**API**

**EndPoints:** é a URL que “chama” alguma requisição da API no controller.

[HttpGet],[HttpPost]... etc

**Organização (design de pastas)**

-Domains: Representam as tabelas do banco de dados, parecido com Models.

-Controllers: Fazem a intermediação entre o front e a API, diz qual rota irá apontar. É o controller que recebe todas as requisições (request) da API.

-Repositories: Faz a comunicação com o banco de dados.

-Interfaces: Diz quais os métodos que existem nas classes.

O formato das respostas (retorno) das requisições são em JSON. (Anotação em JavaScript)

Objeto em json:

**{**

**“**chave**”**: **“**valor**”,**

**“**chave**”**: **{**

**“**objetoFilho**”**: **“**filhinho**”,**

**},**

**}**

**REST:** É um protocolo, padrão de boas práticas e regras que precisa-se seguir para criar uma API.

**Response Status Code:**

**HTTP STATUS CODE: ~~alguns~~**

Level 200 (Success)

Level 400 (Bad Request)

401: Inauthorized (Relacionado com usuário que não esta logado, por exemplo)

403: Forbidden (Proibido, relacionado com permissões, por exemplo, Administradores fazem coisas que usuários não tem permissão para fazer)

Level 500 (Internal server Error)

**JWT (JSON WEB TOKEN)**

Header, payload, signature

Claims (informações do usuário)

**Componentes do Token:**

**issuer:** emissor do token

**audience:** destinatário do token

**claims:** dados definidos acima

**expires:** tempo de expiração

**signingCredentials:** credenciais do token

**Swagger (documentação)**

Swashbuckle.AspNetCore

Introdução (documentação microsoft): <https://docs.microsoft.com/pt-br/aspnet/core/tutorials/getting-started-with-swashbuckle?view=aspnetcore-3.1&tabs=visual-studio>