Criando eShopEnterprise

Setup

Crie uma solution em branco – blank solution

Dentro da pasta do projeto, crie a pasta “src” onde ficará nosso código fonte.

Dentro da pasta “src” vamos criar novas pastas chamada, “building blocks” e “services”.

* Building blocks: Ficará nosso fonte onde será usada em demais projetos, sejam apis ou classlib.
* Services: ficará nossas apis.

Dentro de Building Blocks, crie uma pasta lógica “Core” e crie um projeto do tipo Class Library no .net core, salvando na pasta src\building blocks, com o nome eShopEntrerprise.Core.

Essa mesma estrutura de pastas físicas, vamos criar como pastas lógicas na nossa solução.

Na nossa pasta lógica service, criamos a pasta lógica Identidade vamos criar um novo projeto ASP.NET Core Web Application, chamada eShopEnterprise.Identidade.API na versão 3.1 do .net core, vamos salvar na nossa pasta lógica src\service.

Vamos criar as demais apis, seguindo a criação da pasta lógica e namespance a seguir:

Carrinho

Catalogo

Cliente

Pagamento

Pedido

API Identidade

Instalando e configurando IdentityFrameworkCore

Insatale os seguintes pacotes:

* Microsoft.AspNetCore.Authentication.JwtBearer
* Microsoft.AspNetCore.Identity.EntityFrameworkCore
* Microsoft.AspNetCore.Identity.UI
* Microsoft.EntityFrameworkCore
* Microsoft.EntityFrameworkCore.Tools (migrations)
* Microsoft.EntityFrameworkCore.Abstractions
* Microsoft.EntityFrameworkCore.Relational
* Microsoft.EntityFrameworkCore.SqlServer

Criando contexto do entity

Cria a pasta Data, crie uma classe ApplicationDbContext que herde de IdenittyDbContext, e crie o construtor que recebe DbContextOptions, passando para clase base.

Esses options (DbContextOptions) é que recebrá nossas configurações que irão no startup.

Startup.cs

Adicionaremos, em ConfigureServices, no começo a configuração do nosso contexto:   
services.AddDbContext<ApplicationDbContext>(options => options.UseSqlServer(Configuration.GetConnectionString("ConnectionString")));

Adicionaremos, nossa configuração do identity, logo após o nosso dbcontext:

services.AddDefaultIdentity<IdentityUser>()

.AddRoles<IdentityRole>()

.AddEntityFrameworkStores<ApplicationDbContext>()

.AddDefaultTokenProviders();

Em configure, além do app.UseAuthorization(), vamos adicionar o app.UseAuthentication() ficando assim:

app.UseAuthentication();

app.UseAuthorization();

Criando nossa base, com entity framework e migration

* Add-Migration Initial: onde initial é o nome da nossa migration
* Update-database: para criar base e tabelas do identity

Criação de Model e Controllers para Registrar novo usuário e reliazar login

Vamos criar nossas models, e controllers

Swagger

Instale o seguinte pacote:

* Install-Package Swashbuckle.AspNetCore -Version 5.33