

Stack Overflow

Intro

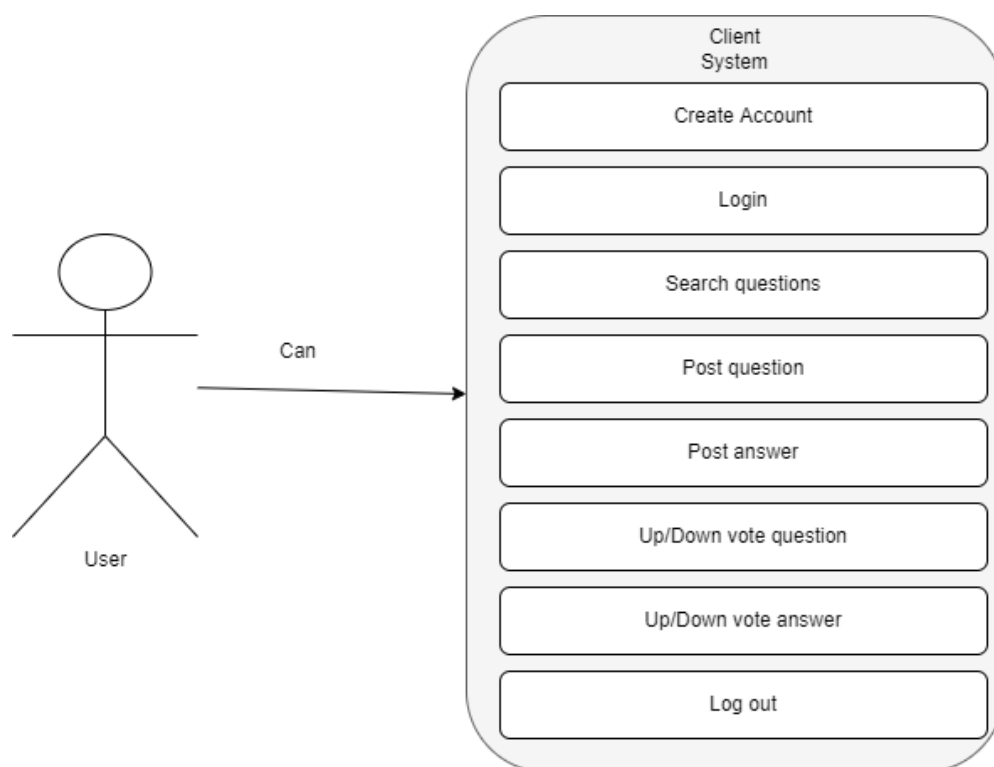
Now that we have a good and stable backend, it is time for the frontend to be our major focus. Since this is the part that people will interact with, it is essential to achieve a functional and clean result.

Tech Stack

The technology that makes this app a reality is angular. Since this is a web app, this allows for easy cross platform development as most modern devices now have the needed requirement, a web browser, to run this application.

Use Case Example

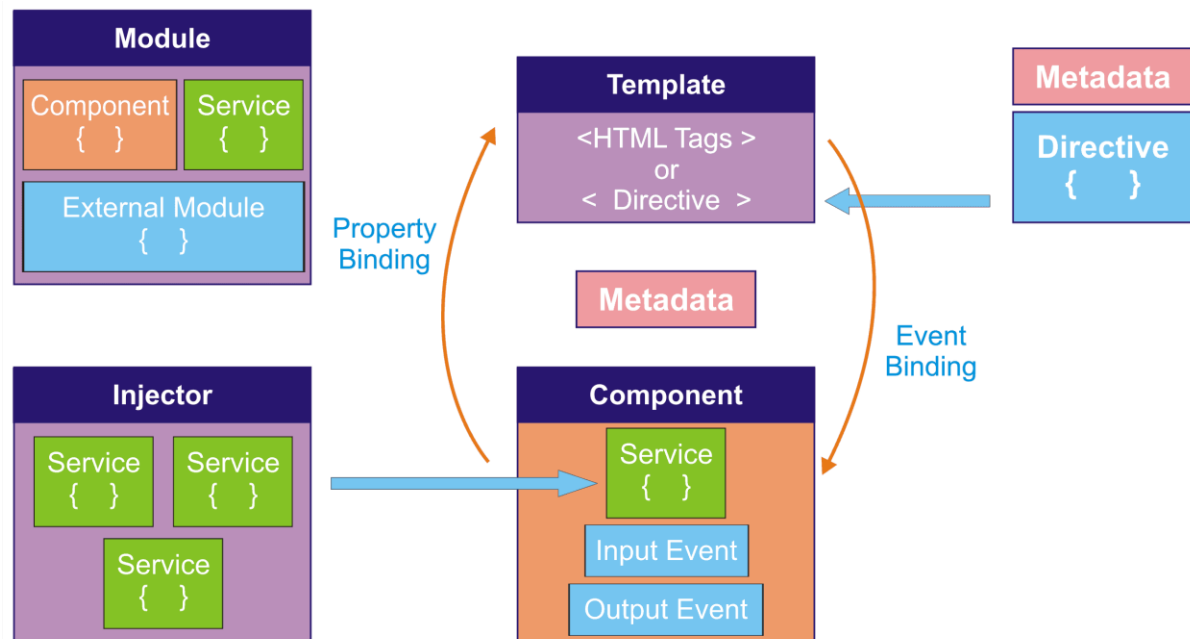
In the following diagram we will represent some use cases, followed by a description of the login use case:



I will describe the way that the login use case works. Once the client accesses the login page of our app, he will be presented with a form that requires an email and password as input. Once the form is completed the login button shall be pressed and the front end will make an API call to the backend to try and create a session for the user. If the user completed the form with incorrect information a message will be displayed detailing, why the login failed. If the login is successful, the user will receive a session token from the API call and he will be redirected to the home page.

Architecture

We focused on using a component-based architecture. This gives some major advantages especially in the reusability point of view. For example, our home page is put together using many components, such as a question, this component is reused quite a bit since there can be multiple questions on a home page.



Bibliography

https://www.ngdevelop.tech/wp-content/uploads/2017/12/Angular_Architecture.png

<https://angular.io/api/common/NgIf>

<https://material.angular.io/components/chips/api#MatChipList>

<https://angular.io/api/common/AsyncPipe>

<https://stackoverflow.com/questions/68017199/cannot-get-http-headers-on-angular-interceptor>

<https://jasonwatmore.com/post/2020/04/28/angular-9-user-registration-and-login-example-tutorial>

