IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 1, April 2026
Impact Factor: 8.2

Bug Tracking System

Patel Aman Roshan[1], Jagtap Rushikesh Yashawant[2], Bidwe Pratik Ramesh[3], Jadhav Vaibhav Ambadas[4]
[1,2,3,4]Student, Diploma in Computer Engineering
Vishweshwarayya Institute of Engineering and Technology, Almala, Maharashtra, India

Abstract: *This project titled "Bug Tracking System" focuses on the design and development of a software application that helps development teams manage and track software bugs effectively during the software development lifecycle. The system provides a centralized platform where testers, developers, and administrators can report, assign, monitor, and resolve software defects in an organized manner.*
The system is developed using Java for the backend, MySQL for database management, and a graphical user interface using Java Swing to ensure easy interaction between users and the system. The application allows testers to report bugs, developers to resolve issues, and administrators to manage the entire bug tracking workflow.
The main objective of the Bug Tracking System is to improve the efficiency of software development by ensuring that bugs are properly recorded, assigned to the responsible developers, and resolved in a timely manner. Compared to traditional manual bug tracking methods, this system provides better organization, improved communication among team members, and faster debugging processes.
The system demonstrates the practical implementation of software engineering principles and improves software quality by maintaining a structured and transparent bug management process.

Keywords: Bug Tracking System, Software Testing, Defect Management, Software Development Lifecycle, Java, MySQL

I. INTRODUCTION

BUG TRACKING SYSTEM – In software development, keeping software products high-quality and reliable is crucial. Creating software applications involves complex processes like coding, testing, debugging, and deploying. During these stages, errors or defects, known as bugs, often appear in the system. A bug is a flaw or error in a software program that causes the system to give incorrect results, act unpredictably, or fail to perform as intended. If bugs aren't found and fixed promptly, they can lower software performance, create security risks, and harm user experience.

In the past, tracking software bugs relied on manual methods like spreadsheets, emails, or written documentation. These older methods are slow, inefficient, and prone to mistakes, especially for large and complicated software projects. In today's software development environments, many developers, testers, and project managers work together at the same time. This makes it hard to keep track of bugs and manage them well without a proper system. This challenge led to the creation of automated Bug Tracking Systems that organize and manage software defects more effectively.

A Bug Tracking System is a software application that records, manages, and monitors software defects throughout the development lifecycle. It offers a centralized place where team members, such as testers, developers, and administrators, can work together easily. Testers can report newly found bugs, developers can review and fix assigned issues, and administrators can oversee the whole defect management process. The system makes sure that each bug is documented with important details like a description, priority level, status, date reported, and the developer responsible for fixing it.

The main goal of the Bug Tracking System is to improve software development efficiency by keeping accurate records of bugs and making sure every issue is tracked until it's solved. The system allows development teams to sort bugs by

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 1, April 2026 Impact Factor: 8.2

severity and priority, helping them focus on the most critical issues first. It also supports project managers in monitoring project progress and coordinating better among different team members.

This project's Bug Tracking System is built using Java and MySQL for database support. It features a user-friendly interface that lets users easily report bugs, assign tasks, update bug status, and manage user accounts. The database holds all information related to bug reports, user details, and resolution status. This organization ensures that all bug-related information is secure, structured, and easily accessible.

The system usually includes different user roles such as Administrator, Developer, and Tester. The administrator manages the system, creates user accounts, and assigns bugs to developers. Testers find and report bugs in the software, while developers review the reported bugs and work on fixing them. This role-based structure promotes accountability and clear task distribution within the development team.

In today's software industry, bug tracking tools like JIRA, Bugzilla, and Redmine are popular for managing software defects. However, these tools can be complex or require advanced setup for small teams or academic projects. Therefore, the Bug Tracking System from this project offers a simpler yet effective way to manage bugs in smaller development settings, such as student projects or small software teams.

By using this system, software teams can significantly enhance defect management, improve communication among members, and ensure the delivery of high-quality software products. The Bug Tracking System shows the practical use of software engineering concepts and emphasizes the importance of organized bug management for successful software development..

II. LITERATURE SURVEY

Software testing is essential for ensuring the quality and reliability of software systems. Many researchers and organizations have created bug tracking tools to effectively manage software defects.

Bug tracking systems help developers spot problems, monitor their status, and ensure all defects are fixed before software deployment. Popular tools like JIRA, Bugzilla, and MantisBT are widely used in the software industry to manage defects and boost development efficiency.

Studies in software engineering show that good bug tracking systems improve communication between developers and testers, cut debugging time, and ensure issues are managed systematically.

However, many commercial bug tracking tools are complicated and need a lot of setup. Small development teams or educational institutions might need a simpler and more userfriendly system.

The Bug Tracking System project aims to provide a straightforward platform for managing bugs in small software development environments. This system allows users to report bugs, assign them to developers, update their status, and monitor progress.

It uses Java for application logic and MySQL for database management, allowing efficient storage and retrieval of bug information.

With this system, software teams can enhance defect management and maintain better control over the software testing process..

III. SCOPE OF THE PROJECT

The Bug Tracking System project aims to create a software application that helps development teams record, monitor, and resolve software defects throughout the development lifecycle. The system provides a centralized platform where testers, developers, and administrators can work together to manage bugs in an organized manner.

• Functional Scope:

The functional scope of the Bug Tracking System outlines the main operations and services the system offers to users. It includes several modules that allow different users to engage with the system based on their roles.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 1, April 2026 Impact Factor: 8.2

1. Bug Reporting Module – A key feature of the system is its ability to let testers or users report bugs found in the software. When a bug is detected, the tester can enter essential details like the bug title, description, priority level, and date of reporting. This data is stored in the system's database, making it accessible to developers and administrators.
2. Bug Assignment Module - Administrators or project managers can assign reported bugs to specific developers. Each bug can be assigned according to its category, priority, or the developer's expertise. This ensures that the appropriate team member handles each bug and helps prevent any duplicate work.
3. Bug Status Tracking - The Bug Tracking System allows users to track the status of each bug during the debugging process. Bugs can be marked as Open, In Progress, Fixed, or Closed. This feature helps the development team understand the progress of bug resolutions and keep tabs on any pending issues.
4. User Management Module - The system has a user management feature that lets administrators create and manage user accounts. Users can be assigned different roles such as Administrator, Developer, or Tester, each with specific permissions and responsibilities within the system.
5. Bug History and Record Maintenance - The system keeps a complete history of all bugs reported in the project. This allows the team to review past bugs, spot common errors, and enhance future development processes..
6. Search and View Bug Reports - Users can search for bugs using various filters like bug ID, status, or assigned developer. This feature simplifies locating specific bug reports quickly and efficiently.

• Non-Functional Scope:
The non-functional scope of the Bug Tracking System emphasizes the quality and performance requirements of the system to ensure it operates effectively and dependably.
1. Usability - The system features a straightforward and intuitive graphical user interface, making it easy for users to navigate the application without needing advanced technical skills.
2. Performance - The Bug Tracking System should respond quickly when users submit bug reports, update bug statuses, or access bug information from the database. Effective database management helps ensure smooth performance.
3. Security - Security is a crucial requirement for the system. User authentication and login mechanisms make sure that only authorized users can access the system and change bug information .
4. Reliability - The system must reliably store and manage bug information without losing any data. The database ensures that all bug reports and user details are kept safe.
5. Scalability - The system should handle a growing number of bug reports and users as the project expands. This guarantees that the system remains useful for larger projects in the future.
6. Maintainability - The software should be simple to maintain and update. Future improvements, such as web-based interfaces, automated notifications, or integration with other project management tools, can be added easily.

IV. METHODOLOGY / APPROACH

The development of the Bug Tracking System follows a systematic software development process to ensure that the system is reliable, efficient, and easy to use. The methodology used in this project is based on the Software Development Life Cycle (SDLC) model, where the system is developed in a sequence of well-defined stages. Each stage focuses on a specific activity such as requirement analysis, system design, development, testing, and implementation.

Step 1: Requirement Analysis and Problem Identification
The first step in creating the Bug Tracking System is to identify the issues with the conventional bug management process. In most software development contexts, the bug management process is done in a conventional manner using emails, spreadsheets, or discussions. The conventional process is inefficient and results in confusion, duplication of work, and loss of critical information.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 1, April 2026
Impact Factor: 8.2

Step 2: System Design and Architecture

After completing the analysis of system requirements, the next phase is designing the architecture of the Bug Tracking System. The system is designed in a modular fashion where different modules of the application are designed to perform distinct tasks.

The architecture of the system is divided into the following major components:

User Interface (UI)

The user interface is designed to create an interactive platform for users to carry out tasks such as bug reporting, status update of bugs, and viewing bug information. The interface is developed using Java Swing, which provides graphical user interface elements such as buttons, forms, tables, and menus to interact with the system.

Application Layer

The application layer holds the central logic of the system. The application layer is developed using Java programming language, which handles user requests, bug data, and system operations.

Database Layer

The database layer is designed to handle storage and management of system data. The project utilizes the MySQL database to store data related to user accounts, bug descriptions, bug status, and assigned developers. The database is designed to ensure that data is stored in a secure manner and can be accessed at any time when needed.

Step 3: System Development and Module Implementation

During the development stage, the Bug Tracking System is developed by breaking down the project into several modules.

Each module has a different functionality in the system.

Login and Authentication Module

This module enables users to log in to the system using their username and password. This module also ensures that only authorized personnel have access to the system and are able to carry out operations according to their roles.

Bug Reporting Module

This module enables users to report bugs whenever they notice errors in the software. The user is required to provide information such as the title of the bug, description, level of severity, among other details.

Bug Assignment Module

This module enables administrators to assign bugs to developers once they have been reported. This ensures that tasks are allocated to developers and that each bug is assigned to the responsible developer.

Bug Tracking Module

This module enables users to track the status of the bug resolution process. The developer is able to change the status of the bug from Open, In Progress, Resolved, or Closed once they have fixed the bugs.

Step 4: System Testing and Quality Assurance

Testing is a critical component of software development to ensure that the system functions correctly and efficiently. Once the software is developed, the Bug Tracking System goes through various testing phases to remove errors and bugs.

Unit Testing

The system is tested module by module to ensure that it functions as expected.

Integration Testing

Integration testing is performed to ensure that all modules function well together. For instance, the login module should be able to connect to the database properly, and the bug reporting module should be able to store bug details successfully.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 1, April 2026
Impact Factor: 8.2

User Interface Testing

The graphical user interface of the system is tested to ensure that the forms, buttons, and tables function well and provide a smooth user experience.

System Testing

System testing is performed on the entire application as a whole to ensure that all functionalities of the system function correctly.

Step 5: Implementation & Deployment

Once the testing is successful, the Bug Tracking System is implemented and deployed for use. The system is installed on the user's computer or server, and the database is set up to store information about the bugs.

The users, including testers, developers, and administrators, can then access the system to log bugs, assign tasks, and monitor the progress of the resolution of the bugs. The system offers a platform for efficient management of software defects.

Maintenance and updates can also be done to enhance the performance of the system and introduce new features in the future.

V. ADVANTAGES

1. Centralized Bug Management: One of the major benefits of the Bug Tracking System is that it gives a central place where all the software bugs can be handled. All the bug reports are maintained in a single database from where they can be easily accessed by the developers, testers, and administrators. There is no need for communication to be done in a scattered manner via emails, spreadsheets, or conversations.
2. Efficient Bug Tracking and Monitoring: The system enables bug status tracking by development teams throughout the software development life cycle. Bug status can be marked as Open, In Progress, Resolved, or Closed. This enables the development team to track the status of bug fixing and ensures that no bug report is overlooked or forgotten.
3. Improved Communication Among Team Members: The Bug Tracking System enhances collaboration between the testers, developers, and project managers. The testers are able to log the bugs into the system, the developers are able to view and close the bugs assigned to them, and the administrators are able to monitor the whole process.
4. Better Task Management for Developers: The system enables administrators or project managers to allocate bugs to certain developers according to their expertise or workload. This will help in allocating tasks among team members in an efficient manner and will also ensure that a certain bug is allocated to a certain developer. Thus, the process of debugging will become more organized.
5. Improved Software Quality: By keeping track of all bugs and making sure that they are fixed before the release of the software, the Bug Tracking System assists in increasing the quality of software. It also helps in reducing the chances of bugs in the software and provides a better user experience.

VI. APPLICATIONS

1. Software Development Companies: The primary use of Bug Tracking System is in software development organizations. In software development organizations, many software developers work on different modules of a software. Many bugs and defects may come up during software development and testing. The Bug Tracking System is used by software development organizations to track these
bugs. This way, software developers can easily identify and resolve bugs in a software product.
2. Software Testing and Quality Assurance: Bug Tracking Systems have a major role to play in the testing phase of the software development life cycle. Testers can report the bugs identified during the testing phases, such as unit testing, integration testing, and system testing, using the Bug Tracking System. The system will ensure that the bugs identified during the testing phase are documented with appropriate details, such as the description of the bug, the level of priority for the bug, etc.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 1, April 2026 Impact Factor: 8.2

3. Project Management: Project managers can utilize the Bug Tracking System in tracking the progress of software development projects. By examining reports of bugs and their status, project managers can analyze the number of bugs that are pending and the number of bugs that have been fixed. This helps project managers in assessing the performance of team members and ensures that the project is being developed according to schedule.
4. Maintenance of Existing Software: Bug Tracking Systems can also be used during the software maintenance phase. Even though the software application has been released, users may encounter certain bugs or unexpected errors in the application. The users can report such errors to the developers, and they can be fixed accordingly, ensuring smooth functioning of the application with the necessary updates.
5. Educational Institutions and Student Projects: The Bug Tracking System can also be used for teaching software development concepts and software testing concepts in educational institutions. The students who are involved in software projects can utilize this system for tracking bugs and understanding the importance of defect management in real-time software development environments.
6. IT Service and Support Organizations: IT service companies and technical support groups can use the Bug Tracking System to track technical issues reported by users or clients. This will enable the support groups to properly document the issues, provide tasks for the engineers, and ensure timely resolution of the issues presented by the clients.

VII. CONCLUSION

The Bug Tracking System designed in this project offers a platform where testers can report bugs, developers can work on bug resolution, and administrators can manage the overall process of bug tracking. The system facilitates the process so that all bugs are registered with appropriate details such as description, priority, status, and developer assigned. This way, it becomes easy for the development team to keep track of how bugs are being addressed and ensure that no bug is left behind.

VIII. ACKNOWLEDGMENT

We express our sincere gratitude to the Vishweshwarayya Institute of Engineering and Technology, Almala for giving us the opportunity to work on the Major Project during my final year of Diploma in Computer Engineering is an important aspect in the field of engineering.
We would like to thank Prof. Kazi A. S. M, Head of Department, Computer Engineering at Vishweshwarayya Institute of Engineering and Technology, Almala for their kind support. We also owe our sincerest gratitude towards Ms. Patil P.D. for her valuable advice and healthy criticism throughout my project which helped me immensely to complete my work successfully.
I would also like to thank everyone who has knowingly and unknowingly helped me throughout my work. Last but not least, a word of thanks for the authors of all those books and papers which I have consulted during my project work as well as for preparing the report.

REFERENCES

[1] Sommerville, Ian. Software Engineering, 10th Edition, Pearson Education, 2015..
[2] Pressman, Roger S., and Bruce R. Maxim. Software Engineering: A Practitioner's Approach, 8th Edition, McGraw-Hill Education, 2019.
[3] Rajaraman, V. Fundamentals of Computers, Prentice-Hall of India, 2014.
[4] Laudon, Kenneth C., and Laudon, Jane P. Management Information Systems: Managing the Digital Firm, Pearson Education, 2017Education, 2017.
[5] Myers, Glenford J., Corey Sandler, and Tom Badgett. The Art of Software Testing, 3rd Edition, John Wiley & Sons, 2011.
[6] MySQL Developer Documentation – https://dev.mysql.com/doc
[7] GeeksforGeeks – Concepts on Software Development and Bug Tracking Systems – https://www.geeksforgeeks.org

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 1, April 2026
Impact Factor: 8.2

[8] Atlassian. Issue and Bug Tracking Concepts – https://www.atlassian.com
[9] Mozilla Developer Network – Software Development and Web Technologies – https://developer.mozilla.org
[8] ResearchGate – Research Articles on Software Testing and Bug Tracking Systems – https://www.researchgate.net
[9] Redmine Issue Tracking System – https://www.redmine.org
[10] Software Testing Help – Concepts of Bug Tracking and Defect Management – https://www.softwaretestinghelp.com
[11] Stack Overflow Developer Community – https://stackoverflow.com