

UNIVERSIDADE FEDERAL DE UBERLÂNDIA (UFU)

Faculdade de Engenharia Elétrica (FEELT)

Curso de Engenharia de Computação

Proposta de Projeto da Disciplina de Engenharia de Software

Integra UFU

Iago - (Coloque sua matrícula aqui)

Larissa - (Coloque sua matrícula aqui)

Matheus da Silva Fernandes - 12121ECP002

Thúlio Santos Arrais - (Coloque sua matrícula aqui)

Uberlândia – MG

Novembro de 2025

1. Descrição do Grupo

A equipe responsável pelo desenvolvimento do projeto **Integra UFU** é composta por quatro estudantes do curso de Engenharia de Computação da Universidade Federal de Uberlândia (UFU), matriculados na disciplina de Engenharia de Software (ESOF).

Cada integrante foi designado a funções específicas, de acordo com suas habilidades técnicas e perfis individuais, a fim de garantir uma distribuição equilibrada de responsabilidades e eficiência no desenvolvimento do sistema.

- **Matheus Fernandes** — Arquiteto de Software e Desenvolvedor Back-end. Responsável pela estrutura do sistema, modelagem de dados e implementação das regras de negócio.
- **Integrante 2** — Desenvolvedor Front-end e Designer de Interface. Responsável pelo desenvolvimento da interface do usuário (UI) e experiência do usuário (UX).
- **Integrante 3** — Analista de Testes e Garantia de Qualidade (QA). Responsável por testes unitários, de integração e validação das funcionalidades, age como suporte técnico e supervisiona os códigos do Front e Back-end.
- **Integrante 4** — Gerente de Projeto e Documentação. Responsável pela organização das sprints, cronograma, reuniões SCRUM e elaboração dos relatórios técnicos, o mesmo é responsável pela apresentação final do projeto.

2. Objetivos do Trabalho

O projeto **Integra UFU** tem como objetivo central o desenvolvimento de uma plataforma integrada que centralize funcionalidades úteis à comunidade acadêmica da Universidade Federal de Uberlândia.

O foco principal é a criação de um **sistema de reservas de laboratórios, salas e anfiteatros**, permitindo o agendamento e controle de uso de espaços físicos da instituição de forma prática e acessível.

Além disso, o projeto prevê a integração futura de um módulo voltado à **consulta de Planos de Ensino do semestre**, facilitando o acesso a informações acadêmicas que atualmente se encontram dispersas em diferentes portais institucionais.

Os objetivos específicos incluem:

- Proporcionar um sistema de reservas intuitivo e responsivo;
- Oferecer gerenciamento centralizado de horários, recursos e permissões;
- Facilitar o acesso a informações institucionais relevantes;
- Promover maior eficiência na comunicação entre alunos, professores e administração;
- Servir como um protótipo de integração acadêmico-administrativa.

3. Metodologia e Etapas

O grupo adotará a metodologia ágil **SCRUM** para a condução do projeto, permitindo uma abordagem iterativa, colaborativa e adaptável às mudanças de requisitos durante o semestre.

3.1 Papéis e Artefatos

- **Product Owner (PO):** Representado coletivamente pelo grupo, responsável pela definição e priorização do backlog.
- **Scrum Master:** Integrante 4, responsável por assegurar o cumprimento das práticas ágeis e organizar as sprints.
- **Time de Desenvolvimento:** Integrantes 1, 2 e 3, encarregados da implementação e testes.

3.2 Sprints Planejadas

O desenvolvimento será dividido em **sprints**, com duração variada ao longo da semana.

1. **Sprint 0:** Levantamento de requisitos e prototipagem inicial;
2. **Sprint 1:** Implementação do sistema de autenticação e base de dados;
3. **Sprint 2:** Desenvolvimento do módulo de reservas de laboratórios e salas;
4. **Sprint 3:** Implementação do painel administrativo e notificações;
5. **Sprint 4:** Testes, correções e refinamentos;
6. **Sprint 5:** Documentação final e apresentação.

3.3 Metodologia Técnica

O sistema seguirá uma arquitetura em camadas (*multilayer architecture*), com separação entre apresentação, lógica de negócios e persistência de dados. Será desenvolvida uma **API RESTful** utilizando **ASP.NET Core**, integrada a um banco de dados **PostgreSQL ou MongoAtlas DB**, e um front-end moderno baseado em **ReactJS**.

4. Entregas e Cronograma

As entregas ao longo do semestre serão divididas conforme o progresso do desenvolvimento. Cada sprint deverá gerar um incremento funcional do produto, bem como sua respectiva documentação.

- **Entrega 1:** Documento de proposta e diagrama de arquitetura;
- **Entrega 2:** Documento de requisitos e protótipos de interface;
- **Entrega 3:** Módulo de autenticação e gerenciamento de usuários;
- **Entrega 4:** Sistema de reservas de salas e laboratórios;
- **Entrega 5:** Relatórios, notificações e testes;
- **Entrega 6:** Apresentação final e documentação técnica completa.

O código-fonte será versionado em um repositório público no **GitHub**, com histórico de commits, issues e releases registrados.

5. Arquitetura de Software

5.1 Visão Geral

O sistema **Integra UFU** será baseado em uma arquitetura cliente-servidor de três camadas:

1. **Camada de Apresentação (Frontend):** Interface web responsiva desenvolvida em ReactJS.
2. **Camada de Aplicação (Backend):** API RESTful desenvolvida em ASP.NET Core.
3. **Camada de Dados:** Banco de dados relacional PostgreSQL ou MongoDB.

5.2 Componentes Principais

- **AuthService:** Controle de autenticação e autorização com JWT;
- **ReservationService:** Lógica de criação e validação de reservas;
- **ResourceService:** Gerenciamento de salas, laboratórios e anfiteatros;
- **NotificationService:** Envio de notificações e alertas;
- **PlanosService:** Módulo futuro para consulta de planos de ensino, teams, moodle ...

5.3 Modelo de Dados (resumo inicial)

- `users(id, nome, email, senha_hash, role)`
- `resources(id, nome, tipo, local, capacidade)`
- `reservations(id, user_id, resource_id, inicio, fim, status)`
- `notifications(id, reservation_id, tipo, data_envio)`

6. Testes e Garantia de Qualidade

Os testes serão realizados em diferentes níveis:

- **Testes Unitários:** Validação de funções e classes do backend;
- **Testes de Integração:** Comunicação entre API e banco de dados;
- **Testes de Interface:** Validação de usabilidade e responsividade;
- **Testes de Aceitação:** Avaliação das funcionalidades sob o ponto de vista do usuário.

7. Considerações Finais

O projeto **Integra UFU** propõe uma solução prática e escalável para integrar e simplificar processos internos da Universidade Federal de Uberlândia, focando na usabilidade, centralização de informações e automação de reservas institucionais.

Por meio da aplicação dos princípios de Engenharia de Software e do uso de metodologias ágeis, espera-se que o projeto proporcione uma experiência significativa de aprendizado e gere um produto de relevância real para a comunidade acadêmica.