Design Specification

Overview:

The project is to develop a web application based to store user information such as, name, email, income, and savings of the user. It is a basic web application made using Java Spring Boot (backend), Thymeleaf templates and Bootstrap CSS (frontend) and MySQL Database (database) which is further converted into a single standalone application using Mavel build tool.
It contains a form with static questions but is fully functional and can be used by many after deployment on a server.

Main Goals:

The project was developed with the intention to demonstrate my back-end development skills, thus major focus was placed on making the back-end business logic failsafe and proper implementation of all the methods and services needed.

Time & Budget Constraints:

The project took about 6 days of work to complete and for it to be production ready.
As for budget, I used only a single Google Sheets API which was free to use.

System features:

The web app has the following features for the users:

1. Easy to use and understand.
2. Easily transferrable to Google Sheets.
3. Other than authorized users, others can only view the sheet but not make changes in it.
4. Easy to deploy by making a simple docker command, using the Dockerfile and docker-compose file present in the project.

System Requirements:

Upon being deployed by using the Docker daemon, no other system requirements are necessary. So, all the requirements come from docker itself.

1. System Memory: Primary 4GB
   Secondary 2GB
2. Docker Daemon: latest version
3. Additional space for persistent MySQL container: 1GB
4. System: 64-bit architecture
5. WSL enabled in case of Windows Devices.

Application Constraints:

The application constraints stem from the use of Google Sheets API, which being the free version has some constrains as follows:

1. Read Requests per minute: 300
2. Write Requests per minute: 300
3. Read Requests per User per minute: 60
4. Write Requests per User per minute: 60

Application Execution:

The application can be executed by following steps:

1. Extract the project into a folder of your choice
2. Open terminal and go to the target folder where the application is present.
3. Run the following docker command:

   ```
   docker-compose -f docker-compose.yaml up
   ```
4. And your application is up and running and ready to use on local server 8080.