# 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**: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_02\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Members Present**:

|  |  |
| --- | --- |
| 1. Preet Chakrani | 4.Jeny Prakashbhai Rangani |
| 2. Yagnik Dhankara | 5. Sakshi Sakshi |
| 3.Niyatiben Narendarbhai Patel | 6. Harjovan Singh |

## 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** |
| **Jeny Rangani** | **Implement already given functions in the mapping.c file** | **N/A** |
| **Nityati Patel** | **Debug blackbox test code and update files in repository** | **N/A** |
| **Harjovan Singh** | **Write whitebox test code and store in repository** | **N/A** |
| **Preet Chakrani** | **Update function test matrix stored in the repository and complete hook for test automation** | **N/A** |
| **Yagnik Dhankara** | **Implement whitebox test code and debug it** | **N/A** |
| **Sakshi Sakshi** | **Complete the scrum report with reflection questions answered for milestone 4** | **N/A** |
|  |  |  |

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** | **N/A** |
| **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 |
| Understanding the requirements of the milestone | **Examining each milestone requirement and discussing important points and tasks and learning about each other’s strengths and weaknesses in terms of technical knowledge.** | Increased familiarity among teammates and a better understanding of the group project. |
| Assigning tasks | **Delegating each member, a specific task based on their interests and abilities** | **Proper distribution of workload** |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

**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 |
| we will proceed further to do the group work by thoroughly understanding the requirements of the test plan and milestone | -proceed step by step and collaborate with other team members  -follow the plan and help the other team member if they face any kind of problem. |
| Ensuring the timely completion of scrum report – Sakshi Sakshi | The decision was made collectively on the basis of mutual understanding and depending on the selected team member’s exceptional abilities |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

**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? |
| Niyatiben Patel, Sakshi Sakshi, Jeny Rangani, Preet Chakrani, Yagnik Dankhara, Harjovan Singh | **Deep analysis of the milestone requirements and each team member suggested how functions could be implemented and which test cases should be added.** | **1.5 hour** | Yes |
| Niyatiben Patel, Sakshi Sakshi, Jeny Rangani, Preet Chakrani, Yagnik Dankhara, Harjovan Singh | Completion of all information required to fill in tables in the scrum report and answer reflection questions. | **1 hour** | 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 |
| Niyatiben Patel (Technical lead), Sakshi Sakshi (Project Manager) | Technical lead and Project Manager will assign tasks next week after lecture depending on the milestone requirements |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

**Major Outcomes of Meeting:**

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

|  |  |
| --- | --- |
| Outcome | Impact on Project |
| Proper distribution of workload | **Timely submission of proper and well-thought-out milestone 4.** |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

**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 |
| Equal distribution and delegation of tasks required to be completed | Based on each team member’s abilities and interests and with mutual understanding. |
| Proper analysis of the milestone | Active participation in group discussions and proper communication among team members. |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

**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 | N/A |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

**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.  
     
   Keeping test results in both the original test document and Jira is a great idea for several reasons. First, it helps make sure that all the important information about the tests is well-documented and easy to find. The original test document contains details about the tests, while Jira is a helpful tool for managing projects and tracking issues. By having the results in both places, we can easily see how the testing process went and who was responsible for it. This is useful for checking that everything was done correctly and for keeping everyone accountable.

It also helps with teamwork and communication. When test results are in Jira, all team members can see them and work together to solve any problems that come up during testing. Overall, recording test results in both the original test document and Jira helps with documentation, accountability, teamwork, and analyzing past performance. It's a practical way to ensure a smooth testing process and improve the overall quality of the software.

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

In our project, we were required to delay writing whitebox tests until the fourth milestone. This decision must have been driven by several factors. Initially, we prioritized blackbox tests to validate the software's core functionalities and ensure it met the required criteria. Waiting for the fourth milestone allowed the codebase to stabilize, reducing the risk of test rework due to frequent changes. Moreover, we adopted a risk-based testing approach, addressing high-risk areas with blackbox tests first. After verifying critical parts, we introduced whitebox testing to gain deeper insights into lower-risk areas and improve overall code quality. Our iterative development process must have also influenced the decision, allowing us to continuously enhance the software with each iteration. Overall, the approach struck a balance between early functionality validation and detailed internal testing in later stages.

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.  
     
   For a given function, int checkSize(const double size), I found more test cases by using the black box techniques in this workshop code. However, I believe to adequately test a program’s code, a combination of both testing techniques is required. Because Blackbox testing mainly deals with feeding different inputs to a code and testing its functionality from a user’s perspective who has little or no understanding of how the software works. However, white box testing is more comprehensive and deals with testing how the code works in different scenarios while having a proper understanding of the internal structure of the program and trying to find the potential bugs in the program which might not be caught with black box testing only. However, black-box tests are generally easier and faster to develop than white-box tests as the tester needs to consider only the inputs and outputs of the code without requiring any knowledge of the actual logic and structure of the written code.
2. 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 helpful tool that makes things easier for developers working on software projects. It automates tasks by setting up actions to happen automatically when certain events occur in the GIT repository, such as adding or changing code. The main purpose of using this automation hook is to improve the software development process and make the software better. The hook automatically runs tests when new code is added or changed, which helps catch bugs early on and ensures that the code works correctly.

Another benefit is code quality checks. The hook examines the code for issues and ensures that it follows the correct format and standards. This helps keep the code clean and easy to understand. The automation hook also plays a role in the code review process. It helps ensure that changes to the code are properly reviewed before they are added to the project, which contributes to higher-quality code. Lastly, the hook helps with version control and auditing, making sure that the code is well-organized and maintained properly.

In summary, the automation hook for GIT is a valuable tool that simplifies the work of developers and enhances the quality of software. It automates testing, checks the code's quality, facilitates code reviews, automates builds and deployment, sends notifications, and assists with version control and auditing. By automating these tasks, the hook contributes to a smoother development process, faster feedback, and improved software reliability and performance.