iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 5 **Issue:** III **Month of publication:** March 2017

DOI: http://doi.org/10.22214/ijraset.2017.3131

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

Volume 5 Issue III, March 2017
ISSN: 2321-9653
International Journal for Research in Applied Science & Engineering Technology (IJRASET)

Defect Tracking System

Syed Parvez[1], Syed Yousef Danish Mehdi[2], Md Sufiyan Ali[3], Prof Shaik Maheboob[4]
[1,2,3,4]Computer Science Department, Lords Institute of Engineering and Technology

Abstract: ***Bug tracking software is used to keep records of the bug encountered by a programmer so that if that error should error in the future it should be solved easily. It's important that everyone on the team is able to find and record bugs Bug Tracking System is the system which enable to detect the bugs. It not merely detects the bugs but provides the complete needs to know information regarding the identified bug. With a bug tracking system, a bug which can be anything from a simple question asked by the user to a detailed technical report of an error or bug can be tracked by priority status. Whenever the tester encounter number of bugs he adds the bug id and information in the database. This paper shows a proposed defect tracking model for the purpose of classifying the inserted defect reports in a step by step method for more enhancement of software quality. the information that is stored in database is accessible to both project manager and developer when user wants a product to be developed. The project manager is responsible for adding users to Bug Tracking System and assigning projects to the user information regarding bugs detected. While obtaining documented work flow and positive feedback for good performance. It provides a chat system where team members can interact with each other to solve Bug problems.***
Keywords: Error, Bugs, Software Development Life Cycle (SDLC), UAT (User Accepting Testing), Defect tracking system.

I. INTRODUCTION

A bug tracking system or defect tracking system is a software application that keeps track of reported software Each bug in the system may have an urgency value assigned to it, based on the overall importance and the future occurrence of that bug. Low or zero urgency bug are minor and should be resolved as time permits. Other details of bugs include the customer experiencing the rise of bug, date of submission, detailed descriptions of the problem being experienced, attempted solutions and other relevant information. As work is done on that bug, the system is updated with new data by the technician. Any attempt at fixing the problem should be noted in the bug system, as each bug maintains a history of each change.

II. PROCEDURE

In the previous Defect tracking system, user does not specify all the information of the bug needed by developers. a bug-tracking system may be used to generate reports on the productivity of programmers at fixing bugs. However, this may sometimes yield inaccurate results because different bugs may have different levels of severity and complexity. The severity of a bug may not be directly related to the complexity of fixing the bug. There may be different opinions among the managers and architects

When a software engineer develops a bug report, most probably, he should have asked the following questions. Some of them are what is the name of the product? What is the type of bug? In which component is the bug? In which modules the bug? In which part of the program the bug is? What type environment in which the bug is arising? In which platform the application is built? In which Operating System the application is running? The report given by developer about bugs might be incomplete initially. When a bug report is submitted by a developer, it is the duty of the software tester to the follow up questions are to be asked about the software bugs which gives a tester complete of the issue he is facing in a particular project. Our bug tracking system provides an "automated questionnaire system" which collect all the information on behalf of the tester. More over answer to a question determines the next possible question. These question and answers helps a tester in narrowing down exact location of the bug in the code.

A. The Following are the Important Components of a Bug Tracking System

First and fore most thing we need is the exact location of the bug in the program. Location gives you the information like the line number of the bug in the program its method, class and so on. This helps a developer to get to the location of erroneous code. This information can be obtained by using software development environments (IDEs).

A major component of a bug tracking system is a database that records facts about known bugs. Facts may include the time a bug was reported, its severity, the erroneous program behaviour, and details on how to reproduce the bug as well as the identity of the person who reported it and any programmers who may be working on fixing it

Bug tracking system will handle structured data including user information, login information, data code information and reports

which are saved in a database. Our project provides these facilities using JSP and HTML languages.HTML provides a user interface for login, sing up etc. Whereas JSP is used to established a connection between the HTML and server. In this we have used MYSQL database to store the information of the users. Our intuitive interface includes a Dashboard view showing you issues 'Assigned to Me', 'Unassigned', 'Reported by Me' and many other standard filters as your go-to page to get started for the day. Defect tracking is the process of finding defects in a product and can come in the form of discrete systems where details of the defect and the status of work associated with resolving it are stored or they may be part of an integrated suite of configuration management tools, where the status of the defect may act as a trigger or key for other events within the system.

III. CONCLUSIONS

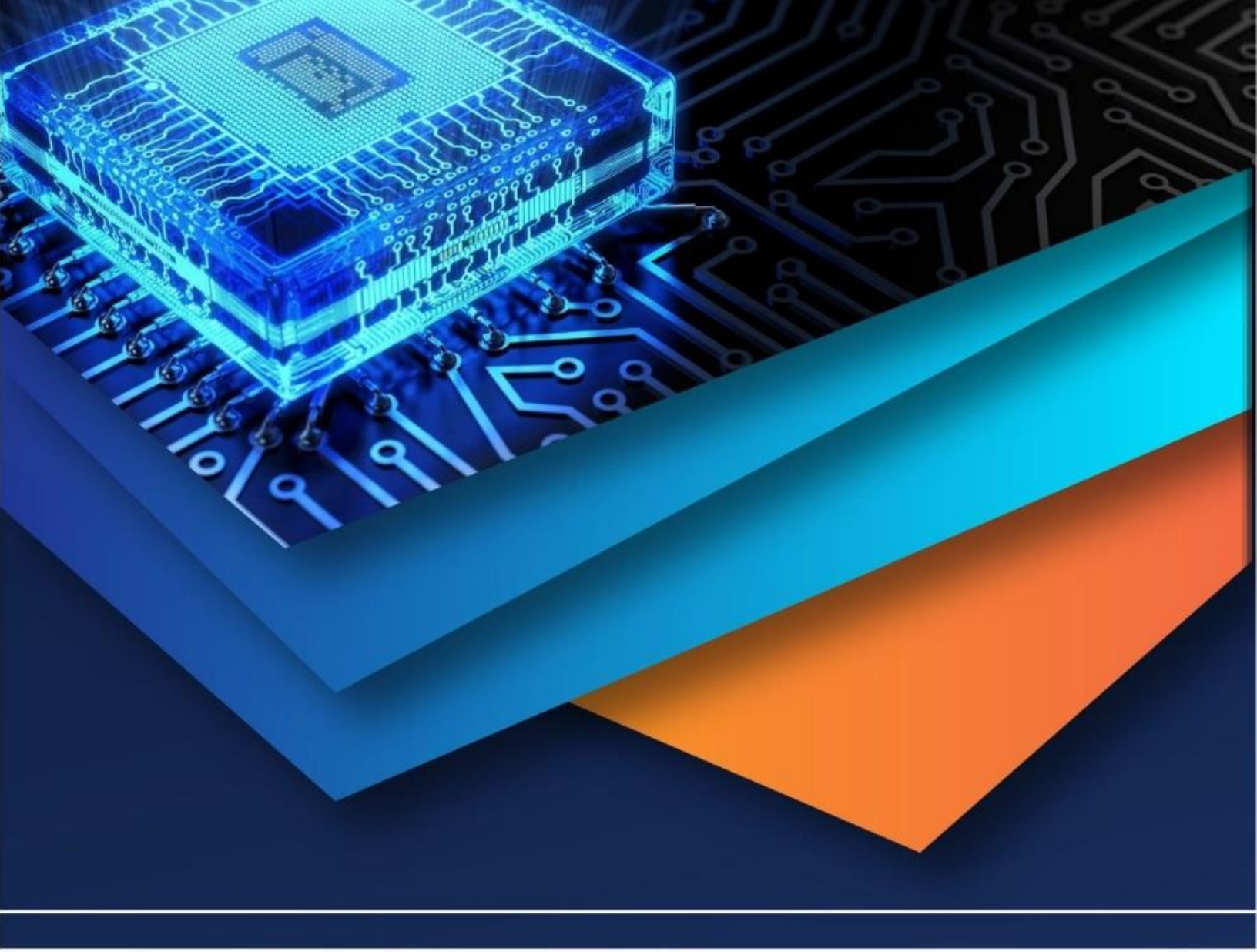
In this paper we have reviewed on the technologies which are being used for finding and improving bug tracking system. It is impossible to not encountering bugs during software development life cycle. Our bug tracking system provides an "automated questionnaire system" which collect all the information on behalf of the tester.

A. Administrator or Project manager can get project status as well as management can get the ongoing status of the project. Present system can further be improved by applying certain standards. Our intuitive interface includes a (Dashboard view) showing you issues and (Timeline feature) allows you to keep across what's happening around you with a live stream on the latest actions by your team mates

B. Customer resource management is integrated with this application which makes this software more efficient in terms of time management. Present system can further be improved by applying certain standards.

REFERENCES

[1] A Practitioners approach “Software Engineering” Roger S. Pressman 5thEdition.
[2] Lessons Learned in Software Testing (Paperback) Publisher: Wiley; 1st edition (December 15, 2001) James Bach
[3] Bret Pettichord,C. Kaner.
[4] Beautiful Testing: Leading Professionals Reveal How They Improve Software (Theory in Practice) (Paperback) Publisher: O'Reilly Media; 1
[5] edition (October 22, 2009) Tim RileyAdam Goucher.
[6] Why Programs Fail, Second Edition: A Guide to Systematic Debugging (Paperback) Publisher: Morgan Kaufmann; 2 edition (June 26, 2009).
[7] Software Engineering,”Rojer S. Pressman” (TATA McGraw Hill Publication).
[8] http://en.wikipedia.org/wiki/Bug_tracking_system
[9] http://www.wisegeek.com/what-is-bug-tracking.htm (2002) The IEEE website. [Online]. Available: http://www.ieee.org/
[10] (Avram, 2007) Gabriela Avram, Anne Sheehan and Daniel K. Sullivan, Defect Tracking
[11] Systems in Global Software Development – a work practice study, Limerick, Ireland, 2007.
[12] (Basili, 1984) Victor R. Basili, and Barry T. Perricone, Software Errors and Complexity: An
[13] Empirical Investigation, Communications of the ACM, P.42-52, 1984.
[14] (Black, 1999) Rex Black, Managing the Testing Process, Wiley, 1999.
[15] (Blaxter, 2006) Loraine Blaxter, Christina Hughes and Malcolm Tight , How to Research THIRD EDITION ,2006.
[16] (Boehm, 2001) Barry Boehm and Vector R. Basili, Software Defect Reduction Top 10 List, p.135–137, 2001.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY