Ecommerce Solution

The solution is a web Api based on .NET and C#. It implements clean architecture principles as it implies strict rules and structure and encourages clean code and the following of all SOLID principles. MSSQL was chosen as the database for this project, alongside with EntityFrameworkCore, which it has excellent integration with and is intuitive and native for the .NET ecosystem. The app follows the repository pattern and that allows decoupling from both the Database and the ORM. The application authentication is implemented with JWT, and although it brings some complexity at implementation phase, later it provides a lot of comfort, security and flexibility. Caching is implemented to reduce the DB traffic and response time as well. Currently it is achieved using the ASP.NET caching mechanism, but ideally this caching would be centralized using tools such as Redis. Serilog was preferred for the application’s logging and Automapper to support and simplify object-mapping. For testing I prefer NUnit, alongside with Moq and InMemory.  
The front-end uses React. Although other frameworks and libraries provide more things out of the box and that helps for faster development, React is the better choice when the application is planned to grow and get bigger and more complex. With the help of redux toolkit, which significantly reduces boilerplate code I decided that this is how it is going to be build. It is built purely on functional components and for some components I used Material UI to introduce a better and smoother styling.  
A lot of functionalities are yet to be added, mostly on the front-end, some of them being – cart interactions, notifications, filtering system, pagination, liking, rating etc.