Semester Project Requirements Document

- Introduction

We will be leveraging metric models created by the CHAOSS community to add additional visualizations to 8Knot. Our additions will be primarily centered around Bus Factor (see explanation below).

- Software Product Overview

The software will be added through a fork to the existing 8Knot repository. The visualizations will be shown through an additional page added to 8Knot.

- System Use

This system will use 8Knot to add additional visualizations of various metrics through metric models by CHAOSS.

    - Actor Survey
        - **Contributors:** These are the actual developers working on a project who are writing the code and committing it to a repository/project
        - **Project Maintainer:** These are the users who are responsible for delegating tasks to contributors, ensuring that the project is staying on schedule - essentially the manager of the project
        - **Company Representative:** A representative from a corporation may use the metrics to make informed choices on whether to use a certain project within their work or company.
- System Requirements
    - Use Case 1: Viewing Metrics of a Project
        - Description

            The user knows of a project that they wish to view metrics for and is tracked by Augur. The software will return various metrics that will indicate how healthy and/or sustainable a project is.
        - Triggers

            The user finds a project that they would like to start tracking due to its merits of potentially being a long-term project that they will engage with and view metrics of.
        - Actors

            Contributors and Project Maintainers of the 8Knot repository who want to add and change metrics of a project.
        - Preconditions

A running 8Knot instance in Docker as well as a repository that 8Knot is tracking.

- Main Success Scenario

  The system successfully retrieves the repository and displays the metrics about it for the user.

- Alternate Success Scenarios

  There are no other success scenarios.

- Failed End Condition

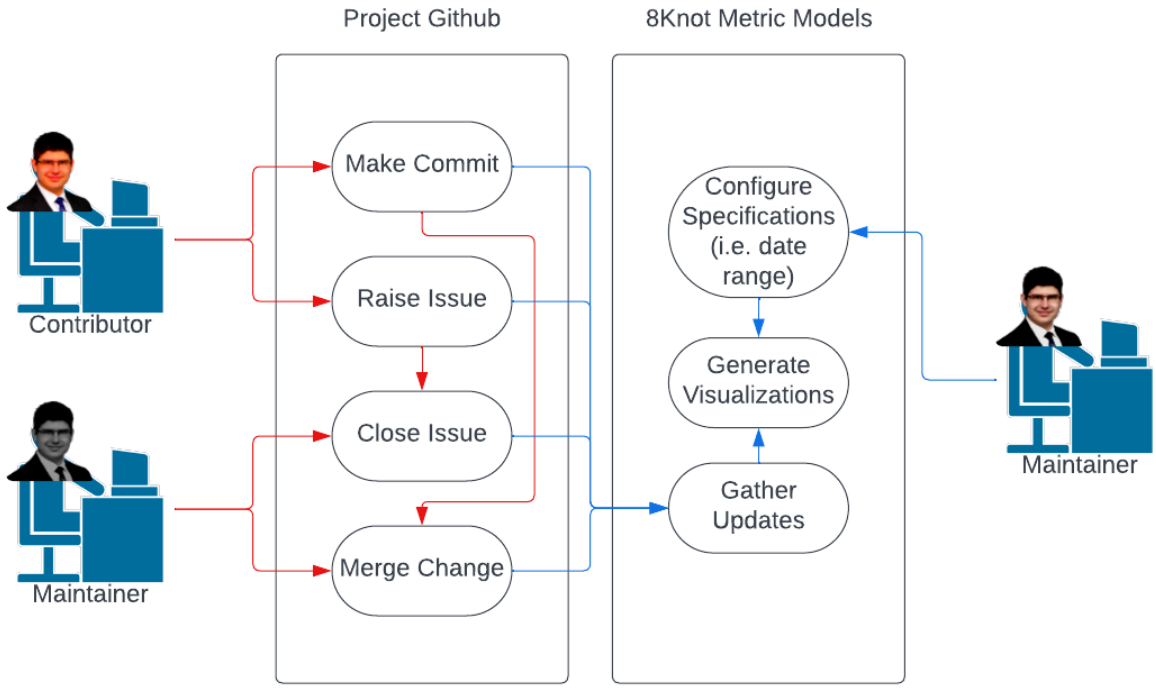
  The user provides a bad URL to a repository that the system either does not have permission to view or does not exist in the first place. The user could have selected a repository where the said metrics could not be generated due to the repository not possessing enough data.

- Extensions

  There are no extensions to this use case.

- Steps of Execution
  - The user inputs a GitHub or GitLab URL into the system
  - The system will compute the various metrics of the repository through the backend
  - The frontend system will display the metrics for the user to view.

- Use Case Diagram



        - Dependent Use Cases

            None

    - System Functional Specifications

    - Non Functional Requirements
        - The system should feel tactile and responsive
        - The system will be able to generate metrics in less than 5 seconds
        - Reports should be able to be printed in an easy format
- Design Constraints
    - Docker support for host machine
    - 8GB of system RAM
    - Supported browser (Safari, Firefox, Chrome, etc) on actor machine
- Purchased Components

    Some purchased components that could be needed to support the system include server hardware to allow for users to host the system through the developers instead of configuring their own instance. This would require servers for both the front end and back end to be purchased and stood up.
- Interfaces

As stated above, there will be an interface through a web-application that will be used to visualize the various data that is tracked via the system's backend.

Roles for metric models

- Companies, admins likely to need bus factor

Explanation of Bus Factor:

Diagram 1: Visual explanation of Bus Factor

The bus factor is the smallest number of people who make 50% or greater of the contributions to the project. If one or more of these people are put out of commission (e.g. gets hit by a bus) then the project health is severely impacted (See Diagram 1).

Requirements List

Component	Priority	Requirement Name	Requirement Description
Starter Project Health Model	1	Bus Factor	Determine the smallest number of people that make 50% of contributions
Starter Project Health Model	2	Time to First Response	Determine the amount of time between when the activity was opened and when it received the first response from a human
Starter Project Health Model	2	Change Request Closure Ratio	Measure the ratio between the total number of open change requests during a time period versus the total number of change requests closed in that same period
Starter Project Health Model	2	Release Frequency	Determine the frequency of project releases
Community Activity Model	1	Contributor Count	Determine how many active commit authors, review participants, issue authors, and issue comments participants there are in the past 90 days.
Community Activity Model	3	Comment Frequency	Determine the average number of comments per issue in the last 90 days
Community Activity Model	1	Commit Frequency	Determine the average number of commits per week in the last 90 days
Community Activity Model	2	Updated Issues Count	Determine the number of issues updated in the last 90 days
Community Activity Model	2	Recent Releases Count	Determine the number of releases in the past year
8Knot	1	Display Visualizations	Determine how to display each metric by modifying the graphs

8Knot	1	Search for Repository	Search for a given repository to generate visualizations
8Knot	3	Save State	Store current visualization for later comparison
8Knot	3	Store Frequently Visited Repos	Recall previously visited repos
Community Activity Model	3	Individual Contributor Metrics	Generate visualizations for specific users
Starter Project Health Model	2	Alternate Bus Factors	Bus factor metrics focused on different types of contributions
Community Activity Model	3	Depth of Issue Discussion	Metric measuring the number/length of comments in response to an issue as a proxy for meaningful examination by maintainer
8Knot	1	Metric Model Pages	Create separate, navigable pages for each metric model