**EXECUTIVE SUMMARY**

#### BACKGROUND:

The NJM HOPC, SIT , Documents , Automation , Integration , Refined Pricing and Billing Center QA project Team were audited with an intent of understanding the project progress and uncovering any gaps in the project and the processes being implemented in the project.

#### SCOPE:

The scope of the audit includes:

* Testing Processes
* Defect Management
* Test Management
* Test Tools
* Quality Management

# OBJECTIVES AND RESULTS:

The main findings of the audit show that the project is on track to meet the planned scheduled date of deployment. There are some areas in the project that have been well planned and successfully implemented. However, the focus of the audit report has been on audit areas that showed that there is a scope and in some cases a definite need for improvement. The suggested improvements and recommendations have been listed out in details in the next section. If none of the recommended changes are implemented on this project then there is an eminent danger of exposing many high severity defects in production or they will be discovered late which will impact the overall schedule and quality.

Following is the summary of the audit areas and related control assessment (Satisfactory, Needs Improvement, and Unsatisfactory):

|  |  |
| --- | --- |
| **Audit Areas** | **Control Assessment** |
| Testing Processes | **Needs Improvement** |
| Defect Management | **Satisfactory** |
| Test Management | **Needs Improvement** |
| Test Tools | **Unsatisfactory** |
| Quality Management | **Needs Improvement** |

#### OBSERVATIONS AND RECOMMENDATIONS:

Following tables detail all the observations from the audit along with recommendations to address these issues:

|  |  |
| --- | --- |
| **Observation #1** |  |
| **Risk Ranking** | **High** |
| **Problem: Documents Testing team following Waterfall Model whereas Dev follows Agile.** | |
| The Development Team is following an Agile Methodology whereas the QA Team is not. When you are developing in Agile, the concept is to have frequent builds deployed to the Test environment, for the Test team to identify the defects early on and then for the developers to fix and re-deploy those changes as soon as possible. QA Team is validating the Final build after the Development sprint is completed.  As a result, the number of "open" defects at any point of time during test execution will always be high because those defects are not fixed/deployed till the next build which is several weeks away. After all the delay, if the defect is not fixed properly and returned by the QA Team, it again won't be fixed till the next build (further 4 weeks away). The number of open high defects also affect the test execution as it might be either blocking or making several test cases ineffective/inaccurate till the defects are fixed.  **There is a high risk of Schedule and Scope Slippage in this scenario.** | |
|
|
| **Recommendations:** |  |
| 1. Team should plan frequent builds for defect fixes every week which will help improve :   * Defect Aging would be reduced. * Testing team utilization will be improved * Fewer fixes between Sprints * Scope creep will be reduced * Improved Communication between Dev and Test team * Flexibly incorporate new requirements. * Cycle time of Dev Sprint would be reduced. * WIP items for Dev team would be reduced.   2. The QA Team should run the smoke tests and the regression suite with every build so that  any new defects injected because of frequent builds are identified quickly. | |

**iiiiiii9999999999999999999999999999999999999999999999999999 ytinn5u7u6hkf7u b**

|  |  |
| --- | --- |
| **Observation #2** |  |
| **Risk Ranking** | **High** |
| **Problem: Interdepencies across Modules is not maintained** | |
| There are 14 modules in this program and VAM QA is involved in 7 Tracks including Automation and System Integration testing. Sprints are based on themes; HOPC, Documents, Integration, Refined Pricing and Billing Center QA team analyzes and work on the user stories for respective sprints.  SIT team is responsible for doing integration for Theme implemented by all the NJM VAM teams (HOPC, Documents, RP, BC, and Integration). In addition to this they are getting integration work w.r.t. modules which are handled by other vendors.  Requirements are not clearly articulated for SIT Team and team spends a lot of time in understanding the Business Interdependencies between all the modules. They schedule meetings with clients to understand the dependencies and design the SIT scenarios.  SIT Team is reaching out to functional teams internally and getting the details but for functional teams managed by other vendor, team is asking for Client intervention and receiving the details.  Also if in one sprint, Team1 has delivered a completed feature which requires to be tested by Team 2 and it was not prioritized by Client in that Sprint. Who is keeping track of this?  **There is a High risk of Scope Slippage in this scenario.** | |
|
|
| **Recommendations:** |  |
| * A detailed Program level Plan where Development plan for upcoming sprints are defined along with their interdepencies w.r.t. all the modules. * SIT Test team should be having a clearly defined scope for sprint and required information / knowledge to come up with SIT scenarios for impacted areas well before the start of the SIT Sprint. * SIT Team should have Functional team QA signoff for In Scope modules for current SIT sprint. | |

|  |  |
| --- | --- |
| **Observation #3** |  |
| **Risk Ranking** | **Medium** |
| **Problem: No Test management Tool ( SIT)** | |
| 1. There is no test tool available to the QA Team for writing test scenarios and test cases. Currently, all the test cases and scenarios are written in a spreadsheet and stored in the SharePoint.  2. There are separate spreadsheets for test cases, traceability, execution plan, design plan and status, test data and many more. Maintaining and managing so many spreadsheets is error prone and a lot of quality time is being spent on a daily basis trying to update these spreadsheets. Also, it can become extremely difficult for a new person joining the team to be efficient within a short span of time as there is a steep learning curve involved in understanding so many test documents apart from the other project documents.  3. The Business may not have enough time to go each and every requirement in Excel and it may result in scope slippage.  4. It is extremely difficult to establish and accurately maintain traceability relationships from requirements to test cases, from test cases to defects and from defects to reports without a proper test tool. Also every Sprint some new requirements are added / updated to existing set of requirements, then in that case maintaining the traceability is a herculean task.  5. Automation cases are maintained in ALM for SIT team but Traceability w.r.t. manual cases are maintained in Excel sheet. | |
|
|
| **Recommendations:** | |
| 1. Team should maintain requirements traceability, test cases, and execution results in ALM so that there is no scope slippage and Test team can pull out the Test coverage report, traceability, execution and defects status with ease.  All other teams are maintaining traceability in ALM except SIT Team and Client is apprehensive about moving to ALM. | |

|  |  |
| --- | --- |
| **Observation #4** |  |
| **Risk Ranking** | **Medium** |
| **Problem: Only one Manual Test cycle per Sprint (SIT)** | |
| Currently the SIT Test Team runs all their tests in just one cycle. This means all the test cases for that build are executed only once in 4 week’s timeframe, even if some/many of the test cases are able to find a good number of defects. The one Cycle approach on executing the scripts in this project is ineffective as there is every possibility of failing to uncover additional defects within that build. | |
|
|
| **Recommendations:** |  |
| 1. Since the test execution for every iteration is usually for a period of 4 weeks it is recommended to run at least two test cycles instead of running all the tests in one Cycle.  The first cycle should focus only on all the high priority test cases. All the defects found in the first Cycle should be logged and fixed before the second test Cycle begins. The second test Cycle should re-run all the failed high priority tests and log any additional defects or return any defect fixes from Cycle 1 that were not fixed correctly. Then the tester should move on to the medium and lower priority tests. | |

|  |  |
| --- | --- |
| **Observation #5** |  |
| **Risk Ranking** | **Medium** |
| **Problem: Require frequent Client Interaction** | |
| Daily Scrum meetings are conducted for all the teams but they are limited to status updates. Teams are not having frequent discussions with Client. | |
|
|
| **Recommendations:** |  |
| Recommend frequent interaction with Client over Call or Video conferencing which will improve :   * Client confidence in the team * Understanding of Issues / requirements from Business perspective * Team will receive First-hand information about the requirements, programs, schedule, builds, priorities etc. | |

|  |  |
| --- | --- |
| **Observation #6** |  |
| **Risk Ranking** | **Medium** |
| **Problem: Inadequate Test Metrics** | |
| 1. The test metrics (test execution and defect metrics) used on this project are standard metrics that are being used across many other projects. However, none of these metrics actually depict the value of testing on this project. Reporting that 1500 test cases are executed or 200 defects are found does not tell the Business/Management on the quality of the product or the residual risks on the product. Are 1500 tests or 200 defects good, bad or expected? The metrics need to be more effective.  2. The team is unable to demonstrate how effective the test cases are in finding defects. | |
|
|
| **Recommendations:** |  |
| Team can capture below metrics in addition to the existing metrics :   * Defect rejection ratio * Test case optimization * Test coverage * Cycle Time * % Automation scripts and Manual scripts * % Scope Change ( User stories Planned Vs User stories actually delivered) * Scenario Slippage (***Number of scenarios added post analyst review/ Original number of scenarios identified and sent for review)*** | |

**CONCLUSIONS:**

To summarize some of the key areas for improvement are:

* Detailed program level plan and schedule should be maintained.
* Test team should be having a clearly defined scope for sprint and required information / knowledge to come up with SIT scenarios for impacted areas well before the start of the SIT Sprint.
* Automate the regression and smoke testing suite and increase the test coverage for the regression suite.
* Have builds on defect fixes delivered more frequently during the Test Execution phase.
* Run the smoke tests and regression tests more frequently.
* Use two Test Cycles instead of only one Test Cycle during Test Execution.
* Use Quality Center or any other similar test tools on the project for writing test cases and establishing traceability.
* Capture test metrics to display the functionalities and the risks that have been tested, the number of defects related to the functionalities and risks that have been identified and the overall quality of the product/ residual risks on the project.
* Capture metrics on Test Case Effectiveness to determine if the regression coverage is focusing on areas where the test cases are more effective in finding defects.