Documento Funcionamento do Teste

1. **Banco de dados**

O Banco de dados foi desenvolvido em Entity Framework e migrations.

O banco de dados está configurado no appSettings,

Todas as Configurações do banco estão na pasta “*Data*”.

Texto

Descrição gerada automaticamente

O *“AppDbContext”* é o modelo do banco no c#.

Os *“Maps”* são usados para configurar os campos das classes com o campo no banco

Para criar o banco de dados do zero é só utilizar os comandos:

-Add-Migration InitialDB -Context AppDbContext

-Update-Database

1. **Projeto**

O Projeto foi desenvolvido utilizando o padrão de projeto de repositórios e API com Swagger.

Para realizar os testes do projeto será necessário rodar o projeto e ter o postman instalado.

Para autentificar o usuário é necessário fazer login. Já tem dois usuários no sistema:

[robin@gmail.com](mailto:robin@gmail.com) com nível 1 e [batman@gmail.com](mailto:batman@gmail.com) com nível 2.

Exemplo:

Interface gráfica do usuário, Texto, Aplicativo, Email

Descrição gerada automaticamente

Depois de logar com o usuário é necessário pegar o token para autorizar os gerentes

Interface gráfica do usuário, Texto, Email

Descrição gerada automaticamente

Colocar o token no Header com Key Autorization e VALUE Bearer ”espaço” +token como no exemplo aqui:

Interface gráfica do usuário, Texto, Aplicativo

Descrição gerada automaticamente

1. **Requisitos**

Um Cliente deve pertencer a apenas um grupo:

Interface gráfica do usuário, Texto, Aplicativo, Email

Descrição gerada automaticamente

Gerentes precisam estar autenticados:

Sem autorização:

Interface gráfica do usuário, Texto, Aplicativo, Email

Descrição gerada automaticamente

Com autorização:

Interface gráfica do usuário, Texto, Aplicativo

Descrição gerada automaticamente

Gerente de nível um pode apenas visualizar grupos, adicionar/remover clientes

Permissão pra visualizar

Interface gráfica do usuário, Texto, Aplicativo, Email

Descrição gerada automaticamente

Interface gráfica do usuário, Texto, Aplicativo, Email

Descrição gerada automaticamente

Interface gráfica do usuário, Texto, Aplicativo, Email

Descrição gerada automaticamente

Sem permissão para deletar

Interface gráfica do usuário, Texto, Aplicativo, Email

Descrição gerada automaticamente

Sem permissão para cadastrar

Interface gráfica do usuário, Texto, Aplicativo, Email

Descrição gerada automaticamente

Gerente de nível dois é o único que pode criar editar e excluir grupos

Devido ter muitos testes eu optei por mostrar as permissões via código.

Texto

Descrição gerada automaticamente

Obs.:Os [Authorize] são os tipos de permissão um ou dois

1. **Objetivo**

Queremos endpoints para operar os grupos e visualizar os clientes de um grupo:

Endpoints para operar os grupos

Interface gráfica do usuário, Aplicativo, Email

Descrição gerada automaticamente

Endpoint para visualizar os clientes de um grupo

