MS4 Report

Simkina Margaryta
01446530

Introduction

To me this task was completely new and I have put pretty a lot of effort in implementation. Me and my team have met and arranged everything in the beginning of a semester and then kept each other updated about what we have done so far. We all decided to go different ways and use different technologies and only shared with each other how it all is going to work through the endpoints. This was very time-efficient and convenient.

Deviation

I have completely reconsidered my implementation after SUPD. The reason was I decided to spend quite a lot of time on a project and wanted it to be modern and up-to-date technology based. That's why I decided instead of using css, html and java/jsp, use only javaScript, the library of react and some scss instead css. Even though the design wasn't special at all, I really liked the interaction between me and interface, buttoms and the "advanced" commands.

Interface

My service consumes the REST Apis of MS1 and MS2 and does not offer an API itself, so i am not exactly sure what i should describe here, but just in case i will shorty describe what routes of MS1 and MS2 are used in my frontend. e

The MS3 only got 3 interfaces:

1.Ip:port/login: Takes a json with email and password On Success: if username and password are correct, the interface returns the token, message:success & 200 Code to the response body happens via POST On fail return error.

2.Ip:port/register: takes a JSON POST body with email, username, and password On Success: saves the username+ email + password to the database. On Fail: sends a 400 code (should be handled better) and message “password is incorrect”

3.Advertisement page with all the Objects, that can be saved or the 400 fail is to be sent.

Tutorial

A short step to step guide.

1. Before starting the app you have to configure the endpoint urls, , this can be done in the files login/page/register, and currently already configured for my team members’ IPs.

2. Start MS1 and MS2 (import maven project in Eclipse of MS1 and MS2, this loads all dependencies what are needed for the project)

3. Afterwards you have to install all relevant node_modules using yarn or npm. This can be done in repository of MS4 and installed node there by calling “npm install”(the package consisting json object).

4. Once this was done, I have put “npm run dev” in order to run my server side.

5. If everything works out, the interfaces ip:port/login & ip:port/register are available on the same page in a browser.

Testing

I did not complete the testing correctly, though tried with Cypress, but only username and email worked. It was done through installing cypress and running it locally. But I have cloned the files of my team members and through NGROK I could run those both servers while simply giving two different localhosts and then calling them in NGROK separately.

So I first run for instance MS1 on port 7777. Second step of mine was to either stop it or to run DemoApplication of MS2 on port lets say 8080. This allowed me to represent my second page of Advertisement.

This allowed me to test my Advertisement page, as well as adding, editing and deletion of objects there.

Design

Was definitely not my strongest side but I still do love its simplicity and nativeness. Also haven’t seen this one before in terms of advertisements placed on the page.