System Analysis and Design of Web-Based Data Filtering System

Introduction

The web-based Data filtering system aims to provide a platform to make filtering excel files easier. The filtering system will cater to the needs of the people in the university.

The system's scope encompasses various functionalities, including uploading a file to the web app so the user can upload any excel file and upload it when it is uploaded the user can view the excel file and the user will have a filter button to make certain data that the user needs to see easier to see. The system will also feature a graph of that filtered data.

It's important to note certain limitations. The system does not support other file types it will only support excel file types (xlsx).

I. Tools

The following tools are to be used on said project:

- Visual Studio Code (IDE)
- Python
- HTML
- CSS
- Java
- JavaScript
- Streamlit

II. Functional Requirements

- File Upload: Enable users to upload Excel files containing data.
- Content Display: Displays the Uploaded Data
- Filtering: Allows the users to filter the uploaded data
- Show Filtered data: Displays the filtered data.
- Graphical Representation: Provide a graphical representation of (such as charts or graphs) based on the filtered data.

III. Non-Functional Requirements

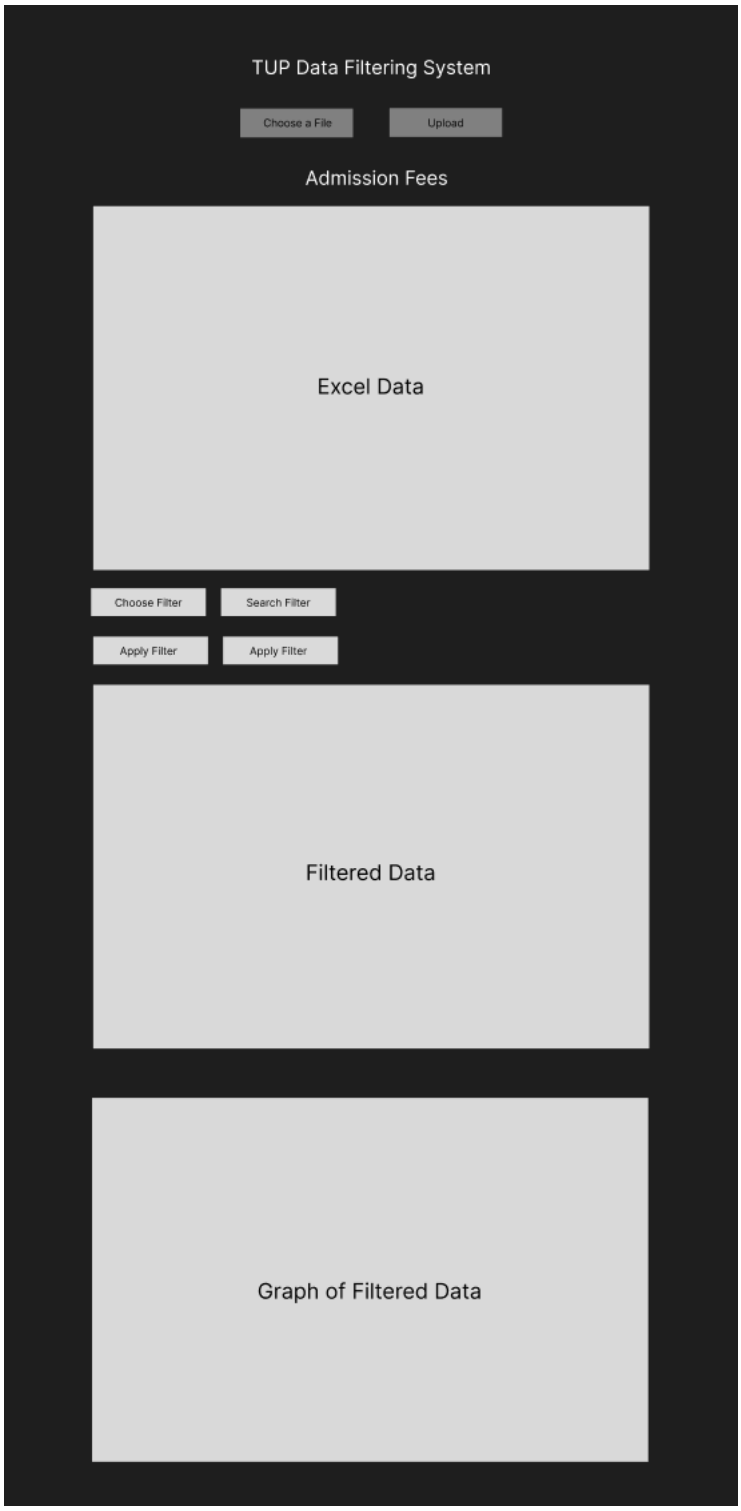
- Performance: The system should be able to handle large Excel files efficiently and process filtering operations in a timely manner.
- Usability: The user interface should be intuitive and user-friendly, allowing users to easily upload files, define filters, and interpret the displayed data.

Conclusion

In conclusion, the software analysis for planning provides an overview of the analyzed system, its functionalities, and non-functional requirements. The importance of implementing the recommended

changes should be emphasized, highlighting the potential improvements and benefits they can bring to the planning process.

Initial Concept Design



Developer Internship Training Plan

“Web-Based Data Filtering System”

Duration: 4 weeks

Week 1:

Day 1: Define the project scope and objectives.

- Identify the tools needed for the project.

- Determine the features that the project will have.
- Set Specific goals for every week.

Day 2-3: Create wireframes and design the UI:

- Develop a concept of the web application.

Day 4: Study the chosen technology and set up development environment:

- Study python and its libraries
- Setup python environment

Week 2-3:

Day 5-8: Front-end Development

- Implement user interface using streamlit or HTML, CSS
- Ensure that the web app looks good across various devices.

Day 9-12: Back-end Development

- Develop the server side of the system.

Day 13-14: Developing the features.

- Implement the needed features for the web app.

Day 15-16: Testing and bug fixing:

- Perform Specific tests to know if the web app is properly working.
- Identify and fix any known bugs.

Week 4:

Day 17 to 20: Prepare for final testing and finalization:

- Prepare final testing and debugging.
- Prepare documentation and user guides.