

# Sistema de Gerenciamento de Tarefas (To-Do/Task Manager)

## Desafio:

- O objetivo do projeto é desenvolver uma aplicação web para o gerenciamento de tarefas, permitindo que usuários organizem suas atividades, anexem documentos e colaborem em equipe.

## Parceiro:

### → GSW

- ◆ A GSW é uma empresa especializada em gerar soluções para o gerenciamento, controle de processos e negócios. Referência no desenvolvimento de soluções em Tecnologia da Informação, oferece a seus clientes sistemas de TI que se traduzem em melhorias significativas para o dia a dia do negócio, principalmente no que se refere à redução de custos e ao aumento da produtividade.

## Requisitos do Projeto:

### → Funcionais

- ◆ O sistema deve permitir a criação, edição, exclusão e conclusão de tarefas.
- ◆ O sistema deve permitir a atribuição de tarefas a qualquer usuário.
- ◆ O sistema deve permitir anexar arquivos (como comprovantes, recibos, imagens, documentos) às tarefas.
- ◆ O sistema deve permitir que todo usuário possa realizar login e cadastro utilizando e-mail e senha.
- ◆ O sistema deve permitir a visualização do prazo de data e hora das tarefas.
- ◆ O sistema deve permitir a visualização de um log de auditoria das alterações de tarefas (para administradores).
- ◆ O sistema deve permitir a visualização de tarefas gerais da equipe (para administradores).
- ◆ O sistema deve permitir a visualização de tarefas atribuídas ao usuário logado.
- ◆ O sistema deve permitir alternar entre a visualização de "Todas as Tarefas" e "Minhas Tarefas" (para administradores).
- ◆ O sistema deve permitir o recebimento de notificações na interface para tarefas relacionadas ao usuário.
- ◆ O sistema deve permitir o acesso à tarefa diretamente através da notificação.
- ◆ O sistema deve exibir um histórico de notificações.
- ◆ O sistema deve permitir a recuperação de senha via e-mail.
- ◆ O sistema deve permitir a criação e gerenciamento de equipes (nome e descrição).

- ◆ O sistema deve permitir adicionar e remover membros de uma equipe.
- ◆ O sistema deve permitir a visualização da lista de todas as equipes e seus membros.
- ◆ O sistema deve permitir a sincronização de prazos de tarefas com calendários externos (ex: Google Calendar).
- ◆ O sistema deve permitir definir/editar o prazo de conclusão de uma atividade.
- ◆ O sistema deve sinalizar visualmente atividades próximas do prazo ou atrasadas.
- ◆ O sistema deve permitir a categorização de atividades (por prioridade, tipo, área).
- ◆ O sistema deve permitir filtrar atividades por categoria.
- ◆ O sistema deve permitir filtrar tarefas por meio de uma barra de pesquisa e um botão de filtro (por categoria, nome, título, descrição, prazo, usuário e status).
- ◆ O sistema deve permitir arrastar e soltar tarefas entre diferentes colunas de status (Não Iniciada, Em Andamento, Concluída, Atrasada).
- ◆ O sistema deve exibir um modal com o resumo da tarefa, histórico de alterações (para administradores) e sinalização visual de prazos vencidos ao clicar em uma tarefa.
- ◆ O sistema deve permitir que administradores filtrem o log de auditoria por data, usuário e tipo de alteração.

## → Não Funcionais

- ◆ Manual do Usuário (requisito Fatec – obrigatório);
- ◆ Documentação API – Application Programming Interface;
- ◆ Modelagem de Banco de Dados ou Arquivo de dados;
- ◆ A aplicação deve ser responsiva.
- ◆ O código deve ser organizado de forma a permitir novos recursos no futuro sem grandes problemas.
- ◆ Todo usuário deve ter usuário e senha e só pode ver atividades que ele esteja vinculado.
- ◆ A interface deve ser de fácil utilização.

## Dados de Teste:

### → Email

- ◆ **admin@gsw.com**
  - Testar funcionalidades de perfil, visualização de todas as tarefas.
  - Testar o tratamento de erros e a segurança do sistema.
  - Testar funcionalidade de log de auditoria.
- ◆ **thedevs@gsw.com**
  - Testar login, visualização de "Minhas Tarefas", criação, edição, exclusão e conclusão de tarefas atribuídas a ele, anexar arquivos.
  - Testar o tratamento de erros e a segurança do sistema
- ◆ **usuario@gsw.com**
  - Similar ao thedevs, mas para testar a atribuição de tarefas a um terceiro usuário e verificar se as tarefas aparecem corretamente para ele.

- Testar o tratamento de erros e a segurança do sistema

#### → Senha

- ◆ thedevs123
- ◆ admin123
- ◆ usuario123

#### → Tipos de Arquivos para Anexos:

- ◆ documento.pdf
- ◆ imagem.jpg
- ◆ planilha.xlsx
- ◆ video.mp4

### Tecnologias:

#### → Frontend:

- ◆ HTML, CSS, JavaScript, TypeScript, React, Tailwind, Node.js.

#### → Backend:

- ◆ Java, Spring, Apache Maven.


#### → Banco de Dados:

- ◆ MongoDB.

#### → Controle de Versão:

- ◆ Git

### Fluxograma UI:

→  Fluxo UI.png

### Dependências:

- **História 1.2 (Anexos)** depende de **História 1.1 (CRUD)**
- **História 1.3 (Atribuição)** depende de **História 1.1 (CRUD)**
- **História 6 (Visualização)** depende de **História 1.1 (CRUD)** e **História 1.3 (Atribuição)**
- **História 5 (Log de Auditoria)** depende de **História 1.1 (CRUD)**
- **História 6 (Visualização)** depende de **História 1.1 (CRUD)** e **História 1.3 (Atribuição)**
- **História 7 (Notificações de Tarefas)** depende de **História 1.1 (CRUD)**, **História 1.3 (Atribuição)** e **História 1.4 (Autenticação Básica)**
- **História 8 (Recuperação de Senha)** depende de **História 1.4 (Autenticação Básica)**
- **História 2.0 (Calendário)** depende de **História 1.1 (CRUD)** e **História 1.4 (Autenticação Básica)**
- **História 2.1 (Modal de Tarefa)** depende de **História 1.1 (CRUD)** e **História 1.5 (Log de Auditoria)**
- **História 2.2 (Filtragem do Log)** depende de **História 1.5 (Log de Auditoria)**

- **História 2.3 (Filtro de Tarefas) depende de História 1.1 (CRUD)**
- **História 2.4 (Arrastar e Soltar Tarefas) depende de História 1.1 (CRUD) e História 1.5 (Log de Auditoria)**