

Documentation assignment 2

Lacatus Arpad-Alex

Group 30433

1. Architecture

This part of the application focuses on the frontend. As such I used as framework Angular. Angular is a platform and framework for building single-page client applications using HTML and TypeScript. Angular is written in TypeScript. It implements core and optional functionality as a set of TypeScript libraries that you import into your applications. Some key features of Angular are data binding, routing, and efficient use of reusable components.

In order to make the skeleton and face of our application we use html and scss. These two used together give us the main part of our frontend. With the use of html we were able to create buttons , forms , divisions, lists that with the help of scss were styled according to our needs. Some html elements were also attributed dynamically through typescript. For example the questions and answers page were filled dynamically with dummy data provided in a typescript file (to note that the dummy data is saved in an array , and this data will be replaced eventually with the data from the server).

This portion of the app is divided into 2 main parts heading and content. The head is always the same for our app and consists of the logo, which is a hyperlink to the home page, a search bar and a userprofile link. The second part is the content which through the help of routing can have many forms. These forms are split in mainly 3 user (login, signup, userprofile), question (questions, addquestion), answer (answers, addanswer).

2. Routing

The routing is handled with the help of router-outlet which through the function routelink or href routes our app to the desired page mentioned in app.module. With the help of Angular that handles the routing part by providing the class router the URL is change to the desired one and the content is mapped to the component that needs to be shown for the specific URL.

Here we can see the current routes in our application:

```
9 import {AnswersPageComponent} from "../pages/answer/answers/answers-page.component";
10
11 const routes: Routes = [
12   { path: '', component: LoginPageComponent },
13   { path: 'login-component', component: LoginPageComponent },
14   { path: 'signup-component', component: SignupPageComponent },
15   { path: 'user-component', component: UserPageComponent },
16   { path: 'question-component', component: QuestionPageComponent },
17   { path: 'questions-component', component: QuestionsPageComponent },
18   { path: 'answer-component', component: AnswerPageComponent },
19   { path: 'answers-component', component: AnswersPageComponent }
20 ];
```

In order to navigate the app we can not that we have the following routes linked to out content:

Login -> questions

-> signup

signup -> questions

-> login

Questions -> answers

-> addquestion

Answers -> addanswer

Using the heading stay always the same we can go to the homepage(login) and userprofile.

