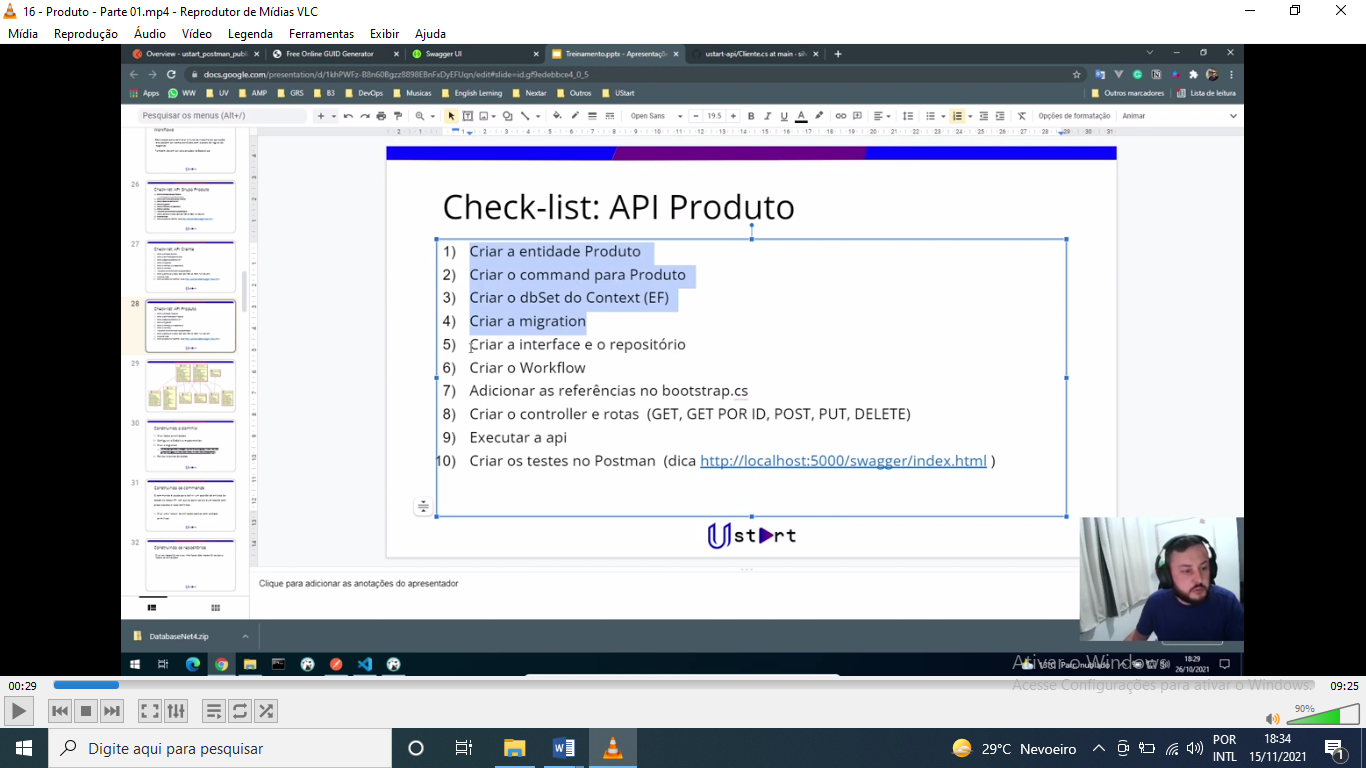
Vídeo: **16 - Produto - Parte 01**

|  |
| --- |
| **Produto** |
| Guid Id  Guid Familia**Id**  Familia Familia  String NomeProduto  String TipoUnidade  Decimal Qtd  Decimal QtdReservada  Decimal Preco  String UrlImagem  Obs.: Não usar \_ porque ao criar a migratin é criado o \_ em automático ficando só no banco de dados. |



Passo 1 e 2:

Domain\Entities\Produto.cs // Usar primeira letra maiúscula p/ cria os atributos que serão criados na tabela Produto no banco de dados

Id = command.Id == **Guid.Empty ? Guid.NewGuid() : command.Id; // Se não vir o GUID e cria novo ou usa o GUID que veio via command.**

Domain\Commands\ProdutoCommand.cs // Cria parte dos atributos todos públicos

Passo 3:

Infrastructure\Context\UStartContext.cs // Criar só comando: public DbSet<Produto> Produtos { get; set; }

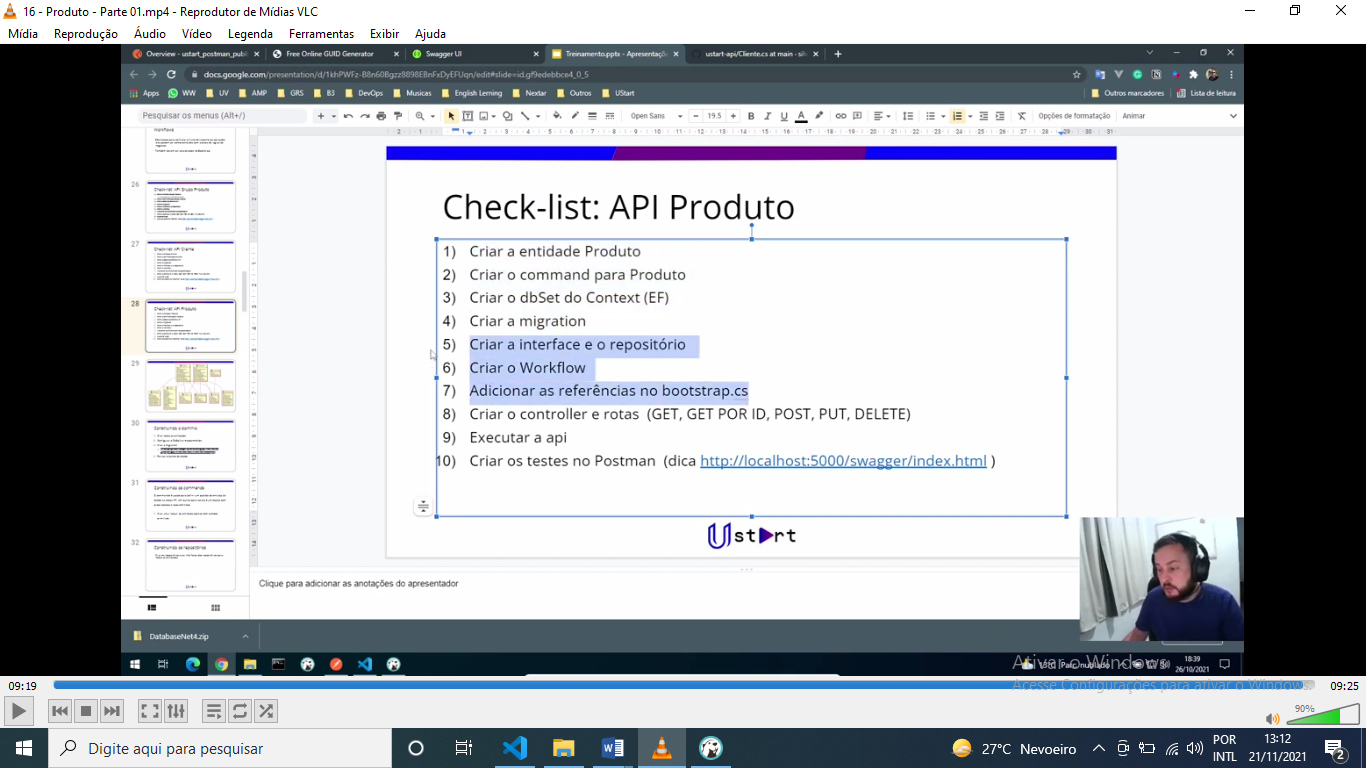
Passo 4:

dotnet ef migrations add produto -c UStartContext --project ../Infrastructure/Infrastructure.csproj //Rodando esse comando p/ gerar dois arquivos.

Infrastructure\Migrations\20211121160038\_produto.cs //arquivo criado em automático, esse arquivo cria o banco de dados.

Infrastructure\Migrations\20211121160038\_produto.Designer.cs //arquivo criado em automático, esse arquivo cria o banco de dados.

Vídeo: **17 - Produto Parte 02**



Passo 5:

Infrastructure\Repositories\ProdutoRepository.cs //Copiar e editar do similar, ver vídeo até 3min22seg porque tem detalhes para buscar na pesquisa a descrição da tabela pai/superior, e também para cria a interface;

Domain\Contracts\Repositories\IProdutoRepository.cs // Copiar similar e **Extrair a Interface** do “ProdutoRepository.cs”

Passo 6:

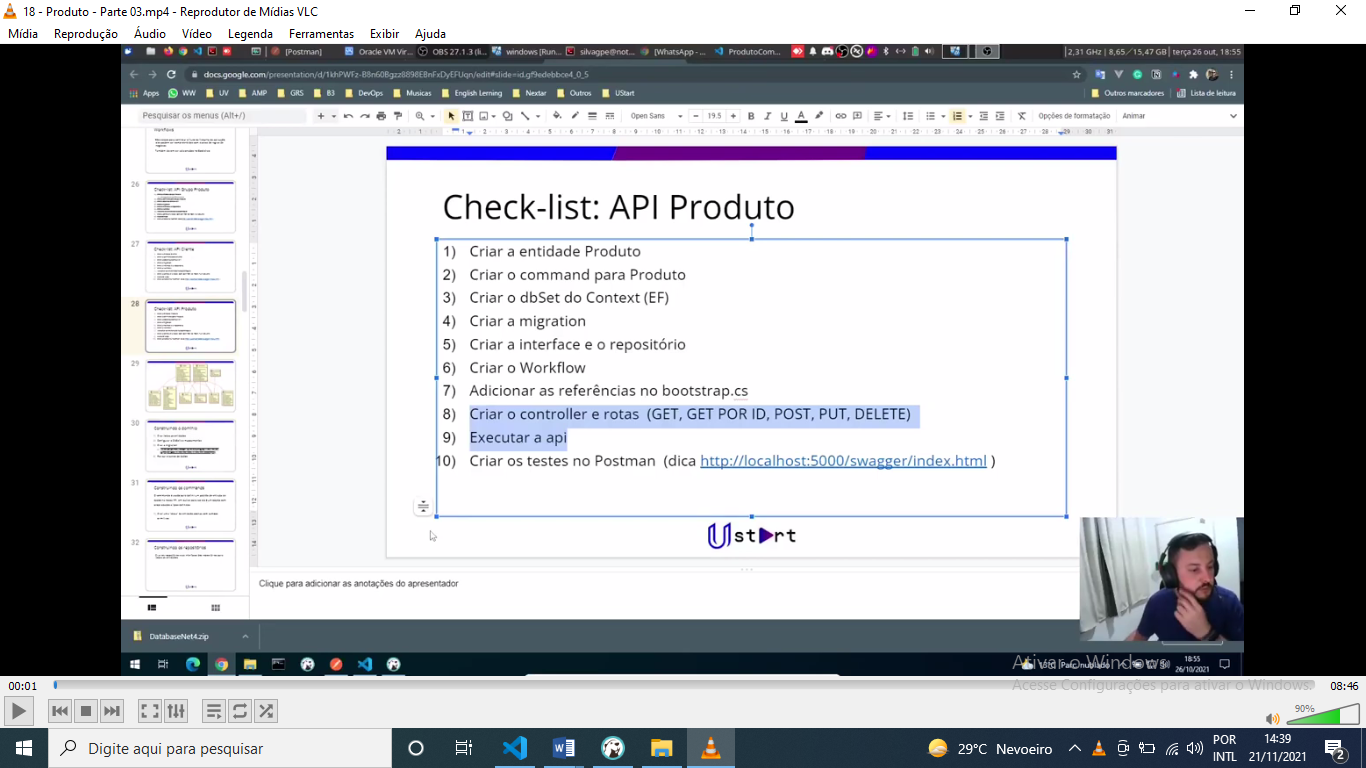
Domain\Workflows\ProdutoWorkflow.cs //Copiar e editar do similar,

Passo 7:

API\Bootstrap.cs //Ir no método RegisterRepositories e adicionar isso: services.AddTransient<IProdutoRepository, ProdutoRepository>();

API\Bootstrap.cs //Ir no método RegisterWorkflows e adicionar isso: services.AddTransient<ProdutoWorkflow>();

Vídeo: **18 - Produto - Parte 03**



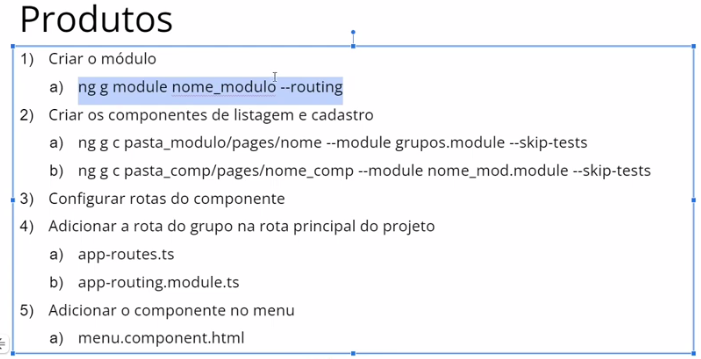
Passo 8, 9 e 10;

API\Controllers\ProdutoController.cs //Copiar e editar do similar,

Testado no Postman e no http://localhost:5000/swagger/index.html

**ANGULAR:**

Vídeo: **26 - Angular - Produto - Parte 01- Revisão**



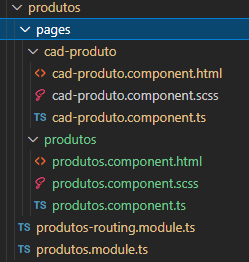
Passo 1:

src\app\produtos\produtos-routing.module.ts // roda o script npx ng g module produtos –routing

Passo 2:

src\app\produtos\produtos.module.ts // roda o script npx ng g c produtos/pages/produtos --module produtos.module  --skip-tests

e o script npx ng g c produtos/pages/cad-produto --module produtos.module --skip-tests



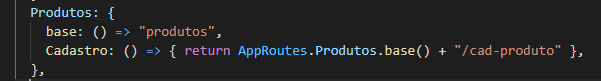
Passo 3: Configurar as rotas internas, rota raiz e duas rotas para o cadastro sendo uma sem parâmetro e outra com parâmetro

src\app\produtos\produtos-routing.module.ts

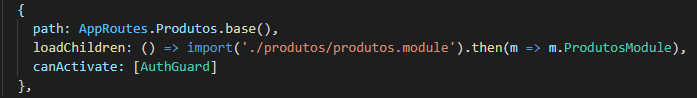


Passo 4:

src\app\app-routes.ts // criar a rota principal “base/raiz” e a rota cadastro

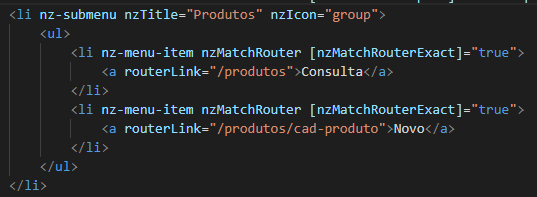


src\app\app-routing.module.ts // carregar o módulo produtos para a navegação web



Passo 5:

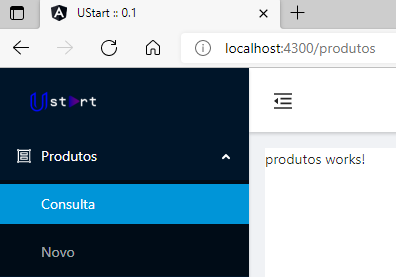
src\app\menu\menu.component.html





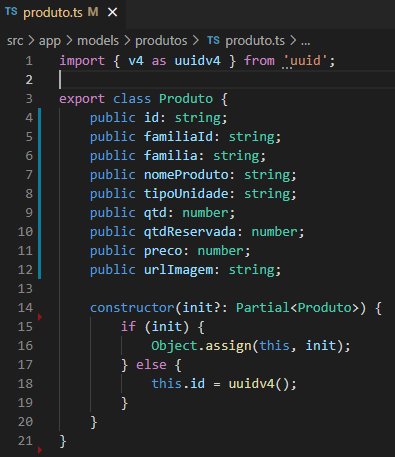
Passo 1 parte dois:

Teste OK



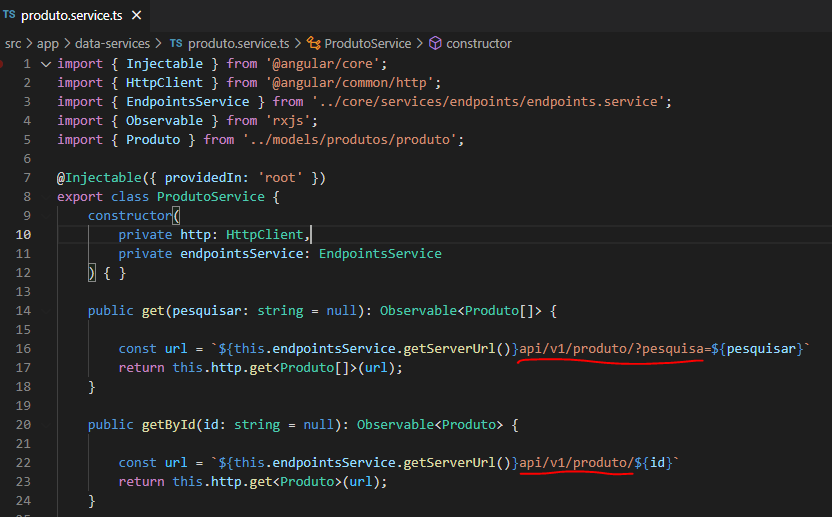
Passo 2 parte dois:

src\app\models\produtos\produto.ts //é o que se espera que volta a API



Passo 3 parte dois:

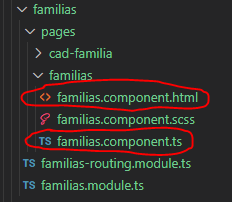
src\app\data-services\produto.service.ts // Chama a **API** passando as pesquisas, inclusão, exclusão....



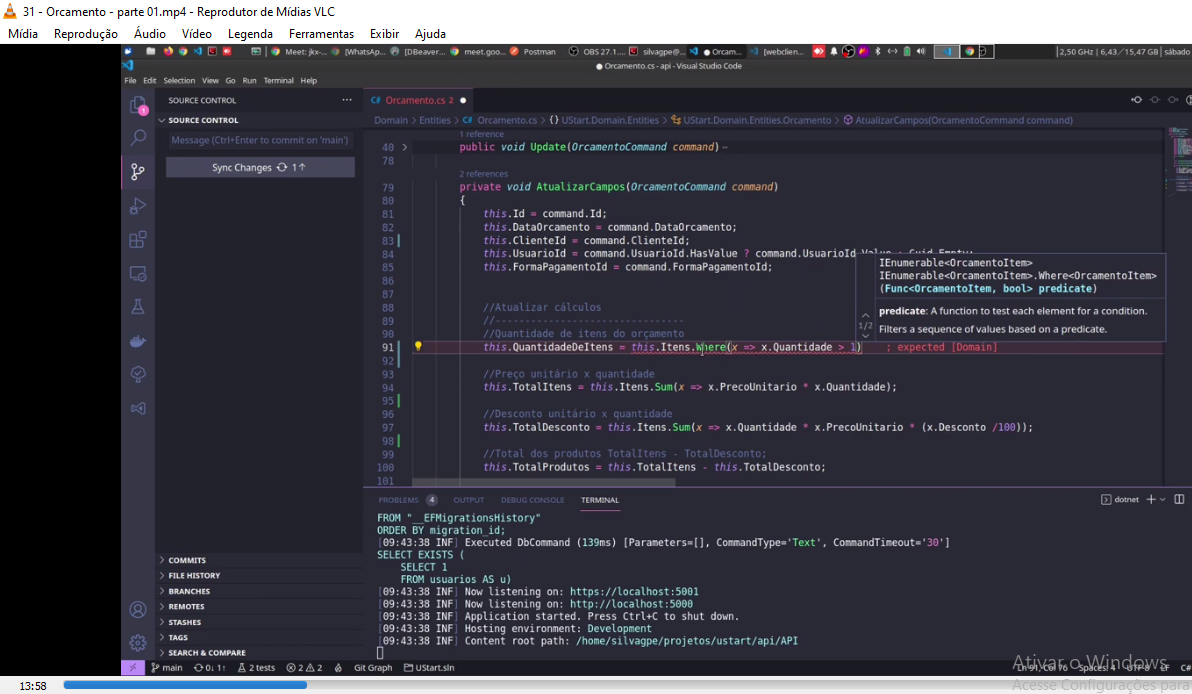
Passo 4 parte dois: Criar os componentes na tela web “desenhar a tela web

src\app\produtos\pages\produtos\produtos.component.html

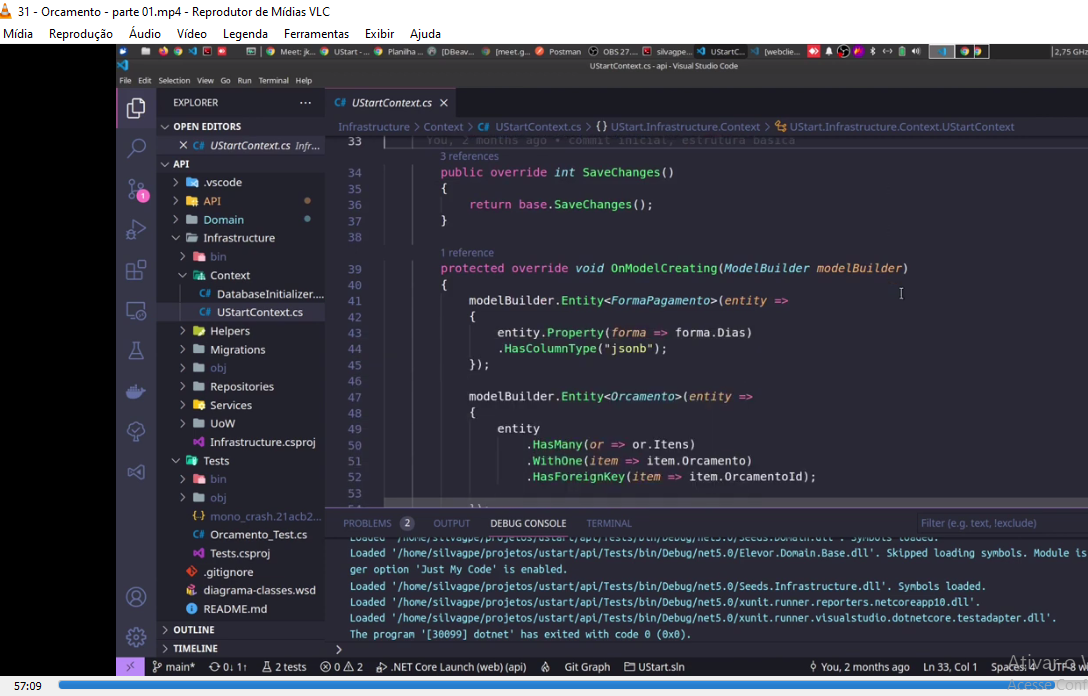
src\app\produtos\pages\produtos\produtos.component.ts

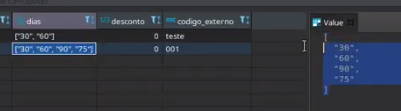
 o “cad-familia” a tela que é chamada ao clicar em novo, editar.

Exemplo de poder quase de consulta de banco de dados, utilizando “ public ICollection<ReservaItem> Itens { get; private set; } “

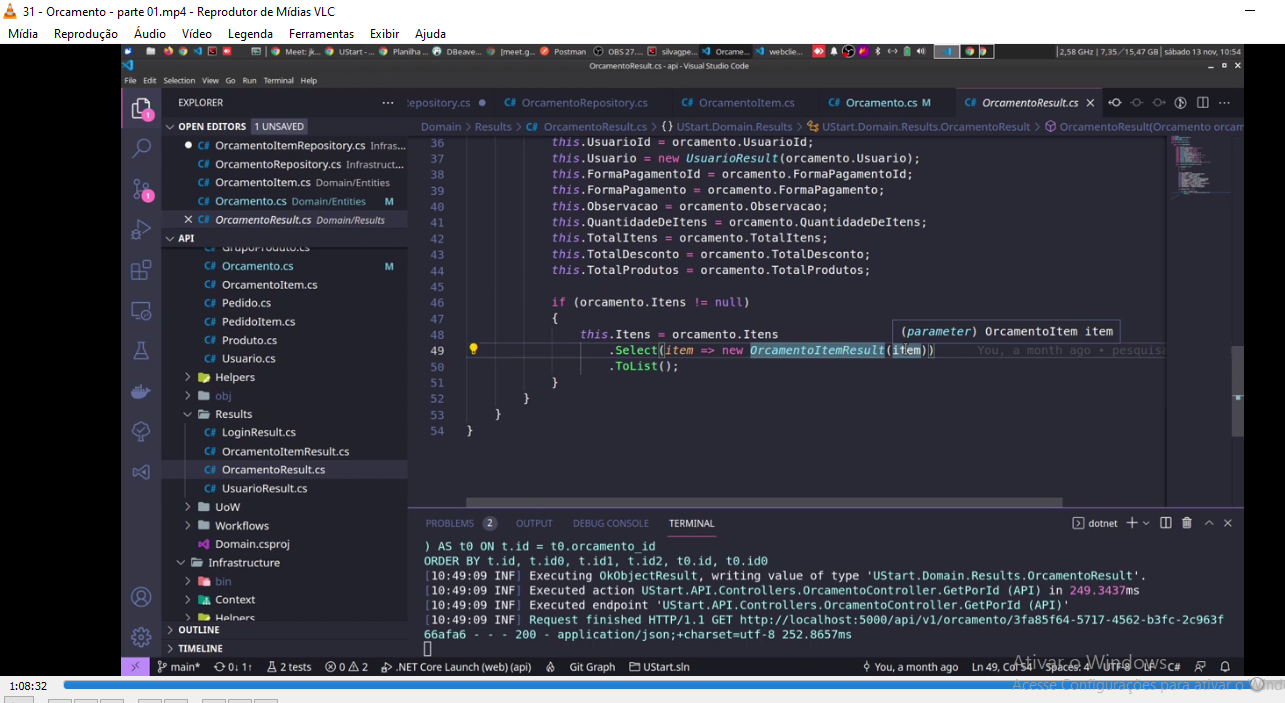


A propriedade dias tem esse implemento a mais, que é ter uma arquivo json no banco de dados.





Novo!!! Results



**~~DOTNET~~**

~~1) Criar a entidade Produto~~

~~Domain\Entities\Familia.cs~~

~~Domain\Commands\FamiliaCommand.cs~~ [~~\\ver~~](file:///\\ver) ~~porque nesse estão os atributos da tabela a ser criada no banco.~~

~~Vídeo 11 - 02:00 a 05:12 - Cria os atributos que serão criados no banco de dados.~~

~~3) Criar o dbSet do contex EF~~

~~Infrastructure\Context\UStartContext.cs~~

~~Vídeo 11 - 05:12 a 06:20~~

~~4) Criar a migration~~

~~Vídeo 11 - 06:20 a - Executar script que vai gerar os atributos na pasta Infrastructure\Migrations que será gerado o banco em automático.~~

~~Infrastructure\Repositories\FamiliaRepository.cs~~

~~Domain\Contracts\Repositories\IFamiliaRepository.cs~~

~~API\Bootstrap.cs~~

~~API\Controllers\FamiliaController.cs~~

~~8) Criar o Controller e Rotas~~

~~API\Controllers\FamiliaController.cs Cria as~~ **~~Rotas~~** ~~de interfase e os [HttpGet, HttpPost]~~

~~Infrastructure\Repositories\FamiliaRepository.cs~~

~~Domain\Contracts\Repositories\IFamiliaRepository.cs~~

~~Domain\Workflows\FamiliaWorkflow.cs~~

~~Vídeo 11 – 19:00 s xx:xx – Criado parte do controller~~

~~Vídeo 13 0:00 até o fim – Criado Update, executado o pull.~~

~~Vídeo 14 0:00 até o fim – Criado Delete, executado o pull.~~

~~2) Criar o command para Grupo Produto~~

~~Domain\Commands\GrupoFamiliaCommand.cs~~

~~Domain\Entities\GrupoFamilia.cs~~

**~~Vídeo - 12~~** ~~0:00 a 07:03 – O Command passa os dados para a Entities que vai salvar no banco de dados.~~

~~6) Criar Workflow~~

~~Domain\Workflows\ GrupoFamiliaWorkflow.cs~~

~~Domain\Contracts\Repositories\IGrupoFamiliaRepository.cs~~

~~Infrastructure\Repositories\ GrupoFamiliaRepository.cs~~

~~API\Controllers\ GrupoFamiliaController.cs~~

~~API\Bootstrap.cs~~

~~Executar a Depuração partindo do Postman é mostrado em 28 minutos do vídeo 12~~

**~~Vídeo - 12~~** ~~07:03 até o fim – Foi criado os métodos de inserir no banco.~~

**~~ANGULAR:~~**

**~~Vídeo: 20 - Angular - Grupos - Parte 01 - Rotas~~**

~~npx ng g module familias –routing - Cria a pasta e dois arquivos p/ configuração de~~ **~~rotas~~** ~~e de módulos.~~

~~npx ng g c familias/pages/familias --module familias.module  --skip-tests - Lista que vai usar p/ pesquisar~~

~~npx ng g c familias/pages/cad-familia --module familias.module --skip-tests - Editar os registros~~

~~Editar as~~ **~~Rotas~~**~~:~~

~~src\app\familias\familias-routing.module.ts //~~**~~Rota~~**~~, rotas internas do componente/objeto/produto~~

~~src\app\app-routes.ts //~~**~~Rota~~** ~~raiz, antes da Rota familia familias-routing.module.ts~~

~~src\app\menu\menu.component.html //Adicionar no menu do navegador/web~~

~~src\app\app-routing.module.ts //Adiciona as~~ **~~rotas~~**

[~~https://ng.ant.design/components/icon/en~~](https://ng.ant.design/components/icon/en) ~~//Link para pegar imagens de ícones para utilizar no grupos-routing.module.ts~~

**~~Vídeo: 21 - Angular - Grupos Parte 02 - Tela consulta grupos~~** ~~- criar as telas visuais~~

~~Criar tela no navegador:~~ [~~Button | NG-ZORRO (ant.design)~~](https://ng.ant.design/components/button/en)

~~src\app\familias\familias.module.ts //Onde adiciona as importações/biblioteca dos botão/tabela...~~

~~src\app\familias\pages\familias\familias.component.html //Começar a construir a página web em 04:30 até o fim~~

**~~Vídeo: 22 - Angular - Grupos - Parte 03 - Serviço~~**

~~src\app\models\familias\familia.ts // Criar o models que é a comunicação com o servidor “uuidv4” gera um novo guid~~

~~ficou o debugger; nessa classe caso apresenta erro ver se tem que tirar.~~

~~src\app\data-services\familias.service.ts //Classe para comunicar com a API~~

~~src\app\core\services\endpoints\endpoints.service.ts //o endpoint devolve qual é a URL de servidor e a URL de login~~

**~~23 - Angular - Grupos - Parte 04 - Eventos e binds~~**

~~Explicação~~

~~src\app\familias\pages\familias\familias.component.ts //Cria os eventos/ação~~

~~[] ao utilizar colchetes em [nzTitle] é informado que vai só receber comando, se tiver [(nzTitle)] se tiver colchetes e parênteses vai receber e disparar comandos. Exemplo no vídeo 23 em 3min40seg até 5min35seg.~~

**~~24 - Angular - Grupos - Parte 05 – Consulta~~**

~~Explicação~~

**~~25 - Angular - Grupos - Parte 06 - Cadastro~~**

~~Explicação~~

~~src\app\menu\menu.component.html // Cria o menu da tela web~~

~~"email":"suporte@email.com",~~

~~"senha": "123456"~~