

O projeto consiste em cadastramentos de clientes, onde temos alguns requisitos colocados no projeto.

Em relação ao backend :

Rotas

Salvamento dos dados
(POST)

- Guardar um novo usuário
- Gerar um novo token

Retorno (GET)

- Exibir o usuário da sessão
- Exibir todos os clientes do usuário

Alteração de dados (PUT)

- Atualizar os dados do usuário da sessão

Deletar dados (DELETE)

- Excluir o usuário da sessão

Esses são só requisitos que não precisam estar autenticados, pois são novos dados.

Requisitos que precisam ser autenticados:

Salvamento dos dados
(POST)

- Guardar um novo cliente do

usuário Retorno (GET)

- Exibir um cliente do

usuário Alteração de dados (PUT)

- Atualizar os dados de um cliente do usuário

Deletar Dados (Delete)

- Excluir um cliente do usuário

Utilizamos nas duas rotas os metodos http que trabalham com a interface onde envia uma requisição para oque se quer via web.

As camadas utilizadas para esse projeto ser realizado foram

Camada de Controle: Que consiste no recebimento de uma requisição e direciona pro serviço correto

Camada DTO: Que serve para manipulação dos dados . DTO seria uma classe que provê exatamente aquilo que é necessário para um determinado processo. E que facilitam o transporte dos dados

Camada de Entidade: Onde terá os atributos, as principais tabelas da aplicação (objetos) (Id, nome, Telefone,cep) que são utilizadas na autenticação dos usuários e clientes. Aplicação das regras de negócio.

Camada de Repositório: A camada que foi trabalhada a persistência dos dados, onde associa a persistência do banco. Ou seja o salvamento dos dados

Camada de serviços: Responsável pelo contato com a camada de repositório, para que se houver outras funcionalidades externas da aplicação sejam realizadas.

Camada de Visualização: Foi adicionada json para o front conseguir consumir os dados.

Na parte visual do projeto mais conhecida como o frontend, foi utilizado Angular e bootstrap 4.0.

Para login é preciso criar o usuário e para isso precisa de uma requisição POST diretamente para o backend, para realizar as requisições para o back foi utilizado o HttpClient do angular.

Na hora do login, depois da criação do usuário, é enviado as credenciais onde o backend retornará o token JWT, e esse token é enviado nos headers (cabeçalho) para garantia que é o próprio usuário que está realizando essa requisição.

Para garantir a autenticação do usuário as paginas seguintes, o projeto se utiliza do

canActivate do routes, que libera o acesso se o usuário estiver logado e com o token JWT.