Test Execution and Closure:

In-Sprint Automation:

Contributed in In-sprint automation and automated 94 test cases out of 153 which helped in completion of all tasks as per test schedule. Due to In-sprint Automation contribution, I was able to execute all progression test cases everyday which found defects earlier and made the product quality as well leads to test execution and closure at the scheduled time.

Progression Execution:

Contributed in Progression test case execution which covers Negative scenarios, defects fix, Product changes and Migration work which was not happen through In-sprint Automation. Handled all Major US’s and executed 109 scenarios from the US’s which assigned to me.

Manual Regression:

Contributed in executing Manual test cases, logged test cases report. Executed 300 test cases out of 1448 test cases and contributed more in Automation regression cases which reduce manual efforts and time on delivery.

Sanity Regression:

Doing Maintenance of Sanity work which executing 116 test cases wherein fixing the script, providing new changes in script as per functionality changes and since it’s a gateway of other execution, following up developers if any defects logged and resolved it which absolutely helps in timely completion of all tasks as per test schedule.

One Click Regression:

Contributing to One Click Regression activities every release and supporting in executing 2800 test cases wherein providing initial set up, analyzing failures, fixing script and re-executing test cases which helps in executing more number of test cases.

In-Sprint Automation/ Progression Execution/ Manual Regression/ Sanity Regression/ One Click Regression in Quarter 2: Executed 3264 Test cases which includes both Regression and Progression in Quarter 2.

Test Design and Development:

Test cases/Test scenarios Designed:

I am involved in testing a project from OP Account with flexible timelines. I was documented test cases with best available template and got it approved from client/Developers. Once the build started releasing to QA team, My duty was, mechanically follow executing 10 to 15 test cases per day, update document with pass/fail result and attached logs.

Design of Test cases and Development.

Iam following the below steps to design test cases and development.

1. A concept which provides detailed information what to test, steps to be taken and expected result of the same
2. Since our testing is off shored and development is onsite. Writing test cases with details will help both dev and QA team in sync.
3. Good test coverage can be achieved by dividing application in test scenarios and it reduces repeatability and complexity of product

No of Test cases/scenarios designed:

Handled 14 US’s out of 26 in Q1 and designed 140 scenarios and 800 test cases which covered entire functionality and got approved by Developers and Client. Contribution of In-sprint automation completes huge number of scenarios execution quickly which approved by client and helps in timely completion of all test cases/test scenarios with complete coverage against the business requirements.

No of Test cases/scenarios designed in Quarter 2:

Handled 7 US’s out of 17 in Q2 and designed 70 scenarios and 420 test cases which covered entire functionality and got approved by Developers and Client. Contribution of In-sprint automation completes huge number of scenarios execution quickly which approved by client and helps in timely completion of all test cases/test scenarios with complete coverage against the business requirements

Test requirements Study

Requirements KT session:

Testers cannot assume that requirements will be well-defined and Static. So Iam effectively contributing in Requirements KT session such as FS Walkthrough, FS Review and most of the times scrum call which helped me in understanding exact details of the new requirements .

Strategy before the start of Test design and execution:

If I have requirements, many times I'll list the [use cases](http://searchsoftwarequality.techtarget.com/definition/use-case), specifications, or [stories](http://searchsoftwarequality.techtarget.com/definition/story) that I'm trying to cover as part of my US. Since my requirements is only a list of goals, and not actual steps, the impact of changing requirements is greatly reduced by using this method. If traceability is an issue, I've used several tools in the past to trace my requirements.

When executing my tests, if the requirement changed from the time I chartered my testing to the time I executed it, then I change my testing on the fly. If the requirement has changed too much from the original requirement and no longer fits with my mission for the test session, I'll simply note that I didn't test it and create a new charter for that requirement after I'm done testing.

# of defects Identified on Requirements Phase:

9 defects identified on Requirements Phase and developers provided a fix for all those identified defects which avoided defect leakage.

Test Effectiveness:

Test effectiveness of a technique or a system or a team is the ability to find defects and isolate them, from a product or deliverable. Test effectiveness is to ensure quality and close the two quality gaps, namely producer’s quality gap and customer’s quality gap. As definition of quality goes, quality is both process and product quality which is meeting customer requirements and conformance to product specification. These metrics should be quantified, as they closely relate to quality, and for many people the term quality is relative.

Test effectiveness Metrics.

*- Test effectiveness = Number of defects found / Number of test cases executed.  
- Test effectiveness = (total number of defects injected +total number of defect found) / (total number of defect escaped)\* 100  
- Test Effectiveness = Loss due to problems / Total resources processed by the system*

*Test Effectiveness in Q1.*

*Number of defects found= 33 defects*

*Number of Test cases executed=3826 Test cases.*

*Test effectiveness=32/3826=>0.0083/100=>0.83%*

*Test effectiveness in Q2*

*Number of defects found= 12 defects*

*Number of Test cases executed=3000 Test cases.*

*Test effectiveness=12/3000=>0.004/100=>0.4%*