ASSIGNMENT-3

Abhiram Kolisetty(G01506075)
Chinmay Raz Vaiisshnav Maddi(G01504185)

Naga Venkata Sai Chennu(G01514409)

Frontend :

The frontend provides interface which is needed in this assignment .

Step1:

- In the home page we can see two buttons Student Survey and View surveys. By clicking on student survey the person with the link can access and give the survey. In view survey's we can see who has given the survey and we can edit/delete the survey according to our need.

Step2:

- We have used HTML, Bootstrap , CSS for styling and responsive design.
- JavaScript for Client-side form validation and data handling.
- The color scheme we gave is GMU color scheme
  - Green: #00633
  - Yellow:#FFCC33

Step3:

- The file names are:
  - index.html
  - survey.html
  - list.html

Step4:

- The survey page includes of **:**

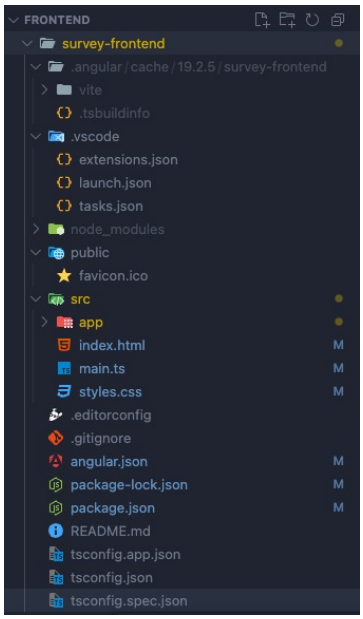
- First name (alphabets only)
- Last name
- Street, city, state, zip

- Telephone (exactly 10 digits)
- E-mail
- Date of survey
- Checkboxes for liked campus aspects
- Radio for interest source
- Dropdown for recommendation
- Text area for additional comments
- **Submit**: validate, store data
- **Cancel**: reset form, discard data

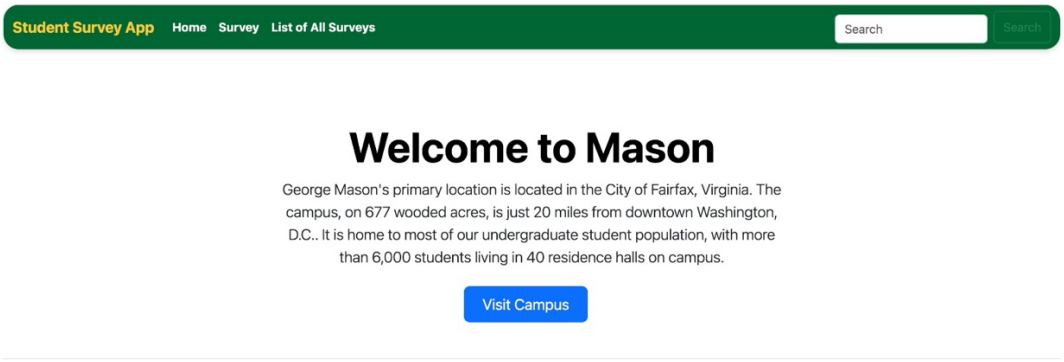
Step5:

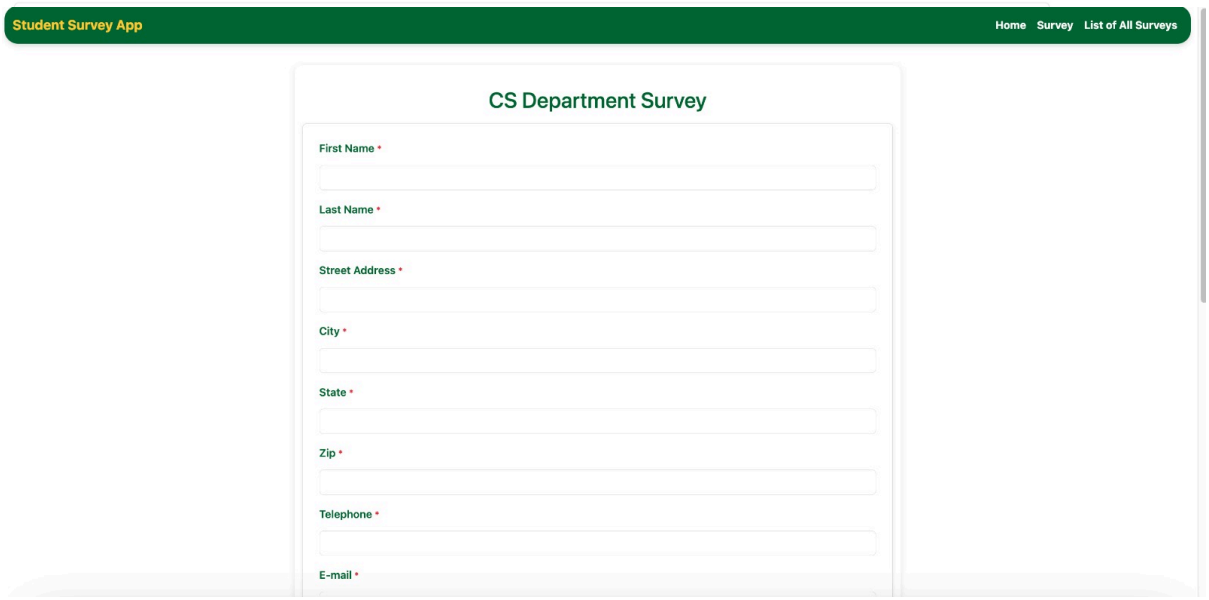
- For executing the frontend:
  - Install a quick static server, e.g. npm install -g http-server.
- Commands needed:
  - npm install -g http-server
  - ng new survey-frontend
  - cd survey-frontend (// to run our project use this command to change the directory and then can use the next command)
  - ng serve –open

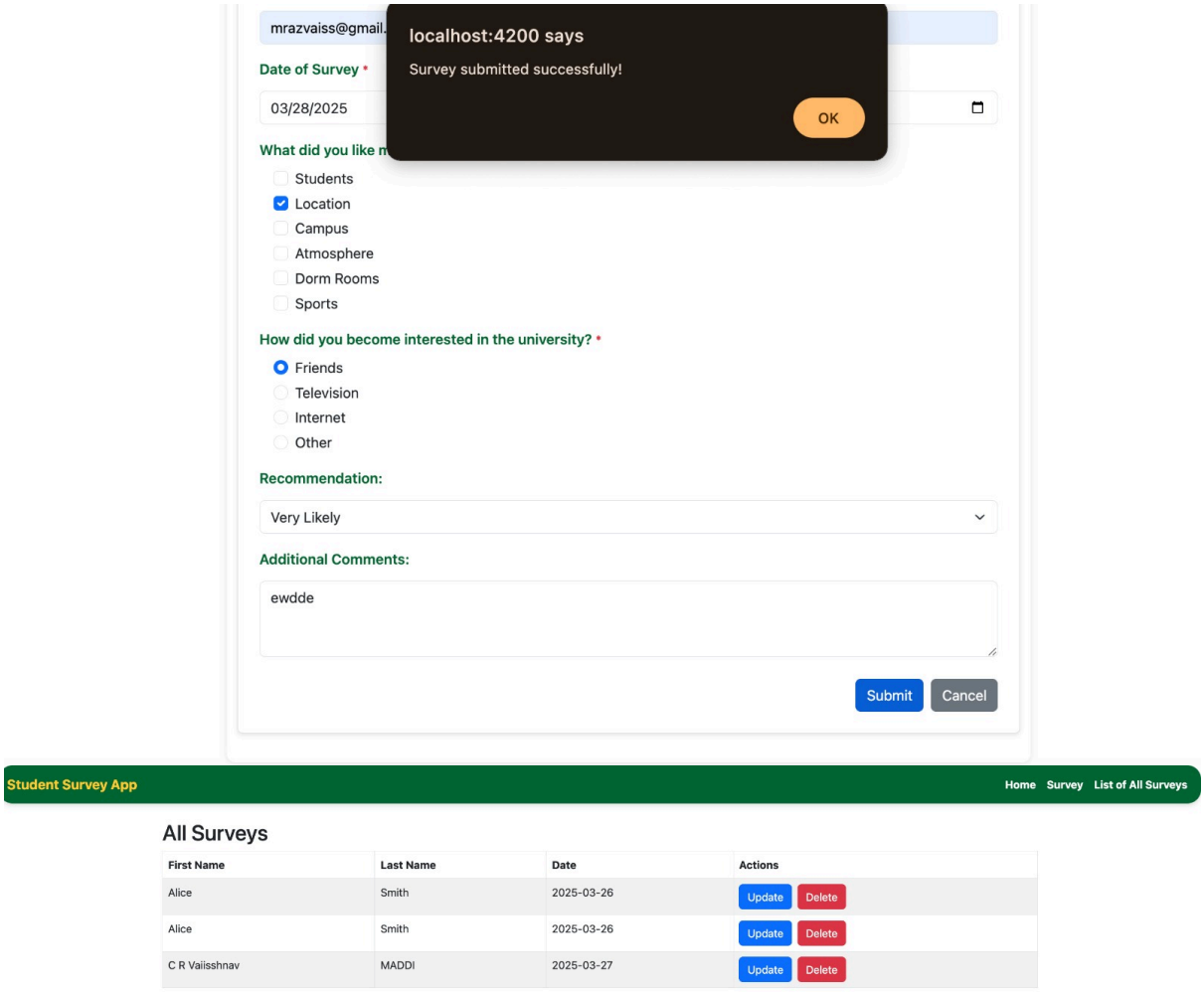
Structure of Frontend



Outputs for Front-end part:







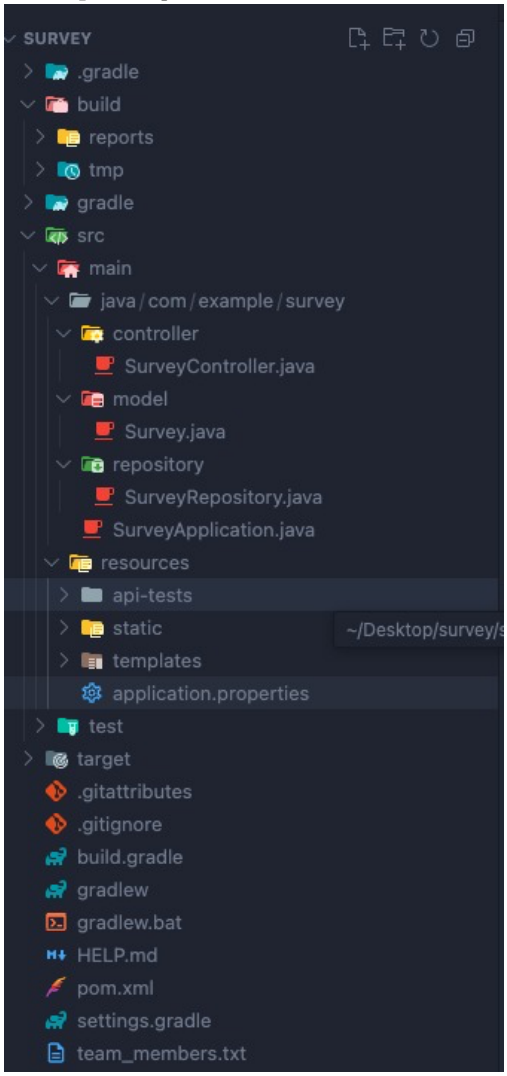
Backend:

This Spring Boot application provides a RESTful API for managing survey data. It uses Spring Data JPA to communicate with a MySQL database hosted on Amazon RDS. The API supports standard CRUD operations (Create, Read, Update, Delete).

Step1:

- **Java 17**
- **Maven** installed (or use Maven Wrapper)
- An **Amazon RDS** MySQL instance set up:
    - Note your RDS endpoint, database name, username, and password.

Step:2 (Backend Structure):



Step3:
Configure the Database using

 # Amazon RDS MySQL connection

spring.datasource.url=jdbc:mysql://<RDS-ENDPOINT>:3306/<DB-NAME>?useSSL=false&serverTimezone=UTC

spring.datasource.username=<RDS-USERNAME>

spring.datasource.password=<RDS-PASSWORD>

spring.datasource.driver-class-name=com.mysql.cj.jdbc.Driver

 # JPA/Hibernate settings

spring.jpa.hibernate.ddl-auto=update

spring.jpa.show-sql=true

spring.jpa.properties.hibernate.dialect=org.hibernate.dialect.MySQLDialect

- **Replace these terms:**
  <RDS-ENDPOINT> with your Amazon RDS instance endpoint (e.g., survey-db.cwtgce0qa7ju.us-east-1.rds.amazonaws.com).
- <DB-NAME> with your database name (e.g., survey_db).
- <RDS-USERNAME> and <RDS-PASSWORD> with your credentials.

Step4(Commands needed)

We should install all the mvm packages

mvn clean install

mvn spring-boot:run

http://localhost:8080/api/surveys