Technical challenge

Jose Ramón Serralvo Rojo

Resume

The objective of this technical challenge is to put into practice my ability to develop a functional project in a limited period of time. The project to be developed for this challenge will be a Memes and Gifs web page in which users can upload their files and share it with everyone.

Objective

The objectives of this project are divided into several types:

- Main functionalities:
    - Login and registration.
    - Home page that will show the content uploaded by users.
    - Main Navbar that will include:
        - Content search bar:
            - You will have at least three options to filter the main content.
        - Links to the main categories.
        - Information of the logged in user.
            - If you are not logged in, the button to log in will be displayed.
    - Upload content page.
        - If the user is logged in, a multimedia resource can be uploaded from their computer or from a an external link.
    - Page of the selected multimedia element:
        - The user can obtain the link of the multimedia resource to include it on any website.

- Key requirements:
    - Use at least one third-party API.
    - Use at least one third-party library.
    - Creation of your own database to store all the information.
    - Anyone can access the website and view the content of other users, but only previously registered users can upload multimedia content.

- Extra:
    - The development of additional features to add more value to the project will be valued positively.

- Dependencies:
    - You can use the dependencies that you consider necessary for the development of the challenge.

Development

The following is an elaboration of how I have developed this project.

- **In client side:**.

In client side I decided to use React with redux. The main reason is the ease of controlling the application states from redux.
The architecture of redux is divided into three areas:

- Auth:
  Here I control all the states of current user.
- Entities:
  In this area I store the data of the two entities of the application: Media (Gifs and Memes) and user.
- Ui:
  It is in charge of controlling all the pages.

- **In server side**.

In server side I decided to use Node Js and Mongodb. The main architecture MVC model with a repositories to connect with database using mongoose. Finally, I have managed to create a Rest API.

- **Features.**

I have met all the project objectives, plus added some extra functionalities:

- The user can modify his username and profile picture.
- The user can choose different languages (English and Spanish).
- The user can view other user profiles with memes and gifs uploaded by the user.

Work organization

The delivery of the challenge was 48 hours.

In the first day, I focused to implement all authentication functionality in client and server side. I tried to prepare all the architecture for the next day I only had to perform technical tasks.

In the last day, I focused to implement all content view and search and can also implement language selection.

Some of the improvements I had planned to implement, but could not due to lack of time:

- Pagination in server side with mongoose-paginate-v2.
- Infinity scroll.
- The user can give a like.
- The user can set his profile image with any meme or gif in the web.

Conclusions

I have finally achieved all the objectives and I am very satisfied with the work done. I think the project is very easy to scale due to the great architecture implemented. I have always used the best practices and learned a few more along the way. I think I have made a good fully responsive website with the MERN stack.