acadêmica. Os principais objetivos deste projeto são : Primeiro, visa ampliar significativamente a adesão da comunidade, fomentando uma participação ativa e envolvente, em segundo, centralizar e consolidar todas as informações acadêmicas, incluindo avisos, notícias e eventos, em um único local de fácil acesso, e por fim, almeja revolucionar a maneira como as informações são acessadas, tornando o processo mais eficiente, intuitivo e amigável.

Dentro do ambiente acadêmico em constante evolução, os alunos frequentemente se deparam com desafios ao tentar acessar e acompanhar as informações da Faculdade. Atualmente, existem diversos canais de comunicação não oficiais na instituição, e o que é oficial frequentemente não conta com a adesão da comunidade. Devido a essa cultura já estabelecida, a ausência de um local centralizado para essas informações tornou evidente a necessidade de criar um website que resolvesse essa questão de maneira eficiente.

Além disso, o projeto também visa promover a transparência e a atualização em tempo real das informações acadêmicas, contribuindo para um ambiente mais informado e conectado na comunidade acadêmica. A implementação de funcionalidades que permitam aos estudantes, professores e funcionários compartilhar e acessar informações de maneira colaborativa será um dos aspectos fundamentais do projeto.

O presente relatório, abordará em detalhes a trajetória do desenvolvimento do site, destacando as tecnologias cuidadosamente selecionadas para sua construção. Além disso, apresentará uma análise minuciosa do processo de levantamento de requisitos, discutindo as etapas que levaram à identificação das funcionalidades cruciais para alcançar os objetivos mencionados. A implementação dessas funcionalidades também será descrita de maneira abrangente, fornecendo uma visão clara de como o projeto foi efetivamente realizado. O Projeto está distribuído da seguinte forma : Capítulo 1 – Introdução, Capítulo 2 – Engenharia de Software, Capítulo 3 – Banco de Dados, Capítulo 4 – Desenvolvimento, Capítulo 5 – Considerações Finais, Capítulo 6 – Referências Bibliográficas.

2 ENGENHARIA DE SOFTWARE

2.1 REQUISITOS

Requisitos de software são as diretrizes, descrições e critérios que definem o que um sistema de software deve fazer ou como deve se comportar para atender às necessidades dos usuários e aos objetivos do projeto. Eles podem ser divididos em duas categorias: requisitos funcionais e requisitos não funcionais (ALFF, 2023).

2.2 REQUISITOS FUNCIONAIS

Requisitos funcionais são critérios específicos que descrevem as funções que um sistema ou software deve realizar (ALFF, 2023).

Dentro do contexto das necessidades da comunidade acadêmica, é evidente que o método atual de acesso e compartilhamento de notícias enfrenta desafios e limitações. Portanto, é fundamental estabelecer uma lista de requisitos mínimos que devem ser considerados no desenvolvimento do website dedicado à centralização de notícias da Instituição.

Primeiramente, é imperativo que o *website* permita à Faculdade criar e compartilhar postagens de maneira eficaz. Essa funcionalidade essencial deve incluir a capacidade de adicionar, remover e atualizar postagens, garantindo um controle efetivo sobre o conteúdo publicado seria gestão da comunicação.

Além disso, é crucial que os discentes sejam capazes de realizar buscas por postagens com base em palavras-chave, o que agiliza a busca por informações específicas e incentiva os discentes a tratar e solucionar suas dúvidas ou problemas, portanto, melhorando a competência do usuário.

A disposição das postagens na página inicial do *website* é um aspecto importante para fornecer informações atualizadas de forma acessível. Portanto, é considerada importante a funcionalidade de exibição das postagens mais recentes na página inicial, permitindo aos alunos se manterem informados de maneira conveniente.

A Tabela 1 apresenta os requisitos funcionais deste projeto.

Identificação	Requisito Funcional	Prioridade
RF001	Conta da Faculdade: funcionalidade que permite à faculdade criar e compartilhar postagens.	Essencial
RF002	Adição, Remoção e Atualização de Postagens: funcionalidades que permitem a manipulação de postagens na plataforma.	Essencial
RF003	Busca por postagem: capacidade de pesquisar postagens com base em palavras-chave	Essencial
RF004	Exibição de postagens mais recentes na página inicial: funcionalidade de ordem de exibição de postagens	Importante

Tabela 1 - Lista de Requisitos Funcionais

2.3 REQUISITOS NÃO FUNCIONAIS

Requisitos não funcionais são critérios que não descrevem funcionalidades específicas do sistema, e sim as qualidades que o sistema deve possuir (ALFF, 2023). No caso apresentado, os requisitos não funcionais destacam a usabilidade, a Segurança, a Responsividade e o Design Visual. Eles visam garantir uma experiência de usuário satisfatória e a proteção das informações dos usuários.

Ao considerar a importância dos requisitos não funcionais, é crucial assegurar que o sistema não apenas atenda às funcionalidades necessárias, mas que também proporcione uma experiência eficaz e segura para os usuários. Desse modo, o sistema deve adotar uma abordagem centrada na experiência do usuário, fornecendo uma interface intuitiva e de fácil navegação. Essa ênfase na usabilidade é considerada essencial para garantir que os usuários possam aproveitar as funcionalidades do sistema sem dificuldades.

Além disso, a segurança dos dados dos usuários é uma prioridade. Nesse sentido, as senhas dos usuários devem ser criptografadas, garantindo a proteção das informações confidenciais. Essa medida é essencial para proteger a privacidade e a integridade das contas dos usuários.

Outro requisito fundamental é a Responsividade do site, o que permite que ele se adapte a diferentes dispositivos e tamanhos de tela. Isso é essencial para atender às diversas maneiras pelas quais os usuários acessam o site, seja em um computador *desktop*, *tablet* ou dispositivo móvel.

Além disso, o site deve aderir a uma paleta de cores específica, seguindo padrões de Design Visual. Isso é considerado essencial para garantir uma identidade visual coesa e uma experiência de usuário consistente.

A Tabela 2 apresenta os requisitos não funcionais deste projeto.

Identificação	Requisito não funcional	Categoria	Prioridade
RNF001	O sistema deve possuir uma interface intuitiva e de fácil navegação, permitindo que os usuários utilizem as funcionalidades sem dificuldades.	Usabilidade	Essencial
RNF002	As senhas dos usuários devem ser criptografadas.	Segurança	Essencial
RNF003	O site deve ser responsivo.	Distribuição	Essencial
RNF004	O site deve seguir uma paleta de cores específica.	Padrões	Essencial

Tabela 2 - Lista de Requisitos Não Funcionais

2.4 MODELAGEM

Na fase da modelagem é feita a documentação do sistema, apresentando diagramas que facilitam a compreensão do Projeto de forma padronizada. Será utilizada a linguagem de modelagem “*Unified Modeling Language*” (UML) para modelar os casos e diagramas.

2.5 DIAGRAMA DE CASO DE USO

Os diagramas de caso de uso descrevem um cenário de funcionalidades do ponto de vista do usuário. Dentro do diagrama, os atores são tratados e representados pelos bonecos, as funcionalidades são representadas pelos balões com a ação escrita, e as relações são representadas pelas linhas (LUCIDCHART, s.d.)

Os atores que interagem com o sistema são: a faculdade e o aluno. O sistema é um caso de uso explícito e se trata do sistema em si em que os casos de uso acontecem.

Usuário administrador: É um ator que representa os membros da instituição de ensino, como professores ou administradores, que interagem com o sistema de comunicação faculdade-aluno. Como ator principal, o usuário administrador desempenha um papel fundamental na criação, atualização e remoção de postagens no sistema. Além disso, ele é responsável por autenticar-se no sistema para acessar suas funcionalidades.

Usuário: é um ator que representa os estudantes que acessam o sistema de comunicação faculdade-aluno para visualizar as postagens. Embora não tenha a capacidade de adicionar ou modificar postagens, o usuário desempenha um papel importante ao buscar postagens específicas no sistema, facilitando o acesso a informações relevantes.

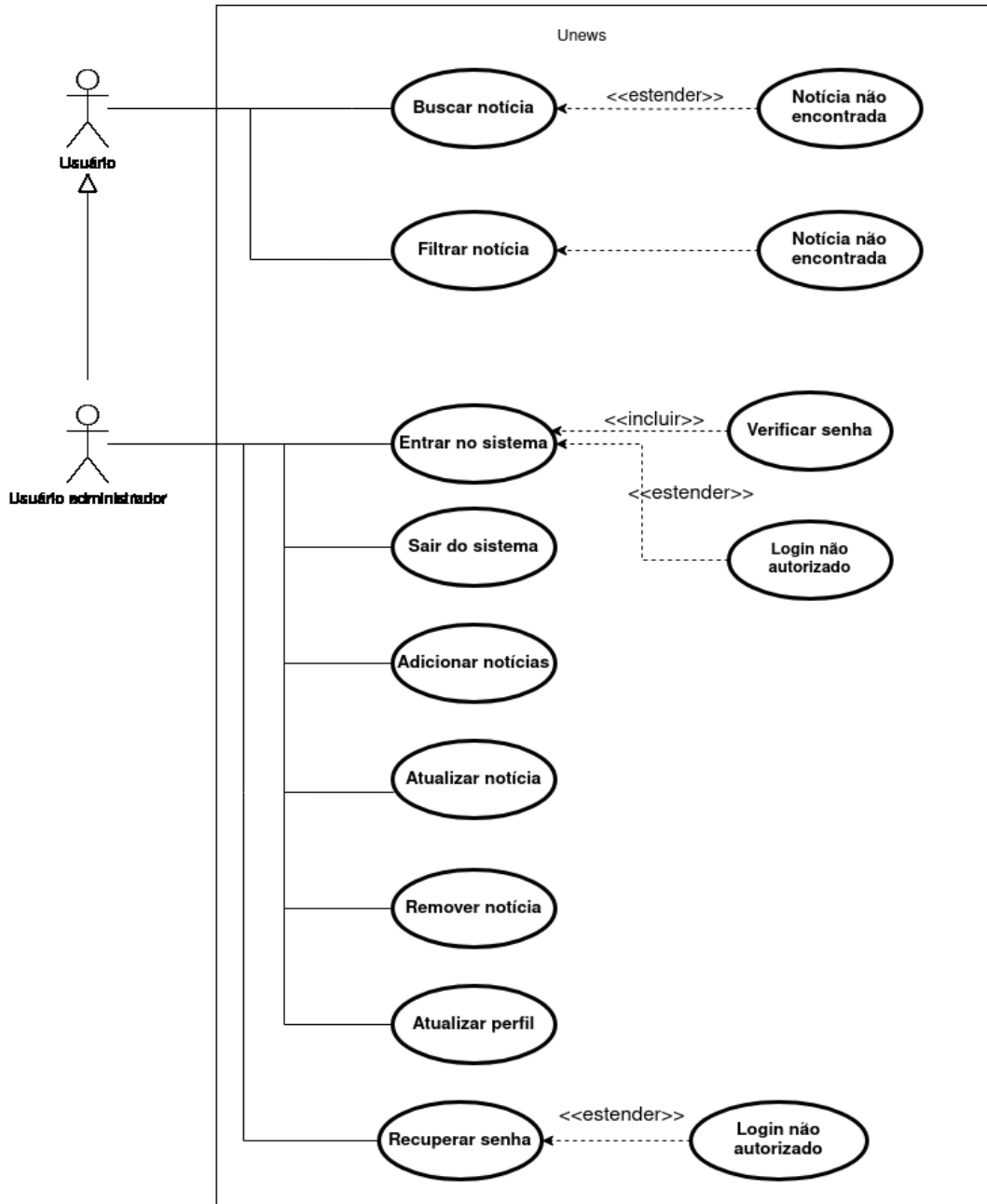


Figura 1 - Diagrama de Caso de Uso

Nome do caso de uso	Autenticação
Atores envolvidos	Usuário administrador

Objetivo	Este caso de uso descreve os passos do website
Ações do ator	Ações do Sistema
1. O usuário administrador deve preencher as informações de autenticação(email e senha)	
	2. O sistema se encarrega do processo de autenticação de usuário, checando o banco de dados, e retorna com uma resposta.
	3. Ao retornar com a resposta, o sistema redireciona para a página de tela inicial.
Validações	<ul style="list-style-type: none"> • Para que a autenticação seja realizada, o usuário deve entrar com um email e senha cadastrados no banco de dados.

Tabela 3 – Caso de uso “Autenticação”.

Nome do caso de uso	Adicionar Notícia
Atores envolvidos	Usuário administrador
Objetivo	Este caso de uso descreve os passos para adição de notícias no website
Ações do ator	Ações do Sistema
1. O usuário administrador deve clicar no botão “Adicionar notícia” presente em seu perfil de usuário.	
	2. O sistema carrega o formulário com os campos para adição da informação que será publicada
3. O usuário administrador deve preencher os campos e clicar no botão “Adicionar”	
	4. O sistema deve retornar uma mensagem e finalizar a ação.
Validações	<ul style="list-style-type: none"> • Para a criação de notícias o usuário deve ser validado. • Para a criação de notícias todos os campos do formulário devem estar preenchidos.

Tabela 4 – Caso de uso “Adicionar Notícia”.

Nome do caso de uso	Atualizar Notícia
Atores envolvidos	Usuário administrador
Objetivo	Este caso de uso descreve os passos para atualização de notícias no website
Ações do ator	Ações do Sistema
1. O usuário administrador deve clicar no botão “Atualizar” presente nas notícias já publicadas.	
	2. O sistema carrega o formulário com os campos para atualização das informações já publicadas.
3. O usuário administrador deve preencher ao menos um campo.	
	4. O sistema deve retornar uma mensagem e finalizar a ação.
Validações	<ul style="list-style-type: none"> • Para a atualização de notícias o usuário deve ser validado e dono da notícia. • Para a atualização da notícia ao menos um campo do formulário deve estar preenchido.

Tabela 5 – Caso de uso “Atualizar Notícia”.

Nome do caso de uso	Remover Notícia
Atores envolvidos	Usuário administrador
Objetivo	Este caso de uso descreve os passos para remover notícias no website
Ações do ator	Ações do Sistema
1. O usuário administrador deve clicar no botão “Deletar” presente nas notícias já publicadas.	
	2. O sistema deve apresentar uma confirmação de deleção
3. O usuário administrador deve confirmar a deleção clicando no botão “Sim”	

	4. O sistema deve retornar uma mensagem e finalizar a ação.
Validações	<ul style="list-style-type: none"> • Para a remoção de notícias o usuário deve ser validado e dono da notícia. • Para a remoção da notícia o usuário deve confirmar a ação.

Tabela 6 – Caso de uso “Remover Notícia”.

Nome do caso de uso	Buscar Notícia
Atores envolvidos	Usuário
Objetivo	Este caso de uso descreve os passos para busca de notícias no website
Ações do ator	Ações do Sistema
1. O usuário deve digitar o título da notícia no campo “Buscar notícia” presente na barra de navegação.	
	2. O sistema deve retornar todas as notícias com o título buscado presentes no banco de dados.
Validações	<ul style="list-style-type: none"> • Para a busca da notícia, o título buscado não precisa ser idêntico, mas deve coincidir ou estar contido nos títulos das notícias do banco de dados. • Para a busca de notícia, o campo de busca não deve estar vazio antes de enviar a consulta.

Tabela 7 – Caso de uso “Buscar Notícia”.

Nome do caso de uso	Filtrar Notícia
Atores envolvidos	Usuário
Objetivo	Este caso de uso descreve os passos para busca de notícias no website pelo filtro de categorias
Ações do ator	Ações do Sistema
1. O usuário deve selecionar a categoria desejada presente no filtro.	

	2. O sistema deve retornar todas as notícias com a categoria presentes no banco de dados.
Validações	<ul style="list-style-type: none"> • Para filtrar notícias, pelo menos uma categoria deve ser selecionada antes de enviar a consulta.

Tabela 8 – Caso de uso “Filtrar Notícia”.

2.6 DIAGRAMA DE CLASSES

O Diagrama de Classes é uma ferramenta estática que desempenha um papel fundamental na descrição da estrutura de um sistema. Ele foca em destacar as classes, suas propriedades, operações e interações entre objetos, oferecendo uma visão abrangente da organização do sistema. Essa representação visual ajuda a compreender a arquitetura do sistema de forma mais clara, contribuindo para uma visão sólida de como as partes do sistema se relacionam entre si (LUCIDCHART, s.d.).

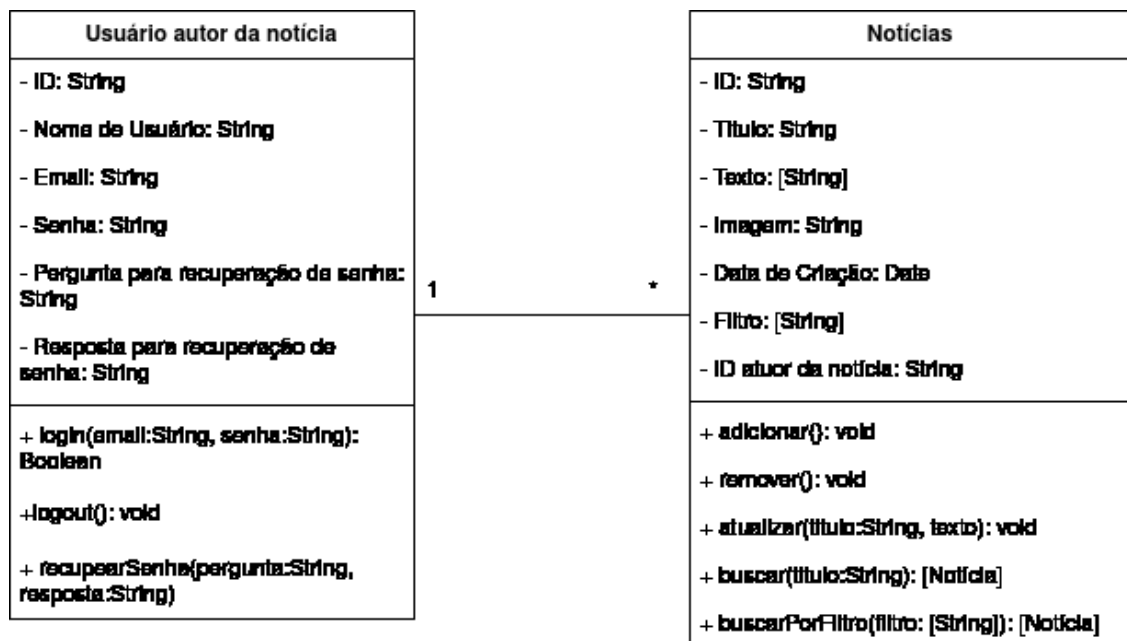


Figura 2 – Diagrama de Classes.

2.7 DIAGRAMA DE ATIVIDADES

O Diagrama de Atividade desempenha um papel crucial na representação visual dos processos de utilização do sistema. Ele age de forma semelhante a um fluxograma de

controle, proporcionando uma representação intuitiva dos passos de uma atividade para outra. Essa abordagem torna a compreensão dos processos do sistema mais acessível e facilita a comunicação entre a equipe. Dessa forma, o diagrama de atividade se torna uma ferramenta valiosa para descrever com clareza como o sistema é utilizado (LUCIDCHART, s.d.).

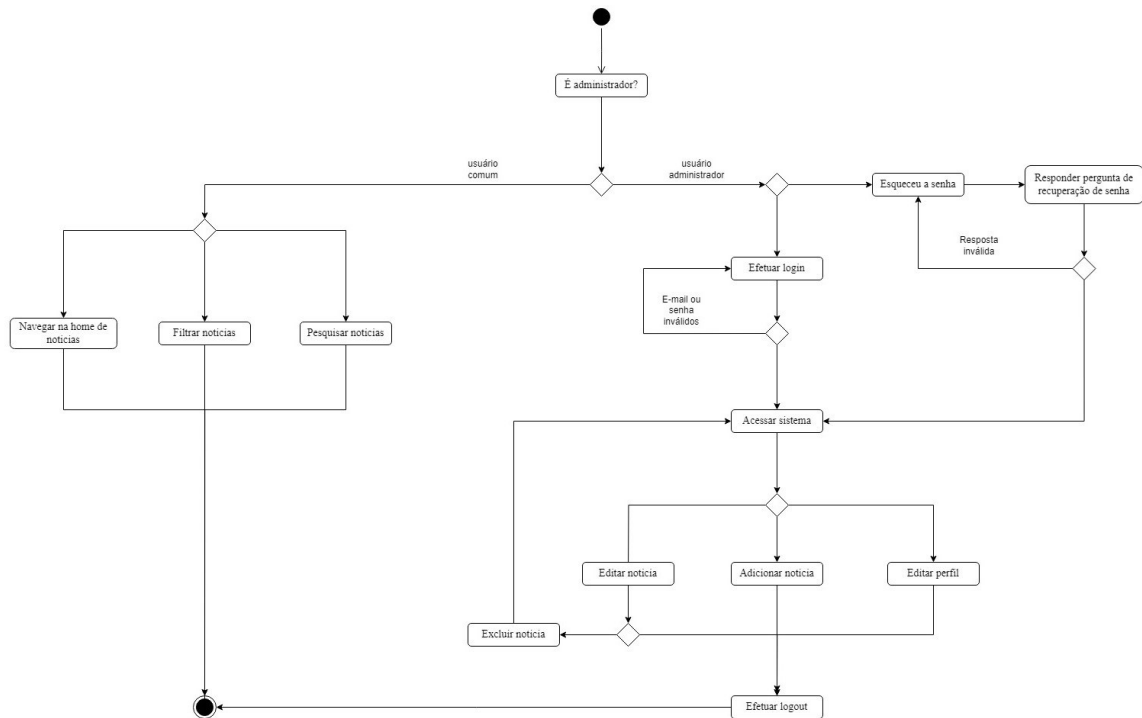


Figura 3 – Diagrama de Atividades.

2.8 PLANO DE TESTES

Para garantir a integridade do sistema, é crucial incorporar testes para identificar e solucionar quaisquer erros que possam surgir devido a lacunas na validação. Levando em consideração o Diagrama de Caso de Uso do Projeto, nesta implementação segue de perto cada cenário de uso, garantindo que todas as ações sejam validadas para evitar possíveis falhas.

A seguir estão as Tabelas 9 à 15 com o plano de testes de seus respectivos casos de uso:

Caso de Teste – Login	
Cenário: Abrir tela de login	Dado que o usuário está na tela inicial do sistema Quando o usuário clica no botão "Entrar" Então é exibida a tela de login E é verificada a exibição da logo do projeto na tela de login

Cenário: Login com campos em branco	Dado que o usuário está na tela de login Quando o usuário clica no botão "Entrar" sem informar o e-mail e a senha Então é exibido o alerta "Entre com o seu e-mail e senha."
Cenário: Login sem informar a senha	Dado que o usuário está na tela de login Quando o usuário informa somente o e-mail E clicar no botão "Entrar" Então é exibido o alerta "O campo Senha é obrigatório."
Cenário: Login sem informar o e-mail	Dado que o usuário está na tela de login Quando o usuário informa somente a senha E clicar no botão "Entrar" Então é exibido o alerta "O campo Email é obrigatório."
Cenário: Login com e-mail inválido	Dado que o usuário está na tela de login Quando o usuário informa um e-mail inválido E informar uma senha válida E clicar no botão "Entrar" Então é exibido o alerta "Digite um endereço de e-mail válido."
Cenário: Login com senha inválida	Dado que o usuário está na tela de login Quando o usuário informa um e-mail válido E informar uma senha inválida E clicar no botão "Entrar" Então é exibido o alerta "A senha deve ter pelo menos 6 caracteres."
Cenário: Verificar atribuições do usuário	Dado que o usuário está logado no sistema Quando o sistema carrega as funcionalidades do usuário Então somente as funcionalidades com permissões atribuídas devem ser exibidas
Cenário: Login em execução	Dado que o usuário está na tela de login Quando o usuário clica no botão "Entrar" Então todos os componentes devem estar desabilitados para impossibilitar a navegação e evitar erros fatais.

Tabela 9 - Caso de teste "Login"

Caso de Teste – Adicionar Notícia	
Cenário: Adicionar uma nova notícia	Dado que o usuário está logado no sistema como a faculdade Quando o usuário clica no botão "Adicionar Notícia" e preenche os campos obrigatórios (Título, Texto, Imagem) Então a notícia é adicionada com sucesso
Cenário: Adicionar Notícia sem preencher campos obrigatórios	Dado que o usuário está logado no sistema como a faculdade Quando o usuário clica no botão "Adicionar Notícia" sem preencher todos os campos obrigatórios Então é exibido um alerta indicando que todos os campos obrigatórios devem ser preenchidos

Tabela 10 - Caso de teste "Adicionar Notícia"

Caso de Teste – Atualizar Notícia	
Cenário: Atualiza uma notícia existente	Dado que o usuário está logado no sistema como a faculdade Quando o usuário clica no botão “Editar” presente na notícia deseja Então uma nova tela é aberta com campos para edição E o usuário faz as alterações desejadas (Título, Texto, Imagem) Então a notícia é atualizada com sucesso com as novas informações
Cenário: Atualizar notícia sem preencher ao menos um campo	Dado que o usuário está logado no sistema como a faculdade Quando o usuário tenta atualizar uma notícia sem preencher ao menos um campo Então é exibido um alerta indicando que ao menos um campo deve ser preenchido

Tabela 11 - Caso de teste “Atualizar Notícia”

Caso de Teste – Remover Notícia	
Cenário: Remover uma notícia existente	Dado que o usuário está logado no sistema como uma faculdade Quando o usuário clica em “Editar” Então uma nova tela é aberta E o usuário clica no botão “Remover” E o usuário confirma a deleção Então a notícia é removida com sucesso do sistema

Tabela 12 - Caso de teste “Remover Notícia”

Caso de Teste – Visualizar Notícia	
Cenário: Visualizar uma notícia clicando no título	Dado que o usuário está na página inicial do sistema Quando o usuário clica no título da notícia Então a notícia será exibida em um Pop-Up Modal
Cenário: Visualizar uma notícia clicando em “Ler Mais”	Dado que o usuário está na página inicial do sistema Quando o usuário clica em “Ler Mais” de uma notícia Então a notícia será exibida em um Pop-Up Modal
Cenário: Visualizar uma notícia clicando no ícone de seta de uma notícia	Dado que o usuário está na página inicial do sistema Quando o usuário clica no ícone de seta de uma notícia Então a notícia será exibida em um Pop-Up Modal

Tabela 13 - Caso de teste “Visualizar Notícia”

Caso de Teste – Buscar Notícia	
Cenário: Buscar uma notícia pelo título	Dado que o usuário está na página inicial do sistema Quando o usuário clica no campo de busca E o usuário digita o título E o usuário tecla enter Então as notícias com o título inserido para busca devem ser retornadas
Cenário: Buscar uma notícia por fragmento de título	Dado que o usuário está na página inicial do sistema Quando o usuário clica no campo de busca E o usuário digita uma parte do título da notícia E o usuário tecla enter Então as notícias com o fragmento do título inserido para busca devem ser retornadas
Cenário: Buscar notícia por título inexistente	Dado que o usuário está na página inicial do sistema Quando o usuário clica no campo de busca E o usuário digita o título de notícia não existente E o usuário tecla enter Então nenhuma notícia deve ser retornada

Tabela 14 - Caso de teste "Buscar Notícia"

Caso de Teste – Logout	
Cenário: Realizar Logout	Dado que o usuário está logado no sistema Quando o usuário clica no botão "Sair" Então o sistema encerra a sessão do usuário E o usuário é redirecionado para a página inicial

Tabela 15 - Caso de teste "Logout"

3 BANCO DE DADOS

Nesse projeto duas tabelas principais compõem o Banco de Dados, sendo elas a tabela *Faculdade* que representa o usuário administrador, e a tabela *Notícias* que representa as notícias criadas e exibidas no site.

3.1 DIAGRAMA DE ENTIDADE E RELACIONAMENTO

Diagrama de Entidade-Relacionamento é uma ferramenta visual que representa a estrutura do Banco de Dados. Ele usa retângulos para mostrar entidades e linhas para indicar como essas entidades se relacionam. Isso ajuda na organização e no entendimento de como os dados estão conectados ao sistema.(LUCIDCHART, s.d)

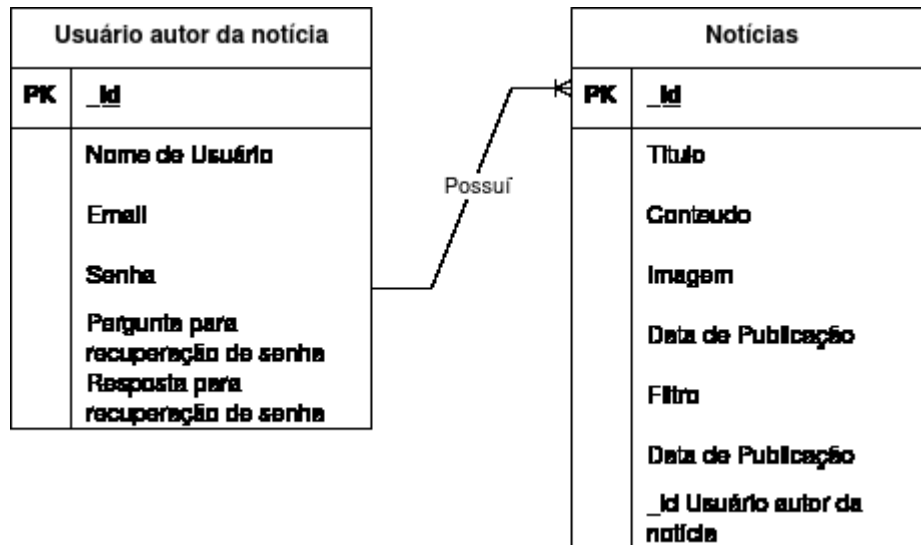


Figura 4 – Diagrama de Entidade e Relacionamento.

3.2 DICIONÁRIO DE DADOS

Um Dicionário de Dados é um documento que contém uma lista de todos os elementos de dados relevantes para um sistema. Isso inclui informações detalhadas sobre cada elemento de dados, como seu nome, tipo, formato e significado. Em essência, o dicionário fornece um catálogo completo das informações que o sistema utiliza.

Entidade: <i>Usuário Autor da notícia</i>			
Atributo	Classe	Domínio	Descrição
ID	Simple	Texto	identificador único da faculdade
email	Simple	Texto	endereço de e-mail da faculdade
nome de usuário	Simple	Texto	nome de usuário da faculdade
senha	Simple	Texto	senha do usuário
pergunta para recuperação de senha	Simple	Texto	pergunta para recuperar senha do usuário
resposta para recuperação de senha	Simple	Texto	resposta para a pergunta de recuperação de senha

Tabela 16 – Dicionário de Dados da entidade Faculdade.

Entidade: *Notícias*

Atributo	Classe	Domínio	Descrição
ID	Simple	Texto	Identificador único da notícia
Título	Simple	Texto	Título da notícia
Texto	Simple	Conjunto de textos	Conteúdo textual da notícia
Imagem	Simple	Texto	URL ou caminho da imagem da notícia
Data de Criação	Simple	Data e Hora	Data de criação da notícia
Filtro	Simple	Conjunto de textos	Filtros associados à notícia
Usuário Autor da Notícia	Simple	Texto	ID da faculdade autora da notícia

Tabela 17 - Dicionário de Dados da entidade Notícias.

4 DESENVOLVIMENTO

Neste capítulo, detalha o processo de desenvolvimento, com o objetivo de fornecer uma visão completa do trabalho realizado durante a criação desta solução, serão apresentadas as etapas, métodos e metodologias aplicadas. Esta análise permite compreender as fases de Planejamento, Implementação e Desenvolvimento da aplicação, além das decisões tomadas ao longo do desenvolvimento, visando cumprir os requisitos estabelecidos e garantir uma experiência de utilização eficiente.

4.1 IDENTIFICAÇÃO

No contexto acadêmico em constante expansão, os alunos enfrentam o desafio de acompanhar as notícias e atualizações relevantes da Faculdade. A falta de uma plataforma centralizada para a disseminação de informações acadêmicas tem se mostrado uma barreira significativa. Os estudantes se vêem sobrecarregados com a necessidade de pesquisar em diversas fontes em busca de notícias, muitas vezes espalhadas e desorganizadas, pela Instituição.

A ausência de uma Política de Comunicação no ambiente da Faculdade impõe uma carga adicional sobre os discentes, que buscam eficiência na obtenção de notícias relevantes. É evidente a necessidade de desenvolver um *website* voltado para atender às necessidades da comunidade acadêmica da instituição. Esse *website* simplificaria o processo de busca e acesso a notícias da Faculdade, atendendo aos requisitos identificados junto à comunidade estudantil.

O desenvolvimento do sistema foi caracterizado por uma abordagem ágil, baseada nos princípios do Manifesto Ágil. A equipe concentrou-se nas necessidades do usuário, abraçando uma abordagem iterativa e incremental no desenvolvimento das funcionalidades. À medida que o projeto progredia, o processo incluía a autoavaliação contínua da equipe para garantir que o sistema atendesse aos requisitos e objetivos estabelecidos.

Embora tenham sido evitadas metodologias formais estritas, o desenvolvimento incorporou práticas ágeis, como a priorização dinâmica de requisitos e uma comunicação constante entre a equipe. Essa abordagem conferiu à equipe a liberdade e agilidade necessárias para a eficaz criação do *website*, permitindo uma resposta dinâmica às mudanças e um alinhamento constante com as necessidades do projeto.

4.2 DESCRIÇÃO DA SOLUÇÃO PROPOSTA

Com base nos requisitos definidos, a solução proposta será implementada utilizando uma abordagem técnica que se baseia em tecnologias modernas, com o propósito de fornecer uma plataforma eficiente de centralização de notícias acadêmicas.

Para atender ao requisito de que a Faculdade possa criar e compartilhar postagens, será criada uma conta da Faculdade. Essa conta terá acesso ao painel de administração, onde poderá adicionar, deletar e atualizar notícias.

O desenvolvimento das funcionalidades de "Adição, Remoção e Atualização de Postagens" será realizada no painel de administração, permitindo que a faculdade gerencie o conteúdo de forma eficaz. Isso garante que as informações estejam sempre atualizadas.

Para atender ao requisito de "Busca por Postagem," será implementada uma funcionalidade de pesquisa por meio de palavras-chave, garantindo que os alunos possam encontrar rapidamente as informações desejadas.

Com relação ao requisito de "Exibição de Postagens mais Recentes na Página Inicial," a solução incluirá a exibição das postagens mais recentes na página inicial do website. Isso proporcionará aos usuários uma experiência de usuário eficiente, onde eles podem acessar as notícias mais recentes de forma conveniente.

Para garantir que a solução seja bem-sucedida em termos de usabilidade, segurança, responsividade e design visual, as seguintes abordagens serão implementadas:

A interface do website será projetada de forma intuitiva, com navegação fácil para que os usuários possam aproveitar as funcionalidades sem dificuldades.

A segurança dos dados dos usuários será priorizada, com todas as senhas dos usuários criptografadas para garantir a proteção das informações confidenciais.

O website será desenvolvido de forma responsiva, adaptando-se a diferentes dispositivos e tamanhos de tela, proporcionando uma experiência de usuário consistente em todas as plataformas.

A identidade visual do website seguirá uma paleta de cores específica, aderindo a padrões de design.

Com esta solução, os requisitos funcionais e não funcionais do Projeto serão atendidos, proporcionando uma plataforma eficaz para centralização de notícias acadêmicas da Faculdade

4.3 FERRAMENTAS

Este capítulo visa descrever as ferramentas utilizadas durante o desenvolvimento do Projeto.

4.3.1 MERN STACK

A MERN Stack é uma pilha tecnológica que se baseia em diversas tecnologias. Ela inclui o *MongoDB* como Banco de Dados, o *Express JS* como *Framework Web* no lado do servidor, o *React JS* para construção de interfaces no lado do cliente e o *Node JS* como servidor web. Essas ferramentas se unem para criar uma plataforma versátil e poderosa para o desenvolvimento de aplicativos web (MONGODB, s.d.).

A Pilha é baseada em uma comunicação constante entre o lado do cliente, o servidor e o Banco de Dados, como exemplificado, conforme figura 5 abaixo :

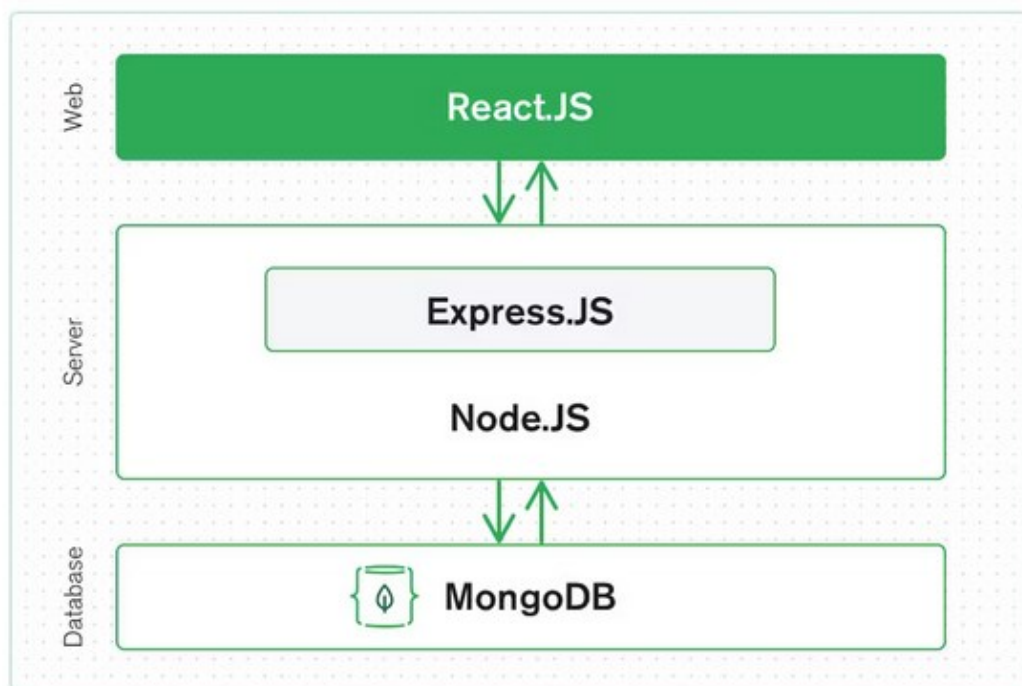


Figura 5 – Arquitetura da MERN Stack (Fonte: MongoDB).

4.3.2 MONGO DB

MongoDB é um sistema de gerenciamento de Banco de Dados *NoSQL* conhecido por sua natureza orientada a documentos. Em vez de usar tabelas e linhas, como um Banco de Dados relacional, o *MongoDB* armazena dados em documentos semelhantes ao *JSON*. Cada documento pode conter dados em um formato flexível, ideal para aplicativos com padrões de dados dinâmicos ou mutáveis. É amplamente utilizado para armazenar, recuperar e gerenciar dados em aplicativos web e móveis (MONGODB, s.d.).

4.3.3 EXPRESS JS

O *Express.js* opera no ambiente do servidor *Node.js* e é uma plataforma web ágil e eficaz, oferecendo um modelo robusto para encaminhamento de URLs e gerenciamento de solicitações e respostas HTTP.

Quando são realizadas solicitações no *front-end* usando *React.js*, seja através de *XMLHttpRequest* (XHR) ou por meio de solicitações GET ou POST, estabelece-se uma conexão com as funcionalidades fornecidas pelo *Express.js*, que impulsionam o aplicativo. Essas funcionalidades, por sua vez, fazem uso do driver *Node.js* para *MongoDB*, aproveitando tanto retornos de chamada quanto promessas, a fim de acessar e atualizar os dados no banco de dados *MongoDB* (MONGODB, s.d.).

4.3.4 REACT JS

O *React JS* é um *Framework JavaScript* empregado na criação dinâmica de aplicativos no lado do cliente. Ele opera por meio de componentes que se conectam aos dados do servidor e são renderizados como HTML. Essa abordagem permite uma experiência mais interativa e ágil para os usuários, tornando o desenvolvimento de aplicações web ainda mais envolvente e eficiente (MONGODB, s.d.).

4.3.5 NODE JS

Node.js atua como o servidor na *MERN Stack*. É comum usar o *Express.js*, um *Framework Node.js*, para criar rotas e manipular solicitações HTTP. O *Node.js* se comunica com o *MongoDB* para buscar e enviar dados conforme necessário. Ele age como a camada intermediária que recebe solicitações do *Front-End React*, processa essas solicitações e interage com o Banco de Dados *MongoDB* para recuperar ou atualizar os dados. O *Node.js* é altamente eficiente. Ele é conhecido por ser orientado a eventos, o que o torna eficiente ao trabalhar com conexões simultâneas (MONGODB, s.d.).

4.3.6 ROBOT FRAMEWORK

O *Robot Framework* é um *Framework* de automação baseado em *Python* que é altamente flexível e orientado por palavras-chave. É aplicado em testes de aceitação e automação de processos robóticos. Além disso, é adequado para ambientes distribuídos e heterogêneos, onde a automação envolve o uso de diferentes tecnologias e interfaces.

Uma das características notáveis do *Robot Framework* é a sintaxe de tabela amigável para escrever casos de teste de forma consistente. Ele também oferece a capacidade de criar palavras-chave de alto nível reutilizáveis a partir de palavras-chave existentes. Os relatórios de resultados são gerados em formato HTML, facilitando a leitura.

O *Robot Framework* é independente de plataforma e aplicação, o que o torna versátil. Além disso, fornece uma API de biblioteca simples para criar bibliotecas de teste personalizadas que podem ser implementadas nativamente em Python. Também é possível utilizar sua interface de linha de comando e arquivos de saída baseados em XML para integração em sua infraestrutura de construção existente, como sistemas de integração contínua.

Ele suporta uma variedade de tipos de testes, incluindo testes de aplicativos web, testes de APIs REST, testes de aplicativos móveis, execução de processos, conexão a sistemas remotos via Telnet ou SSH, e muito mais. Além disso, oferece suporte para a criação de casos de teste baseados em dados e inclui recursos integrados para o uso de variáveis, o que é particularmente útil ao testar em diferentes ambientes.

O *Robot Framework* também permite marcar e categorizar casos de teste para facilitar a seleção e execução. Ele pode ser facilmente integrado ao controle de origem, pois os conjuntos de teste são apenas arquivos e diretórios que podem ser versionados juntamente com o código de produção. Além disso, oferece funcionalidades de configuração e desmontagem em nível de caso de teste e conjunto de testes. Sua arquitetura modular suporta a escrita de testes mesmo para aplicativos com várias interfaces. Em resumo, o *Robot Framework* é uma ferramenta versátil e poderosa que facilita a automação de testes em uma variedade de cenários e ambientes (ROBOT FRAMEWORK, s.d.).

4.3.7 GIT

O Git, uma ferramenta de controle de versão distribuída, gratuita e de código aberto, é uma peça fundamental no desenvolvimento de software. Com sua capacidade de gerenciar Projetos de todos os tamanhos de maneira ágil e eficiente, o Git se destaca como uma escolha indispensável para qualquer equipe de desenvolvimento.

A flexibilidade do Git torna-o adequado tanto para pequenos Projetos pessoais quanto para grandes empreendimentos empresariais. Sua versatilidade e eficiência garantem que projetos de todos os tamanhos possam ser gerenciados de forma eficaz. Essa abordagem garante que mesmo projetos enormes possam ser tratados com eficiência, mantendo a integridade do código e facilitando a colaboração.

Uma das maiores vantagens do Git é a sua natureza de código aberto. Isso significa que é acessível a todos, o que impulsiona a inovação e o desenvolvimento da comunidade. Com uma ampla variedade de ferramentas e serviços que se integram ao Git, permite personalizar a experiência de desenvolvimento para atender às necessidades específicas do Projeto (GIT, s.d.).

4.3.8 GITHUB

O GitHub é uma plataforma de hospedagem de código que desempenha um papel essencial na gestão de versões e colaboração em Projetos de desenvolvimento de software, ele permite que as equipes de projetos compartilhem seus códigos, acompanhem as mudanças feitas por outros membros da equipe, proporcionando a criação de software de maneira colaborativa e eficiente (GITHUB, s.d.)

4.4 BACKEND

O *Backend* é responsável por lidar com as solicitações do cliente, processar dados, interagir com o banco de dados e fornecer respostas apropriadas. O *Backend* desse Projeto é dividido em quatro componentes principais: rotas, *middlewares*, *controllers* e *services*.

4.4.1 ROTAS

As Rotas desempenham um papel crucial na comunicação entre o cliente (client) e o servidor (server). Por meio delas, o cliente pode executar operações como adicionar, remover, atualizar e ler dados do Banco de Dados, utilizando solicitações HTTP (REACT ROUTER, s.d.).

No contexto do sistema *Unews*, as rotas desempenham um papel fundamental na manipulação de notícias. Abaixo, descrevemos as principais rotas e suas funcionalidades:

- **Rota "url/news/" (Método GET):** Esta rota é responsável por recuperar todas as notícias disponíveis no sistema. O controlador correspondente busca as notícias no banco de dados e responde com a lista completa.
- **Rota "url/news/" (Método POST):** Esta rota é acionada quando uma nova notícia é criada no frontend. Os dados da notícia são enviados para esta rota, e o controlador associa esses dados a um novo registro no banco de dados.
- **Rota "url/news/:id" (Método PATCH):** Esta rota é acessada ao passar um ID específico e dados atualizados. O controlador responsável encontra a notícia correspondente ao ID e atualiza suas informações com os novos dados.
- **Rota "url/news/:id" (Método DELETE):** Esta rota é usada para solicitar a exclusão de uma notícia. O controlador encontra a notícia pelo ID e a remove do banco de dados.

4.4.2 MIDDLEWARES

Os *middlewares* são funções intermediárias que operam entre as rotas e os *controllers*, desempenhando um papel fundamental na construção de uma aplicação. O Projeto inclui suas principais funções e garantem que os usuários estejam autorizados a realizar determinadas ações, como a criação, atualização e exclusão de notícias, além de assegurar a formatação adequada dos dados enviados pelo cliente. Caso um usuário não esteja autorizado ou os

dados estejam formatados de maneira incorreta, o *Middleware* responderá ao cliente com uma mensagem de erro correspondente ao problema identificado (MICROSOFT AZURE, s.d.).

4.4.3 CONTROLLERS

Os controladores desempenham um papel fundamental ao receber as requisições enviadas através das rotas, processá-las e fornecer uma resposta apropriada. Em geral, cada rota está associada a um controlador específico. Por exemplo, o controlador de notícias é responsável por lidar com solicitações relacionadas à busca, criação, atualização e exclusão de notícias. Para executar essas operações no banco de dados, o controlador interage com o serviço correspondente (MICROSOFT LEARN, 2023).

4.4.4 SERVICES

Os serviços são responsáveis pela comunicação direta com o Banco de Dados. Através deles, o controlador pode interagir com o banco de dados para obter dados ou efetuar alterações (ANGULAR, s.d.).

4.5 FRONTEND

Para o desenvolvimento do *Frontend* foi escolhido usar o *Framework React* pois por ser focado na criação de interfaces de usuário ele beneficia com a possibilidade da criação de componentes, e com sua extensão de *Javascript*, o *JSX*, proporciona uma versatilidade na hora de desenvolver as interfaces permitindo inserir a lógica de programação direto no componente HTML.

O Projeto é dividido em cinco principais páginas, sendo elas, Home, Login, Perfil, Create e Edit. A página Home é onde está todo o conteúdo principal, todas as notícias se encontram na Home separadas por um componente chamado Card, em Login está presente o formulário para o login do usuário admin. As páginas que são acessadas apenas se o usuário estiver logado são a página de Perfil, para editar dados do usuário caso necessário, e as páginas Create e Edit para criar e editar as notícias, ambas as páginas possuem o mesmo formulário porém com a diferença de a página Edit já possuir os dados preenchidos, facilitando a manutenção de notícias já existentes.

4.6 INTEGRAÇÃO

A conexão do *Frontend* com o *Backend* foi realizada utilizando a biblioteca *Axios* que permite a integração do React com qualquer API (*Interface de Programação de Aplicação*), usando o *Axios* para integrar o *Frontend* com o *Backend*, foi criado um serviço para o usuário e para as notícias, separando cada rota em funções específicas que direcionam para o *Backend*. Esse projeto também implementa a utilização de *cookies* para o controle de usuário após a autenticação, utilizando a biblioteca *JavaScript Cookie JS* para o gerenciamento do cookie do

usuário, através dos cookies podemos ter controle de segurança principalmente quando o *Frontend* chama uma rota que precise de autorização.

4.7 INTERFACES DE USUÁRIO E USABILIDADE

O Design do *frontend* do Projeto foi cuidadosamente planejado para ser responsivo. O objetivo era fornecer aos usuários uma experiência consistente, independentemente do dispositivo que estivessem utilizando, seja um *Desktop*, *Tablet* ou *Smartphone*. A figura 6 apresenta a tela inicial do sistema, que demonstra a responsividade do design. As opções da barra de navegação, como login, adicionar notícias, buscar notícias e perfil, são acessíveis a partir de um menu hambúrguer. Isso garante que a experiência do usuário seja otimizada em diferentes dispositivos.

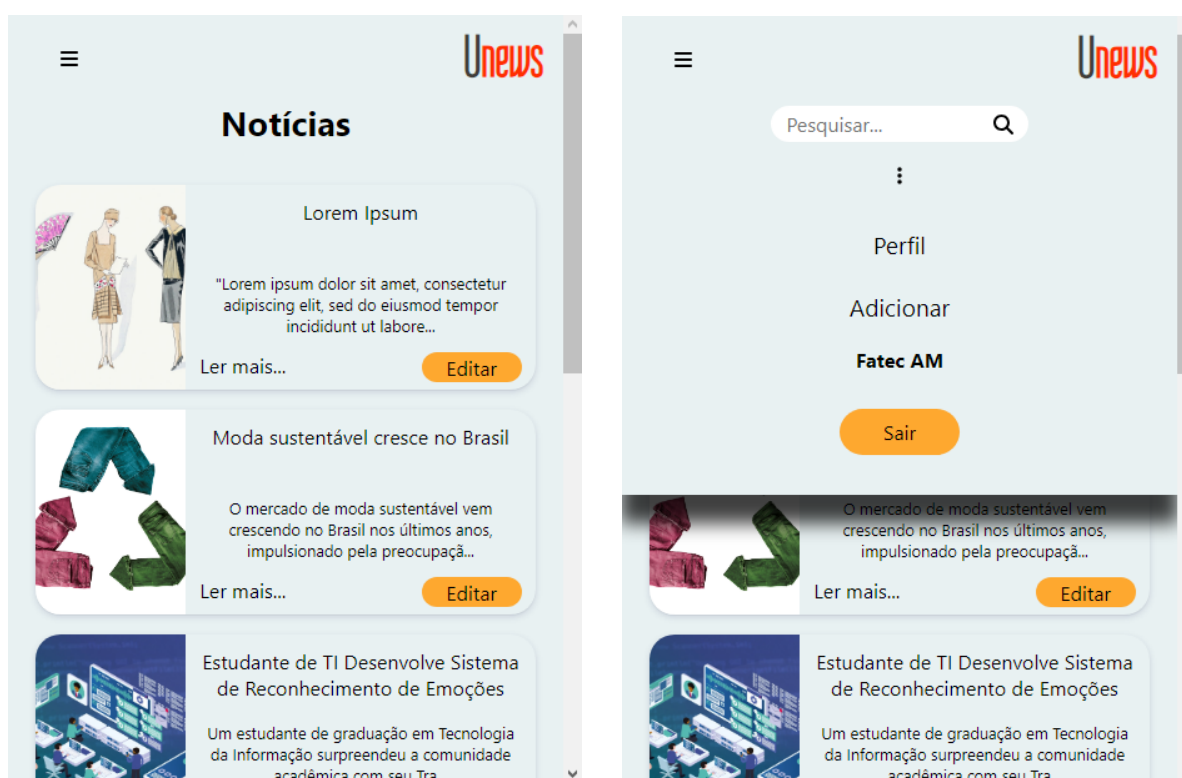


Figura 6 - Responsividade do sistema

Nas figuras 7 e 8, pode-se observar a tela inicial para usuários não administradores. Nessa tela, as notícias mais recentes são exibidas, e os usuários têm a opção de acessá-las por meio do título, do botão 'Ler Mais' ou da seta. Ao abrir uma notícia, ela é exibida em formato de pop-up, proporcionando uma visualização mais detalhada.

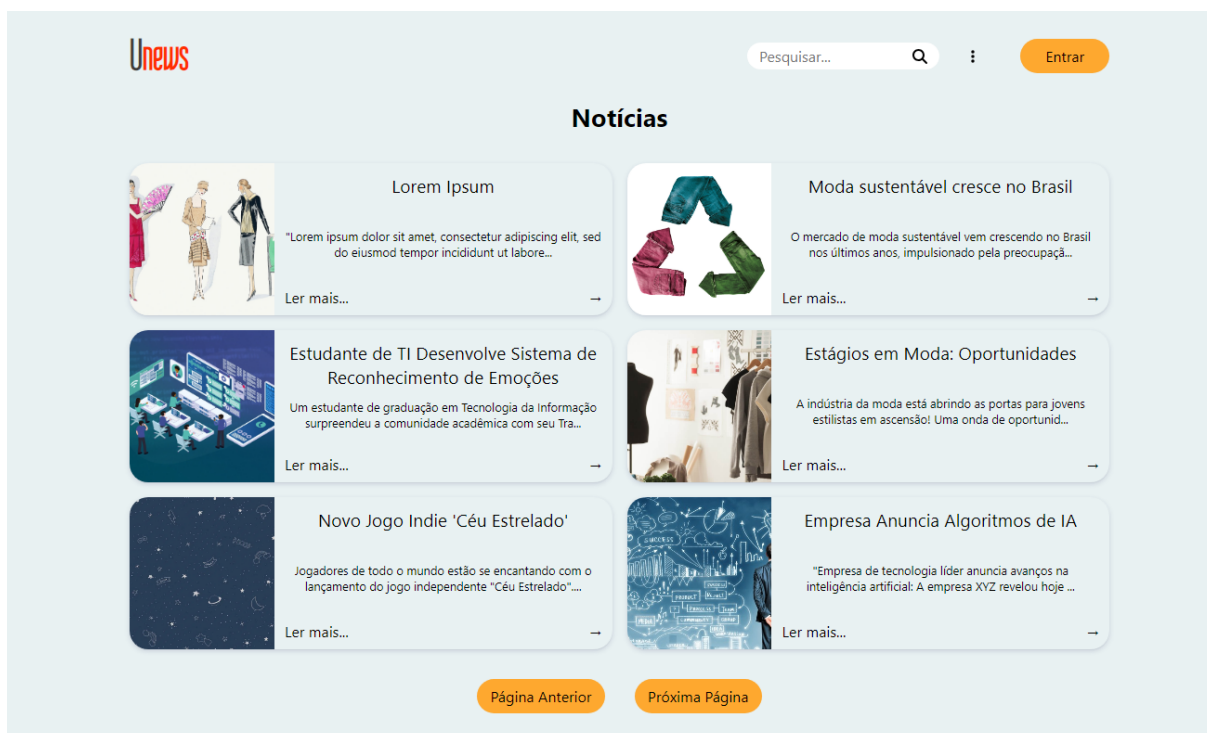


Figura 7 - Tela inicial para usuários não administradores

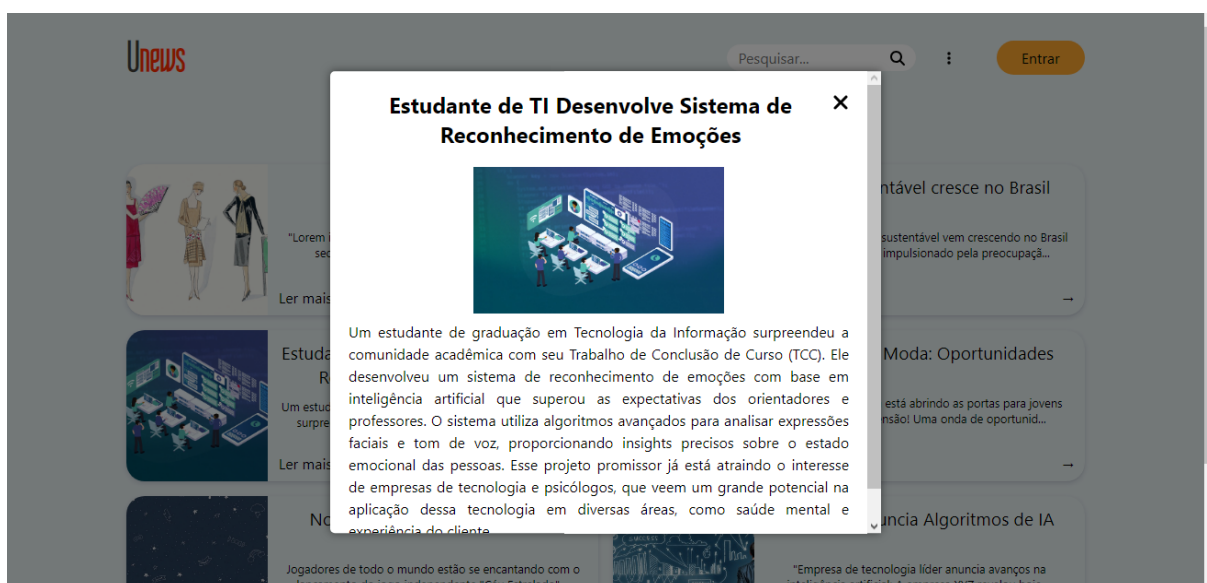


Figura 8 - Acesso ao conteúdo da notícia

A figura 9, exibe a tela inicial para administradores, onde a seta é substituída pelo botão 'Editar'. Ao clicar na seta, é aberta outra tela, a figura 10, onde as informações da notícia são carregadas e podem ser facilmente alteradas.

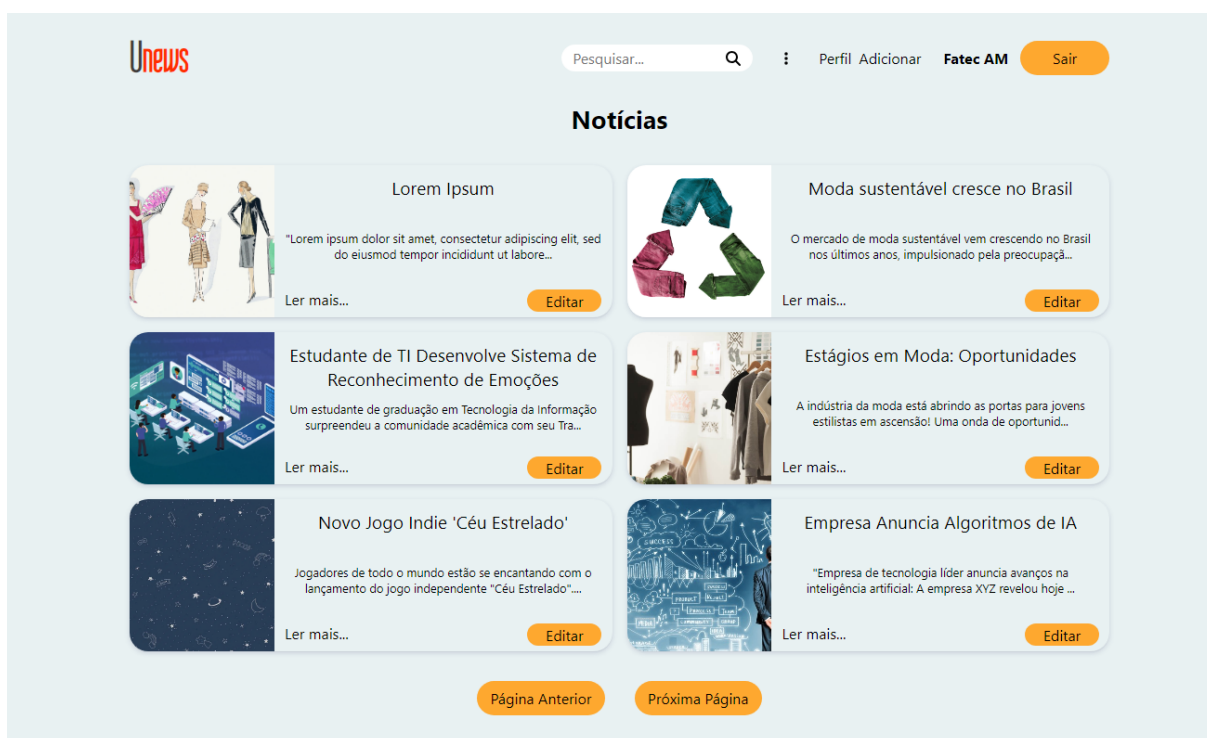


Figura 9 - Tela inicial para usuários administradores

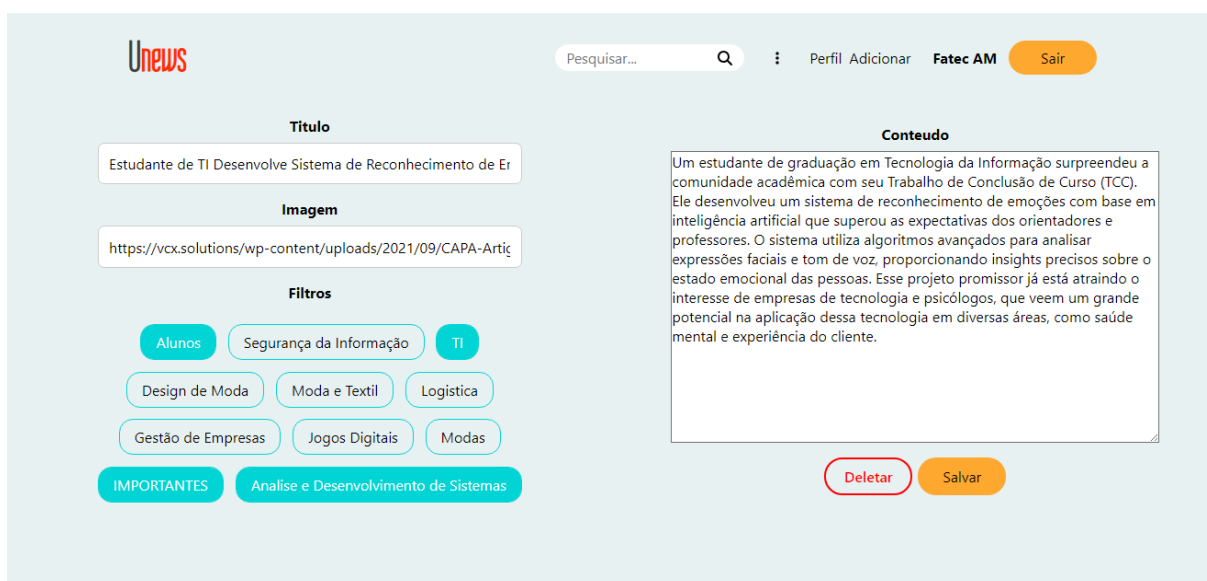
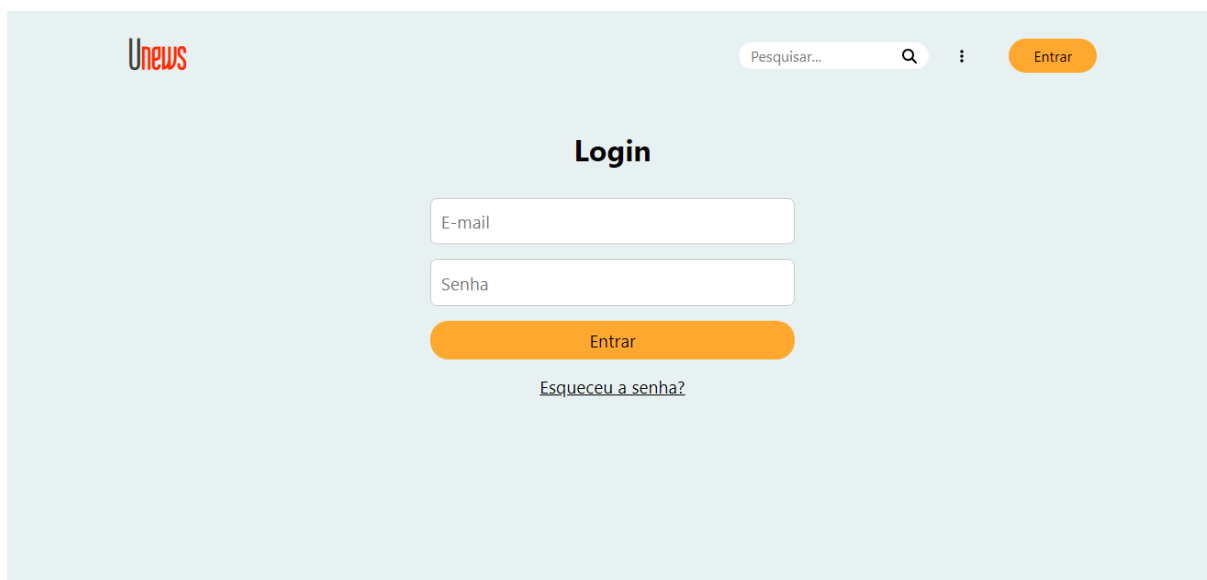


Figura 10 - Tela para edição de notícias

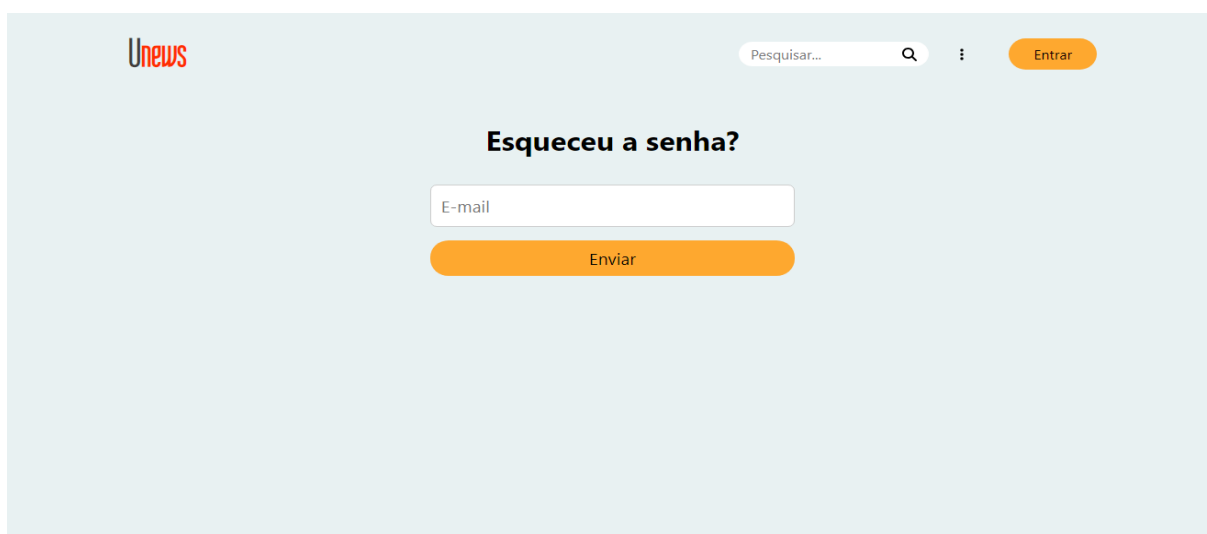
A figura 11 representa a tela de acesso, onde os usuários podem inserir seu email e senha para acessar o sistema.



The screenshot shows the login page of the Unews system. At the top left is the Unews logo. At the top right, there is a search bar with the placeholder text "Pesquisar..." and a magnifying glass icon, followed by a vertical ellipsis icon and an orange button labeled "Entrar". The main heading is "Login". Below it are two input fields: "E-mail" and "Senha". Underneath these fields is an orange button labeled "Entrar". At the bottom, there is a link that says "Esqueceu a senha?".

Figura 11 - Tela de acesso ao sistema

As figuras 12 e 13 correspondem às etapas de recuperação de senha. Na figura 12, é solicitado o *e-mail*. Se o e-mail existir no banco de dados, a figura 13 é apresentada e onde a pergunta de recuperação de senha é retornada. Finalmente, a resposta é validada e, se correta, permite o acesso ao sistema.



The screenshot shows the password recovery page of the Unews system. At the top left is the Unews logo. At the top right, there is a search bar with the placeholder text "Pesquisar..." and a magnifying glass icon, followed by a vertical ellipsis icon and an orange button labeled "Entrar". The main heading is "Esqueceu a senha?". Below it is an input field labeled "E-mail". Underneath this field is an orange button labeled "Enviar".

Figura 12 - Tela para recuperação de senha: inserção de email

Unews

Pesquisar... Q :

Entrar

Esqueceu a senha?

fatecam@gmail.com

Pergunta: Quando foi inaugurado o novo campus da fatec am?

Resposta

Enviar

Figura 13 - Tela para recuperação de senha: inserção de resposta

As figura 14 e 15, exibe a barra de navegação do sistema, onde a figura 14 corresponde ao usuário não autenticado, enquanto a figura 15 corresponde ao usuário administrador do sistema. Em ambas, a logo do sistema direciona o usuário para a página inicial, a barra de pesquisa permite filtrar notícias por título e a opção de login é disponível. Já na figura 15, a opção 'adicionar', também está presente, juntamente com as opções de acessar o perfil, nome do usuário autenticado e a opção de sair do sistema.



Figura 14 - Barra de navegação: usuário não autenticado

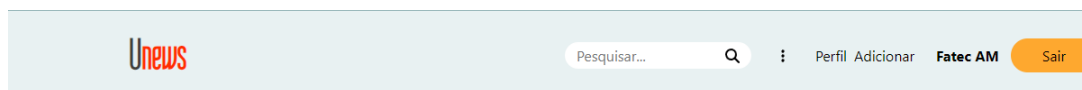


Figura 15 - Barra de navegação: usuário autenticado

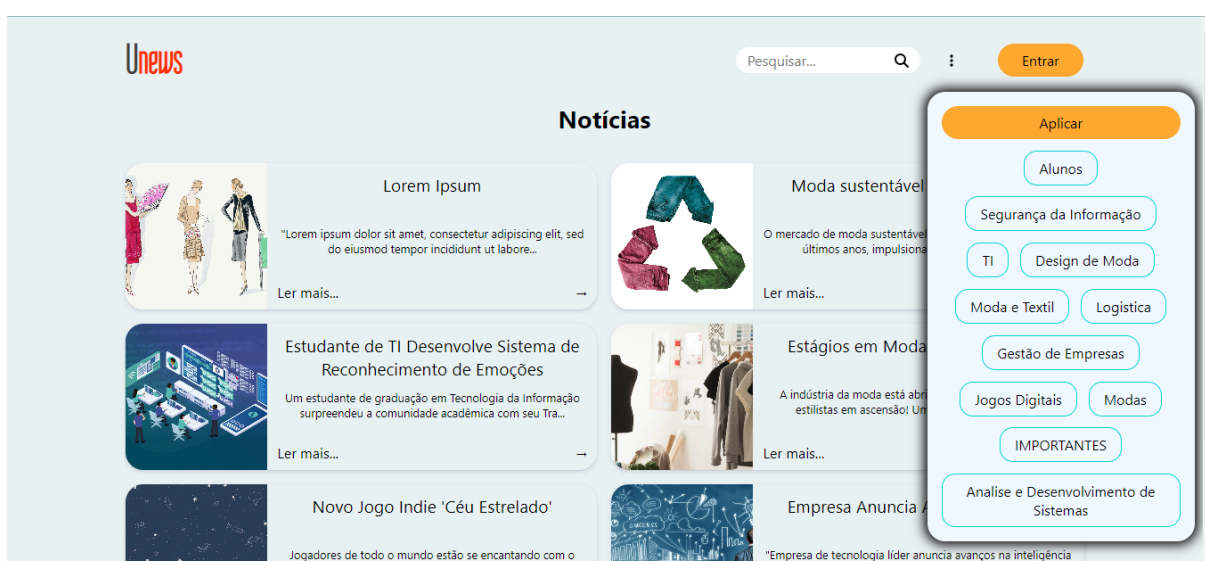
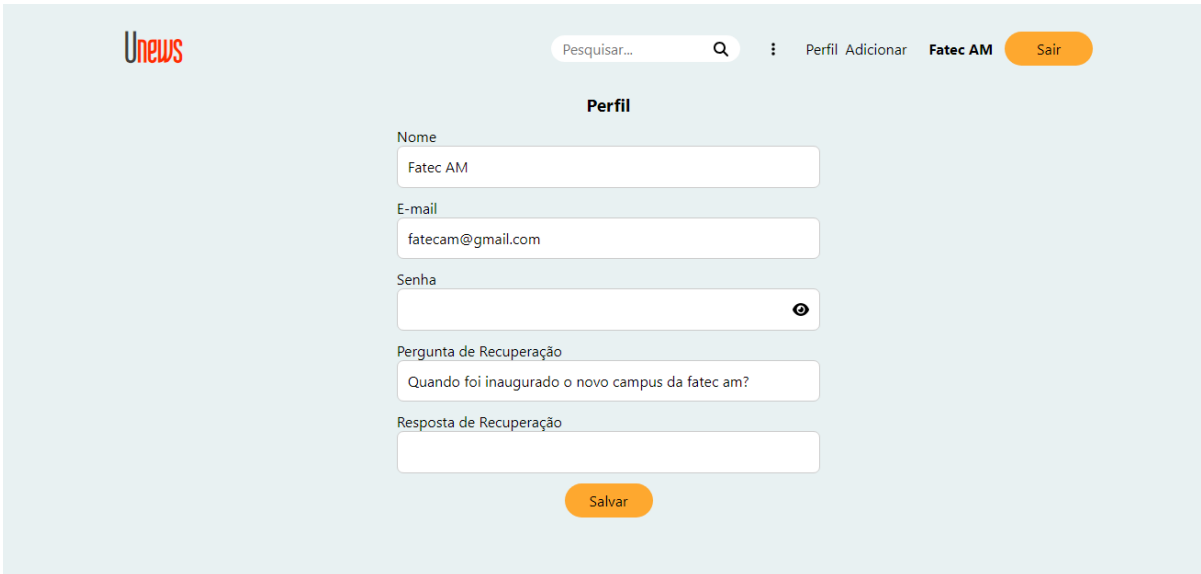


Figura 16 - Modal com opções de filtros

A figura 17, apresenta a tela para alteração de dados do usuário. Nesta tela, os administradores têm a capacidade de realizar alterações em seu perfil de forma conveniente

e eficiente, possibilitando atualizações de informações conforme necessário.



The screenshot displays the 'Perfil' (Profile) editing interface. At the top left is the 'Unews' logo. A search bar with the placeholder 'Pesquisar...' and a magnifying glass icon is positioned to the right. Further right are navigation links: 'Perfil Adicionar', 'Fatec AM', and a 'Sair' (Logout) button. The main content area is titled 'Perfil' and contains the following fields:

- Nome:** A text input field containing 'Fatec AM'.
- E-mail:** A text input field containing 'fatecam@gmail.com'.
- Senha:** A password input field with a toggle icon for visibility.
- Pergunta de Recuperação:** A text input field containing 'Quando foi inaugurado o novo campus da fatec am?'.
- Resposta de Recuperação:** An empty text input field.

At the bottom center of the form is an orange 'Salvar' (Save) button.

Figura 17 - Tela para edição de perfil

4.8 TESTES

No desenvolvimento do sistema, foi adotado o *Framework Robot*, baseado em *Python*, para a execução de testes automatizados. Simultaneamente, testes manuais foram conduzidos ao longo de todo o processo de desenvolvimento, a fim de assegurar a qualidade da plataforma.

Além disso, a escolha do *Framework Robot* proporcionou a capacidade de automação e flexibilidade, contribuindo para a criação de testes eficazes e eficientes, resultando na confiabilidade do sistema. Os testes manuais desempenharam um papel fundamental na identificação de problemas, especialmente relacionados à usabilidade e à experiência do usuário, que podem não ser facilmente detectados por meio de testes automatizados.

5 CONSIDERAÇÕES FINAIS

O desenvolvimento do Projeto do site Unews proporcionou uma rica experiência na criação de uma plataforma para a comunidade acadêmica criar e compartilhar notícias.

Os maiores desafios encontrados durante o desenvolvimento, estavam relacionados especialmente em torno da segurança e da usabilidade. Para garantir a segurança dos usuários, as senhas foram devidamente criptografadas, utilizando o mecanismo bcrypt, e um sistema de recuperação de senha foi implementado. Além disso, a validação do usuário para a troca de senha por meio de perguntas e respostas, que é criptografada antes de ser armazenada no Banco de Dados, tornou-se uma funcionalidade valiosa.

Uma funcionalidade adicional que se destacou foi a capacidade de filtrar as notícias por cursos, proporcionando uma organização mais específica e facilitando aos usuários a busca por informações relevantes ao seu campo de estudo.

Como possíveis desenvolvimentos futuros, considera-se a implementação de contas de alunos, que permitiriam a interação com as postagens, como avaliar notícias e adicionar comentários. Isso aumentaria o engajamento dos usuários e a interatividade na plataforma, tornando-a ainda mais valiosa para a comunidade acadêmica.

O website encontra-se disponível para download diretamente no repositório correspondente. Com o intuito de facilitar o uso do sistema, o usuário pode encontrar instruções detalhadas na página inicial deste repositório.

Adicionalmente, para aprimorar a acessibilidade ao código-fonte e às dependências do Projeto, foi implementada a opção de escanear um *Quick Response* (QR) Code, representado na figura 17, que redireciona os interessados para a página inicial do repositório do Projeto. Esse método simplifica significativamente o processo de acesso ao código-fonte do Projeto, tornando-o mais acessível a todos, proporcionando uma experiência mais fluida e descomplicada. Para concluir, gostaríamos e ficaríamos honrados que a Instituição viesse adotar esta Ferramenta, a fim de centralizar os diversos canais de informações.



Figura 18 – Acesso ao repositório do projeto

6 REFERÊNCIAS BIBLIOGRÁFICAS

ALFF, Francilvio Roberto. **Requisitos funcionais e não funcionais. entenda os 2 tipos.** Análise de Requisitos, 2018. Última atualização em 17 de outubro de 2023. Disponível em: <https://analisederequisitos.com.br/requisitos-funcionais-e-nao-funcionais/>. Acesso em 04 nov. 2023.

ANGULAR. **Introduction to services.** Disponível em: <https://angular.io/guide/architecture-services#:~:text=Service is a broad category,specific and do it well>. Acesso em: 04 nov. 2023.

AXIOS. **Introdução | Axios Docs.** Disponível em: <https://axios-http.com/ptbr/docs/intro>. Acesso em 25 out. 2023.

GIT. **Git --everything-is-local.** Disponível em: <https://git-scm.com/>. Acesso em: 20 out. 2023.

GITHUB. **Hello World - GitHub Docs.** Disponível em: <https://docs.github.com/en/get-started/quickstart/hello-world>. Acesso em: 20 out. 2023.

JS-COOKIE. **js-cookie: A simple, lightweight JavaScript API for handling browser cookies.** Disponível em: <https://github.com/js-cookie/js-cookie/tree/latest#readme>. Acesso em: 25 out. 2023.

LUCIDCHART. **O que é um diagrama de atividades?** Disponível em: <https://www.lucidchart.com/pages/pt/o-que-e-diagrama-de-atividades-uml>. Acesso em: 04 nov. 2023.

LUCIDCHART. **O que é um diagrama de classe em UML?** Disponível em: <https://www.lucidchart.com/pages/pt/o-que-e-diagrama-de-classe-uml>. Acesso em 04 nov. 2023.

LUCIDCHART. **O que é um diagrama entidade relacionamento?** Disponível em: <https://www.lucidchart.com/pages/pt/o-que-e-diagrama-entidade-relacionamento>. Acesso em 04 nov. 2023.

LUCIDCHART. **O que é diagrama de caso de uso?** Disponível em: <https://www.lucidchart.com/pages/pt/diagrama-de-caso-de-uso-uml>. Acesso em 04 nov. 2023.

MDN WEB DOCS. **Introdução Express/Node.** Disponível em: https://developer.mozilla.org/pt-BR/docs/Learn/Server-side/Express_Nodejs/Introduction. Acesso em: 04 nov. 2023.

MICROSOFT AZURE. **O que é Middleware.** Disponível em: <https://azure.microsoft.com/pt-br/resources/cloud-computing-dictionary/what-is-middleware/>. Acesso em: 04 nov. 2023.

MICROSOFT LEARN. **Visão geral do ASP.NET Core MVC.** Disponível em: <https://learn.microsoft.com/pt-br/aspnet/core/mvc/overview?view=aspnetcore-7.0#mvc-pattern>. Acesso em: 25 out. 2023.

MONGODB. **How To Use MERN Stack: A Complete Guide.** Disponível em: <https://www.mongodb.com/languages/mern-stack-tutorial>. Acesso em: 19 out. 2023.

MONGODB. **What Is The MERN Stack? Introduction & Examples.** Disponível em: <https://www.mongodb.com/mern-stack>. Acesso em: 19 out. 2023.

NODE JS **About Node.js** Disponível em: <https://nodejs.org/en/about>. Acesso em: 04 nov. 2023.

REACT **Describing the UI – React** Disponível em: <https://react.dev/learn/describing-the-ui>. Acesso em 04 nov. 2023.

REACT **React Reference Overview** Disponível em: <https://react.dev/reference/react>. Acesso em 04 nov. 2023.

REACT ROUTER **Main Concepts v6.18.0 | React Router** Disponível em: <https://reactrouter.com/en/main/start/concepts>. Acesso em 04 nov. 2023.

ROBOT FRAMEWORK. **User Guide.** Disponível em: <https://robotframework.org/robotframework/latest/RobotFrameworkUserGuide.html>. Acesso em: 20 out. 2023.