

# **Assignment 9: Individual Requirements Analysis**

Haley Massa

# **1. Introduction**

## **1.1 Purpose**

The purpose of this document is to describe the software requirements and architecture of this Open Source Software Health and Sustainability Metrics Tool, which we will refer to as the system. This will be done by examining the software product, system use, system requirements, design constraints, purchased components, and interface.

## **1.2 Scope**

This system will help open source software contributors and communities gauge the health and sustainability of their projects by presenting users with valuable timeseries metrics on their projects.

## **1.3 Assumptions and Dependencies**

Every account in the system must have a specified role (or roles, in specific cases) that determines what information is visible to the user and what abilities users have

# **2. Software Product Overview**

This system will be comprised of a database populated with open source project data as well as numerous metric functions that use various user entered parameters as well as a timeframe. These functions will then query the project database and return the results as a metric that can be used by open source contributors, communities, and stakeholders to understand the health and sustainability of their projects. These project metrics will then help contributors know where to place their efforts and allow open source communities attract new members, ensure quality, and reward valuable project additions.

# **3. System Use**

## **3.1 Actor Survey**

Project contributors:

Project contributors are the individuals responsible for identifying issues, fixing bugs, adding new features, and doing anything else that furthers the progress of the project. Contributors are able to insert data to the database, as well as view the project metrics and advanced health reports determined by the system. The purpose of providing contributors with these capabilities is to enable them to determine where their efforts are needed and most valuable.

## Project Leaders

A project leader is an individual or group of individuals that are in charge of making decisions and directing projects. A project leader can either be a contributor or investor, and in addition to all of the abilities of these users, project leaders have the ability to determine what project information, reports, and metrics are available to the public and what is only available to contributors.

### Potential contributors:

Potential contributors are those who are looking for a project they find relevant and believe they can contribute to. These users have the ability to view and compare public metrics for various projects. The purpose of these features is to help developers determine which projects they would have the greatest impact on and how they can successfully contribute.

### Investors/funding organization:

Investors are individuals or organizations that have provided money and/or resources to select projects. Like potential contributors, investing and funding agencies have the ability to view and compare metrics for various projects. These users are also able to view in depth metrics for the projects they fund and even compare these advanced reports for the various projects they fund. The purpose of these functions is to help investors make informed decisions about where to devote money and resources to in the future and where to encourage contributors to add to.

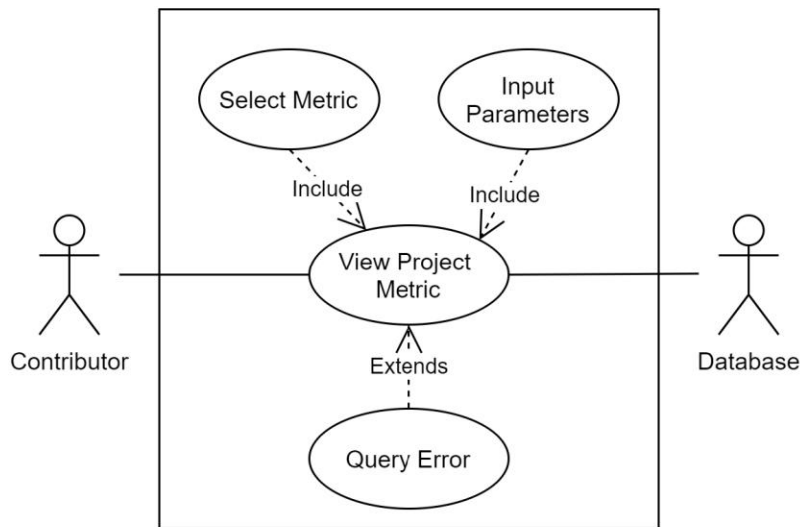
## Project Administrators

Project administrators are responsible for creating projects in the system, creating user accounts, assigning roles, and maintaining current users in the system for each project or organization.

# 4. System Requirements

## 4.1 Use Cases

Use Case 1: Contributor views project metric



### **Use Case: Contributor views project metric**

**Description:** Contributors can view a specified metric for their project

**Actors:** Contributor, database

**Triggers:** Project contributor selects a metric type, inputs the parameters, and submits request to query database.

**Preconditions:** The contributor must be logged in and must be a contributor for that project.

**Main success scenario:** The correct results are returned and displayed to the user.

**Steps of execution:**

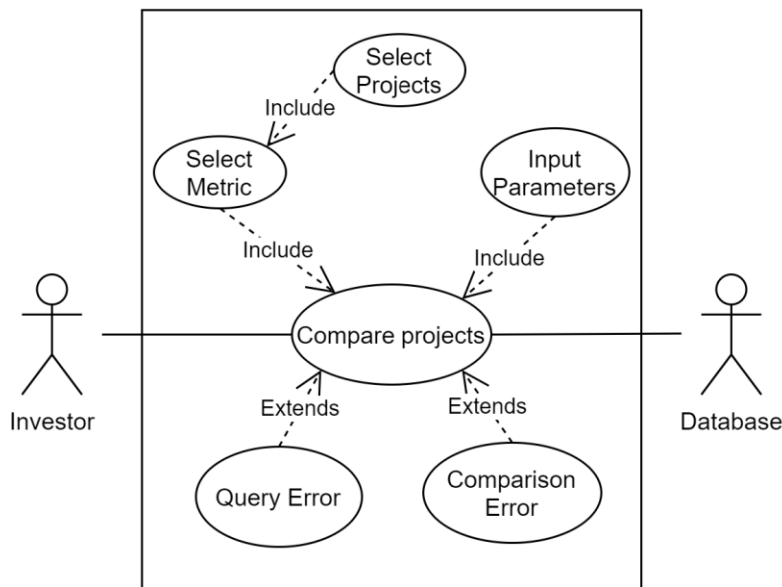
1) The user navigates to the metrics page.

**Failed end condition:** The query cannot be completed because the user does not have authorization or the database does not have the required data.

**Extensions:**

- 1) The required data from the project is not in the database.
  - a) An error message is displayed and the contributor is prompted to retry, select a different metric, or return to their dashboard.
- 2) The user is not authorized to view that metric
  - a) An error message is displayed and the contributor is prompted to retry, select a different metric, or return to their dashboard.
- 3) A database/program error prevents the query from being executed and displayed.
  - a) An error message is displayed and the contributor is prompted to retry, select a different metric, or return to their dashboard.

## Use Case 2: Investor compares projects



### Use Case: Investor compares projects

**Description:** Investors can compare the metrics for two or more projects

**Actors:** Investor, database

**Triggers:** Investor navigates to the metrics page and clicks on the “Compare Projects” tab

**Preconditions:** The user must be logged in and must be an investor for that project.

**Main success scenario:** The correct results are returned and displayed to the user.

**Steps of execution:**

- 1) The user logs into the system.
- 2) The user navigates to the “Compare Projects” tab of the metrics page.
- 3) The user selects which projects they would like to compare and selects which metrics to base the comparison off of run.
- 4) Once the required parameters are input, the user hits submit, and the function is called.

**Failed end condition:** The query cannot be completed because the user does not have authorization or the database does not have the required data.

**Extensions:**

- 1) The required data from either of the selected projects is not in the database.
  - a) An error message is displayed and the user is prompted to retry, select a different metric, or return to their dashboard.
- 2) The user is not authorized to view that metric for either of the selected projects.
  - a) An error message is displayed and the user is prompted to retry, select a different metric, or return to their dashboard.
- 3) A database/program error prevents the query from being executed and displayed.
  - a) An error message is displayed and the user is prompted to retry, select a different metric, or return to their dashboard.

## **4.2 Functional Specification**

- 1) The system shall have a login system that all user roles must use
- 2) The system shall enable users to access a “Dashboard” page that lists general information as well as any specific metric the user would like to keep track of.
- 3) The system shall allow users to find and view metrics they are authorized for using user specified parameters.
- 4) The system shall allow users to save certain metrics to their dashboard
- 5) The system shall allow users to change roles, if applicable.
- 6) The system shall allow users to log out.

## **4.3 Non-functional Requirements**

- 1) The login must not fail on more than 0.001% of attempts.
- 2) Each page should not take longer than 10 seconds to load.
- 3) The database must not fail on more than 0.01% of queries or manipulations.
- 4) No more than 5 seconds should pass between the time the user submits a query request and the time the results are displayed.
- 5) The system must not go down more than once every 3 months.
- 6) User manuals must be provided to all users and must describe every ability of every user role.
- 7) The system must adhere to the usability standards defined on Usability.gov wherever possible.

## **5. Design Constraints**

- 1) The system will be available to users as a web application.
- 2) The system should be backwards compatible.
- 3) Users should be able to access the system through all modern web browsers (Chrome 31+, Opera 12+, Firefox 43+, Safari 6+, Internet Explorer 10+, Microsoft Edge).
- 4) The system must run on Windows, MacOS, Linux, and Chrome OS.
- 5) The system must use an Oracle database.

## **6. Purchased Components**

### **6.1 Software**

No outside software appears to be necessary at this point in development.

## **6.2 Hardware**

1. Web server
2. Database server
3. Application server

## **7. Interfaces**

### **7.1 User Interfaces**

As previously mentioned, the interface for all users will be web-based. User will only be able to access the interface once they have successfully logged in. The interface each user sees is determined by their role. Project contributors, leaders, potential contributors, and investors will all have similar interfaces since many of their main abilities and functions are the same. Project administrators will have a different interface since their responsibilities are quite different.