

# Sprint 1 Design Document - Group 8

## Relevant Links:

[https://oss-augur.readthedocs.io/en/dev/\\_images/schema.png](https://oss-augur.readthedocs.io/en/dev/_images/schema.png)

<https://oss-augur.readthedocs.io/en/dev/getting-started/create-a-metric/create-a-metric-toc.html>

<https://github.com/computationalmystic/sengfs19/blob/master/assignments/9-sprint-1.pdf>

## Goals:

1. Simple and intuitive UI
2. Users can view relevant metrics about github projects
3. Have visualizations beyond just listing data
4. Include summarizing functions beyond raw data
5. Documentation within project
6. Error handling for API calls that don't contain the requested data
7. Choose API endpoints we want to work with

## Use Cases

1. **Use Case #1:** List issues for a repo with their relevant assignees
  - a. Design decisions
    - i. Input a repo, generates a list of issues
    - ii. Clicking an issue generates a list of contributors assigned to that issue
  - b. Data sources, functions, etc from Augur that will be used for each case:
    - i. SQL will pull from the augur\_data.issues\_assignees table
    - ii. Also will pull relevant data from augur\_data.issues and augur\_data.contributors
  - c. Individual (team member) requirement connection:
    - i. Jacob has an interest in back-end work so pulling from two tables using a SQL query will fit his skills
    - ii. There will be a variable list to display issues, so front-end work can be done by Claire in this regard.
    - iii. The API metric function will need to be tailored to the SQL call, thus Davin and Chris can help here
2. **Use Case #2:** Compare number of contributors for two repos
  - a. Design decisions:
    - i. Dropdown menu to choose repos
    - ii. Data displayed in bar graph with number of contributors per month
  - b. Data sources, functions, etc from Augur that will be used for each case:
    - i. SQL will pull from the augur\_data.contributors table
  - c. Individual (team member) requirement connection:

- i. Chose to work with the issues db table by pulling two specific repos since Jacob's interest is in backend and this will allow him to create a new API call.
  - ii. Chose to submit the repos with dropdown menus since Claire's interest is in frontend and wants to have an intuitive UI.
  - iii. Displaying the data in a bar graph will provide a quick visual comparison for users.
- 3. **Use Case #3:** Compare number of active issues for two repos
  - a. Design decisions:
    - i. Dropdown menu to choose repos
    - ii. Data displayed in bar graph with number of issues per month
  - b. Data sources, functions, etc from Augur that will be used for each case:
    - i. [http://augur.osshealth.io/api\\_docs/#api-Evolution-issues\\_active\\_repo](http://augur.osshealth.io/api_docs/#api-Evolution-issues_active_repo)
  - c. Individual (team member) requirement connection:
    - i. Chose to work with the issues db table by pulling two specific repos since Jacob's interest is in backend and this will allow him to create a new API call.
    - ii. Chose to submit the repos with dropdown menus since Claire's interest is in frontend and wants to have an intuitive UI.
    - iii. Displaying the data in a bar graph will provide a quick visual comparison for users, and this can be done by Chris and Davin.

## Endpoints

- 1. Endpoint for Use Case #1:
  - a. /repo-groups/:repo\_group\_id/repos/:repo\_id/issue\_assignees
    - i. This will be a new endpoint
- 2. Endpoint for Use Case #2:
  - a. /repo-groups/:repo\_group\_id/repos/:repo\_id/contributors
    - i. May create new endpoint with a simple count of contributors in the SQL query
- 3. Endpoint for Use Case #3:
  - a. /repo-groups/:repo\_group\_id/repos/:repo\_id/issues-active
    - i. May create new endpoint with a simple count of issues in the SQL query

Frontend Wireframe (next page)

