# 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**: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_5\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Members Present**:

|  |  |
| --- | --- |
| 1. Siripa Purinruk | 5. Kishan Dewasi |
| 2. Bussarin Apichitchon | 6. Dhrumit Ketan Parekh |
| 3. Seyed Iman Modarres Sadeghi | 7. Jaskaran Singh |
| 4. Farbod Maoyari | 8. Varshilkumar Ileshkumar Parikh |

## 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** |
| Siripa Purinruk | Traceability-matrix + Scrum report + Reflection question 1 + Function and test description documents | - |
| Bussarin Apichitchon | Blackbox testing | - |
| Seyed Iman Modarres Sadeghi | Function specification and implementation | - |
| Kishan Dewasi | Fixing returnDistance function. | - |
| Dhrumit Ketan Parekh | Function description documents. | - |
| Jaskaran Singh | Function specification and implementation | - |
| Farbod Maoyari | Function specification and implementation | - |
| Parikh Varshilkumar Ileshkumar | Test description documents | - |

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** | **-** |
| **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 |
| Assigning the tasks | Siripa Purinruk: responsible for scrum report + test automation + white box testing + function implementation | **-** |
|  | Bussarin Apichitchon: responsible for checking whitebox testcases, reflection question 1. | **Work is good and organized with no significant mistakes.** |
|  | Seyed Iman Modarres Sadeghi: responsible for function implementations. | **Fast work. However, there is a mistake hinders the testing process.** |
|  | Dewasi Kishan: responsible for reflection question | **Work is organized and completed punctually.** |
|  | Jaskaran Singh: responsible for function implementations. | **Good work. The work is progressing quickly.** |
|  | Dhrumit Ketan Parekh: responsible for reflection question | **Work is organized and completed punctually.** |
|  | Farbod Maoyari: responsible for function implementations. | **Good work. The work is progressing quickly.** |
|  | Parikh Varshilkumar Ileshkumar: responsible for Whitebox testing. | **Great effort. However, works are unorganized and have many mistakes, leading to delayed completion in MS4.** |

**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 |
| Assigning work | For efficient task distribution and time management. |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

**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? |
| Siripa Purinruk | Discussion + assigning work | 30 mins | **yes** |
| Bussarin Apichitchon | work assigned | 30 mins | **yes** |
| Seyed Iman Modarres Sadeghi | work assigned | 30 mins | **yes** |
| Dewasi Kishan | work assigned | 30 mins | **yes** |
| Jaskaran Singh | work assigned | 30 mins | **yes** |
| Dhrumit Ketan Parekh | work assigned | 30 mins | **yes** |
| Farbod Maoyari | work assigned | 30 mins | **yes** |
| Varshilkumar Ileshkumar Parikh | work assigned | 30 mins | **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 |
| Project Manager | Will designate with the task in the next week. |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

**Major Outcomes of Meeting:**

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

|  |  |
| --- | --- |
| Outcome | Impact on Project |
| Work assigned | Group members can prepare according to their roles. |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

**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 |
| Good collaboration. | Works completed punctually. |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

**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 |
| Delayed response | **-** |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

**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.  
     
     
   The advantages of recording the results in both test documents and in Jira are as follows:
   1. Traceability: When findings are noted in the initial test document, it makes it easier to link test cases to their results, simplifying the monitoring of each test case's progress and spotting any problems that might emerge throughout the testing process.
   2. Integration with Test Management Tools (Jira): To keep track of their projects, many development teams use issue tracking and project management software like Jira. Jira's ability to record test results enables seamless integration with the broader development process, offering a comprehensive picture of the project's state, including testing development, defect monitoring, and defect eradication.
   3. Compliance and auditing: Keeping thorough test paperwork on hand is a legal necessity in various businesses. By documenting outcomes in two locations, you ensure you have the evidence needed for compliance and potential auditing needs.

1. Why did we wait until the fourth milestone to write the whitebox tests?  
     
     
   Focusing on blackbox testing before whitebox testing allows us to verify that the functions deliver the desired results and meet the required specifications from an external perspective. Blackbox testing checks the functionality without considering internal implementation details, making it easier to validate the correctness and effectiveness of the functions independently of their code structure. This approach ensures that the functions work as intended before we delve into internal code details during whitebox testing, helping us identify potential issues and refine the code with a clearer understanding of the overall functionality.
2. 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.   
     
   For most functions, our team have created more white-box tests than black-box tests. The reason is whitebox testing allows us to directly examine the internal logic and code paths of the function. This enables us to design test cases that specifically target different branches and conditions within the code, ensuring comprehensive coverage and better handling of edge cases and potential issues.
3. Explain the purpose of the automation hook for GIT and explain how it can improve the quality of the software in the project.

Git automation hooks are scripts that execute automatically on particular version control events, with the "pre-commit" hook being the most popular. They give developers the ability to adapt and automate a wide range of processes, such as testing, code quality checks, prohibiting the commit of sensitive data, enforcing commit message standards, branch protection, dependency management, and unique workflows.

By utilizing automation hooks, the software project can greatly increase the quality of its output by discovering problems earlier, ensuring code uniformity, cooperation, and optimizing the workflow, leading to a more effective and reliable project.