

Name: Umair ul mulk

ID: 14136

Course Name: Software verification and validation

Date: 15/4/2020

Testing tool: BUGZILLA

- **Bugzilla:-**

Bugzilla is a "Defect Tracking System" or "Bug-Tracking System". Defect Tracking Systems allow individual or groups of developers to keep track of outstanding bugs in their product effectively.

- **Pros and Cons of Bugzilla :-**

- **Pros :-**

- Whenever a bug is logged or its status is changed then an email notification is triggered to the concerned persons without any delay.
- Its interface is very easy to understand and work on instead of having any extra technical knowledge.
- It has a good documentation part where can filter out all bugs as per its status. And the report is very beautiful and very easily readable.

- It is an open source and free to use. SO great to have without purchasing any license.

➤ Cons :-

- There should be some sort of association of the multiple affected test cases from the bug which I cannot see.
- There should be a place for a regression test suite in Bugzilla so that any bug found after every regression suite execution can be termed as regression testing failure result.
- Documentation is limited to Bugs instead it should more of project-focused and display the health of a project with the bugs being found or resolved.
- It cannot be part of agile methodologies as no sprint association can happen.



● **Functionalities of Bugzilla :-**

➤ Functionalities for users :-

- Advanced Search Capabilities
- Email Notifications Controlled By User Preferences
- Bug Lists in Multiple Formats (Atom, iCal, etc.)
- Scheduled Reports (Daily, Weekly, Hourly, etc.) by Email
- Reports and Charts
- Automatic Duplicate Bug Detection

- File/Modify Bugs By Email
- Time Tracking
- Request System
- Private Attachments and Comments
- Automatic Username Completion or Drop-Down User Lists
- Patch Viewer
- "Watch" Other Users
- Move Bugs Between Installs
- Save and Share Searches.

➤ Functionalities for administrators :-

- Excellent Security
- Extension Mechanism for Highly Customizable Installations
- Custom Fields
- Custom Workflow
- Full Unicode Support
- Localization
- mod_perl Support for Excellent Performance
- Webservices (XML-RPC) Interface
- Control Bug Visibility/Editing with Groups
- Impersonate Users
- Multiple Authentication Methods
- Support for Multiple Database Engines
- Sanity Check.



- **Supporting languages of Bugzilla :-**

- PERL
- PERL-6
- PYTHON
- RUBY
- JAVA
- PHP5 or 6
- D programming
- C#



- **Supporting tests of Bugzilla :-**

- Optimized database structure for increased performance and scalability
- Excellent security to protect confidentiality
- Advanced query tool that can remember your searches
- Integrated email capabilities
- Editable user profiles and comprehensive email preferences
- Comprehensive permissions system
- Proven under fire as Mozilla's bug tracking system.



- **Code :-**

- To output an HTML template to the browser, you can use this code:

```
1) ...
2) my $cgi = Bugzilla->cgi;
3) ...
4) print $cgi->header();
5) $template->process("file-name.html.tmpl", $vars)
6)    || ThrowTemplateError($template->error());
```

Bug:-

Bugzilla var is not defined,
We have to first use Bugzilla like this.
(use Bugzilla;)



- **How to send and receive information from database :-**

Bugzilla is using standard DBI functions to interact with the database, through `Bugzilla->dbh`. The current recommended method for creating a query is as follows:

```
1) use Bugzilla;
2) my $dbh = Bugzilla->dbh; # Connects if not already
3) connected.
                                     # Also handles db, user,
4) password...
5) my $data = $dbh->selectall_arrayref("SELECT foo, bar
6) FROM bath WHERE log = ?",
7)                                     undef, "foobar");

8) foreach my $row (@$data) {
9)     my ($foo, $bar) = @$row
    # do whatever with $foo and $bar
10) }
```

Bug: -

In line 9 semicolon(;) is missing.



THE END

