

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ

VINICIUS CESAR DE OLIVEIRA

**SISTEMA PARA SUPORTE AO PROCESSO MATRÍCULAS ACADÊMICAS DA
UTFPR**

GUARAPUAVA

2025

VINICIUS CESAR DE OLIVEIRA

**SISTEMA PARA SUPORTE AO PROCESSO MATRÍCULAS ACADÊMICAS DA
UTFPR**

System to support the UTFPR academic enrollment process

Projeto de Trabalho de Conclusão de Curso de Graduação apresentado como requisito para obtenção do título de Graduação em Sistemas para Internet do Curso de graduação em Sistemas para Internet da Universidade Tecnológica Federal do Paraná.

Orientador: Prof^o Dr. Luciano Ogiboski

Coorientador: Prof^o Dr. Emerson André Fedechen

GUARAPUAVA

2025



[4.0 Internacional](https://creativecommons.org/licenses/by/4.0/)

Esta licença permite compartilhamento, remixe, adaptação e criação a partir do trabalho, mesmo para fins comerciais, desde que sejam atribuídos créditos ao(s) autor(es). Conteúdos elaborados por terceiros, citados e referenciados nesta obra não são cobertos pela licença.

SUMÁRIO

1	INTRODUÇÃO	2
1.1	Objetivos	2
1.1.1	Objetivo geral	2
1.1.2	Objetivos específicos	3
2	DETALHAMENTO DA SOLUÇÃO	4
3	MATERIAIS E MÉTODOS	5
3.1	Materiais	5
3.2	Métodos	5
4	RESULTADOS PARCIAIS	8
4.1	Levantamento de requisitos	8
4.1.1	Requisitos Funcionais	8
4.2	História de usuários	9
5	CONSIDERAÇÕES FINAIS	11
	REFERÊNCIAS	12

1 INTRODUÇÃO

O processo de matrícula é uma etapa decisiva da vida acadêmica, pois influencia diretamente o ritmo de progressão no curso e o desempenho dos estudantes ao longo da graduação. Em muitos casos, no entanto, os sistemas utilizados pelas instituições de ensino permitem que os alunos escolham disciplinas sem verificar de forma adequada se os pré-requisitos necessários já foram cumpridos. Isso resulta em situações em que estudantes se matriculam em matérias para as quais não possuem a base de conhecimento exigida, o que pode gerar dificuldades de compreensão, aumento das reprovações e atrasos na conclusão do curso.

Esse cenário evidencia uma lacuna importante na forma como a escolha das disciplinas é conduzida. Sem um recurso que relacione o histórico acadêmico do aluno às dependências entre as matérias, as decisões acabam sendo tomadas de maneira pouco estruturada, limitando o aproveitamento do percurso formativo. Em cursos com currículos complexos, nos quais diversas disciplinas dependem do cumprimento de outras, essa ausência de orientação pode comprometer o desenvolvimento acadêmico.

O uso de uma ferramenta capaz de organizar as disciplinas em função dos pré-requisitos já concluídos surge como uma alternativa para enfrentar esse problema. Ao considerar as matérias cursadas e identificar automaticamente as opções disponíveis para o próximo período, esse tipo de solução pode apoiar escolhas mais conscientes, reduzir índices de reprovação e otimizar o tempo necessário para a formação. Além de beneficiar diretamente os estudantes, a instituição também se favorece com um fluxo acadêmico mais consistente e alinhado às exigências curriculares.

Ao final, espera-se apresentar um modelo de recomendação capaz de identificar, com base no histórico acadêmico do estudante, quais disciplinas estão elegíveis para matrícula, considerando seus pré-requisitos e relações de dependência. Dessa maneira, o projeto busca tornar o planejamento da fase de matrícula acadêmicas mais claro e objetivo, reduzindo incertezas no processo de escolha de disciplinas e oferecendo uma solução que poderá orientar melhorias futuras em ferramentas institucionais de gestão acadêmica.

1.1 Objetivos

1.1.1 Objetivo geral

Projetar e implementar um modelo capaz de recomendar disciplinas adequadas ao estudante com base nos pré-requisitos e na matriz curricular do curso.

1.1.2 Objetivos específicos

- Mapear as disciplinas do curso de Sistemas para Internet da UTFPR/Guarapuava, identificando seus pré-requisitos, cargas horárias e relações de dependência, de forma a subsidiar a construção da modelagem do sistema.
- Representar a matriz curricular, as disciplinas, suas relações de dependência e o histórico acadêmico dos estudantes, garantindo consistência e adequação ao processo de recomendação.
- Desenvolver uma modelagem de processamento responsável por analisar o histórico acadêmico do estudante e verificar os pré-requisitos atendidos, retornando um ou mais conjuntos de disciplinas elegíveis para matrícula.
- Validar o funcionamento da solução por meio de cenários simulados ou testes controlados, verificando se os conjuntos de disciplinas recomendadas estão coerentes com a matriz curricular e as regras de dependência definidas.

2 DETALHAMENTO DA SOLUÇÃO

A solução proposta consiste no desenvolvimento de uma modelagem de processamento responsável por analisar a estrutura curricular do curso de Sistemas para Internet da UTFPR/-Guarapuava e identificar quais disciplinas estão elegíveis para matrícula a partir de um conjunto de disciplinas previamente concluídas pelo estudante.

Inicialmente, o sistema receberá como entrada um conjunto de disciplinas já cursadas pelo aluno, representando seu histórico acadêmico. A partir desse conjunto, será realizada a consulta à matriz curricular do curso, previamente modelada no sistema, contendo todas as disciplinas, seus respectivos pré-requisitos, cargas horárias e relações de dependência.

Em seguida, será aplicado um processo de filtragem das disciplinas, no qual cada disciplina da matriz curricular será avaliada individualmente. Para cada disciplina analisada, o sistema verificará se todos os seus pré-requisitos estão contidos no conjunto de disciplinas já concluídas pelo estudante. As disciplinas que atenderem integralmente a esse critério serão classificadas como elegíveis para matrícula, enquanto aquelas que possuírem ao menos um pré-requisito não atendido serão consideradas não elegíveis no momento.

Após a verificação dos pré-requisitos, as disciplinas classificadas como elegíveis poderão ser submetidas a uma etapa adicional de análise baseada na carga horária. Nessa etapa, o sistema poderá agrupar ou organizar as disciplinas de forma a respeitar limites de carga horária definidos para o período letivo, possibilitando a formação de um ou mais conjuntos de disciplinas recomendadas para matrícula.

Ao final do processamento, o sistema retornará como resultado um conjunto estruturado contendo as disciplinas elegíveis para matrícula, considerando as restrições de carga horária. Esse retorno fornece uma visão clara da situação acadêmica do estudante em relação à matriz curricular, servindo como base para apoio à decisão no processo de matrícula e para futuras extensões da solução.

3 MATERIAIS E MÉTODOS

Neste capítulo, são apresentados os materiais e métodos empregados no planejamento e desenvolvimento do projeto. O conteúdo está dividido em duas seções. Na seção 3.1 é especificado cada material utilizado, enquanto na seção 3.2 são descritos os métodos utilizados nas etapas de planejamento e desenvolvimento.

3.1 Materiais

Esta seção descreve os materiais e as tecnologias empregadas no desenvolvimento do sistema, destacando suas características e sua contribuição para a implementação da solução proposta.

Ruby on Rails: O Ruby on Rails é um framework baseado na linguagem Ruby, amplamente utilizado para o desenvolvimento de aplicações web. Sua escolha fundamenta-se na capacidade de concentrar, em um único ambiente, interface, lógica de negócio e acesso a dados, permitindo a implementação de um sistema monolítico coeso (RAILS, 2025).

PostgreSQL: O PostgreSQL será utilizado como sistema de gerenciamento de banco de dados relacional, escolhido por sua robustez, conformidade com padrões SQL e eficiência no tratamento de dados estruturados. (POSTGRESQL, 2025).

Docker: O Docker será utilizado para padronizar o ambiente de desenvolvimento e facilitar a execução do sistema em diferentes máquinas. Utilizando contêineres, torna-se possível isolar dependências, garantir reprodutibilidade e reduzir problemas de incompatibilidade entre ambientes (DOCKER, 2025).

Visual Studio Code: O Visual Studio Code será empregado como editor principal de desenvolvimento, selecionado pela familiaridade do autor com a ferramenta, sua interface intuitiva e seu amplo ecossistema de extensões voltadas ao desenvolvimento web. A combinação dessas características contribui para maior produtividade e organização durante a implementação do sistema (MICROSOFT, 2025).

Git e GitHub: O Git (GIT, 2025) será utilizado para controle de versão, permitindo o acompanhamento, gerenciamento e armazenamento das alterações realizadas no código-fonte. Complementarmente, o GitHub (GITHUB, 2025) servirá como repositório remoto da aplicação, facilitando o versionamento, a colaboração e a documentação do projeto ao longo de seu desenvolvimento.

3.2 Métodos

Esta seção descreve a metodologia de desenvolvimento do projeto, detalhando as etapas adotadas para concepção, implementação e validação do sistema proposto.

Etapa 1 - Levantamento de requisitos: Como ponto de partida, será realizado o levantamento de requisitos do sistema. Essa etapa ocorrerá por meio de entrevistas com coordenadores e professores diretamente envolvidos no processo de matrícula, uma vez que eles conhecem as principais dificuldades e limitações do modelo atual. A partir das informações coletadas, serão identificadas as necessidades reais dos usuários, seus problemas recorrentes e critérios relevantes para a recomendação de disciplinas. Esses dados servirão como base para orientar o desenvolvimento da solução de maneira contextualizada e alinhada aos objetivos do projeto.

Etapa 2 - Priorização dos requisitos: Após o levantamento, os requisitos identificados serão priorizados utilizando a técnica MoSCoW (CONSORTIUM, 2025), que os classifica em quatro categorias:

- **Must:** requisitos essenciais ao funcionamento do sistema; devem obrigatoriamente ser implementados.
- **Should:** requisitos importantes, mas não vitais; adicionam valor significativo, porém o sistema continua funcional sem eles.
- **Could:** requisitos desejáveis, com menor impacto; serão implementados apenas se houver tempo e recursos disponíveis.
- **Won't:** requisitos não previstos para esta versão, mas que podem ser considerados futuramente.

Essa priorização permitirá organizar o desenvolvimento de forma estratégica, garantindo que as funcionalidades essenciais sejam tratadas primeiro.

Etapa 3 - Definição das Histórias de Usuário: Com base nos requisitos priorizados, serão elaboradas User Stories (Histórias de Usuário), detalhando cada funcionalidade sob a perspectiva do usuário. Para isso, será utilizado o formato proposto por Mike Cohn (COHN, 2025): “Como [tipo de usuário], quero [ação], para que [benefício].”

Etapa 4 - Modelagem do banco de dados: Em seguida, será realizada a modelagem do banco de dados, definindo as entidades, atributos e relacionamentos necessários ao funcionamento do sistema. Essa modelagem permite compreender melhor a estrutura dos dados e facilita a implementação das funcionalidades e da interface.

Etapa 5 - Desenvolvimento incremental e organização das Sprints: A implementação do sistema seguirá uma abordagem incremental, na qual as funcionalidades serão desenvolvidas, revisadas e ajustadas continuamente. Com base nas etapas anteriores, as tarefas serão estimadas de acordo com sua complexidade e organizadas em sprints. Para gerenciar essas sprints e acompanhar o progresso, será utilizada a ferramenta GitHub Projects.

Etapa 6 - Testes e validação: Testes serão aplicados de forma contínua durante o desenvolvimento, garantindo que cada funcionalidade atenda aos requisitos definidos. Os pro-

fessores/coordenadores participarão da definição dos cenários de uso e da validação dos resultados, assegurando que as recomendações de disciplinas e demais funções do sistema se mantenham coerentes com a realidade acadêmica. Essa abordagem contribui diretamente para a confiabilidade e a qualidade da solução final.

4 RESULTADOS PARCIAIS

Nesse capítulo serão apresentados os resultados parciais obtidos até o presente momento do projeto.

4.1 Levantamento de requisitos

O levantamento de requisitos foi realizado por meio de uma conversa com um professor que já atuou como coordenador de curso e possui experiência com os desafios enfrentados pelos estudantes no processo de matrícula. A partir de sua vivência, foi possível identificar problemas recorrentes relacionados à organização da matriz curricular, à definição de pré-requisitos e às dificuldades encontradas pelos alunos ao planejar a matrícula. As informações coletadas serviram como base para definir os requisitos funcionais do sistema, garantindo que a solução proposta esteja alinhada às necessidades reais do contexto acadêmico.

4.1.1 Requisitos Funcionais

RF01 — Permitir o cadastro de cursos de graduação (*Prioridade: Must*): O sistema deve permitir o cadastro de cursos de graduação, contendo informações como nome do curso, código e carga horária total.

RF02 — Permitir o cadastro de disciplinas (*Prioridade: Must*): O sistema deve permitir o cadastro de disciplinas, contendo informações como nome, código, carga horária, período recomendado e curso ao qual pertence.

RF03 — Permitir a configuração da matriz curricular do curso (*Prioridade: Must*): O sistema deve possibilitar o cadastro e a manutenção da matriz curricular de um curso, permitindo associar disciplinas a períodos específicos do curso.

RF04 — Permitir a definição de regras de dependência entre disciplinas (*Prioridade: Must*): O sistema deve permitir a configuração de pré-requisitos entre disciplinas, possibilitando que uma disciplina dependa de uma ou mais disciplinas previamente concluídas.

RF05 — Gerar recomendações de disciplinas para matrícula (*Prioridade: Must*): O sistema deve receber o conjunto de disciplinas já cursadas pelo estudante e gerar uma lista de disciplinas recomendadas para o próximo semestre.

RF06 — Verificar elegibilidade das disciplinas para matrícula (*Prioridade: Must*): O sistema deve validar se o estudante atende a todos os pré-requisitos exigidos por cada disciplina antes de incluí-la na lista de disciplinas recomendadas.

RF07 — Disponibilizar as recomendações via JSON (*Prioridade: Must*): O sistema deve disponibilizar as recomendações de disciplinas elegíveis para matrícula, em formato JSON.

RF08 — Permitir que o estudante informe a carga horária que pretende dedicar ao semestre (*Prioridade: Should*): O sistema deve permitir ao estudante definir seu limite de horas semanais ou mensais, influenciando a recomendação gerada.

RF09 — Permitir a atribuição de grau de complexidade às disciplinas (*Prioridade: Should*): O sistema deve possibilitar que cada disciplina receba um grau de complexidade, utilizado como critério adicional na recomendação das disciplinas, especialmente para equilibrar a carga do semestre.

RF10 - Criar interfaces gráficas para interação com os usuários (*Prioridade: Could*): O sistema poderá disponibilizar telas para cadastro, visualização e gerenciamento das informações, caso seja necessário para facilitar o uso.

RF11 — Permitir exportar recomendação (*Prioridade: Could*): O sistema deve possibilitar que a relação de disciplinas recomendadas seja exportada via PDF.

RF12 — Integração com sistemas oficiais (*Prioridade: Won't*): Integração com sistemas acadêmicos oficiais para importação de histórico ou submissão de matrícula.

RF13 — Flexibilidade para adaptação a diferentes cursos (*Prioridade: Won't*): O sistema deve ser projetado de forma genérica o suficiente para que suas regras e estruturas possam ser aplicadas a cursos diferentes, bastando ajustar a matriz curricular e seus atributos.

4.2 História de usuários

Nessa seção será apresentada as histórias de usuários. Elas são utilizadas para representar casos de uso das funcionalidades do sistema, de acordo com as conclusões obtidas pelo levantamento de requisitos e a priorização do projeto.

História de Usuário

Feature: Configurar matriz curricular do curso

Como administrador do sistema, **quero** cadastrar e organizar as disciplinas que compõem a matriz curricular de um curso, **para que** o sistema possua uma representação estruturada do curso e possa utilizá-la no processo de recomendação de matrícula.

História de Usuário

Feature: Permitir a definição de regras de dependência entre disciplinas

Como professor ou coordenador, **quero** configurar os pré-requisitos de cada disciplina, **para que** o sistema entenda corretamente as dependências e possa gerar recomendações consistentes.

História de Usuário

Feature: Realizar recomendações de matrícula baseadas nos pré-requisitos atendidos

Como estudante, **quero** receber uma lista de disciplinas recomendadas considerando as disciplinas já cursadas, **para que** possa realizar uma matrícula compatível com meu estágio atual na graduação.

5 CONSIDERAÇÕES FINAIS

O processo de matrícula, na forma como é realizado atualmente, pode se tornar confuso e pouco intuitivo para grande parte dos estudantes. A ampla quantidade de combinações possíveis para montar a grade semestral, somada à ausência de ferramentas capazes de orientar de forma clara e personalizada as escolhas do discente, acaba dificultando a tomada de decisão. Assim, o aluno frequentemente precisa interpretar sozinho a matriz curricular, compreender pré-requisitos, identificar dependências entre disciplinas e avaliar sua própria carga horária.

Os efeitos colaterais de uma matrícula mal planejada são muitos, dentre eles podemos destacar o atraso na formação, a matrícula em disciplinas para as quais o aluno ainda não possui o preparo adequado, a sobrecarga de carga horária, o aumento do risco de reprovações e, em casos mais graves, a desistência do curso por frustração ou acúmulo de dificuldades ao longo dos semestres. Para o próprio curso, esses equívocos também geram impactos negativos, como turmas desbalanceadas, dificuldade de organização do fluxo acadêmico e prejuízos na gestão do calendário e da oferta de disciplinas.

Esses elementos motivaram o desenvolvimento de uma ferramenta que apoie os estudantes no momento da matrícula, oferecendo orientações em acordo com a matriz curricular e no histórico individual de cada discente. Com o auxílio de um professor experiente, que já atuou como coordenador de curso e possui conhecimento das dificuldades enfrentadas pelos alunos, foi possível identificar problemas recorrentes no processo de matrícula e levantar possíveis estratégias para mitigá-los. Questões como a interpretação incorreta dos pré-requisitos, a pouca clareza sobre a ordem recomendada das disciplinas e a escolha de combinações inadequadas foram elementos para a definição dos requisitos do sistema.

Diante desse cenário, o presente projeto visa desenvolver uma solução que auxilie o aluno na construção de sua grade semestral. O sistema buscará alinhar as disciplinas já concluídas pelo discente com as exigências da matriz curricular, propondo recomendações mais adequadas ao seu nível e consistentes com o fluxo previsto para o curso. Dessa maneira, além de reduzir erros no processo de matrícula, espera-se proporcionar ao aluno maior clareza e confiança em suas escolhas acadêmicas, contribuindo também para a melhoria da organização interna do curso e para a eficiência do fluxo acadêmico como um todo.

REFERÊNCIAS

COHN, M. **User Stories**. 2025. <https://www.mountaingoatsoftware.com/agile/user-stories>. Acesso em: 19 nov. 2025.

CONSORTIUM, A. B. **MoSCoW Prioritisation**. 2025. <https://www.agilebusiness.org/dsdm-project-framework/moscow-prioritisation.html>. Acesso em: 19 nov. 2025.

DOCKER. **Docker**. 2025. <https://www.docker.com/>. Acesso em: 19 nov. 2025.

GIT. **Everything is local**. 2025. <https://git-scm.com/>. Acesso em: 19 nov. 2025.

GITHUB. **Build and ship software on a single, collaborative platform**. 2025. <https://github.com/>. Acesso em: 19 nov. 2025.

MICROSOFT. **Visual Studio Code**. 2025. <https://code.visualstudio.com/>. Acesso em: 19 nov. 2025.

POSTGRESQL. **PostgreSQL**. 2025. <https://www.postgresql.org/>. Acesso em: 19 nov. 2025.

RAILS. **Ruby on Rails**. 2025. <https://rubyonrails.org/>. Acesso em: 19 nov. 2025.