Table of Contents

1.Introduction

1.1 Purpose

This document describes the software requirements and specification for the system in the user and system level, detailed functional requirement are mentioned in the document. The requirement will be illustrated and presented with the help of diagrams are used to show complicated interactions.

The product environment consists of the following items: users:- people who use the system, network: – company intranet for secure communication, database: - main persistent storage for storing data, including projects, user and group information, bug reports, File system:- Additional storage format for keeping the exported queries and template.

1.2 Scope

The "**BUG TRACKING SYSTEM"** has been developed to override the problems prevailing in the practicing manual system. Bug and issue tracking systems are often implemented as a part of integrated project. Some bug trackers are designed to be used with the distributed revision control software. These distributed bug trackers allow bug reports to be conveniently read, added to the database or updated while a developer is offline. Recently, commercial bug tracking system have also begun to integrate with distributed revision control.

All type of bug tracking systems are conventionally viewed as a distinct types of software. Bugzilla is a type of bug tracking system which is non distributed. Bugzilla is currently supported by MySQL, PostgreSQL, Oracle and SQLite. Bugzilla is usually installed on Linux using Apache HTTP server.

1.3 Definition, Acronyms and Abbreviations

Definitions:

Bug: In the computer world, a bug is an error in a software program. It may cause a program to crash or show undesired events that results in the problem suffered by the users when they use the software.

Acronyms:

UC: Use Case

SRS: Software Requirements Specification

BTS: Bug Tracking System

1.4 Terminology

Authenticate User: The first thing required for the Bug Tracking System is to activate the login form. Login Page will help to know who logged into the software. It requires the username and password and then it lets the user to proceed.

View: It refers to the display view of the bugs in which user can easily monitor the bugs. It is of two type hierarchy in which bugs are shown in form of Parent/Child node and second one is Tree/Log view.

Logout: In this the user will be provided the option to save the file and safely logout from the software and reach to the first or main page of the software which is the login screen.

1.5 References

[1] Wikipedia: https://en.wikipedia.org/Bug_tracking_system

[2] Jonathan Corbet: https://lwn.net/Articles/281/849/

1.6 Overview

The rest of this SRS document contains all the requirements for the Bug Tracking System presented in several ways and organized into different section.

Section 2 contains general information that is not too specific and it provides a background for the following section. It contains description of all components of the Bug Tracking System its functions and constraints.

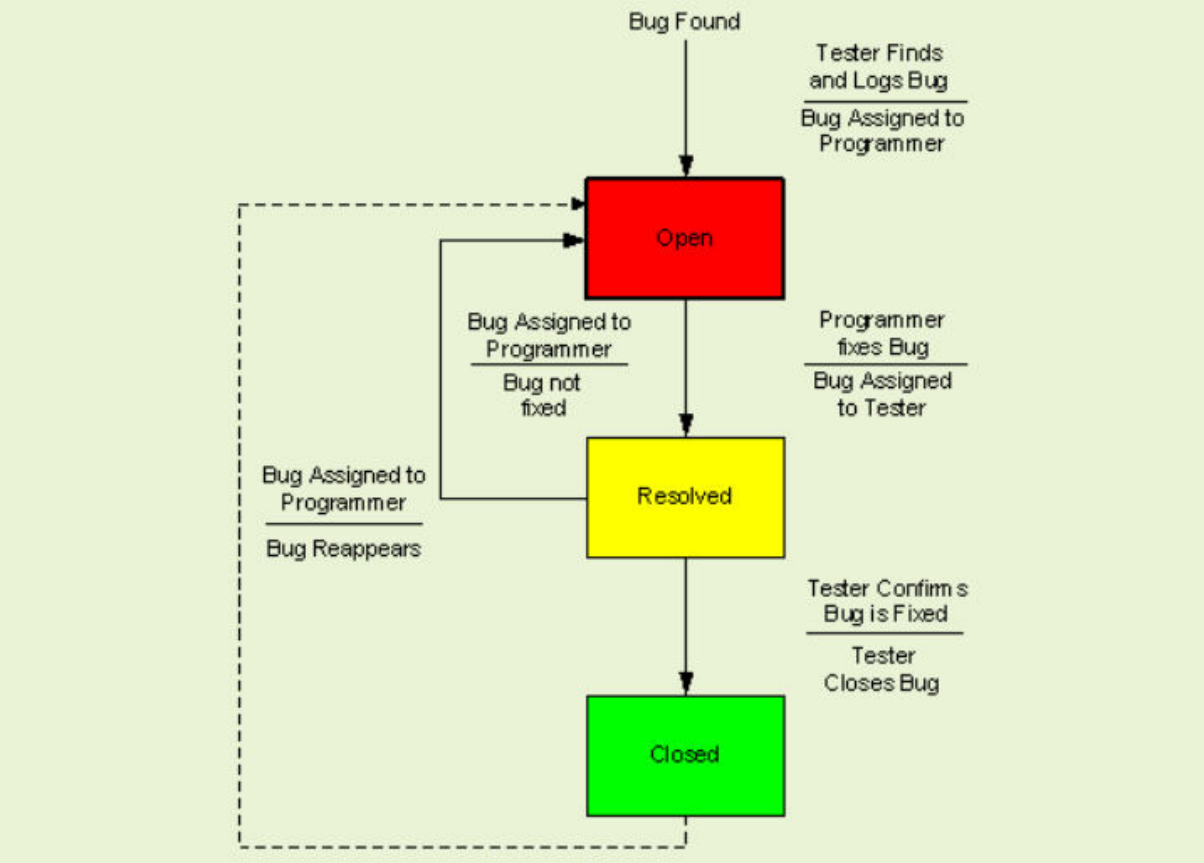
Section 3 contains the Specific requirements that includes Functional and Non Functional requirements of Bug Tracking System.

Section 4 contains External Interfaces which tells about the User interface and communication of software with the user.

Section 5 contains the Sequential Diagram and Use Case diagram.

Section 6 contains the 3 Test Cases of Bug Tracking System.

2.General Description



The product described in this SRS report is the software that can track the Bugs. This software is very much useful when it helps the user to find the flaws/bugs in their software. It normally provides a place to store the information about the reported bugs.

2.1 Product Perspective

Tracker tool is used for collecting bugs. Ticket management concept is used in the tracker tool for collection and solving of bugs.

Ticket Management: Ticket Management includes collecting ticket from users. Tickets collection means collecting bugs from the users.

Assignee: Assignee is a person who solves the bugs. So he is actual resolver of the bug. A bug can be solved by an individual person or by a group of user.

Bug Type: Bugs are of two types:

They are standard issue and all sub task issue.

- Standard issue type shows whether the bug is related to cloud bug or improvement related bug.
- Sub task issue type shows whether the bug is functional or technical.

Issue Status: In model of bug tracking system there are different type of issue status like,

- Open:- The bug is open for removing the problem.

- Reopened:- The bug was closed previously but opened again.
- Development in progress:- The bug solving is going on.
- Waiting for peer review:- The bug is solved and it is in queue to be reviewed by the tester.
- Problem during testing:- Problem is encountered while solving the bugs.
- Waiting for integration:- The bug needs to be integrated with other component and it is in waiting.
- Waiting for testing:- The bug is in queue for testing.

Description:- Provides complete description of the bug that occurs, output and expected output.

Affects Version:- Current version of software in which the bug is encountered.

Components:- Other component affected due to bug.

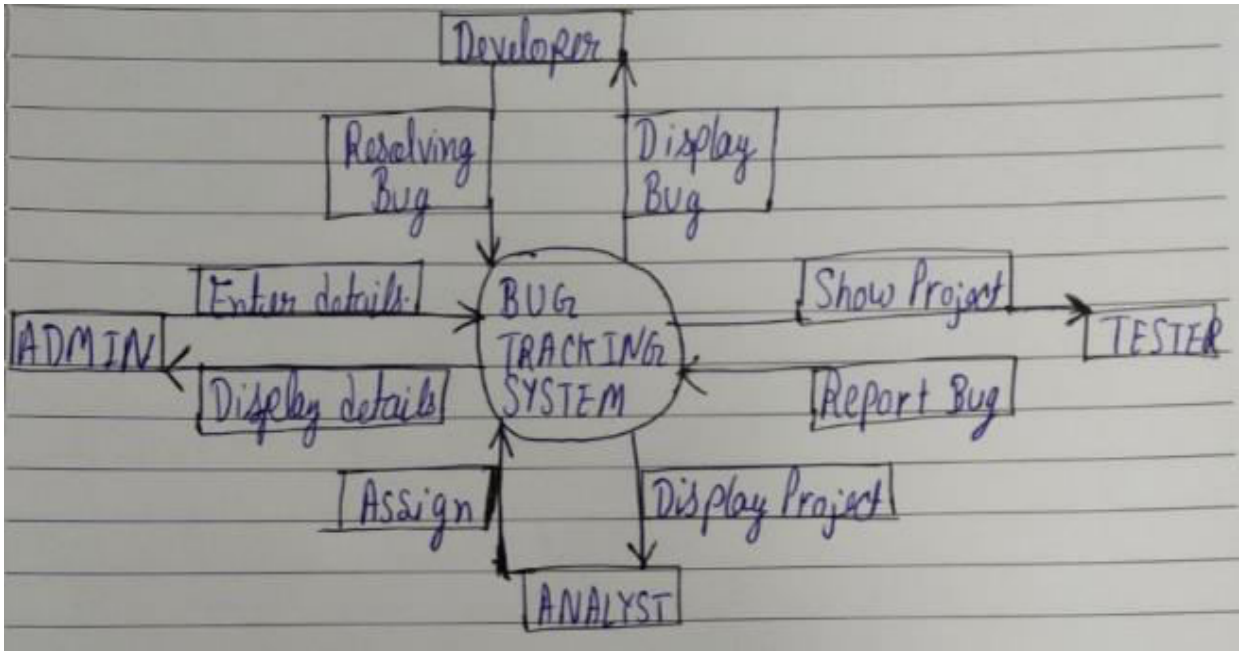
Security level:- It determine who can be the viewer of the bug stored in the software. If security level is high, then a very less number of people can view the bug. There are different type of security level like minor security issues, serious security issues.

Assigning priority to issue:- It determines the severity of the bug. Priority can be major, minor and critical.

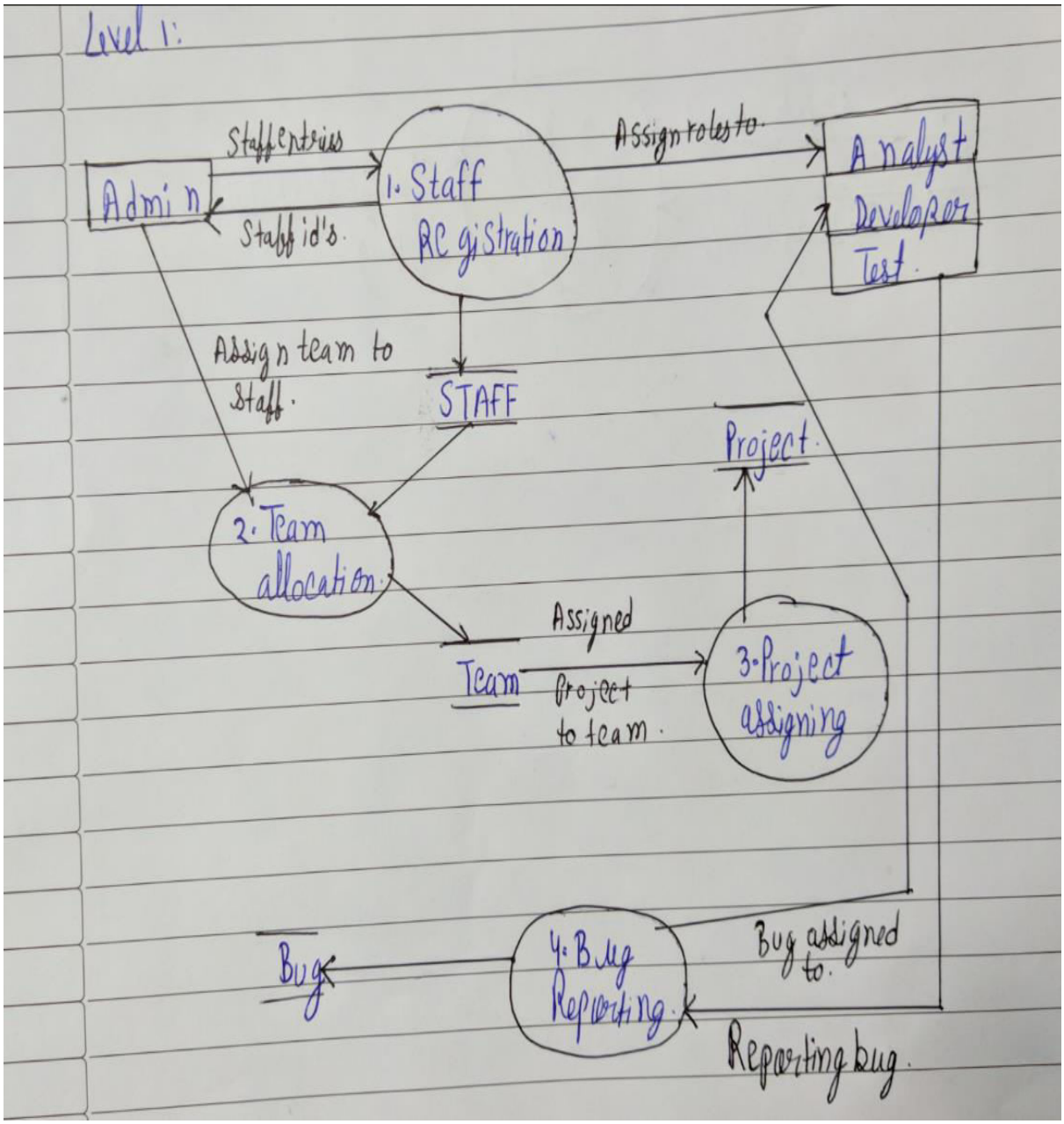
Attachment:- It allows the user to send file as an attachment to view the bug. It will help the developer to exactly view the bug.

Closing the bug:- It includes closing of the bug with notification like fixed, not fixed, not a bug, duplicate, serious issue, high priority, etc.

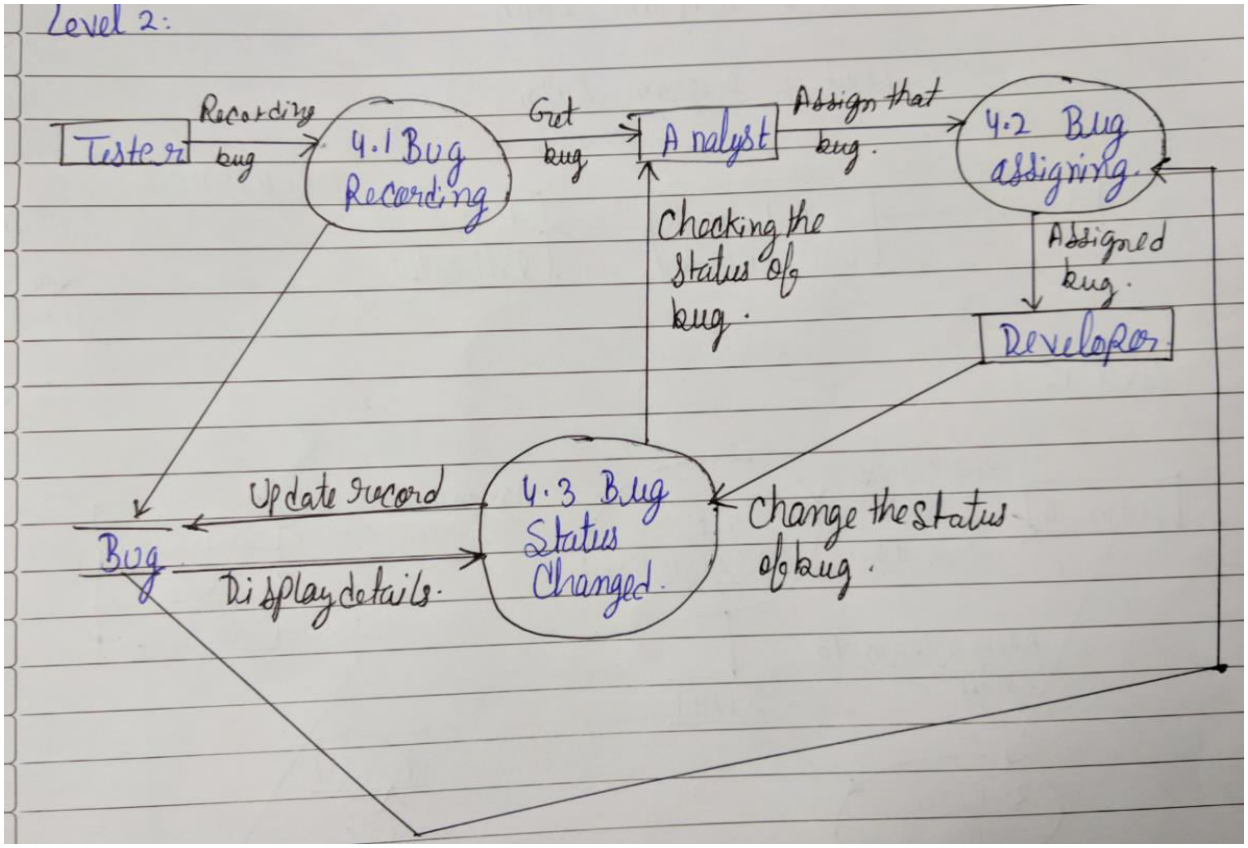
CONTEXT LEVEL DIAGRAM: [LEVEL 0]



CONTEXT LEVEL DIAGRAM: [LEVEL 1]



CONTEXT LEVEL DIAGRAM: [LEVEL 2]



2.2 Product Functions

The primary function of the Bug Tracking System is to track the bug in the project and store it in the database. Actually this software is the main tool for the bug tracking or managing teams. It is used by entering the bug information. The records in the bug tracking software are constantly updated to reflect the progress of work on the tracked bugs. By accessing the bug tracking software, a team can have the record of what is happening with the issue.

The benefit or importance of having a history of a bug is so that the progress can be monitored and if should something go wrong it can be tracked back to its origin or if a question arises then the team member will know who to ask because it also stores the name of the person who had handled the previous bug.

2.3 User Characteristics

The user of the Bug Tracking System is the other component of the software that interact with it through the user interface. The user should have the characteristics to update the bug correctly with full information and his name on it.

2.4 General Constraints

The Bug Tracking System is very essential tool when working on a large project with a lot of members in a team.

The software should follow:

- The fast and secure database so that the bug report should be updated easily and faster.
- Protection at Admin level, must have security for unauthorized access of data.
- Log generation for offline work, so that the logs are saved locally in computer when there is no internet connection and uploaded when internet is available.
- The user interface should also be user friendly so that everyone can use it.

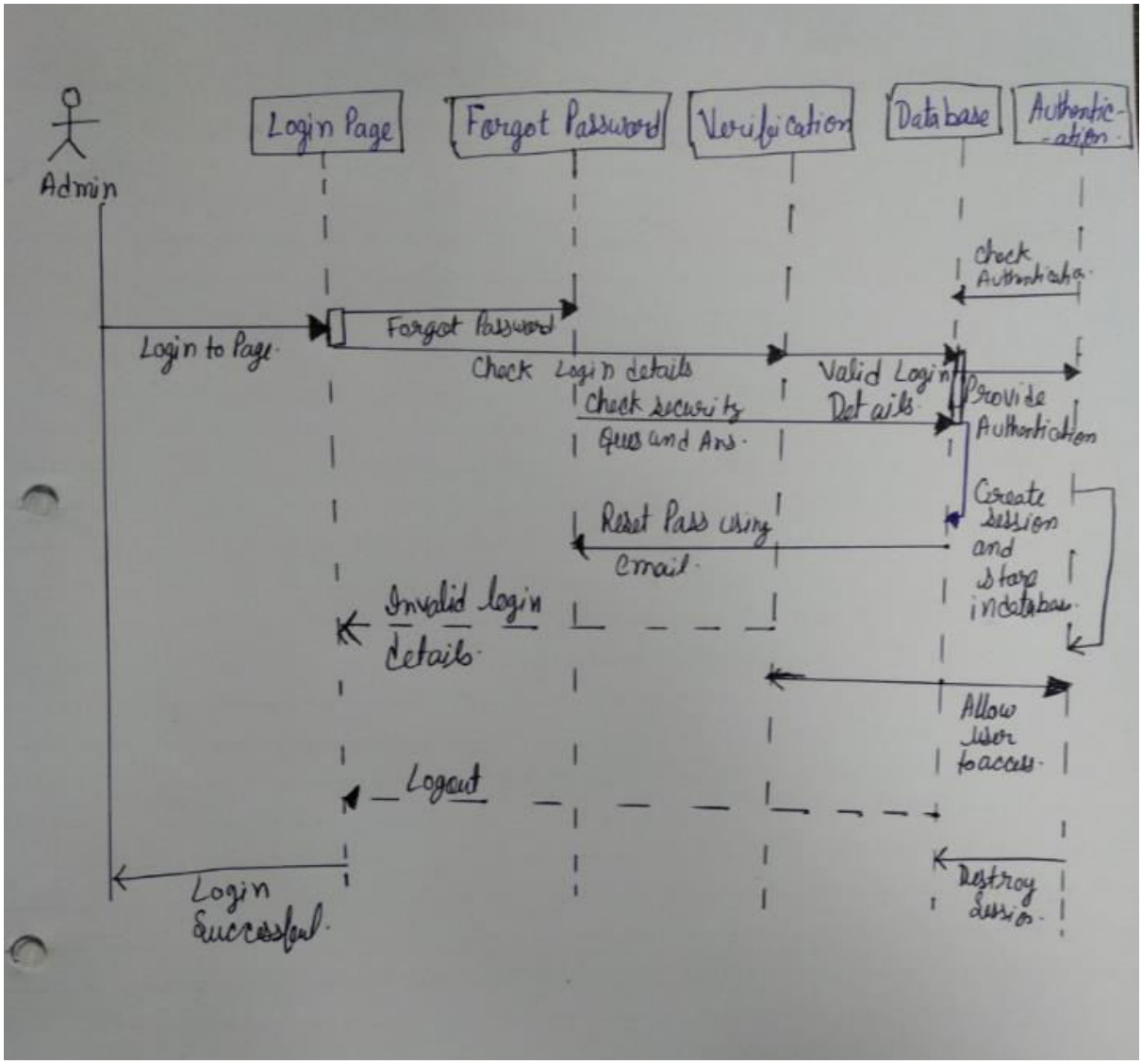
3.Specific Requirement

3.1 Functional Requirements

There are two main characters in Bug Tracking System Admin and the Manager.

• Admin: This module has the entire access to all other modules, admin creates the project and assigning the projects to the created manager, adding members to the managers, assigning bugs based on the priority.

• Manager: Manager has the full access to the particular project assigned by the admin and controls the team member’s access to the bugs assigned.



This is the login sequence diagram of the Bug Tracking System, where Admin will we able to login and select members and assign bus according to the priority. Members can work on the project of solving bugs after the Admin assign the task to them.

3.2 Non-Functional Requirements

This section defined the system and the constraints and properties of the products. Mainly these properties deal with the External requirements.

Product requirements

This product requirement is mainly focused on performance, reliability and portability properties of the product. The performance and reliability issues are closely related with the software and hardware that the product will run on. Any effort in increasing the system

throughout and improving the availability may result in needs for more powerful software or hardware.

Efficiency requirements

The system shall have capabilities of handling at least 50 mixed requests per second. For requests involving intensive processing and massive data transfers, the minimum number of requests per second will be 10. In case of 100 concurrent users using the system, the user should experience less than 10 seconds response time.

Reliability requirements

The system shall have a mean failure rate less than once per month. This rate counts both the software and hardware failures. The system shall be cluster ready for this very purpose.

Portability requirements

Due to the diversity of the client environment, the Bug tracking system must be portable. That is to say, it shall run on all modern operating systems and be able to interface major relational database systems from various vendors.

4.External Interface

4.1 User Interface

- User interface should be very easy so that anyone can use the software without having any kind of trouble in knowing the meaning of some terms.
- Language: Language support should be there so that it can be easily understood by the members who don't speak English language.
- Security: Every system should have the security feature to make the user data hack proof and leak-proof.
- Using wide range supported language: To make the software compatible with all platforms.

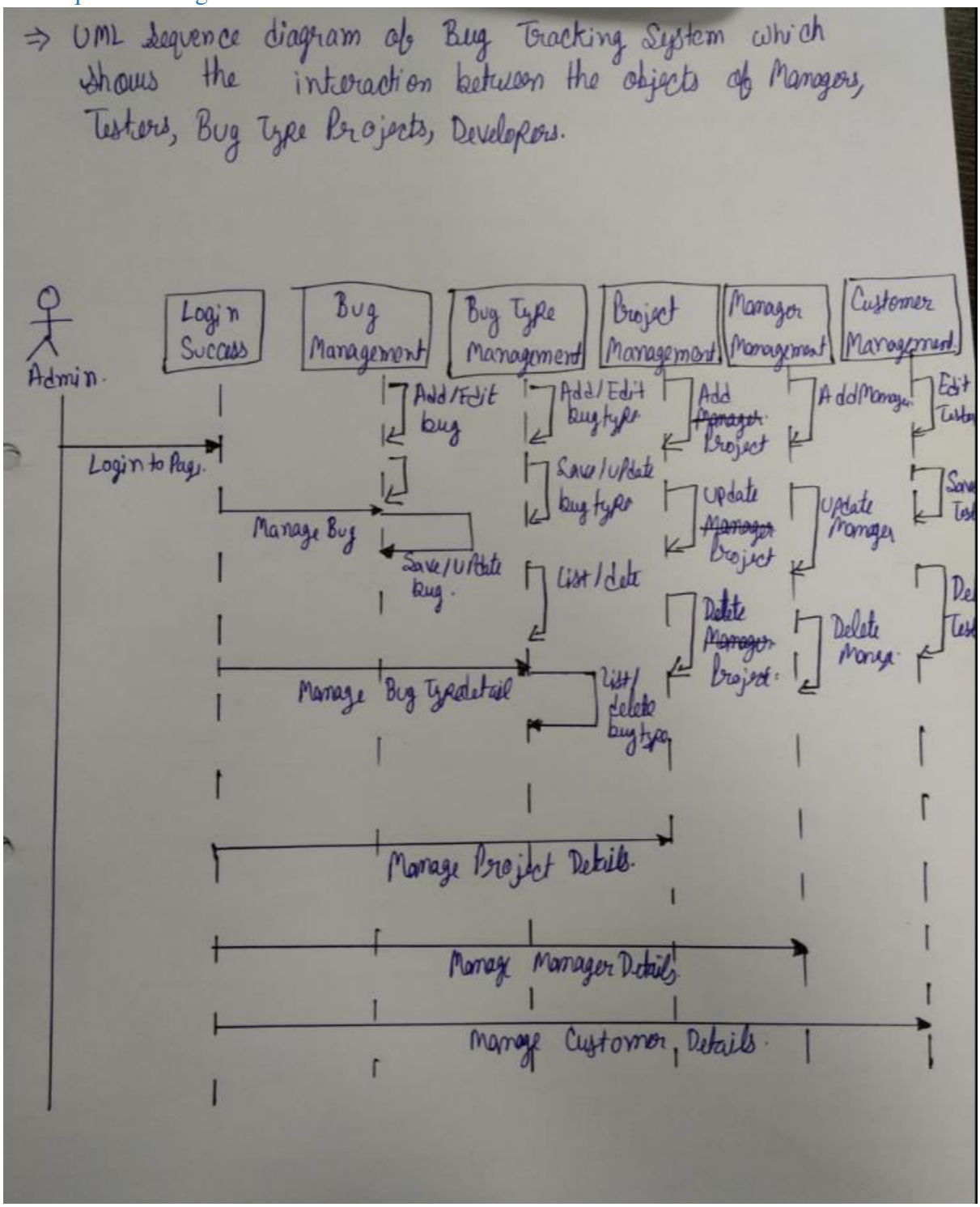
4.2 Communication interfaces

Software should use latest version of database and HTTPS protocol to maintain the communication between the devices.

A notification could be sent by email or sms using SMTP protocol.

5. Diagrams

5.1 Sequential Diagram



5.2 Use Case


Use Case diagram of Bug Tracking System:
Manage Users and full application
Manage bugs
Manage Project Manager
Login and Logout from System
Update my profile
Change acc. Password
Manage Tester
Manage Projects
Manage Developers
Change Status
Add comments
View Bugs
Assign bugs
Create bugs
Bug report
Super Admin
System user
Develop
Teste

6. Test Case

Sr no.	Input	Expected Output	Actual Output	STATUS
1.	Username: admin Password: admin	Popup message showing wrong Username or Password	Popup message showing wrong Username or Password	PASS
2.	Input numbers instead of name of user while creating new bug report	Number should not be inputted	Cannot enter numbers in username field.	PASS
3.	Turning off internet and submitting the file.	Log should be created and file should upload when internet restored.	Log Created and file uploaded on internet.	PASS