# 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. 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 5 Tasks

In this milestone, you should write, implement, and execute integration tests. Integration tests test how multiple functions work together to complete a task. Depending on what is being tested, you might be able to write unit tests to do the testing and automatically compare the results. In other cases, you might need to manually check the output to check it. This will all be stated in the tests where it discusses how they should be run.

As you update the function-test matrix, you will need to add a very brief description for each integration test so the matrix will clearly show what the tests are testing. Acceptance tests will be tested against actual user requirements and will list all the tests for each requirement.

Acceptance tests are the final tests and are largely aimed at showing the customer that the correct output is produced for different inputs. This will largely require manual testing.

**Deliverables Due at end of Lab:**

* Completed SCRUM report and reflections

**Deliverables Due at 23:59 12 Days after Lab:**

* integration tests written and stored in repository,
* integration tests written (store in repo), executed (results in Jira and in test documents) and debugged.
* acceptance tests written and stored in repository.
* Updated function-integration-requirements-test matrix stored to the repository.

**Rubric**

|  |  |  |
| --- | --- | --- |
| Individual | Group Participation | 75% |
| Teamwork | 10% |
| SCRUM Report and reflections | 15% |
| Group | integration tests (well-designed, written and documented) | 20% |
| acceptance tests (well-designed, written and documented) | 20% |
| Test Execution (performed, results recorded, issues created) | 15% |
| Debugging (Bugs fixed, documented, Jira updated) | 5% |
| Function-test matrix 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 | Scrum report + test automation + white box testing + function implementation | **-** |
| Bussarin Apichitchon | Whitebox testing + reflection question 1 | **-** |
| Seyed Iman Modarres Sadeghi | Function implementations | **-** |
| Kishan Dewasi | Reflection question | **-** |
| Dhrumit Ketan Parekh | Function implementations | **-** |
| Jaskaran Singh | Reflection question | **-** |
| Farbod Maoyari | Function implementations | **-** |
| Parikh Varshilkumar Ileshkumar | Whitebox testing | **Delayed** |

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** | **Whitebox testing** |
| **Reason for delay or block** | **Due to the incomprehension of the code and wrong assumption.** |
| **Impact on Project** | **The team is not be able to complete the MS4 on time.** |
| **Solution or work-around** | **Next Milestone, assign the work to another member who has an ability to complete.** |
|  |  |
| **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+ traceability matrix. | **-** |
|  | Bussarin Apichitchon: responsible for traceability matrix. | **-** |
|  | Seyed Iman Modarres Sadeghi: responsible for integration testing. | **-** |
|  | Dewasi Kishan: responsible for adding some code. | **-** |
|  | Jaskaran Singh: responsible for responsible for adding some code. | **-** |
|  | Dhrumit Ketan Parekh: responsible for adding some code. | **-** |
|  | Farbod Maoyari: responsible for responsible for integration testing. | **-** |
|  | Parikh Varshilkumar Ileshkumar: responsible for traceability matrix. | **-** |

**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 | * Integration tests * Acceptance tests * Updated function-integration-requirements-test matrix |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

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

**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**:

1. At this point, you are using the GIT hook to automate testing. Have you found that any of the tests failed and prevented you from pushing your code to the repository? If so, how did you handle the situation?  
     
   Our team encountered no obstacles from GIT hooks as we made efforts to resolve test case issues before pushing the code. In case any prevention does arise, we commit to addressing the bugs before attempting to push the code again.
2. Explain why we are automating the testing process and what the advantages of this automation are.  
     
   We use GIT hook to automate testing to execute tests and verify the software's functionality.

So, there are several advantages as follows:

* + Time-saving and increased efficiency due to faster test execution.
  + Consistency in test execution, reducing human errors.
  + Broad test coverage, including edge cases and negative scenarios.
  + Effective regression testing to check for new issues after code changes.
  + Cost-effectiveness in the long run by saving time and effort during testing.

1. Did you find the integration and acceptance tests more difficult to write than the black box and white box tests? If so, why were they harder to write? Did you write more white box and black box tests or more integration and acceptance tests?  
     
   In my experience, I find writing black box and white box tests more challenging due to the need to cover all potential user inputs, ensuring the software can handle unexpected behavior effectively. However, once the unit test process is executed thoroughly, it is easier to write integration tests and acceptance tests in the subsequent stages.
2. Explain why it is necessary to write integration and acceptance tests given that all of the code has already passed black box and white box tests.

The reason is integration test can ensure that the interactions between each function work cohesively as a whole. In some cases, even if all individual functions pass black box and white box tests, there might be issues when they interact with each other. On the other hand, we also need the confirmation from the end-user perspectives. Therefore, acceptance tests are written to validate that the software meets the specified requirements and fulfills the users' needs.  
To conclude, both tests will ensure that our software behaves as expected and meets the overall project goals.