# **MILESTONE 3** -- 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**: 5

**Members Present**:

|  |  |
| --- | --- |
| 1. Gabriel Yeung | 4.Sheng Chieh Lin |
| 2.Ali Riza Sevgili | 5.Sheida Hashem Dabbaghian |
| 3.Rong Gang Xu | 6. Hon Kit Mok |

## Milestone 3 Tasks

In this milestone you will create issues to design the functions, design all of the functions you need to complete the project and store the specifications in the repository. As soon as the specifications start to be produced, you can start to design the blackbox tests (what they test, how to perform them and test data). Once tests are written, they can be implemented and added to the repository and any team members not otherwise busy can start to implement the functions. You will also build a function-test matrix that shows the blackbox tests for each function. This will be maintained through the testing cycle as new tests are added.

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

* A set of AT LEAST 4 function specifications stored in the repository.
* A set of blackbox tests as test documents 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.
* Start writing blackbox test code (for the functions above) and store in repository (at least 1 is required for this milestone).
* Start implementing the functions and store them in repository (optional).
* A requirements traceability matrix added to the repository and shows the mapping between the requirements and test cases.
* Updated Jira project to show activities and progress.
* Completed scrum report including reflection questions answered.

**Rubric**

|  |  |  |
| --- | --- | --- |
| **Individual** | Group participation (includes GitHub commits and Jira usage) | 80% |
| Teamwork | 20% |
| **Group** | Function specifications (documented, complete, well-written, added to the project) | 10% |
| Blackbox test cases document (well-written, complete, good test data) | 15% |
| Blackbox test code (well-designed and documented) | 10% |
| Functions implementation (coded in the C project & well documented) | 10% |
| Requirements traceability matrix (complete, added to GitHub) | 10% |
| Git usage (used properly with good structure) | 5% |
| Jira usage (creates issues, tracks progress) | 10% |
| Scrum report & reflections | 20% |
| Meets deadlines | 10% |

**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** |
| **Gabriel** | * **Study the source code and share knowledge with teammates** * Manage git repository * Provide better way for teammates to write the scrum report * Study test plan * Write Data Structure |  |
| **Rong Gang Xu** | * **Study the source code** * Get familiar with data structure completed by teammate * Write the scrum report * Work on making test plan point 6 to 10 and review other parts of the test plan |  |
| **Hon Kit Mok** | * Study the source code * Discuss with teammate and prepare the draft of test plan point 1-5 * Prepare Week 2 scrum report and reflection |  |
| **Ali Riza Sevgili** | * Studying the source code and other related staff * Checking whole project file to utilize main concept about Milestone-2 * Discuss with team member about function and related content * Create Black Box coding module and contribute to team members * Discuss about duties' organizations among members. |  |
| **Sheng Chieh Lin** | * **Study the source code and share knowledge with teammates** * Update git and jira boards and share knowledge with team members. * Continuously adjusting scrum reports and updating test plans * Work on making test plan point 11 to 14 |  |
| **Sheida Hashem Dabbaghian** | * Update the Jira board * Do the scrum report * Do the first function implementation * Study the function specs |  |
|  |  |  |

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** | **0** |
| **Reason for delay or block** | **NA** |
| **Impact on Project** | **NA** |
| **Solution or work-around** | **NA** |
|  |  |
| **Delayed or Blocked Task** | **NA** |
| **Reason for delay or block** | **NA** |
| **Impact on Project** | **NA** |
| **Solution or work-around** | **NA** |

**Summary of Meeting:**

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

|  |  |  |
| --- | --- | --- |
| Topic | Discussion Summary | Outcome |
| Scrum reflection assignment | **PIC: Roger** | **Agree** |
| Bb test doc assignment | **PIC: Gabriel** | **Agree** |
| Function code assignment | **PIC: Sheida** | **Agree** |
| Function spec assignment | **PIC: Tony** | **Agree** |
| Bb test code assignment | **PIC: Ali** | **Agree** |
| Traceability matrix assignment | **PIC: Simon** | **Agree** |
|  |  |  |

**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 |
| Task assignment | According to members’ response time and their willingness |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

**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? |
| Gabriel Yeung | **Scrum report** | **10 min** | **Yes** |
| Gabriel Yeung | **Bb test doc** | **10 min** | **Yes** |
| Rong Gang Xu | **Scrum report** | **10 min** | **Yes** |
| Rong Gang Xu | **Reflection** | **15 min** | **Yes** |
| Hon Kit Mok | **Feasibility matrix** | **10 min** | **Yes** |
| Hon Kit Mok | **Scrum report** | **10 min** | **Yes** |
| Sheng Chieh LIn | **Function spec** | **10 min** | **Yes** |
| Sheng Chieh LIn | **Scrum report** | **10 min** | **Yes** |
| Sheida Hashem Dabbaghian | **Scrum Report** | **10 mins** | **Yes** |
| Sheida Hashem Dabbaghian | **Coding implementation** | **30 mins** | **Yes** |

|  |  |  |  |
| --- | --- | --- | --- |
| Ali Riza Sevgili | **Scrum Report** | **10 mins** | **Yes** |
| Ali Riza Sevgili | **Coding implementation for Black Box Testing** | **1.5 hours.** | **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 |
| Gabriel Yeung | Blackbox test implementation |
| Gabriel Yeung | Jira board management |
| Gabriel Yeung | Git management |
| Gabriel Yeung | Project management |
| Rong Gang Xu | Complete reflection |
| Rong Gang Xu | Update the assigned task in Jira board |
| Hon Kit Mok | Coming workflow of implementing the test case |
| Hon Kit Mok | Update the delegated task in Jira board |
| Sheng Chieh Lin | Complete the task for the document. |
| Sheng Chieh Lin | Update the delegated task in Jira board |
| Sheng Chieh Lin | Complete the file of the document. |
| Sheida Hashem Dabbaghian | Coding implementation |
| Ali Riza Sevgili | Complete reflection |
| Ali Riza Sevgili | Updating Jira statement |

**Major Outcomes of Meeting:**

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

|  |  |
| --- | --- |
| Outcome | Impact on Project |
| Task would not be assigned but picked by members | **Disagreement on task allocation.** |
| Function Specification must be finished earlier than other tasks | **All other tasks depend on function specification** |
| Blackbox test document must be finished earlier | **Blackbox test implementation is dependent on the document** |
|  |  |
|  |  |
|  |  |
|  |  |

**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 |
| Identify tasks for this week | **Instruction is clear by reading the rubric** |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

**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 |
| Task assignment | **Members have disagreement on task assignment. Instead of letting members to choose what they want to work on after meeting, we will sit down and discuss the task assignment during meeting.** |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

**Reflections**:

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

1. In this milestone, we write the blackbox tests but not the whitebox tests. Explain why we can write the blackbox tests but not the whitebox tests.   
     
   At this milestone where we write function specifications, the focus is on defining what each function is supposed to do, its parameters and type. This specification is written in textual format without any coding contents. This makes it possible to write black-box tests based on these specifications. Since black-box testing is a testing method where the tester examines the functionality of the software without looking at its internal code structure. White-box testing, on the other hand, requires a deeper understanding of the code's internal structure and logic. It involves inspecting the code paths, branches, and data flows within the function. This level of white box testing is more effective when the code is closer to completion, and it often involves techniques like code reviews, code coverage analysis, and unit testing. Therefore, at this stage, upon working on and completing the function specifications, we just start to code the implementation of functions, it is too early and not yet feasible to do the white box testing.
2. Explain why we need the function-test matrix and why it is important in a large project.

Large projects heavily rely on the Function-Test Feasibility Matrix, also called an RTM (Requirements Traceability Matrix). It is essential for software development and project management since it fulfills a number of important roles.

First, it makes it easier to track requirements precisely, guaranteeing that every requirement is strictly followed from the point of creation to the point of implementation and testing. It is assured by this approach that every criteria is not only met but also carefully verified.

It also ensures thorough test coverage by mapping requirements to matching test cases. In addition to helping to find testing gaps, this mapping enables a more concentrated effort on quality assurance.

By confirming that the project follows the specifications, the matrix also serves as a quality assurance tool, reducing the chances of expensive mistakes and repeating work.

Additionally, it is very important in risk and change management, assisting teams in tracking and evaluating the effects of requirement modifications while setting priorities for testing.

In addition, the matrix ensures transparency and alignment by promoting efficient communication and teamwork within multidisciplinary project teams.

Last, it helps with documentation by providing a general summary of the project's testing plan. This is very helpful for sharing knowledge among team members and proving adherence to rules and guidelines. To sum up, the Function-Test Feasibility Matrix is an essential tool for big projects due to its many benefits that include quality, efficiency, and risk reduction.

1. Other life cycle models left team members idle while waiting for parts of the project to be completed. Describe how an agile model, like the one we are using, avoids this problem and keeps the whole team busy all the time. Does this make managing the project simpler or more complex and why?

An agile model is a project management model that keeps everyone on the team busy. It breaks down the project into smaller pieces and works on them in iterative cycles. While some team members are working on one part of the project, other members can simultaneously work on different parts. This can prevent any downtime or idle periods waiting for parts of the project and can support other team members to be completed. Therefore, this project management model is more complex which requires continuous communication for feedback support, collaboration, adjustments, close monitoring of schedules, task distribution, and priorities a mong team members.