AuthECAPI Project Overview

Wednesday, March 26, 2025 5:23 PM

This project is the .NET core web API using ASP.NET Core Identity for user authentication and Entity Framework Core (EF Core) for database access.Let's go folder by folder

🌐 **Controllers** (Entry Point for API Calls)

This folder cntains the API endpoints that handle incoming HTTP requrests

1. IdentityUserEndpoints.cs:
   - Handles user authentication and registration
   - Routers like /signup and /signin process user registration and login, respectively

     Data flow:
     a. API receives HTTP request (like POST/signup)
     b. Calls UserManager to create or authenticate users
     c. Returns responses like JWT token or error messages

2. OrderEndpoints.cs (Possibly for managing orders)
   a. Handles order-related endpoints if your project involves orders.
3. WeatherForecastController.cs
   a. This is usually a default controller created by .NET templates for testing purposes

☑ **Data Flow:**

- Incoming HTTP requests → **Controllers** → Call services or UserManager → Return response.

⚙ **Extensions** (Helper Methods and Configurations)

Extensions provide reusable methods to extend built-in classes or simplify configurations

1. AppConfigExtensions.cs
   Handles general app configurations like middleware,CORS ,etc.
2. EFCoreExtensions.cs
   Configures Entity Framework CORE(like setting up the database connection)
3. IdentityExtensions.cs
   Configures Identity settings (like password policies,roles,etc)
4. SwaggerExtensions.cs
   Adds Swagger/OpenAPI support to document your API

☑ **Data Flow:**

- Extensions are called in Program.cs to set up services when the app starts.

📜 **Migrations** (Database Schema Management)

Contains EF Core migrations to handle database schema changes

a. **InitialMigration.cs**:
   i. Creates the initial database structure.
b. **20250314030310 3_NamesAddedInUserEntity.cs**:
   i. Updates the AppUser entity, adding new columns like FullName.
c. **AppDbContextModelSnapshot.cs**:
   i. Represents the latest state of the database schema.

☑ **Data Flow:**

- Run dotnet ef database update to apply migrations → **EF Core** updates the database.

Models (Data Structures)

Defines the data models that represent tables in the database

1. AppDbContext.cs
   Manages the database connection and maps models to tables
2. AppSettings.cs
   Holds app configuration settings like the JWTSecret used for token generation
3. AppUser.cs
   Extends IdentityUser to add custom field like FullName

Data Flow:

- AppDbContext handles data persistence by interacting with the database.
- AppUser maps to the AspNetUsers table.

appsettings.json (Configuration File)

Stores app configuration like database like connection strings and JWT secrets.

Data Flow:

- Program.cs reads settings from this file when the app starts.

🚦 **Program.cs** (App Entry Point)

- Configures the app and sets up services like UserManager and DbContext.

☑ **Data Flow:**

- Runs once at app startup to initialize everything.

🔍 **How Data Passes Through**

1.Request:

- The Client (e.g a frontend app or postman) sends an HTTP request.

2.Controller:

- The corresponding controller method handles the request (/signup -> CreateUser).

3.Services and Identity

- Calls **UserManager** to create or authenticate users
- Users **AppDbContext** to query the database

4.JWT Creation (For Login)

- If login success, the app generates a JWT and returns it to the client

5.Response:

- The controller returns response (e.g., the JWT or a success mesage)