BSTS: Bug and Status Tracking System with Data Visualization

Yashomathi K S[1], Dr. Leena Giri G[2]

[1]Student, [2]Professor, Department of CSE, Dr. Ambedkar Institute of Engineering and Technology, Bengaluru, India.

ABSTRACT

The Bug Tracking System (BTS) is a software application that can track the bug in the application and report them. In automation testing, there are many tools to track bug and fix the bug. but they have drawbacks and may sometimes give inaccurate results depending pn severity and priority of the bugs. This cause is inability to find out the complete bug present in the system. This work presents a tracking system where that tracking of the bug is done manually and reported to the Administrator and Manager. The Administrator maintained records of all the bugs reports and he keep track of status of projects, status of bugs, status ogf tasks. The manager communicates with the developers, the developers clear the system off the bug and reported back to the manager as well as the administrator. A data visualization technique is uses to visualize real-time data in the database. Instead of getting feedback about project by managers. The administer can directly view the current status of this system through charts made by developers. This is done with help of STS.

Keywords: BTS, STS, Bug Report, Data Visualization

I. Introduction

It is the universe of data. BTS and STS are normally connected as the face of the incorporated project management system. BSTS is a web-based application that can be accessed by the Organizations through network. Bug report usually contains Bug-Id, Bug-Name, Bug-severity, Bug-priority, Bug-status. BSTS can be used for logging defects, assign defects to individual and tracking the defects to resolution by exploiting manual testing techniques to fix the bug. The administrator maintains the master details such as Manager details, project details and user details. The Manager and administrator have authority to update the status details of projects. The aim of this work is to present BSTS, a bug tracking system that is useful for tracking bugs in an application developed by in an organization. Scalability issues are considered and the system is built to manage hundreds of tasks in multiple areas. The important goal of this BSTS is track all the defects or bugs in a given software and make it a user friendly and bug free system.

i. Motivation

Testing is one of the Crucial exercises which ought to be performed during the software improvement aim to Discover the errors as per client prerequisite. The BSTS for improving programming unwavering quality is to give better support of the head or valuable for applications created in an association. For Improving Programming Dependability, Bug Following Framework (BTS) gives the office to characterize the undertakings and enable the administrators to follow the Imperfections and time spent by every worker for that specific assignment. This instrument can help managers for (Defects) Bugs estimation per venture. This device likewise causes representatives to report their Imperfections and investigate the nature of their yield. The outcomes show that product Experts utilize a wide arrangement of methodologies and procedures when performing manual testing. Testers is need to and use to techniques even if apply exploratory testing. Know the execution time of test cases is important to perform test schedule, priority and progress activity monitoring.

ii. Contribution

Manual testing can be done on all kinds of applications. A newly designed test case should be executed manually. In a professional workplace, the BTS might be utilized to create the reports on the efficiency of the software engineers at fixing the bugs. Real-time data is frequently used in tracking system. The STS tracking

the status of the bugs, tasks and projects. The DTS and STS made the overall project management much easier & flexible. Complete status of the system can be visualized on dashboard.

II. Proposed System

BTS test the application for bugs and reports it to the project manager. Stores the bug information with a unique id in the database for future reference. This makes the job of handling the bug easy. The system is designed to be user friendly. Software can be used both for bug tracking and for status tracking project management.
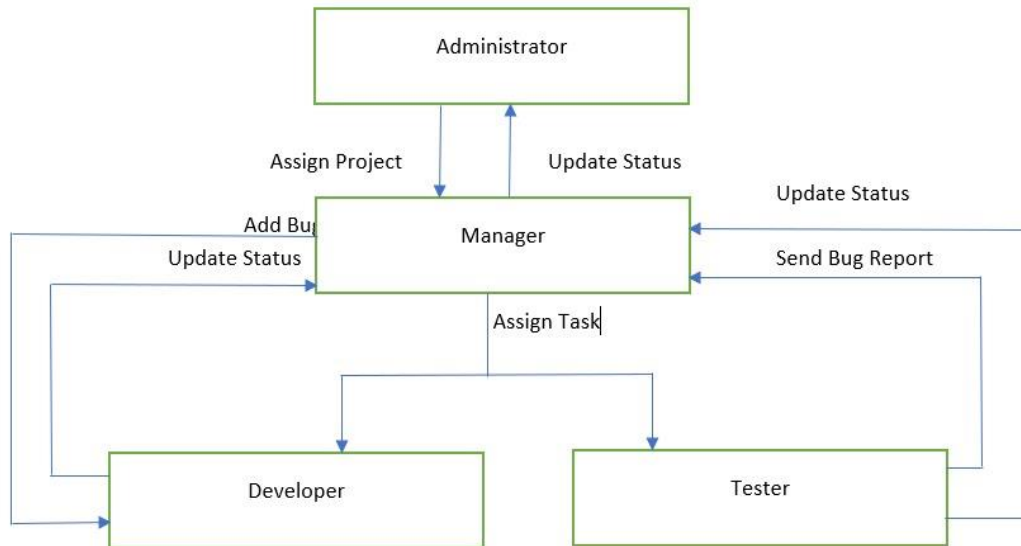

Fig 2.1: Block Diagram of BSTS

Fig 2.1: BSTS consists of administrator, manager, developer, tester.
The administrator assigns the project to the manager and decides on what should be and how it should be done. he can add new projects, add managers and view the status of entire project progress. the administrator can view the status of the system with the charts given by the developers while it is a data visualization technique created by developers. The manager delegates the task to developer and tester, the task he received from the administrator. The manager is given the ability to add developer, tester and bugs. He can also view the project status and task status and bug status. The assigns the Bugs for fixing, to the developer and assigns the tasks to both developer and tester. The developer is the responsibility for fixing the bug in the tasks he received by the manager. The developer also updates the program and completion of the track to the manager. The tester module is responsible for finding the bug in the track he receives from the manager. The tester also updates the progress and completion of a task to the managers.

i. Algorithm and implementation

- **Algorithm (Pseudo Code):**

Step 1 : Start
Step 2 : Initialize input Data
Step 3 : Initialize priority = "p1, p2, p3, p4"
Step 4 : Initialize Severity = "Blocker, Critical, Major, Minor"
Step 5 : Set stored Data = input
Step 6 : If stored Data = input

```
        {
                Success
        }
Step 7 : If stored Data! = input
        {
                Switch (bug)
                {
                        Case priority:
                        If (bug == Blocker)
                        {
                                Then p1
                  // check error in
                Log Files
                        }else
                        if (bug == Critical)
                        {
                                Input = p2
                // Check error in Log Files
                        } else
                        If (bug == Major)
                        {
                                Input == p3
                                // Check error in Log Files
                        } else
                        If (bug == Minor)
                        {
                                Input == p4
                                // Check error in Log Files
                        }
                        Break;

                }
        }
        Step 8: End
```

Severity: How the bug influences the application. How basic bug is and what is the impact of the bug on the whole system functionalities. The severity is a parameter set through the analyser(tester) while he opens a bug and is much of the time in oversee of the analyser(tester).

Priority: is define the priority in which the bug has to be resolved. If there are a couple of bugs, the priority wants to decides which bug has to be fixed and tested right now which one bug can be fixed a bit later. It commonly set by the manager.

Examples
High Priority, Low Severity bug :- If the table-ladled name is incorrectly spelled in the page of the site, at that point the priority is high and severity is low to fix it.
High Severity, Low Priority :- When user clicks on a link, web page not found (user's does not visit that page generally).
Low Priority, Low Severity:- if any spellings issues which is inside a passage or in the report.
High Priority, High Severity:- A mistake which happens on the fundamental usefulness of the application and won't enable the client to utilize the system. (E.g. user is not able to login to the application).

- **Implementation**

BSTS is implemented in this project manually, instead of using any automation tools. The bugs are found out manually more effectively than the automation test. The BTS and STS are designed to be user-friendly. BSTS can be used bug tracking and status tracking project management. Information representation is utilizing numerous orders as an advanced likeness visual correspondence. It includes the approach and investigation of the unmistakable delineation of information. To speak facts without a doubt and efficiently, records visualization makes use of statistical graphics, plots, records pics. The STS can be used Socket Implementation is used to display the status in dashboard with charts. Socket.io is a JavaScript library that are used in data visualization technique. After updating any data in database, we can see the updated data in the screen without refreshing the web page. To visualize project status, bug status, and task status. The project status will show progress in no of projects, completed projects and yet to start a project. task status will show progress in several tasks, completed tasks and yet to start tasks. bug status will show the fixed bugs and yet to fix bugs.

III. Experimental Results

In this section, we evaluate the pseudocode for finding the bugs. By using this pseudocode testers will test the application. if any bugs found, testers want to make a report using below the bug report template (Fig 3.1) and reported to the manager through email and updating status. The manager will
see the bug list (Fig 3.3) and add bug to a developer for fixing it. when once the developer is done with the fixing bugs, they update the bug status. and in this section, we expect the administrator and manager will see the current status of the project tasks and bugs (Fig 3.4 and Fig 3.5). In BSTS, we visualize the Realtime-data(information) of the project, tasks, and bugs by using graphs, charts.

Bug Report

Bug-Name:
Bug-ID:
Priority: [Low/Medium/High]
Severity: [Low/Medium/Hight]
Assigned To: Developer
Created by: Tester

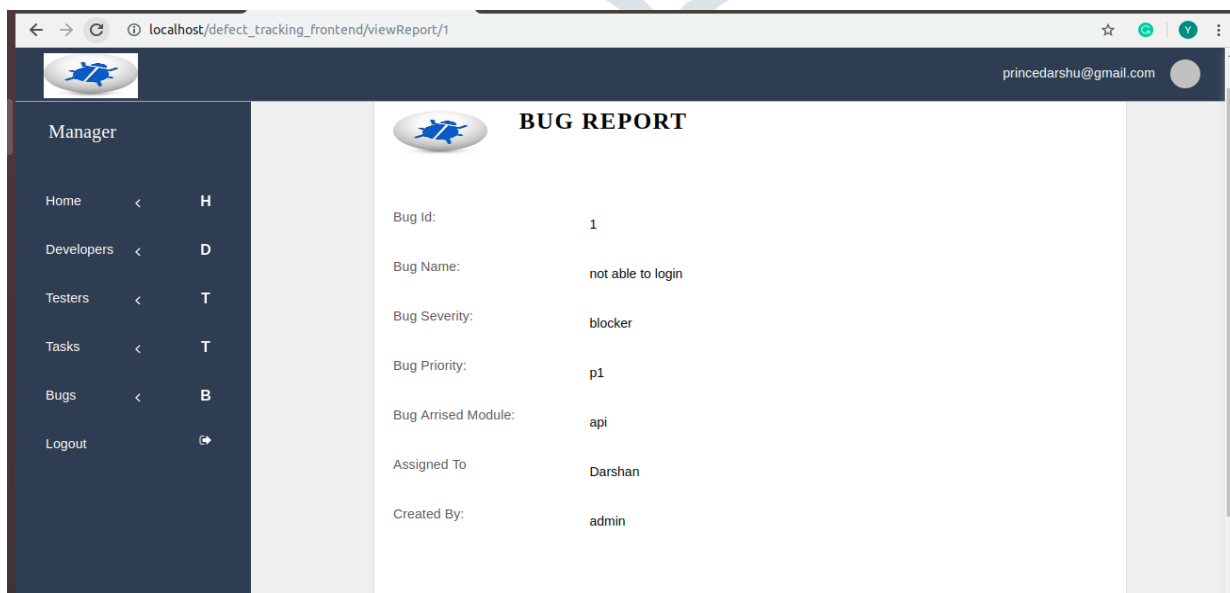**Fig 3.1 : Template for Bug Report**

Snapshots

Fig 3.2: Bug Report

PROJECT BUGS

BUG ID	BUG NAME	BUG SEVERITY	BUG PRIORITY	ASSIIGNED TO	BUG ARRISED MODULE	CREATED BY	STATUS
1	not able to login	major	p3	Darshan	front end screen	tester	
3	not able to access	minar	p4	Darshan	api	tester	
4	functionality not working	critical	p2	Darshan	api	tester	
5	button not defing	blocker	p1	Darshan	fornt end screen	tester	
6	image not found	minar	p4	Darshan	api	tester	
7	integration error	blocker	p1	Darshan	api	tester	

Showing **1 to 6** of 6 entries
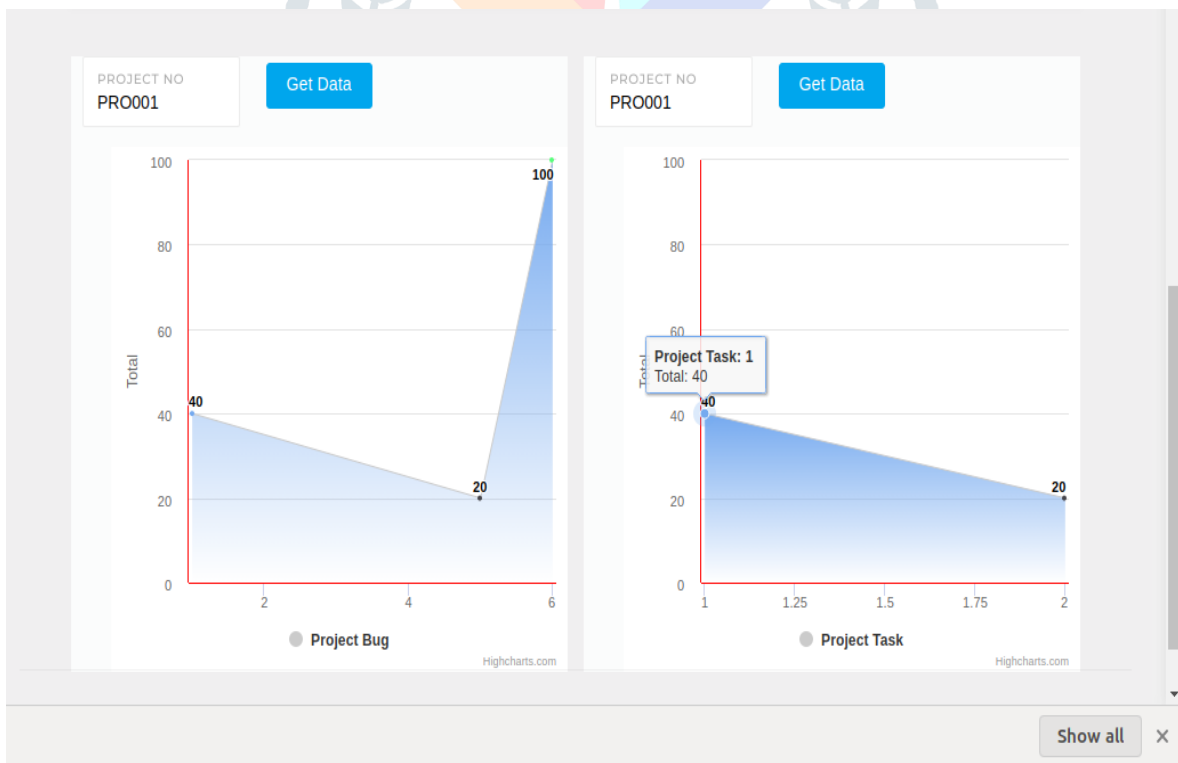
1

Fig 3.3: Bug List
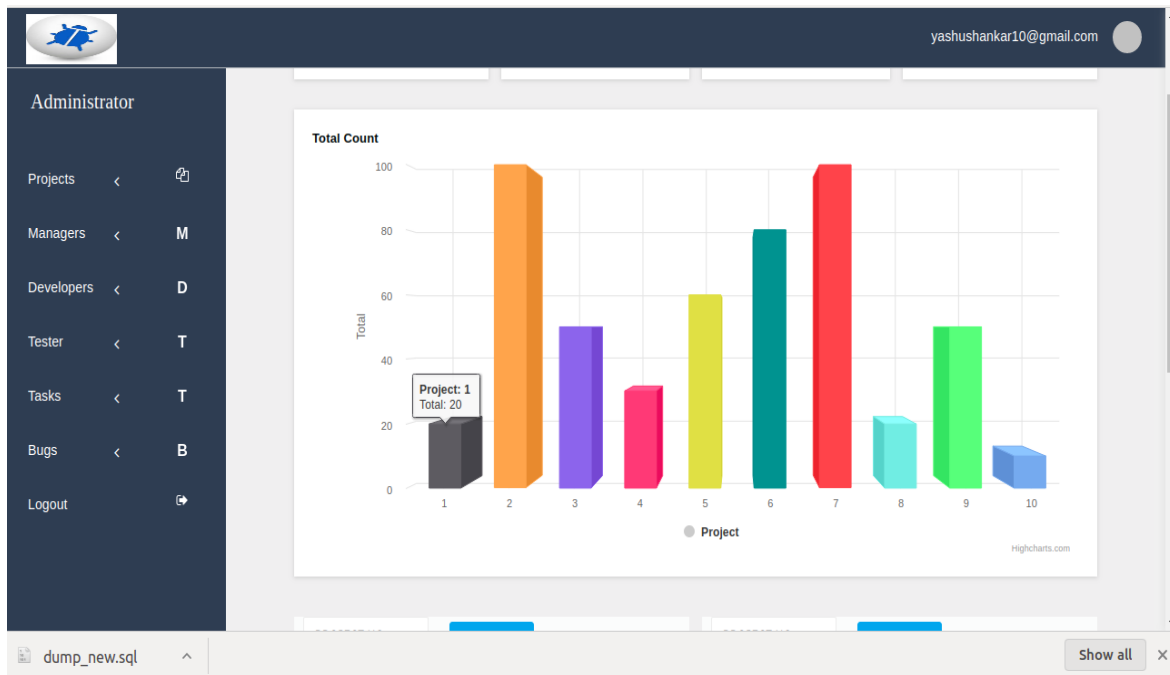

Fig 3.4: Bug and Task Status

Fig 3.5: Project Status

IV. CONCLUSION

BTS and STS are the showings to fix the bugs in a web-based application by using manual testing. And track the status of each task and progress of the project in each stage of processing. An application that will allow one to control her/his computer by login into a system and update the project detail and task detail in the system so it ts easy to get information about the project status and description of the project and it will save their time and no need of a paperwork. We can visualize all the status on dashboard.

REFERENCE

[1] Dr. Sangeetha Yalamanchili K.Sitha kumari(Ph.D) "Comparison of manual and automatic testing using Genetic algorithm for information handling system", *International conference on Signal Processing, Communication, Power and Embedded System (SCOPES),* PP. 1795-1799, 2016.

[2] Juha Itkonen, Mika V. Mäntylä and Casper Lassenius,"How Do Testers Do It? An Exploratory Study on Manual Testing Practices", *Third International Symposium on Empirical Software Engineering and Measurement,* PP. 495-497, 2009.

[3] Sahar Tahvil, Mehrdad Saadatmand, Markus Bohlin, Wasif Afzal, Sharvathul Hasan Ameerjan."Towards Execution Time Prediction for Manual Test Cases from Test Specification", *43rd Euromicro Conference on Software Engineering and Advanced Applications,* PP. 421-425, 2017.

[4] Elmar Juergens, Benjamin Hummel Florian Deissenboeck, Martin Feilkas Christian Schlögel, Andreas Wübbeke, "Regression Test Selection of Manual System Tests in Practice", *15th European Conference on Software Maintenance and Reengineering,* PP. 309-312, 2011.

[5] A.S.Syed Fiaz, N.Devi, S.Aarthi, "Bug Tracking and Reporting System",*International Journal of Soft Computing and Engineering (IJSCE),* vol.3, pp.42-45, March 2013.