



UFOP

Universidade Federal
de Ouro Preto

**Universidade Federal de Ouro Preto
Instituto de Ciências Exatas e Aplicadas
Departamento de Computação e Sistemas**

GradeUFOP: Uma aplicação web para ajuste de grades curriculares

João Marcos Torres Gardingo

TRABALHO DE CONCLUSÃO DE CURSO

**ORIENTAÇÃO:
George Henrique Godim da Fonseca**

**Outubro, 2024
João Monlevade–MG**

João Marcos Torres Gardingo

GradeUFOP: Uma aplicação web para ajuste de grades curriculares

Orientador: George Henrique Godim da Fonseca

Monografia apresentada ao curso de Engenharia de Computação do Instituto de Ciências Exatas e Aplicadas, da Universidade Federal de Ouro Preto, como requisito parcial para aprovação na Disciplina “Trabalho de Conclusão de Curso II”.

Universidade Federal de Ouro Preto

João Monlevade

Outubro de 2024



FOLHA DE APROVAÇÃO

João Marcos Torres Gardingo

GradeUFOP: Uma aplicação web para ajuste de grades curriculares

Monografia apresentada ao Curso de Engenharia de Computação da Universidade Federal de Ouro Preto como requisito parcial para obtenção do título de Bacharel em Engenharia de Computação.

Aprovada 17 em de outubro de 2024

Membros da banca

Dr. George Henrique Godim da Fonseca - Orientador - Universidade Federal de Ouro Preto
Dr. Filipe Nunes Ribeiro - Universidade Federal de Ouro Preto
Dr. Samuel Souza Brito - Universidade Federal de Ouro Preto

George Henrique Godim da Fonseca, orientador do trabalho, aprovou a versão final e autorizou seu depósito na Biblioteca Digital de Trabalhos de Conclusão de Curso da UFOP em 19/10/2024



Documento assinado eletronicamente por **George Henrique Godim da Fonseca, PROFESSOR DE MAGISTERIO SUPERIOR**, em 19/10/2024, às 11:55, conforme horário oficial de Brasília, com fundamento no art. 6º, § 1º, do [Decreto nº 8.539, de 8 de outubro de 2015](#).



A autenticidade deste documento pode ser conferida no site http://sei.ufop.br/sei/controlador_externo.php?acao=documento_conferir&id_orgao_acesso_externo=0, informando o código verificador **0797902** e o código CRC **8F8483C8**.

Este trabalho é dedicado à memória da minha avó Laurice, aos meus pais João Batista e Sueli, minha irmã Nathane, minha noiva Victória, e a todos que me acompanharam e torceram por mim durante esse período.

Agradecimentos

Agradeço primeiramente a Deus, meu esteio durante toda essa jornada, por ter me concedido saúde e força para superar todos os desafios.

Aos meus pais, João Batista e Sueli, que não mediram esforços para que eu estudasse e realizasse meus sonhos. Obrigado também à minha irmã Nathane e à minha noiva, Victória. Vocês foram essenciais para que eu concluísse essa jornada. Obrigado por me apoiarem nas minhas decisões, pela compreensão com minhas ausências e pelo apoio constante durante todo esse processo.

A todos os meus familiares, que torceram e me apoiaram, em especial minhas avós Laurice e Margarida, que sempre me incentivaram e deram todo apoio para que eu concluísse o curso, principalmente nos momentos em que pensei em desistir.

Aos meus amigos, tanto os de longa data quanto os que fiz durante a graduação, que foram essenciais para a conclusão dessa jornada. Um agradecimento especial ao meu companheiro de casa, Estevão, que durante anos foi como uma família para mim, e ao meu amigo Matheus Lucas, por toda ajuda e companheirismo durante essa jornada no curso.

Ao meu orientador, Dr. George Henrique Godim da Fonseca, pela disposição em me ajudar e pela paciência na realização deste trabalho, mesmo em um momento de transição na sua carreira.

A todos os professores que fizeram parte dessa jornada e que, ao compartilharem seus conhecimentos, foram essenciais no meu aprendizado e formação.

*“Confia teus negócios ao Senhor
e teus planos terão bom êxito.”*

— Provérbios 16:3

Resumo

Este trabalho apresenta o desenvolvimento do GradeUFOP, uma aplicação web projetada para auxiliar os estudantes da Universidade Federal de Ouro Preto (UFOP) na montagem e otimização de suas grades curriculares. O sistema utiliza técnicas de raspagem de dados para coletar informações atualizadas sobre cursos, disciplinas e horários da universidade, integrando-as em uma plataforma de montagem de grades curriculares personalizadas. O GradeUFOP implementa um algoritmo de geração de grades que considera as preferências individuais dos alunos, restrições acadêmicas e disponibilidade de horários. A interface intuitiva permite aos usuários criar 'slots' de disciplinas, definir prioridades e visualizar múltiplas opções de grade geradas pelo sistema. A metodologia de desenvolvimento incluiu a criação de um robusto processo de raspagem de dados, implementação de uma API RESTful utilizando Next.js, e o desenvolvimento de uma interface de usuário. O sistema demonstrou resultados promissores na simplificação do processo de planejamento acadêmico. Os resultados obtidos indicam que o GradeUFOP oferece uma solução eficaz para o desafio de montagem de grades curriculares, potencialmente melhorando a experiência acadêmica dos estudantes da UFOP ao facilitar a tomada de decisões informadas sobre seu percurso acadêmico.

Palavras-chave: Planejamento acadêmico. Sistemas de recomendação. Raspagem de dados. Otimização de grade curricular. Aplicação web.

Abstract

This work presents the development of GradeUFOP, a web application designed to assist students at the Federal University of Ouro Preto (UFOP) in creating and optimizing their academic schedules. The system employs web scraping techniques to collect up-to-date information on university courses, subjects, and schedules, integrating them into a platform for personalized academic schedule assembly. GradeUFOP implements a schedule generation algorithm that considers individual student preferences, academic constraints, and time availability. The intuitive interface allows users to create course 'slots', set priorities, and visualize multiple schedule options generated by the system. The development methodology included creating a robust data scraping process, implementing a RESTful API using Next.js, and developing a responsive user interface. The system demonstrated promising results in simplifying the academic planning process. The results obtained indicate that GradeUFOP offers an effective solution to the challenge of academic schedule assembly, potentially improving the academic experience of UFOP students by facilitating informed decision-making about their academic journey.

Keywords: Academic planning. Recommendation systems. Web scraping. Course schedule optimization. Web application.

Lista de ilustrações

Figura 1 – Interface atual do sistema de matrícula da UFOP	16
Figura 2 – Fluxo do processo de raspagem de dados	25
Figura 3 – Modelo relacional do banco de dados	26
Figura 4 – Protótipos das telas de Onboarding	27
Figura 5 – Protótipos das telas de Montagem de Grades	28
Figura 6 – Tela de Login do GradeUFOP	34
Figura 7 – Tela de Onboarding - Introdução	34
Figura 8 – Tela de Onboarding - Seleção do Curso	35
Figura 9 – Tela de Onboarding - Seleção de Disciplinas Cursadas	35
Figura 10 – Tela de Onboarding - Confirmação dos Dados	36
Figura 11 – Tela de Montagem de Grades - Estado inicial	37
Figura 12 – Tela de Montagem de Grades - Menu de Filtros	37
Figura 13 – Tela de Montagem de Grades - Menu de Preferências	37
Figura 14 – Tela de Montagem de Grades - Adição de Disciplinas ao Slot	38
Figura 15 – Tela de Grades Geradas do GradeUFOP	38
Figura 16 – Tela de Perfil - Informações do Usuário	39
Figura 17 – Tela de Perfil - Grades Salvas	39

Lista de tabelas

Tabela 1 – Requisitos Funcionais do GradeUFOP	23
Tabela 2 – Requisitos Não Funcionais do GradeUFOP	24
Tabela 3 – Rotas principais da API do GradeUFOP	41

Lista de abreviaturas e siglas

UFOP Universidade Federal de Ouro Preto

PPC Projeto Pedagógico do Curso

API Application Programming Interface

PDF Portable Document Format

ORM Object-Relational Mapping

SSR Server-Side Rendering

SSG Static Site Generation

REST Representational State Transfer

VPS Virtual Private Server

ICEA Instituto de Ciências Exatas e Aplicadas

UX User Experience

UI User Interface

regexes expressões regulares

Sumário

1	INTRODUÇÃO	12
1.1	O problema de pesquisa	12
1.2	Objetivos	13
1.3	Metodologia	13
1.4	Organização do trabalho	14
2	FUNDAMENTAÇÃO TEÓRICA	15
2.1	Ajuste de matrícula em instituições de ensino superior	15
2.1.1	Desafios na seleção de disciplinas	15
2.1.2	Impacto na vida acadêmica dos estudantes	16
2.2	Sistemas de recomendação acadêmica	17
2.2.1	Conceitos básicos de sistemas de recomendação	17
2.2.2	Coleta de dados em sistemas de recomendação acadêmica	18
2.3	Tecnologias web modernas	18
2.3.1	Framework Next.js	18
2.3.2	Prisma ORM	19
2.3.3	Autenticação com Next-auth	19
2.4	Algoritmos de otimização para geração de grades individuais	20
2.4.1	Problema de seleção de disciplinas para alunos	20
2.5	Trabalhos Relacionados	20
2.5.1	Grafo de dependências de disciplinas da UFOP	20
2.5.2	Sistemas de geração e otimização de Grades Curriculares	21
2.5.3	Análise de trajetórias acadêmicas	21
3	METODOLOGIA	23
3.1	Especificações do Sistema	23
3.1.1	Requisitos Funcionais	23
3.1.2	Requisitos Não Funcionais	24
3.2	Coleta e processamento de dados	24
3.2.1	Algoritmo de raspagem	24
3.2.2	Ambiente de execução	25
3.3	Desenvolvimento da aplicação web	26
3.3.1	Prototipação da interface	26
3.3.2	Implementação	28
3.3.3	Funcionalidades Principais	30
3.4	Hospedagem e Configuração do Ambiente de Produção	31

3.4.1	Processo de Deploy na Vercel	31
3.4.2	Configuração do Banco de Dados PostgreSQL	31
3.4.3	Inicialização do Banco de Dados	32
3.4.4	Considerações sobre o Ambiente de Produção	32
4	RESULTADOS	33
4.1	Interface do Usuário	33
4.1.1	Tela de Login	33
4.1.2	Tela de Onboarding	34
4.1.2.1	Introdução	34
4.1.2.2	Seleção do Curso	34
4.1.2.3	Seleção de Disciplinas Cursadas	35
4.1.2.4	Confirmação dos Dados	36
4.1.3	Tela de Montagem de Grades	36
4.1.4	Tela de Grades Geradas	38
4.1.5	Tela de Perfil	39
4.2	Processo de Geração de Grades	40
4.3	Integração com Dados da UFOP	40
4.4	Visão Geral da API	40
4.5	Disponibilidade em Produção e Repositórios	41
5	CONSIDERAÇÕES FINAIS	42
5.1	Trabalhos Futuros	43
5.1.1	Aprimoramento da Experiência do Usuário	43
5.1.2	Aperfeiçoamento Técnico	43
5.1.3	Escalabilidade e Infraestrutura	44
5.1.4	Expansão do Escopo	44
	REFERÊNCIAS	45
	APÊNDICES	46
	APÊNDICE A – CASOS DE USO	47
A.1	UC01: Autenticação do Usuário	47
A.2	UC02: Cadastro de Curso e Disciplinas Cursadas	48
A.3	UC03: Geração de Grade Curricular	49
A.4	UC04: Visualização e Seleção de Grade	50
A.5	UC05: Atualização de Perfil	51

1 Introdução

A Universidade Federal de Ouro Preto (UFOP), reconhecida como uma das instituições de ensino superior mais prestigiadas do Brasil, oferece uma ampla variedade de disciplinas que abrangem diversas áreas acadêmicas. No entanto, a tarefa de montar uma grade curricular eficiente, que se alinhe aos interesses, habilidades e objetivos de carreira dos alunos, apresenta-se como um desafio complexo e muitas vezes frustrante para a comunidade acadêmica.

O processo atual de seleção de disciplinas, especialmente durante o período de ajuste de matrícula, revela-se particularmente problemático. Os alunos enfrentam dificuldades significativas ao tentar construir uma grade curricular que atenda às suas necessidades e preferências, sem conflitos de horário. Este desafio é especialmente evidente em cursos como Sistemas de Informação, onde a flexibilidade do Projeto Pedagógico do Curso (PPC) permite uma ampla escolha de disciplinas eletivas e optativas. (UFOP, 2024)

O método vigente exige que os estudantes testem as disciplinas individualmente para identificar possíveis conflitos de horário, um processo que consome tempo e gera frustração. Além disso, nem todas as disciplinas eletivas e facultativas estão disponíveis na plataforma de ajuste, obrigando os alunos a consultar tabelas de horários separadamente e navegar entre diferentes seções da plataforma. Este processo burocrático e a necessidade de verificação manual de cada combinação de disciplinas não apenas é ineficiente, mas também propenso a erros.

1.1 O problema de pesquisa

O problema de pesquisa central deste trabalho é a ineficiência e complexidade do processo de montagem de grades curriculares na UFOP. Especificamente, busca-se abordar:

- A dificuldade dos alunos em criar grades curriculares sem conflitos de horário;
- A falta de uma ferramenta integrada que considere as preferências individuais dos estudantes;
- A necessidade de otimizar o tempo gasto pelos alunos no processo de matrícula;
- A ausência de um sistema que facilite a visualização e comparação de diferentes opções de grade.

1.2 Objetivos

O presente trabalho consiste no desenvolvimento do GradeUFOP, um sistema web inovador projetado para otimizar e personalizar o processo de montagem de grades curriculares na UFOP.

Este trabalho possui os seguintes objetivos específicos:

- Desenvolver um sistema web capaz de gerar grades curriculares personalizadas e livres de conflitos;
- Implementar um algoritmo eficiente para a geração e otimização de grades curriculares;
- Criar uma interface intuitiva e amigável que simplifique o processo de seleção de disciplinas;
- Integrar dados atualizados sobre disciplinas, horários e pré-requisitos da UFOP;
- Validar a eficácia do sistema através de testes com dados reais da UFOP.

1.3 Metodologia

A metodologia adotada para este trabalho segue uma abordagem de desenvolvimento de software centrada no usuário, com as seguintes etapas:

- Revisão da literatura sobre sistemas de recomendação acadêmica e tecnologias web modernas;
- Desenvolvimento de um algoritmo de raspagem de dados para coletar informações atualizadas sobre disciplinas e horários da UFOP;
- Implementação de um banco de dados relacional para armazenar as informações coletadas;
- Desenvolvimento de protótipos de interface do usuário para guiar o design do sistema;
- Implementação do sistema web utilizando o framework Next.js e a [API RESTful](#);
- Desenvolvimento e otimização do algoritmo de geração de grades curriculares;
- Realização de testes de usabilidade e performance do sistema;
- Análise e discussão dos resultados obtidos.

1.4 Organização do trabalho

O trabalho está organizado em 6 capítulos. O Capítulo 2 apresenta a fundamentação teórica, abordando conceitos de sistemas de recomendação acadêmica e tecnologias web relevantes para o projeto. O Capítulo 3 detalha a metodologia de desenvolvimento, incluindo a arquitetura do sistema e o processo de implementação. O Capítulo 4 apresenta os resultados obtidos, incluindo as funcionalidades implementadas e os testes realizados. Capítulo 5 discute as implicações dos resultados, as limitações do sistema e as possibilidades de trabalhos futuros. Por fim, o Capítulo 6 apresenta as conclusões do trabalho e suas contribuições para a comunidade acadêmica da UFOP.

2 Fundamentação Teórica

Este capítulo apresenta os conceitos fundamentais e o contexto necessário para compreender o problema abordado pelo GradeUFOP, bem como as tecnologias e métodos utilizados em sua implementação. A compreensão destes elementos é crucial para assimilar a importância e a inovação trazida pelo sistema proposto.

2.1 Ajuste de matrícula em instituições de ensino superior

O ajuste de matrícula em instituições de ensino superior é um procedimento crítico que ocorre periodicamente, geralmente no início de cada semestre acadêmico. Este processo envolve a seleção e registro de disciplinas que os estudantes cursarão no período letivo seguinte. Embora seja um procedimento essencial para a organização do ano acadêmico, ele frequentemente apresenta desafios significativos tanto para os estudantes quanto para as instituições.

2.1.1 Desafios na seleção de disciplinas

A seleção de disciplinas é uma tarefa complexa que requer a consideração de múltiplos fatores. Os principais desafios enfrentados pelos estudantes incluem:

- **Conflitos de horário:** Os estudantes frequentemente se deparam com disciplinas desejadas ou necessárias que são oferecidas nos mesmos horários, forçando-os a fazer escolhas difíceis que podem impactar seu progresso acadêmico (PARANHOS, 2021);
- **Pré-requisitos:** Muitas disciplinas exigem a conclusão prévia de outras disciplinas. Os estudantes precisam navegar por uma rede complexa de pré-requisitos para garantir que estão elegíveis para as disciplinas desejadas;
- **Balanceamento de carga:** Os estudantes precisam equilibrar a carga de trabalho, considerando a dificuldade e a demanda de tempo de cada disciplina (SANTOS, 2017);
- **Disponibilidade limitada:** Algumas disciplinas podem ter vagas limitadas ou ser oferecidas apenas em semestres específicos, adicionando outra camada de complexidade ao processo de seleção (PARANHOS, 2021);
- **Navegação entre múltiplas opções:** Em cursos com grande flexibilidade curricular, como Sistemas de Informação na UFOP, os estudantes se deparam com uma vasta gama de opções de disciplinas eletivas e optativas.

No contexto específico da UFOP, estes desafios são exacerbados pela natureza do sistema atual. Os estudantes precisam testar manualmente cada combinação de disciplinas para verificar conflitos de horário, um processo tedioso e propenso a erros. Além disso, a plataforma de ajuste não inclui todas as disciplinas eletivas e optativas disponíveis, forçando os alunos a consultar tabelas de horários separadas e navegar entre diferentes seções da plataforma.

The screenshot shows the UFOP enrollment system interface. At the top, it displays the university name and the enrollment system. Below this, there are fields for the student's name, course (PIM - ENGENHARIA DE PRODUÇÃO), and semester (2). It also shows the weekly credit limit (32) and the total credits to be enrolled (32). There are links for 'Matr', 'Horário', 'Ajuste', and 'Histórico'. A section for selecting disciplines to be enrolled is visible, with checkboxes for different course types. A dropdown menu for 'Disciplina' and 'Turno' is present, with an error message: 'ERROR: Disciplina CEAS4 11 São Paulo com CEAS6 31'. Below this, there is a 'Atividade Simulado' section with a link to 'Disciplinas Matriculadas'. The main part of the interface is a table showing the class schedule for the selected course and semester. The table has columns for days of the week (Segunda, Terça, Quarta, Quinta, Sexta, Sábado) and rows for time slots from 07:30:00 to 21:50:00. The table shows the following schedule:

	Segunda	Terça	Quarta	Quinta	Sexta	Sábado
07:30:00						
08:20:00						
09:20:00						
10:10:00						
11:10:00						
12:00:00						
13:00:00	CEAS6-31T					
14:00:00	CEAS6-31T					
15:00:00				CEAS6-31T		
16:00:00				CEAS6-31T		
17:00:00						
18:00:00						
19:00:00		CEAS1-31T			ENP05-11T	
19:50:00		CEAS1-31T			ENP05-11T	
21:00:00			ENP05-11T		CEAS6-31T	
21:50:00			ENP05-11T		CEAS6-31P	

At the bottom of the table, there is a message: 'Para gerar sua Solicitação de Ajuste, clique no botão Gerar antes de sair da Aplicação.' Below this message are two buttons: 'Gerar' and 'Sair'.

Figura 1 – Interface atual do sistema de matrícula da UFOP

2.1.2 Impacto na vida acadêmica dos estudantes

O processo de matrícula, quando ineficiente, pode ter impactos significativos e de longo alcance na vida acadêmica dos estudantes:

- **Atrasos na conclusão do curso:** Quando os estudantes não conseguem se matricular nas disciplinas necessárias devido a conflitos de horário ou falta de vagas, isso pode levar a atrasos na conclusão do curso (PARANHOS, 2021);
- **Estresse e ansiedade:** O processo de tentar montar uma grade curricular viável pode ser extremamente estressante (SANTOS, 2017);
- **Subotimização do aprendizado:** Quando forçados a fazer escolhas baseadas principalmente em evitar conflitos de horário, ao invés de seus interesses e necessidades acadêmicas, os estudantes podem acabar em disciplinas que não são ideais para seu percurso acadêmico (GONÇALVES; SILVA; CORTES, 2018);
- **Desmotivação:** Repetidas experiências frustrantes com o processo de matrícula podem levar à desmotivação geral em relação ao curso e à instituição (PARANHOS, 2021).

De acordo com [Paranhos \(2021\)](#), no Instituto de Ciências Exatas e Aplicadas (ICEA) da UFOP, uma disciplina fundamental como Cálculo Diferencial e Integral 1 apresenta um índice de aprovação de apenas 40,59%, com uma média de notas de 37,11 em uma escala de 0 a 100. Este cenário tem um impacto direto e prolongado na trajetória acadêmica dos estudantes, pois uma reprovação em disciplinas básicas como Cálculo I, que é pré-requisito para diversas outras disciplinas, força o aluno a reorganizar completamente sua grade curricular nos semestres subsequentes. A partir deste momento, o estudante deixa de seguir a grade regular do curso e precisa fazer ajustes semestre a semestre, buscando a melhor combinação possível de disciplinas que se adeque tanto aos pré-requisitos quanto aos horários disponíveis. Esta necessidade constante de adaptação da grade curricular torna essencial a existência de ferramentas que auxiliem os estudantes a visualizar e explorar diferentes combinações de disciplinas de forma eficiente, permitindo que tomem decisões mais informadas sobre sua trajetória acadêmica.

2.2 Sistemas de recomendação acadêmica

Os sistemas de recomendação acadêmica emergiram como uma solução promissora para muitos dos desafios associados ao processo de seleção de disciplinas. Estes sistemas aplicam técnicas de inteligência artificial e análise de dados para auxiliar os estudantes na tomada de decisões informadas sobre suas escolhas acadêmicas.

2.2.1 Conceitos básicos de sistemas de recomendação

Os sistemas de recomendação são ferramentas de software que buscam prever as preferências ou relevância de itens para um usuário. No contexto acadêmico, estes 'itens' são tipicamente disciplinas ou caminhos curriculares ([CAZELLA; NUNES; REATEGUI, 2010](#)). Existem várias abordagens para implementar sistemas de recomendação, sendo as principais:

- **Filtragem colaborativa:** Esta abordagem faz recomendações baseadas em padrões de comportamento semelhantes entre usuários. Por exemplo, se muitos estudantes que cursaram e gostaram da disciplina A também gostaram da disciplina B, o sistema pode recomendar a disciplina B para um novo estudante que mostrou interesse em A ([REATEGUI; CAZELLA, 2007](#));
- **Filtragem baseada em conteúdo:** Neste método, as recomendações são feitas com base nas características dos itens e no perfil do usuário. No contexto acadêmico, isso pode envolver recomendar disciplinas com conteúdo ou habilidades similares às aquelas que o estudante já demonstrou interesse ou sucesso ([FERREIRA; RAABE, 2013](#));

- **Abordagens híbridas:** Combinam elementos de filtragem colaborativa e baseada em conteúdo para produzir recomendações mais robustas e precisas (LOPES; RIBEIRO; MACIEL, 2019).

No caso do GradeUFOP, o sistema implementa uma abordagem que pode ser considerada uma variante da filtragem baseada em conteúdo, onde as 'características dos itens' são os horários das disciplinas, pré-requisitos, e outras restrições acadêmicas, e o 'perfil do usuário' é definido pelas escolhas e preferências expressas pelo estudante.

2.2.2 Coleta de dados em sistemas de recomendação acadêmica

Em sistemas de recomendação acadêmica, a qualidade e atualidade dos dados são fundamentais para o funcionamento eficaz do sistema. Em muitos casos, especialmente quando não há APIs disponíveis, a raspagem web torna-se uma técnica crucial para obter informações atualizadas sobre cursos, disciplinas e horários. Esta técnica permite a coleta automática de dados de fontes online, como sites institucionais e sistemas de gerenciamento acadêmico, garantindo que o sistema de recomendação opere com informações precisas e atuais.

2.3 Tecnologias web modernas

O GradeUFOP foi desenvolvido utilizando tecnologias web modernas que permitem a criação de aplicações robustas, escaláveis e de alta performance. Esta seção detalha as principais tecnologias utilizadas no projeto.

2.3.1 Framework Next.js

O Next.js¹ é um framework React de código aberto criado pela Vercel, que permite a construção de aplicações web modernas com renderização do lado do servidor (SSR) e geração de sites estáticos (SSG). Algumas das principais vantagens do Next.js que justificaram sua escolha para o GradeUFOP incluem:

- **Renderização híbrida:** Permite combinar SSR, SSG e renderização do lado do cliente, otimizando o desempenho e a experiência do usuário;
- **Roteamento automático:** Simplifica a estrutura do projeto e facilita a navegação entre páginas;
- **Otimização de desempenho:** Oferece recursos como divisão de código automática e pré-busca de recursos;

¹ <https://nextjs.org/docs>

- **API Routes:** Permite a criação de APIs *serverless* diretamente no projeto, facilitando a integração entre *front-end* e *back-end*.

No contexto do GradeUFOP, o Next.js foi utilizado para criar uma interface de usuário responsiva e interativa, além de facilitar a implementação da API necessária para a geração de grades curriculares.

2.3.2 Prisma ORM

O Prisma² é um **ORM** (*Object-Relational Mapping*) moderno para Node.js e TypeScript. Ele foi escolhido para o GradeUFOP devido às seguintes características:

- **Type-safety:** Fornece tipos gerados automaticamente, reduzindo erros em tempo de execução;
- **Migrações automáticas:** Simplifica a evolução do esquema do banco de dados;
- **Query builder intuitivo:** Permite escrever consultas complexas de forma mais legível e manutenível;

No GradeUFOP, o Prisma é utilizado para modelar e interagir com o banco de dados, facilitando a manipulação de informações sobre cursos, disciplinas e alunos.

2.3.3 Autenticação com Next-auth

Next-auth³ é uma solução de autenticação completa e flexível para aplicações Next.js. Suas principais vantagens incluem:

- **Múltiplos provedores:** Suporta autenticação via redes sociais, e-mail/senha, e outros métodos;
- **Segurança:** Implementa as melhores práticas de segurança, como JWT e sessões criptografadas;
- **Fácil integração:** Se integra perfeitamente com o ecossistema Next.js.

No GradeUFOP, o Next-auth é utilizado para garantir que apenas usuários autenticados com e-mail institucional (@ufop) possam acessar o sistema, proporcionando uma camada adicional de segurança e privacidade.

² <https://www.prisma.io/>

³ <https://next-auth.js.org/>

2.4 Algoritmos de otimização para geração de grades individuais

O coração do GradeUFOP é seu algoritmo de geração de grades curriculares individuais, que utiliza técnicas de otimização combinatória para criar soluções viáveis e eficientes para os alunos, baseando-se nos horários e disciplinas já definidos pela universidade.

2.4.1 Problema de seleção de disciplinas para alunos

O problema de seleção de disciplinas para alunos em instituições de ensino superior é um desafio complexo, embora diferente do problema clássico de programação de horários institucionais. No contexto do GradeUFOP, o foco está em auxiliar os alunos a criar grades curriculares personalizadas que atendam às suas necessidades e preferências, utilizando os horários e disciplinas já estabelecidos pela universidade.

As principais considerações incluem:

- **Restrições Rígidas:** Conflitos de horário entre disciplinas escolhidas, pré-requisitos acadêmicos;
- **Restrições Flexíveis:** Preferências dos alunos quanto a dias e horários, balanceamento da carga horária, minimização de 'janelas' entre aulas.

2.5 Trabalhos Relacionados

O GradeUFOP se insere em um contexto de pesquisas e desenvolvimentos voltados para a otimização e visualização de percursos acadêmicos. Diversos trabalhos têm abordado desafios similares, cada um com um foco específico. Esta seção apresenta alguns desses trabalhos, destacando suas contribuições e como o GradeUFOP se relaciona com eles.

2.5.1 Grafo de dependências de disciplinas da UFOP

Um trabalho particularmente relevante no contexto da UFOP é o projeto desenvolvido por [Costa \(2021\)](#), que criou um sistema para visualização de grafos de dependências (pré-requisitos) das disciplinas da UFOP. Este projeto oferece uma representação visual das relações entre as disciplinas, permitindo que os estudantes compreendam melhor a estrutura curricular de seus cursos.

Costa enfrentou desafios similares aos do GradeUFOP no que diz respeito à obtenção de dados. Assim como no presente trabalho, Costa teve que desenvolver um processo de raspagem de dados para coletar informações sobre as disciplinas e suas relações de pré-requisitos, uma vez que a UFOP não disponibiliza uma API oficial para acesso a esses dados. A experiência de Costa forneceu insights valiosos para o processo de raspagem.

O trabalho de Costa se concentra na visualização das relações entre disciplinas, transformando os dados coletados em um grafo interativo que permite aos estudantes explorar visualmente a estrutura de pré-requisitos de seus cursos. Por outro lado, o GradeUFOP expande esse conceito, incorporando essas relações de pré-requisitos em um sistema mais amplo de planejamento e recomendação de grades curriculares.

2.5.2 Sistemas de geração e otimização de Grades Curriculares

Ferreira e Raabe (2013) desenvolveram um sistema de recomendação de disciplinas para cursos de graduação utilizando técnicas de filtragem colaborativa. O sistema analisa o histórico acadêmico dos estudantes, incluindo notas, sequência de matrículas e taxas de aprovação, para identificar padrões de comportamento. Com base nesses dados, o sistema utiliza algoritmos de similaridade para encontrar alunos com perfis acadêmicos semelhantes e, a partir disso, gerar recomendações personalizadas de disciplinas optativas. Os resultados do estudo demonstraram uma taxa de acerto de 75% nas recomendações, indicando a eficácia da abordagem para auxiliar estudantes na escolha de disciplinas optativas.

O GradeUFOP, por outro lado, adota uma abordagem diferente. Em vez de recomendar disciplinas, o sistema gera combinações completas de grades curriculares com base nas escolhas explícitas e preferências dos estudantes. O foco está em otimizar essas combinações considerando restrições de horários e outras preferências pessoais especificadas pelo usuário. Esta abordagem coloca o poder de decisão inteiramente nas mãos do estudante, fornecendo-lhe as ferramentas para explorar diferentes cenários de grade curricular.

2.5.3 Análise de trajetórias acadêmicas

Silva et al. (2019) desenvolveram uma ferramenta robusta para análise de trajetórias acadêmicas, focando na identificação de padrões de desempenho e na previsão de risco de evasão. O sistema utiliza técnicas de mineração de dados e aprendizado de máquina para processar históricos acadêmicos completos, incluindo dados como frequência, notas e tempo de permanência em cada disciplina. A ferramenta identifica padrões críticos que podem indicar risco de evasão, como reprovações consecutivas em disciplinas específicas ou queda consistente no desempenho ao longo do tempo. Os resultados do estudo mostraram uma precisão de 82% na identificação precoce de estudantes em risco de evasão, permitindo intervenções preventivas mais eficazes.

Embora o foco principal do GradeUFOP não seja a análise de desempenho, a incorporação de elementos de análise de trajetória poderia ser uma extensão valiosa para o sistema no futuro. A integração dessas análises permitiria não apenas auxiliar os estudantes a compreender melhor o impacto de suas escolhas de disciplinas no longo prazo, mas também fornecer insights sobre padrões de sucesso acadêmico baseados em dados históricos.

O GradeUFOP se distingue destes trabalhos ao focar na geração e otimização de grades curriculares com base em preferências explícitas dos usuários. Diferentemente de sistemas de recomendação automática ou análise preditiva, o GradeUFOP capacita os estudantes a explorarem ativamente diferentes combinações de disciplinas, respeitando as restrições acadêmicas e suas preferências pessoais. A abordagem do sistema enfatiza a autonomia do estudante no planejamento acadêmico, fornecendo ferramentas e informações necessárias para tomadas de decisão mais conscientes. Ao integrar dados atualizados da instituição e oferecer uma interface intuitiva para a geração de grades curriculares, o GradeUFOP busca preencher uma lacuna importante no suporte ao planejamento acadêmico dos estudantes.

3 Metodologia

Este capítulo detalha a metodologia empregada no desenvolvimento do GradeUFOP, abrangendo desde a coleta de dados até a implementação e testes do sistema. O processo de desenvolvimento seguiu uma abordagem iterativa e incremental, permitindo ajustes e melhorias contínuas ao longo do projeto.

3.1 Especificações do Sistema

Esta seção apresenta um resumo das especificações do GradeUFOP. Os requisitos funcionais e não funcionais são detalhados abaixo, enquanto os casos de uso completos podem ser encontrados no Apêndice A.

3.1.1 Requisitos Funcionais

Tabela 1 – Requisitos Funcionais do GradeUFOP

Requisito	Descrição
RF01	O sistema deve permitir que os usuários se autentiquem usando seu e-mail institucional da UFOP.
RF02	O sistema deve permitir que os usuários selecionem seu curso atual.
RF03	O sistema deve permitir que os usuários marquem as disciplinas já cursadas.
RF04	O sistema deve permitir a importação de dados do "Faltam Cursar" do portal da UFOP.
RF05	O sistema deve gerar combinações de grades curriculares baseadas nas preferências do usuário.
RF06	O sistema deve permitir que os usuários filtrem disciplinas por pré-requisitos, tipo (obrigatórias/eletivas) e horários.
RF07	O sistema deve permitir que os usuários criem e ordenem slots de disciplinas para gerar combinações de grades.
RF08	O sistema deve exibir as grades geradas em um formato de tabela semanal.
RF09	O sistema deve permitir que os usuários editem suas informações de perfil e preferências.
RF10	O sistema deve ter a capacidade de atualizar os dados das disciplinas e cursos a cada semestre.

3.1.2 Requisitos Não Funcionais

Tabela 2 – Requisitos Não Funcionais do GradeUFOP

Requisito	Descrição
RNF01	Desempenho: O sistema deve gerar combinações de grades em menos de 5 segundos.
RNF02	Usabilidade: A interface do usuário deve ser intuitiva e fácil de usar, exigindo no máximo 3 cliques para acessar qualquer funcionalidade.
RNF03	Segurança: O sistema deve criptografar todos os dados sensíveis dos usuários.
RNF04	Compatibilidade: O sistema deve funcionar nos principais navegadores web (Chrome, Firefox, Safari, Edge).

O GradeUFOP foi projetado para atender às necessidades específicas dos estudantes da UFOP, permitindo uma gestão eficiente de suas grades curriculares. Os principais requisitos incluem autenticação segura, seleção de cursos, marcação de disciplinas cursadas, geração de grades curriculares personalizadas e uma interface intuitiva para visualização e seleção de grades. Os casos de uso abrangem as principais interações dos usuários com o sistema, desde o login até a geração e seleção de grades curriculares. Estes casos de uso foram elaborados para garantir que o sistema atenda de forma eficaz às necessidades dos estudantes durante todo o processo de planejamento acadêmico.

3.2 Coleta e processamento de dados

A primeira etapa crucial do projeto foi a obtenção dos dados necessários da UFOP. Dada a ausência de uma API oficial, foi necessário desenvolver um algoritmo robusto de raspagem de dados.

3.2.1 Algoritmo de raspagem

O algoritmo de raspagem foi desenvolvido em Python¹, utilizando bibliotecas como Selenium, BeautifulSoup, e tabula-py. Esta última foi fundamental para a extração de dados de arquivos PDF, um desafio significativo no processo de coleta de informações.

Para melhor visualização do processo completo de raspagem de dados, a Figura 2 apresenta um diagrama que ilustra o fluxo de execução do algoritmo, desde o acesso inicial ao site da UFOP até o armazenamento final dos dados processados no banco de dados.

¹ <https://www.python.org/>

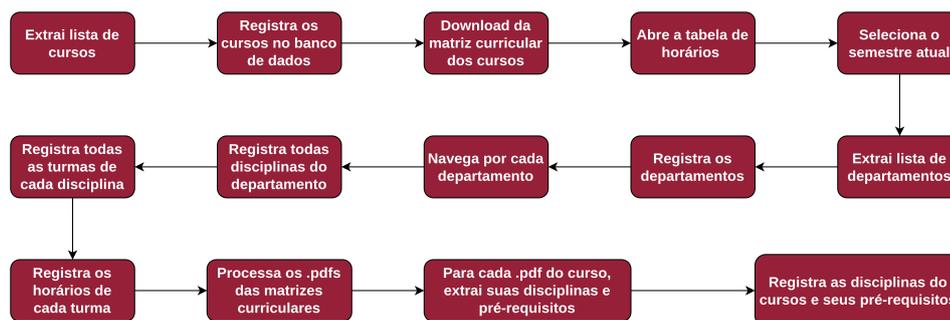


Figura 2 – Fluxo do processo de raspagem de dados

Um dos principais desafios enfrentados foi a análise dos cabeçalhos das colunas para identificar corretamente o tipo de informação contida. Para isso, foram utilizadas expressões regulares (**regexes**) para reconhecer padrões específicos, como códigos de disciplinas.

O processamento dos pré-requisitos exigiu atenção especial, pois estes poderiam estar distribuídos em múltiplas linhas ou conter informações adicionais. Novamente, o uso de regex foi crucial para extrair corretamente essas informações.

Outro desafio significativo foi lidar com disciplinas equivalentes e disciplinas de ainda não registradas na tabela de horários. Para resolver isso, foi implementado um sistema de equivalência que permite identificar disciplinas com códigos diferentes, mas com nomes idênticos, e criar novas entradas no banco de dados quando necessário.

O algoritmo também precisou lidar com quebras de linha inesperadas e conciliar dados distribuídos em múltiplas linhas, garantindo a integridade das informações coletadas.

3.2.2 Ambiente de execução

O algoritmo de raspagem é executado em um ambiente Docker², facilitando a portabilidade e a consistência entre diferentes ambientes de desenvolvimento. O container Docker inclui:

- Uma imagem PostgreSQL³ para armazenamento dos dados.
- Um script de inicialização que configura a estrutura do banco de dados.
- O ambiente Python necessário para a execução do algoritmo de raspagem.

Esta abordagem garante que o processo de coleta e armazenamento de dados seja consistente e reproduzível.

² <https://www.docker.com/>

³ <https://www.postgresql.org/>

Para armazenar todos os dados coletados de forma estruturada, foi desenvolvido um modelo relacional abrangente, ilustrado na Figura 3. Este modelo contempla todas as entidades necessárias para o sistema, incluindo cursos, disciplinas, departamentos e suas respectivas relações, garantindo a integridade e consistência dos dados.

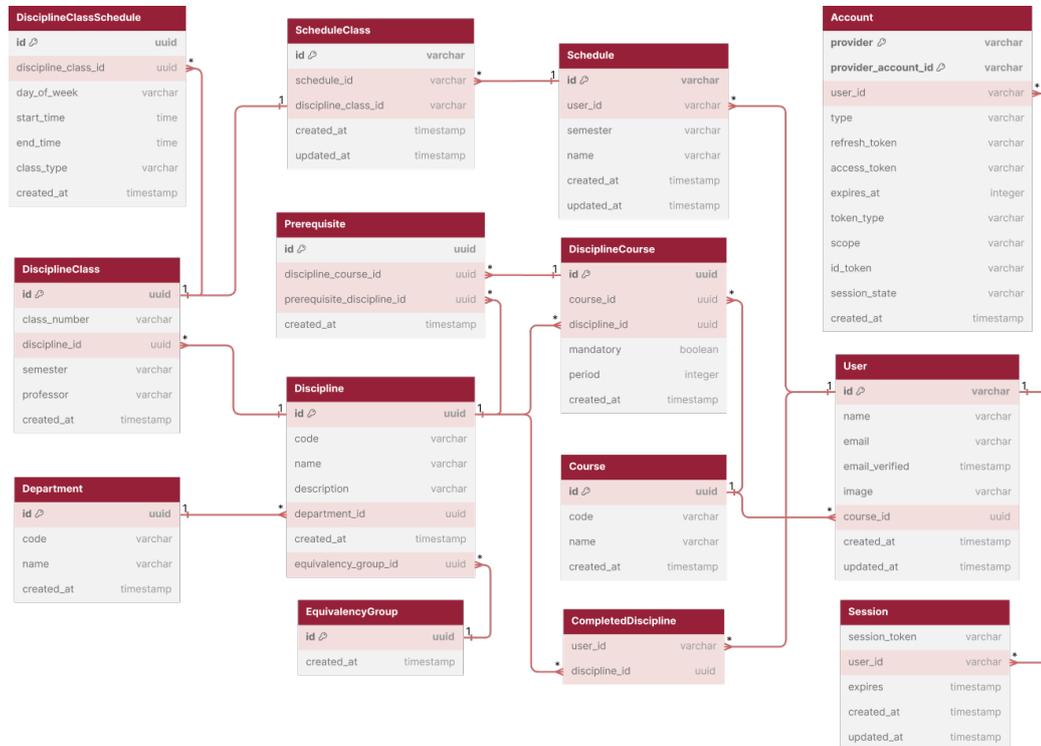


Figura 3 – Modelo relacional do banco de dados

3.3 Desenvolvimento da aplicação web

O desenvolvimento da aplicação web do GradeUFOP foi realizado de forma iterativa, com várias versões sendo produzidas até chegar à versão final. O processo incluiu as seguintes etapas:

3.3.1 Prototipação da interface

Inicialmente, foram criados protótipos de baixa fidelidade das principais telas da aplicação. Estes protótipos evoluíram ao longo do desenvolvimento, refletindo as mudanças nos requisitos e no feedback recebido.

É importante pontuar que, embora o foco inicial tenha sido na versão desktop, o aplicativo foi desenvolvido para funcionar em resoluções menores, incluindo dispositivos móveis. No entanto, não foi criada uma versão responsiva específica para esses dispositivos. As telas iniciais prototipadas incluem:

- Telas de Onboarding: Desenvolvida para permitir ao usuário preencher suas informações como curso e disciplinas já cursadas.
- Telas de Montagem de grades: Desenvolvida com a finalidade de permitir o usuário escolher suas disciplinas desejadas entre as disponíveis e gerar combinações de grades curriculares.

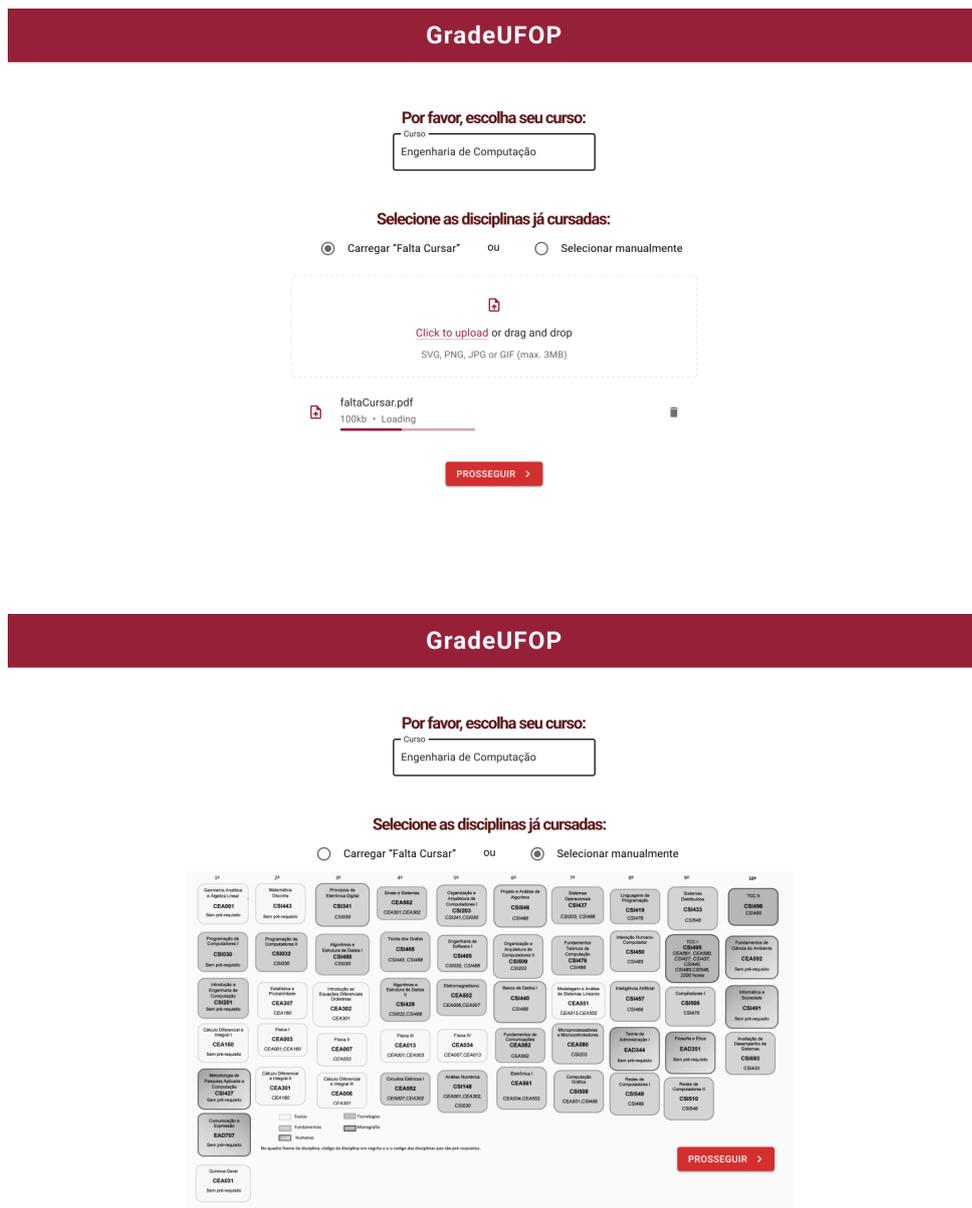


Figura 4 – Protótipos das telas de Onboarding

GradeUFOP

Por favor, escolha as disciplinas que deseja cursar:

FILTRAR ▼

Disciplina
×
+

⋮

CEA051 - Cálculo II - T31 Rafael Thebaldi ×

CEA051 - Cálculo II - T32 Herson ×

Disciplina
×
+

⋮

Discreta ×

CSI443 - Matemática Discreta - T21 ×

CSI443 - Matemática Discreta - T22 ×

CEA341 - Principios de Elettronica Digital - T21 Joao Oliveira ×

ADICIONAR SLOT
GERAR GRADES

GradeUFOP

Disciplinas Escolhidas

Código	Turma	Disciplina	Professor
CEA582	22	Fundamentos de Comunicações	Mylena Cruzinha da Silva
CSI419	11	Linguagens de Programação	Luiz Carlos Bambirra Torres
CSI509	11	Organização e Arquitetura de Computadores II	Harlei Miguel de Arruda Leite
CSI705	11	Aprendizagem Profunda	Eduardo da Silva Ribeiro
CSI802	11	Gestão da Tecnologia da Informação	Lucineia Souza Maia

< ANTERIOR
PRÓXIMA >

Figura 5 – Protótipos das telas de Montagem de Grades

3.3.2 Implementação

A implementação da aplicação web foi realizada utilizando o framework Next.js, uma escolha estratégica devido à sua capacidade de renderização híbrida, roteamento eficiente e integração entre frontend e backend. O processo de implementação foi dinâmico e iterativo, adaptando-se às necessidades emergentes do projeto. As principais etapas e características do desenvolvimento incluíram:

1. **Desenvolvimento do backend e API RESTful:** O Next.js, com sua funcionalidade de API Routes, permitiu a criação de uma API RESTful robusta e integrada diretamente no projeto. Isso simplificou significativamente o processo de desenvolvimento, eliminando a necessidade de um servidor backend separado. As API Routes

facilitaram a criação de endpoints para todas as funcionalidades necessárias, como gerenciamento de usuários, cursos, disciplinas e geração de grades;

2. **Implementação do frontend:** Inicialmente baseado nos protótipos desenvolvidos, o frontend evoluiu constantemente para atender às novas necessidades identificadas durante o desenvolvimento. Novas telas foram adicionadas, como a tela de perfil do usuário, e funcionalidades existentes foram expandidas, como a tela de onboarding e a interface de montagem de grades. O uso do React⁴ para desenvolvimento de componentes permitiu uma estrutura modular e facilmente adaptável;
3. **Integração com banco de dados:** A integração com o banco de dados PostgreSQL foi realizada através do Prisma ORM. O uso do TypeScript⁵ em conjunto com o Prisma proporcionou uma experiência de desenvolvimento mais segura e produtiva. A inferência de tipos do Prisma permitiu uma integração de ponta a ponta mais robusta, reduzindo erros e facilitando o desenvolvimento de queries complexas;
4. **Algoritmo de geração de grades:** O desenvolvimento do algoritmo de geração e otimização de grades curriculares foi um processo iterativo e central para o funcionamento do GradeUFOP. Este algoritmo utiliza uma abordagem combinatória para criar grades curriculares personalizadas com base nas escolhas e preferências explícitas dos usuários.

O processo de geração de grades funciona da seguinte forma:

- **Criação de slots:** Os usuários criam 'slots' que representam espaços em sua grade curricular. Cada slot pode conter múltiplas opções de disciplinas;
- **Ordenação de prioridades:** Os slots são ordenados por prioridade, permitindo que os usuários indiquem quais disciplinas são mais importantes para eles;
- **Geração de combinações:** O algoritmo percorre os slots ordenados, gerando todas as combinações possíveis de disciplinas que respeitam as restrições de horário;
- **Verificação de conflitos:** Durante o processo de geração, o algoritmo verifica constantemente por conflitos de horário, descartando combinações inválidas;
- **Aplicação de filtros:** São aplicados pesos adicionais baseados nas preferências do usuário, como minimização de 'janelas' entre aulas e distribuição equilibrada ao longo da semana;
- **Ranqueamento de soluções:** As combinações geradas são então ranqueadas de acordo com o grau de atendimento às preferências do usuário. A função de pontuação (score) é calculada considerando duas métricas principais:

⁴ <https://react.dev/>

⁵ <https://www.typescriptlang.org/>

$$Score_{total} = (N_{dias} \times P_{dias}) + (N_{janelas} \times P_{janelas}) \quad (3.1)$$

Onde:

- N_{dias} é o número de dias letivos com aulas na semana;
- P_{dias} é o peso definido pelo usuário para distribuição das aulas;
- $N_{janelas}$ é o número total de horas vagas entre aulas no mesmo dia;
- $P_{janelas}$ é o peso definido pelo usuário para minimização de janelas.

O algoritmo ordena as combinações primeiro pelo número de disciplinas incluídas (maximizando) e depois pelo score total (minimizando), buscando assim grades que incluam o máximo de disciplinas possível com a melhor distribuição de horários conforme as preferências do usuário.

O algoritmo foi projetado para ser capaz de gerar e avaliar um grande número de combinações em um curto espaço de tempo, permitindo que os usuários explorem diversas possibilidades de grade curricular rapidamente. É importante ressaltar que durante a fase de desenvolvimento, os testes foram focados nos cursos oferecidos no campus do [ICEA](#), permitindo uma validação mais controlada do sistema antes de sua expansão para os demais campi da UFOP;

5. **Adaptação contínua:** Ao longo do desenvolvimento, novas necessidades foram identificadas, levando à inclusão de funcionalidades adicionais e à reestruturação de telas existentes. Por exemplo, a tela de onboarding foi alterada para um fluxo de passo a passo, contendo informações adicionais, e a interface de montagem de grades foi aprimorada com filtros avançados e visualizações mais intuitivas.

3.3.3 Funcionalidades Principais

- **Onboarding de Usuários:** Permite que novos usuários cadastrem seu curso e disciplinas já cursadas;
- **Filtros Avançados:** Possibilita aos usuários filtrarem disciplinas por pré-requisitos, tipo (obrigatórias/eletivas) e horários;
- **Sistema de Slots:** Permite a criação e ordenação de slots de disciplinas para gerar combinações de grades;
- **Geração de Grades:** Algoritmo que gera múltiplas opções de grades curriculares com base nas preferências do usuário;
- **Visualização de Resultados:** Exibe as grades geradas em um formato de tabela semanal, facilitando a análise pelo usuário.

3.4 Hospedagem e Configuração do Ambiente de Produção

A etapa final do desenvolvimento do GradeUFOP envolveu a hospedagem da aplicação e a configuração do banco de dados em um ambiente de produção. Para isso, optamos por utilizar os serviços oferecidos pela Vercel⁶, uma plataforma especializada em hospedagem de aplicações web modernas, particularmente adequada para projetos desenvolvidos com Next.js.

3.4.1 Processo de Deploy na Vercel

A Vercel oferece uma integração perfeita com projetos Next.js, tornando o processo de deploy extremamente simples e eficiente. As principais vantagens incluem:

- **Integração com Git:** A Vercel se integra diretamente com repositórios Git, permitindo deploys automáticos a cada push;
- **Configuração Zero:** Para aplicações Next.js padrão, nenhuma configuração adicional é necessária;
- **Previews de Deploy:** Cada pull request gera um preview da aplicação, facilitando a revisão de mudanças;
- **Escalabilidade Automática:** A infraestrutura da Vercel lida automaticamente com picos de tráfego.

O processo de deploy do GradeUFOP na Vercel foi realizado em poucos passos:

1. Conexão do repositório GitHub à conta Vercel;
2. Seleção do projeto GradeUFOP;
3. Configuração das variáveis de ambiente necessárias;
4. Iniciação do deploy automático.

3.4.2 Configuração do Banco de Dados PostgreSQL

Além da hospedagem da aplicação, a Vercel oferece um serviço de banco de dados PostgreSQL integrado, que foi utilizado para armazenar os dados raspados da UFOP. As vantagens deste serviço incluem:

⁶ <https://vercel.com/>

- **Integração Simplificada:** O banco de dados é facilmente conectado à aplicação Next.js;
- **Plano Gratuito:** Suficiente para as necessidades iniciais do projeto;
- **Backups Automáticos:** Garantindo a segurança dos dados.

A configuração do banco de dados envolveu:

1. Criação de uma instância PostgreSQL no painel da Vercel;
2. Obtenção das credenciais de conexão;
3. Configuração das variáveis de ambiente no projeto Next.js.

3.4.3 Inicialização do Banco de Dados

Para inicializar o banco de dados com as tabelas necessárias, foi necessário executar um script SQL diretamente no PostgreSQL da Vercel. O processo foi o seguinte:

1. Adaptação do script de inicialização SQL usado no ambiente de desenvolvimento do algoritmo de raspagem;
2. Acesso ao console SQL do PostgreSQL na Vercel;
3. Execução do script para criar as tabelas e estruturas necessárias.

Este processo permitiu que o banco de dados estivesse pronto para receber os dados raspados da UFOP assim que a aplicação fosse implantada.

3.4.4 Considerações sobre o Ambiente de Produção

A utilização da Vercel para hospedagem e banco de dados proporcionou uma solução completa e sem custos para o deployment do GradeUFOP. A facilidade de configuração e a integração perfeita com Next.js permitiram que o foco permanecesse no desenvolvimento das funcionalidades, enquanto a infraestrutura era gerenciada de forma eficiente pela plataforma. Esta abordagem não apenas simplificou o processo de implantação, mas também garantiu um ambiente de produção robusto e escalável, capaz de atender às demandas do sistema à medida que ele cresce e evolui.

4 Resultados

Este capítulo apresenta os resultados obtidos no desenvolvimento do GradeUFOP, incluindo uma descrição detalhada das interfaces do usuário, uma visão geral da API implementada e uma análise de cada componente do sistema.

4.1 Interface do Usuário

Antes de detalhar os resultados obtidos, é importante ressaltar que a interface do usuário do GradeUFOP teve um enfoque maior na experiência de usuário (**UX**) e interface de usuário (**UI**) para a versão desktop. Esta decisão foi tomada considerando que a maioria dos usuários provavelmente acessaria o sistema através de computadores para realizar o planejamento detalhado de suas grades curriculares. No entanto, é importante notar que o aplicativo também funciona em dispositivos móveis, garantindo acessibilidade em diferentes plataformas.

A interface foi desenvolvida com foco na usabilidade e intuitividade, seguindo os requisitos estabelecidos. As principais telas da aplicação são apresentadas a seguir:

4.1.1 Tela de Login

A tela de login é uma interface simples e direta. Consiste em uma mensagem de boas-vindas e um botão para conectar com o Google, que quando clicado, inicia o fluxo de login do Google. Após a autenticação bem-sucedida com um e-mail institucional da UFOP, o usuário é redirecionado para a tela de onboarding. Durante este processo, se o usuário ainda não estiver registrado no sistema, sua conta é automaticamente criada.

Em caso de erro na autenticação, uma tela auxiliar para erros de autenticação é exibida, dando mais detalhes sobre o erro ocorrido e contendo um botão para se autenticar novamente.

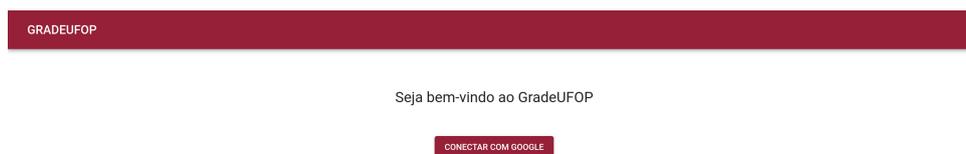


Figura 6 – Tela de Login do GradeUFOP

4.1.2 Tela de Onboarding

A tela de onboarding é dividida em etapas, utilizando um componente de *stepper* no topo para guiar o usuário. Esta tela tem como objetivo coletar informações essenciais do usuário e introduzi-lo ao sistema. As etapas são:

4.1.2.1 Introdução

Uma mensagem de boas-vindas que contextualiza as informações necessárias.

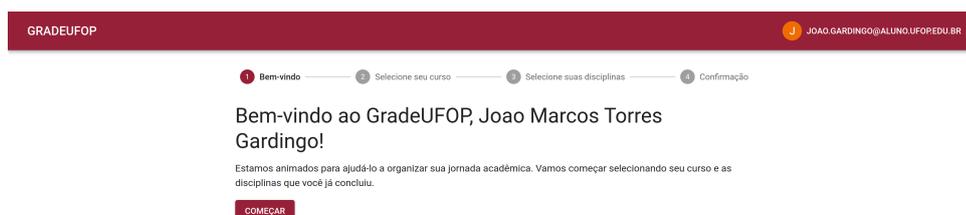


Figura 7 – Tela de Onboarding - Introdução

4.1.2.2 Seleção do Curso

Um campo de entrada autocompletável com a lista de todos os cursos da UFOP, onde o usuário informa seu curso.

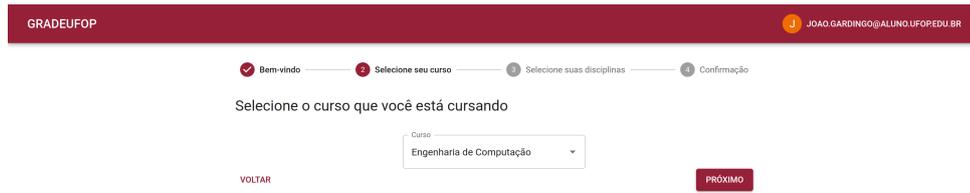


Figura 8 – Tela de Onboarding - Seleção do Curso

4.1.2.3 Seleção de Disciplinas Cursadas

Esta etapa permite ao usuário indicar quais disciplinas já foram cursadas. A interface inclui:

- Uma tabela onde cada coluna representa um período do curso;
- Opção de seleção manual das disciplinas já cursadas;
- Funcionalidade de upload do arquivo 'faltaCursar' do sistema MinhaUFOP;
- Processamento automático das disciplinas cursadas com base no arquivo 'faltaCursar'.

O usuário pode combinar a seleção manual com o upload do arquivo para maior precisão.

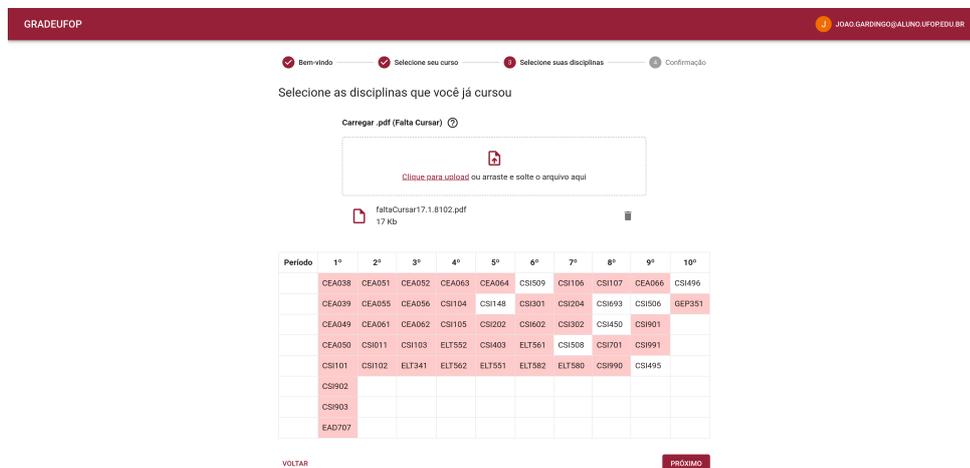


Figura 9 – Tela de Onboarding - Seleção de Disciplinas Cursadas

4.1.2.4 Confirmação dos Dados

A última etapa exibe um resumo das informações fornecidas, incluindo:

- O curso selecionado
- As disciplinas marcadas como já cursadas
- Informação de que essas opções podem ser alteradas posteriormente na tela de perfil

Ao clicar em "Salvar", os dados do usuário são salvos ou atualizados no sistema.

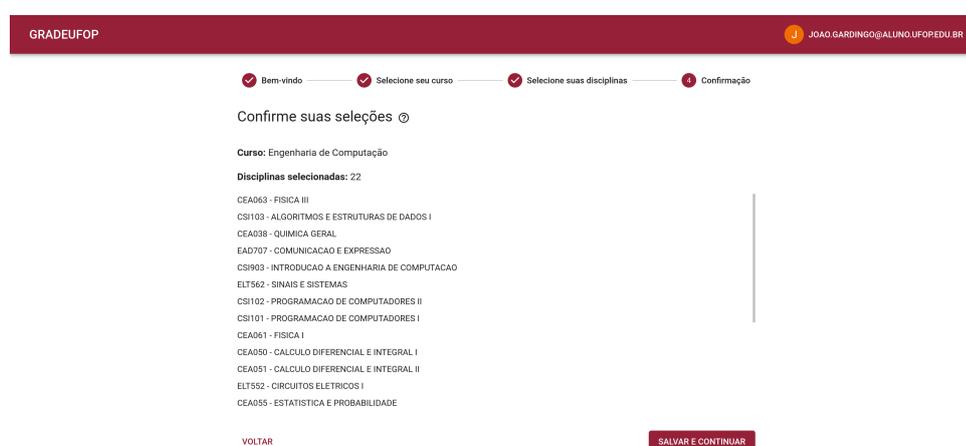


Figura 10 – Tela de Onboarding - Confirmação dos Dados

4.1.3 Tela de Montagem de Grades

A tela de montagem de grades é o núcleo da aplicação GradeUFOP. Ela oferece as seguintes funcionalidades:

- Adição de slots de horário;
- Seleção de disciplinas equivalentes para cada slot;
- Opção de adicionar todas as turmas de uma mesma disciplina ao slot;
- Filtros avançados:
 - Inclusão ou exclusão de disciplinas eletivas;
 - Opção de ignorar pré-requisitos (simulando quebra de pré-requisito);
 - Filtro por horários e dias da semana;
 - Atribuição de pesos para janelas entre aulas e dias com aula.

- Botão para gerar grades, que invoca o algoritmo de geração.

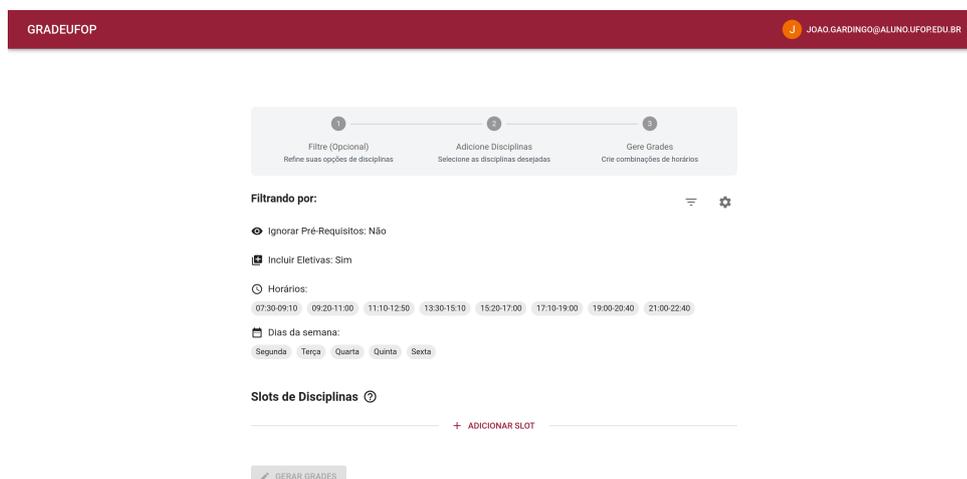


Figura 11 – Tela de Montagem de Grades - Estado inicial

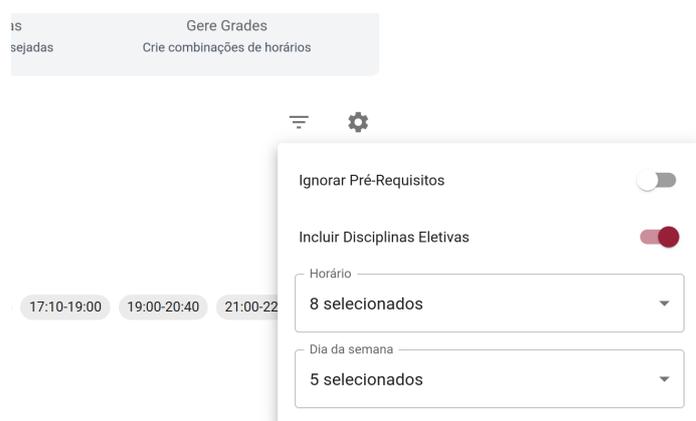


Figura 12 – Tela de Montagem de Grades - Menu de Filtros

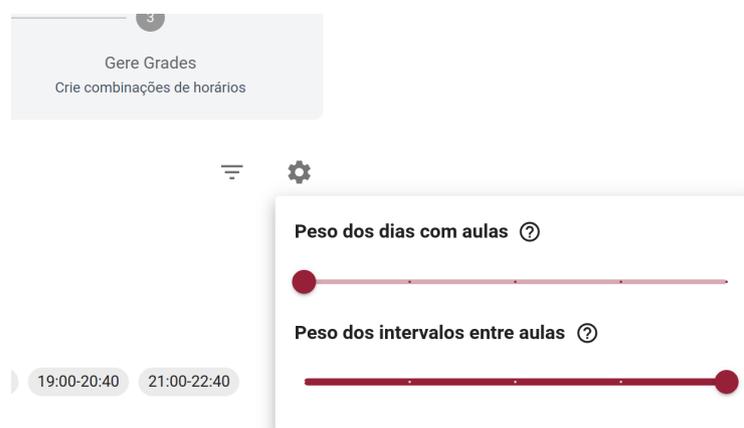


Figura 13 – Tela de Montagem de Grades - Menu de Preferências

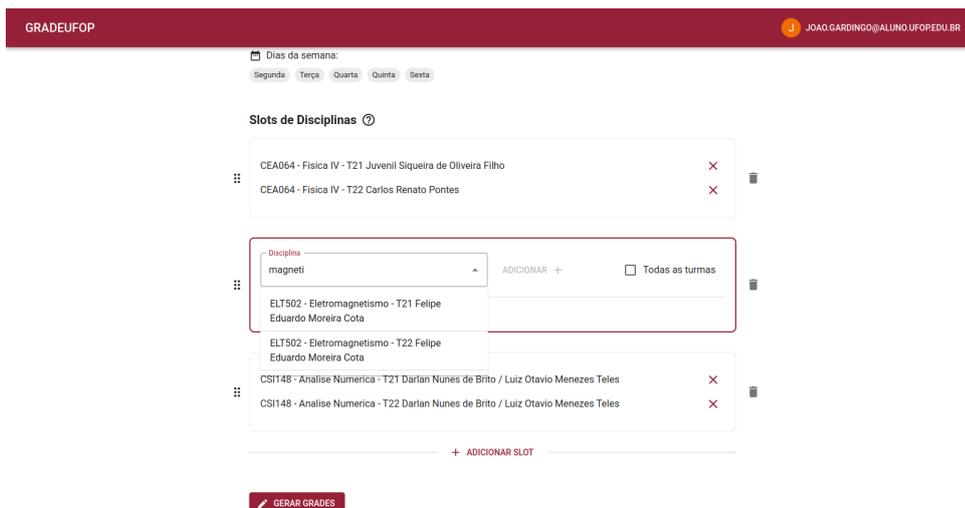


Figura 14 – Tela de Montagem de Grades - Adição de Disciplinas ao Slot

4.1.4 Tela de Grades Geradas

Esta tela exibe o resultado do algoritmo de geração de grades em um modal:

- Exibe até 10 melhores grades geradas pelo algoritmo;
- Apresenta uma tabela de horários com os códigos das disciplinas;
- Exibe a descrição detalhada das disciplinas escolhidas na grade (código, nome, professor e turma);
- Permite navegação entre as opções de grades;
- Oferece a opção de salvar a grade exibida.

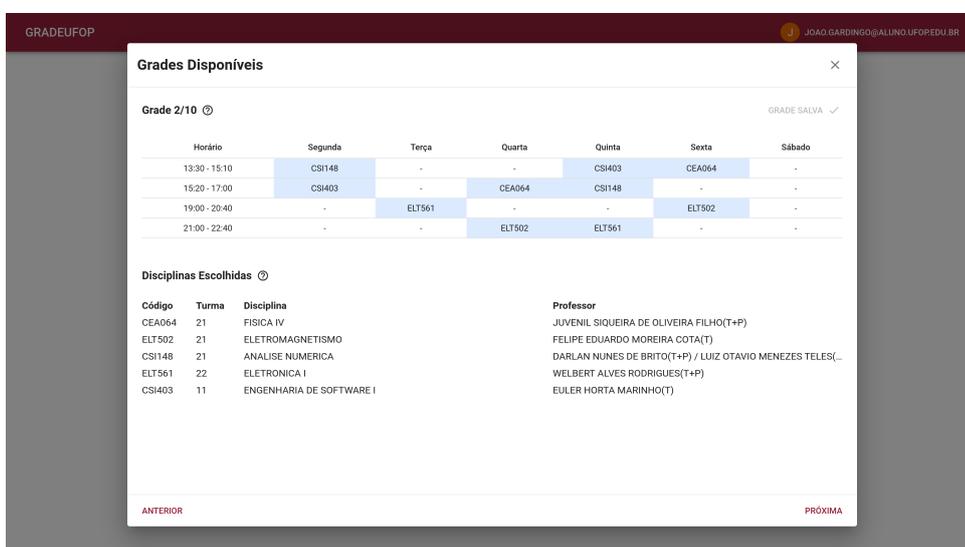


Figura 15 – Tela de Grades Geradas do GradeUFOP

4.1.5 Tela de Perfil

A tela de perfil do usuário inclui:

- Informações do usuário (nome, foto, email, etc.);
- Opção de logout;
- Opções para alterar informações do usuário:
 - Alterar curso;
 - Atualizar disciplinas já cursadas.
- Visualização e remoção de grades salvas.

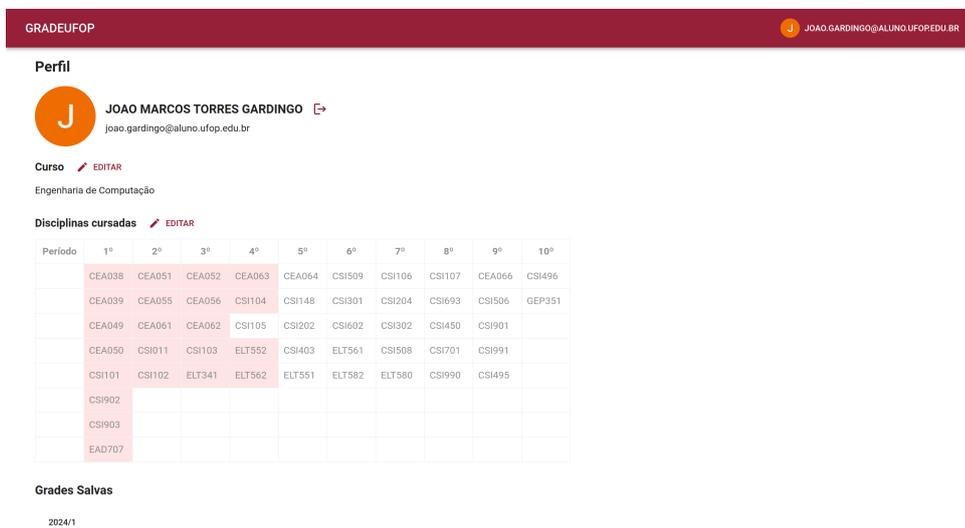


Figura 16 – Tela de Perfil - Informações do Usuário

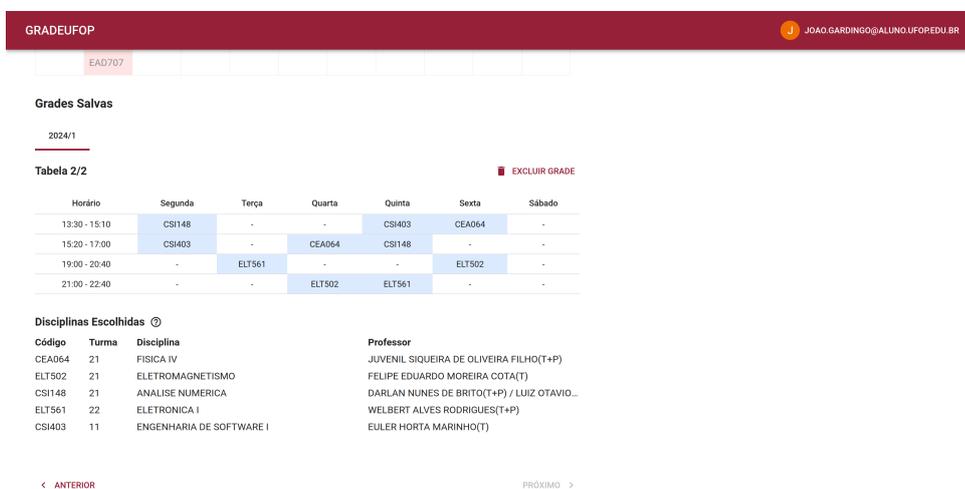


Figura 17 – Tela de Perfil - Grades Salvas

4.2 Processo de Geração de Grades

O processo de geração de grades foi implementado com sucesso, atendendo aos requisitos de desempenho estabelecidos. O processo leva em consideração:

- Disciplinas já cursadas pelo aluno;
- Pré-requisitos das disciplinas;
- Preferências do usuário (horários, dias da semana);
- Disponibilidade de turmas e professores.

O tempo médio de geração de grades ficou abaixo dos 5 segundos estipulados no requisito não funcional RNF01, mesmo para cenários complexos com múltiplos slots e disciplinas.

4.3 Integração com Dados da UFOP

A integração com os dados da UFOP foi realizada através do processo de raspagem de dados descrito na metodologia. Esta integração permitiu:

- Obtenção atualizada dos cursos oferecidos;
- Listagem precisa das disciplinas de cada curso;
- Informações sobre pré-requisitos e co-requisitos;
- Dados sobre professores e turmas disponíveis.

A execução deste processo garante que o sistema sempre trabalhe com os dados mais recentes da instituição.

4.4 Visão Geral da API

A API do GradeUFOP foi desenvolvida utilizando o framework Next.js, aproveitando a funcionalidade de API Routes. As principais rotas implementadas são:

Tabela 3 – Rotas principais da API do GradeUFOP

Rota	Método	Descrição
/api/auth	POST	Autenticação do usuário
/api/courses	GET	Obter lista de cursos
/api/disciplines	GET	Obter lista de disciplinas
/api/user	GET, PUT	Obter e atualizar dados do usuário
/api/generateSchedules	POST	Gerar combinações de grades

Além das rotas principais, foram desenvolvidos endpoints adicionais para coletar outros dados, como informações sobre departamentos, embora não sejam diretamente relevantes para a funcionalidade principal da aplicação.

É importante notar que todas as requisições à API requerem autenticação. Um middleware foi desenvolvido para verificar a presença do cookie de autenticação, garantindo que apenas usuários autenticados possam acessar os endpoints da API.

4.5 Disponibilidade em Produção e Repositórios

O processo de desenvolvimento e deploy do GradeUFOP foi concluído com sucesso, resultando em um sistema funcional e acessível em ambiente de produção. Os principais resultados neste aspecto são:

- **Acesso em Produção:** O GradeUFOP está disponível e pode ser acessado através do seguinte endereço: <<https://grade-ufop-puce.vercel.app>>. Este link fornece acesso à versão em produção do sistema, permitindo que usuários reais possam beneficiar-se das funcionalidades desenvolvidas;
- **Repositórios do Projeto:** O código-fonte do projeto está disponível publicamente, dividido em dois repositórios principais:
 - Repositório do Aplicativo Principal: <<https://github.com/joaomarcostg/grade-ufop>>
 - Repositório do *Scraper*: <<https://github.com/joaomarcostg/grade-ufop-scraper>>

Estes repositórios contêm todo o código desenvolvido para o GradeUFOP, incluindo o aplicativo web e o sistema de raspagem de dados, respectivamente.

A disponibilização do sistema em produção e a publicação dos repositórios não apenas demonstram a conclusão bem-sucedida do projeto, mas também permitem que a comunidade acadêmica da UFOP possa utilizar a ferramenta e, potencialmente, contribuir para seu desenvolvimento futuro.

5 Considerações Finais

O desenvolvimento do GradeUFOP representa um importante passo na otimização do processo de planejamento acadêmico para os estudantes da Universidade Federal de Ouro Preto. Através da implementação de um sistema web intuitivo e funcional, o projeto alcançou seu objetivo principal de simplificar a montagem de grades curriculares, oferecendo uma solução para os desafios enfrentados pelos alunos durante o período de ajuste de matrícula.

Os principais resultados alcançados incluem:

- Desenvolvimento de uma interface de usuário intuitiva e focada na experiência desktop;
- Implementação de um algoritmo eficiente para geração de grades curriculares;
- Integração bem-sucedida com os dados da UFOP através de um processo de raspagem;
- Disponibilização do sistema em ambiente de produção, acessível aos estudantes.

Além disso, é importante ressaltar que o GradeUFOP vai além de ser apenas um sistema para ajuste de grades curriculares. O projeto também disponibiliza uma API que centraliza dados atualizados da UFOP, os quais nem sempre são de fácil acesso. Esta característica amplia significativamente o potencial e o valor do projeto, pois:

- Fornece uma fonte unificada e atualizada de dados acadêmicos da UFOP;
- Abre possibilidades para o desenvolvimento de outros projetos e aplicações que podem consumir esses dados;
- Facilita pesquisas acadêmicas que necessitam de informações atualizadas sobre cursos, disciplinas e estruturas curriculares da universidade;
- Cria oportunidades para a evolução contínua da própria API, potencialmente expandindo seu escopo e funcionalidades.

Esta característica do GradeUFOP como uma plataforma de dados centralizada demonstra seu potencial para impactar positivamente não apenas o processo de planejamento acadêmico individual, mas também para contribuir com o ecossistema de desenvolvimento e pesquisa da UFOP como um todo.

5.1 Trabalhos Futuros

Apesar dos avanços alcançados, o GradeUFOP apresenta diversas oportunidades de melhoria e expansão. As seguintes áreas foram identificadas como potenciais focos para trabalhos futuros, agrupadas em quatro áreas principais:

5.1.1 Aprimoramento da Experiência do Usuário

Esta área foca na melhoria da interação dos usuários com o sistema, tornando-o mais acessível, responsivo e adaptado às necessidades dos estudantes.

Acessibilidade e Usabilidade: Realização de testes de acessibilidade, especialmente com usuários com deficiências visuais, e implementação de melhorias baseadas nesses resultados.

Feedback e Melhoria Contínua: Implementação de um sistema de feedback para coletar opiniões dos alunos e realização de testes com um grupo maior de usuários.

Responsividade e Design Mobile: Desenvolvimento de uma versão mobile refinada da aplicação e otimização da experiência do usuário em dispositivos móveis.

Expansão de Funcionalidades: Implementação de um sistema de recomendação de grades acadêmicas, incorporação de métricas de dificuldade das disciplinas e desenvolvimento de um sistema de pontuação para as grades geradas.

5.1.2 Aperfeiçoamento Técnico

Esta área concentra-se em melhorias técnicas para aumentar a robustez, eficiência e qualidade do código do sistema.

Testes e Qualidade de Código: Desenvolvimento de testes de unidade e integração para o aplicativo principal e o scrapper, aumentando a cobertura de testes.

Melhorias no Processo de Raspagem: Integração de modelos de IA para processamento de dados e refatoração do código do scrapper para funcionar como um job programável.

Otimização do Algoritmo: Refinamento do algoritmo de geração de grades, explorando abordagens além da combinatória atual e implementação de técnicas de otimização.

5.1.3 Escalabilidade e Infraestrutura

Esta área aborda questões relacionadas à capacidade do sistema de lidar com um maior número de usuários e dados

Infraestrutura e Escalabilidade: Migração do projeto para uma **VPS** (*Virtual Private Server*) para lidar com maior tráfego e otimização do sistema para suportar um uso massivo sem comprometer o desempenho.

5.1.4 Expansão do Escopo

Esta área foca na ampliação do alcance e da aplicabilidade do sistema para atender de forma efetiva toda a comunidade acadêmica da UFOP.

Expansão da Cobertura de Cursos: Embora o algoritmo de raspagem desenvolvido já seja capaz de coletar dados de todos os cursos ofertados pela UFOP, incluindo os campi de Ouro Preto, Mariana e João Monlevade, a validação completa do sistema foi concentrada inicialmente nos cursos do **ICEA**, devido a limitações de tempo. Esta decisão permitiu um teste mais controlado e aprofundado das funcionalidades do sistema.

Para garantir a confiabilidade e eficácia do GradeUFOP para toda a comunidade acadêmica, é necessário:

- Realizar testes extensivos com os cursos dos campi de Ouro Preto e Mariana;
- Validar o tratamento correto de casos específicos de cada curso;
- Verificar a precisão das relações de equivalência entre disciplinas de diferentes cursos;
- Confirmar a correta aplicação das regras de pré-requisitos específicas de cada matriz curricular.

Esta expansão na validação é fundamental para assegurar que o sistema atenda com precisão às necessidades específicas e particularidades de cada curso oferecido pela UFOP.

A implementação destas melhorias e expansões tem o potencial de transformar o GradeUFOP em uma ferramenta ainda mais poderosa e abrangente, beneficiando não apenas os estudantes da UFOP, mas potencialmente servindo como modelo para outras instituições de ensino superior. O caminho para estes desenvolvimentos futuros está aberto, e espera-se que a comunidade acadêmica e de desenvolvedores possa contribuir para a evolução contínua deste projeto.

Referências

- CAZELLA, S. C.; NUNES, M. A. S. N.; REATEGUI, E. B. A ciência da opinião: Estado da arte em sistemas de recomendação. *JAI: Jornada de Atualização em Informática da SBC*, p. 161–216, 2010. Citado na página 17.
- COSTA, M. A. d. O. *Grafos de Dependências de Disciplinas da UFOP*. 2021. GitHub. Acesso em: 8 de outubro de 2024. Disponível em: <<https://github.com/MarcoAOC/GrafosDependenciasUFOP>>. Citado na página 20.
- FERREIRA, V. H.; RAABE, A. L. A. Um sistema de recomendação de disciplinas em cursos de graduação. *Revista Brasileira de Informática na Educação*, v. 21, n. 01, p. 12–25, 2013. Citado 2 vezes nas páginas 17 e 21.
- GONÇALVES, T. C.; SILVA, J. C. d.; CORTES, O. A. C. Técnicas de mineração de dados: um estudo de caso da evasão no ensino superior do instituto federal do maranhão. *Revista Brasileira de Computação Aplicada*, v. 10, n. 3, p. 11–20, 2018. Citado na página 16.
- LOPES, C. P.; RIBEIRO, B. F.; MACIEL, A. M. F. Uma abordagem híbrida para sistemas de recomendação aplicados à educação. *Revista Brasileira de Informática na Educação*, v. 27, n. 01, p. 1–25, 2019. Citado na página 18.
- PARANHOS, H. P. *Desenvolvimento de um dashboard para análise e visualização dos dados educacionais dos discentes do Instituto de Ciências Exatas e Aplicadas da UFOP*. Dissertação (Mestrado) — Universidade Federal de Ouro Preto, João Monlevade, 2021. Citado 3 vezes nas páginas 15, 16 e 17.
- REATEGUI, E.; CAZELLA, S. C. Sistemas de recomendação. In: *XXV Congresso da Sociedade Brasileira de Computação*. [S.l.: s.n.], 2007. p. 306–348. Citado na página 17.
- SANTOS, J. S. d. *Business intelligence: uma proposta metodológica para análise da evasão escolar em instituições federais de ensino*. Dissertação (Mestrado) — Universidade Federal do Rio Grande do Norte, Natal, 2017. Citado 2 vezes nas páginas 15 e 16.
- SILVA, L. A. d. et al. Desempenho acadêmico e dados socioeconômicos: uma análise por meio de regressões gama e logística. *Avaliação: Revista da Avaliação da Educação Superior (Campinas)*, v. 24, n. 2, p. 434–454, 2019. Citado na página 21.
- UFOP. *PROJETO POLÍTICO-PEDAGÓGICO DO CURSO DE BACHARELADO EM SISTEMAS DE INFORMAÇÃO*. 2024. <https://decsi.ufop.br/sites/default/files/decsi/files/congrad141_anexo_retificado_07.02.2024.pdf>. Acesso em: 8 de outubro de 2024. Citado na página 12.

Apêndices

APÊNDICE A – Casos de Uso

A.1 UC01: Autenticação do Usuário

Descrição: Este caso de uso descreve o processo de autenticação de um estudante no sistema GradeUFOP.

Ator Principal: Estudante

Pré-condições:

- O estudante possui um e-mail institucional da UFOP.
- O sistema está acessível e operacional.

Fluxo Principal:

1. O estudante acessa o GradeUFOP.
2. O sistema exibe a opção de login com Google.
3. O estudante clica no botão "Conectar com Google".
4. O estudante se autentica com o Google.
5. O estudante aceita as condições.
6. O sistema cadastra o usuário, caso for um novo email.
7. O sistema autentica o estudante e redireciona para a página principal.

Fluxos Alternativos:

- 5a. Se as credenciais forem inválidas:
 1. O sistema exibe uma mensagem de erro.
 2. O sistema retorna ao passo 2 do fluxo principal.

Pós-condições:

- O estudante está autenticado e tem acesso às funcionalidades do sistema.

A.2 UC02: Cadastro de Curso e Disciplinas Cursadas

Descrição: Este caso de uso descreve o processo de um estudante selecionar seu curso atual e marcar as disciplinas já cursadas no sistema GradeUFOP.

Ator Principal: Estudante

Pré-condições:

- O estudante está autenticado no sistema.
- O estudante ainda não configurou seu curso e disciplinas cursadas.

Fluxo Principal:

1. O sistema exibe a tela de onboarding para seleção de curso e disciplinas.
2. O estudante seleciona seu curso atual no autocomplete.
3. O sistema exibe a lista de disciplinas obrigatórias do curso selecionado.
4. O estudante marca as disciplinas já cursadas.
5. O estudante tem a opção de importar dados do "Faltam Cursar" do portal da UFOP.
6. O sistema processa as informações e exibe um resumo das seleções.
7. O estudante confirma as informações.
8. O sistema salva os dados do estudante.

Fluxos Alternativos:

- 5a. Se o estudante opta por importar dados do "Faltam Cursar":
 1. O estudante faz upload do arquivo do portal da UFOP.
 2. O sistema processa o arquivo e marca automaticamente as disciplinas cursadas.
 3. O fluxo continua no passo 6.

Pós-condições:

- O curso do estudante está configurado no sistema.
- As disciplinas cursadas pelo estudante estão registradas.

A.3 UC03: Geração de Grade Curricular

Descrição: Este caso de uso descreve o processo de um estudante gerar combinações de grades curriculares baseadas em suas preferências.

Ator Principal: Estudante

Pré-condições:

- O estudante está autenticado no sistema.
- O estudante já configurou seu curso e disciplinas cursadas.

Fluxo Principal:

1. O estudante acessa a tela de montagem de grades.
2. O sistema exibe as opções de filtros e preferências.
3. O estudante configura os filtros desejados (horários, dias da semana, preferências).
4. O estudante cria slots de disciplinas.
5. O estudante adiciona disciplinas aos slots criados.
6. O estudante ordena os slots por prioridade usando drag and drop.
7. O estudante clica no botão "Gerar Grades".
8. O sistema processa as preferências e gera as combinações de grades possíveis.
9. O sistema exibe as grades geradas.

Fluxos Alternativos:

- 8a. Se nenhuma combinação válida for encontrada:
 1. O sistema exibe uma mensagem informando que não foi possível gerar grades com as preferências atuais.
 2. O sistema sugere ao estudante revisar seus filtros e preferências.
 3. O fluxo retorna ao passo 2.

Pós-condições:

- O sistema apresenta as combinações de grades geradas para o estudante.

A.4 UC04: Visualização e Seleção de Grade

Descrição: Este caso de uso descreve o processo de um estudante visualizar e selecionar uma grade curricular entre as opções geradas.

Ator Principal: Estudante

Pré-condições:

- O estudante está autenticado no sistema.
- O sistema gerou combinações de grades curriculares.

Fluxo Principal:

1. O sistema exibe as grades geradas em formato de tabela semanal.
2. O estudante visualiza as diferentes opções de grade.
3. O estudante seleciona uma grade para visualização detalhada.
4. O sistema exibe informações detalhadas da grade selecionada (horários, professores, códigos das disciplinas).
5. O estudante decide salvar a grade selecionada.
6. O sistema salva a seleção do estudante.

Fluxos Alternativos:

- 5a. Se o estudante decide não salvar a grade:
 1. O estudante retorna à lista de grades geradas.
 2. O fluxo retorna ao passo 2.

Pós-condições:

- A grade selecionada pelo estudante está salva no sistema.

A.5 UC05: Atualização de Perfil

Descrição: Este caso de uso descreve o processo de um estudante atualizar suas informações de perfil e preferências no sistema.

Ator Principal: Estudante

Pré-condições:

- O estudante está autenticado no sistema.

Fluxo Principal:

1. O estudante acessa a tela de perfil.
2. O sistema exibe as informações atuais do perfil do estudante.
3. O estudante edita as informações desejadas (curso, disciplinas cursadas).
4. O estudante confirma as alterações.
5. O sistema valida as alterações.
6. O sistema salva as novas informações.

Fluxos Alternativos:

- 5a. Se as alterações não forem válidas:
 1. O sistema exibe mensagens de erro indicando os campos inválidos.
 2. O fluxo retorna ao passo 3.

Pós-condições:

- As informações atualizadas do perfil do estudante estão salvas no sistema.