

# Desenvolvimento de uma Plataforma Web para Gerenciamento do Plano Individual Docente do IFFar/FW

André Augusto Boniatti<sup>1</sup>, Árton Pereira Dorneles<sup>2</sup>

<sup>1</sup>Curso de Bacharelado em Ciência da Computação do Instituto Federal de Educação, Ciência e Tecnologia Farroupilha (IFFar) – Frederico Westphalen – RS – Brasil

<sup>2</sup>Instituto Federal de Educação, Ciência e Tecnologia Farroupilha (IFFar) – Frederico Westphalen – RS – Brasil

[andre123boniatti@gmail.com](mailto:andre123boniatti@gmail.com), [arton.dorneles@iffarroupilha.edu.br](mailto:arton.dorneles@iffarroupilha.edu.br)

**Abstract.** *This paper presents the development of a web platform for managing the Individual Teaching Plan (PID) at the Farroupilha Federal Institute (Frederico Westphalen Campus). The system aims to streamline faculty activity planning by enabling the registration, editing, and approval of plans, as well as providing comprehensive reports and public viewing of approved activities. Using Angular, .NET, and PostgreSQL, the solution integrates Google authentication and includes administrator functionalities for approving or rejecting submissions, meeting the primary requirements for flexibility and transparency in academic management.*

**Resumo.** *Este trabalho apresenta o desenvolvimento de uma plataforma web para gerenciamento do Plano Individual Docente (PID) no Instituto Federal Farroupilha (Campus Frederico Westphalen). O sistema busca otimizar o planejamento das atividades docentes, permitindo o cadastro, edição e homologação dos planos, além de oferecer relatórios consolidados e visualização pública das atividades aprovadas. Utilizando as tecnologias Angular, .NET e PostgreSQL, a solução integra um fluxo de autenticação com o Google e fornece funcionalidades de aprovação ou rejeição por parte dos administradores, atendendo às principais necessidades de flexibilidade e transparência na gestão acadêmica.*

## 1. Introdução

O presente trabalho tem como objetivo principal o desenvolvimento de uma plataforma web para a gestão do Plano Individual Docente (PID) dos professores do Instituto Federal Farroupilha - Campus Frederico Westphalen (IFFar-FW). A proposta surgiu da necessidade de criar uma ferramenta que otimize a administração das atividades docentes, facilitando tanto o processo de planejamento quanto a geração de relatórios personalizados que atendam às demandas da instituição.

Atualmente, o gerenciamento dos PIDs é realizado através do Sistema Integrado de Gestão de Atividades Acadêmicas (SIGAA), uma plataforma que, apesar de robusta, apresenta limitações que impactam diretamente na eficácia e transparência desse processo. Entre os principais problemas identificados estão a ausência de relatórios detalhados que permitam uma visão organizada das diferentes atividades (ensino, pesquisa, extensão) desempenhados pelos docentes, bem como mecanismos automatizados para tornar essas atividades transparentes para a comunidade externa e interna.

No contexto da gestão acadêmica, torna-se evidente a importância de sistemas que acompanhem a evolução das necessidades institucionais e que contribuam para uma

administração mais transparente e eficiente. Desta forma, o foco deste trabalho está em desenvolver uma ferramenta que supere as limitações do sistema atual, oferecendo também novas funcionalidades, como o acompanhamento em tempo real do preenchimento dos PIDs pelos docentes.

O restante deste texto encontra-se estruturado da seguinte forma: a Seção 2 apresenta os conceitos e as tecnologias utilizadas no desenvolvimento do sistema; na Seção 3 são descritos trabalhos correlatos à proposta; na Seção 4 detalha-se a metodologia de projeto adotada; em seguida, a Seção 5 expõe os principais resultados obtidos durante a implementação; e, por fim, a Seção 6 reúne as considerações finais e propõe futuros aprimoramentos.

## **2. Tecnologias utilizadas**

Nesta seção serão apresentadas as tecnologias utilizadas para o desenvolvimento do sistema, bem como os principais conceitos e definições necessários para a compreensão do trabalho.

### **2.1. C# e .NET**

O C# (C Sharp) é uma linguagem de programação moderna, orientada a objeto e fortemente tipada. A linguagem permite que os desenvolvedores criem muitos tipos de aplicativos seguros e robustos que são executados no .NET (MICROSOFT, 2024a). O .NET é uma plataforma de desenvolvimento gratuita multiplataforma e de código aberto que permite executar programas escritos em vários idiomas, sendo C# o mais popular. Além de ser mantido pela Microsoft e pela comunidade, a plataforma foi projetada para fornecer produtividade, desempenho, segurança e confiabilidade (MICROSOFT, 2024b).

### **2.2. JavaScript, TypeScript e Angular**

O JavaScript é uma linguagem leve e interpretada, usada com frequência como a linguagem de script para prover comportamento dinâmico em páginas web. É uma linguagem baseada em protótipos, multi-paradigma e dinâmica, suportando estilos de orientação a objetos, imperativos e declarativos (MDN, 2024a). O TypeScript, por sua vez, é baseado no JavaScript mas com sintaxe para tipos, oferecendo todas as vantagens de uma linguagem de programação fortemente tipada (MICROSOFT, 2024c).

O Angular é uma estrutura de design de aplicativos e uma plataforma de desenvolvimento para criar aplicativos de página única eficientes e sofisticados, mantida pelo Google e desenvolvida em TypeScript (GOOGLE, 2024a). Como plataforma, o Angular inclui: (a) uma estrutura baseada em componentes; (b) uma coleção de bibliotecas bem integradas que abrangem uma ampla variedade de recursos, incluindo roteamento, gerenciamento de formulários, comunicação cliente-servidor, e muito mais, e (c) um conjunto de ferramentas para auxiliar a desenvolver, compilar, testar e atualizar o código (GOOGLE, 2024b).

### **2.3. HTML e CSS**

HTML (Linguagem de Marcação de HiperTexto) é o bloco de construção mais básico da web. "Hipertexto" refere-se aos links que conectam páginas da Web entre si, seja dentro de um único site ou entre diferentes sites. Os links são um aspecto fundamental pois permitem carregar conteúdo na Internet e vinculá-lo a páginas criadas por outras pessoas. O HTML usa

"marcação" para anotar texto, imagem e outros conteúdos para exibição em um navegador da Web e criar a estrutura de um site ou aplicação (MDN, 2024b).

Por fim, o CSS (Folhas de Estilo em Cascata) é uma linguagem de estilo usada para descrever a apresentação de um documento escrito em HTML. O CSS descreve como elementos são exibidos na tela, no papel, na fala ou em outras mídias. A linguagem CSS é uma das principais tecnologias da open web e é padronizada em navegadores web de acordo com as especificações da W3C (MDN, 2024c).

## **2.4. PostgreSQL**

O PostgreSQL é um sistema de banco de dados objeto-relacional e de código aberto. Além de mais de 35 anos de desenvolvimento ativo, possui uma forte reputação de confiabilidade, robustez de recursos e desempenho (POSTGRESQL, 2024).

## **3. Trabalhos Relacionados**

Sistemas de gestão acadêmica são ferramentas cruciais que consolidam e gerenciam informações vitais sobre a comunidade acadêmica. Souza e Monteiro (2015) destacam que esses sistemas permitem o gerenciamento eficiente das atividades acadêmicas, consolidando informações essenciais para as instituições. À medida que a instituição cresce, a complexidade e a importância desses sistemas aumentam, integrando processos variados e gerando informações que auxiliam na tomada de decisões estratégicas. Grandes instituições usam sistemas integrados que abrangem todos os processos institucionais, melhorando a eficiência administrativa e acadêmica (Rocha Neto e Lima, 2009, apud Souza e Monteiro, 2015). Entre os sistemas acadêmicos existentes, podemos destacar o Sistema Integrado de Gestão de Atividades Acadêmicas (SIGAA), que é utilizado no Instituto Federal Farroupilha.

O SIGAA foi desenvolvido pela Superintendência de Informática da Universidade Federal do Rio Grande do Norte (UFRN) para interligar sistemas institucionais que até então operavam de forma isolada. Usando tecnologias livres, o SIGAA foi posteriormente disponibilizado para outras instituições federais brasileiras, incluindo o IFFar (Filho, Aquino, e Rosa, 2013). Segundo a própria UFRN, o SIGAA é um sistema web corporativo que informatiza os procedimentos acadêmicos por meio de diversos módulos que cobrem ensino, pesquisa, extensão, monitoria, bibliotecas, ingresso, apoio ao estudante, patrimônio, entre outros. Para a sala de aula virtual, o SIGAA oferece funcionalidades para professores e alunos, como planos de curso, fóruns, chat, notícias, controle de frequência e notas, além de ferramentas para material didático, enquetes, tarefas e questionários (Filho, Aquino, e Rosa, 2013). Esses recursos promovem uma gestão acadêmica e administrativa mais eficiente, facilitando a interação e o gerenciamento no ambiente universitário.

Uma das funções administrativas essenciais em instituições de ensino é o gerenciamento do Plano Individual Docente (PID), que deve ser feito uma vez no início de cada semestre pelos docentes da instituição e é utilizado para nortear seus trabalhos realizados durante um período letivo. A gestão do PID envolve, principalmente, a distribuição da carga horária semanal entre ensino, pesquisa, extensão e outras atividades conforme o regime de trabalho de cada docente. Mas além disso, é necessário gerir os horários em que cada atividade acontece, bem como organizar os relatórios de atividades ao final de cada semestre. O SIGAA oferece o módulo para gestão do PID que contempla apenas a distribuição da carga horária de atividades e homologação pela chefia imediata dos docentes. Dada a relevância na gestão do serviço de educação, vários estudos têm abordado o

desenvolvimento de sistemas para auxiliar na elaboração e gestão dos planos de trabalho docente, enfrentando desafios devido ao alto volume de dados e à necessidade de conformidade com regulamentos acadêmicos.

Em NATALI e CANDIDO (2023) os autores propõem automatizar o preenchimento do plano de trabalho docente, integrando informações de portarias, projetos e associações de professores para facilitar a elaboração e validação das cargas horárias. O processo de desenvolvimento deste sistema seguiu etapas definidas, partindo da coleta de requisitos junto aos professores e direção de ensino para entender as necessidades específicas, até a modelagem e descrição de casos de uso. A implementação foi realizada utilizando tecnologias Java para o back-end com banco de dados Postgres e, para o front-end foi usado o framework Angular em Javascript. Quanto aos recursos do sistema, incluem funcionalidades como níveis de acesso e alguns cadastros de atividades docentes, como atuação nos cursos, componentes curriculares, portarias, projetos, atividades de pesquisa e extensão.

## 4. Metodologia

Nessa seção serão apresentados os processos utilizados para o desenvolvimento do projeto do sistema: o levantamento dos requisitos funcionais e não funcionais, o desenvolvimento do diagrama de casos de uso e a modelagem do banco de dados da aplicação.

### 4.1. Análise de Requisitos

De acordo com Sommerville (2011), os requisitos de um sistema são as descrições de como o sistema deve agir, os serviços que oferece e as restrições do seu funcionamento. Esses requisitos refletem a finalidade do sistema a fim de atender as necessidades dos usuários e frequentemente são classificados como requisitos funcionais ou não funcionais.

Os requisitos funcionais descrevem o que o sistema deve fazer, como ele deve reagir a entradas específicas, como deve se comportar em determinadas situações e, em alguns casos, também podem descrever o que o sistema não deve fazer. Já os requisitos não funcionais definem restrições aos serviços ou funções oferecidos pelo sistema. Além disso, ao invés de especificarem características individuais ou serviços do sistema, normalmente eles definem características que se aplicam ao sistema como um todo.

Na Tabela 1 são apresentados os requisitos funcionais (RF01-RF08) e não funcionais (RNF01-RNF03) elaborados de acordo com o estudo das necessidades do sistema proposto.

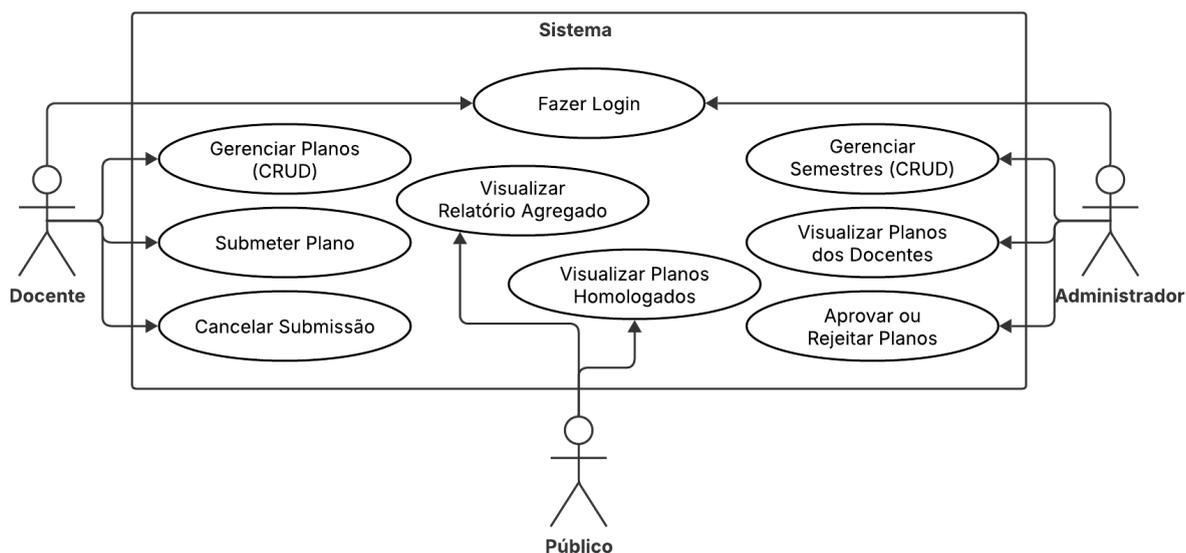
Id	Requisito	Descrição
RF01	Autenticação de Usuários	Permitir que docentes e administradores façam login no sistema utilizando suas credenciais do Google, garantindo acesso seguro às funcionalidades.
RF02	Cadastro de Semestres	Possibilitar que administradores criem e gerenciem os semestres (por exemplo, "2025/2"), definindo ano, semestre e outras informações pertinentes.
RF03	Gestão de Planos pelos Docentes	Permitir que cada docente crie, edite e visualize o próprio Plano Individual Docente (PID) em um semestre, preenchendo as informações de atividades e carga horária semanal.
RF04	Submissão de Plano	Disponibilizar a ação de submeter o plano para análise do administrador, alterando o status para "enviado" e bloqueando edições até que o plano seja aprovado ou rejeitado.

RF05	Cancelamento de Submissão de Plano	Permitir ao docente que cancele a submissão enquanto o status for “enviado”, retornando o status para “pendente” e liberando a edição do plano.
RF06	Aprovação ou Rejeição de Plano	Habilitar o administrador a aprovar (definindo o status como “homologado”) ou rejeitar planos que estejam com status “enviado”. Em caso de rejeição, obrigar o preenchimento do motivo.
RF07	Visualização Pública de Planos Homologados	Permitir que qualquer usuário (público em geral) visualize, na página inicial, os planos com status “homologado” para cada semestre.
RF08	Relatório Agregado de Semestre	Gerar e disponibilizar para visualização um relatório consolidado dos planos homologados de cada semestre, com possibilidade de filtrar os tipos de atividades ou ver todas as atividades agregadas.
RNF01	Desempenho	O sistema deve responder às requisições de visualização de planos e relatórios em até 3 segundos, em condições normais de acesso.
RNF02	Segurança	Implementar protocolos de comunicação segura (HTTPS) e seguir boas práticas de autenticação OAuth2 para acesso via Google, garantindo a proteção dos dados dos usuários.
RNF03	Usabilidade	O sistema deve possuir uma interface intuitiva, adaptada para dispositivos móveis, facilitando a navegação e o preenchimento das informações.

**Tabela 1. Levantamento de requisitos**

## 4.2. Diagrama de Casos de Uso

De acordo com Sommerville (2011), os casos de uso identificam as interações individuais entre o sistema e seus usuários ou outros sistemas a fim de identificar quais são os requisitos necessários. Cada caso de uso é documentado com um diagrama de alto nível e com uma descrição textual. Seu conjunto representa todas as possíveis interações que serão descritas nos requisitos do sistema. De acordo com os requisitos levantados na seção anterior foi elaborado o diagrama de casos de uso da Figura 1.



**Figura 1. Diagrama de casos de uso**

O diagrama apresenta três atores: docente, administrador e público em geral. O docente pode fazer login no sistema, cadastrar e gerenciar seu plano (incluir atividades, editar e excluir informações), submeter o plano para análise e cancelar a submissão enquanto não estiver homologado. O administrador pode fazer login no sistema, cadastrar os semestres, visualizar os planos dos docentes e aprovar ou rejeitá-los. Ao rejeitar um plano, o administrador deve informar o motivo, permitindo que o docente faça os ajustes necessários. Por fim, o público em geral tem acesso à visualização dos planos que já foram homologados em cada semestre, bem como a um relatório agregado de cada período, sem a necessidade de realizar login.

### 4.3. Modelagem do banco de dados

Conforme Sommerville (2011), a modelagem de dados em engenharia de software consiste em três etapas: Conceitual, Lógica e Física. Na modelagem conceitual realizada a criação de um modelo abstrato usando diagramas de entidade-relacionamento para representar dados, suas relações e atributos, focando nas necessidades do usuário sem restrições técnicas. Na modelagem lógica é feita a transformação do modelo conceitual em um esquema de banco de dados normalizado, adequado para um Sistema de Gerenciamento de Banco de Dados específico, incluindo a definição de tabelas, chaves e restrições de integridade. Finalmente, na modelagem física é realizada a implementação do modelo lógico no SGBD, especificando detalhes como tipos de dados e índices para otimização nas operações do banco de dados.

O modelo lógico do sistema desenvolvido é apresentado na Figura 2 e é composto por cinco tabelas. A tabela “Users” armazena as informações dos usuários (docentes e administradores), incluindo nome, e-mail, tipo de usuário e carga horária. A tabela “Periods” registra cada período letivo (ex.: 2025/2), com campos como ano e semestre, além de dados de criação e atualização; a tabela “Plans” vincula um docente a um período, armazenando a situação do plano (pendente, enviado, homologado), motivo de rejeição e datas de criação/atualização. A tabela “PlanActivities” detalha as atividades que compõem cada plano, relacionando-as a um tipo de atividade e possibilitando indicar descrições e alocações de carga horária. Finalmente, a tabela “ActivityTypes” lista e categoriza os tipos de atividades possíveis, como ensino, pesquisa, extensão, entre outras.

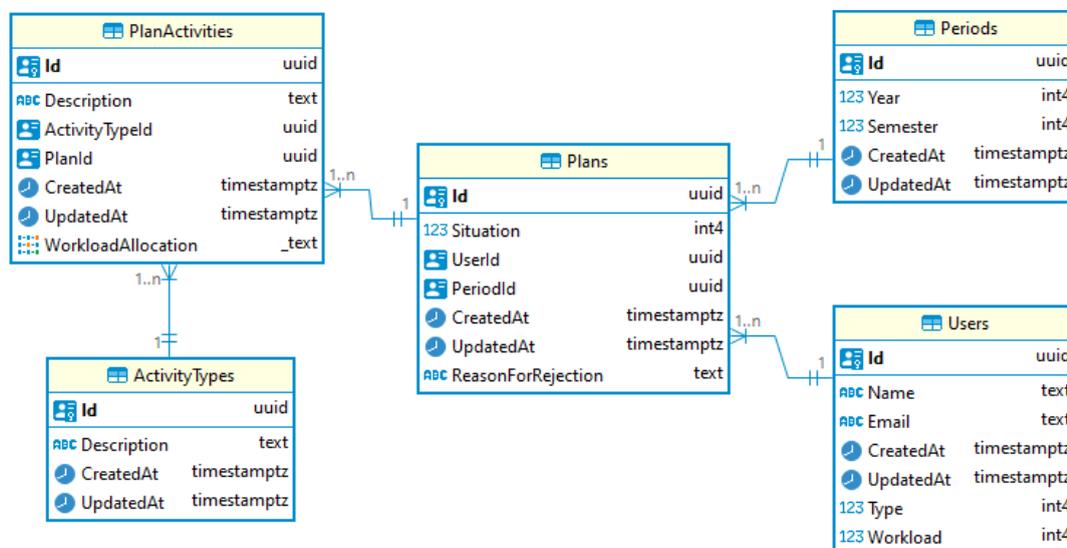


Figura 2. Modelo lógico do banco de dados do sistema

## 5. Desenvolvimento e Resultados

Com as etapas metodológicas finalizadas, deu-se início ao desenvolvimento do sistema, cujo código-fonte encontra-se disponível no GitHub em: <https://github.com/AndreBoniatti/PID>. Nas seções seguintes é apresentado o fluxo dos planos individuais docentes e as principais telas do sistema desenvolvido.

### 5.1. Fluxo dos Planos Individuais Docentes

O sistema proposto tem como principal premissa gerenciar os planos individuais docentes ao longo de vários períodos letivos. Esse gerenciamento exige o acompanhamento do status de cada plano, que pode variar entre “Pendente”, “Enviado” e “Homologado” conforme o fluxo apresentado na Figura 3. Em cada período letivo, cada docente tem seu plano iniciando com o status "Pendente", situação em ele deve cadastrar e editar suas atividades. Ao finalizar o preenchimento, ele pode submeter o plano para análise, alterando o status para "Enviado". Nesse status, o docente ainda pode cancelar a submissão caso precise realizar ajustes, retornando o status novamente para "Pendente". Se mantido como "Enviado", o plano será avaliado pelo administrador, que pode aprovar ou rejeitar. Caso rejeitado, o status volta a ser "Pendente", permitindo que o docente faça correções e reenvie o plano. Se aprovado, o status se torna "Homologado", finalizando o processo e impedindo novas alterações. As implementações que mantêm esse fluxo organizado no sistema proposto serão apresentadas nas seções seguintes.

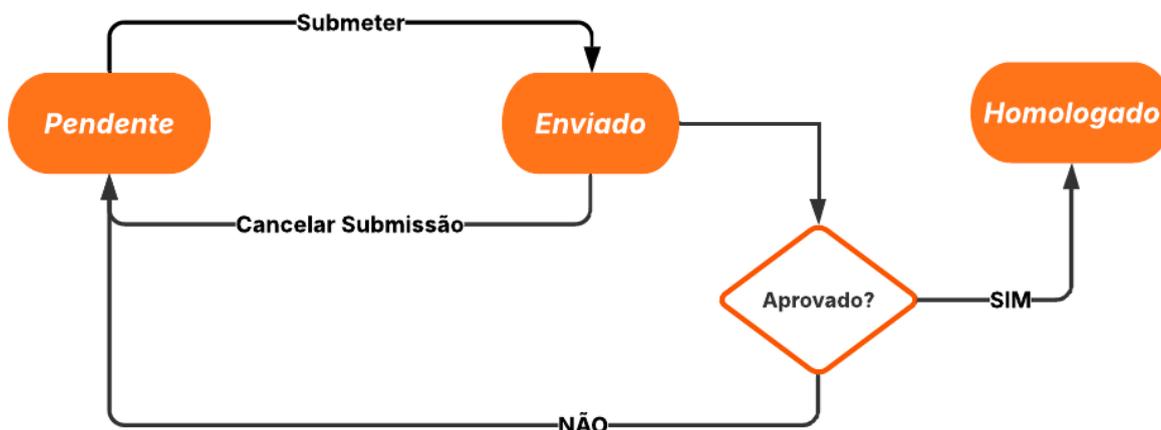
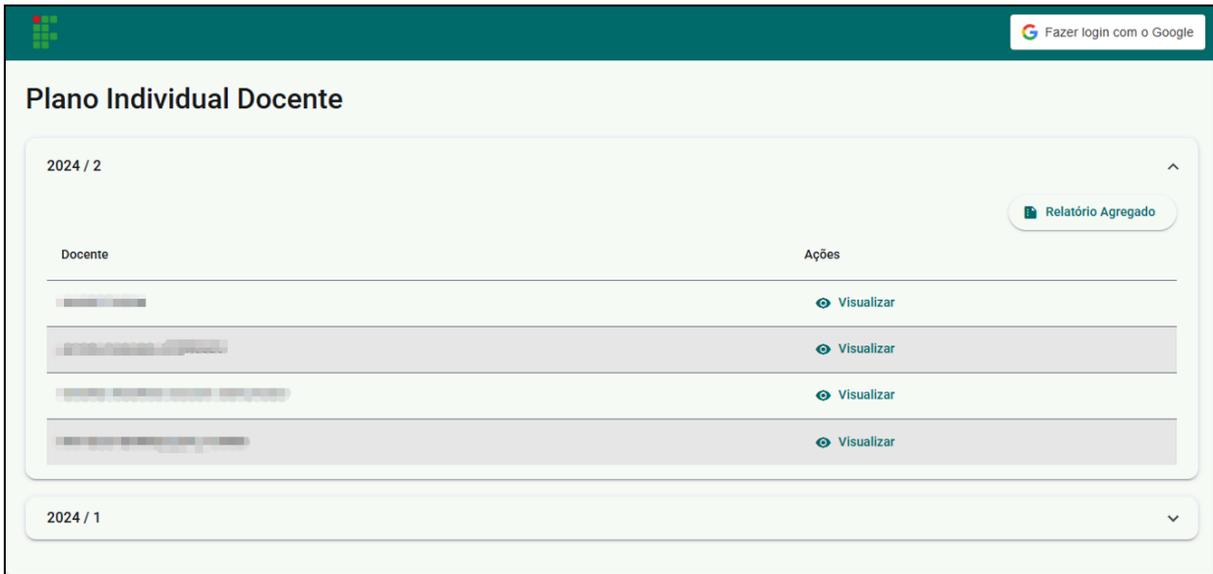


Figura 3. Fluxograma de status do plano

### 5.2. Acesso público

A tela inicial, apresentada na Figura 4, é acessível ao público em geral sem a necessidade de login. Nela são exibidos os períodos letivos cadastrados no sistema (como “2024 / 2”) e, para cada um deles, é apresentada uma listagem com os docentes que possuem planos homologados no período. Por meio do botão “Visualizar”, o público em geral consegue acessar o plano individual de cada docente.



**Figura 4. Página inicial**

Já o botão “Relatório Agregado” permite acessar um relatório reunindo informações consolidadas de todos os docentes, apresentado na tela da Figura 5.

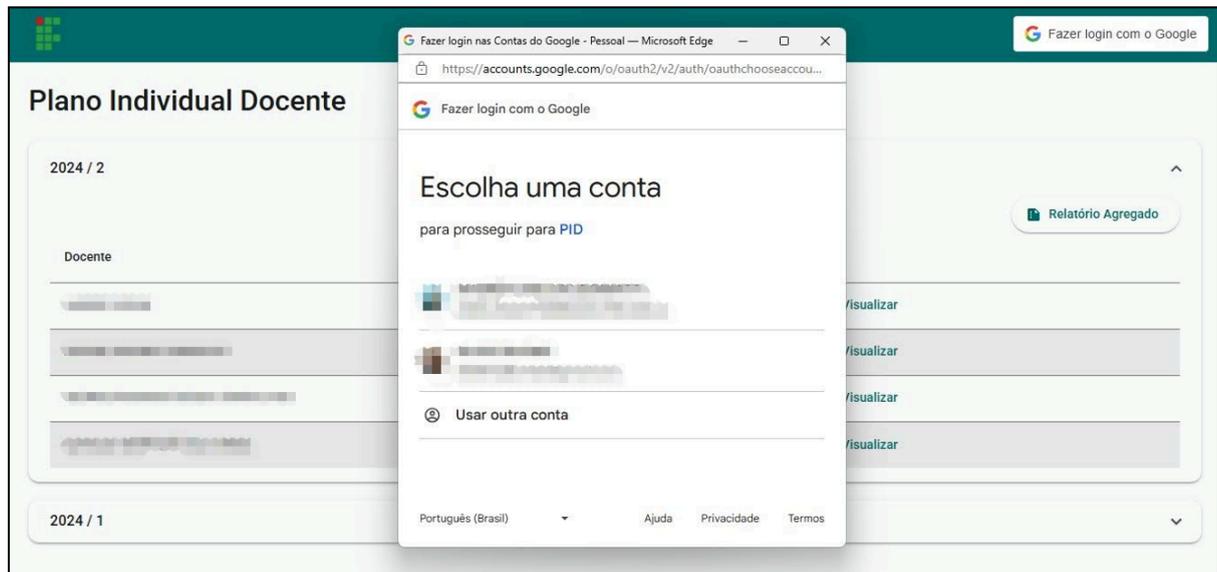
Horário	Seg	Ter	Qua	Qui	Sex	Sab
T1 13:20-14:10	Representação Institucional Funções Administrativas Funções Administrativas Funções Administrativas	Representação Institucional Funções Administrativas Funções Administrativas Funções Administrativas	Representação Institucional Funções Administrativas Funções Administrativas Funções Administrativas	Extensão Funções Administrativas Funções Administrativas Funções Administrativas	Extensão Pesquisa Funções Administrativas Funções Administrativas	
	Representação Institucional	Representação Institucional	Representação Institucional	Extensão	Extensão	

**Figura 5. Relatório agregado do semestre**

Nesse relatório é possível ver as atividades homologadas de todos os docentes organizadas em uma tabela. Cada célula desta tabela apresenta o nome do docente e os diferentes tipos de atividade organizadas em cores distintas. Este relatório fornece uma visão ampla da organização de atividades realizadas na instituição ao longo de cada período da semana. O usuário pode filtrar as atividades de acordo com o tipo como, pesquisa, ensino, extensão, entre outras.

Finalmente, na parte superior direita da tela inicial, encontra-se o botão “Fazer login com o Google”, que permite que docentes e administradores se autentiquem para acessar

funcionalidades restritas usando o seu e-mail institucional. Ao clicar no botão de login, o sistema redireciona o usuário para a tela de autenticação via Google apresentada na Figura 6, onde é possível escolher uma conta para prosseguir no acesso. Esse processo utiliza OAuth 2.0, permitindo que apenas docentes e administradores se autentiquem no sistema.



**Figura 6. Login com o Google**

### 5.3. Acesso do docente

Após realizar o login, os usuários docentes são redirecionados para a tela da Figura 7, que apresenta os períodos letivos cadastrados pelo administrador até o momento. É possível ver para cada período a situação em relação a submissão do plano. Em períodos com situação “Homologado” só é possível visualizar o conteúdo do plano clicando em “Visualizar”, sem possibilidade de alterá-lo.



Período	Situação	Ações
2024 / 2	Pendente	<a href="#">Editar</a>
2024 / 1	Homologado	<a href="#">Visualizar</a>

**Figura 7. Planos do docente**

Em períodos que aparecem com a situação “Pendente” é possível trabalhar na construção do plano clicando em “Editar”, levando o sistema para a tela da Figura 8. Nessa tela, o docente pode gerenciar seu plano, adicionando ou removendo atividades. No topo da tela, o sistema exibe a carga horária total que o usuário precisa registrar, uma lista com as atividades já cadastradas no plano e um contador que indica a quantidade de horas que ainda resta para cadastrar. Na parte inferior da tela, há uma grade de horários que exibe as atividades já planejadas em cores diferentes, facilitando a visualização. Além disso, estão disponíveis ações para “Editar” e “Deletar” atividades. Ao finalizar o plano, o docente pode

enviá-lo para avaliação, por meio do botão “Submeter”. Também é possível imprimir ou visualizar um PDF do plano por meio do botão “Imprimir”.

Atividade	Tipo	CH	Ações
Algoritmos e Programação	Ensino	7	<a href="#">Editar</a> <a href="#">Deletar</a>
Banco de Dados II	Ensino	5	<a href="#">Editar</a> <a href="#">Deletar</a>

Horário	Seg	Ter	Qua	Qui	Sex	Sab
M1 07:40-08:30	Banco de Dados II	Algoritmos e Programação	Algoritmos e Programação			
M2 08:30-09:20	Banco de Dados II	Algoritmos e Programação	Algoritmos e Programação			
M3 09:20-10:10	Banco de Dados II	Algoritmos e Programação				
M4 10:30-11:20	Banco de Dados II	Algoritmos e Programação				

**Figura 8. Plano em detalhes**

Ao clicar em “Adicionar Atividade” a tela da Figura 9 aparece para o docente.

Horário	Seg	Ter	Qua	Qui	Sex	Sab
M1 07:40-08:30	Banco de Dados II	Algoritmos e Programação	Algoritmos e Programação	Engenharia de Software II		
M2 08:30-09:20	Banco de Dados II	Algoritmos e Programação	Algoritmos e Programação	Engenharia de Software II		
M3 09:20-10:10	Banco de Dados II	Algoritmos e Programação	Engenharia de Software II	Engenharia de Software II		
M4 10:30-11:20	Banco de Dados II	Algoritmos e Programação	Engenharia de Software II	Engenharia de Software II		
M5 11:20-12:10	Banco de Dados II	Algoritmos e Programação	Engenharia de Software II	Engenharia de Software II		
T1 12:20-13:10						

**Figura 9. Adicionar atividade**

Nesta tela é possível selecionar o tipo de atividade (por exemplo, Ensino) e inserir uma descrição (como “Engenharia de Software II”), em seguida, escolher na grade de horários os blocos em que essa atividade ocorrerá, os quais são destacados em uma cor específica, facilitando a visualização das alocações. Ao clicar em “Salvar”, a nova atividade é adicionada ao plano, atualizando a carga horária do docente, bem como o cronograma semanal. Após submeter o plano, este tem a sua situação alterada para “Enviado”. Além disso, o botão “Submissão” se transforma em “Cancelar submissão”, e as ações de editar, deletar ou adicionar novas atividades são bloqueadas. O docente pode cancelar a submissão para fazer novas edições apenas enquanto o plano não for homologado.

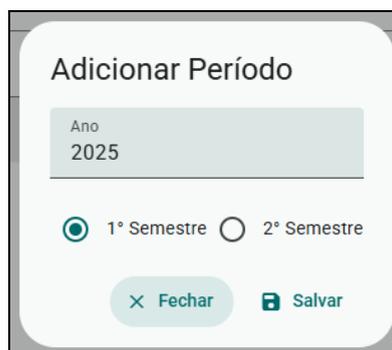
## 5.4. Acesso do administrador

Após realizar o login, o administrador é redirecionado para a tela da Figura 10, onde são listados os períodos (semestres) cadastrados, exibindo opções para visualizar os planos de cada período ou excluir um período existente (caso não haja planos vinculados), além de um botão para adicionar novos semestres. Dessa forma, o administrador gerencia todos os períodos disponíveis no sistema, preparando o ambiente para que os docentes registrem e submetam seus respectivos planos.



**Figura 10. Listagem de Períodos Letivos**

Ao clicar no botão “Adicionar Período”, a tela da Figura 11 é apresentada. Nela devem ser informados o ano e o respectivo semestre do período letivo. Após salvo, o período fica disponível para os docentes submeterem seus planos.



**Figura 11. Adicionar período**

Ao clicar em “Visualizar” (Figura 10), o administrador é redirecionado para a tela da Figura 12, nela o administrador visualiza os planos enviados pelos docentes no período selecionado, com a possibilidade de aplicar filtros por usuário e situação. A listagem exibe cada docente, sua carga horária total e a situação do plano, além de um botão para “Visualizar” o conteúdo de cada plano, permitindo ao administrador revisar as informações e tomar as ações necessárias (aprovar ou rejeitar).

Usuário	CH	Situação	Ações
[Redacted]	40	Enviado	Visualizar
[Redacted]	40	Enviado	Visualizar
[Redacted]	40	Enviado	Visualizar
[Redacted]	40	Enviado	Visualizar

**Figura 12. Planos do período**

Conforme apresentado na Figura 13, o administrador pode avaliar um plano e tomar a decisão de aprovar ou rejeitar. Ao clicar em “Rejeitar”, o administrador pode informar o motivo da recusa. Já ao selecionar “Aprovar”, o plano é automaticamente homologado e fica indisponível para novas edições, concluindo o processo de avaliação. A Figura 14 exibe um plano rejeitado que poderá ser submetido novamente pelo docente após as correções.

**Figura 13. Rejeitar plano**

Atividade	Tipo	CH	Ações
Algoritmos e Programação	Ensino	7	Editar Deletar
Banco de Dados II	Ensino	5	Editar Deletar

**Figura 14. Plano com mensagem de rejeição**

## 6. Considerações finais

Com o objetivo de atender às principais demandas dos cursos de graduação do Instituto Federal Farroupilha (Campus Frederico Westphalen), este trabalho apresentou a construção de um sistema web utilizando o framework Angular no front-end e o .NET no back-end, incluindo funcionalidades para o cadastro, gestão e homologação do Plano Individual Docente. O sistema engloba desde a criação e edição de atividades pelos docentes até a aprovação ou rejeição por parte dos administradores, buscando simplificar o processo de planejamento acadêmico.

Como contribuição, todo o código-fonte desenvolvido neste trabalho está disponível com uma licença livre em repositório público, permitindo que outros desenvolvedores colaborem com aprimoramentos ou implementem soluções similares. De acordo com o escopo inicial, a ferramenta foi implementada respeitando as necessidades levantadas e oferecendo as funcionalidades essenciais para o gerenciamento de planos docentes, de modo a atingir os objetivos propostos.

Como possibilidades de trabalhos futuros, sugere-se: (i) integrar o sistema com o SIGAA para evitar a duplicação de dados e de esforços no preenchimento das informações pelos docentes; (ii) implementar um cálculo automatizado de horas em conformidade com as diretrizes institucionais; (iii) incluir uma lista oficial das atividades da instituição, facilitando o cadastro ao permitir que o usuário selecione em vez de digitar manualmente; (iv) incluir uma tela de controle para os administradores especificarem quais usuários do domínio podem acessar o sistema.

## Referências

- FILHO, I. B.; AQUINO, G.; ROSA, J. G. S. SIGAA Mobile – O caso de sucesso da ferramenta de gestão acadêmica na era da computação móvel. In: Anais do XXIV Simpósio Brasileiro de Informática na Educação (SBIE 2013). Campinas, SP, Brasil: [s.n.], 2013. Disponível em: <<http://milanesa.ime.usp.br/rbie/index.php/sbie/article/view/2487>>. Acesso em: 5 set. 2024.
- GOOGLE. Introduction to the Angular docs. Disponível em: <<https://v17.angular.io/docs>>. Acesso em: 5 set. 2024a.
- GOOGLE. What is Angular?. Disponível em: <<https://v17.angular.io/guide/what-is-angular>>. Acesso em: 5 set. 2024b.
- MDN. JavaScript. Disponível em: <<https://developer.mozilla.org/pt-BR/docs/Web/JavaScript>>. Acesso em: 5 set. 2024a.
- MDN. HTML: Linguagem de Marcação de Hipertexto. Disponível em: <<https://developer.mozilla.org/pt-BR/docs/Web/HTML>>. Acesso em: 5 set. 2024b.
- MDN. CSS. Disponível em: <<https://developer.mozilla.org/pt-BR/docs/Web/CSS>>. Acesso em: 5 set. 2024c.

- MICROSOFT. Documentação do C#. Disponível em: <<https://learn.microsoft.com/pt-br/dotnet/csharp/tour-of-csharp>>. Acesso em: 5 set. 2024a.
- MICROSOFT. Introdução ao .NET. Disponível em: <<https://learn.microsoft.com/pt-br/dotnet/core/introduction>>. Acesso em: 5 set. 2024b.
- MICROSOFT. TypeScript is JavaScript with syntax for types. Disponível em: <<https://www.typescriptlang.org>>. Acesso em: 5 set. 2024c.
- NATALI, J. V.; CANDIDO, T. F. Sistema de Apoio ao Cadastro e Correção do PIT (Plano Individual de Trabalho). Colatina. 7, dez. 2023.
- POSTGRESQL. PostgreSQL: The World's Most Advanced Open Source Relational Database. Disponível em: <<https://www.postgresql.org>>. Acesso em: 5 set. 2024.
- ROCHA NETO, A. F.; LIMA, G. A. F. Turma virtual do SIGAA como ferramenta de apoio ao ensino. Natal: UFRN, 2009.
- SOMMERVILLE, Ian. Engenharia de Software. 9. ed. SP: Pearson, 2011.
- SOUZA, Maria Naires Alves de; MONTEIRO, André Jalles. Os docentes da Universidade Federal do Ceará e a utilização de alguns dos recursos do sistema integrado de gestão de atividades acadêmica (SIGAA). Ensaio: aval.pol.públ.Educ., Rio de Janeiro , v. 23, n. 88, p. 611-630, Sept. 2015 . Disponível em <<https://goo.gl/CsLmk4>>. Acesso em: 5 set. 2024.
- UFRN. Sistema Integrado de Gestão de Atividades Acadêmicas. Disponível em: <<https://sig.ifsc.edu.br/admin/public/loginunificado/index.jsf>> Acesso em: 5 set. 2024.