Individual Requirements for Front End

Xiaoyi Li

Introduction

The purpose of this document is to illustrate the software requirements for the term project. It details how to implement the design of the web front end and outlines the software architecture and web page behavior and system requirements. It also includes an introduction to design constraints and the internal workings of web pages.

Software product overview

This page is designed to get information from the Augur API and display it on the web. Everyone can use it. The web page will show the data related to each group, show the information on open issues and show the information for closed issues according to the user's input.

System use

Actor survey

Employee

The engineer is responsible for adding endpoints and improving the website.

User

Users can view some of their own github information on this website. Through this website, users can more easily view the data related to each group of their own, information about open issues and information about the closed issues. In this way, users can have an intuitive feeling about their work in each group, and it is more convenient for users to coordinate their time and work.

System Requirements

Use Cases

1. **Use Case One:** Show data related to each group (Group-Related)
    a. This will be a tab on our home page that, when selected, will initially show data pulled from the augur endpoints pertaining to.
        i. Repo code changes
        ii. Repo issue response time
        iii. Comments added per repo
    b. Users will be allowed to search/sort the data loaded by repo name as well
2. **Use Case Two:** Show info on open issues (Repo-Related)
    a. Issue Backlog - Number of Issues open for Repo
    b. Issue Duration - Most Recent/Oldest issue for a repo
    c. Issue Response Time - Average time it takes to resolve issue

System function specification

The system is a web that uses the server. The data will be pulled out through the server. There will be a main web page that will be displayed on the corresponding page based on user input.

Non-functional requirements

Safety

1. The website will not reveal user information.

Reliability

1. The web will always be available.

2. The only error may come from the closure of the API.

Performance

1. The website can respond to user input within 0.5 seconds.

2. The response time may be slow depending on the amount of data retrieved by the calling API.

Usability

1. The web will be user friendly and easy to use.

2. Users can use this web on any device.

Design Constraints

1.The web must be completed by December 4th

2. Web technology

    a. HTML

    b. Javascript

    c. CSS

3. Server backend technology

    a. PHP

4. Augur API endpoints

5. A web server

Purchased Components

A server for the web. Users can go to the website and use all information.

Interfaces

The web application interface will be created using HTML, CSS, and Javascript. The interface will be able to be accessed by any user via the web