Software Requirement Specification

CS4320 Software Engineering

Synhiranakkrakul, Thunnathorne

Computer Science, University of Missouri
Spring/2020

1. Introduction:

This document is specifying the requirements and detail of the software architecture of Augur. This Software Requirements Specification (SRS) will describe how API work on the system in term of overview, functional requirement and also a non-functional requirement, design constraints, purchased components and Interface design as a scheme. As such, any section of this document is subject to change based on the feedback and outcomes of each implementation sprint.

2. Product overview:

Augur is a product from CHAOSS that will help visualize the metric in each category by using API and the result of API will be a JSON file, the developer can using the JSON file to analyze data and visualize as a metric in a kind of statistic like a bar, pie, line graph. all of these could be finding the correlation between 2 things that the user focused. moreover, this API can summarize the cumulative data. for example, the top-committer, top-contributor in each group, the issues and etc. The developer can be adapted to use in their own purpose which might be bringing up higher performance for their application by using Augur API for research or specific commerce.
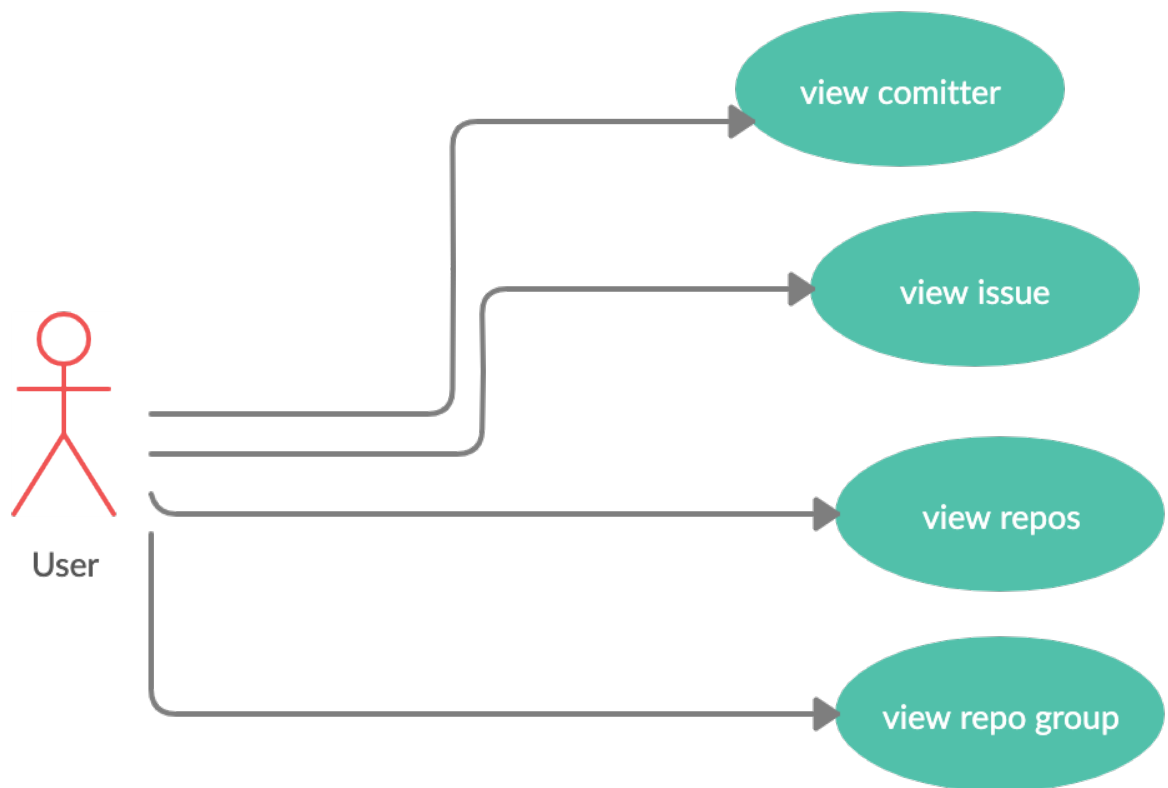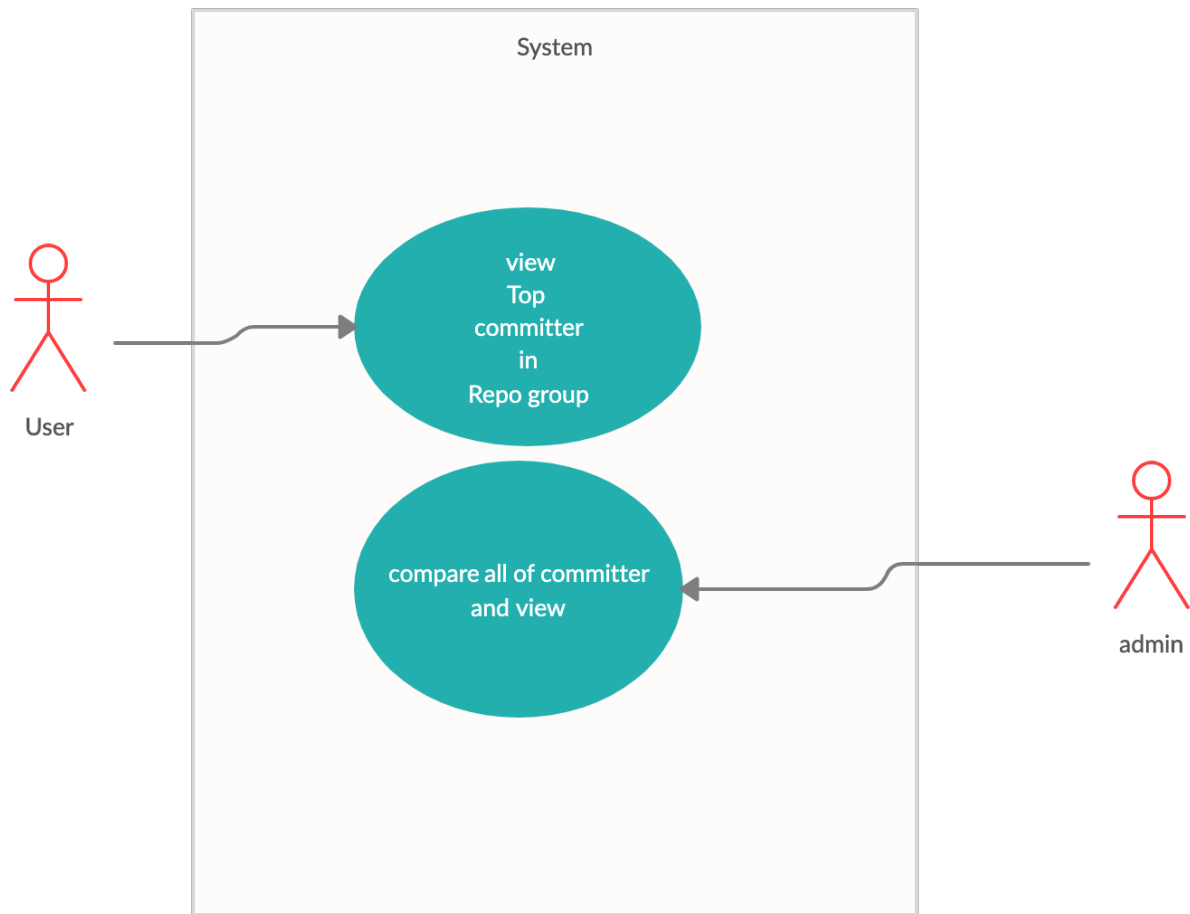
3. System Use:

Actor survey:

User: The user can query the data from the database by request data using the API with the GET approach and some of them might have to using some params to get a unique result for their specific purpose. after the user request data with API, they will get the result as a JSON format file and each key in the file mean specifically. the user can be used to view the data, plots in the graph as a statistic forms such as metrics to find the relationship between two data, the frequency of issue, commit and risk in the system for improving the system and make it higher performance to support the things in the future.

4. System Requirements:

Use Case 1: The user view information from the system.

Use Case 2: The user view top committer and admin compare all of committer and view from the system.

User

System

view
Top
committer
in
Repo group

compare all of committer
and view

admin

4.1   Functional specification:

- Find Top committers in each repo group
- Pie and Line graph
- Find the issues active
- Summary

4.2   Non-Functional specification:

- Colorful user interface
- Decoration with CSS

5. Design Constraints:

The JSON File and Field have to be valid. The application has to connected to internet for using API, jQuery, XHTTP with GET method and Format for API (params) has to be correct format

6. Purchased Components:

Augur is an open-source and using the API with no costs, but the components that will have to pay might be a server, maintenance services that depend on the scale of the provider and also the place like leasing real estates such as apartment or building for placing the server and other devices

7. Interfaces:

Welcome

Top Commiters

Issues Active

Summary

Top Commiters

Top Commiters

- antoine@pytho…
- szucs.krisztian…
- neal.p.richards…
- kou@clear-code.com
- wesm+git@ap…
- jorisvandenbos…
- other_contribut…

13.1%
10.2%
9.4%
9.2%
7.5%
46.1%