# **MILESTONE 4** -- SFT221 SCRUM Report and Reflection

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

**GROUP**: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_A\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Members Present**:

|  |  |
| --- | --- |
| 1. Heungbin Oh | 5. Hiu Fung Chan |
| 2. Tien Vu To | 6. Trung Kien Phan |
| 3. Arshia Keshavarz Motamedi | 7. Ying Cheung Ellis Fung |
| 4. Aditya Tambe |  |

## Milestone 4 Tasks

**Deliverables due 4 days after your lab day:**

* Finish implementing/coding the functions.
* Finish implementing/coding blackbox tests. Store in repo, executed, results in Jira (and on corresponding test documents, and debugged.
* A set of whitebox tests as test documents (in an Excel file) with test data for the functions you created. At least 4 sets of test data are required for each function. You must have test cases for at least 6 functions (including all your custom function). Stored in the repository.
* Whitebox tests implemented (in the C++ testing project), stored in repository, executed, results in Jira and on corresponding test documents, and debugged (at least 1 SET is required).
* Updated requirements traceability matrix stored in the repository.
* Completed hook file (for EACH team member) for test automation stored in the repository.
* Completed scrum report including reflection questions answered.

**Rubric:**

|  |  |  |
| --- | --- | --- |
| **Individual** | Group participation (includes GitHub commits and Jira usage) | 80% |
| Teamwork | 20% |
| **Group** | Implemented functions and main (well-designed, and documented) | 10% |
| Finish coding blackbox code (well-designed, written, and documented) | 5% |
| Whitebox test case document (well written, complete, good test data) | 10% |
| Whitebox test code (well designed and documented) | 20% |
| Updated requirements traceability matrix | 5% |
| Test execution (performed, results recorded, issues created) | 10% |
| Debugging (bugs fixed, documented, Jira updated) | 5% |
| Hook files | 10% |
| Git usage (used properly with good structure) | 5% |
| Jira usage (creates issues, tracks progress) | 10% |
| Scrum report & reflections | 10% |
| **Deadline** | 20% deduction for each day you are late |  |

**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** |
| Arshia Keshavarz Motamedi | **Worked on the function coding** | **-** |
| Heungbin Oh | **Worked on the function coding** | **-** |
| Tien Vu To | **Worked on the function coding** | **-** |
| Aditya Tambe | **Worked on the scrum report and assisted with one function coding** | **-** |
| Hiu Fung Chan | **Worked on the unit test coding for Blackbox cases and created Whitebox cases** | **-** |
| Trung Kien Phan | **Worked on the unit test coding for Blackbox cases and created the Whitebox cases** | **-** |
| Ying Cheung Ellis Fung | **Worked on the unit test coding for Blackbox cases and Whitebox cases implementation** | **-** |

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 |
| Whitebox cases studying for functions | It was distributed amongst each other based on the expertise each member has in. | Everyone raised the issue according to task pick by their expertise |
| Blackbox cases studying for function | It was distributed amongst each other based on the expertise each member has in. | Everyone raised the issue according to task pick by their expertise |
| Function Coding | It was distributed amongst each other based on the expertise each member has in. | Everyone raised the issue according to task pick by their expertise |
| Unit test coding for Blackbox cases | It was distributed amongst each other based on the expertise each member has in.. | Everyone raised the issue according to task pick by their expertise |
|  |  |  |
|  |  |  |
|  |  |  |

**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 |
| Studying the functions and creating the Whitebox cases | Task was executed as we mutually agreed and told the responsibilities of each task which was being held by all by themselves. |
| Studying the functions and working on unit test coding for Blackbox cases | Task was executed as we mutually agreed and told the responsibilities of each task which was being held by all by themselves. |
| Working on Function Coding | Task was executed as we mutually agreed and told the responsibilities of each task which was being held by all by themselves. |
|  |  |
|  |  |
|  |  |
|  |  |

**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? |
| Arshia Keshavarz Motamedi | Worked on the part of functions coding | 10hrs | Y |
| Heungbin Oh | Worked on the part of functions coding | 9.5hrs | Y |
| Tien Vu To | Worked on the part of functions coding | 10hrs | Y |
| Aditya Tambe | Worked on the scrum report and worked on one function coding | 4hrs | Y |
| Hiu Fung Chan | Worked on unit test coding for Blackbox cases and created the Whitebox cases | 11hrs | Y |
| Trung Kien Phan | Worked on unit test coding for Blackbox cases and created the Whitebox cases | 11hrs | Y |
| Ying Cheung Ellis Fung | Worked on the unit test coding for Blackbox cases and Whitebox cases implementation | 11hrs | Y |

**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 |
| Arshia Keshavarz Motamedi | Working on the Function Coding |
| Heungbin Oh | Working on the Function Coding |
| Tien Vu To | Working on the Function Coding |
| Aditya Tambe | Working on one function coding, reflection and scrum report |
| Hiu Fung Chan | Working on Blackbox and Whitebox test cases (Building and implementation of it) |
| Trung Kien Phan | Working on Blackbox and Whitebox test cases (Building and implementation of it) |
| Ying Cheung Ellis Fung | Working on Blackbox and Whitebox test cases (Building and implementation of it) |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

**Major Outcomes of Meeting:**

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

|  |  |
| --- | --- |
| Outcome | Impact on Project |
| As we mutually decided that we will raise the issue and assign to ourselves accordingly | This made it easier for all of us to focus on the task which was been issued by us and no need to of other to remind what is been assigned to each other |
| Holding and ensuring to fulfill the commitment | By doing so, no interference was encountered, as well task executed smoothly since all things went smoothly from test planning to execution to programmers to black box testing creators |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

**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 |
| Creating issues | We selected the topic we have expertise on. |
| Meeting the deadlines | As per agreement and topic chose by us, it became easy to deliver the topics on time by providing and submitting it on the time discussed and said in the meeting. |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

**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 |
| - | **-** |
| - | **-** |
| - | **-** |
| - | **-** |
| - | **-** |
| - | **-** |
| - | **-** |

**Reflections**:

Answer the following questions using your own words. Make sure that each answer comprises a minimum of 100 words.

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.  
     
     
   We feel like it is a great idea to record the results in both the places because:
2. Jira offers great traceability and has comprehensive documentation. We can easily assign someone or ask someone to follow up the test cases. We can easily see the history of the issues in Jira, such as the timestamp, tasks performed and who did it.
3. Storing the results in the original test document provides a clear view of each testing result, we can easily find out which test cases passed as well as failed with explanation.
4. Jira integration provides visibility to stakeholders, aligning testing outcomes with project management. This dual recording ensures redundancy and accessibility, enhancing transparency and collaboration.
5. Why did we wait until the fourth milestone to write the whitebox tests?  
     
   We need to wait for writing the Whitebox tests until forth milestone because writing for Whitebox tests is need a deep level of understanding the function logics and analyzing the source codes of the functions. Before the fourth milestone, we are working on the function design and workflow of the program, function specifications are pending to confirmed. In the fourth milestone, we have well documents for the functions design and function specifications, our teams are working on the function implementation. We can view the source code of the functions and start to discuss with development team members and start to write the Whitebox tests.
6. Pick one of the functions you created and list its name. For this function did you produce more blackbox or whitebox tests? Explain why your answer (more blackbox or more whitebox) happens for most functions.  
     
     
   For the checkInput() function, we produce more black box tests for this function and more Blackbox tests on most functions, it is more than the white box cases. Developing black box test cases need to know the input, output, and basic knowledge of the processing of the function. We selected test data for evaluating each parameter of the function to ensure the functions are working properly in normal or valid data. In addition, we also need to pass invalid data to the function to ensure the function can manage the invalid data and provide error message to the user without data corruption, unexpected error, or program/system crash. However, developing Whitebox test cases need to inspect and understand the source code, but we can evaluate the situations that rarely happen.
7. Explain the purpose of the automation hook for GIT and explain how it can improve the quality of the software in the project.

The purpose of automation hooks in Git is to execute the scripts to run the unit tests automatically. For pre-push git hook, the script will be executed before we execute the “git push” command to push the source code to the repository. If tests failed, Git would not push the source codes to the repository, we can ensure the working source codes or software in repository are well tested without known bugs or issues. The automation hook for Git can improve the quality of the software in the project, because it ensure any code changes meet the quality standards that set by the team and only allows the push if all tests pass to avoid time consuming for code reviews and ensure the code remains consistent over time.