 <b>Unifor</b>	<b>UNIVERSIDADE DE FORTALEZA</b>	
<b>Curso: Ciências da Computação</b>		<b>Turma:T242-18</b>
<b>Disciplina: Desenv Plataformas Web</b>		
<b>Aluno(s): Caio Ciriaco Ribeiro, João Guilherme Mapurunga Cavalcante, Anna Lara Magalhães Monteiro Vieira, Jean de Souza Moraes, Pedro Lucas Dias Freitas</b>		
<b>Turno: Noite</b>	<b>Data: 01/04/2025</b>	<b>Período Letivo: 2025.1</b>

## **Plano de Trabalho: Sistema de Gerenciamento dos Serviços Gerais da UNIFOR.**

**FORTALEZA, 2025**

## SUMÁRIO

<b><i>VISÃO GERAL</i></b> .....	<b>3</b>
Introdução.....	3
Objetivo.....	3
<b><i>TECNOLOGIAS</i></b> .....	<b>4</b>
Front-end.....	4
Back-end.....	4
Design e Prototipação.....	4
Versionamento.....	4
<b><i>METODOLOGIA PARA O DESENVOLVIMENTO DO SISTEMA</i></b> .....	<b>5</b>
Descrição da Metodologia.....	5
Definição dos Papéis do Scrum.....	5
Planejamento Inicial.....	5
Planejamento e Estruturação dos Sprints.....	5
Desenvolvimento do MVP (1º Mês).....	6
Iteração e Finalização (2º Mês).....	6
<b><i>ARTEFATOS DO PRODUTO</i></b> .....	<b>8</b>
Atores.....	8
Requisitos Funcionais.....	8
Requisitos Não-funcionais.....	9
Diagrama de casos de uso.....	10
<b><i>DIAGRAMA DE ENTIDADE-RELACIONAMENTO (DER)</i></b> .....	<b>11</b>
<b><i>WIREFRAME</i></b> .....	<b>11</b>

# **VISÃO GERAL**

## **Introdução**

Atualmente, a gestão de estoque e o acompanhamento dos serviços gerais na UNIFOR enfrentam desafios como falta de controle eficiente sobre materiais, dificuldades no monitoramento das atividades executadas pela equipe e demora no atendimento de solicitações. Esses problemas impactam diretamente a organização, gerando desperdícios, atrasos e falta de previsibilidade na reposição de insumos essenciais.

## **Objetivo**

Nosso projeto visa desenvolver um sistema integrado que otimize o gerenciamento de estoque e o acompanhamento dos serviços gerais, garantindo um controle preciso dos materiais, um monitoramento eficiente das atividades da equipe e um fluxo ágil no registro e atendimento de solicitações. Isso permitirá maior eficiência operacional, redução de desperdícios e melhora na organização dos processos.

# TECNOLOGIAS

Para o desenvolvimento do sistema, dividimos o uso das ferramentas da seguinte forma:

## Front-end

Para a parte visual e interativa, será utilizado *HTML*, *CSS*, *JavaScript* e *UI Frameworks* (*Tailwind CSS*).

## Back-end

Será utilizado *MySQL* para o banco de dados e *Node.js* para o servidor.

## Design e Prototipação

Para a parte gráfica do projeto e protótipos, será usado *Figma* e *Excalidraw*.

## Versionamento

Para controle de versão, será utilizado *git* e *github*.

# METODOLOGIA PARA O DESENVOLVIMENTO DO SISTEMA

## Descrição da Metodologia

Adotaremos a Metodologia *Scrum*, que nos permitirá a entrega incremental e iterativa, além de favorecer a colaboração e o ajuste rápido às mudanças, especialmente considerando os prazos curtos para o MVP e a entrega final. Abaixo, segue o processo detalhado:

## Definição dos Papéis do Scrum

- **Product Owner (PO): João Mapurunga**  
Responsável por definir e priorizar os requisitos do sistema. Será a ponte entre os *stakeholders* (setor de serviços gerais) e a equipe de desenvolvimento.
- **Scrum Master: Caio Ribeiro**  
Responsável por facilitar as reuniões, remover impedimentos e garantir que a metodologia seja seguida.
- **Equipe de Desenvolvimento**  
Os 5 integrantes (incluindo o PO e o *Scrum Master*) irão colaborar na implementação das funcionalidades.

## Planejamento Inicial

- Levantamento dos requisitos e listagem de todas as funcionalidades necessárias;
- Criação do *Product Backlog*;
- Definição do escopo do MVP.

## Planejamento e Estruturação dos Sprints

Serão realizados sprints semanais, com a seguinte divisão:

- **Sprint 0 (Planejamento)**
  - Configurar ambiente de desenvolvimento e versionamento (github);
  - Definir e planejar as sprints subsequentes.
- **Sprints 1 a 8 (Execução)**

- **Sprint Planning**  
No início de cada sprint, selecionaremos histórias do Product Backlog que serão realizadas durante o período, definindo tarefas específicas.
- **Desenvolvimento e Revisões Diárias**  
*Dailys* de 15 minutos para alinhamento de atividades, identificação de impedimentos e ajustes.
- **Sprint Review**  
Apresentação das funcionalidades desenvolvidas para o PO, que, caso possível, às mostrará para os usuários (para feedback).
- **Sprint Retrospective**  
Será discutido o que funcionou bem e o que pode ser aprimorado para melhorar os próximos sprints.

## **Desenvolvimento do MVP (1º Mês)**

Durante o primeiro mês, o foco principal será a criação de um MVP (*Minimum Viable Product*) que contemple as funcionalidades essenciais do sistema. Este MVP servirá como base para validar as principais necessidades dos usuários e dos processos internos.

## **Iteração e Finalização (2º Mês)**

Após a entrega do MVP, o segundo mês será dedicado a aprimorar e expandir o sistema, incorporando melhorias e novas funcionalidades, visando a finalização e entrega do projeto.

# DIVISÃO DE FUNÇÕES

## **Caio Ribeiro - Desenvolvedor Front-end**

Responsável pela parte interativa do sistema (JavaScript) e *Scrum Master*.

## **Jean Moraes - Desenvolvedor Front-end**

Responsável pela parte visual do sistema (HTML e CSS).

## **João Mapurunga - Desenvolvedor Back-end**

Responsável pelo banco de dados (MySQL) e *Product Owner* (PO).

## **Anna Iara - UX/UI Design**

Responsável por todo o design da aplicação (Figma e Excalidraw).

## **Pedro Lucas - Desenvolvedor Back-end**

Responsável pelo servidor (Node.js).

# ARTEFATOS DO PRODUTO

## Atores

**Equipe de Manutenção:** Este ator é o responsável por manter sistema, como realizar cadastro de usuários administradores.

**Administrador:** Ator responsável por delegar os serviços aos zeladores e controlar o fluxo de trabalho de forma geral.

**Funcionário:** Responsável por realizar os serviços delegados. No sistema, ele registra suas próprias atividades quando concluídas e visualiza seu fluxo.

## Requisitos Funcionais

Cod	Requisito	Prioridade
RF01	O sistema deve permitir cadastro de usuários administradores.	1
RF01-1	O usuário administrador deve poder cadastrar novos funcionários	1
RF01-2	O novo funcionário cadastrado deve ter um perfil próprio criado automaticamente	1
RF01-3	O sistema deve gerenciar permissões de usuário (admin e funcionário)	1
RF02	O sistema deve possuir controle de estoque de materiais.	1
RF02-1	No estoque, o sistema deve permitir adicionar e remover novos materiais.	1
RF02-2	No estoque, o sistema deve permitir filtrar por materiais e tipo.	2
RF03	O sistema deve permitir solicitação de compra de materiais.	1
RF04	O sistema deve permitir delegar serviços a funcionários.	1
RF04-1	Ao delegar funcionários, o sistema deve exibir formulário para descrição das atividades.	1
RF04-2	O sistema deve exibir uma descrição padrão do serviço a ser realizado.	2
RF05	O sistema deve permitir notificações.	1
RF05-1	O sistema deve notificar funcionários de uma nova delegação de serviço.	1



RF05-2	O sistema deve notificar funcionários sobre mudanças de serviços (seja local, horário e etc.	1
RF06	O sistema deve possuir formulário de serviço realizado pelo funcionário.	1
RF06-1	O formulário deve solicitar indicação de materiais utilizados.	1
RF07	O sistema deve atualizar automaticamente o estoque após o uso de algum material pelo funcionário.	1
RF08	O sistema deve possuir histórico de serviços realizados.	1
RF08-1	O histórico deve exibir, por padrão, os serviços dos últimos 7 dias.	2
RF08-2	O histórico deve permitir filtragem por data.	2
RF09	O sistema deve exibir quadro do fluxo semanal geral de serviços.	2
RF10	O sistema deve permitir leitura de qr code.	3

## Requisitos Não-funcionais

Cod	Requisito	Prioridade
RNF01	O sistema deve permitir o acesso somente com usuário e senha.	1
RNF02	O sistema deve ser criado com HTML, CSS e JavaScript.	3
RNF03	O sistema deve ser integrado com o banco de dados MySQL.	3
RNF04	O servidor do sistema deve ser feito em Node.js	3
RNF05	O sistema deve ser acessado somente conectado à rede internet.	1
RNF06	O sistema deve ser escrito no idioma português brasileiro.	1
RNF07	O sistema deve seguir as normas da LGPD.	1
RNF08	O sistema deve poder ser acessado por navegadores web no desktop e em dispositivos móveis.	1
RNF08-1	O sistema deve se adaptar a diferentes tamanhos de tela	1

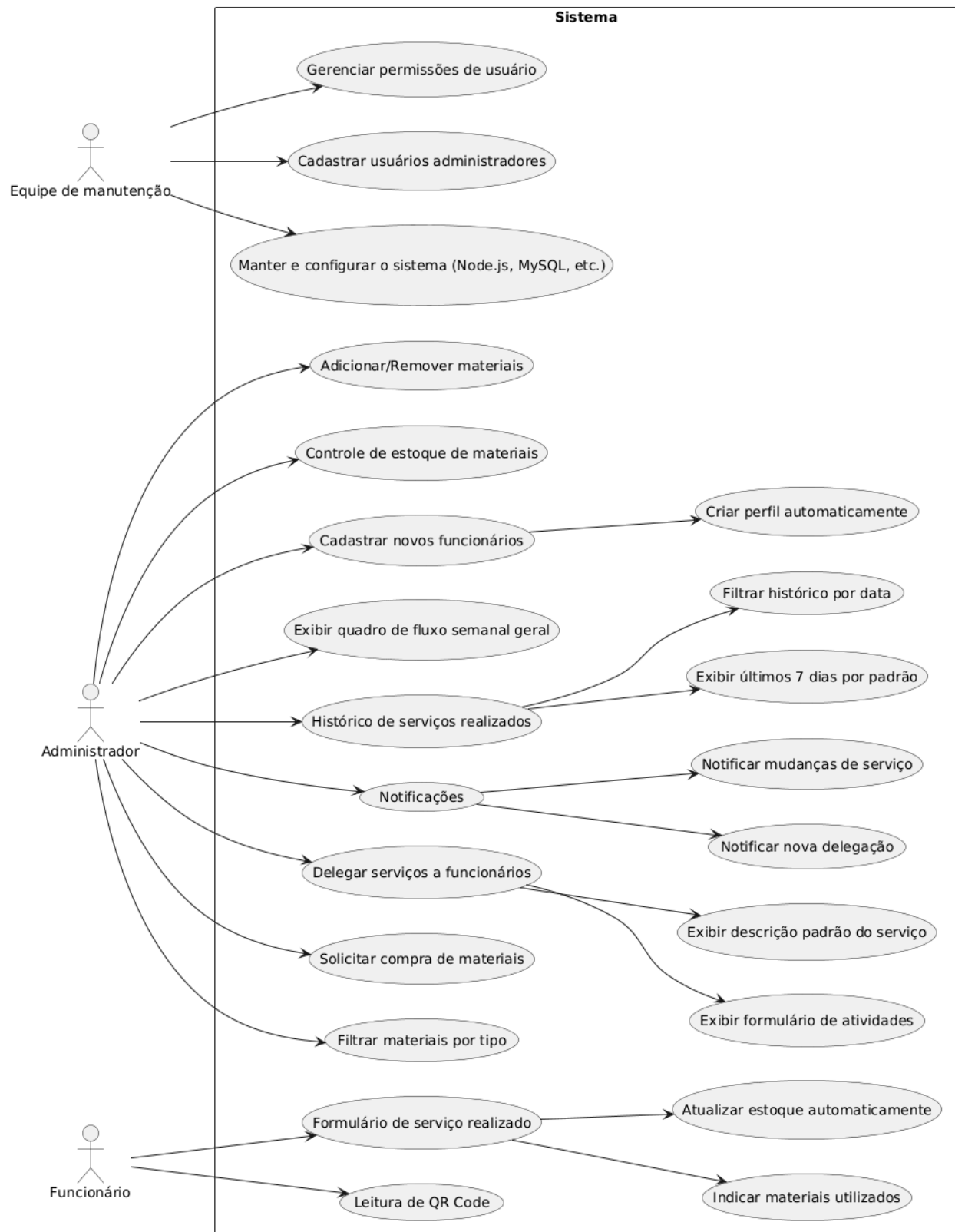
### Classificação para Prioridade dos requisitos:

1 - (Deve ter): São os requisitos essenciais e de alta prioridade, considerados fundamentais para o sucesso do sistema. São os requisitos que devem ser implementados obrigatoriamente, pois sem eles o sistema não atenderia às necessidades básicas dos usuários finais ou não cumpriria os objetivos do projeto.

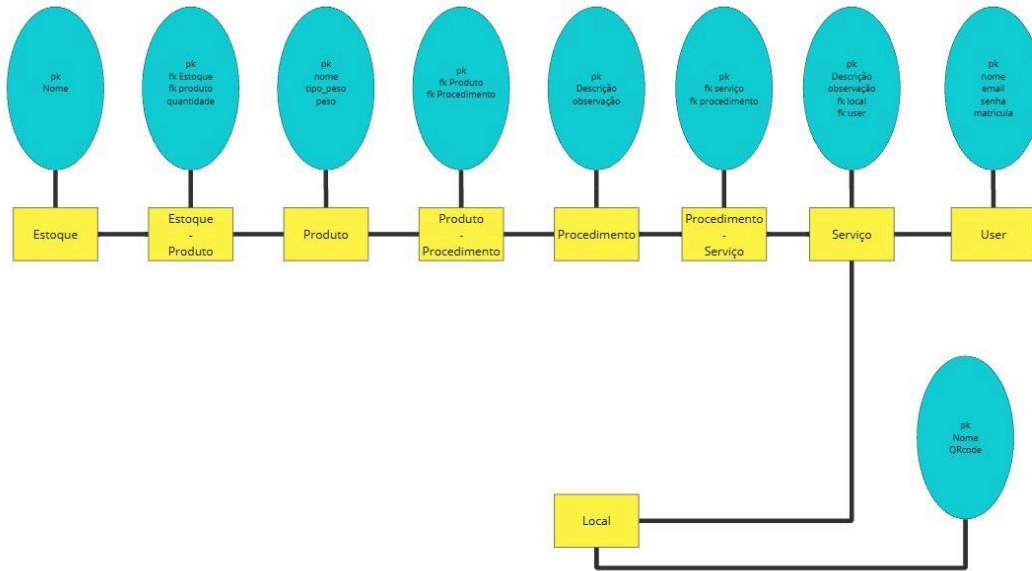
2 - (Deveria ter): São os requisitos importantes, mas não tão críticos quanto os de prioridade 1. Esses requisitos devem ser implementados se possível, mas podem ser adiados caso haja restrições de tempo ou recursos.

3 - (Poderia ter): São os requisitos desejáveis, mas não essenciais. São requisitos que podem trazer benefícios adicionais ao sistema, mas sua ausência não afetaria significativamente a funcionalidade principal.

## Diagrama de casos de uso



# DIAGRAMA DE ENTIDADE-RELACIONAMENTO (DER)



## WIREFRAME

O *wireframe* do projeto pode ser visualizado através do link:

[https://excalidraw.com/#json=2StCO5DF7eUFzOauo1Hvb,hLtBnv2QS8mTANVlknT\\_CA](https://excalidraw.com/#json=2StCO5DF7eUFzOauo1Hvb,hLtBnv2QS8mTANVlknT_CA)