Importing Zimbra Test Automation Reports to Testrail.

**Task Description**

Zimbra QA team is migrating all test cases to Testrail. We have automation frameworks, which are managed through TMS. But we don’t have any mapping between manual test cases and automated test cases. Testrail provides APIs to import automation test results to Testrail test runs. We need to identity way to import TMS automation reports to Testrail.

This document will describe probable solution to this problem.

**Current Challenges - known**

From Zimbra Side

1. Different automation frameworks create reports in different formats.
2. No mapping between current testrail test cases and automated test cases.
3. All automation frameworks identify test cases differently.
4. Currently no provision is made to supply necessary parameters like project\_id, suite\_id to test cases.
5. Getting complete understanding of Testrail APIs.

From Testrail side

1. Testrail supports only json format to import automation reports.
2. Testrail has their own mechanism to generate test cases ids. i.e. There is no way to identify test case ids before head.

We might come across some unknown issues once we actually start implementing this task.

**TestRail\_API – Utility to Import automation reports**

Synacor QE team has already written utility(testrail\_api.rb) for importing automation test results to Testrail. This utility can be found at below location

ssh://git@stash.corp.synacor.com:7999/qe/cucumber-ruby\_te

strail\_utilities.git

All Zimbra automation frameworks can use this utility. Following are some example of using this utility.

To demonstrate use of this utility – We have created sample test run which contains only one test case under test suite “testapi “. The test suite is created under “Zimbra Collaboration QE”.

Utility requires input as json file with following mandatory fields. Below is sample of json file…

{

"project\_id": 16, //Zimbra Collaboration QE

"suite\_id": 1130, //testapi

"case\_ids": [

"280399" //testapi

],

"results": [

{

"status\_id": 1, //1=pass, 5=fail, 2=block etc

"comment": "Testing API again",

"case\_id": "280399" //280399 is test cases id

} ]

}

Apart from this we can also include fields like run\_name, description, assigned\_to\_id, milestone\_id, plan\_name.

Once you have json file following is the way to import automation results …

$:ruby testrail\_api.rb -f test.json -u amit.jagtap@synacor.com -k loOQCfBGQBgmWfbZ7GVt-XXXXXXXXXXXXXX

test.json -- test report in json format

-u -- Username by whose name test run will be updated.

-k -- Testrail API key for user

You can generate key for your use through testrail user interface. Go to Username 🡪 My settings 🡪 API keys and generate key for you.

Above command will update testcase id 280399 as pass of test suite id 1130 created under project id 16. It will also add comment provided in results section.

The example shown above updates result only for single test case. If you want to update results for multiple test cases then we need to provide multiple case\_ids in json file.

**What needs to be done in Zimbra Automation frameworks?**

ZIMBRA automation frameworks can utilize this utility to import test results in Testrail. But to do that we have to deal with following challenges

1. Create mapping between automated and manual test cases. This can be achieved in one of the two ways
   1. Import all automated test cases to test rail and update them with test steps.
   2. Identify automated test cases for already present manual test cases in testrail.
2. Update all automated test cases with test case ids that are generated in testrail.
3. Every automation framework needs to make provision for supplying test rail parameters to test cases/test suites.
4. Each framework need to generate reports in json format. Probably we can take hints from testrail\_formatter.rb present in cucumber-ruby\_testrail\_utilities project.
5. We need to provision separate user for updating test results.
6. Once all above tasks are done then we just need to use testrail\_api utility as shown in demonstration to upload results.

**Conclusion**

With testrail\_api utility already available we just need to overcome challenges mentioned above. The most complicated part seems to conversion of all reports to json format because all automation frameworks are developed in house. So no ready help available for this report conversion on Internet.