

Documentação do Backend em C# .NET

Este projeto implementa um sistema de cadastro de pessoas e transações utilizando a arquitetura MVC (Model-View-Controller) em .NET. Os dados são armazenados em arquivos JSON para simular um banco de dados.

Funcionalidades Principais

- **Cadastro de Pessoas:** Criação, deleção e listagem de pessoas.
- **Cadastro de Transações:** Criação e listagem de transações (despesas e receitas).
- **Consulta de Totais:** Exibição de totais de receitas, despesas e saldo por pessoa e um resumo geral.

Estrutura do Projeto

```
backend
|   Program.cs
|   Startup.cs
|   backend.csproj
|
|---Controllers
|   |   TransactionsController.cs
|   |   UsersController.cs
|
|---Data
|   |   transactions.json
|   |   users.json
|
|---Models
```

```
|      | Transaction.cs
|      | User.cs
|
|-----Services
|      | TransactionService.cs
|      | UserService.cs
|-----Properties
|      | launchSettings.json
```

Modelos

User.cs

```
public class User
{
    public int Id { get; set; }
    public required string Name { get; set; }
    public required int Age { get; set; }
}
```

Transaction.cs

```
public class Transaction
{
    public int Id { get; set; }
    public required string Description { get; set; }
}

    public decimal Value { get; set; }
    public required string Type { get; set; } //
Despesa ou Receita
    public int UserId { get; set; }
```

}

Controllers

UserController.cs

- GET /users → Retorna a lista de todas as pessoas cadastradas.
- POST /users → Cria uma nova pessoa.
- DELETE /users/{id} → Deleta uma pessoa e suas transações associadas.

TransactionsController.cs

- GET /transactions → Retorna a lista de todas as transações.
- POST /transactions → Cria uma nova transação, verificando se a pessoa é maior ou menor de idade para validar o tipo de transação.
- GET /transactions/totals → Retorna os totais de receitas, despesas e saldo por pessoa e o total geral.

Serviços

UserService.cs

- Gerencia operações de criação, listagem e remoção de usuários.
- Garante a exclusão de transações associadas ao remover um usuário.
- Armazena e lê dados do arquivo `users.json`.

TransactionService.cs

- Gerencia operações de criação e listagem de transações.
- Valida transações para menores de idade (apenas despesas permitidas).
- Calcula os totais de receitas, despesas e saldo.
- Armazena e lê dados do arquivo `transactions.json`.

Configurações

- **appsettings.json** e **appsettings.Development.json**:
Contém configurações como caminhos para arquivos JSON e definições de ambiente.

Inicialização

A configuração da aplicação é gerenciada em `Startup.cs`.

- Configurações de CORS.
- Configuração de serviços injetados.
- Configuração de endpoints.

Considerações Finais

Este sistema é uma solução simples baseada em armazenamento local com JSON. Para produção, é recomendado o uso de um banco de dados SQL ou NoSQL. Além disso, a aplicação pode ser expandida com funcionalidades como autenticação e paginação de resultados.