A PROJECT REPORT

on

“BUG TRACKING SYSTEM”

By

Team STAR TECHIES

Acknowledgement

We are profoundly grateful to our mentor Mrs. Shrilata T for standing with us and being a pillar of support for our group even in the hardest of times. Her masterful guidance and constant encouragement is a leading contributor to the success of the final package. And finally we would like to acknowledge our organization for giving us this opportunity.

Raga Rasagna Paruchuri

Gautami S Nair

Anindya Singhmar

Gaurav Pandey

Chaitanya Ashtekar

Vandana Bora

Lokinendi Sai Harshita

Anuraag Ajaykummar

Mounika Kalakonda

Avirupa Saha

Abstract

A bug is the consequence/outcome of a coding fault. A bug tracking system is a software application that keeps track of reported software bugs in software development projects.

The main benefit of a bug-tracking system is to provide a clear centralized overview of development requests including both bugs and improvements and their state. Also, it automates the process of monitoring and tracking bugs and other issues that may hamper the efficient performance of a company's technology and information infrastructure.

The proposed Bug Tracking System stores all data in a database and allows software testers to report bugs for a project. Project managers are able to view all the bugs, assign bugs to developers so that they can work on them and update the bug status.

Using this application effectively a team will be able to deliver high-quality products, improve customer satisfaction and the cost of product development will also be lowered.

List of Contents

List of Figures

1. Introduction

The Bug Tracking System can be used by the Project Managers, Testers and Developers of an organization to improve speed, efficiency and collaboration. Thus, enforcing quality assurance and performance tracking.

This application can have only three kinds of users- project managers, testers and developers. Each of them has a unique id, email and name. It stores all the data in a database by importing the user information from a JSON file. The system can store multiple projects which are created by the project manager and every project has a team of members consisting of a single project manager, multiple developers and a single tester. A tester can report a bug having a unique set of attributes. A project manager can assign bugs to developers. After the defect is resolved, a developer can close the bug.

Thus, this bug tracking system can effectively manage and store information related to the project and the team members giving an insight into the next set of actions or bugs which need to be resolved.

2. Software Requirement Specification

2.1 User Interface

There are three types of users in this system, along with an admin.

- **Project manager-** The Project Manager is the one who handles the team and the status of the bug
- **Tester-** The testers are the users who report bugs of the project to the developers.
- **Developer-** The developer is the one who resolves the bug and sends the completion report through the system.

2.2 Hardware Requirements

- **Processor** - Intel Core i3 and above
- **OS** - Windows 10
- **Memory(RAM)** - 4 GB and above
- **Storage** - 50 GB and above

2.3 Software Requirements

- **User Interface** - JSP, Bootstrap, CSS
- **Client-side Scripting** - Javascript
- **Programming Language** - Java
- **Web Application**s - JDBC, Servlets
- **Database** - MySQL
- **IDE** - Spring Tools Suite or Eclipse Neon, Visual Studio Code, MySql Workbench
- **Server** - Apache Tomcat 8.5

2.4 Functional Requirements:

The Bug Tracking System permits three kinds of users on its application. Every user type has a home page based on their role.

- **Project Manager**
  a. Create project
  b. Form Team

c. Assign and Close Bug

- **Tester**

  a. Report bugs

- **Developer**

  a.Resolve Bugs
  b.Mark Bugs for Closing

2.5 Non-Functional Requirements

- **Performance -** The Bug Tracking System, provides an easy and convenient UI that enhances the performance and responsiveness of the application.
- **Security -** The application denies any unauthorized individual outside of the organization to register themself. Only the authorized person can register on the application using their email id and Role. The password is highly case and format sensitive. The Registered User can login using his email id and password only. The password is encrypted before we store it to avoid any anonymous breach
- **Capacity -** The application is designed in such a way that it can handle huge databases from the local system as well as from the cloud.
- **Reliability -** Once a user logs in to the system he or she can access all the functions on his page without any session drop. In case of inactivity for a considerate time he is logged off the session and needs to login again.
- **Maintainability** - The admin can comfortably debug the issues arising as all the test cases are written in simple if-else form.
- **Usability -** The bug Tracking System has a user friendly interface. It has convenient options which facilitates easy flow of instructions and commands.
- **Portability -**Java programs are portable, which means that the same bytecode program can run on any computer system that has a Java interpreter. The source program of the application does not have to be changed to meet the particular needs of a particular computer system.

3. System Design

3.1 Introduction

System Design Software design sits at the technical kernel of the software engineering process and is applied regardless of the development paradigm and area of application. Design is the first step in the development phase for any engineered product or system. The designer's goal is to produce a model or representation of an entity that will later be built.

Model–view–controller (MVC) is a software design pattern commonly used for developing user interfaces that divide the related program logic into three interconnected elements. This is done to separate internal representations of information from the ways information is presented to and accepted from the user.

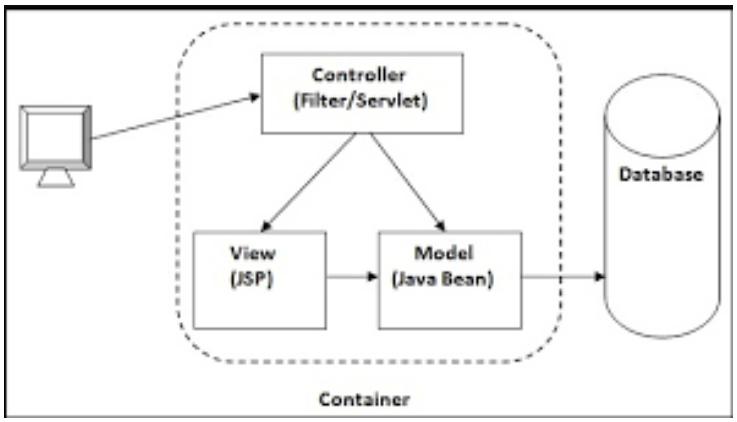

Fig 1. MVC Architecture Layout

3.2 Data Flow Diagram

A data flow diagram is a graphical representation of the flow of data through an information system. A data flow diagram can also be used for the visualization of the data processing.

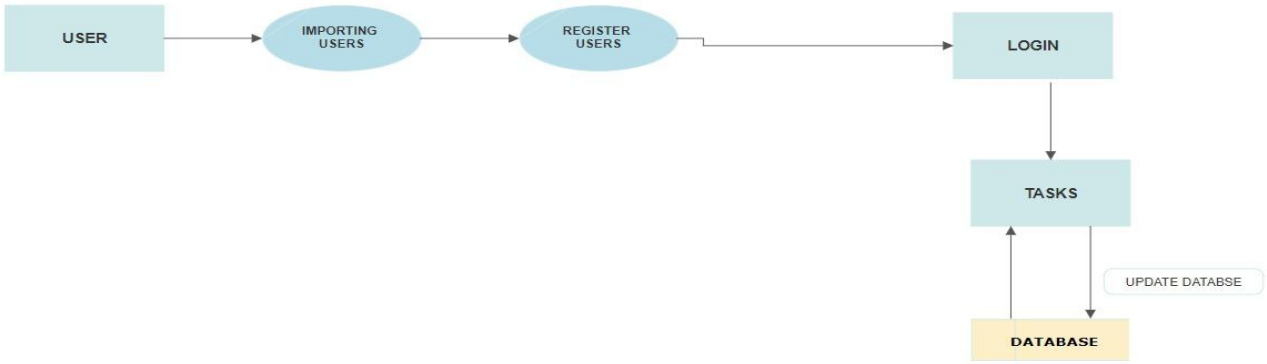

Fig 2. Level 0 DFD

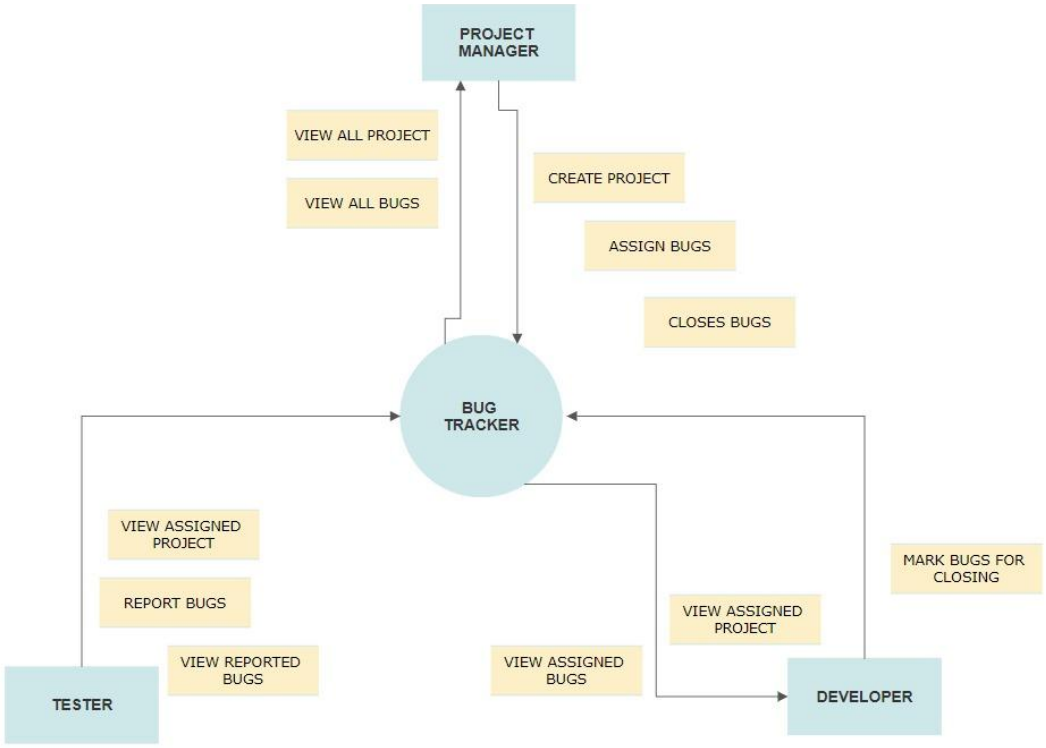

Fig 3. Level 1 DFD

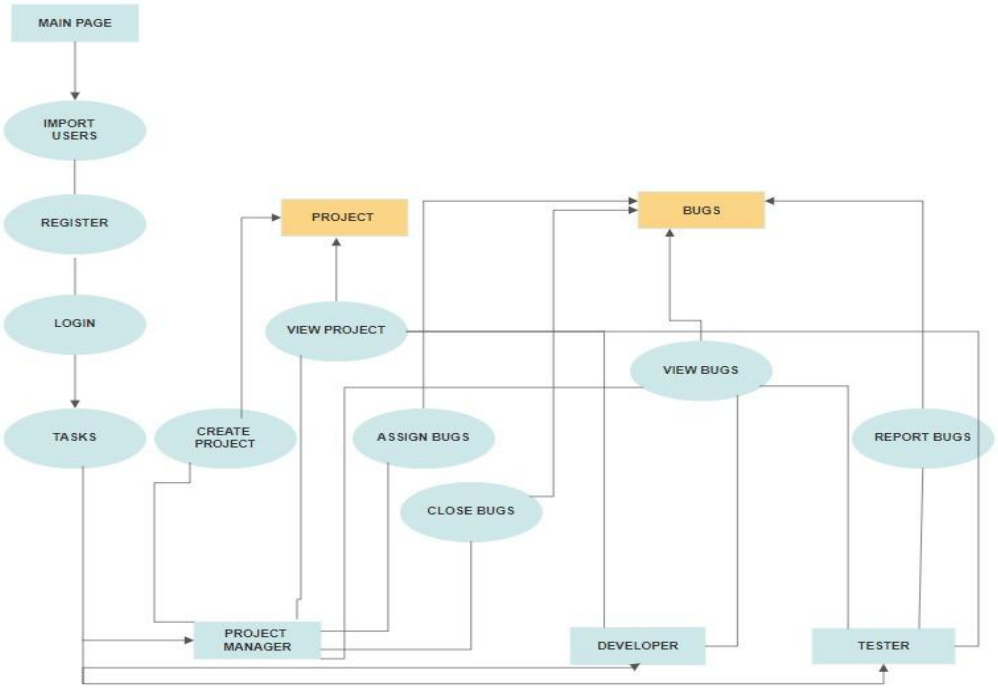

Fig 4. Level 2 DFD

3.3 Use Case diagram

Use case diagrams consist of actors, use cases and their relationships. The diagram is used to model the system/subsystem of an application. A single use case diagram captures a particular functionality of a system.

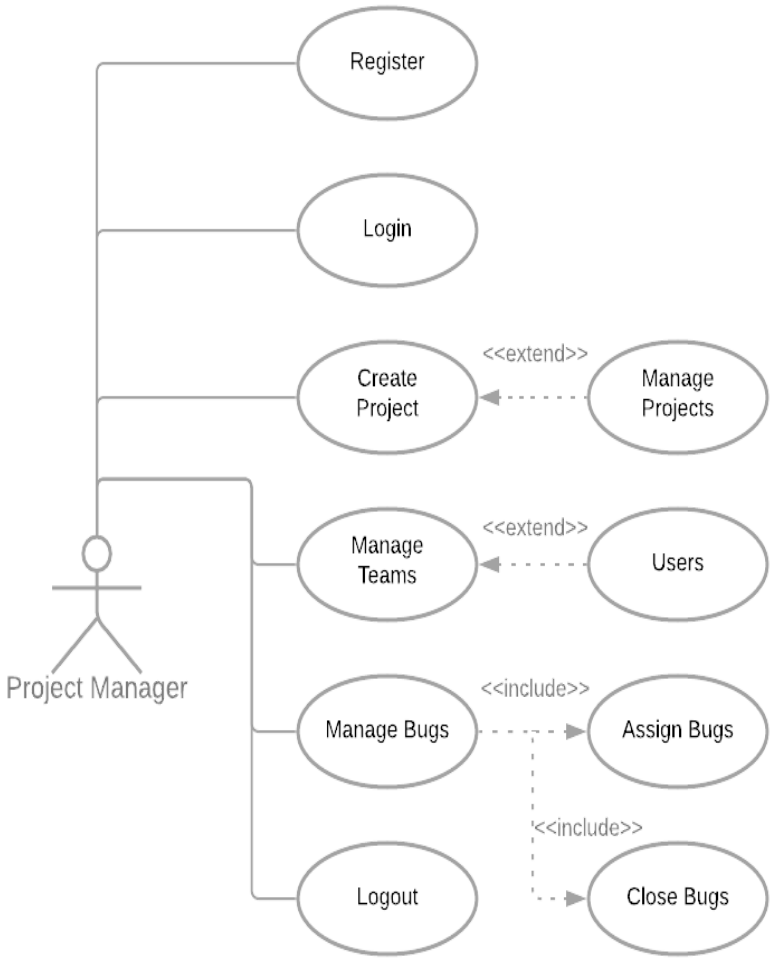

Fig 5. Project Manager Use Case Diagram

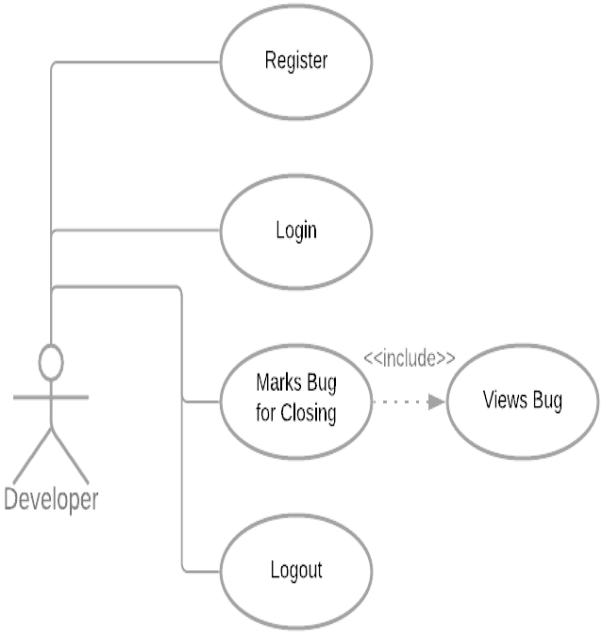

Fig 6. Developer Use Case Diagram

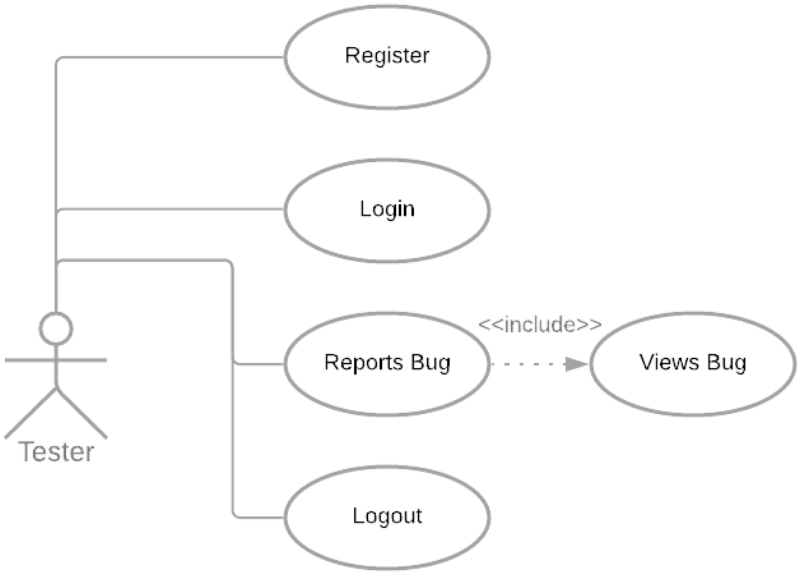

Fig 7. Tester Use Case Diagram

3.4 Class diagram

The class diagram depicts a static view of an application. It represents the types of objects residing in the system and the relationships between them. A class diagram is used to visualize, describe, document various different aspects of the system, and also construct executable software code.

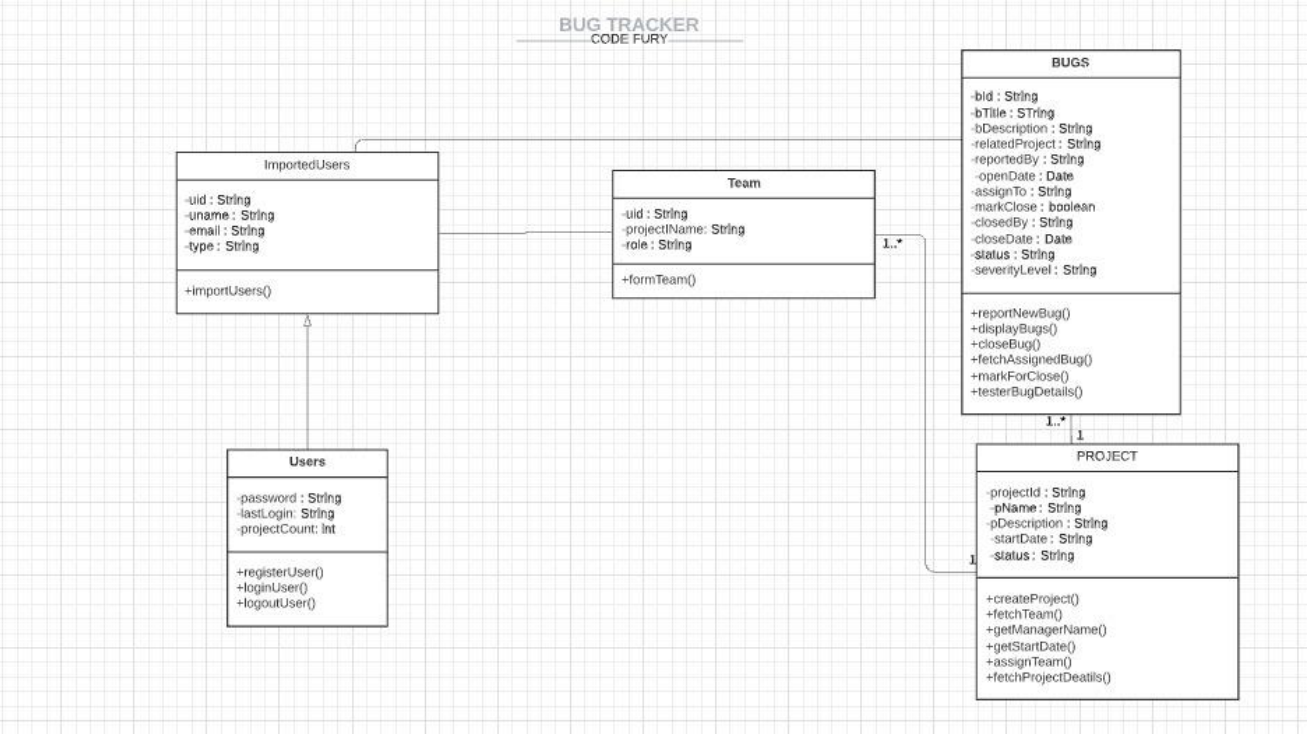

Fig 8. Class diagram

3.5 ER diagram

An entity-relationship (ER) diagram is a specialized graphic that illustrates the relationships between entities in a database. ER diagrams often use symbols to represent three different types of information.

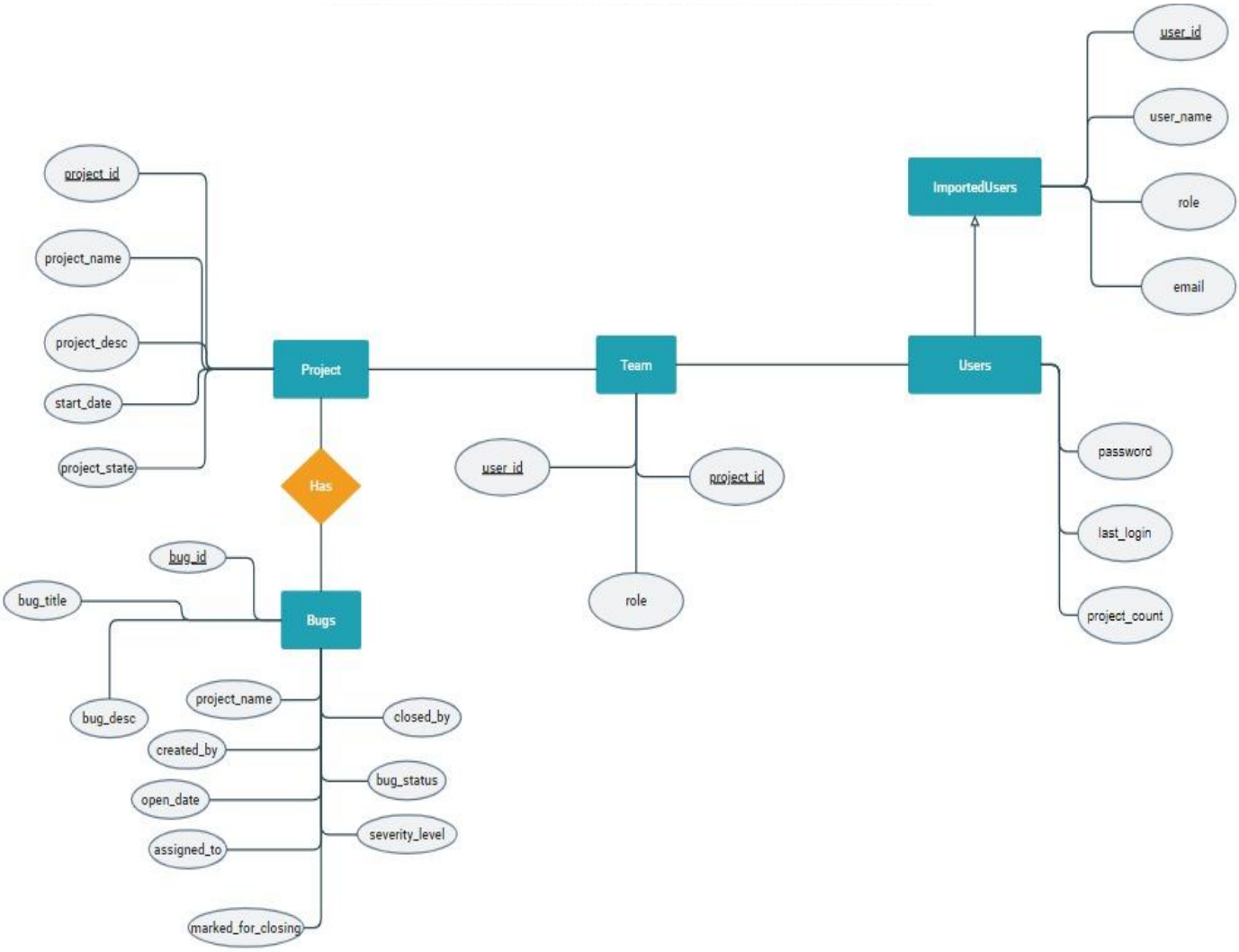

Fig 9. ER Diagram

4.Testing

4.1 Introduction

Testing is the major quality control measure used during software development. It is a basic function to detect errors in the software.Testing determines whether the system appears to be working according to the specifications. It is the phase where we try to break the system and we test the system with real case scenarios at a point.

4.2 Levels of Testing

4.2.1 Unit Testing

The unit testing of the source code is to be done for every individual unit of the module that was developing part of the system and some errors were found for every turn and rectified. This form of testing was used to check for the behavior signifying the working of the system in a different environment as an independent functional unit.

4.2.2 Integration Testing

From the individual parts to the cohesion of each part to make the system as a whole, there is a need to test the working between the assembled modules of the system. The testing process is concerned with finding errors that result from unanticipated interaction between the sub-system and system component. It is also concerned with validating the system meets its functional and non-functional requirements.

4.2.3 System Testing

The requirement specification document that is the entire system is to be tested to see whether it meets the requirement or not.

4.3 Test Cases

TEST CASE ID	TEST SCENARIO	TEST CASE	TEST STEPS	TEST DATA	EXPECTED RESULT	POST CONDITION	ACTUAL RESULT	REMARKS/STATUS
TC_32_DEVELOPER PAGE	**Main Page** Validation	Check the options on the main page of Developer	1.**Click** on Login on the HomePage of application 2.**Enter** the Login Credential 3.**Click** on Profile 4.**Click** on Project Assigned	Email:user1@gmail.com Password:Iam@User1	The Homepage of Developer must be functional .The homepage should display a.Email Id b.Role c.Last logged in as initial view		The Homepage of Developer must be functional .The homepage should display a.Email Id b.Role c.Last logged in as initial view	Pass
TC_33_DEVELOPER PAGE	**Project Assigned Button** Validation	Check the Project Assigned Button on the Developer Home Page	1.**Click** on **Login** on the HomePage of application 2.**Enter** the **Login Credential** 3.**Click** on **Project Assigned**	Email:user1@gmail.com Password:Iam@User1	The Project Assigned Page should display the following a.Project Name b.Project manager c.Start date d.Team members &their role e.Bugs Name Assigned to him f.Bugs Status	The Developer can change the Bug Status after working on it	The Project Assigned Page should display the following a.Project Name b.Project manager c.Start date d.Team members &their role e.Bugs Name Assigned to him f.Bugs Status	Pass
TC_34_DEVELOPER PAGE	**Log Out Button** Validtion	Check the Log Out Button if it is taking the Developer to the home page of application	1.**Click** on **Login** on the HomePage 2.**Enter** the **Login Credential** 3.**Click** on **Log Out**	Email:user1@gmail.com Password:Iam@User1	The Log Out Button sends the user to the HomePage		The Log Out Button sends the user to the HomePage	Pass

Fig 10. Developer Page Test case

TEST CASE ID	TEST SCENARIO	TEST CASE	TEST STEPS	TEST DATA	EXPECTED RESULT	POST CONDITION	ACTUAL RESULT	REMARKS/STATUS
TC_27_TESTER PAGE	**Main Page** Validation	Check the options on the main page of Project manager	1.**Click** on **Login** on the HomePage of application 2.**Enter** the Login Credential 1.**Click** on Report Bugs 2.**Click** on Project Detail. 2a.**Click** on project name for project detail	Email:user3@gmail.com Password:Iam@User3	The Homepage of Tester must be functional The homepage should display a.Email Id b.Role c.Last logged in		The Homepage of Tester must be functional The homepage should display a.Email Id b.Role c.Last logged in	Pass
TC_28_TESTER PAGE	**Report Bug Button** Validation	Check the Report Bug Button on the Tester Home Page	1.**Click** on **Login** on the HomePage of application 2.Enter the Login Credential 3.**Click** on **Report Bug Button**	Email:user3@gmail.com Password:Iam@User3	Report Bugs Main page should open	The Tester can make the required ammendment on the Report Bug page	Report Bugs Main page should open	Pass
TC_29_TESTER PAGE	**Report Bug Button** Validation	Check the Report Bug Button on the Tester Home Page	1.**Click** on **Login** on the HomePage of application 2.**Enter** the Login Credential 3.**Click** on **Report Bug Button** 4.**Select** the **Project name** from the dropdown list of allotted projects 5.**Enter** the **Bug Name** 6.**Enter** the **Bug Description** 7.**Select** the **severity level** of Bug 8.**Click** on **Submit**	Email:user3@gmail.com Password:Iam@User3 Project Name:NewTest Project Bug Name:Test Bug Bug Description:This is a test Bug Severity Level:Major	The Details are submitted and the tester is directed to the project details page,where he/she can check the details submitted		The Details are submitted and the tester is directed to the project details page,where he/she can check the details submitted	Pass
TC_30_TESTER PAGE	**Project Details Button** Validation	Check the Project Details button	1.**Click** on **Login** on the HomePage 2.**Enter** the **Login Credential** 3.**Click** on **Project Details Button** 4.**Click** on **Project Name**	Email:user3@gmail.com Password:Iam@User3	A Projects details page is displayed which contains the data received from Report Bug.i.e Project Name Bug Name, Description ,	The Tester can view the project details by clicking on any project name	A Projects details page is displayed which contains the data received from Report Bug.i.e Project Name Bug Name, Description ,	Pass
TC_31_TESTER PAGE	**Log Out Button** Validtion	Check the LogOut Button if it is taking the Project Manager to the home page of application	1.**Click** on **Login** on the HomePage 2.**Enter** the **Login Credential** 3.**Click** on **Log Out**	Email:user3@gmail.com Password:Iam@User3	The Log Out Button sends the user to the HomePage		The Log Out Button sends the user to the HomePage	Pass

Fig11. Tester Page Test case

A	B	C	D	E	F	G	H	I
TEST CASE ID	**TEST SCENARIO**	**TEST CASE**	**TEST STEPS**	**TEST DATA**	**EXPECTED RESULT**	**POST CONDITION**	**ACTUAL RESULT**	**REMARKS/STATUS**
TC_20_PROJECT MANAGER PAGE	**Main Page** Validation	Check the options on the main page pf Project manager	1.Click on Login on the HomePage 2.Enter the Login Credential 3.Click on New Project 4.Click on Project Detail. 4a.Click on project name for project detail	Email:user2@gmail.com Password:Iam@User2	The Homepage of Project Manager must be functional The homepage should display a.Email Id b.Role c.Last logged in as initial view		The Homepage of Project Manager must be functional The homepage should display a.Email Id b.Role c.Last logged in as initial view	Pass
TC_21_PROJECT MANAGER PAGE	**New Project Button** Validation	Check the New Project Button on the Project Manager Home Page	1.**Click** on **Login** on the HomePage 2.**Enter** the **Login Credential** 3.**Click** on **New Project Button**	Email:user2@gmail.com Password:Iam@User2	The Create New Project Page opens	The New Project details are ammended	The Create New Project Page opens	Pass
TC_22_PROJECT MANAGER PAGE	**New Project Button** Validation	Check the New Project Button on the Project Manager Home Page	1.**Click** on **Login** on the HomePage 2.**Enter** the **Login Credential** 3.**Click** on **New Project Button** 4.**Enter** Project Name 5.**Enter** Submission Date 6.**Enter** Project details 7.**Assign** Team Members	Email:user2@gmail.com Password:Iam@User2 Project Name:NewTest Project Start Date:27/09/2021 Description:This is a test project Select the team from the Developer checkbox and Tester Radio button	We are directed to the Project Details Page	The Project Manager can view about the project		Pass
TC_23_PROJECT MANAGER PAGE	**Project Details Button** Validation	Check the Project Details button	1.**Click** on **Login** on the HomePage 2.**Enter** the **Login Credential** 3.**Click** on **Project Details Button**	Email:user2@gmail.com Password:Iam@User2	A Projects page with the serial wise name of projects appears	The Project Manager can view the project details by clicking on any project name	A Projects page with the serial wise name of projects appears	Pass
TC_24_PROJECT MANAGER PAGE	**Project Details Button** Validation	Check the Project Details button	1.**Click** on **Login** on the HomePage 2.**Enter** the **Login Credential** 3.**Click** on **Project Details Button** 4.**Click** on **Project name** to view Project Details	Email:user2@gmail.com Password:Iam@User2	The Project Details page is open a.Displays Project Name b.Start Date c.Project Manager d.Team Members e.Bug List	The Project Manager can now check the Bug here and assign the Bug to a Develper.He/She can also check the Present Status of the Project and even close it once resolved.	The Project Details page is open a.Displays Project Name b.Start Date c.Project Manager d.Team Members e.Bug List	Pass
TC_25_PROJECT MANAGER PAGE	**Project Details Button** Validation	Check the Project Details button and assign Bug to the developer.	1.**Click** on **Login** on the HomePage 2.**Enter** the **Login Credential** 3.**Click** on **Project Details** Button 4.**Click** on **Project Name** to view project details 5.**Select** severity of the Bug in the Bug list 6.**Assign** the bug to a developer through the dropdown 7.**Check** the project status 8.**Click** on **close** after the bug resolved.It will change the Project Status to complete	Email:user2@gmail.com Password:Iam@User2 Bug Severity:Critical Assign bug:user1	The Project Details page is open a.Displays Project Name b.Start Date c.Project Manager d.Team Members e.Bug List f.The Severity of the Bug is set g.The Bug is assigned to a developer h.The project status can be changed to complete by clicking the Close button	The project status can be changed to complete by clicking the Close button.	The Project Details page is open a.Displays Project Name b.Start Date c.Project Manager d.Team Members e.Bug List f.The Severity of the Bug is set g.The Bug is assigned to a developer h.The project status can be changed to complete by clicking the Close button	Pass
TC_26_PROJECT MANAGER PAGE	**Log Out Button** Validtion	Check the LogOut Button if it is taking the Project Manager to the home page of application	1.**Click** on **Login** on the HomePage 2.**Enter** the **Login Credential** 3.**Click** on **LOG OUT**	Email:user2@gmail.com Password:Iam@User2	The Log Out Button send the user to the HomePage		The Log Out Button send the user to the HomePage	

Fig 12. Project manager Test case

5. Project Snippets

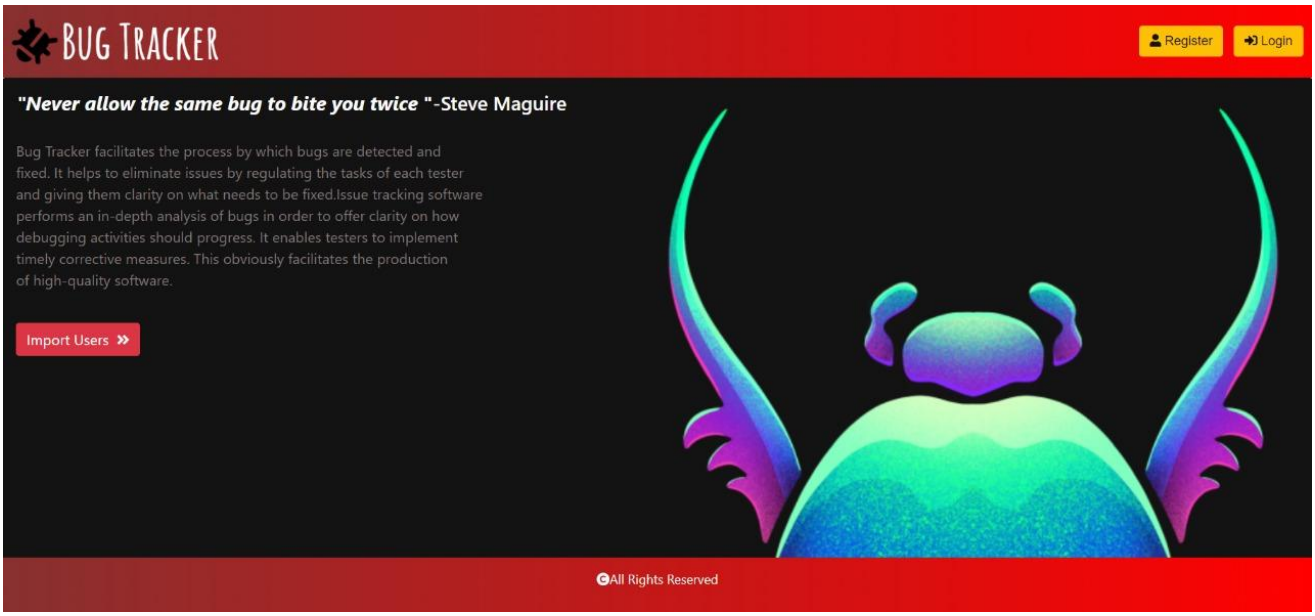

Fig 13. Main Page

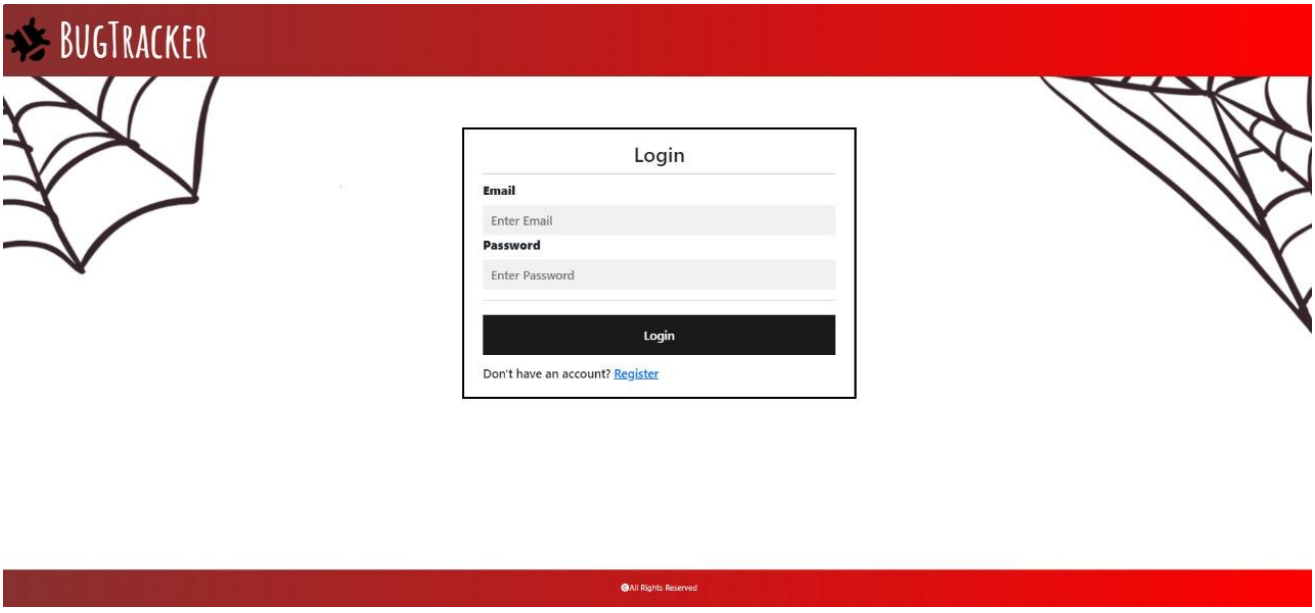

Fig 14. Login Page

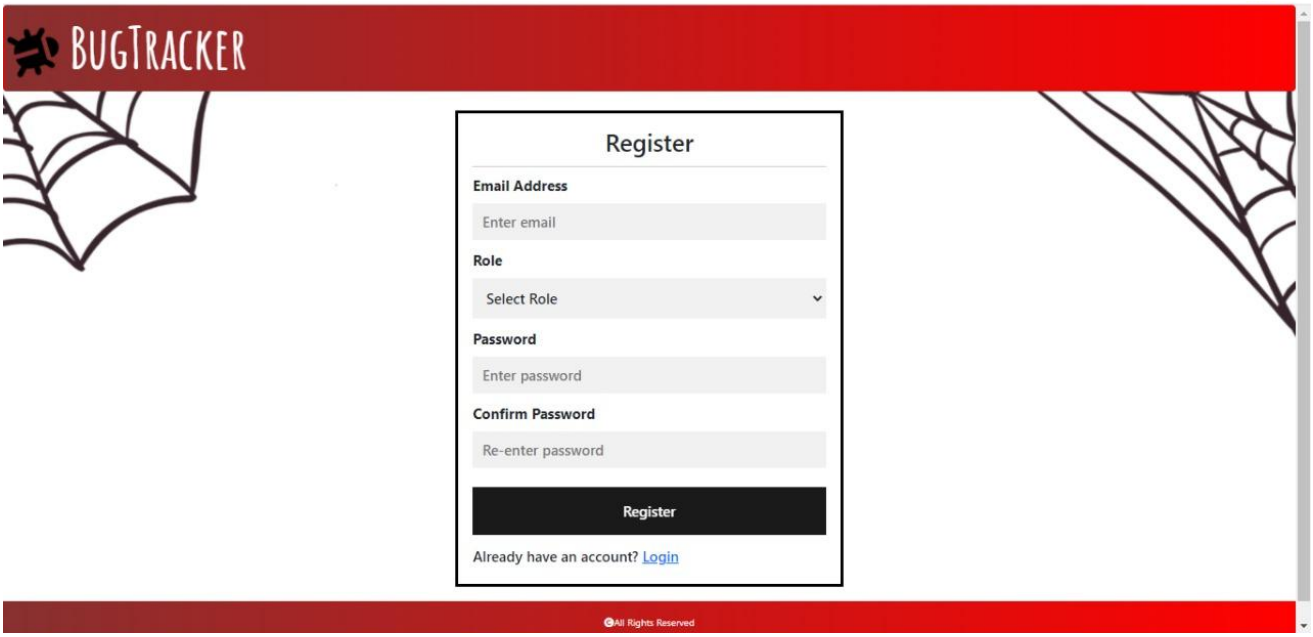

Fig 15.Register Page

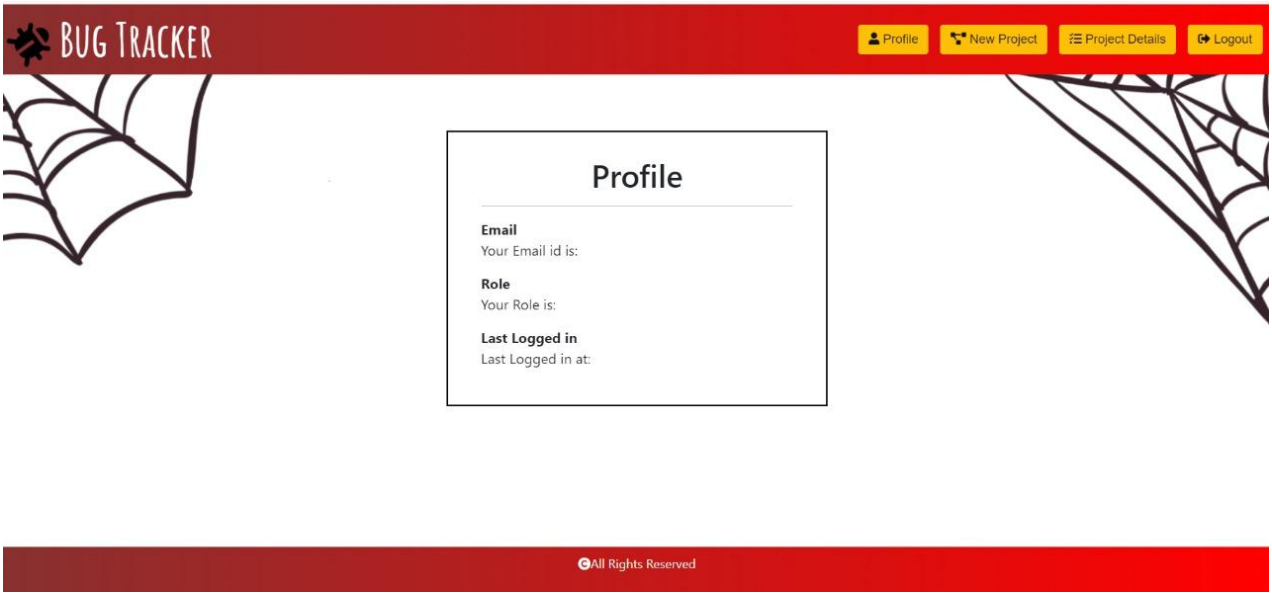

Fig 16. Project manager HomePage

6. Conclusion

A software bug occurs when an application or program doesn't work the way it is designed to function. Most errors are faults or mistakes made by system architects, designers or developers. Testing teams use bug tracking to monitor and report on errors that occur as an application is developed and tested.

The proposed Bug Tracking system is a great application with software development needs. By using the Bug Tracking Software one can eliminate bugs in their project effectively as it contains all the necessary information required by the developers to resolve the issue. The priority list based on severity level of bugs when defining product roadmap or just the next release.
This application will definitely increase communication between the team members and improve the customer satisfaction and efficiency of the application.

7. Future Scope

The application can be further improved by improving it's UI so that the team can prioritize and discuss their team's work in full context with complete visibility. The roles of each user can be increased by providing more functionalities like checking the code that is corrected by the developer by accessing the link directly. The security can be enhanced by restricting the people who can upload the JSON file. Also, an online discussion forum can be created from which a developer will be able to remove commonly known types of bugs.