# SFT221 SCRUM Report and Reflections

This report should be completed in the class and submitted at the end of class. Late submissions cannot be accepted without prior approval of the instructor.

**Members Present**:

Humaira Shaikh

## Milestone 6 Tasks

This is the final milestone where you will run the acceptance tests and fix any remaining bugs found. In addition, you will produce a testing report which lists all the tests conducted, the results and whether the bugs were fixed, and the final test passed. You will also review the test matrix to ensure every test has been performed and passed. You can change the colour of the test in the matrix to show it was run and passed. At the end, all tests in the matrix should have been passed.

The final test report can be tabular like this:

|  |  |  |  |
| --- | --- | --- | --- |
| Function/acceptance/requirement | Test Run | Bugs Fixed | Passed |
| Distance | TF001 | Did not handle negative coordinates | 🗹 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Deliverables Due at end of Lab:**

* SCRUM Report and reflections

**Deliverables Due at 23:59 4 Days after Lab:**

* Execute acceptance tests(results in Jira), and debug.
* Updated function-test matrix stored to the repository.
* Final Testing report listing tests conducted, bugs fixed and the final test passed.

**Rubric**

|  |  |  |
| --- | --- | --- |
| Individual | Group Participation | 75% |
|  | Teamwork | 10% |
|  | SCRUM Report & reflections | 15% |
| Group | Updated test matrix | 20% |
|  | Final test report | 20% |
|  | Test Execution (performed, results recorded, issues created) | 10% |
|  | Debugging (Bugs fixed, documented, Jira updated) | 5% |
|  | Git Usage (used properly with good structure) | 5% |
|  | Jira Usage (creates issues, tracks progress) | 5% |
|  | Meets Deadlines | 5% |
|  | SCRUM Report & reflections | 30% |

**SCRUM Report**

**Summary of Tasks Completed or Delayed in the last week:**

Here you can list all of the tasks completed in the last week along with any tasks which could not be completed with a reason why they could not be completed.

|  |  |  |
| --- | --- | --- |
| **Member** | **Tasks Completed** | **Tasks Delayed/Blocked** |
| **Humaira Shaikh** | **Acceptance testing, test matrix, scrum , reflection, final report.** | **Everything completed.** |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

For every task delayed or blocked, describe the reason for the delay or block, how it impacts the project and the proposed solution or workaround**.**

|  |  |
| --- | --- |
| **Delayed or Blocked Task** | **Function implementation fix match with sample output.** |
| **Reason for delay or block** | **Compile error** |
| **Impact on Project** | **Delay on testing, can not complete the testing suite on function implementation** |
| **Solution or work-around** | **Testing on only existing functions that is working currently** |
|  |  |
| **Delayed or Blocked Task** |  |
| **Reason for delay or block** |  |
| **Impact on Project** |  |
| **Solution or work-around** |  |

**Summary of Meeting:**

A summary of the main points discusses in the meeting and the outcomes of the discussions.

|  |  |  |
| --- | --- | --- |
| Topic | Discussion Summary | Outcome |
| Function implementation | **Fix function implementation for complete project.** | **complete, Function implementation WIP** |
| Acceptance testing | **Finished and reported during MS05** | **Executed** |
| SCRUM | **SCRUM Done** | **completed** |
| Reflection | **Done** | **completed** |
|  |  |  |
|  |  |  |
|  |  |  |

**Tasks Attempted During Meeting:**

Each member is assumed to participate in the SCRUM meeting and contribute to the completion of the SCRUM report and reflections. Since the SCRUM meeting will not take more than 20-30 minutes, there is lots of time left to undertake some of the actual work tasks. In the table below, each member should list what they did to complete the SCRUM report, the reflections, and 1-4 other tasks they completed during the class period. If a task could not be completed, the student should indicate why this was not possible.

|  |  |  |  |
| --- | --- | --- | --- |
| Member | Task Attempted | Time Spent | Complete? |
| Humaira | **Scrum report** | **30min** | **Yes** |
| Humaira | **Jira and Github Project page updated and assigned** | **30min** | **Yes** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**SCRUM Tasks Selected for Next Week**:

The tasks each member has selected to pursue for this class or the next week.

|  |  |
| --- | --- |
| Group Member | Task Description |
| Humaira | Last milestone - continue improving this program since it is currently incomplete. |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

**Major Outcomes of Meeting:**

This is where you should highlight the major accomplishments of the class.

|  |  |
| --- | --- |
| Outcome | Impact on Project |
| Confirmation on Acceptance testing | **Executing program individually (Alpha and beta) and listed out Debugs for matrix** |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

**Reflections**:

1. Although we wrote a report on the testing that shows which tests were run and passed or failed, we also updated the function test matrix. What are the advantages of updating the function test matrix in addition to writing the test report?  
     
   Updating the function test matrix alongside the test report offers a range of advantages that bolster a comprehensive and systematic approach to software testing and quality assurance. The matrix serves as a clear and visual record of the functions or components that have undergone testing, promoting thorough coverage across the software. This traceability fosters accountability by linking specific test cases to corresponding code segments. The process of updating the matrix encourages thoughtful test planning, enabling efficient allocation of resources and prioritization of critical tests. Furthermore, it aids in risk management by highlighting areas with limited test coverage, thereby identifying potential vulnerabilities. The matrix's role in assessing testing comprehensiveness provides valuable insights into software quality, aiding decision-making regarding its release readiness. In the context of evolving software, an up-to-date matrix streamlines regression testing by pinpointing necessary test reruns following code changes. Its presence as a shared reference point facilitates collaboration and communication among development, testing, and management teams. Regular updates also support continuous improvement efforts by identifying trends and areas for testing process enhancement.
2. Did you end up testing the code to the point where you were convinced it worked correctly? Were there any tests that had not passed at the end? If so, what was the impact of this on the project?

The acceptance testing phase revealed that not all tests had passed, indicating that the code did not meet the expected level of functionality and quality. This situation had a notable impact on the project, highlighting potential issues and shortcomings that needed to be addressed. The tests that did not pass exposed areas where the code failed to align with the project's requirements and specifications. This posed challenges in terms of meeting user expectations and delivering a reliable product.

The occurrence of failed acceptance tests prompted a reflection on various aspects of the project. It raised questions about the thoroughness of the initial testing strategy, the accuracy of the requirements gathering process, and the effectiveness of the development approach. The failed tests prompted a reevaluation of the codebase to identify root causes and areas requiring further development or bug fixes. I had to invest additional time and effort to diagnose and fix these issues, which in turn impacted project timelines, resource allocation, and overall project delivery.