

UTFPR-UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ

Bacharelado em Engenharia de Software - 6º Período

DISCIPLINA: *Oficina de Integração 1 - ES46F-ES61*

PROFESSOR: *Eduardo Cotrin Teixeira*

Documento de Projeto de Software

Portal TEDI

Álison Christian Rebouças Vidal de Carvalho

João Marcos Ribeiro da Costa

João Pedro Correia Leite Moreira

Luccas Philot Gonçalves

Yuri Garcia Yoshida

Cornélio Procópio

2025

Sumário

1 INTRODUÇÃO	2
1.1 CONTEXTO.....	2
1.2 JUSTIFICATIVA	2
1.3 PROPOSTA.....	3
1.4 ORGANIZAÇÃO DO DOCUMENTO.....	3
2 DESCRIÇÃO GERAL DO SISTEMA	4
2.1 OBJETIVOS (GERAIS E ESPECÍFICOS)	4
2.2 LIMITES E RESTRIÇÕES.....	4
2.3 DESCRIÇÃO DOS USUÁRIOS DO SISTEMA	4
3 DESENVOLVIMENTO DO PROJETO	6
3.1 TECNOLOGIAS E FERRAMENTAS	6
<i>Ambiente de Desenvolvimento</i>	<i>6</i>
<i>Tecnologias Utilizadas</i>	<i>6</i>
<i>Ferramentas de Apoio.....</i>	<i>7</i>
3.2 METODOLOGIA DE DESENVOLVIMENTO	7
<i>Papéis no Scrum.....</i>	<i>7</i>
<i>Adaptação do Scrum no Projeto</i>	<i>8</i>
3.3 CRONOGRAMA PREVISTO.....	8
4 REQUISITOS DO SISTEMA	10
4.1 REQUISITOS FUNCIONAIS.....	10
<i>Descrição dos Requisitos Funcionais.....</i>	<i>10</i>
4.2 REQUISITOS NÃO-FUNCIONAIS.....	11
4.3 DIAGRAMAS DE CASOS DE USO.....	12
4.4 PROTÓTIPOS DE TELAS	13
<i>Telas de Administrador (Figuras 03 e 04):.....</i>	<i>13</i>
<i>Telas de Usuário (Figura 04):.....</i>	<i>15</i>
5 ANÁLISE DO SISTEMA	16
5.1 MODELO DO BANCO DE DADOS.....	16
<i>Dicionário de Dados.....</i>	<i>17</i>
5.2 DIAGRAMA DE CLASSES	19
5.3 DIAGRAMA DE ATIVIDADES	20
6 IMPLEMENTAÇÃO	21
6.1 DESCRIÇÃO DO CÓDIGO	22
6.2 IMPLANTAÇÃO.....	22
6.3 TELAS PRINCIPAIS	22
7 CONSIDERAÇÕES FINAIS	23
8 BIBLIOGRAFIA	24

1 Introdução

1.1 Contexto

O TEDI (Tecnologia e Educação Digital para a Terceira Idade) é um projeto de extensão da UTFPR – Campus Cornélio Procopio, criado em 2024, com o objetivo de promover a inclusão digital da população idosa por meio de oficinas e cursos gratuitos. A iniciativa nasceu da necessidade urgente de combater a exclusão tecnológica que afeta a terceira idade, dificultando seu acesso a recursos digitais, como celulares, computadores e serviços online. Ao proporcionar familiaridade com essas tecnologias, o projeto busca resgatar a autonomia, a autoestima e a integração social dos idosos.

O projeto é organizado por professores da UTFPR e executado com o apoio de voluntários — alunos da faculdade — que recebem orientação para atuar com metodologias acessíveis ao público idoso. Esses voluntários ministram oficinas práticas e lúdicas com paciência, respeito e foco na troca intergeracional de saberes, promovendo a construção de vínculos entre jovens e idosos.

As atividades são estruturadas em dois módulos independentes:

- Informática Básica, que inclui noções fundamentais sobre o uso de computadores, periféricos e navegação;
- Smartphones e Aplicativos, voltado ao uso de celulares, redes sociais, ferramentas de comunicação e serviços digitais essenciais.

Além de promover a inclusão digital, o TEDI contribui para a autoestima, o bem-estar e a cidadania da pessoa idosa, fortalecendo sua presença ativa na sociedade contemporânea. Atualmente, o projeto também desenvolve materiais didáticos próprios e estuda a expansão das ações, com o objetivo de oferecer capacitações mais completas e duradouras.

1.2 Justificativa

Embora o TEDI já desenvolva ações importantes voltadas à inclusão digital de idosos, ainda carece de uma plataforma online que concentre suas informações, divulgue suas atividades e facilite a participação de novos colaboradores.

A criação de um site institucional visa preencher essa lacuna, oferecendo um meio acessível para que o público conheça o projeto, acompanhe notícias e, principalmente, que voluntários possam se inscrever com facilidade — o que é essencial para a continuidade do trabalho, já que o funcionamento do TEDI depende diretamente dessa participação.

Atualmente, a divulgação e o recrutamento são feitos de forma descentralizada, o que dificulta o alcance e a organização das ações. Um site bem estruturado permitirá maior visibilidade, comunicação eficiente e apoio à expansão do projeto.

1.3 Proposta

Com o intuito de apoiar o projeto TEDI, será desenvolvido um site que visa facilitar o acesso às informações do projeto de forma simples e acessível para todos os públicos, especialmente os idosos.

A aplicação permitirá a visualização de notícias e informativos cadastrados pelos administradores do projeto, garantindo que os usuários estejam sempre atualizados sobre oficinas, cursos e demais atividades. Além disso, contará com uma página de formulário destinada ao cadastro de monitores voluntários, promovendo a ampliação da equipe de apoio. A interface será responsiva, acessível e projetada para uma navegação intuitiva, respeitando as diretrizes de acessibilidade.

1.4 Organização do Documento

Este documento apresenta, de forma estruturada, todas as etapas do desenvolvimento da aplicação web de apoio ao projeto TEDI (Tecnologia e Educação Digital para a Terceira Idade).

- O **Capítulo 1** introduz o contexto, a motivação e os objetivos do projeto, além de justificar a necessidade da solução proposta.
- O **Capítulo 2** descreve o sistema a ser desenvolvido, incluindo os objetivos gerais e específicos, limitações e o perfil dos usuários.
- O **Capítulo 3** aborda o desenvolvimento da aplicação, detalhando as tecnologias adotadas, a metodologia de trabalho e o cronograma previsto.
- O **Capítulo 4** apresenta os requisitos do sistema, funcionais e não funcionais, além de diagramas de caso de uso e protótipos das interfaces.
- O **Capítulo 5** discute a análise do sistema, considerando sua viabilidade e aplicabilidade.
- O **Capítulo 6** trata da implementação, com a descrição do código, processo de implantação e telas principais.
- O **Capítulo 7** reúne as considerações finais sobre o projeto.
- Por fim, o **Capítulo 8** apresenta as referências utilizadas ao longo do desenvolvimento.

2 Descrição Geral do Sistema

2.1 Objetivos (Gerais e Específicos)

Desenvolver uma aplicação web acessível, intuitiva e centrada no usuário idoso, com o intuito de divulgar e apoiar a administração do projeto TEDI, promovendo a inclusão digital e social.

Objetivos Específicos:

- Permitir a visualização clara e simplificada de informações sobre oficinas, cursos e demais atividades do TEDI;
- Disponibilizar atualizações, comunicados e notícias relacionadas ao projeto;
- Oferecer um formulário acessível para o cadastro de monitores voluntários;
- Garantir que a interface da aplicação atenda às diretrizes de acessibilidade digital, considerando as limitações do público idoso;
- Facilitar o gerenciamento de conteúdo por parte da equipe organizadora do projeto.

2.2 Limites e Restrições

A aplicação será desenvolvida exclusivamente para ambiente web, priorizando compatibilidade com os principais navegadores modernos. Inicialmente, não contará com funcionalidades como inscrição automática em cursos, interação entre usuários ou integração com redes sociais. A aplicação exigirá conexão com a internet e seguirá as diretrizes da WCAG 2.2, mas sua experiência pode variar conforme o dispositivo utilizado e o nível de familiaridade tecnológica do usuário.

2.3 Descrição dos Usuários do Sistema

O sistema será utilizado por diferentes perfis, cada um com objetivos e necessidades específicas:

- Alunos: interessados em acompanhar notícias e informações gerais sobre o projeto;
- Idosos: público-alvo dos ofícios e cursos, acessam conteúdos de maneira simplificada e acessível;
- Monitores: voluntários que se cadastram para auxiliar nas atividades do TEDI;
- Parceiros: pessoas ou instituições que buscam informações para possíveis colaborações;

- Equipe do Projeto: responsáveis pela administração do conteúdo, cadastros e manutenção das informações exibidas.

3 Desenvolvimento do Projeto

3.1 Tecnologias e ferramentas

Ambiente de Desenvolvimento

A aplicação será desenvolvida no **Visual Studio Code (VS Code)**, um ambiente de desenvolvimento integrado (IDE) gratuito, leve e muito utilizado por programadores. Ele oferece recursos que facilitam a escrita de código, como destaque de sintaxe, sugestões automáticas, controle de versões com Git, depuração integrada e grande variedade de extensões. Além disso, é compatível com várias linguagens, como TypeScript e JavaScript, que serão utilizadas no projeto.

Tecnologias Utilizadas

Frontend: Angular com TypeScript

A interface do usuário será construída com o **Angular**, um **framework** desenvolvido pelo Google. Um **framework** é um conjunto de ferramentas e regras prontas que facilitam o desenvolvimento de aplicações. Ele fornece estrutura, organização e componentes reutilizáveis, o que torna o processo mais ágil e padronizado.

O Angular é especialmente útil para criar **aplicações web dinâmicas** e de fácil manutenção. Ele funciona em conjunto com o **TypeScript**, uma linguagem que se baseia no JavaScript, mas adiciona recursos como tipagem estática. Isso significa que muitos erros podem ser detectados antes mesmo da execução do código, o que aumenta a segurança e a clareza no desenvolvimento.

Backend: JavaScript com Express

No lado do servidor, será usado **JavaScript** com o **framework Express**. O **Express** é um conjunto de bibliotecas que simplifica a criação de servidores e rotas em aplicações web. Ele permite definir facilmente o que o sistema deve fazer ao receber requisições (como salvar dados ou exibir uma página).

O Express é ideal para desenvolver **APIs RESTful**, que são conjuntos de regras para permitir que diferentes sistemas se comuniquem de forma padronizada. Essas APIs funcionam como "pontes" entre o frontend e o backend, usando operações simples (como buscar, adicionar ou excluir dados) através da internet.

Banco de Dados: PostgreSQL

O sistema utilizará o **PostgreSQL** como banco de dados, que é uma solução gratuita, robusta e amplamente reconhecida por sua confiabilidade. Ele permite armazenar, consultar e organizar grandes volumes de dados com segurança.

O PostgreSQL também oferece suporte a **transações ACID**, um conjunto de regras que garante que as operações realizadas no banco sejam **atômicas, consistentes, isoladas e duráveis**. Em outras palavras,

isso significa que os dados não correm o risco de ficarem corrompidos ou inconsistentes, mesmo se ocorrer uma falha no meio de uma operação, como durante um cadastro ou exclusão.

Ferramentas de Apoio

Figma

As telas do sistema foram prototipadas no **Figma**, uma ferramenta online muito utilizada para o design de interfaces. Ela permite que todos os membros da equipe visualizem, comentem e colaborem em tempo real, mesmo à distância. Com ela, foi possível construir os protótipos de forma clara, garantindo que o visual do site fosse acessível, intuitivo e adequado ao público idoso.

Trello (Kanban)

Para organizar as tarefas e o progresso do projeto, foi adotado o **Trello**, uma ferramenta de gerenciamento de atividades baseada no método **Kanban**. O Kanban utiliza colunas (como “A Fazer”, “Em Andamento” e “Concluído”) e cartões visuais que representam cada tarefa.

Essa organização ajuda a equipe a saber exatamente o que está sendo feito, por quem, e o que ainda precisa ser entregue, garantindo transparência e controle durante toda a execução do projeto.

Astah UML

Os diagramas utilizados para representar o sistema (como os diagramas de classes, atividades e casos de uso) foram criados com o **Astah UML**, um software especializado em modelagem visual. Ele facilita a representação gráfica de como o sistema deve se comportar, como os dados se relacionam e quais são os fluxos de ações, ajudando na comunicação entre os membros da equipe e na fase de codificação.

Canais de Comunicação: WhatsApp e Discord

A comunicação entre os integrantes é feita principalmente através do **WhatsApp**, usado para atualizações rápidas, e do **Discord**, utilizado para reuniões por chamada de voz e compartilhamento de tela durante sessões de desenvolvimento. Esses canais facilitam a interação fora do ambiente presencial, mantendo todos atualizados e conectados.

3.2 Metodologia de desenvolvimento

O projeto adotará a metodologia **Scrum**, um modelo ágil amplamente utilizado no desenvolvimento de software. O Scrum organiza o trabalho em ciclos curtos e fixos, chamados de Sprints, que geralmente duram de uma a quatro semanas. Em cada sprint, a equipe planeja, desenvolve e entrega uma nova versão parcial do sistema, sempre com funcionalidades reais que podem ser avaliadas.

Esse modelo torna o processo mais eficiente, adaptável e colaborativo, pois permite que o sistema seja construído de forma incremental. Isso significa que o produto evolui por meio de entregas frequentes, cada uma representando uma parte já funcional e testável. Com isso, é possível receber feedback constante,

corrigir problemas com agilidade e garantir que o projeto atenda de forma mais precisa às necessidades do cliente ou usuário final.

Papéis no Scrum

O Scrum define papéis específicos dentro da equipe:

- **Product Owner (PO):** é a pessoa responsável por representar os interesses do cliente, priorizar as tarefas e validar o que foi entregue.
No projeto TEDI, o PO é **Álison Christian Rebouças Vidal de Carvalho**.
- **Scrum Master:** atua como facilitador do time, garantindo que o processo do Scrum seja seguido e que não haja obstáculos impedindo o trabalho da equipe.
No grupo, esse papel é desempenhado por **João Marcos Ribeiro da Costa**.
- **Desenvolvedores:** são os membros responsáveis pela implementação prática das funcionalidades.
São eles: **João Pedro Correia Leite Moreira, Luccas Philot Gonçalves e Yuri Garcia Yoshida**.

Adaptação do Scrum no Projeto

O Scrum tradicional prevê **reuniões diárias** chamadas de **Daily Meetings** (ou "dailies"), mas para adaptar o método à realidade acadêmica do grupo, a equipe optou por se reunir **duas vezes por semana**:

- **Quartas-feiras (presencial):**
Essa reunião acontece durante a aula e tem a função de **Sprint Review** (revisar a entrega da sprint anterior) e **Retrospective** (avaliar o que funcionou bem ou precisa melhorar).
- **Domingos (online via Discord):**
É usada como **Sprint Planning**, onde são definidas as metas da próxima semana e distribuídas as tarefas no quadro Kanban do Trello.

Essa adaptação mantém o foco na entrega contínua e organizada, mesmo com o tempo reduzido de encontro entre os membros. Cada tarefa é associada a uma sprint semanal, o que garante avanço constante no desenvolvimento.

3.3 Cronograma previsto

Tarefa	Entrega	Responsáveis
Estrutura inicial - Front	21/04	Yuri Yoshida
Estruturação do banco de dados	21/04	Yuri Yoshida
Login - Front	21/04	Yuri Yoshida e Luccas Philot
Login - Back	21/04	Luccas Philot
Página de notícias (admin) - Front	28/04	Yuri Yoshida e Luccas Philot
Página de notícias (admin) - Back	28/04	João Pedro Moreira e Luccas Philot
Página Inicial - Front	05/05	Yuri Yoshida e Luccas Philot

Página Inicial - Back	05/05	João Pedro Moreira e Lucas Philot
Página de monitores (admin) - Front	12/05	Yuri Yoshida e Lucas Philot
Página d monitores (admin) - Back	12/05	João Pedro Moreira e Lucas Philot
Item 5 - Análise do Sistema	18/05	Álison Carvalho e João Marcos
Roteiro do Vídeo 2	18/05	Álison Carvalho e João Marcos
Página de equipe (admin) - Front	22/05	Yuri Yoshida e Lucas Philot
Página de equipe (admin) - Back	22/05	João Pedro Moreira e Lucas Philot
Página de notícias - Front	26/05	Yuri Yoshida e Lucas Philot
Páginas de notícias - Back	26/05	João Pedro Moreira e Lucas Philot
Página específica de notícia - Front	02/06	Yuri Yoshida e Lucas Philot
Página específica de notícia - Back	02/06	João Pedro Moreira e Lucas Philot
Página de monitores - Front	09/06	Yuri Yoshida e Lucas Philot
Página de monitores - Back	09/06	João Pedro Moreira e Lucas Philot
Página de equipe - Front	16/06	Yuri Yoshida e Lucas Philot
Página de equipe - Back	16/06	João Pedro Moreira e Lucas Philot
Itens 6 a 8 do documento de projeto	25/06	Álison Carvalho e João Marcos
Roteiro do Vídeo Final	25/06	Álison Carvalho e João Marcos

4 Requisitos do Sistema

4.1 Requisitos Funcionais

Código	Descrição	Prioridade
RF01	O usuário deve ser capaz de cadastrar monitores com RA e CPF únicos, impedindo duplicações.	Essencial
RF02	O administrador deve ser capaz de realizar o login e logout na plataforma.	Essencial
RF03	O administrador deve ser capaz de gerenciar monitores (visualizar e excluir).	Essencial
RF04	O administrador deve ser capaz de copiar os e-mails de todos os monitores cadastrados com apenas um clique.	Importante
RF05	O administrador deve ser capaz de gerenciar notícias (cadastrar, editar e excluir).	Essencial
RF06	O sistema deve exibir uma página pública com o feed de notícias.	Essencial
RF07	Na página inicial, o sistema deve exibir miniaturas das últimas notícias.	Desejável
RF08	O usuário deve ser capaz de aplicar filtros na página de notícias por data, palavra-chave e categoria.	Importante
RF09	O sistema deve exibir uma página com informações sobre a equipe (foto, nome e função).	Importante
RF10	O administrador deve ser capaz de gerenciar membros da equipe (cadastrar, editar e excluir).	Essencial
RF11	O sistema deve disponibilizar um formulário para entrar em contato com a equipe.	Importante
RF12	O sistema deve exibir uma barra de navegação fixa visível em todas as páginas.	Essencial
RF13	O administrador deve ser capaz de filtrar informações na página de notícias, de monitores e de equipe.	Importante
RF14	O administrador pode adicionar os usuários cadastrados (monitores e membros da equipe) a um grupo.	Desejável

Tabela 01: Requisitos Funcionais do Sistema

Descrição dos Requisitos Funcionais

- **RF01 – Cadastro de Monitores:**
O sistema deve permitir que alunos da UTFPR se cadastrem como monitores preenchendo um formulário com nome, RA, CPF e e-mail. O sistema deve validar RA e CPF para evitar duplicações.
- **RF02 – Login e Logout:**
O sistema deve permitir que o administrador acesse as funcionalidades restritas com login e senha. O botão de logout deve estar visível nas páginas protegidas.
- **RF03 – Gerenciamento de Monitores:**
O sistema deve exibir uma lista dos monitores cadastrados, permitindo que o administrador visualize

ou exclua os dados.

- **RF04 – Cópia de E-mails:**
O sistema deve oferecer um botão para copiar os e-mails dos monitores já cadastrados, facilitando a comunicação por fora da plataforma.
- **RF05 – Gestão de Notícias:**
O sistema deve permitir ao administrador publicar, editar e excluir notícias com título, corpo e imagens, com exibição automática no site.
- **RF06 – Feed de Notícias:**
O sistema deve exibir uma página com o feed público de notícias, ordenadas por data, acessível a todos os visitantes.
- **RF07 – Carrossel de Notícias na Página Inicial:**
O sistema deve exibir na página inicial um carrossel automático com miniaturas, título e imagem das últimas notícias.
- **RF08 – Filtro de Notícias:**
O sistema deve permitir filtrar notícias por data, palavras-chave e categorias na página de notícias.
- **RF09 – Página da Equipe:**
O sistema deve exibir uma página com nome, foto e função dos membros da equipe do projeto.
- **RF10 – Gestão da Equipe:**
O sistema deve permitir que o administrador cadastre, edite e exclua informações sobre os membros da equipe.
- **RF11 – Formulário de Contato:**
O sistema deve disponibilizar uma área com formulário de contato, enviando as mensagens para a equipe do projeto.
- **RF12 – Barra de Navegação:**
O sistema deve conter uma barra de navegação fixa no topo do site com acesso a todas as páginas.
- **RF13 – Filtro de Informações Administrativas:**
O sistema deve permitir que o administrador filtre informações nas páginas de notícias, monitores e equipe para facilitar a busca e o gerenciamento.
- **RF14 – Criação de Grupos:**
O sistema deve permitir que o administrador crie grupos de usuários, de forma que ele possa copiar os e-mails de um determinado grupo.

4.2 Requisitos Não-funcionais

Código	Descrição	Categoria	Prioridade
RNF01	O sistema deve ser responsivo, funcionando bem em dispositivos móveis e desktops.	Usabilidade	Essencial
RNF02	O sistema deve ser hospedado em domínio e servidor da própria faculdade.	Ambiente	Essencial
RNF03	O sistema deve ser compatível com navegadores modernos (Chrome, Firefox, Edge).	Compatibilidade	Importante
RNF04	O sistema deve armazenar a senha do único administrador de forma criptografada.	Segurança	Essencial
RNF05	O sistema deve implementar validações para evitar	Segurança	Importante

	entradas inválidas e ataques como XSS e SQL Injection.		
RNF06	O sistema deve seguir princípios de acessibilidade (como contraste, navegação por teclado e textos alternativos).	Usabilidade	Importante
RNF07	O sistema deve utilizar ferramentas e bibliotecas gratuitas, compatíveis com o ambiente da faculdade.	Recursos	Essencial
RNF08	O código do sistema deve ser organizado em camadas separadas: apresentação, lógica e dados.	Padronização	Desejável

Tabela 02: Requisitos Não Funcionais do Sistema

4.3 Diagramas de Casos de Uso

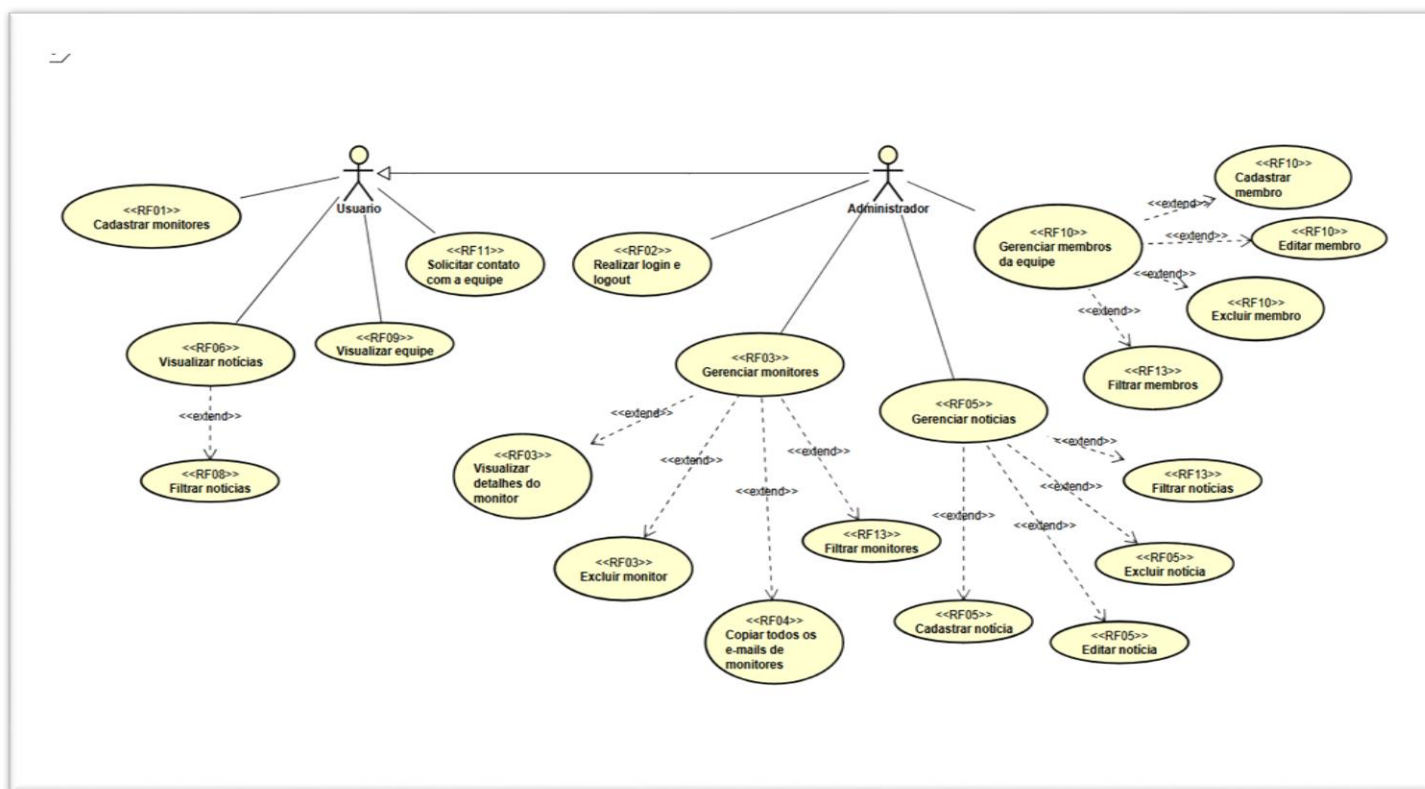


Figura 01: Diagrama de Caso de Uso Geral

4.4 Protótipos de Telas

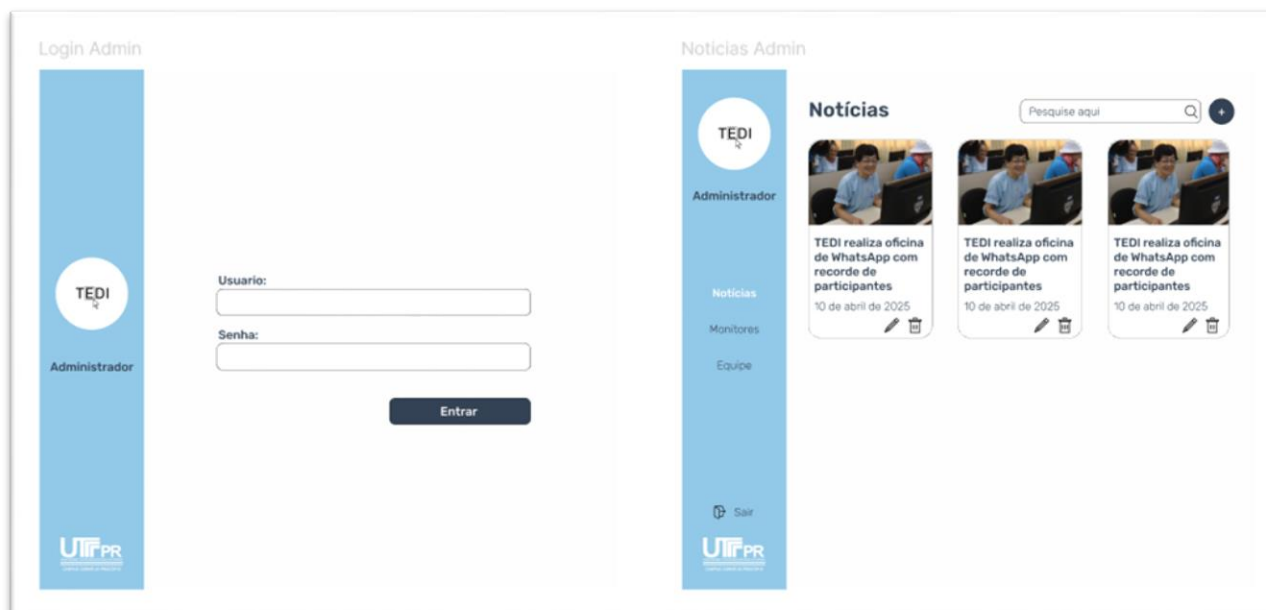


Figura 02: Protótipo de Tela de Administrador

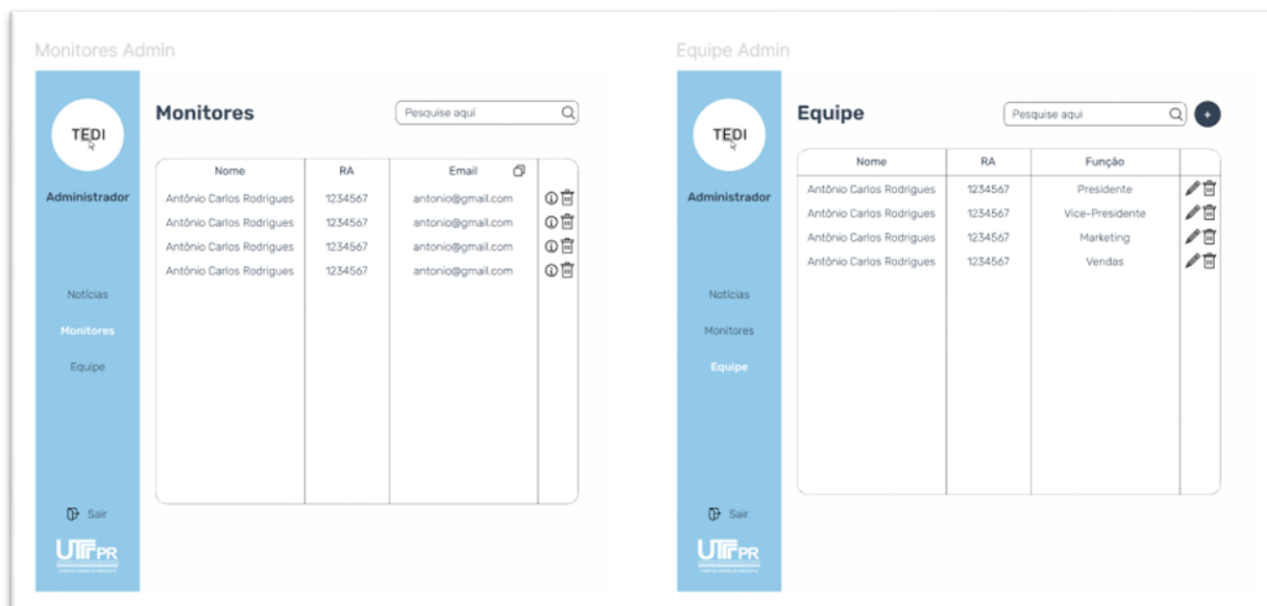


Figura 03: Protótipo de Telas de Administrador

Telas de Administrador (Figuras 03 e 04):

- **Login Admin (RF02):**
Página de autenticação exclusiva para o administrador do sistema. O login é realizado por meio de senha previamente definida, garantindo acesso às funcionalidades restritas. Um botão de logout

estará disponível após o login para garantir a segurança da sessão.

- **Notícias Admin (RF05):**
Página onde o administrador pode gerenciar as notícias do sistema. É possível cadastrar novas notícias, editar conteúdos existentes, pesquisar por palavras-chave e excluir publicações. Cada notícia inclui título, corpo e imagem.
- **Monitores Admin (RF03):**
Área onde o administrador visualiza a lista de monitores cadastrados, pode remover registros e copiar todos os e-mails. A exibição inclui informações como nome, RA, CPF e e-mail.
- **Equipe Admin (RF10):**
Tela de administração dos membros da equipe do projeto. O administrador pode adicionar novos integrantes, editar dados existentes ou excluir membros, com base em nome, função e imagem.

O fluxo de navegação de usuário administrador parte do login, caso bem-sucedido, para qualquer uma das outras telas – existirá um menu de navegação para isso.

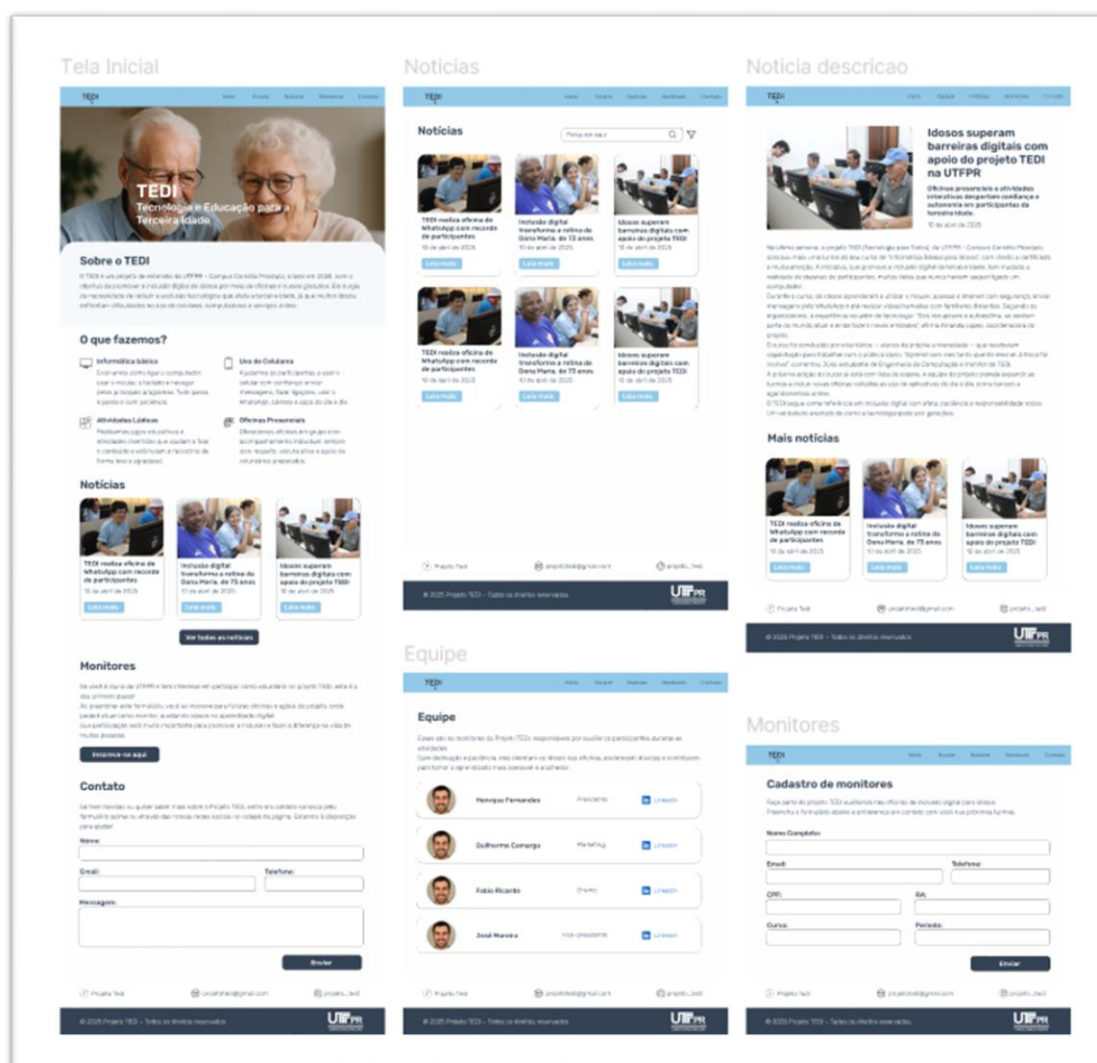


Figura 04: Protótipo de Telas de Usuário

Telas de Usuário (Figura 04):

- **Tela Inicial (RFs 07, 11 e 12):**
Página principal do site, onde os usuários têm acesso à barra de navegação fixa no topo (presente em todas as páginas). A tela inicial exibe um carrossel com miniaturas das últimas notícias, permite navegar para outras seções do site e conta com um formulário de contato para envio de mensagens à equipe do projeto.
- **Notícias (RFs 06 e 08):**
Página dedicada à exibição de todas as notícias publicadas no projeto. Os usuários podem visualizar a lista completa de notícias e aplicar filtros por data, palavras-chave e categoria para refinar a busca.
- **Notícia Descrição (RFs 06 e 08):**
Tela individual de uma notícia específica, exibindo todas as informações relacionadas a ela, como título, data, conteúdo completo e imagem destacada. Os filtros aplicados anteriormente podem continuar influenciando a navegação.
- **Equipe (RFs 09 e 08):**
Página que apresenta os membros ativos do projeto com suas respectivas fotos, nomes e funções. Pode conter filtros para facilitar a visualização dos integrantes conforme a categoria ou função no projeto.
- **Monitores (RF01):**
Tela destinada ao cadastro de monitores da UTFPR interessados em participar das oficinas. Um formulário coleta informações como nome, RA, CPF e e-mail, com validações para evitar duplicidade de RA e CPF no sistema.

A partir da tela inicial será possível acessar quaisquer umas das telas subsequentes e vice-versa; o menu de navegação estará disponível em todas elas.

5 Análise do Sistema

Esta seção apresenta os principais artefatos utilizados na modelagem e análise do sistema, com o objetivo de apoiar a implementação da aplicação web do projeto TEDI. Os modelos aqui descritos foram elaborados com base nos requisitos levantados e refletem a estrutura lógica e o comportamento esperado da aplicação. Inicialmente, é apresentado o **modelo conceitual do banco de dados**, que descreve as entidades, atributos e relacionamentos necessários para o funcionamento do sistema. Em seguida, é detalhado o **dicionário de dados**, contendo as especificações técnicas de cada campo e suas restrições. Também são incluídos o **diagrama de classes**, que ilustra a estrutura orientada a objetos do sistema, e o **diagrama de atividades**, que representa os fluxos de execução de funcionalidades relevantes, como o cadastro de monitores e a gestão de notícias.

Esses artefatos foram elaborados com o intuito de fornecer uma base sólida para o desenvolvimento, facilitar a manutenção futura e garantir que todos os envolvidos tenham uma visão clara da estrutura e do comportamento do sistema.

5.1 Modelo do Banco de Dados

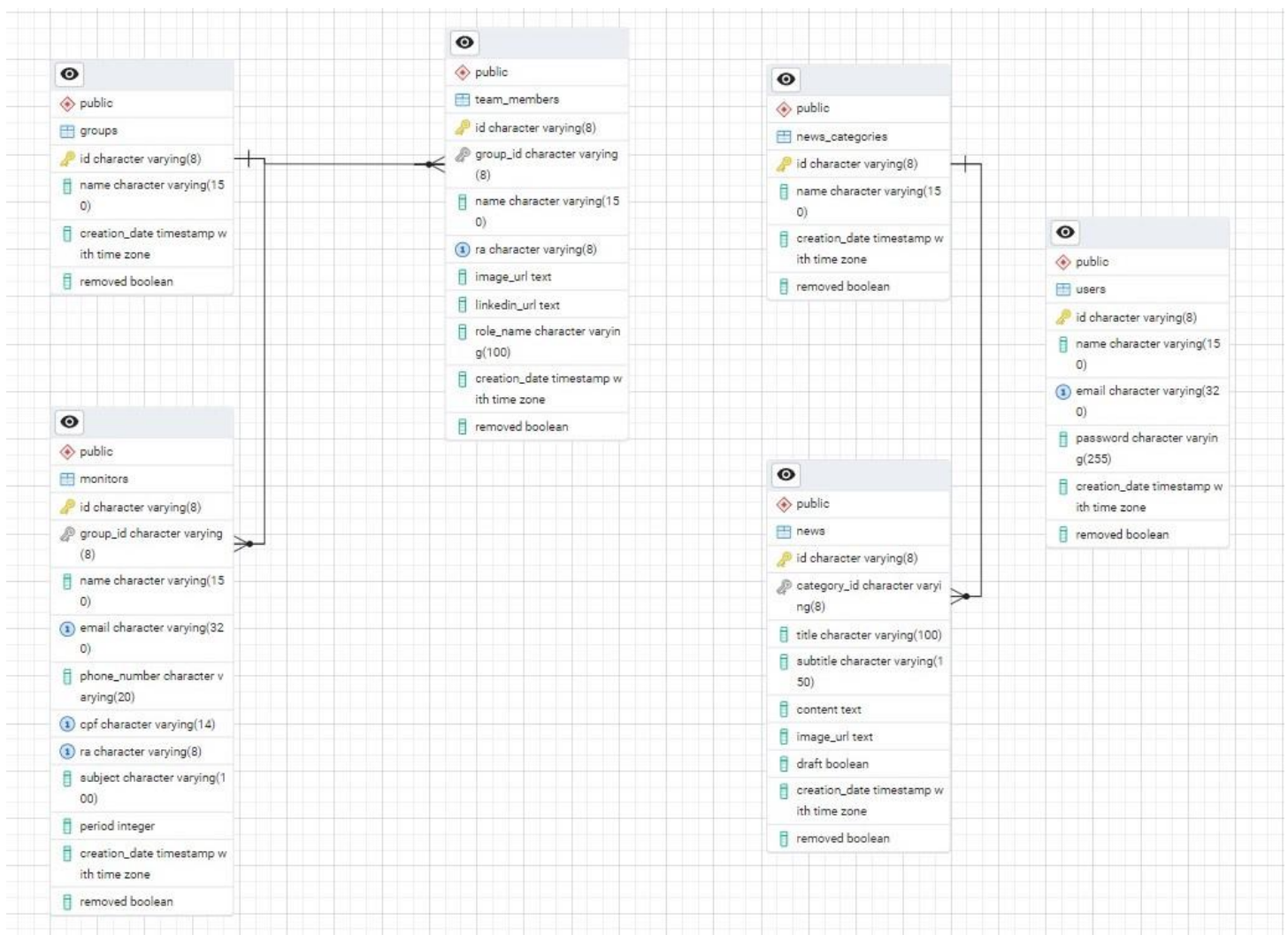


Figura 05: Modelo Conceitual (PostgreSQL)

Dicionário de Dados

Tabela: Users

Descrição: Armazena as informações dos usuários do sistema.

Nome	Descrição	Tipo de dado	Tamanho	Restrições
id	Identificador único	VARCHAR	8	PK, NOT NULL
name	Nome do usuário	VARCHAR	150	NOT NULL
email	Email do usuário	VARCHAR	320	UNIQUE, NOT NULL
password	Senha do usuário	VARCHAR	255	NOT NULL
creation_date	Data de criação	TIMESTAMP WITH TZ	-	NOT NULL, DEFAULT NOW()
removed	Indicador de remoção	BOOLEAN	-	NOT NULL, DEFAULT FALSE

Tabela: News Categories

Descrição: Armazena as categorias de notícias.

Nome	Descrição	Tipo de dado	Tamanho	Restrições
id	Identificador único	VARCHAR	8	PK, NOT NULL
name	Nome da categoria	VARCHAR	150	NOT NULL
creation_date	Data de criação	TIMESTAMP WITH TZ	-	NOT NULL, DEFAULT NOW()
removed	Indicador de remoção	BOOLEAN	-	NOT NULL, DEFAULT FALSE

Tabela: News

Descrição: Armazena notícias publicadas no sistema.

Nome	Descrição	Tipo de dado	Tamanho	Restrições
id	Identificador único	VARCHAR	8	PK, NOT NULL
category_id	Categoria da notícia	VARCHAR	8	FK, NOT NULL
title	Título da notícia	VARCHAR	100	NOT NULL
subtitle	Subtítulo da notícia	VARCHAR	150	-
content	Conteúdo da notícia	TEXT	-	NOT NULL
image_url	URL da imagem	TEXT	-	NOT NULL
draft	Indica se é rascunho	BOOLEAN	-	NOT NULL, DEFAULT TRUE
creation_date	Data de criação	TIMESTAMP WITH TZ	-	NOT NULL, DEFAULT NOW()
removed	Indicador de remoção	BOOLEAN	-	NOT NULL, DEFAULT FALSE

Tabela: News

Descrição: Armazena notícias publicadas no sistema.

Nome	Descrição	Tipo de dado	Tamanho	Restrições
id	Identificador único	VARCHAR	8	PK, NOT NULL
category_id	Categoria da notícia	VARCHAR	8	FK, NOT NULL
title	Título da notícia	VARCHAR	100	NOT NULL
subtitle	Subtítulo da notícia	VARCHAR	150	-
content	Conteúdo da notícia	TEXT	-	NOT NULL
image_url	URL da imagem	TEXT	-	NOT NULL
draft	Indica se é rascunho	BOOLEAN	-	NOT NULL, DEFAULT TRUE
creation_date	Data de criação	TIMESTAMP WITH TZ	-	NOT NULL, DEFAULT NOW()
removed	Indicador de remoção	BOOLEAN	-	NOT NULL, DEFAULT FALSE

Tabela: Monitors

Descrição: Armazena informações sobre monitores.

Nome	Descrição	Tipo de dado	Tamanho	Restrições
id	Identificador único	VARCHAR	8	PK, NOT NULL
group_id	Grupo associado	VARCHAR	8	FK
name	Nome do monitor	VARCHAR	150	NOT NULL
email	Email do monitor	VARCHAR	320	UNIQUE, NOT NULL
phone_number	Telefone do monitor	VARCHAR	20	NOT NULL
cpf	CPF do monitor	VARCHAR	14	UNIQUE, NOT NULL
ra	Registro acadêmico	VARCHAR	8	UNIQUE, NOT NULL
subject	Disciplina monitorada	VARCHAR	100	NOT NULL
period	Período do curso	INTEGER	-	NOT NULL, CHECK (1–10)
creation_date	Data de criação	TIMESTAMP WITH TZ	-	NOT NULL, DEFAULT NOW()
removed	Indicador de remoção	BOOLEAN	-	NOT NULL, DEFAULT FALSE

Tabela: Team Members

Descrição: Armazena informações dos membros da equipe.

Nome	Descrição	Tipo de dado	Tamanho	Restrições
id	Identificador único	VARCHAR	8	PK, NOT NULL
group_id	Grupo associado	VARCHAR	8	FK
name	Nome do membro	VARCHAR	150	NOT NULL
ra	Registro acadêmico	VARCHAR	8	UNIQUE, NOT NULL

Nome	Descrição	Tipo de dado	Tamanho	Restrições
image_url	URL da imagem	TEXT	-	NOT NULL
linkedin_url	URL do LinkedIn	TEXT	-	-
role_name	Nome do cargo	VARCHAR	100	NOT NULL
creation_date	Data de criação	TIMESTAMP WITH TZ	-	NOT NULL, DEFAULT NOW()
removed	Indicador de remoção	BOOLEAN	-	NOT NULL, DEFAULT FALSE

5.2 Diagrama de Classes

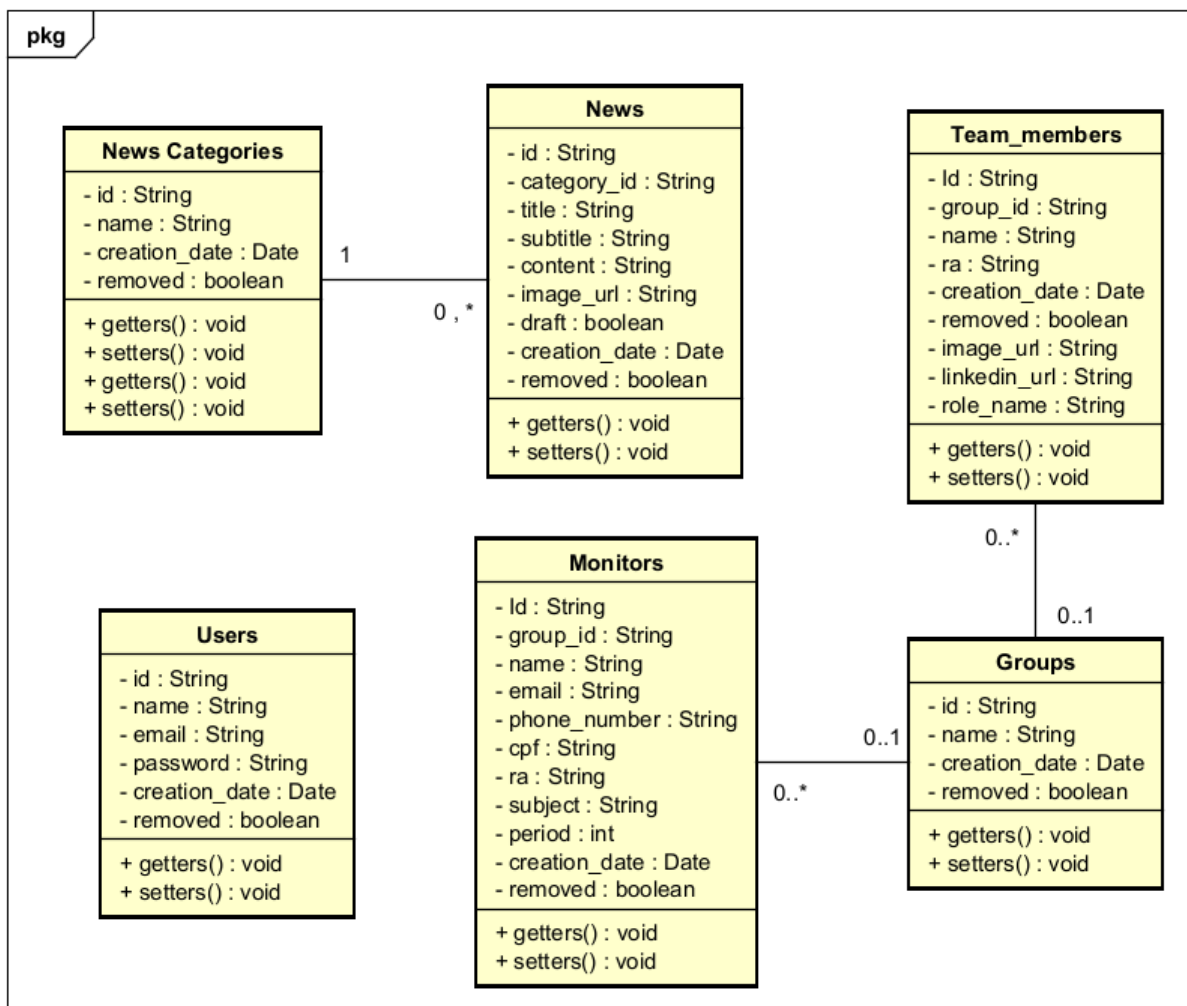


Figura 05: Diagrama de Classes

5.3 Diagrama de Atividades

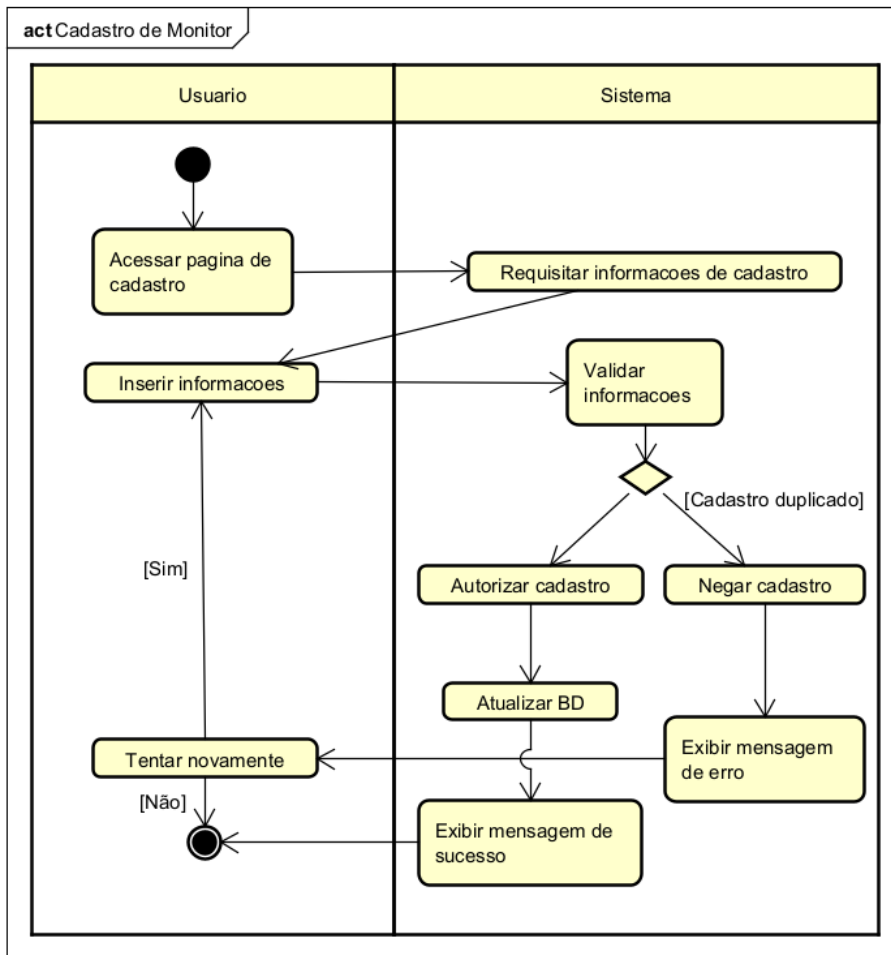


Figura 06: Diagrama de Atividades do Cadastro de Monitor

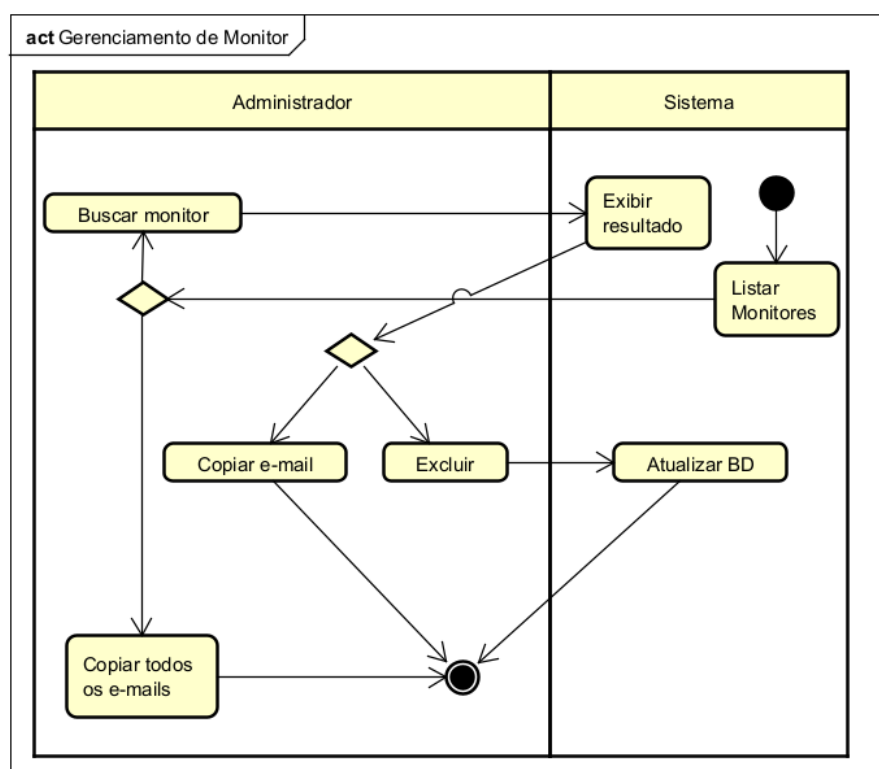


Figura 07: Diagrama de Atividades do Gerenciamento de Monitor

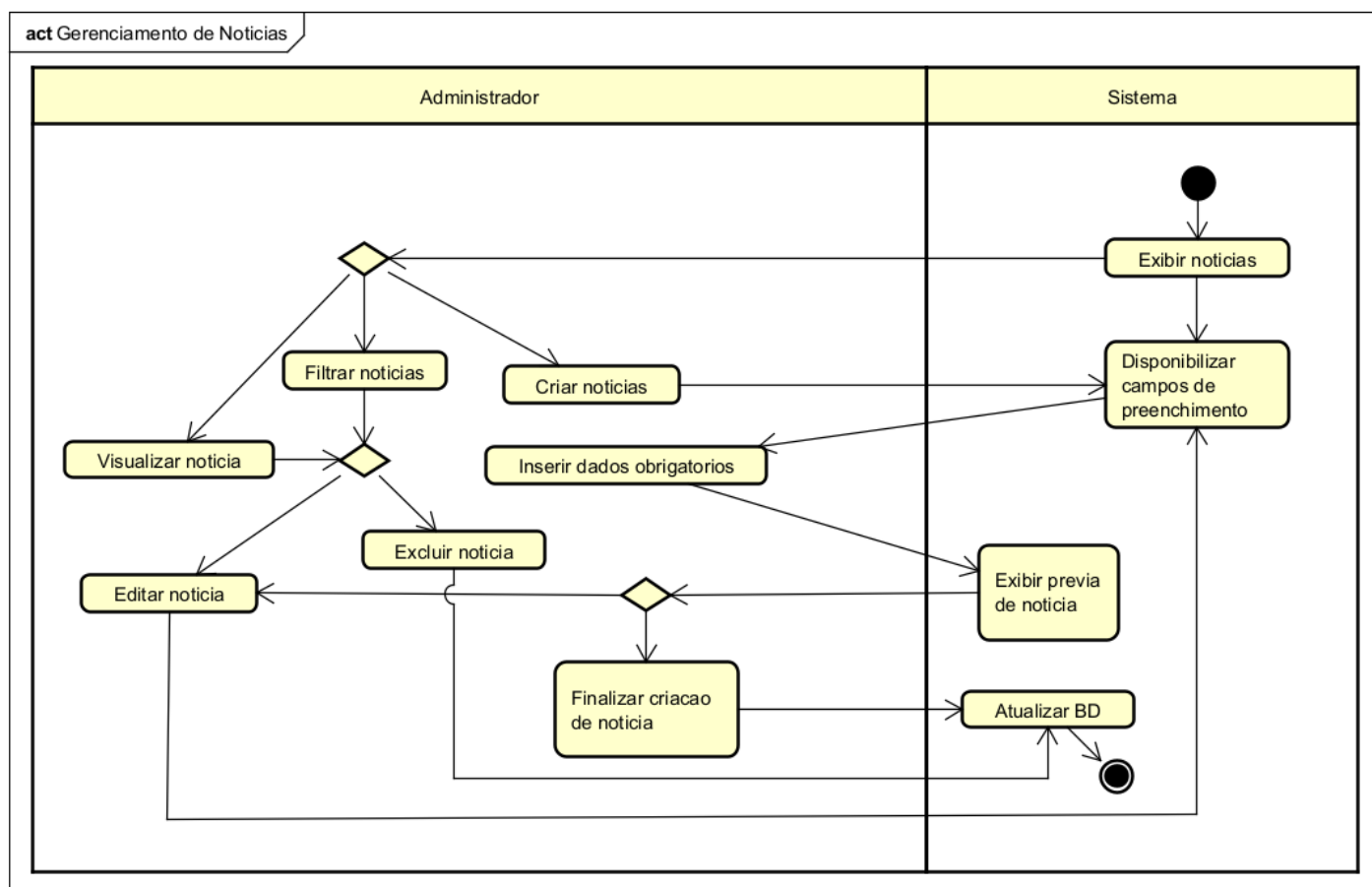


Figura 08: Diagrama de Atividades do Gerenciamento de Notícias

6 Implementação

6.1 Descrição do código

Descrever o sistema quanto ao código gerado. Explicar a organização dos arquivos, pacotes, classes ou quaisquer estruturas utilizadas no desenvolvimento do projeto, listando os componentes criados e sua estrutura. Use diagramas (Diagrama de Componentes, Diagrama de Pacotes) para ilustrar a implementação.

Descrever também convenções e padronizações para comentários no código, nomenclatura de classes, objetos, funções, etc. Se necessário, use exemplos.

6.2 Implantação

Explicar passo-a-passo para execução do software desenvolvido. Citar os requisitos necessários, configuração e providências para execução do projeto entregue. Se necessário, usar links, referências ou indicação dos recursos imprescindíveis para execução.

6.3 Telas principais

Apresentar as telas mais significativas do sistema, aquelas importantes para demonstração do seu funcionamento. Assim como nos protótipos, cada tela deve ser acompanhada de uma descrição sucinta de seu objetivo e sua dinâmica de navegação. O objetivo aqui é demonstrar o produto final.

7 Considerações Finais

Apresentar e discutir os resultados obtidos e sua aplicabilidade. Abordar o que foi atingido e o que não foi, as limitações, possíveis integrações com outros projetos e continuação do sistema em trabalhos futuros.

8 Bibliografia

Apresentar todas as obras (livros, artigos, Internet, revistas, etc...) utilizadas na elaboração da documentação e na implementação do projeto.