

## ATIVIDADE PRÁTICA - XP

### Integrantes do grupo

RA	NOME COMPLETO
12722211571	Odinelson Leandro Ferreira dos Santos
12722210600	Yan Caique Santos Muniz
12722133121	Diego da Conceição Santos
1272221138	João Pedro Ferreira Guimarães
12724141297	Luan Cerqueira São Pedro

### Visão do geral

Este projeto é uma aplicação de Lista de Tarefas (To-Do List) com suporte a categorias personalizadas, permitindo que o usuário organize suas atividades de forma prática e eficiente. É possível agrupar tarefas por temas como Trabalho, Pessoal e Estudo assim facilitando a visualização e o gerenciamento do dia a dia.

### Funcionamento

**Criação de Tarefas:** O usuário pode adicionar novas tarefas informando um título, uma descrição opcional e associando-as a uma categoria existente.

**Gerenciamento de Categorias:** É possível criar, editar e excluir categorias personalizadas, permitindo uma organização flexível das tarefas.

**Listagem e Filtros:** As tarefas são exibidas agrupadas por categoria e podem ser filtradas por status, como *pendente* ou *concluída*, facilitando o acompanhamento.

**Edição e Conclusão de Tarefas:** O usuário pode marcar tarefas como concluídas, além de editar ou excluir tarefas a qualquer momento.

**Persistência de Dados:** As informações são salvas através do banco de dados local utilizando o método JASON para melhor estruturação.

### Decisões do Projeto

**Organização por categorias:** Permite agrupar tarefas por temas, facilitando a visualização e o gerenciamento de atividades relacionadas.

**Interface simples e intuitiva:** O design prioriza a usabilidade, garantindo uma experiência rápida, clara e acessível ao usuário.

**Armazenamento persistente:** As tarefas e categorias são mantidas mesmo após o fechamento do navegador ou aplicação, assegurando continuidade no uso.

**Controle de status (pendente/concluída):** A marcação de tarefas como concluídas permite acompanhar o progresso e manter o foco nas atividades pendentes.

## Tecnologias Utilizadas

**Frontend:**HTML CSS

**Backend:**Node.js e JavaScript

**Banco de dados:** Local utilizando o método JASON.

## Aplicação dos Princípios do XP no Projeto

Durante o desenvolvimento deste projeto, buscamos adotar alguns princípios e práticas do Extreme Programming (XP) com foco na qualidade, simplicidade e colaboração contínua. O XP nos ajudou a focar não apenas na rapidez da implementação, mas também na qualidade, na adaptação às mudanças e na comunicação constante entre os membros da equipe.

**Desenvolvimento Iterativo:** O projeto foi construído em ciclos curtos, permitindo validar funcionalidades básicas, como criação de tarefas, antes de adicionar recursos mais avançados como categorias e filtros.

**Refatoração contínua:** A estrutura do código foi aprimorada ao longo do projeto para manter a clareza, reutilização e organização, especialmente após cada nova funcionalidade.

**Simplicidade:** Optamos por soluções diretas e objetivas, evitando complexidade desnecessária. Cada funcionalidade foi implementada apenas no momento em que se mostrou necessária.

**Testes Frequentes:** Implementamos testes básicos para validar as operações principais, como criação, edição, exclusão e persistência das tarefas, assegurando o funcionamento correto e prevenindo erros.

### Histórias de usuários:

**1** - Como usuário, quero adicionar uma nova tarefa com título e categoria, para que eu possa organizar meus afazeres.

**Critérios de Aceite:** O frontend possui um campo para o título e outro para a categoria, e um botão de adicionar. Ao clicar em adicionar, a tarefa é enviada para o backend via POST e aparece na lista.

**Implementação:** Criamos a função `adicionar_tarefa()` para atender esses requisitos.

**2** - Como usuário, quero visualizar todas as minhas tarefas com seus títulos, status de conclusão e categorias, para ter uma visão geral do que preciso fazer.

**Critérios de Aceite:** Ao carregar a página, as tarefas são buscadas do backend e exibidas claramente, mostrando se estão completas e a qual categoria pertencem.

**Implementação:** As funções `get tarefas()` no frontend e a rota `app.get('/tarefas')` no backend, junto com `display_Tarefas()`, implementam essa funcionalidade.

**3** - Como usuário, quero editar o título, status de conclusão e categoria de uma tarefa existente, para corrigir ou atualizar suas informações.

**Critérios de Aceite:** Existe um botão "Editar" para cada tarefa que abre um modal com os campos pré-preenchidos. Após editar e salvar, as alterações são enviadas ao backend via PUT e a lista é atualizada.

**Implementação:** As funções `open_modal()`, `editar_tarefa()` e a rota `app.put('/tarefas/:id')` no backend suportam essa história e tem como objetivo o que foi solicitado.

**4** - Como usuário, quero excluir uma tarefa, para remover itens que não são mais necessários.

**Critérios de Aceite:** Há um botão "Deletar" para cada tarefa. Ao clicar, a tarefa é removida do backend via DELETE.

**Implementação:** A função `deletar_tarefa()` no frontend e a rota `app.delete('/tarefas/:id')` no backend.

### **Programação em Par**

Nossa equipe de desenvolvimento trabalhou em pares na construção do documento, e na construção de estilização do front end, para que o usuário fique satisfeito também visualmente. Durante a programação do back-end foi cada um ajudando o outro.