**Documentation assignment 2**

Name: Stan Andreea

Group: 30433

**1. Architecture of the Frontend**

In order to implement this part of the assignment, I have used the Angular framework. The Angular framework follows a component-based architecture, which encourages the modular design of applications. This approach makes it easier to maintain and scale the application over time.

The project consists of several components, including the login component, the signup component, the logout component, the home component, the questions-page component, and the question-detail component. Each component has its own responsibilities and encapsulates a specific part of the application's functionality.

* The login and signup components handle user authentication and registration, respectively.
* The logout component is used to handle users logging out of their accounts, removing them from the current sessionStorage.
* The home component displays a welcome message to the user and provides links to other parts of the application, links which depend on wheter the user is logged in or not.
* The question list component displays a list of all the questions available in the application, and a form for the user currently logged in to be able to ask a new question.
* The question detail component displays the details of a particular question, including answer. Here there is also the functionality for the user logged in to be able to answer a question
* The profile component is used for displaying the information of the user currently logged in.

O imagine care conține text, captură de ecran, software

Descriere generată automat

The application also makes use of several services, including the Authentication service, AuthenticationGuard service, HTTP interceptor service. and the Question service. The use of services ensures that the business logic of the application is separated from the UI logic, making it easier to test and maintain.

* The Authentication service handles user authentication, including methods for the login and registering of an user, for checking if a user is logged in or for retrieving some data about the user currently logged in.
* The Question service is responsible for retrieving and managing questions and their associated data.
* The AuthenticationGuard service manages the access authorization to certain links.It uses an event called canActivate. The canActivate is like a constructor. It will be called before accessing the routes. The canActivate has to return true to access the page. If it returns false, the user cannot access the page.
* O imagine care conține text, captură de ecran, Font

  Descriere generată automatThe HttpInterceptor service is used to intercept all HTTP requests sent by the Angular application. The service checks if the user is logged in and adds authentication headers to the request if the user is authenticated. If the user is not authenticated, the service passes the request without modifying it.

In the Angular project, there are several models/interfaces defined to represent the data objects used throughout the application. These models/interfaces provide a structure for the data objects and help maintain consistency in the way data is passed between components and services.

There are three models/interfaces in this project: User, Question, and Answer.

* The User interface defines the properties of a user object, including id, username, password, and email. This interface is used to create user objects and ensure that each user object has the necessary properties.
* The Question interface defines the properties of a question object, including id, authorId, title, description, and answers. The answers property is an array of Answer objects. This interface is used to create question objects and ensure that each question object has the necessary properties.
* The Answer interface defines the properties of an answer object, including id, questionId, authorId, text. The author property is a string representing the username of the user who posted the answer. This interface is used to create answer objects and ensure that each answer object has the necessary properties.

O imagine care conține text, captură de ecran, Font, software

Descriere generată automat

In this project, HTML, CSS, and TypeScript were to create a dynamic and responsive web application.

HTML is the standard markup language used to create the structure and content of web pages. In this project, HTML was used to define the layout and structure of the various pages, as well as to create forms and tables to display data.

SCSS is a style sheet language used to describe the presentation of a document written in HTML. CSS was used extensively in this project to style the various elements of the web pages, including fonts, colors, backgrounds, borders, and positioning.

TypeScript is a superset of JavaScript that adds optional static typing and other features to the language. In this project, TypeScript was used to write the business logic and data handling code for the application, including services, components, and models.

O imagine care conține text, captură de ecran, software, Pagină web

Descriere generată automat

**2. Routing**

The routing module in an Angular application maps specific URLs to specific components. In this project, the AppRoutingModule is used to define the application routes.

The RouterModule and Routes are imported from the @angular/router package. The Routes array defines the different routes in the application, with each route having a path and a corresponding component.

The path property in the route specifies the URL path that should match with the corresponding component. If the URL matches with the path, the corresponding component is rendered.

In this project, the AuthGuard is also used to restrict access to certain routes based on the user's login status. For example, the QuestionDetailComponent is guarded by the AuthGuard, which means that only authenticated users can access the component.

Once the routes are defined, they are exported as an AppRoutingModule class which is then imported in the app.module.ts file to be used by the application. This allows the application to navigate to different pages by clicking on links or typing URLs in the browser.

O imagine care conține text, captură de ecran, Font

Descriere generată automatO imagine care conține text, captură de ecran, software

Descriere generată automat