Name : Vidhi Manglik **QE Competency**

Assignment Bug Reporting and Test Management

1. What is bug In Software Testing?

Ans. In Software testing, when the expected and actual behavior is not matching, an incident is raised which is known as Defect and this defect is found by the tester. Moreover, the tester reports this incident to the developer and if the defect is acknowledged by the developer it becomes a bug.

2. Differentiate Error, Defect, And Failure?

Ans.

ERROR: An error is a mistake the part of a software developer.

DEFECT: It can be simply defined as a difference between expected and actual. A defect is an error found after the application goes for quality assurance.

FAILURE: A failure is the inability of a software system or component to perform its required functions within specified performance requirements.

3. What Are The Different Types Of Status Of Defects?

Ans.

New: Tester provides new status while Reporting (for the first time)

Open: Developer / Dev lead /DTT opens the Defect

Rejected: Developer / Dev lead /DTT rejects if the defect is invalid or defect is duplicate.

Fixed: Developer provides fixed status after fixing the defect

Deferred: Developer provides this status due to time etc...

Closed: Tester provides closed status after performing confirmation Testing

Re-open: Tester Re-opens the defect with valid reasons and proofs.

4. Explain About the Defect/Bug Life Cycle?

Ans.

New: When a defect is logged and posted for the first time. Its state is given as new.

Assigned: When the tester assigns the bug to developer its status changes to “assigned”

Open: At this state, the developer has started analyzing and working on the defect fix.

Fixed: When the developer makes necessary code changes and verifies the changes then he/she can make bug status as ‘Fixed’ and the bug is passed to the testing team.

Pending retest: After fixing the defect the developer has given that particular code for retesting to the tester. Here the testing is pending on the testers end. Hence its status is pending retest.

Retest: At this stage, the tester does the retesting of the changed code which the developer has given to him to check whether the defect got fixed or not.

Verified: The tester tests the bug again after it got fixed by the developer. If the bug is not present in the software, he approves that the bug is fixed and changes the status to “verified”.

Reopen: If the bug still exists even after the bug is fixed by the developer, the tester changes the status to “reopened”. The bug goes through the life cycle once again.

Closed: Once the bug is fixed, it is tested by the tester. If the tester feels that the bug no longer exists in the software, he changes the status of the bug to “closed”. This state means that the bug is fixed, tested and approved.

Duplicate: If the bug is repeated twice or the two bugs mention the same concept of the bug, then one bug status is changed to “duplicate“.

Rejected: If the developer feels that the bug is not genuine, he rejects the bug. Then the state of the bug is changed to “rejected”.

Deferred: The bug, changed to deferred state means the bug is expected to be fixed in the next releases.

Not a bug: The state given as “Not a bug” if there is no change in the functionality of the application. For an example: If the customer asks for some change in the look and feel of the application like a change of color of some text then it is not a bug but just some change in the look of the application.

5. A bug is identified by the tester it is assigned to whom?

Ans. After the tester has posted the bug, the lead of the tester approves that the bug is genuine and he assigns the bug to the corresponding developer and the developer team.

6. Why is JIRA used? Explain step by step how an issue is created in JIRA.

Ans. Jira is a proprietary issue tracking product developed by Atlassian which allows bug tracking and agile project management. It is used for bug tracking, issue tracking, and project management.

Steps to create an issue in JIRA-

- ➔ **Search for issues** option will bring you to a window where you can see the issues created by you like here we have issues ST1 and ST2
- ➔ You can see the issue "**Bug detected while User Acceptance Testing**" and all the details related to it. From here, you can perform multiple tasks like you can **stop the progress on issues, edit the issues, comment on the issues, assigning issues** and so on
- ➔ Even you can export issue details to an XML or Word document.
- ➔ Also, you can view activity going on the issue, reviews on the issue, work log, history of the issue and so on.
- ➔ Under the time tracking option, you can even see the estimation time to resolve the issues
- ➔ Click **Create** at the top of the screen to open the **Create Issue** dialog box.
- ➔ Select the relevant **Project** and **Issue Type** in the **Create Issue** dialog box.
- ➔ Type a **Summary** for the issue and complete any appropriate fields — at least the required ones that are marked by an asterisk.
- ➔ If you want to access fields that are not shown in this dialog box, or you want to hide existing fields:
- ➔ Click the **Configure Fields** button at the top right of the screen.
- ➔ Click **Custom** and select the fields you want to show or hide by selecting or clearing the relevant checkboxes respectively, or click **All** to show all fields.
- ➔ When you next create an issue, these selected fields will be displayed.
- ➔ Optional: To create a series of similar issues – with the same **Project** and **Issue Type** – select the **Create another** checkbox at the bottom of the dialog. Depending on your configuration and the values you may have specified when creating previous issues, some of the fields in the new Create Issue dialog box may be pre-populated. Make sure you check they're all correct before creating the next issue.
- ➔ When you are satisfied with the content of your issue, click the **Create** button.

7. What is Defect Density?

Ans. **Defect Density** is the number of defects confirmed in software/module during a specific period of operation or development divided by the size of the software/module.(Defect Density = Defect count/ size of the release)

8. What is the difference between defect density and defect triage?

Ans.

- ➔ **Defect Density** is the number of defects confirmed in software/module during a specific period of operation or development divided by the size of the software/module.
  (Defect Density = Defect count/ size of the release)
  It enables one to decide if a piece of software is ready to be released.
- ➔ **Defect triage** is a process where each bug is prioritized based on its severity, frequency, risk, etc.

9. Explain Bug reporting and parameters of bug?

Ans. Bug reporting is a process in which if a bug occurs, the person finding the bug should be able to report (document & send) the bug to people in charge of fixing that error or failure. A well written bug report helps developer to reproduce bug at their terminal. This helps them to diagnose the issue.

Parameters of Bug:

- ➔ **Defect ID** - Unique identification number for the defect.
- ➔ **Defect Summary/Title** - It must be clear and descriptive so that you get an idea of what it is about at a glance.
- ➔ **Defect Description** - Detailed description of the Defect including information about the module in which Defect was found. It helps the developer to understand the Defect.
- ➔ **Version** - Environment/Browser Version in which defect was found.
- ➔ **Date Raised** - Date when the defect is raised
- ➔ **Detected By** - Name/ID of the tester who raised the defect
- ➔ **Status** - Status of the defect
- ➔ **Assignee** - Name of the developer whom it is assigned to fix
- ➔ **Severity** which describes the impact of the defect on the application
- ➔ **Priority** is related to defect fixing urgency.

10. What is defect management? Explain the defect management process.

Ans. Defect management can be defined as a process of detecting bugs and fixing them. It is necessary to say that bugs occur constantly in the process of software development. Hence, every software development project requires a process that helps detect defects and fix them.

The process of defect management usually includes four steps.

- ➔ The first step is the stage of defect detecting. We have already mentioned that it can be conducted either by the team of developers or by the users. Regardless of the type of testing, its main goal is to detect all bugs in the final product or its part.

- ➔ The second step of the bug management process is dedicated to the formulation of bug reports. These are the documents that include all the necessary information about certain bugs. Usually, they contain data on the type of bug and the possible ways of its correction.
- ➔ The third step is the stage of bug fixing. After the bugs are fixed, they should be tested once more to make sure that the software works properly.
- ➔ During the final step, the bug list is created. This is the document that contains information about all bugs that occurred during the project's performance. The team often uses the bug list because similar bugs' occurrence is not rare.

11. What is Test estimation? Explain Work Breakdown Structure test estimation technique with an example?

Ans. Test Estimation is a management activity that approximates how long a Task would take to complete.

In the Work Breakdown Structure test estimation technique, a complex project is divided into modules. The modules are divided into sub-modules. Each sub-module is further divided into functionality. It means to divide the whole project task into the smallest tasks.

12. What are test reports? What parameters are used in test reports?

Ans. **Test Reports** are a document that contains a summary of test activities and final test results.

Parameters used in Test Reports:

- ➔ Summary
- ➔ Execution Type
- ➔ Estimated Duration
- ➔ Priority

13. What are the test management tools?

Ans. **Test management tools** are used to store information on how testing is to be done, plan testing activities and report the status of quality assurance activities. The tools have different approaches to testing and thus have different sets of features. Generally, they are used to maintain and plan manual testing, run or gather execution data from automated tests, manage multiple environments and to enter information about found defects. For Eg: Test Link

14. What is a test link? How do you write test cases in TestLink?

Ans. Test-link is the most widely used web-based open source test management tool. It synchronizes both the requirements specification and test specification together. A user can create a test project and document test cases using this tool. With Test-Link you can create an account for multiple users and assign different user roles.

To write test cases in TestLink

➔ **Creating a Test Project**

Click on the tab "create" to create a new project.

Enter all the required fields in the window like a category for a test project, name of the project, prefix, description, etc. After filling all the necessary details, click on the tab "Create" at the end of the window.

➔ **Creating a Test Plan**

From the home-page, click on Test Plan Management from home-page

It will open another page, at the bottom of the page click on a tab "Create".

Fill out all the necessary information like name, description, create from the existing test plan, etc. in the open window, and click on "create tab"

➔ **Build Creation**

Click on Builds/Releases under Test Plan from the home page

➔ **Creating Testsuite**

Click on the test specification option from the home page.

On the right-hand side of the panel, click on the setting**.** It will display a series of test operations.

Click on the "create" tab for the test suite

Fill-up all the details for test-suite and click on save it tab.

Enter the test suite name

Enter the details about your test suite

Click on save button to save the details of test-suite

➔ **Creating a Testcase**

Click on the test suite folder on the left side of the panel under a folder tree structure.

Click on the setting icon in the right side panel. A list of test case operations will be displayed on the right side panel. A new window will open, to create test cases click to create a test case.

15. Explain steps how to upload the Test case sheet on TestLink?

Ans.

Step 1 − To import test cases, go to Test Specifications → Test Specification from the dashboard.

Step 2 − Select the nearest test suite folders, where the test cases should be imported.

Step 3 − Click the Actions icon on the right pane.

It displays Test Case Operations.

Step 4 − Click the Import icon

Step 5 - Select the file and upload it.

Step 6 - Click the Upload file button.

16. What is the severity and priority in bug/defect?

Ans. Severity is defined as the degree of impact a Defect has on the development or operation of a component application being tested.

Priority is defined as the order in which a defect should be fixed.

17. While placing an order for clothing website, in order confirmation page there is a logo error. It is a?
    a) High priority, high severity
    b) Low severity low priority
    c) Low severity, high priority (of low severity as it not going to affect the functionality of the website but can be of high priority as you don't want any further shipment to proceed with the wrong logo.)
    d) High severity low priority

Ans. c) Low severity, high priority

18. The website home page failed to load.
    a) High priority, high severity (Major functionality failure like log in is not working, crashes in the basic workflow of the software are the best example of High Priority and High Severity)
    b) Low severity low priority
    c) Low severity, high priority
    d) High severity low priority

Ans. a) High priority, high severity

19. The application works perfectly for 50k sessions but beings to crash after a higher number of sessions.
    a) Low severity low priority
    b) High priority, high severity
    c) Low severity, high priority
    d) High severity low priority (This problem needs to be fixed but not immediately.)

Ans. d) High severity low priority

20. An application (web) is made up of 20 pages. On one of the pages, there is a sentence with a grammatical error.
    a) Low severity low priority - This bug may go unnoticed to the eyes of many and won't affect any functionality or the credibility of the company.
    b) High priority, high severity
    c) Low severity, high priority
    d) High severity low priority

Ans. a) Low severity low priority

21. Find bugs and report the same on JIRA for below-mentioned modules in the website: **URL:** Testwebsite1
    1. My Account
    2. Add to basket
    3. Search
    4. Homepage

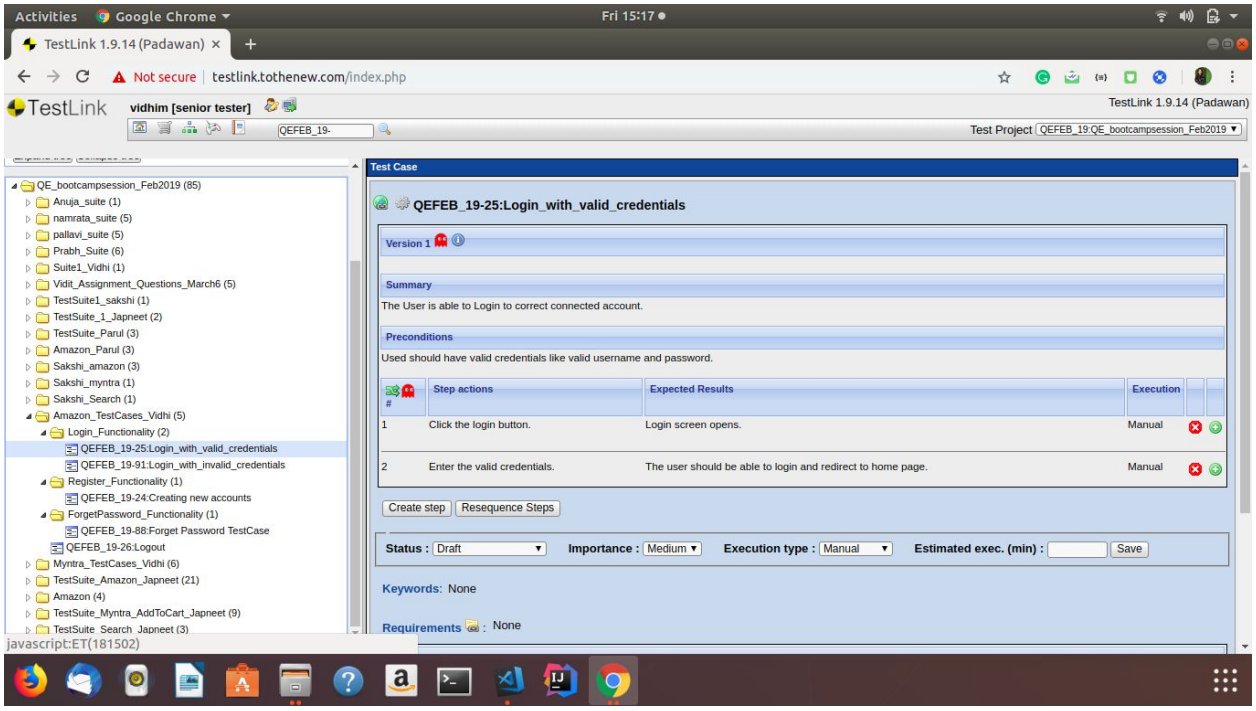
Ans. Assigned On Jira

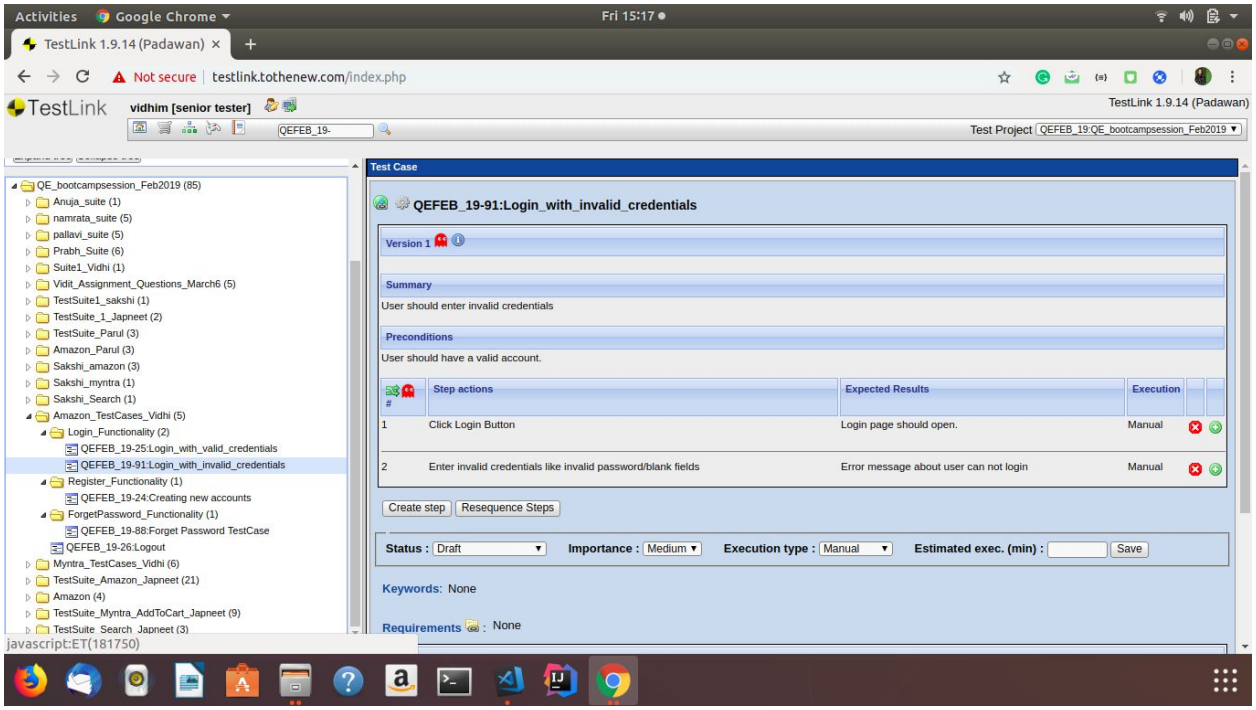
22. Find bugs and report the same on JIRA for below-mentioned modules in the website: URL: Testwebsite2
    1. Join Now
    2. Top header navigation options
    3. UI bugs for the complete website.

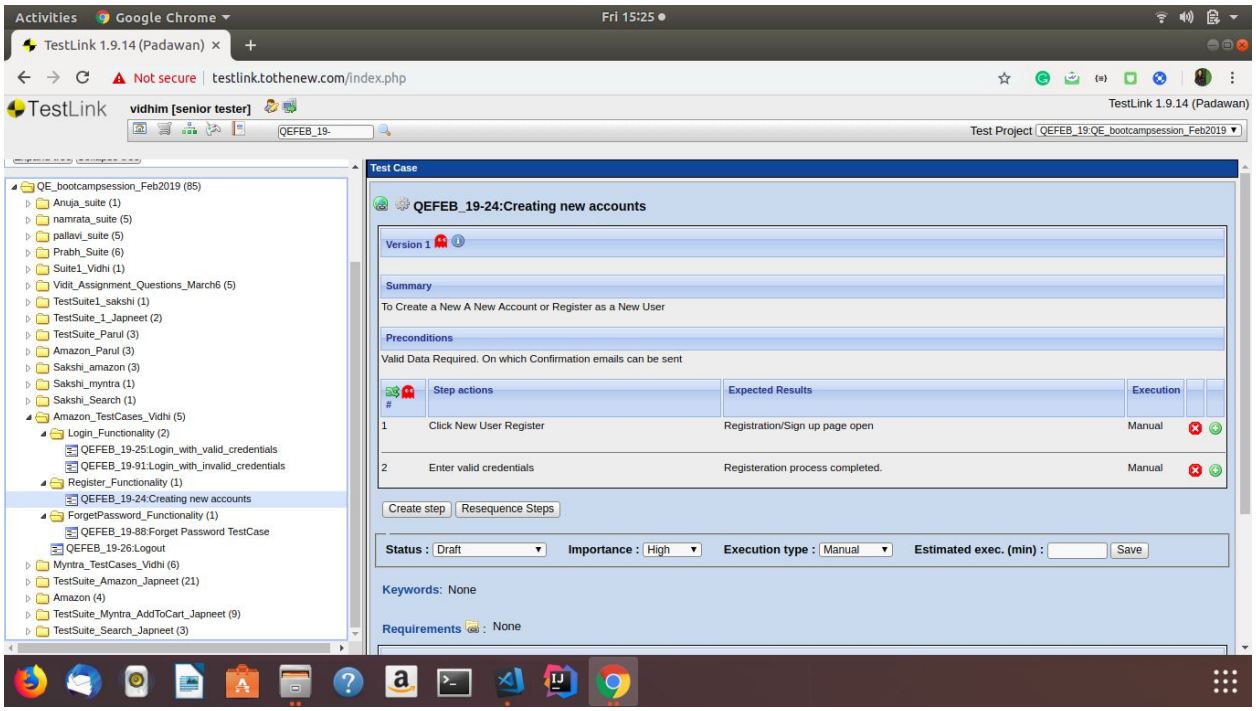
Ans. Assigned On Jira

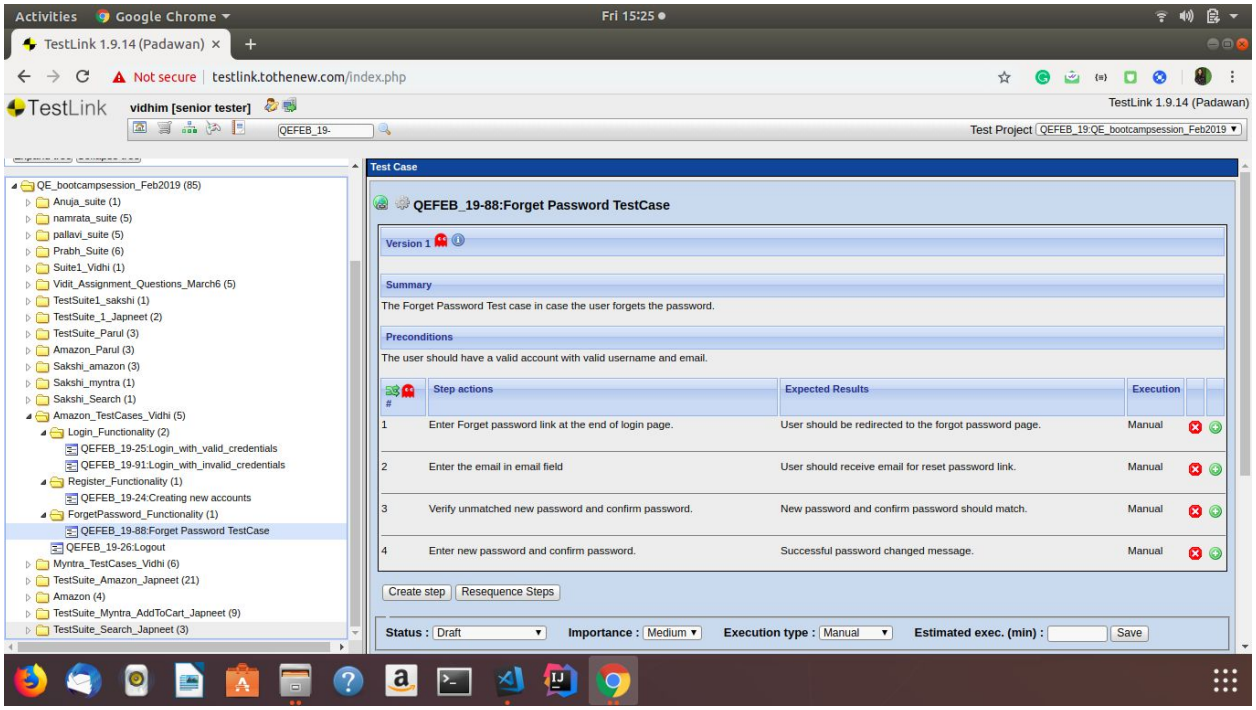
23. Write Test Cases for Amazon login, Sign up and Forgot password on TestLink.

Ans.



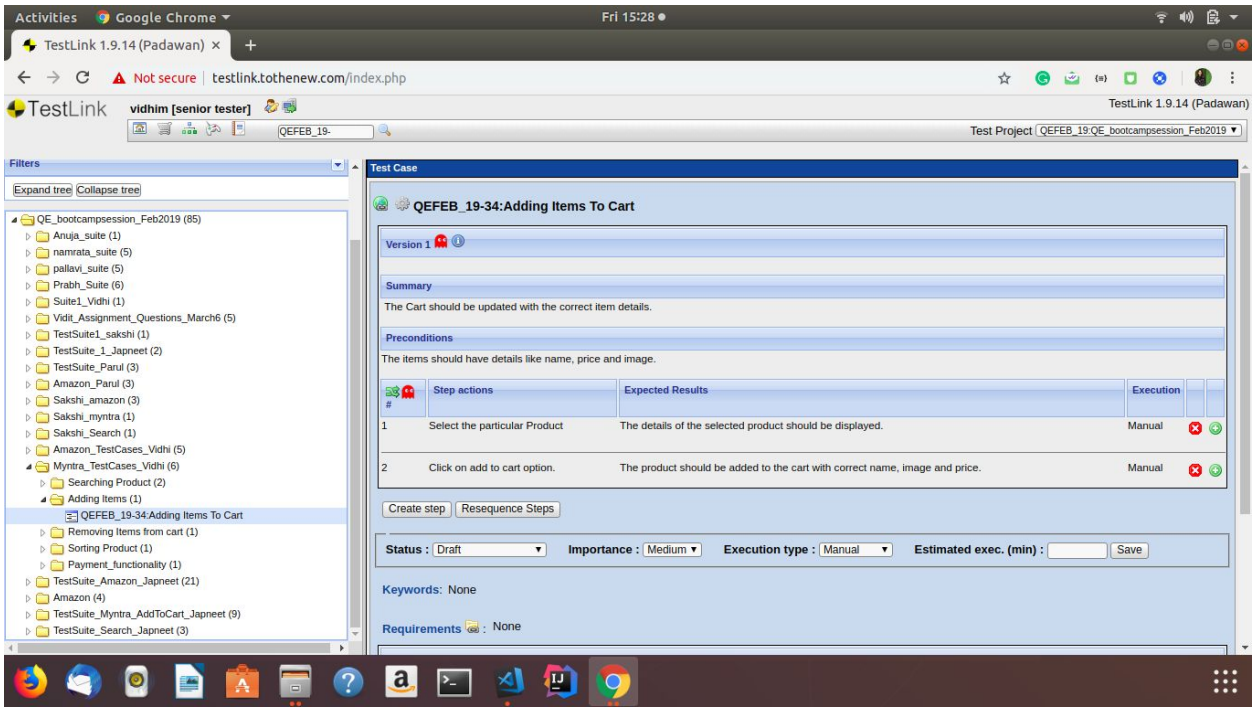

Activities
Google Chrome
Fri 15:17
TestLink 1.9.14 (Padawan)
Not secure
testlink.tothenew.com/index.php
TestLink
vidhim [senior tester]
TestLink 1.9.14 (Padawan)
QEFEB_19-
Test Project
QEFEB_19:QE_bootcampsession_Feb2019
QE_bootcampsession_Feb2019 (85)
Anuja_suite (1)
namrata_suite (5)
pallavi_suite (5)
Prabh_Suite (6)
Suite1_Vidhi (1)
Vidit_Assignment_Questions_March6 (5)
TestSuite1_sakshi (1)
TestSuite_1_Japneet (2)
TestSuite_Parul (3)
Amazon_Parul (3)
Sakshi_amazon (3)
Sakshi_myntra (1)
Sakshi_Search (1)
Amazon_TestCases_Vidhi (5)
Login_Functionality (2)
QEFEB_19-25:Login_with_valid_credentials
QEFEB_19-91:Login_with_invalid_credentials
Register_Functionality (1)
QEFEB_19-24:Creating new accounts
ForgetPassword_Functionality (1)
QEFEB_19-88:Forget Password TestCase
QEFEB_19-26:Logout
Myntra_TestCases_Vidhi (6)
TestSuite_Amazon_Japneet (21)
Amazon (4)
TestSuite_Myntra_AddToCart_Japneet (9)
TestSuite_Search_Japneet (3)
javascript:ET(181750)
Test Case
QEFEB_19-91:Login_with_invalid_credentials
Version 1
Summary
User should enter invalid credentials
Preconditions
User should have a valid account.
Step actions Expected Results Execution
1 Click Login Button Login page should open. Manual
2 Enter invalid credentials like invalid password/blank fields Error message about user can not login Manual
Create step
Resequence Steps
Status : Draft
Importance : Medium
Execution type : Manual
Estimated exec. (min) :
Save
Keywords: None
Requirements : None

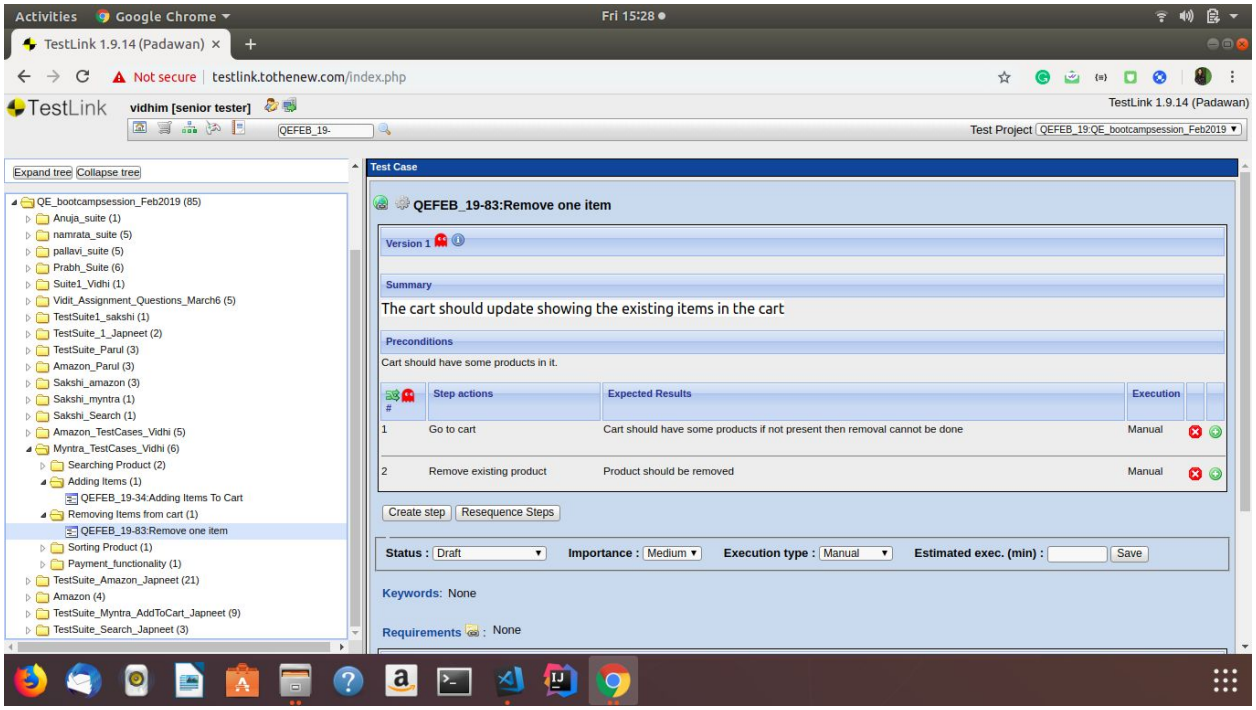

Activities
Google Chrome
Fri 15:25
TestLink 1.9.14 (Padawan)
Not secure
testlink.tothenew.com/index.php
TestLink
vidhim [senior tester]
TestLink 1.9.14 (Padawan)
QEFEB_19-
Test Project
QEFEB_19:QE_bootcampsession_Feb2019
QE_bootcampsession_Feb2019 (85)
Anuja_suite (1)
namrata_suite (5)
pallavi_suite (5)
Prabh_Suite (6)
Suite1_Vidhi (1)
Vidit_Assignment_Questions_March6 (5)
TestSuite1_sakshi (1)
TestSuite_1_Japneet (2)
TestSuite_Parul (3)
Amazon_Parul (3)
Sakshi_amazon (3)
Sakshi_myntra (1)
Sakshi_Search (1)
Amazon_TestCases_Vidhi (5)
Login_Functionality (2)
QEFEB_19-25:Login_with_valid_credentials
QEFEB_19-91:Login_with_invalid_credentials
Register_Functionality (1)
QEFEB_19-24:Creating new accounts
ForgetPassword_Functionality (1)
QEFEB_19-88:Forget Password TestCase
QEFEB_19-26:Logout
Myntra_TestCases_Vidhi (6)
TestSuite_Amazon_Japneet (21)
Amazon (4)
TestSuite_Myntra_AddToCart_Japneet (9)
TestSuite_Search_Japneet (3)
Test Case
QEFEB_19-24:Creating new accounts
Version 1
Summary
To Create a New A New Account or Register as a New User
Preconditions
Valid Data Required. On which Confirmation emails can be sent
Step actions Expected Results Execution
1 Click New User Register Registration/Sign up page open Manual
2 Enter valid credentials Registeration process completed. Manual
Create step
Resequence Steps
Status : Draft
Importance : High
Execution type : Manual
Estimated exec. (min) :
Save
Keywords: None
Requirements : None

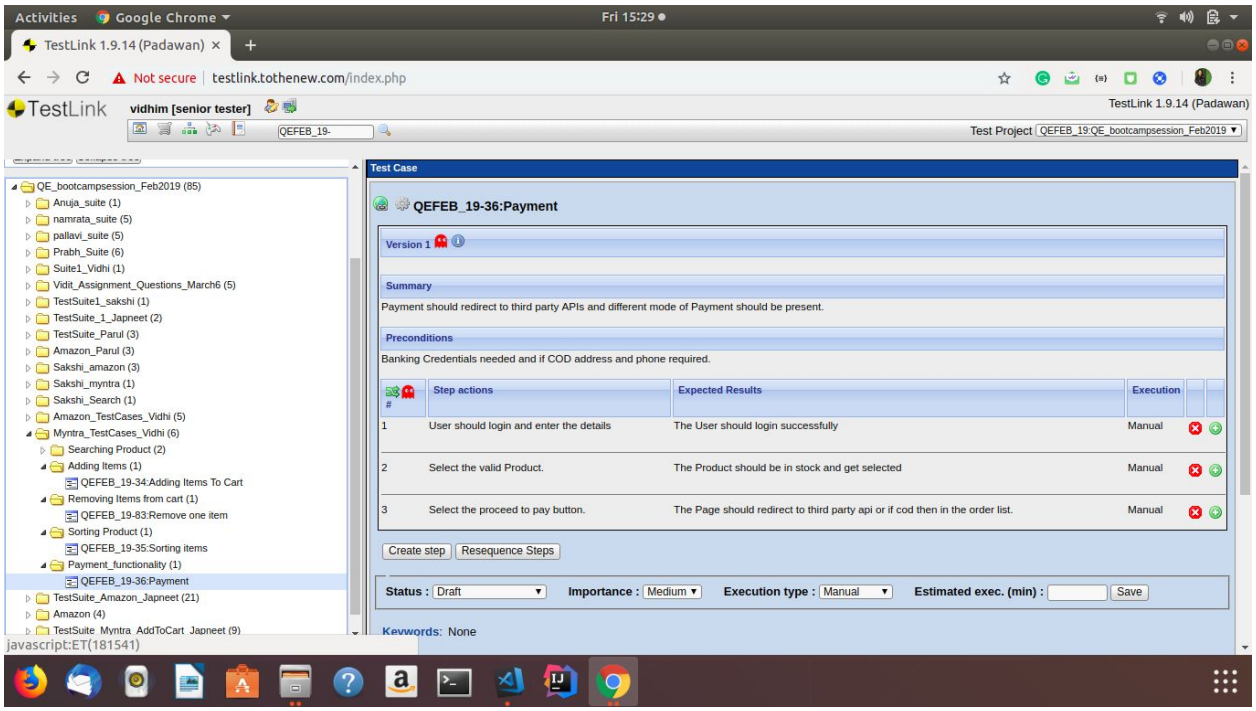


24. Write Test Cases for placing an order in Myntra on TestLink.

Ans.

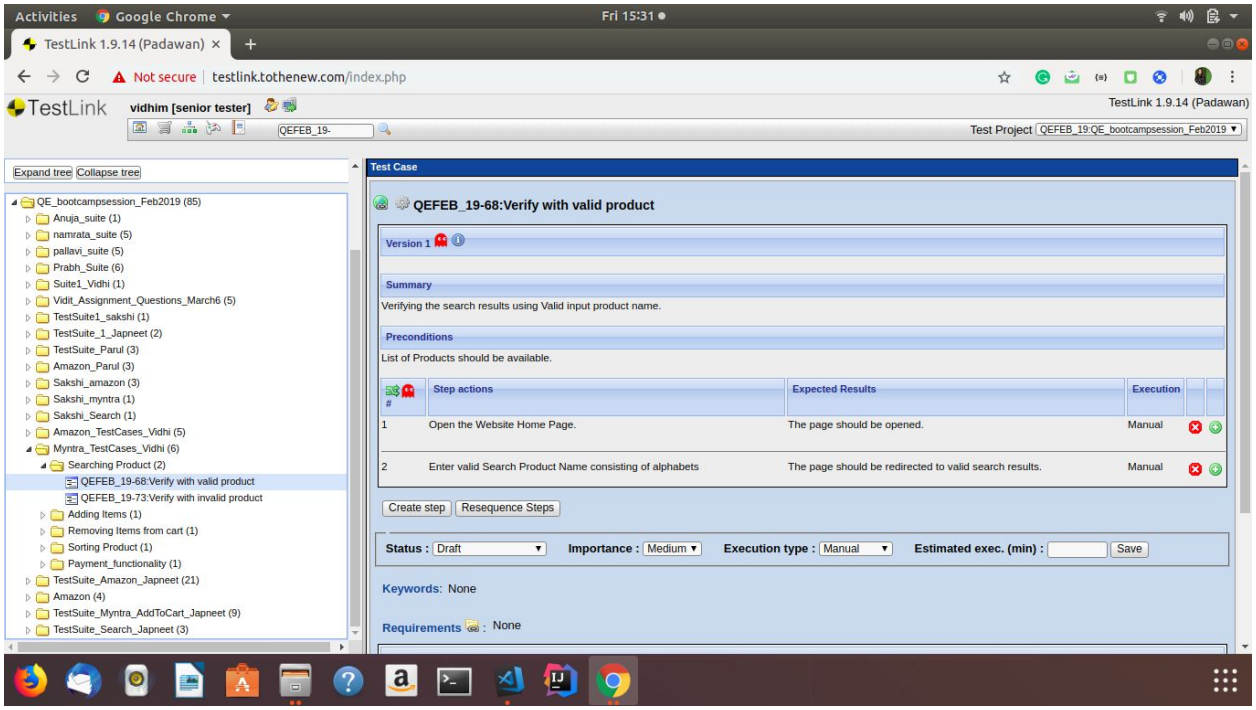


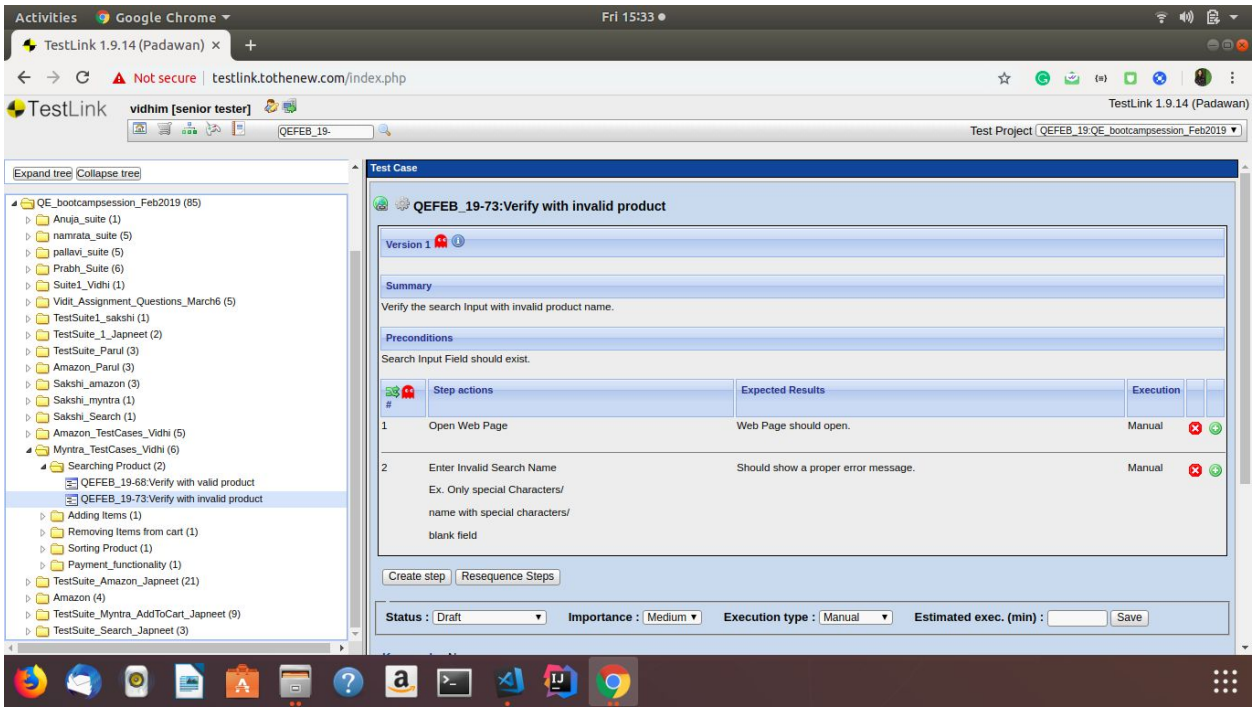




25. Write Test Cases for Search functionality on TestLink.

Ans.


Activities
Google Chrome
Fri 15:31
TestLink 1.9.14 (Padawan)
Not secure
testlink.tothenew.com/index.php
TestLink
vidhim [senior tester]
TestLink 1.9.14 (Padawan)
QEFEB_19-
Test Project
QEFEB_19:QE_bootcampsession_Feb2019
Expand tree
Collapse tree
QE_bootcampsession_Feb2019 (85)
Anuja_suite (1)
namrata_suite (5)
pallavi_suite (5)
Prabh_Suite (6)
Suite1_Vidhi (1)
Vidit_Assignment_Questions_March6 (5)
TestSuite1_sakshi (1)
TestSuite_1_Japneet (2)
TestSuite_Parul (3)
Amazon_Parul (3)
Sakshi_amazon (3)
Sakshi_myntra (1)
Sakshi_Search (1)
Amazon_TestCases_Vidhi (5)
Myntra_TestCases_Vidhi (6)
Searching Product (2)
QEFEB_19-68:Verify with valid product
QEFEB_19-73:Verify with invalid product
Adding Items (1)
Removing Items from cart (1)
Sorting Product (1)
Payment_functionality (1)
TestSuite_Amazon_Japneet (21)
Amazon (4)
TestSuite_Myntra_AddToCart_Japneet (9)
TestSuite_Search_Japneet (3)
Test Case
QEFEB_19-68:Verify with valid product
Version 1
Summary
Verifying the search results using Valid input product name.
Preconditions
List of Products should be available.
Step actions Expected Results Execution
1 Open the Website Home Page. The page should be opened. Manual
2 Enter valid Search Product Name consisting of alphabets The page should be redirected to valid search results. Manual
Create step
Resequence Steps
Status : Draft
Importance : Medium
Execution type : Manual
Estimated exec. (min) :
Save
Keywords: None
Requirements : None


Activities
Google Chrome
Fri 15:33
TestLink 1.9.14 (Padawan)
Not secure
testlink.tothenew.com/index.php
TestLink
vidhim [senior tester]
TestLink 1.9.14 (Padawan)
QEFEB_19-
Test Project
QEFEB_19:QE_bootcampsession_Feb2019
Expand tree
Collapse tree
QE_bootcampsession_Feb2019 (85)
Anuja_suite (1)
namrata_suite (5)
pallavi_suite (5)
Prabh_Suite (6)
Suite1_Vidhi (1)
Vidit_Assignment_Questions_March6 (5)
TestSuite1_sakshi (1)
TestSuite_1_Japneet (2)
TestSuite_Parul (3)
Amazon_Parul (3)
Sakshi_amazon (3)
Sakshi_myntra (1)
Sakshi_Search (1)
Amazon_TestCases_Vidhi (5)
Myntra_TestCases_Vidhi (6)
Searching Product (2)
QEFEB_19-68:Verify with valid product
QEFEB_19-73:Verify with invalid product
Adding Items (1)
Removing Items from cart (1)
Sorting Product (1)
Payment_functionality (1)
TestSuite_Amazon_Japneet (21)
Amazon (4)
TestSuite_Myntra_AddToCart_Japneet (9)
TestSuite_Search_Japneet (3)
Test Case
QEFEB_19-73:Verify with invalid product
Version 1
Summary
Verify the search Input with invalid product name.
Preconditions
Search Input Field should exist.
Step actions Expected Results Execution
1 Open Web Page Web Page should open. Manual
2 Enter Invalid Search Name
Ex. Only special Characters/
name with special characters/
blank field
Should show a proper error message. Manual
Create step
Resequence Steps
Status : Draft
Importance : Medium
Execution type : Manual
Estimated exec. (min) :
Save