# 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. All students are expected to attend the in-class SCRUM meetings and to participate. Failure to do so will result in greatly reduced grades.

**GROUP**: \_\_\_\_\_\_\_\_\_\_\_Group 4\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Members Present**:

|  |  |
| --- | --- |
| 1. Jo Eric - 137057188 | 4. Peralta Joe - 114751209 |
| 2. Nguyen Huu Linh - 118197227 | 5. Dominguez Daniel - 110835188 |
| 3. Jang Hyeri - 115328221 | 6. Sadat Morsal - 102693215 |

## Milestone 4 Tasks

**Deliverables Due at end of Lab:**

* Completed SCRUM report and reflections

**Deliverables Due at 23:59 6 Days after Lab:**

* Implemented Functions
* Implemented blackbox tests (store in repo), executed (results in Jira and on corresponding test documents) and debugged,
* whitebox tests written and stored in repository.
* whitebox tests implemented (store in repo), executed (results in Jira and on corresponding test documents) and debugged.
* Updated function-test matrix stored in the repository.
* Completed hook for test automation

**Rubric**

|  |  |  |
| --- | --- | --- |
| Individual | Group Participation | 75% |
| Teamwork | 5% |
| SCRUM Report | 10% |
| Automation Hook | 10% |
| Group | Implemented Functions (well-designed, written and documented) | 20% |
| Whitebox tests (well-designed, written and documented) | 20% |
| Test Execution (performed, results recorded, issues created) | 20% |
| 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 and Reflections | 20% |

**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** |
| **Eric Jo** | * Filling scrum Report. * Tasks Assignment * Created unit test code * Test Implementation | **There are no tasks delayed or blocked** |
| **Joe Peralta** | * Created function specifications. * Created and implemented functions. * Helped in answering questions in scrum report. * Created and implemented unit tests. | **There are no tasks delayed or blocked** |
| **Nguyen Huu Linh** | * Filling scrum Report. * Function specification. * Black-box document. * Helping to implement functions. | **There are no tasks delayed or blocked** |
| **Daniel Dominguez** | * Created and updated function-test matrix * Created `Business Requirements` document * Helped complete scrum report reflection * Created test cases | **There are no tasks delayed or blocked** |
| **Hyeri Jang** | * Created the test data for blackbox testing and documented them * Created the test description * Filling scrum report | **There are no tasks delayed or blocked** |
|  |  | **There are no tasks delayed or blocked** |
|  |  | **There are no tasks delayed or blocked** |

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** | **There are no tasks delayed or blocked** |
| **Reason for delay or block** | **There are no tasks delayed or blocked** |
| **Impact on Project** | **There are no tasks delayed or blocked** |
| **Solution or work-around** | **There are no tasks delayed or blocked** |
|  |  |
| **Delayed or Blocked Task** | **There are no tasks delayed or blocked** |
| **Reason for delay or block** | **There are no tasks delayed or blocked** |
| **Impact on Project** | **There are no tasks delayed or blocked** |
| **Solution or work-around** | **There are no tasks delayed or blocked** |

**Summary of Meeting:**

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

|  |  |  |
| --- | --- | --- |
| Topic | Discussion Summary | Outcome |
| Task Distribution | **White box testing deliverable was split to implementation and debugging.** | **Each deliverable except for Automation was assigned to each member.** |
| Understanding of assigned task | **Each member must understand the requirements needed to successfully complete their assigned tasks.** | **Each member has reviewed the necessary requirements to complete their assigned task.** |
| Allocation of tasks | **Each member has been asked to meet and discuss the deliverables or tasks needed to complete the milestone 4** | **Each member participated and collaborated in allocating and determining the tasks needed to complete milestone 4.** |
| Jira bug reports | **Members who work on tests implementation will write the bug reports on Jira** | **We made the task more organized.** |
|  |  |  |
|  |  |  |
|  |  |  |

**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 |
| Improving Blackbox test cases | Linh is assigned to improving blackbox test cases to increase certainty of the test |
| Detail-oriented approach | If someone found a detail that other members missed, he or she should notify them of it. |
| Determine design flaws | To ensure that software functionality runs correctly and is free from errors and bugs. |
| Tests must be conducted | To ensure the credibility of each function, various testing must be implemented and conducted. |
| Track all discovered bugs on Jira | Provides the testers an additional form of accessing and tracking new/existing bugs found in tests. |
| Improved test data | Make sure that all test data are written correctly and cover all the aspects of user validation. |
|  |  |

**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? |
| Nguyen Huu Linh | **Filling Reflection 3 and 4**  **Filling “Things That Went Well in This Meeting:”**  **“Things That Did NOT go Well in This Meeting”** | **25m** | **yes** |
| Hyeri Jang | **Completed reflection question #1** | **15m** | **yes** |
| Daniel Dominguez | **Completed reflection question #2, partially #4, overall scrum report refinement** | **20m** | **yes** |
| Eric Jo | **Assigned Tasks to each member** **Decision making**  **Summary of meeting** | **15m** | **yes** |
| Joe Peralta | **Filling the scrum report questions** | **30m** | **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 |
| Joe Peralta | Implement functions and debug each function that has failed the tests. |
| Hyeri Jang | Write whitebox test data and blackbox test data |
| Daniel Dominguez | Updated function-test matrix stored in the repository. |
| Nguyen Huu Linh | Implemented Blacbox tests, executed and debugged. White box tests written. |
| Eric Jo | Whitebox test implementation and Jira bug report |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

**Major Outcomes of Meeting:**

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

|  |  |
| --- | --- |
| Outcome | Impact on Project |
| Members have agreed upon collaboration | **Tasks have been defined and assigned to each member. Tasks are organized, structured, and well distributed for each member.** |
| All members have participated well | **Reduces the amount of time planning the requirements that needed to be done to successfully complete milestone 4.** |
| Scrum Report Completed | **We completed Scrum Report for this week providing feedback on time management.** |
| Tasks Assigned | **The tasks that were created have been assigned to each member fairly and correctly.** |
|  |  |
|  |  |
|  |  |

**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 |
| Team Collaboration and Communication | **Everyone actively participated, shared their ideas, and respectfully listened to others. This open and constructive communication allowed us to cover various topics effectively and make progress on our objective** |
| Task Assignment | **Each team member was assigned tasks and actively engaged in discussions to understand their assignments.** |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

**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 |
| Time Management and Meeting agenda Allocation | **During the meeting, one area that did not go as smoothly as expected was time management. We noticed that some discussions took longer than planned, which resulted in less time for other important agenda items. This could have been due to the complexity of certain topics and the enthusiasm to dive deep into them.** |
| Improving Meeting Effectiveness Through Clear Objectives and Agenda Setting | **During the meeting, things did not go as smoothly as we hoped. It was the lack of clear objectives and agenda at the beginning. To improve this our future meetings, we can implement a well-defined agenda and set clear objectives before the meeting starts.** |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

**Reflections**:

1. After you run your blackbox and whitebox tests you are asked to record the results in both the original test document as well as in Jira. Explain why it is a good idea to record the results in both places.  
   Recording the results of blackbox and whitebox tests in both the original test document and in Jira is beneficial for various reasons. It ensures that the testing process is well-documented and transparent, allowing team members to understand the conducted tests and their outcomes. The test document allows the team to track testing progress over time. Also, by reporting and documenting bugs in Jira, it becomes easier to track and prioritize them effectively. Jira’s organized bug tracking system helps to quickly resolve issues, making the development process more efficient and dependable. The combination of these practices fosters better collaboration, improved software quality, and streamlined workflows throughout the entire development lifecycle.

1. Why did we wait until the fourth milestone to write the whitebox tests?

Whitebox tests were delayed until the fourth milestone to prioritize black box testing, ensure that the core functionalities were implemented. Starting with blackbox testing provided quick feedback on user-central features, and it allowed the team to learn about potential functional and usability issues early on.

Delayed whitebox testing also helped conserve limited resources and facilitated the preparation for test automation which will be done now. By milestone 4, the code had been overall refined, making it more feasible to write whitebox tests without frequent code changes.

We also avoid creating inaccurate/poorly written whitebox tests by not starting on them prematurely, this increases the overall quality and comprehensiveness of our whitebox testing since we are making continuous improvements of test coverage and to the project as a whole. Since we will be introducing automated testing via hooks on git, this will improve the testing quality of life as it will remove the workload off our testers and allow them to focus on further improvements regarding the source code and other/existing tests.

1. For a given function did you produce more blackbox or whitebox tests? Explain why your answer (more blackbox or more whitebox) happens for most functions.

We produced more black-box tests for the given functions. The Reason for this is that black-box testing focuses on testing the functionality of the functions from the user’s perspective, without considering the internal implementation details. We designed black-box tests based on the function's specifications and expected behaviors, rather than the code’s structure or logic.

Black-box testing is ideal for testing functions because it allows us to validate their correctness based on their inputs and outputs, and it covers various scenarios and edge cases that the functions should handle. These tests are useful for ensuring that the functions meet their requirements and behave as expected regardless of how they are implemented internally.

White-box testing, on the other hand, requires knowledge of the code’s implementation details, such as specific paths, branches, and logic. For functions with complex internal structures or algorithms, white-box testing can be valuable to ensure full code coverage and to catch potential bugs related to specific implementation details. However, for simpler functions or function with well-defined behavior, black-box testing is often sufficient and more straightforward to design.

1. Explain the purpose of the automation hook for GIT and explain how it can improve the quality of the software in the project.

The automation hook for Git is a script or program that runs automatically at specific points during the Git workflow. It can improve software quality by enforcing coding standards, triggering automated testing, integrating with CI/CD pipelines, preventing sensitive data exposure, automating repetitive tasks, and maintaining accurate change logs. Overall, automation hooks enhance code consistency, correctness, and collaboration while streamlining the development process.

Enforcing Coding Standards: Automation hooks can enforce coding standards, enhancing code consistency and readability.

Automating Testing: They can trigger automated tests to catch bugs early and ensure new changes don’t break existing functionality.

Security Enhancements: Pre-commit hooks can prevent sensitive data leaks, ensuring security.

Documentation: They can automatically update project documentation, maintaining accuracy.