



User-Centric Bug Tracking Issue Logging and Resolution Workflow Management System

Mrs. K. Deepthi¹, Sk. Chinna Nagul Meera², P. Naga Babu³, T. Gangadhara Rao⁴,
P. Venkata Hemanth Kumar⁵

Assistant Professor, Department of Information Technology,

KKR & KSR Institute of Technology and Sciences, Guntur, Andhra Pradesh, India¹

B. Tech Student, Department of Information Technology,

KKR & KSR Institute of Technology and Sciences, Guntur, Andhra Pradesh, India^{2,3,4,5}

Abstract: This paper is based on the web application that can be used to identify, manage, and resolve the important bugs in the software development. This system allows users to register the problems occurring in the software as bugs, also it tracks the status of the operation, and also it ensures that they are resolved in a particular time. As frontend this application uses the ReactJs Single Page Application, and also mainly it provides simple, fast and better experience for the user. On backend RESTful services are used by using Java Spring Boot, in this application there is a clear separation between the front end and the back end.

Index Terms: ReactJS SPA, Java Spring Boot, MySQL, REST- full Services, CRUD operations.

I. INTRODUCTION

Nowadays the software is constantly changing and adding new features. It is natural for small errors to occur. A good system is needed to properly observe, record and quickly resolve these bugs with new features. While doing this we are getting small bugs. We need a suitable system to observe and register the bugs.

The User-Centric Bug Tracking System is a type of system in which the issues faced by the users are easily registered and the information in what stage the issue is at and who is solving it clearly. The entire software team like both developers and testers can talk together and solve the problems quickly. In this way the coordination can be improved, human work can be reduced and time can be saved. While the software is changing it is difficult to identify the bugs. By having simple and well-organized system bugs can be easily solved by this the users can get the good and quality software.

In many organizations the bug tracking is still done using emails and spreadsheets. It causes duplicate issues and delays in fixing problems. There is no clear identification of bug status which affects the overall development process. To overcome these challenges we use a user-centric bug tracking system.

II. LITERATURE REVIEW

The Online Bug Tracking System provides web based solutions to report, track and fix software bugs. This system supports teamwork, and it allows developers, testers and managers to communicate easily. By this system the users can enroll the bugs, they can be given priority and we can easily know the current state in real time. It allows developers, testers, and project managers to communicate and work together on the same platform. This helps the project team understand the problems clearly and solve them faster. It defines how the online bug tracking system reduces manual work [2].

The bug tracking and modification system helps to improve the communication between the bug report and developers. This system allows bugs to report in a clear and organized way. The final status shows better information and improves software maintenance [2]. The bug tracking and logging toolkits are used in software engineering. It compares different tools based on the features [3].

III. METHODOLOGY

A. System Overview

This system provides the required learning materials to the students and as well as it gives guidance to the students, and



provides the answers for the questions.

B. System Architecture

The system architecture is classified into four modules: user interface module, resource management module, assistant module, ai based recommendation module. The students while learning the resources can easily communicate with the system through a web based interface. These resources are structured based on the subject, and academic regulations. It analyzes the student data and provides learning materials.

C. Data Collection and Processing

In this data collection and processing the data can be collected from the student like logging activity and resource request pattern. The collected information from the student is processed by the modules which helps to understand the student learning status and improve academic status. And also it maintains privacy and security through the process.

D. Implementation Steps

It contains multiple stages which are requirement analysis implementation, system design implementation, database implementation, testing implementation, deployment implementation.



Fig. 1: System architecture diagram showing module interactions

E. Testing and Evaluation

The testing process helps to check if everything works correctly in which it focused on features and functions works correctly and effectively. Different modules were tested to verify the accurate operations. The evolution of User-Centric Tracking Issue Logging and Resolution Workflow Management System follows the step by step process then it becomes user-friendly and also it uses modern technologies and it includes the features and it helps the team to deliver the better quality software.

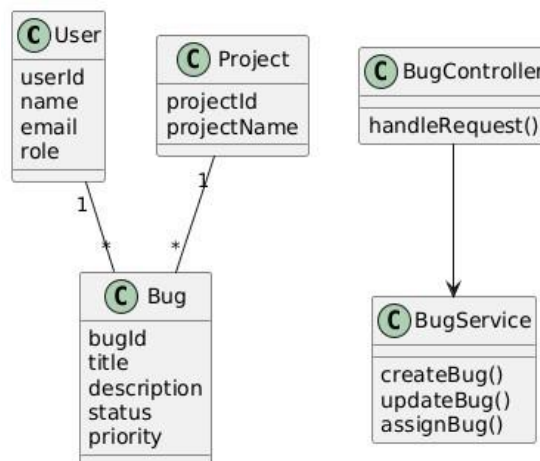


Fig. 2. Component breakdown of the system modules

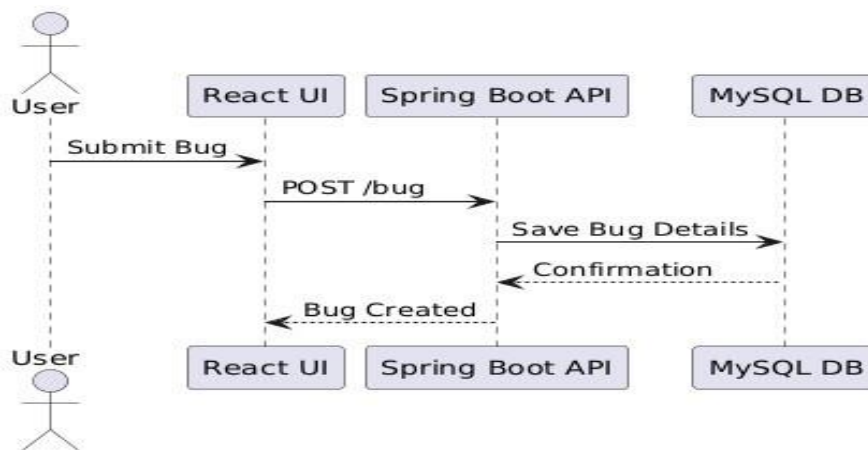


Fig. 3: Implementation workflow showing development stages

IV. RESULT

The User-Centric Bug Tracking Issue Logging and Resolution Workflow Management System can be successfully designed by using the Java Full Stack Technology. In this system different user roles are there, so that everyone can easily do their work. They organize the admin user and assign it to the developers. The developers work on the received bugs and update the status of the bugs. The testers can report the new issues, and check if the fixed bugs are working properly or not. In this way based on the working effectively based on the roles it increases the responsibility and the team can work together.

The user interface made with ReactJs is very simple, and smooth. And it helps to avoid the page reloading multiple times. The backend made with Spring Boot responds in a short time and it makes better connectivity with the frontend. By this the software quality is improved and also the team productivity is improved.

Based on the web applications the users can easily register the bugs, we can see all the information, and we can easily identify the stage of the bug. Finally, the project provides a useful tool using Java Full Stack tools, the made of project software system is proved. In feature automatic manages, delivery reports, adding features such as connections with other software tools, and it can make it more useful for large development teams.

V. DISCUSSION

This project was created to resolve the bug issues faced by software teams. Usually, work can be delayed due to improper documentation of bugs, and having the improper communication of the team. In this team all bugs are registered at one place, and it will be visible to everyone. From bug reporting to bug fixing, every stage is clearly visible. With that everyone should understand and get the clarity on what is to be done. In this everyone has separate roles. The admin checks who should be assigned which work. The developers fix the bugs. The testers report the bugs and to check if the bugs are fixed or not. By keeping the work separate there is no need to do the same work again and again, and the understanding of the team is increased. From a technical point of view this system is easy to understand and easy to use. The modification of the bug can be done easily and adding bugs is simple. In this all the data can be stored securely. While during the project development if some minor issues occur they can be easily tested and fixed.

Finally, this project helps to handle the bugs clearly and easier. And also in future more features are added, but it works well.

VI. CONCLUSION

This project was completed correctly and it is designed to help the teams to manage bugs easily and clearly and as expected. In this system the users can directly report the problems, they can easily check their status, and if focused on completing the fixing process step by step. To develop this project we use Java Full Stack technologies. By working on this project, we are observed and gained realtime experience. It helps us to understand how the user interface, and database work together in a single system. The User-Centric Bug Tracking Issue Logging and Resolution Workflow Management System is very useful for team members and helps in building high quality software.



REFERENCES

- [1]. U. B. Torun, M. T. Demircan, and M. T. Gon, "Bug tracking in the generative AI era," *International Journal of Engineering Research and Sciences & Technology*, vol. 21, no. 4, 2025.
- [2]. K. Vasumathi, S. Selvakani, and P. Sridhar, "Comprehensive bug tracking and modification system," *Asian Journal of Electrical Sciences*, vol. 11, no. 2, 2022.
- [3]. M. Rajeswari and K. Parimala, "An empirical study of test management in bug tracking systems," *International Journal of Trend in Scientific Research and Development*, vol. 1, no. 4, 2017.
- [4]. A. V. Jain, Y. Khurana, and M. Kharbanda, "Bug tracking system based on platform and system," *International Journal of Development Research*, 2017.
- [5]. Sharma and R. Gupta, "Web-based issue tracking systems for software maintenance," *International Journal of Computer Applications*, vol. 176, no. 9, pp. 20–26, 2020.
- [6]. M. Verma and S. Agarwal, "Design of single-page applications using ReactJS," *International Journal of Engineering Research and Technology (IJERT)*, vol. 9, no. 6, 2020.
- [7]. P. Kumar and S. Das, "RESTful services using Spring Boot for enterprise applications," *International Journal of Software Engineering*, vol. 11, no. 3, pp. 55–62, 2021.
- [8]. S. Reddy and K. Kumar, "Secure database design for web-based applications," *International Journal of Computer Science and Information Security*, vol. 18, no. 5, pp. 98–104, 2020.
- [9]. R. Singh and A. Malhotra, "Workflow automation in software issue tracking systems," *Journal of Information Systems*, vol. 14, no. 2, pp. 89–97, 2021.
- [10]. A. Jain and P. Malhotra, "Role-based access control in web applications," *International Journal of Network Security*, vol. 23, no. 4, pp. 610–617, 2021.
- [11]. T. Nguyen and L. Tran, "Collaborative platforms for software development teams," *Journal of Software Engineering*, vol. 19, no. 1, pp. 33–41, 2022.
- [12]. R. Das and P. Chatterjee, "User-centric software maintenance systems," *Education and Information Technologies*, vol. 27, no. 5, pp. 6891–6904, 2022.
- [13]. J. Lee and K. Park, "Usability evaluation of defect tracking tools," *Journal of Human-Centered Computing*, vol. 6, no. 2, pp. 145–153, 2022.
- [14]. OWASP Foundation, "OWASP top 10 web application security risks," 2023.
- [15]. A. Gupta, R. Singh, and V. Kumar, "Issue tracking systems for agile software development," *IEEE Access*, vol. 10, pp. 112345–112356, 2022.