



DEFECT TRACKING SYSTEM

KEYWORDS

defect tracking, bugs finding system.

AWARE SACHIN B

MAULANA MUKHTAR
AHMED NADVI TECHNICAL
CAMPUS, "MANSOORA"
MALEGAON

PAWAR KEDA S

BOMBALE RAVINDRA R

ABSTRACT

Defect tracking system is mainly used to track or find the bugs in the organization. It will helpful to track the defect and give the report of defect in online system. This system will helpful because it will add the result in any position as well as user can add the results at any position in the system.

INTRODUCTION

This project is aimed at developing an online defect tracking system useful for applications developed in an organization. The Defect Tracking System (DTS) is a web based application that can be accessed throughout the organization. This system can be used for logging defects against an application/module, assigning defects to individuals and tracking the defects to resolution. There are features like email notifications, user maintenance, user access control, report generators etc in this system.

The software is fully integrated with CRM (Customer Relationship Management) as well as CMS (Content Management System) solution and developed in a manner that is easily manageable, time saving and relieving one from manual works.

PROPOSED SYSTEM:

The development of the new system contains the following activities, which try to automate the entire process keeping in view of the database integration approach.

User friendliness is provided in the application with various controls. The system makes the overall project much easier and flexible to the end user. It can also be deployed over the Internet. Various classes have been used to provide file upload and mail features. There is no risk of data mismanagement at any level while the project development is under process. It provides high level of security using different protocols like https etc. The system Interface provides good environment to solve any kind of defect.

The system capture the overall details of any defect from this user interface and send it to the particular administrator. Every defect details must be stored in the centralized data base server which can be organized by the company.

MODULES:

- 1) Administrator
- 2) Operational Administrator
- 3) Operator
- 4) Defect Tracking
- 5) Search
- 6) Reports
- 7) Registration and Authentication

In engineering, defect tracking is the process of finding

defects in a product (by inspection, testing, or recording feedback from customers), and making new versions of the product that fix the defects. Defect tracking is important in software engineering as complex software systems typically have tens or hundreds or thousands of defects: managing, evaluating and prioritizing these defects is a difficult task. When the numbers of defects gets quite large, and the defects need to be tracked over extended periods of time, use of a defect tracking system can make the management task much easier.

DEFECT PREVENTION:

The objective of defect prevention is to identify the defects and take corrective action to ensure they are not repeated over subsequent iterative cycles. Defect prevention can be implemented by preparing an action plan to minimize or eliminate defects, generating defect metrics, defining corrective action and producing an analysis of the root causes of the defects.

Defect prevention can be accomplished by accomplishing the following steps:

1. Calculate defect data with periodic reviews using test logs from the execution phase: this data should be used to segregate and classify defects by root causes. This produces defect metrics highlighting the most prolific problem areas.
2. Identify improvement strategies.
3. Escalate issues to senior management or customer where necessary.
4. Draw up an action plan to address outstanding defects and improve development process. This should be reviewed regularly for effectiveness and modified should it prove to be ineffective.
5. Undertake periodic peer reviews to verify that the action plans are being adhered to.
6. Produce regular reports on defects by age. If the defect age for a particular defect is high and the severity is sufficient to cause concern, focussed action needs to be taken to resolve it.
7. Classify defects into categories such as: critical defects, functional defects, and cosmetic defects.[1]

A bug tracking system is a software application that is designed to help quality assurance and programmers keep track of reported software bugs in their work. It

may be regarded as a type of issue tracking system.

Many bug tracking systems, such as those used by most open source software projects, allow users to enter bug reports directly. Other systems are used only internally in a company or organization doing software development. Typically bug tracking systems are integrated with other software project management applications.

Having a bug tracking system is extremely valuable in software development, and they are used extensively by companies developing software products. Consistent use of a bug or issue tracking system is considered one of the "hallmarks of a good software team".

COMPONENTS

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 behavior, 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.^[2]

Typical bug tracking systems support the concept of the life cycle for a bug which is tracked through status assigned to the bug. A bug tracking system should allow administrators to configure permissions based on status, move the bug to another status, or delete the bug. The system should also allow administrators to configure the bug statuses and to what status a bug in a particular status can be moved. Some systems will e-mail interested parties, such as the submitter and assigned programmers, when new records are added or the status changes.

USAGE

The main benefit of a bug-tracking system is to provide a clear centralized overview of development requests (including both bugs and improvements, the boundary is often fuzzy), and their state. The prioritized list of pending items (often called backlog) provides valuable input when defining the product road map, or maybe just "the next release".

In a corporate environment, 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.

A local bug tracker (LBT) is usually a computer program used by a team of application support professionals (often a help desk) to keep track of issues communicated to software developers. Using an LBT allows support profes-

sionals to track bugs in their "own language" and not the "language of the developers." In addition, an LBT allows a team of support professionals to track specific information about users who have called to complain — this information may not always be needed in the actual development queue. Thus, there are two tracking systems when an LBT is in place.

BUG TRACKING SYSTEMS AS A PART OF INTEGRATED PROJECT MANAGEMENT SYSTEMS

Bug and issue tracking systems are often implemented as a part of integrated project management systems. This approach allows including bug tracking and fixing in a general product development process, fixing bugs in several product versions, automatic generation of a product knowledge base and release notes.

Distributed bug tracking

Some bug trackers are designed to be used with distributed revision control software. These distributed bug trackers allow bug reports to be conveniently read, added to the database or updated while a developer is offline.^[3]Fossil and Veracity both include distributed bug trackers.

Recently, commercial bug tracking systems have also begun to integrate with distributed version control. FogBugz, for example, enables this functionality via the source-control tool, Kiln.^[4]

Although wikis and bug tracking systems are conventionally viewed as distinct types of software, ikiwiki can also be used as a distributed bug tracker. It can manage documents and code as well, in an integrated distributed manner. However, its query functionality is not as advanced or as user-friendly as some other, non-distributed bug trackers such as Bugzilla.^[5] Similar statements can be made about org-mode, although it is not wiki software as such.

BUG TRACKING AND TEST MANAGEMENT

While traditional test management tools such as HP Quality Center and IBM Rational Quality Manager come with their own bug tracking systems, other tools integrate with popular bug tracking systems.

CONCLUSIONS

It is a comparison of issue tracking systems which are notable, including bug tracking systems, help desk and service desk issue tracking systems, and asset management systems. The comparison includes client-server application, distributed and hosted systems.

REFERENCE

1. ^ Joel Spolsky (November 8, 2000). "Painless Bug Tracking". Retrieved 29 October 2010. | 2. ^ Multiple (wiki). "Bug report". Docforge. Retrieved 2010-03-09. | 3. ^ Jonathan Corbet (May 14, 2008). "Distributed bug tracking". LWN.net. Retrieved 7 January 2009. | 4. ^ "FogBugz Features". Fogbugz.com. Retrieved 2010-10-29. | 5. ^ Joey Hess (6 April 2007). "Integrated issue tracking with ikiwiki". LinuxWorld.com. IDG. Retrieved 7 January 2009. |