**Bug Reporting and Tracking System**

Bug Reporting and Tracking System

NAJEEM VT

ARUNIMA JAYARAJ

MARIA BENNY

UMA DEVI

MUHAMMAD AJNAS AP

**SOFTWARE REQUIEMENTS SPECIFICATION**

## Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Description** | **Author** |
| <28/06/22> | <1.0> | SRS 1.0 | Team-1 |
| <02/07/22> | <2.0> | SRS 2.0 | Team-1 |
| <04/07/22> | <3.0> | SRS 3.0 | Team-1 |
| <07/07/22> | <4.0> | SRS 4.0 | Team-1 |

**Table of Contents**

1. Introduction
   1. Purpose
   2. Scope
   3. Definitions, Acronyms, and Abbreviations
   4. References
   5. Overview
2. Overall Description
3. Specific Requirements
   1. Functionality
      1. Admin Login.
      2. Admin add Engineer’s
      3. Admin view active bug’s
      4. Admin view bug history
      5. Admin update bug status
      6. Admin create user account
      7. Admin update user account details
      8. Engineer’s login
      9. Engineer’s view bug Details
      10. Admin Disable/Enable user account.
      11. Engineer update current bug status.
      12. User Login
      13. User report bug
      14. User view bug history
      15. User delete reported bug

* 1. [Usability](#_bookmark2)
     1. Console as Interface
     2. [Accessibility](#_bookmark1)
  2. [Reliability & Availability](#_bookmark3)
  3. [Performance.](#_bookmark4)
  4. [User friendly](#_bookmark5)
  5. [Design constraints](#_bookmark6)
     1. [Standard development Tools](#_bookmark7)
  6. [Purchased Components](#_bookmark8)
  7. [Licensing Requirements](#_bookmark9)
  8. [Legal, Copyright, and Other Notices](#_bookmark10)
  9. [Applicable Standards](#_bookmark11)

1. [Supporting information](#_bookmark12)
   1. [Use Case Diagram](#_bookmark13)
   2. [Sequence Diagram](#_bookmark14)

4.3 [Class Diagram](#_bookmark15)

# **Software Requirements Specification**

1. **Introduction.**

The introduction of the Software Requirements Specification (SRS) provides an overview of the entire SRS with purpose, scope, definitions, acronyms, abbreviations, references and overview of the SRS. The aim of this document is to gather and analyze and give an in-depth insight of the complete Bug Reporting and Tracking System by defining the problem statement in detail. Nevertheless, it also concentrates on the capabilities required by stakeholders and their needs while defining high-level product features. The detailed requirements of the Bug Reporting and Tracking System are provided in this document.

## Purpose

The Bug Reporting and Tracking System is a software that is designed to manage all the details of Bug reported by user. It helps the admin to maintain the database of bugs reported user like active bugs, bug history, users and available Engineers details etc. and also it provides the functionalities to the admin such as adding the details of new Engineer and adding and update details of users. In short, the purpose of this SRS document is to provide a detailed overview of our software product, its parameters and goals. This document describes the project's target audience and its user interface, hardware and software requirements. It defines how our client, team and audience see the record and its functionality. Nonetheless, it helps any designer and developer to assist in Software Development Lifecycle (SDLC) processes.

## Scope

The name of the software is "Bug Reporting and Tracking System ". This system authenticates users at their login. It provides the details of Engineers within the organization. It provides Admin to view, assemble, organize and manage the Engineer’s information and allocate reported bugs to Engineer’s according to availability. Admin can add details of new Engineer’s in the organization into the database.

This SRS is also aimed at specifying requirements of software to be developed but it can also be applied to assist in the modification of the bug’s data. The standard can be used to create software requirements specifications directly or can be used as a model for defining an organization or project specific standard. It does not identify any specific method, nomenclature or tool for preparing an SRS

## Definitions, Acronyms, and Abbreviations

|  |  |
| --- | --- |
| SDLC | Software Development Life cycle |
| SRS | Software Requirements Specification |
| BRTS | Bug Reporting and Tracking System |

**1.3 References**

The references are:

* + - <https://www.geeksforgeeks.org/>
    - <https://www.tutorialpoint.com/>
    - <https://docs.microsoft.com/>

## Overview

Bug Reporting and Tracking System allows a small company or organization to maintain details of reported bug’s information in a more operative manner that will help them to manage and resolve bug’s effectively. Bug Reporting and Tracking System is a software that helps to maintain a database that is useful to store data of the reported bug’s within an Application and generating reports upon request. It helps in assembling, organizing and managing the information of the Engineers and reported bug’s as required by the user.

This system will reduce all the manual work and the whole process can be managed just through single clicks and edits. It reduces the manual paperwork through it and gives proper information of bug’s that have been recorded. Bug Reporting and Tracking System makes easy for the Admin to keep a track and monitor the reported bug’s and manage them.

## Overall Description

This report describes the project development of the Bug Reporting and Tracking System that was developed to manage the reported bug’s details effectively. It can improve management of bug’s information that are available in the database. The System read the data using Console and C++ from the database using File system. The interface between employee data and varies controls like Add, delete, view, update, etc. are done by Console and File system.

## Specific Requirements

The specific requirements are –

## Functionality

Introduction –

This subsection contains the requirements for the Bug Reporting and Tracking System. These requirements are organized by the features discussed in the vision document. Features from vision documents are then refined into Use Case diagrams and to Sequence diagrams to best capture the functional requirements of the system

## Admin Login:

* + - * The system allows the field to enter username and password to login.
      * The system will validate the username and password.
      * The system will notify the Admin if the username or password is not valid.
      * The system will show an error message if it's an invalid User.
      * If the login is successful then Admin can login and view any information.

## Admin Add Engineer’s Details:

* + - * The Admin should login to add Engineer’s.
      * The Admin needs to enter the Details of Engineer’s.
      * The system will display all the fields to fill up for Engineer’s Details.
      * The system will ask to enter the Engineer’s Name, Id.
      * The system will display the message if added successfully.

## Admin view Active bug’s Details

* + - * The system will enable the Admin to view all the active bug’s details in the database.
      * Admin can assign tasks to Engineer’s using Engineer’s Id.
      * The system will display all the active bugs.

## Admin view bug history

* + - * The system will enable the Admin to view all the reported bug’s details available in the database.
      * Admin can sort bug’s Details from Database according to reported year.
      * The system will display all the reported bug’s details.

## Admin update bug status

* + - * The Admin should login to update active bug’s status.
      * The Admin needs to update active bug’s status.
      * The system will display all the fields to update for active bug’s Details.

## Admin create user account

* + - * The system will enable the Admin to create user details.
      * Admin can create user of Id, Name, role and password.
      * The system will display a message user created successfully.

## Admin update user account details

* + - * The Admin should login to update user Details.
      * The system will enable the Admin to view all the added user details in the database.
      * The Admin can disable/enable user account.

## Engineer’s Login

* + - * The system will display the fields to fill up for adding username and password.
      * The system will validate the username and password.
      * The system will notify the Engineer if the username or password is not valid.
      * The system will show an error message if it's an invalid User.
      * If the login is successful then the Engineer can login and view active bugs and bug’s history.

## Engineers View bug Details

* + - * The system will allow the registered Engineer’s to login the system.
      * The system will enable the Engineer to view all the active bug’s and bug’s in the database.
      * The system will enable the Engineer to view the bug’s history in the database.
      * The system will enable the Engineer to view bug’s details like Id, category, details, evidence, reported date and expected deadline.

## Admin disable/enable user account

* + - * The Admin should login to Apply for leave.
      * The Admin can disable/enable user according to their Id.
      * The system will display all the fields to disable/enable user.
      * The system will ask to enter submit button.
      * The system will show a message “User Account Disabled Successfully” on successful disable.

## Engineer update current bug status

* + - * The Engineer should login to update bug status.
      * The Employee needs to update Details of bug.
      * The system will display all the fields to update for bug’ Details.
      * The system will ask to confirm bug status.
      * The system will show updated status of bug.

## User login

* + - * The system will display the fields to fill up for adding username and password.
      * The system will validate the username and password.
      * The system will notify the user if the username or password is not valid.
      * The system will show an error message if it's an invalid User.
      * If the login is successful then the user can login and view bug’s history.

## User report bug

* + - * The system will display the fields to fill up for reporting bug.
      * The system will create an Id for the bug.
      * The system will ask for category, details and evidence of the bug.
      * The system will show a message on successful reporting of a bug.

## User view bug history

* + - * The system will display bug’s history according to their Id.
      * The system will show Id, Category, priority, created date, assigned to which Engineer, deadline and current status.

## User delete reported bug

* + - * The system will display bug’s history according to their Id.
      * The system will show field’s which can be modified by user.
      * The system will show a message on successful deletion of bug.

## Usability

* + 1. **Graphical Interface**

The system shall provide a uniform look and feel.

## Accessibility

The system shall be accessible to admins, Engineer and in an organization and the customer.

## Reliability & Availability

In order to ensure reliability, this system is being designed using software that is established to be stable and easy to use.

This system is designed to run 24/7 and be readily available to the User.

## Performance

The performance shall depend upon hardware components of the client/customer.

## User friendly

The software is made user friendly for smooth usage by the end user.

**3.6 Design Constraints**

**3.6.1 Standard Development Tools**

The system shall be built using a Visual studio

## Purchased Components.

Not Applicable.

## Licensing Requirements

Not Applicable.

## Legal, Copyright

Bug Reporting and Tracking System should display the disclaimers, copyright, word mark, trademark and product warranties of the Organization.

## Applicable Standards

It shall be as per the industry standard.

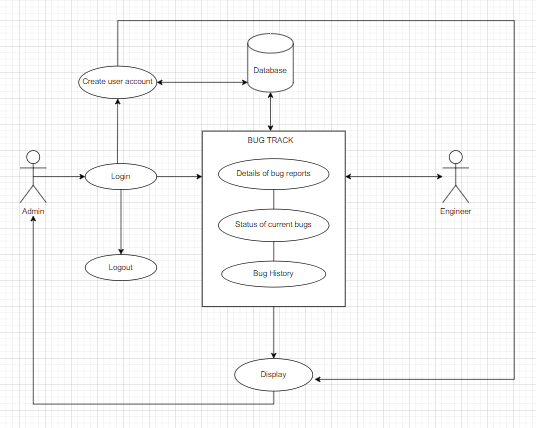
## Supporting Information

Please refer the following document:

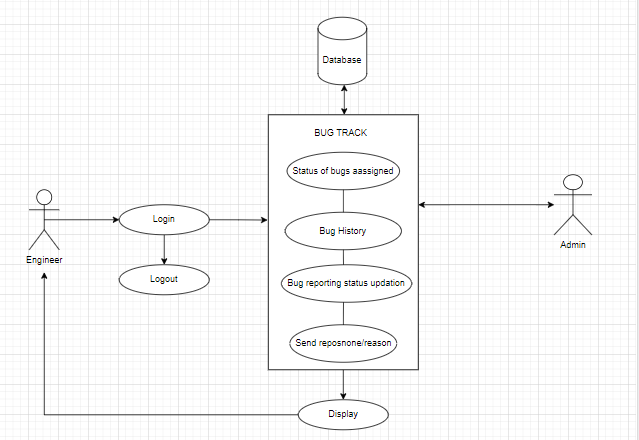
* 1. Use Case Diagram.
  2. Sequence Diagram.
  3. Class Diagram.

## Use case diagram

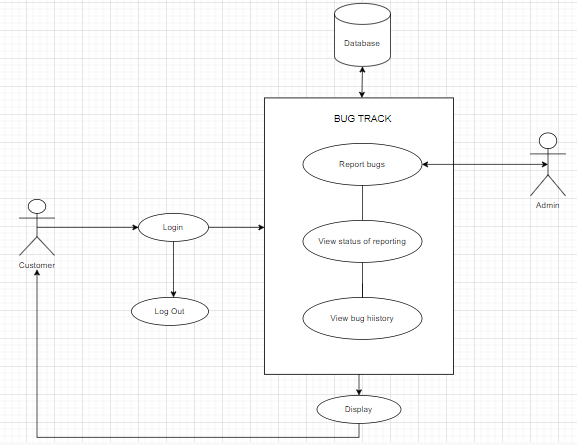
* + 1. **Admin**

****

**4.1.2 Engineer**

****

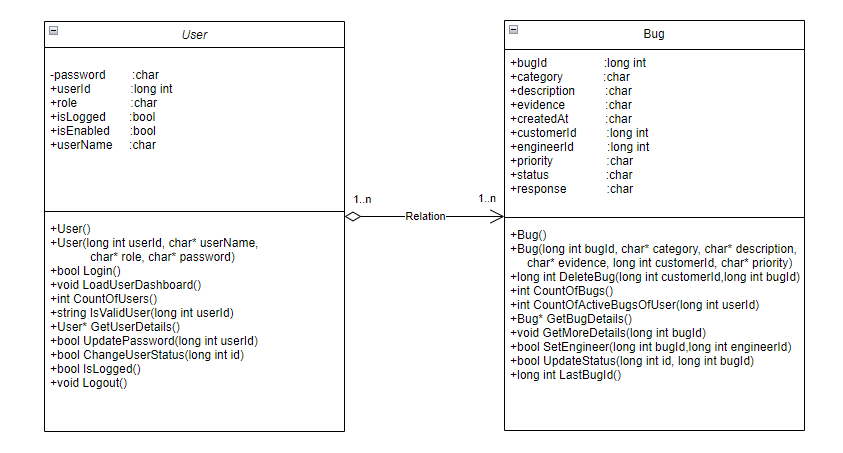
**4.1.3 User/Customer**

****

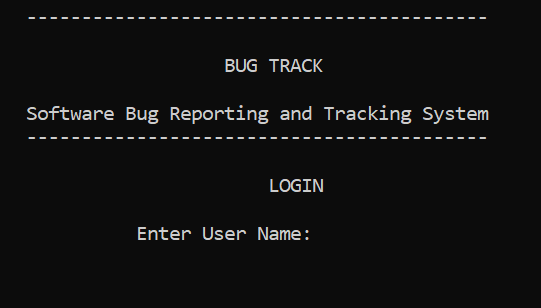
## Sequence diagram

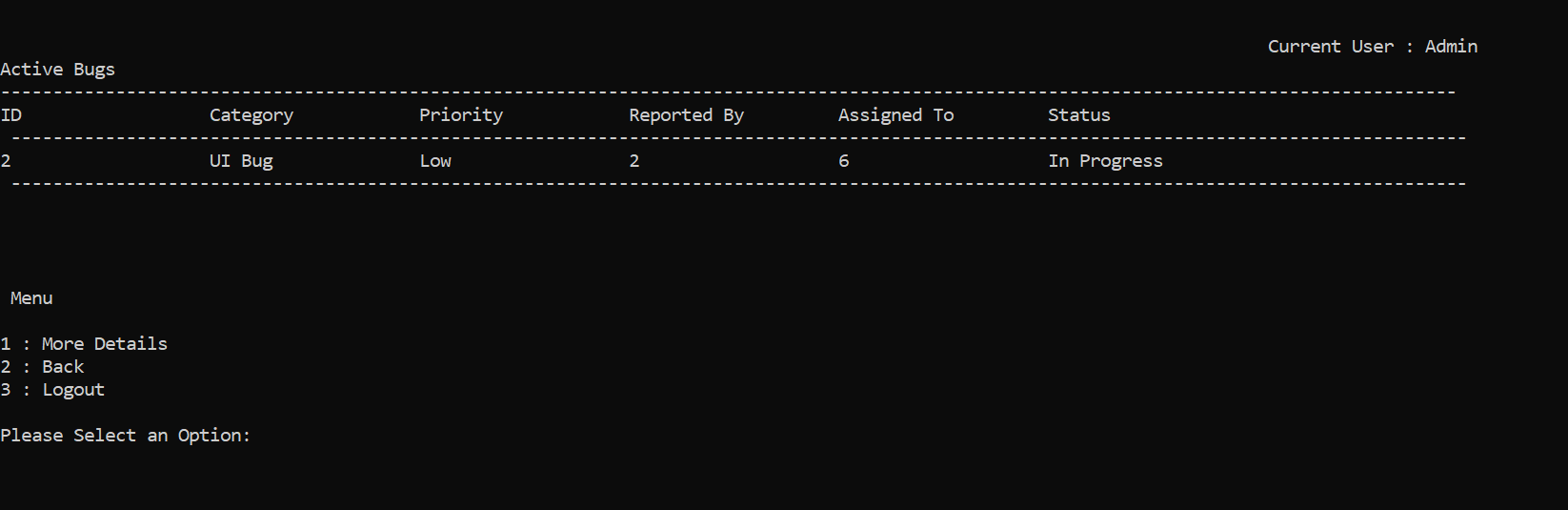
## 

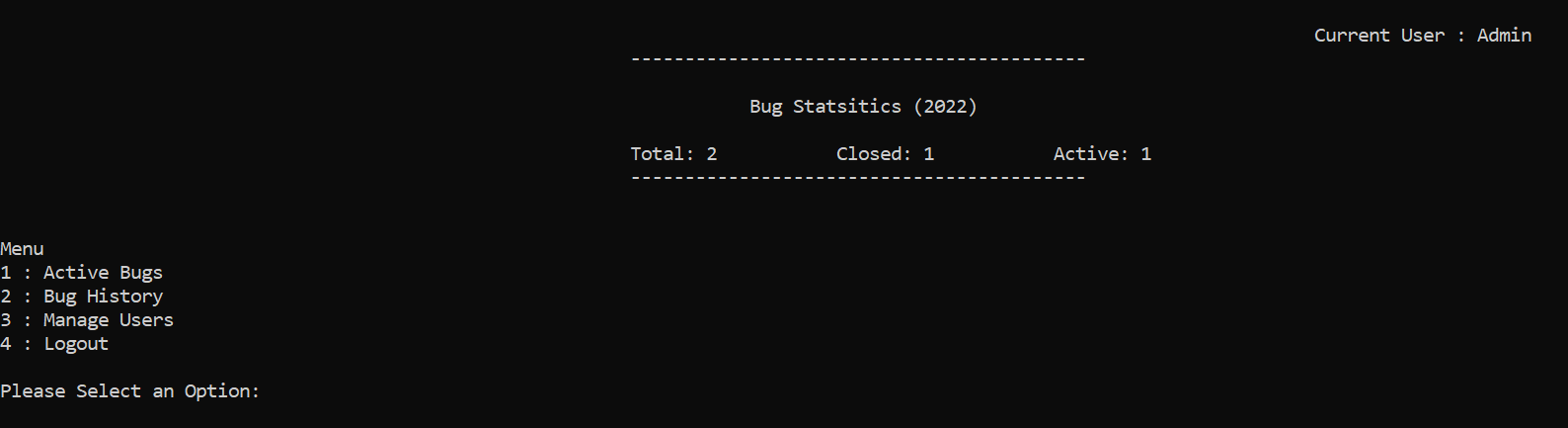
## Class Diagram

****

**Demo:**





****



