



CURSO DE BACHARELADO EM ENGENHARIA DE SOFTWARE

VITOR HUGO OLIVEIRA PALOCO

MARCELO YUKIO YAMAMOTO

SISTEMA DE HORÁRIOS - UNIFIL

Londrina
2025

SUMÁRIO

1	INTRODUÇÃO	2
1.1	PROPOSTA E OBJETIVOS	3
1.2	JUSTIFICATIVA	3
2	DIAGRAMAS	4
2.1	Diagrama de Caso de Uso	4
2.2	Diagrama de Classes	4
2.3	Diagrama Entidade-Relacionamento (DER)	5
2.4	Diagrama de Implantação	6
2.5	Diagrama de Sequência	7
3	TELAS	8
3.1	Tela de Login e Cadastro	8
3.2	Tela de Listagem de Usuários	9
3.3	Tela de Formulário de Usuário	9
4	WORKFLOW (AS IS) NA NOTAÇÃO BPMN	10
5	RECURSOS E AMBIENTE DE DESENOVOLVIMENTO	11
5.1	Linguagem de Programação	11
5.2	Banco de dados	11
5.3	Outras ferramentas	11
6	CRONOGRAMA	12
	REFERÊNCIAS	13

1 INTRODUÇÃO

Este documento descreve o estágio supervisionado realizado no contexto do desenvolvimento do "Sistema de Horários", um projeto destinado à gestão e organização dos cronogramas acadêmicos dos cursos de Computação do Centro Universitário Filadélfia (UniFil). O projeto é conduzido pelo Núcleo de Prática de Informática (NPI) e tem como principais envolvidos o desenvolvedor do projeto (autor deste documento), o orientador do NPI, que atua como principal stakeholder, e o professor orientador, responsável pelo acompanhamento acadêmico do estágio.

A proposta deste estágio surgiu a partir da identificação de um problema recorrente no gerenciamento dos horários acadêmicos: a utilização de planilhas para a elaboração e distribuição dos cronogramas de aulas. Este método, além de ser propenso a erros, demanda um alto esforço para manutenção, especialmente diante de mudanças frequentes nos horários. A dificuldade em manter os alunos informados sobre alterações também se mostrou um obstáculo significativo.

Diante desse contexto, o estágio supervisionado tem como foco o desenvolvimento de um sistema web para centralizar o gerenciamento e consulta de horários acadêmicos. Esse sistema permitirá que os professores elaborem e modifiquem cronogramas de forma eficiente e que os alunos acessem suas grades horárias de maneira intuitiva e automatizada. O desenvolvimento do sistema utiliza tecnologias como Laravel no backend e Vue.js no frontend, garantindo escalabilidade e facilidade de manutenção.

No decorrer deste documento, serão apresentados os objetivos do projeto, suas funcionalidades principais, os desafios enfrentados e as soluções propostas. Também será detalhado o processo de desenvolvimento, incluindo a escolha das tecnologias, requisitos funcionais e não funcionais, e os critérios de sucesso estabelecidos pelo orientador do NPI e pelo professor orientador.

Além de aprimorar a gestão acadêmica, este sistema busca facilitar a rotina dos professores e proporcionar aos alunos um acesso mais rápido e prático às informações sobre suas aulas. A automatização desse processo reduz o risco de erros e melhora a comunicação dentro da instituição. Com a implementação dessa solução, espera-se um impacto positivo no cotidiano acadêmico, trazendo mais organização e eficiência para todos os envolvidos. O desenvolvimento desse projeto representa um desafio técnico e ao mesmo tempo uma oportunidade valiosa de aprendizado prático na área de engenharia de software.

Assim, este trabalho tem como finalidade descrever a experiência do estágio, e demonstrar como o Sistema de Horários pode impactar positivamente a gestão acadêmica, oferecendo uma solução eficiente para um problema real da instituição.

1.1 PROPOSTA E OBJETIVOS

O presente projeto tem como objetivo geral desenvolver um sistema web eficiente para a gestão e organização dos horários acadêmicos, substituindo o uso de planilhas e otimizando o processo de atualização e consulta dos cronogramas acadêmicos. Como objetivos específicos de sistema, temos:

- Criar uma plataforma intuitiva e acessível para professores e alunos acessarem e gerenciarem seus horários.
- Automatizar o processo de atualização dos cronogramas acadêmicos, reduzindo a necessidade de intervenção manual.
- Garantir que os alunos possam visualizar apenas os horários referentes às suas respectivas turmas.
- Assegurar que o sistema seja escalável e preparado para futuras integrações, como geração automática de horários por inteligência artificial.
- Implementar mecanismos de segurança para proteger os dados acadêmicos e garantir a integridade das informações armazenadas.

1.2 JUSTIFICATIVA

O Sistema de Horários proposto permitirá a gestão centralizada dos cronogramas acadêmicos dos cursos de Computação da UniFil. Professores poderão criar e atualizar horários de forma rápida e prática, enquanto os alunos terão acesso direto às suas grades horárias sem a necessidade de buscar informações em arquivos extensos.

Atualmente, a gestão dos horários acadêmicos é feita manualmente, utilizando planilhas e documentos compartilhados, o que torna o processo suscetível a erros, desatualizações e dificuldades na comunicação entre professores e alunos. A proposta do Sistema de Horários justifica-se por atender a uma demanda real do Núcleo de Prática de Informática (NPI), trazendo mais eficiência, transparência e agilidade para a organização dos horários acadêmicos. Além disso, o sistema possibilita futuras expansões, como a integração com inteligência artificial para otimização dos cronogramas, garantindo uma solução moderna e adaptável às necessidades da instituição.

2 DIAGRAMAS

2.1 DIAGRAMA DE CASO DE USO

O diagrama de caso de uso é uma representação gráfica das interações entre atores externos e um sistema de software. Ele descreve as relações entre os diferentes casos e atores. Os elementos principais incluem atores, casos de uso e relacionamentos entre eles. É usado para capturar e comunicar os requisitos funcionais de um sistema de forma clara e compreensível, como pode ser verificado na Figura 1.

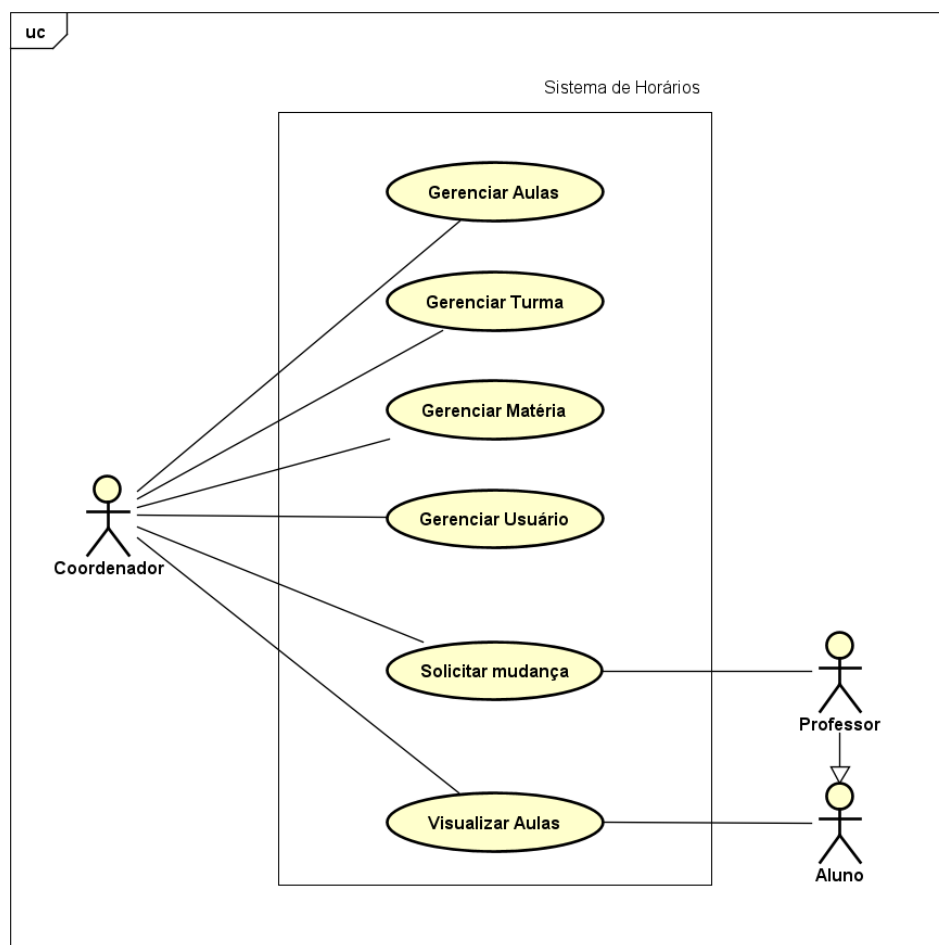


Figura 1 – Diagrama de Caso de Uso (Sistema de Horários)

2.2 DIAGRAMA DE CLASSES

O diagrama de classes descreve a estrutura estática do sistema, mostrando as classes que o compõem, seus atributos, métodos e os relacionamentos entre elas. Ele facilita a visualização da organização do código e o entendimento de como as enti-

dades do sistema interagem. A Figura 2 apresenta o diagrama de classes do Sistema de Horários.

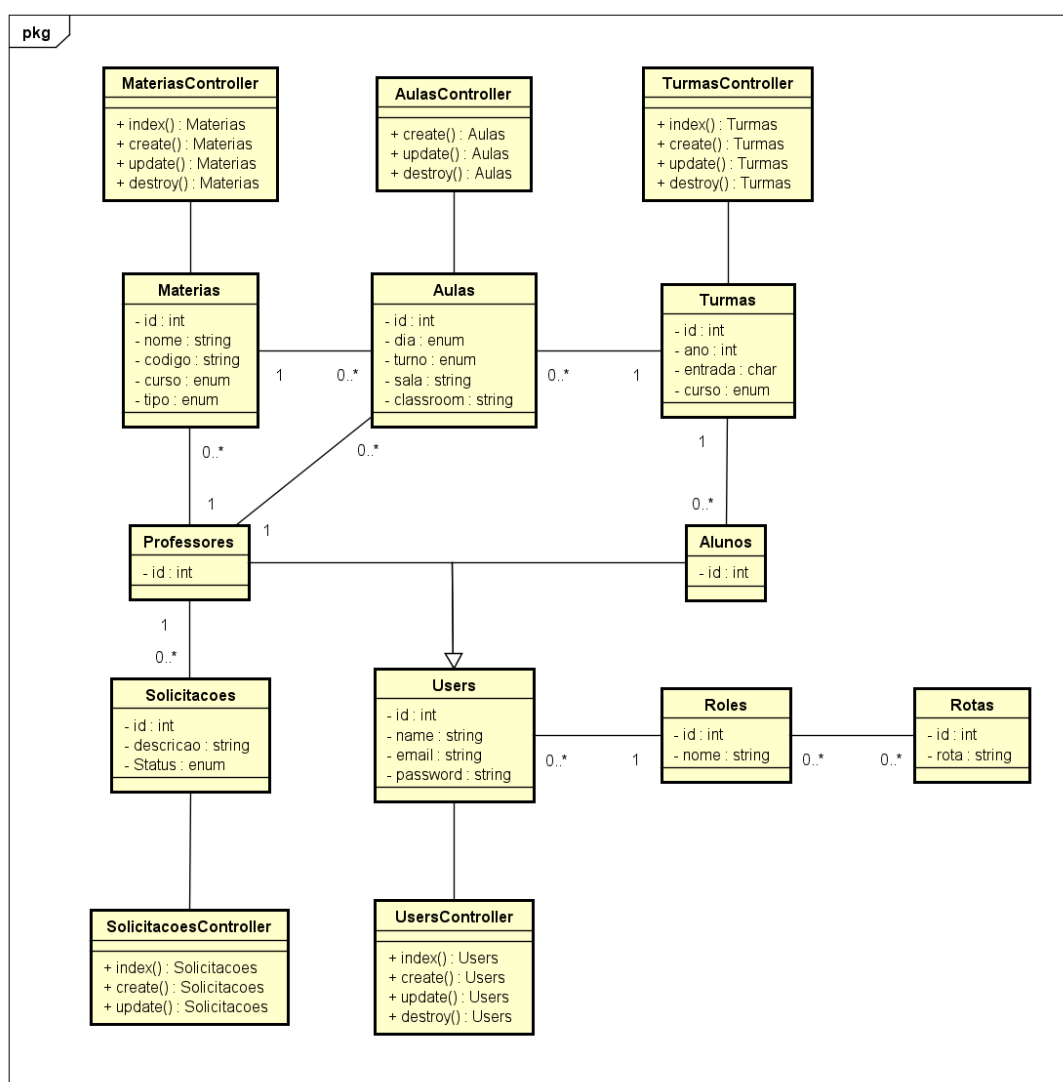


Figura 2 – Diagrama de Classe (Sistema de Horários)

2.3 DIAGRAMA ENTIDADE-RELACIONAMENTO (DER)

O diagrama Entidade-Relacionamento (DER) modela os dados do sistema, identificando as entidades principais, seus atributos e os relacionamentos entre elas. Ele serve como base para a criação do banco de dados relacional. A Figura 3 exibe o DER do Sistema de Horários.

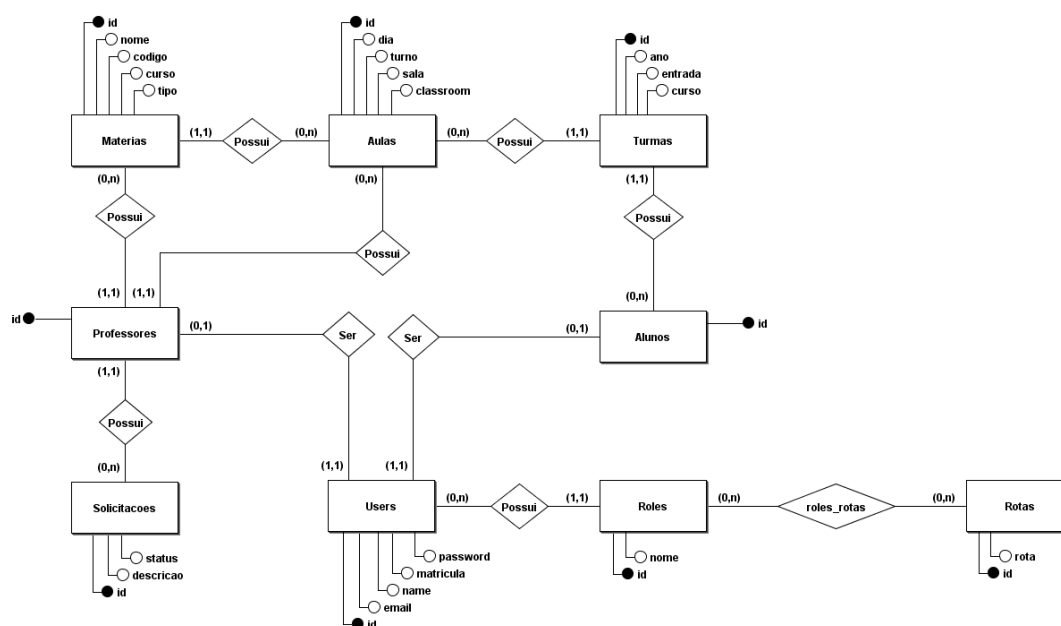


Figura 3 – Diagrama Entidade-Relacionamento (Sistema de Horários)

2.4 DIAGRAMA DE IMPLANTAÇÃO

O diagrama de implantação mostra a distribuição física dos componentes de software em diferentes nós de hardware. Ele auxilia no planejamento da infraestrutura necessária para executar o sistema. A Figura 4 apresenta o diagrama de implantação do Sistema de Horários.

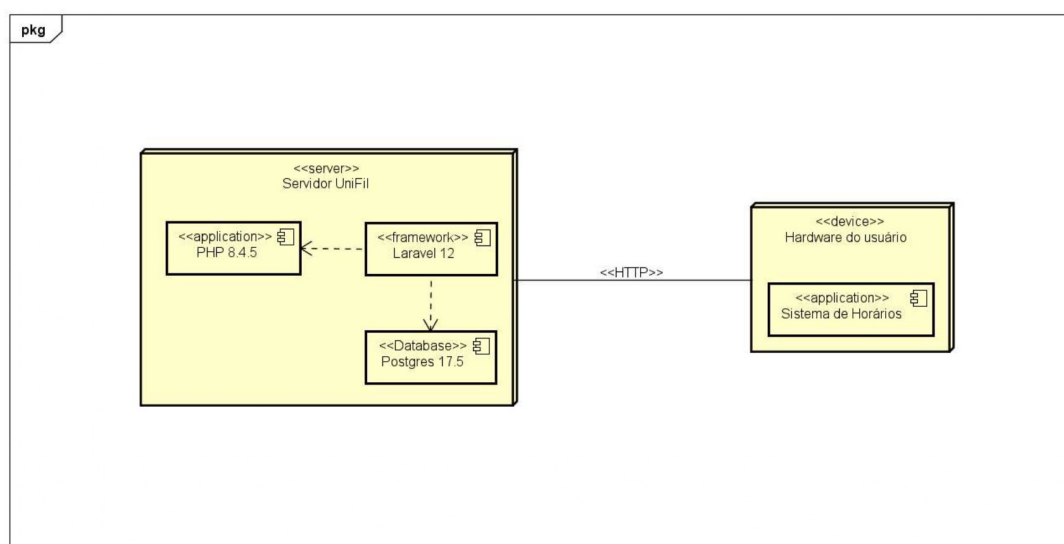


Figura 4 – Diagrama de Implantação (Sistema de Horários)

2.5 DIAGRAMA DE SEQUÊNCIA

O diagrama de sequência ilustra a interação temporal entre os objetos envolvidos em um determinado caso de uso. Ele mostra a ordem das mensagens trocadas entre os componentes do sistema. A Figura 5 apresenta o diagrama de sequência para o caso de uso "Gerenciar Usuários".

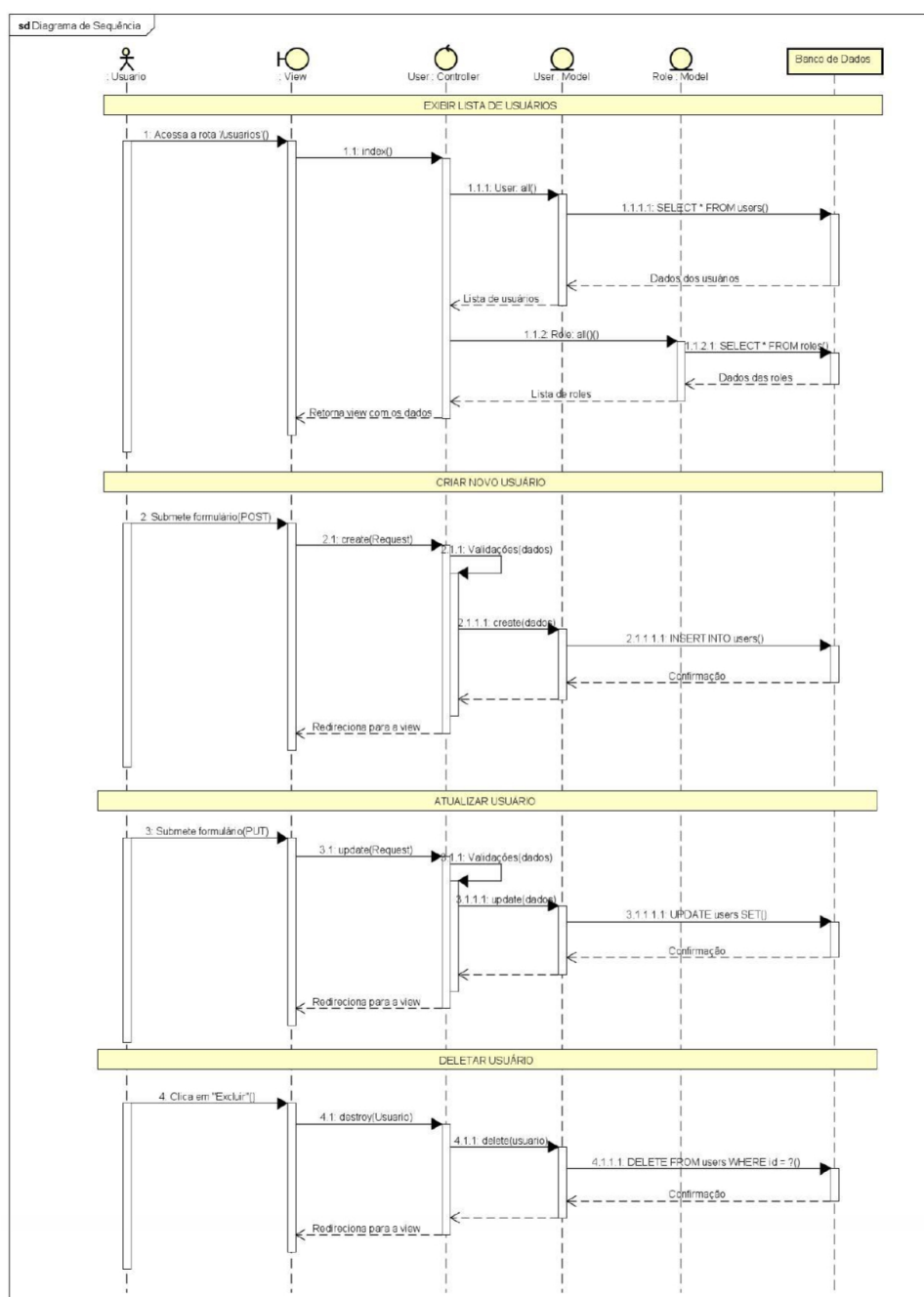


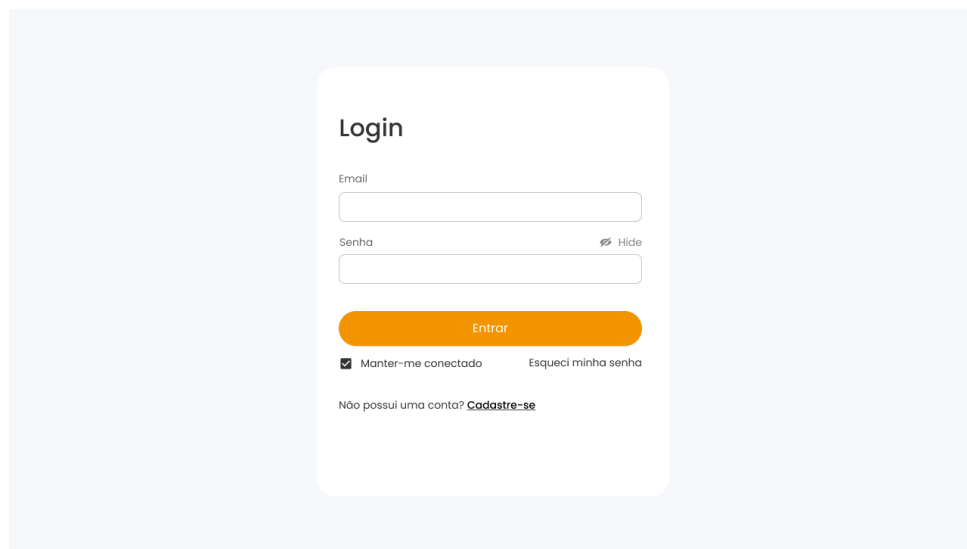
Figura 5 – Diagrama de Sequência (Sistema de Horários)

3 TELAS

As telas a seguir (Figura 6 à Figura 9) representam como o sistema se parecerá, Acesse o Figma para melhor visualização.

3.1 TELA DE LOGIN E CADASTRO

A tela de login permite que os usuários insiram suas credenciais para acessar o sistema, conforme mostrado na Figura 6. A tela de cadastro permite o registro de novos usuários no sistema, como visto na Figura 7.

A mockup of a login form titled "Login" centered on a light blue background. The form is white with rounded corners and contains the following elements: an "Email" label above a text input field; a "Senha" label above a password input field with a "Hide" toggle icon; an orange "Entrar" button; a checkbox labeled "Manter-me conectado" and a link "Esqueci minha senha"; and a link "Cadastre-se" preceded by the text "Não possui uma conta?".

Login

Email

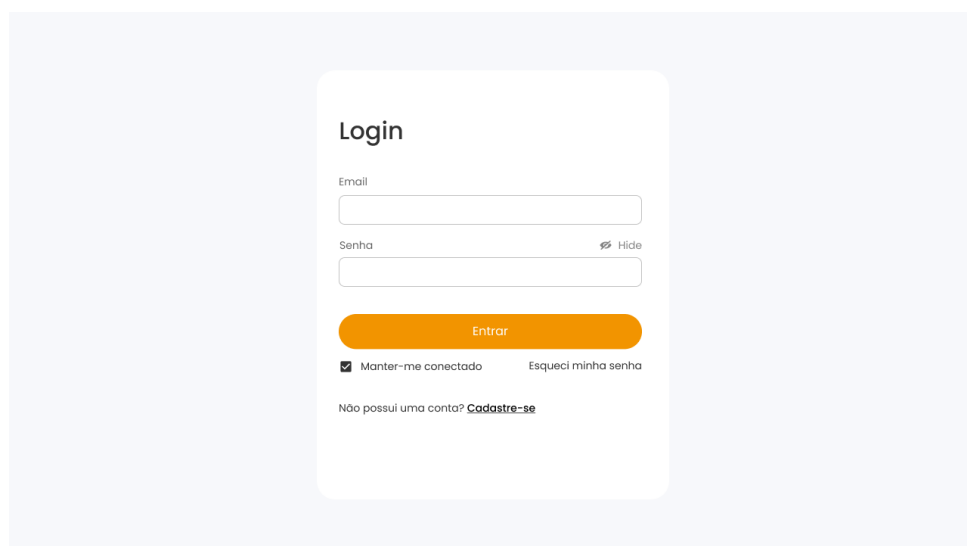
Senha Hide

Entrar

☒ Manter-me conectado [Esqueci minha senha](#)

Não possui uma conta? [Cadastre-se](#)

Figura 6 – Tela de login

A mockup of a login form titled "Login" centered on a light blue background. The form is white with rounded corners and contains the following elements: an "Email" label above a text input field; a "Senha" label above a password input field with a "Hide" toggle icon; an orange "Entrar" button; a checkbox labeled "Manter-me conectado" and a link "Esqueci minha senha"; and a link "Cadastre-se" preceded by the text "Não possui uma conta?".

Login

Email

Senha Hide

Entrar

☒ Manter-me conectado [Esqueci minha senha](#)

Não possui uma conta? [Cadastre-se](#)

Figura 7 – Tela de cadastro

3.2 TELA DE LISTAGEM DE USUÁRIOS

A tela mostrada na Figura 8 permite que o coordenador consulte e manipule as informações dos usuários. Seguindo um padrão de layout e regras de negócio, as demais telas de gerenciamento (matérias, turmas e solicitações) serão bastante semelhantes.

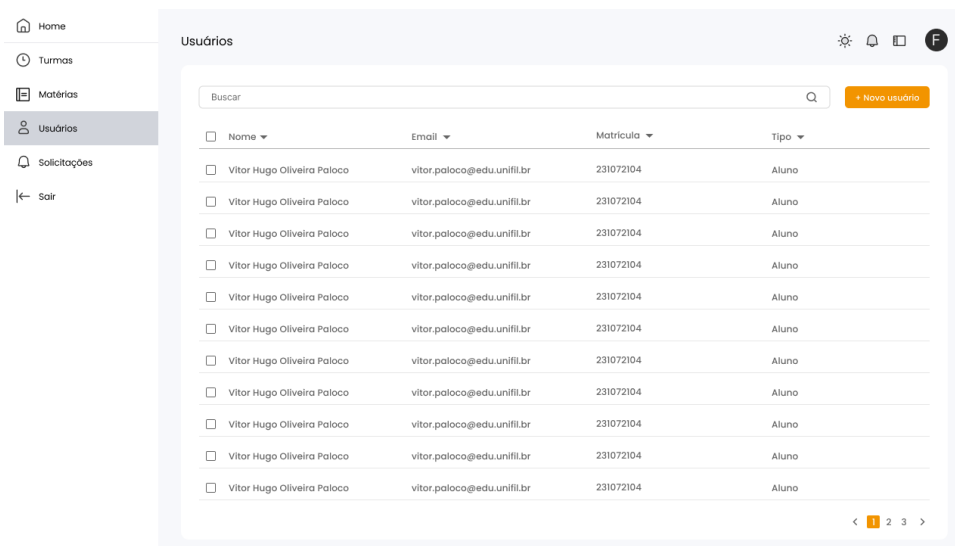


Figura 8 – Tela de listagem de usuários

3.3 TELA DE FORMULÁRIO DE USUÁRIO

A Figura 9 exibe o formulário utilizado para a criação e atualização de usuários. Esta interface permite ao coordenador inserir ou editar informações detalhadas de cada usuário, como nome, e-mail, perfil de acesso e outros dados relevantes.

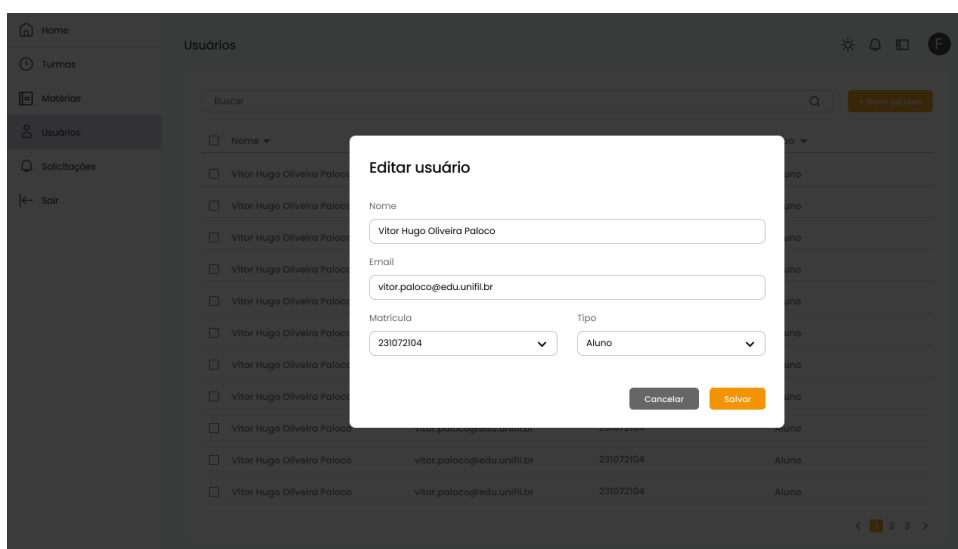


Figura 9 – Formulário de criação/atualização de usuários

4 WORKFLOW (AS IS) NA NOTAÇÃO BPMN

A Figura 10 apresenta o workflow do processo atual de criação dos horários das aulas, modelado na notação BPMN, representando o cenário As Is.

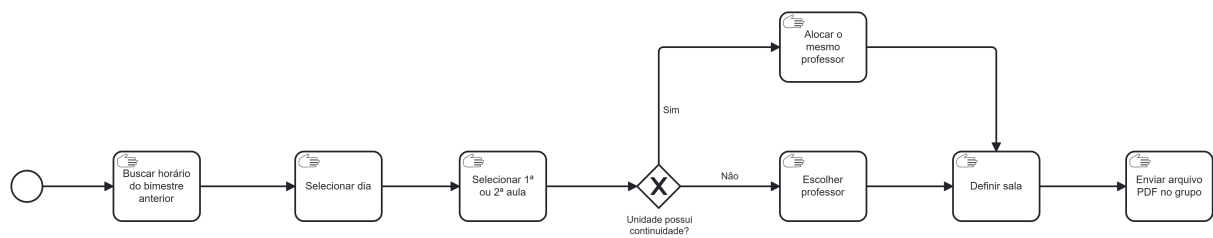


Figura 10 – Workflow AS-IS

5 RECURSOS E AMBIENTE DE DESENOVOLVIMENTO

5.1 LINGUAGEM DE PROGRAMAÇÃO

O sistema será desenvolvido utilizando a linguagem PHP 8.4.5 (PHP, 2025) no backend, com o framework Laravel 12 (LARAVEL, 2025), garantindo uma estrutura robusta e segura. No frontend, será utilizado Vue.js v3.5.13 (VUEJS, 2025), um framework JavaScript progressivo, que permitirá a criação de interfaces dinâmicas e responsivas. Além disso, serão aplicados padrões de arquitetura como MVC (Model-View-Controller) para garantir organização e escalabilidade do código.

5.2 BANCO DE DADOS

O banco de dados escolhido para o projeto é o PostgreSQL 17.4 (POSTGRESQL, 2025), utilizando PDO (PHP Data Objects) para garantir flexibilidade e segurança na manipulação de dados. A escolha do PostgreSQL se deve à sua capacidade de suportar estruturas mais complexas e funcionalidades avançadas em comparação ao MySQL, tornando-se mais adequado para sistemas que exigem maior robustez e escalabilidade.

5.3 OUTRAS FERRAMENTAS

- Git e GitHub: Controle de versão e gerenciamento do código-fonte.
- DBeaver: Ferramenta de gerenciamento do Banco de Dados PostgreSQL

6 CRONOGRAMA

REFERÊNCIAS

LARAVEL. *Laravel Documentation*. [S.l.], 2025. Acesso dia 27 de março de 2025. Disponível em: <<https://laravel.com/docs/12.x>>. 11

PHP. *PHP Documentation*. [S.l.], 2025. Acesso dia 27 de março de 2025. Disponível em: <<https://www.php.net/docs.php>>. 11

POSTGRESQL. *Sobre PostgreSQL*. [S.l.], 2025. Acesso dia 27 de março de 2025. Disponível em: <<https://www.postgresql.org/about/>>. 11

VUEJS. *Vue.js Documentation*. [S.l.], 2025. Acesso dia 27 de março de 2025. Disponível em: <<https://vuejs.org/guide/introduction>>. 11