CS319
Object-Oriented Software Engineering

Final Report

Section 1

Can Tücer (22203239)
Begüm Filiz Öz (22203470)
Orhun Ege Çelik (22202321)
Ruşen Ali Yılmaz (22203805)
Mehmet Emin Avşar (22202995)
Göktuğ Ozan Demirtaş (22202913)

Build Instructions

Front-End Build Instructions

1. Clone the repo to a folder. The GitHub repo can be found through this link: https://github.com/avsarcs/TOYS and when the green Code button is clicked, different instructions to clone the repo will be given, based on the way you wish to clone the repo.
2. Cd to the toys-frontend directory.
3. Run “npm install” command. Wait for the installation to complete.
4. Run “npm run build” command to build.
5. Run “npm run preview” to run the build in local host.
6. Visit the url localhost:4173 in your browser to view the front-end.

Back-End Build Instructions

1. Our dependency for the back-end database is Firebase Firestore. First, create a Firebase account. Create a project. Follow the regular guidelines for these steps.
2. Structure the database according to the database dbs (database structure) file.
3. Create a service account and get the required credentials.
4. Move the relative credentials file into the resources folder and rename it “googlecreds.json”.
5. Download the necessary maven packages and dependencies using the IDE.
6. Build the package with a predefined maven clean lifecycle step.
7. Run the Python script scraper files in the data folder to get high school and university data.
8. Rename “İhsan Doğramacı Bilkent University” json file to “bilkent.json”
9. Run “combine” python script to unify the data into a single json file.
10. Move the this Json file in the relative path of ./data/universities.json of the jar package.
11. Run toys.jar

Work Allocation

We separated our work in a way that every member of our group had a similar workload. We paid attention to giving similar and related tasks to the same person so the learning phases are shortened and incompatibilities are prevented. Our detailed work allocation is listed below.

1. Can Tücer

- Designed and implemented high school analysis, university analytics, tour analytics, and user manual pages.
- Created backend documentation for high school analysis, university analytics, tour analytics pages.
- Worked on backend-frontend integration of data analysis pages.
- Wrote user manuals.
- Prepared state diagrams, activity diagrams, sequence diagrams, and non-functional requirements for the design phase.
- Was responsible with formatting of deliverable reports.

2. Begüm Filiz Öz

- Developed data scraping systems for high school data.
- Designed and implemented fair-related pages, payment pages, and the navigation bar.
- Worked on backend-frontend integration of fair-related and payment pages.
- Wrote user manuals.
- Worked on the initial design of every page on Figma.
- Prepared sequence diagrams, class diagrams, and non-functional requirements for the design phase.
- Wrote the initial pseudocode for the guide matching algorithm.

3. Orhun Ege Çelik

- Developed data scraping systems for university data.
- Designed and implemented profile pages.
- Worked on backend-frontend integration of profile pages.
- Worked on the backend endpoints of data analysis pages.
- Prepared class diagrams for the design phase.

4. Ruşen Ali Yılmaz

- Designed the database and backend server.
- Architectured the backend infrastructure.
- Handled DevOps operations, cloud and bare metal.
- Developed every non-analytical endpoint for the REST API.
- Prepared class diagrams for the design phase.

5. Mehmet Emin Avşar
    - Designed and implemented tour application pages, tour details pages, tour review pages, and personnel management pages.
    - Worked on backend-frontend integration of those pages.
    - Worked on the initial design of every page on Figma.
    - Prepared the backend documentation for every page except data analysis pages.
    - Prepared enumerations and interfaces for the front end.
    - Worked on use case diagram and non-functional requirements for the design phase
6. Göktuğ Ozan Demirtaş
    - Designed and implemented login and register pages, dashboard page.
    - Worked on backend-frontend integration of those pages.
    - Worked on the use case diagram and tech stack for the design phase.
    - Assisted members in the development of many of the pages in the frontend.
    - Streamlined authentication across the frontend pages.
    - Designed pages on Figma.
    - Designed and implemented the Dashboard page.