



Session 10

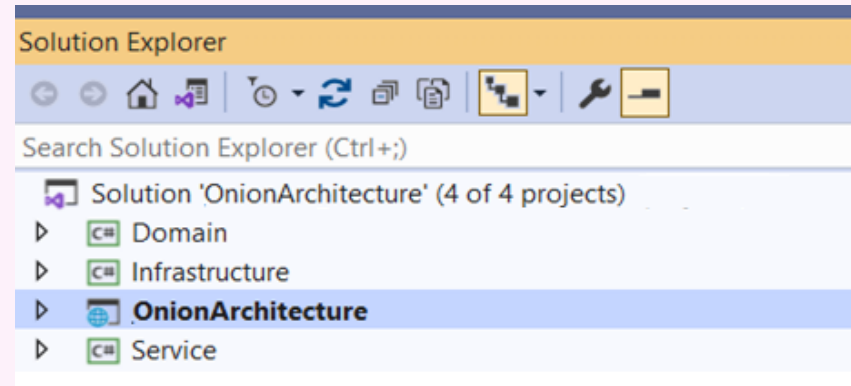
Onion Architecture in ASP.NET Core - II

Session Overview

- Explain project structure for Onion Architecture
- Describe the process of implementation of the four projects – Domain layer, Infrastructure layer, Service layer, and Onion Architecture Web API

Project Structure of Onion Architecture

Figure 10.1: Layers of Onion Architecture



Domain Entities Layer (1-2)

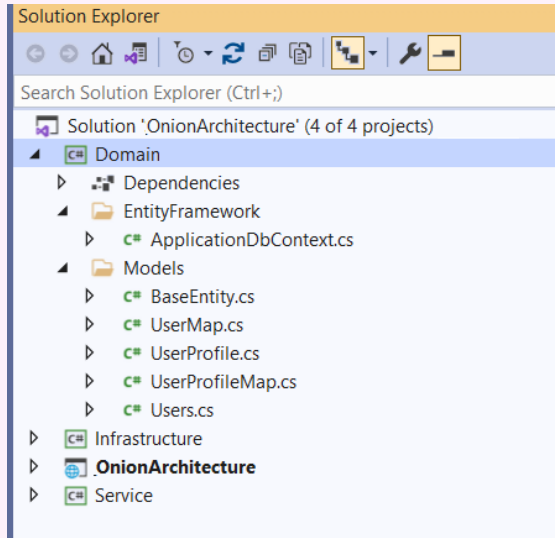


Figure 10.2: Domain Entity Layer

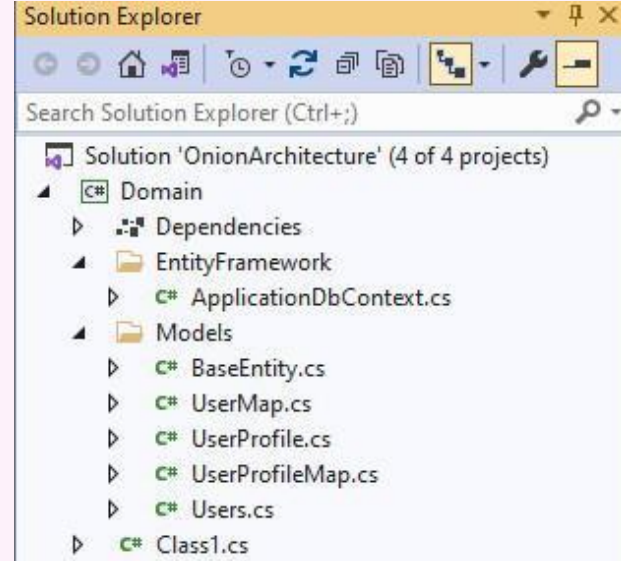


Figure 10.3: Model and EntityFramework Folders

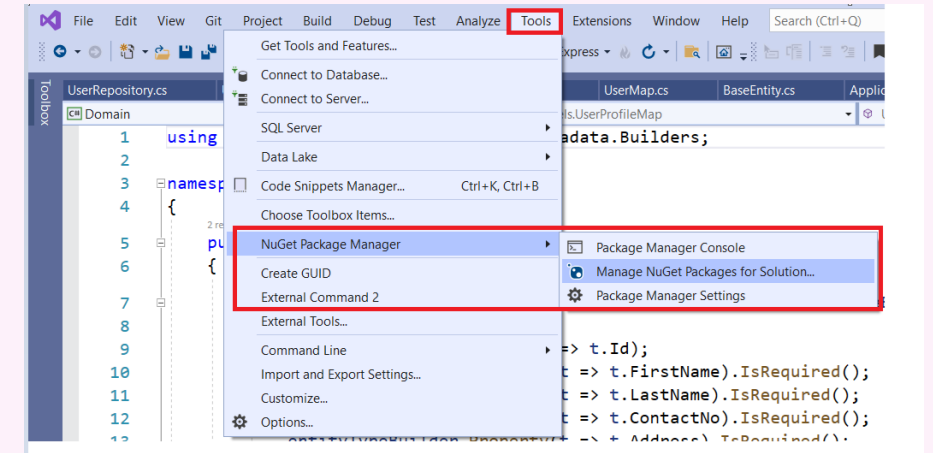


Figure 10.4: NuGet Package Manager

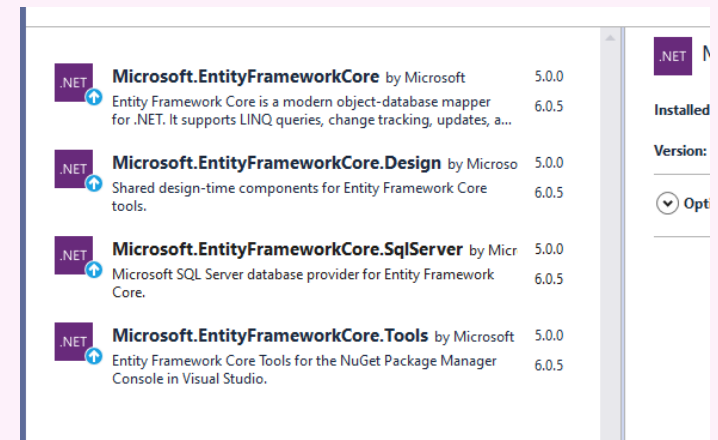


Figure 10.5: EntityFramework Projects

Domain Entities Layer (2-2)



Figure 10.6: Package Manager Console

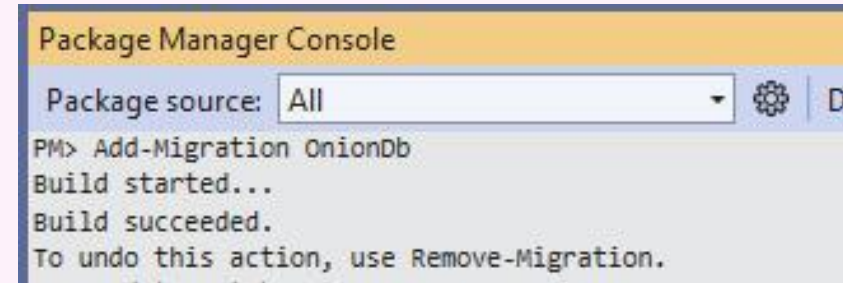


Figure 10.7: Build Successful Message

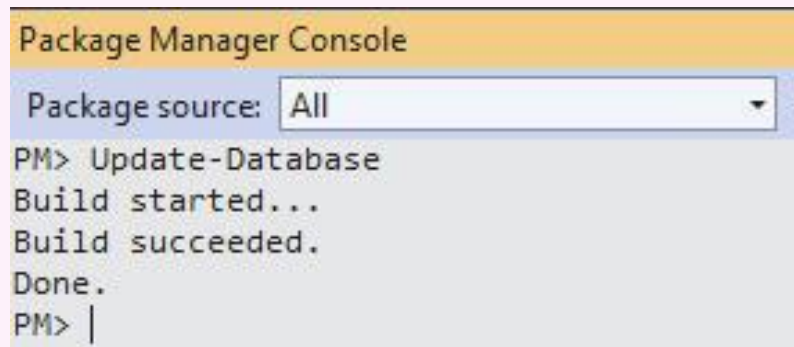
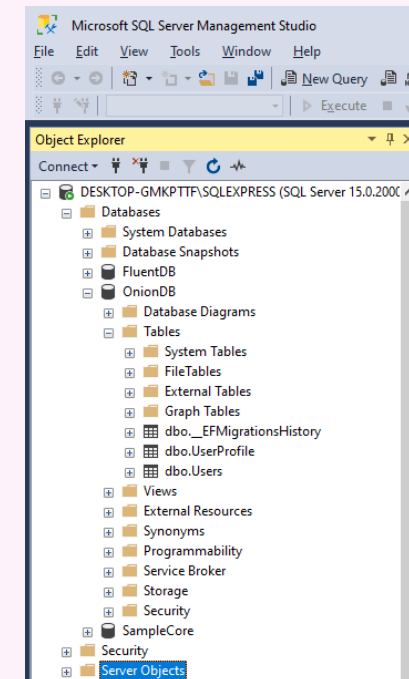
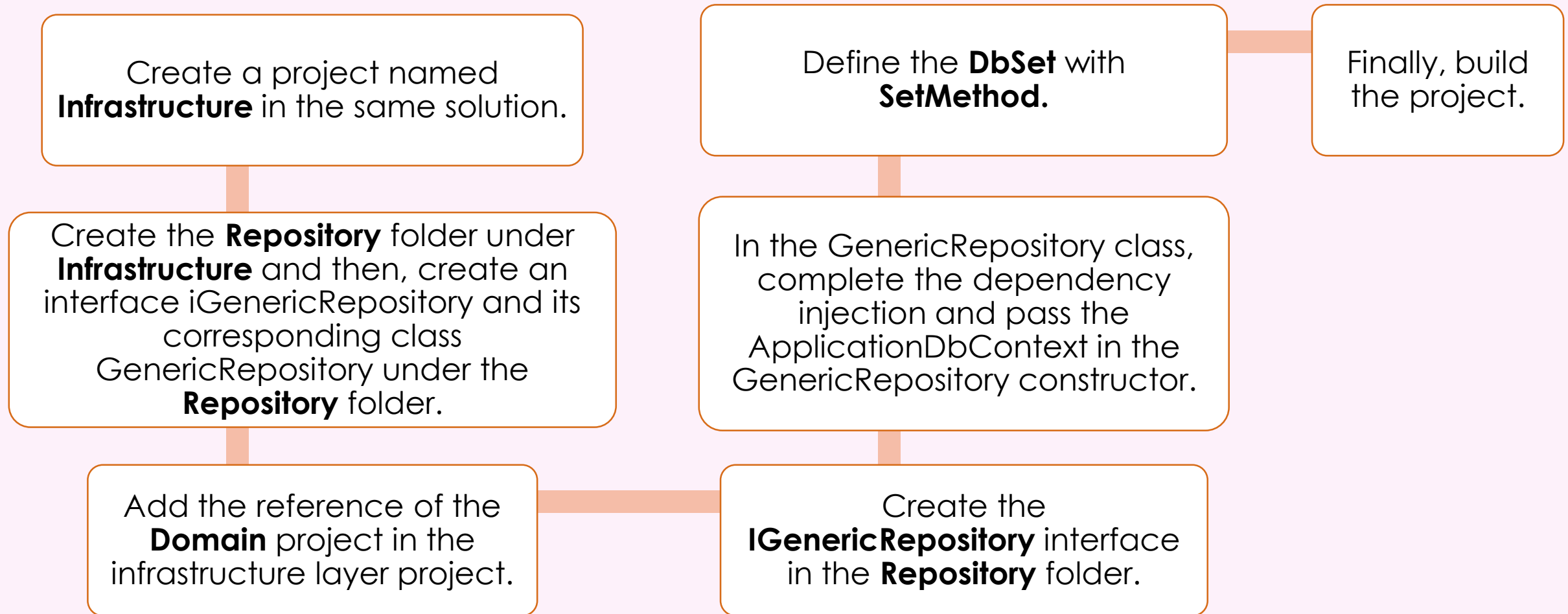


Figure 10.8: Update Database

Figure 10.9:
Database and
Tables



Infrastructure Layer



Service Layer

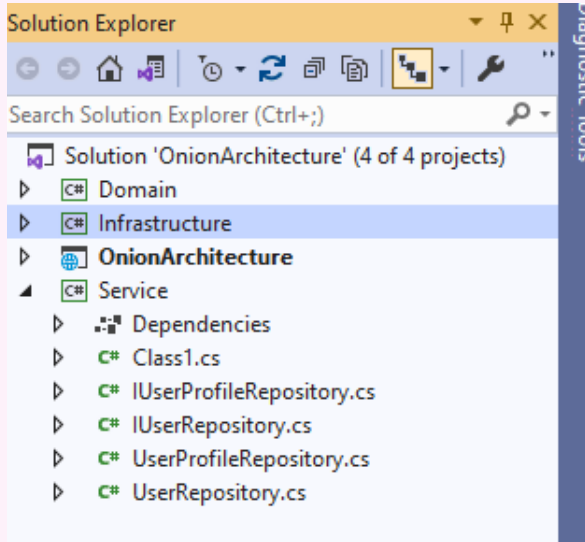


Figure 10.12: Service Layer

Add the reference of the domain layer and infrastructure layer in the service layer.

Create the interface **IUserRepository** in the service layer.

Create the interface **IUserProfileRepository** in the service layer.

Create the class **UserProfileRepository** in the service layer.

Create the class **UserRepository** in the service layer with code.

Onion Architecture Web API (1-2)

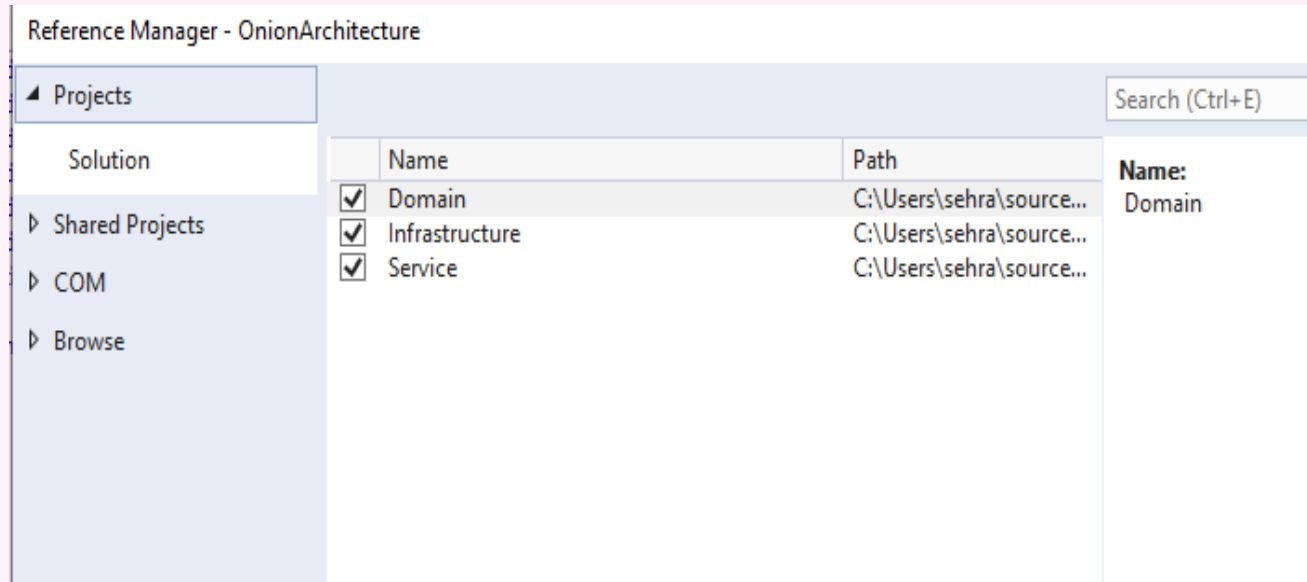


Figure 10.13: Onion Architecture

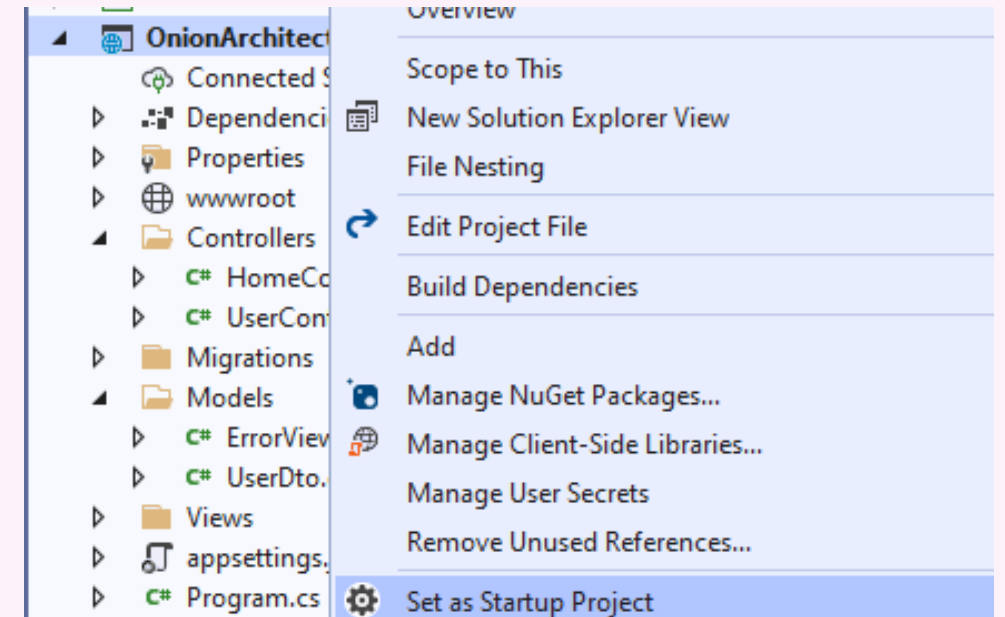


Figure 10.14: Set as Startup Project

Onion Architecture Web API (2-2)

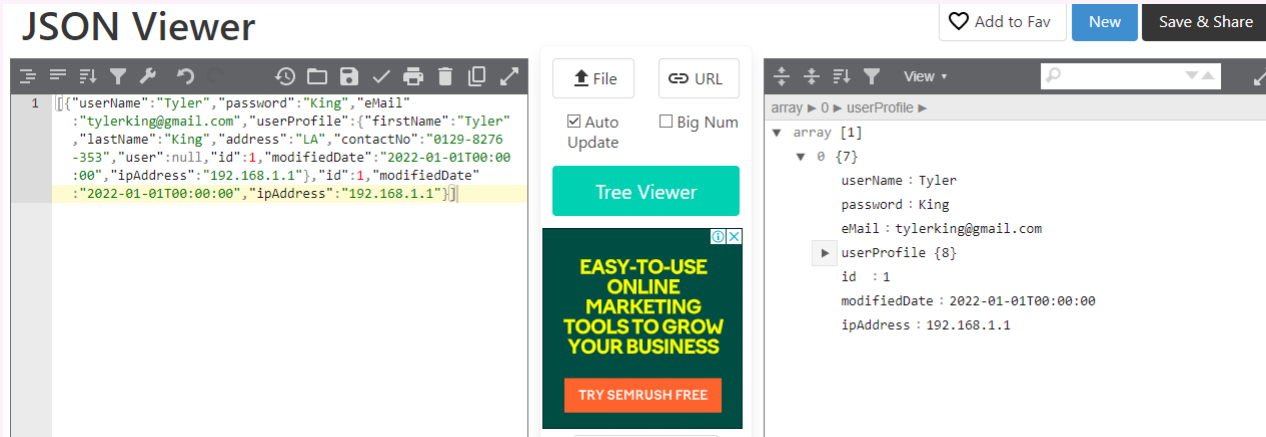


Figure 10.15: JSON Viewer

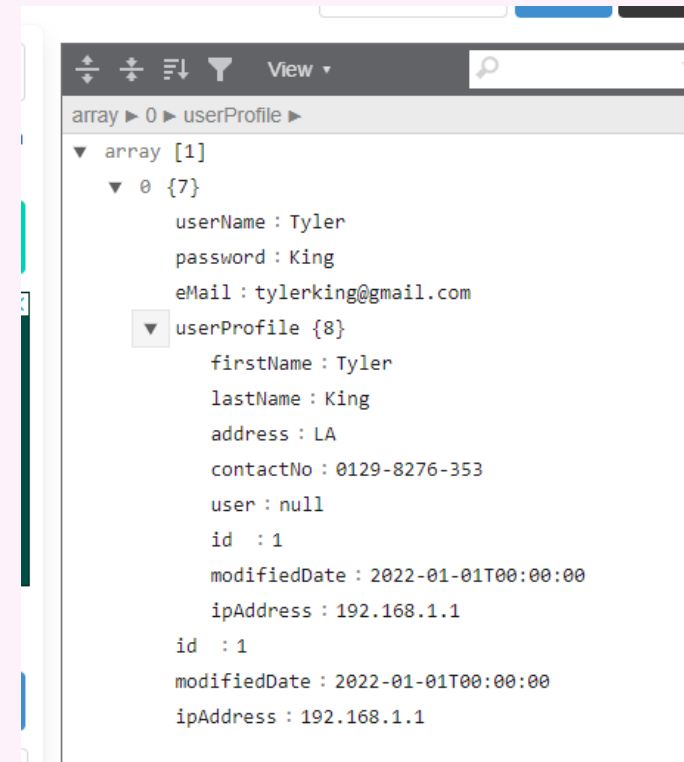


Figure 10.16: User Profile

Summary

- ✓ Four different layers of the Onion Architecture are domain entities layer, repository layer, service layer, and UI layer.
- ✓ For each of these layers, a corresponding project must be created to implement the Onion Architecture in an ASP.NET Core application.
- ✓ Domain entities layer contains the class library, POCO, and configuration classes. It also helps in creating database tables.
- ✓ The repository layer implements the interface for the generic repository class. It also contains the DbContext class.
- ✓ The service layer contains the business logic and user interfaces.
- ✓ The UI layer is the entry point of the program and the exterior layer of the Onion Architecture.