Software Architecture: Assignment 1

Books review site, group 6

Subject:
Software Architecture

Teacher:
José Luis Assadi

Students:
Manuel Bentjerodt
Nicolás Gebauer
Jorge Plaza de los Reyes

Delivery date:
August 8

1. ARCHITECTURE

This site was developed with Express as Web Framework and CouchDB as Database Engine. Express.js works with Javascript programming language, which is a fast compiling tool(JIT) mainly used for front-end dynamic development. However, It can be used for server-side and backend development thanks to Node.js. This is where Express applies, a web framework useful for single and multi-page applications, it works with a client-server model in order to separate client and backend methods, and linking them through routes which contain requests and responses via urls linked to the endpoints. CouchDB works as a non-relational database, storing information in JSON formats and API calls through HTTP. Allowing to treat the receiving data as arrays for easier CRUD implementations. The front-end design was made with React.js.

2. WORKINGS

As mentioned above, this architecture works with a client-server model. It can be seen in our folders, where the server handles the fake data(MOCK_DATA.json file), and the API calls with CouchDB(server.js file). In the client folder, the root HTTP is in the “index.js” file, which is linked to the “App.js” where the urls are linked to the components. These components can be seen in the components folder, where are the CRUD functions of each element, the requested tables and the search engine. To update and obtain data for these components, the API present in the server is called.

The connection to the database was made thanks to the nano library. For which we had to define a url and a port for the API calls, done in a private “.env” file.

3. DEVELOPMENT

The team was not familiar with either of the two defined architectures. Therefore, the first hours of work consisted of research and study of the workings, mainly on youtube and in the respective documentations. The familiarity with the javascript language accelerated this process, so it took around 2 hours. The main project was built, the implementation of React made the frontend design easier and its connection with the backend. As for the backend, the database implementation was simple and smooth thanks to the CouchDB facility. Some minor inconveniences appeared during installation but were quickly fixed. The main problems arose due to the nature of the development architecture, since not having a relational database or the ability to implement scaffolding for CRUDs, this was the most time-consuming part. Where calls to the API(aka: queries) and the database had to be redefined several times. As already mentioned, the use of react accelerated the frontend a lot, so the main problem there was the separation logics of the data obtained for the tables (stackoverflow and ChatGPT were key for this process). In terms of time for each task, the project initiation took 30 min to complete, each table took about 1 hour to complete depending on the complexity of the data handling, the harder ones took close to 3 hours but on average it was 1 hour for each view, and the DB and api refurbishes took about 30 minutes each(3 were made). Due to its complexity, the last thing performed was the CRUD operations, which also took about 1 hour.