# Teste Prático

## Descrição do Projeto

Desenvolva uma aplicação full-stack simples para um sistema de gerenciamento de tarefas (To-Do List) usando .NET Core (ou .NET Framework) no backend e jQuery ou Angular no frontend. O banco de dados deve ser o SQL Server (ou MySQL) com o uso de Entity Framework ou Dapper como ORM.

## Requisitos Técnicos

### Backend:

Desenvolver a API utilizando .NET Core (ou .NET Framework) para criação de uma API REST que gerenciará as tarefas. Implementar os seguintes endpoints para manipulação de tarefas: Criar tarefa (título, descrição, data de vencimento) Listar tarefas Atualizar o status da tarefa (Pendente/Concluída) Deletar tarefa Implementar regras de negócio básicas, como validação de dados de entrada e tratamento de erros.

### Frontend:

Desenvolver usando jQuery ou Angular. A interface deve permitir: Exibir a lista de tarefas com data de vencimento formatada. Um formulário para adicionar uma nova tarefa. Uma opção para alterar o status da tarefa (Pendentes/Concluídas). Um botão para deletar tarefa.

### Banco de Dados:

Usar SQL Server ou, opcionalmente, o MySQL como banco de dados. Implementar a persistência de dados usando Entity Framework ou Dapper como ORM.

### Geral:

Controle de versionamento usando Git com commits significativos e frequentes para registrar o desenvolvimento. Incluir um arquivo README com instruções de configuração e execução do projeto.

## Funcionalidades

### API para Gerenciamento de Tarefas:

Criar tarefa: título, descrição e data de vencimento. Listar tarefas (pode incluir paginação dependendo da quantidade de dados). Atualizar status da tarefa (de Pendente para Concluída e vice-versa). Deletar tarefa. As tarefas devem ser armazenadas no banco de dados utilizando Entity Framework ou Dapper.

### Interface de Usuário:

Página que lista as tarefas, mostrando também o status (Pendente ou Concluída) e a data de vencimento formatada em fuso horário local. Um formulário permitir a criação de novas tarefas. Cada tarefa deve ter uma opção para alterar seu status e uma opção para excluir a tarefa. Usar jQuery ou Angular para o frontend, com trabalho mínimo em estilização (CSS/Bootstrap).

## Critérios de Avaliação

Qualidade e organização do código: Código limpo, organizado e seguindo boas práticas. Implementação correta das funcionalidades solicitadas: Funcionalidades descritas devem ser corretamente implementadas e funcionais. Uso adequado do .NET Core (ou Framework) e integração com frontend: Domínio e boa utilização da stack solicitada. Integração correta com o banco de dados (SQL Server/MySQL): Consultas estão funcionando corretamente, utilizando Entity Framework ou Dapper com ligações adequadas. Front-end: Implementar de forma adequada as interações com o backend, com tratamento básico (como erros de rede e falhas de API). *Uso eficiente do Git: Histórico de commits bem documentado e organizado. Manipulação correta de datas: Conversão de datas, manipulação de timezones e formatação para exibição no fuso horário correto do cliente/usuário.

## Entrega

Crie um repositório público no GitHub. Inclua no repositório um arquivo README.md contendo: Instruções detalhadas para configuração do ambiente e execução do projeto (incluindo setup do banco de dados). Passo a passo para rodar o projeto tanto para frontend quanto para o backend. Breve explicação sobre a estrutura e as decisões técnicas do seu projeto. Considerações extras ou melhorias implementadas (caso tenha incluído algum diferencial como boas práticas de performance, por exemplo).

Após finalizar, responda esse email com o repositório do projeto.

Tecnologia Extra (Diferencial) Caso tenha experiência com Node.js e deseja agregar valor ao teste, você pode implementar um serviço backend adicional escrevendo um microserviço em Node.js para realizar alguma funcionalidade complementar, como o envio de notificações/lembranças por data de vencimento das tarefas via e-mail (não obrigatório, apenas um diferencial).

Prazo de Entrega Você tem 3 dias após o recebimento deste desafio técnico para concluí-lo e enviar. Boa sorte!