**Software Engineering**

**Software Requirements Specification**

**(SRS) Document**

**Java Bug Tracking Software**

**19/01/2021**

**1.0**

**Ben White**

|  |
| --- |
| **Revisions** |

| Version | Primary Author(s) | Description of Version | Date Completed |
| --- | --- | --- | --- |
| 1.0 | Ben White | First version of software with all functionality implemented and tested, all bugs addressed. |  |

|  |
| --- |
| **Table of Contents** |

[1. Introduction](#_Toc244519333) 2

1.1 Purpose 2

1.2 Document Conventions…………………………………………………………………….2

1.3 Intended Audience…………………………………………………………………………2

1.4 Scope……………………………………………………………………………………….2

[2. General Description 2](#_Toc244519334)

2.1 Product Perspective………………………………………………………………………...2

2.2 Product Features 2

2.3 User Class Characteristics 2

2.4 Operating Environment 2

2.5 Assumptions and Dependencies 3

[3. System Requirements 3](#_Toc244519335)

[4. External Interface Requirements](#_Toc244519336) 3

[4.1 User Interfaces](#_Toc244519337) 3

[4.2 Software Interfaces](#_Toc244519338) 3

[5. Non Functional Requirements 4](#_Toc244519341)

1. Introduction

* 1. **Purpose:** The completion of a comprehensive bug tracking system, allowing users to add any occurring bugs to the system and provide sufficient information on the bug so that another user could both understand the cause and fix the issue. This system should provide users with a platform to manage errors within their projects and systematically address them to aid the user maintain a high product quality.
  2. **Document conventions:** The term ‘bug(s)’ will be used to describe errors or exceptions. Each bug that is submitted with appropriate information will be classed as a ‘ticket’ awaiting resolution.

* 1. **Intended audience:** Includes any interested relevant parties such as software development companies, individuals who are undertaking software projects and myself as the developer.
  2. **Scope:** Software will align with goals of any users that wish to ensure high quality software deliverables and wish to accurately track issues during software production. The goal is to allow businesses to get a clear overview of their current issues and visually see the priority values of them.

## 2. General Description

**2.1 Product perspective:** Software will be deployed within a business or development environment and will be hosted on local servers and machines.

**2.2 Product features:** Software will include ability to save and search for bugs as well as an archive function to view resolved bugs from previous tickets. Creation of a bug ticket will enable various information to be supplied including the file that the bug originated from, description of what happened, severity or priority level and the date/time of the ticket

* 1. **User class and characteristics:** Primary users would be software developers who would be both logging and searching for bugs within the system, these users would be expected to interact with the system multiple times daily. Secondary users would include managerial staff or business owners who would use the system to check work progress on bugs and to also view number/ severity of issues the overall system is facing.
  2. **Operating environment:** Will operate in an office environment for use within software development teams.
  3. **Assumptions and dependencies:** Assumed access to MySQL database hosted on an accessible server to user/company or a locally hosted MySQL database that would enable individual use. Assumed users have passable understanding of technology and UI navigation.

## 3. System Requirements

**Functional requirements**

User should be able to make a new ticket for a bug, should be able to provide details as required for this ticket.

User should be able to search for a ticket through a priority or date order.

User should be able to search through currently under-development bugs and see who is working on them within the team.

User should be able to search for past bugs and see who was responsible for fixing them/ when they were completed.

The software automatically updates the priority level of tickets when a search is conducted.

Only managerial level staff can delete a ticket that has not been ‘completed’

System should use a MySQL database to store bug information.

System should implement a security framework (JAAS) to secure log in.

## 4.External Interface Requirements

4.1 User Interfaces

To be decided.

4.2 Software Interfaces

Front end –TBD

Backend will implement JAAS authentication frameworks and rest will be custom personally created code.

Server will implement MySQL and interface with app which will contain a management class for the database and will directly manipulate it as required.

Libraries include SAAS related libraries, TBD

The interaction of the software to be developed with other software components such as frontend and the backend framework to the used, the database management system and libraries describing the need and the purpose behind each of them.

## 5. Non-Functional Requirements

**Performance requirements**

Response time should be within 1 second for the lookup of any type of bug whether past or present and should be scalable to numbers within the tens of thousands in the case of long term, ongoing projects with an allowance of up to 2 seconds response for those larger data sets.

**Safety requirements**

Must ensure that database is secure and log in system is safe so as to reduce chance of misuse by outside parties to disrupt or damaged the software being developed by the user.

**Security requirements**

No real data protection issues. Only personal data to be used is name of developer who will supply a username and password to log in. Will ensure no extraneous or unnecessary data is collected.

**Software quality attributes**

Maintainability will be very important for long term modernisation and the code should be flexible and adaptable enough that it can be altered to allow for updated features. There must also be a high level of reliability as if there are problems with this during a company’s development lifecycle they could lose all bug tracking information which would not be acceptable.

Usability requirements are minimal and users are expected to have a basic grasp of technology and how to navigate a UI.

**Other requirements**

TBD

These may include the legal requirements, resource utilizations, future updates etc.