# 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.

**GROUP**: \_\_\_\_\_\_\_\_\_\_\_\_\_\_4\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Members Present**:

|  |  |
| --- | --- |
| 1. Anna Francesca Dela Cruz (Cesca) | 4. In Tae Chung |
| 2. Irish Banga | 5. |
| 3. Gulpreet Kaur | 6. |

## 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 | þ |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**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** |
| **1 (Cesca)** | * SCRUM report details * Reflection Question #2 | N/A |
| **2 (Irish)** | * Test Report |  |
| **3 (Gulpreet)** | * Reflection Question # |  |
| **4 (In Tae)** | * Reflection Question # |  |

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** | N/A |
| **Reason for delay or block** |  |
| **Impact on Project** |  |
| **Solution or work-around** |  |
|  |  |
| **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 |
| Final test Report | Discussed the components and format of the final test report; Went over each required section and reviewed how they should be written | Each member has a clear understanding of the requirements and organization of the final test report |
| Remaining Bugs | Discussed remaining bugs in program; There one bug present. | Although the instructor allowed us to ignore it, we decided address it to ensure the project is completed at the highest standard possible |
| Division of tasks | Divided tasks to complete this week’s milestone: Irish completes test report, and SCRUM tasks are divided by the rest of the group | Every member is aware and held accountable for their responsibilities this week |
| Finalization of project | Discussed final tasks to finish/deliver the project: review code once more, review documents, complete JIRA logs | The entire project is ensured to have met all requirements and deliverables |

**Summary of Decisions Made:**

This will include major architecture and design decisions, testing decisions, prioritization of tasks, dealing with problems encountered and other major outcomes from the meeting.

|  |  |
| --- | --- |
| Decision | Rationale |
| Addressing the “bug” | Although the instructor allowed us to ignore it, we decided address it to ensure the project is completed at the highest standard possible. Furthermore, because most of the project is completed, we had enough time before the project deadline to do so. |
| Test names in tabular | For the final test report tabular, we decided to use the names of the test classes, not every test case name so that it mirrors the naming conventions of the function-test-matrix. This ensures consistency. |
| Division of tasks | To ensure all deliverables for this final milestone are met, all tasks were divided among the team |
| Increase JIRA usage | Because we lost marks in the previous milestone due to inadequate JIRA usage, it is our goal to be more active on the platform |

**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? |
| 1, 2, 3, 4 | Review requirements and format of final test report | 20min | YES |
| 1, 2, 3, 4 | Assign individual tasks for final milestone | 15min | YES |
| 1, 2, 3, 4 | Discuss remaining “bug” in program | 10min | YES |
| 1, 2, 3, 4 | Discuss finalization of project such as reviewing/revising documentation | 10min | YES |

**~~SCRUM Tasks Selected for Next Week:~~ (N/A)**

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

|  |  |
| --- | --- |
| Group Member | Task Description |
| N/A | N/A |

**Major Outcomes of Meeting:**

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

|  |  |
| --- | --- |
| Outcome | Impact on Project |
| Awareness of every member’s role and responsibilities | This ensures accountability among every team member, leading to the success of the project’s delivery. Furthermore, it would help smoothen the team’s workflow and prevent any conflicts with other members' tasks. |
| Time / resource prioritization | Having a shared understanding of the remaining tasks helped us determine the appropriate steps and time needed to address them. This helped us prioritize our time and resources appropriately, ensuring all deliverables and all aspects of the project was completed in a timely manner. |
| Addressing bugs | Addressing remaining bugs ensures our program functions at the highest possible quality. Developing the code to the very best of our abilities increases the likelihood of client/instructor satisfaction. |
|  |  |
|  |  |
|  |  |
|  |  |

**Things That Went Well in This Meeting:**

Here you can highlight things which worked well. This indicates that the way you worked on these items is working and should be continued.

|  |  |
| --- | --- |
| Topic/Work Item | Reason for Success |
| Division of tasks | Because most of the project was complete, dividing the remaining tasks was quick and simple; Everyone was able to complete their assigned tasks. |
| Communication of software quality | After discussing the program’s current state, all members understood what issues needed to be addressed. |
| Personal Reflections | Having our final meeting led to team members reflecting on the outcomes of the project, the team dynamic throughout it, and the overall experience working as a team. This allowed us to learn and acknowledge what we can do/change/bring in future projects to ensure smooth collaboration and a healthy group dynamic. |

**Things That Did NOT go Well in This Meeting:**

This is where you can list things which did not go well in the class. You should analyze why this happened and suggest how you can improve it next time. This will lead to the goal of *continuous process improvement*.

|  |  |
| --- | --- |
| Topic/Work Item | Reason for Problem and How to do Better |
| N/A |  |
|  |  |

**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?  
     
   Answer. Updating the function test matrix in tandem with the test report provides significant benefits to software testing. The function test matrix functions as a map, indicating which components of the software were tested and how well they performed. It allows us to identify at a glance which parts are performing well, and which may require additional attention. We can instantly identify any missing or forgotten tests by comparing this map to the complete test report. This is like double-checking to ensure that we did not overlook a vital test.

By maintaining this map up to current, we can be proactive about retesting when the program changes. If the program is upgraded, we merely need to retest the impacted sections. This saves time and makes testing more efficient. The map is also useful for showing others, such as software developers or managers, how much testing we completed and how well the software is performing. It's like a clear image that tells everyone about our testing adventure. In a nutshell, updating the function test matrix is like having a guide that helps us understand the big picture of testing, identify gaps, and ensure our product is strong and trustworthy.

1. Teamwork on a project like this is vital to success. How well did your team work? If it worked well, what contributed to its success? If it did not work well, what contributed to the problems?  
     
   Overall, our team worked very well. Acknowledging and respecting each member’s unique strengths and weaknesses contributed to our success. For example, some members were more skilled in software development, while others were better at testing and documentation so we divided those roles and tasks accordingly.

Furthermore, every team member had a strong sense of responsibility and contributed to the project fairly. Each member consistently attended all meetings and fulfilled their assigned tasks on time. The establishment of such a high standard within the team right from the very first milestone served as a driving force for us to uphold this level of dedication and performance throughout the entirety of the project timeline.

Finally, effective communication played a pivotal role in the project's success. We maintained a consistent flow of information, whether through our comments on JIRA or discussions within our private group chat on Microsoft Teams. This approach to communication ensured that all team members remained well-informed at all times. This helped us prevent conflicts, but also enabled us to quickly address any questions or concerns individuals may have had.

In summary,  by assuming responsibility, playing to our strengths, and keeping open lines of communication, we were able to complete the project successfully and reach our goals.

1. In every milestone you were asked what worked and did not work along the way. Were you able to incorporate what you learned to improving your team’s performance on the next milestone? Did your team learn from its mistakes and improve? If so, why? If not, why?  
     
   Answer. We've been examining what worked and what didn't in each phase as we go. By taking new steps, we can improve as a result. In the coming steps, we've been utilizing the lessons we've learned from both successful and unsuccessful endeavors to improve our team's performance.

Our team has done a terrific job of improving and learning from its missteps. We take this action because we're willing to test new approaches and figure things out. When things don't go according to plan, we investigate the situation to determine why and devise solutions for the future. It's like using our mistakes to improve our methods.

However, it's crucial to understand that not all changes take place immediately. To get some things correctly, extra time is needed. And on occasion, factors beyond our control may have an impact on how effectively we advance. Overall, we are making a lot of effort to improve upon our past missteps. We make advantage of what we already know to keep improving.

1. 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?

Answer. Yes, we did make sure to thoroughly test the code to ensure that it was functioning properly. To verify the code's functionality and speed, we ran a variety of tests. Before we declared the code to be complete, these tests helped us identify and address any issues.

We discovered a few tests that didn't function as intended during the testing. Not every code is flawless straight away, which is normal. We investigated the situation and spoke with the code creators when these tests failed. We collaborated to identify and address the problems.

Sorting things out took a little longer when tests failed. The project occasionally took a little longer because of this. However, it is preferable to identify and address these issues before the project is complete. It indicates that the outcome is more reliable and effective. Thus, even if there were a few minor delays, it ultimately improved the project.