

Sistema de Controle de Viagens

Diogo Henrique da Silva

André Luiz Ferreira

10/06/2025

Turma: ADS Noite

1. Resumo do Projeto e do Problema Abordado

O projeto tem como objetivo o desenvolvimento de uma API para controle e gerenciamento de viagens. A aplicação é voltada para cenários que exigem controle eficiente de dados de deslocamentos, como agências de turismo, empresas com viagens corporativas e sistemas logísticos. A solução permite registrar, consultar e excluir informações de viagens, proporcionando agilidade e organização na gestão de dados.

2. Lista das Funcionalidades Implementadas

- Registro de viagens
- Consulta de todas as viagens
- Exclusão de viagens
- Atualização de dados de viagens
- Dashboard contabilizando a quantidade de viagens registradas e o valor total das viagens.

3. Descrição Técnica das Funcionalidades e da Integração com a API

A aplicação foi estruturada em camadas, com destaque para a modelagem da entidade 'Viagem', que inclui atributos como destino, data de ida e volta, meio de transporte, status da viagem e informações do viajante. A classe Viagem.cs serve como modelo de dados e base para o banco SQLite.

O contexto do banco é definido por ViagensContext.cs, que permite as operações CRUD e a atualização das datas de inclusão e datas de atualização dos registros. As rotas estão organizadas em arquivos separados para cada tipo de operação (GET, POST, DELETE, PUT), facilitando manutenção e expansão futura.

A integração da API é feita por meio de endpoints RESTful, que permitem aos sistemas externos e usuários acessarem os dados de maneira segura e eficiente. A inserção inicial de dados é realizada via [SeedViagens.cs](#).

Foi implementado a Data de criação para ter o controle sobre a data de geração daquele registro. Também implementado a Data de Atualização para quando ocorrer uma atualização do registro. A lógica de implementação se encontra no [ViagensContext.cs](#) e na model Viagens.cs

4. Endpoints

Método	Rota	Descrição
GET	/api/viagens	Retorna todas as viagens
GET	/api/viagens/{id}	Retorna uma viagem por ID
POST	/api/viagens	Adiciona uma nova viagem
DELETE	/api/viagens/{id}	Remove uma viagem
PUT	/api/viagens/{id}	Atualiza uma viagem

4. Endpoints

Para executar o projeto você precisa:

Rodar o comando na pasta do projeto “**dotnet restore**”

Aplicar as migrations e criar o banco de dados

Se for a primeira vez rodando, crie o banco e as tabelas:

dotnet ef database update

Executar a API

dotnet run