**Quote Quizzes**

1.FrontEnd

- Summary

For create client side of the app is used framework REACT JS. Following the latest tendention, mostly functional components are used. However, a couple of class components were added to show knowledge about them. Due to the time limit, there are no fancy components like spinners, loaders, errors, etc.

Context are added to escape props drilling.

Users can register with username, email and password.

Upon registration they will receive JWT token and will be automatically authenticated (no need to login); Authentication persists on page refresh.

Users can create quizzes, play them and check their results. Owners of the quizzes can delete them (I apologize that couldn’t implement update in the required time).

-Libraries

Font Awesome library is installed to make app look better

React-router – dom is used for routing

2.BackEnd

- Summary:

ASP NET web api for server. Maintains different requests (ape) and returns proper response. It maintains authentication and authorizations. CORS policy is applied. On startup creates database. Server is divided mainly in three points (Data, Services, Web);

* Database

Entity framework core is used for communication with MSSQL server database. Lazy loading is enabled.

Registration of the AppDbContext is implemented through DesignTimeDbContextFactory. It handles request to database, performs sorting, store and gets data. Couple of abstract classes and interfaces are used to help storing , deleted and managing data.

* Services

We have a couple of services they are implemented as interfaces and inherited by classes. All the login is there. They are called by the controllers and make requests to the database.

* Web.

The controllers are there. They are cleared of the login and are used to call services and to return the response.