# **Milestone 4 Scrum Report**

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

**GROUP**: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_4\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Members Present**:

|  |  |
| --- | --- |
| 1. Kaitlyn Cassiela Marino | 4. Grace Gabrielle |
| 2. Nolan Grossi | 5. |
| 3. Bilal Umar | 6. |

## Milestone 4 Tasks

* 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 in the repository, ensuring it shows both passed (green) and failed (red) tests.
* 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) | 10% |
| Updated requirements traceability matrix | 10% |
| Test execution (performed, results recorded, issues created) | 5% |
| Debugging (bugs fixed, documented, Jira updated) | 5% |
| Hook files | 15% |
| Git usage (used properly with good structure) | 5% |
| Jira usage (creates issues, tracks progress) | 15% |
| 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** |
| **Kaitlyn** | **Second half of black box test cases in document, scrum report** |  |
| **Grace** | **Unit test project creation, 1 set of black box code implementation, Jira Project Update** |  |
| **Bilal** | **4 function specifications, first half of black box test cases in document** |  |
| **Nolan** | **Function implementation, requirements traceability matrix** |  |
|  |  |  |
|  |  |  |
|  |  |  |

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 |
| Finished code implementation | **We discussed which member would finish the code implementations** | **We decided that Bilal would finish what’s left of the code implementations** |
| Finished black box tests | **We discussed which member would finish the black box tests implementation** | **We decided that Bilal would finish the black box test cases for the rest of the functions to be tested** |
| White box test document | **We discussed which member/s would write white box test cases for at least 6 functions** | **We decided that Kaitlyn and Grace would divide the white box test cases in half. The first half goes to Kaitlyn and the other goes to Grace** |
| Implemented white box test set | **We discussed which member/s would implement at least 1 white box test set** | **We decided that Nolan would implement one white box test set** |
| Updated requirements traceability matrix | **We discussed which member will update the requirements traceability matrix based on results** | **We decided that Nolan would update the requirements traceability matrix** |
| Hook files | **We discussed how the hook files would be stored in the repository** | **Each member will store their hook files with their student ID into the repository** |
| Scrum report + Jira | **We discussed which members would write the scrum report and which member/s will update the Jira project** | **We decided that Kaitlyn will do the scrum report and Grace will do the Jira updates, both with feedback/input from other members** |

**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 |
| Finished code implementation | Bilal will finish the code implementation |
| Finished black box tests | Bilal will finish the rest of the black box test cases for the other functions |
| White box test document | Kaitlyn and Grace will work on the white box test cases for 6 functions |
| Implemented white box test set | Nolan will implement white box test cases for at least one function |
| Updated requirements traceability matrix | Nolan will update the requirements traceability matrix depending on the test results |
| Hook files | Each member will store their hook files in the repository |
| Scrum report + Jira | Kaitlyn will work on the scrum report and Grace will update Jira, both with feedback/modifications from other members |

**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? |
| Kaitlyn | **Started white box test cases, started scrum report** | **~1hr** | **Yes** |
| Nolan | **Finished first part of function implementations** | **~1hr** | **Yes** |
| Bilal | **Finishing function implementations/black box test cases** | **~1hr** | **Yes** |
| Grace | **Started white box test cases** | **~1hr** | **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 |
| Kaitlyn | Half of integration test cases document, some acceptance test cases, scrum report, assisting in implementation of integration/acceptance tests |
| Bilal | Finish implementing/debugging white box test cases, assisting in implementation of integration/acceptance tests |
| Grace | Implementation of integration tests, some acceptance test cases, Jira updates |
| Nolan | Other half of integration test cases document, updates on the traceability matrix |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

**Major Outcomes of Meeting:**

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

|  |  |
| --- | --- |
| Outcome | Impact on Project |
| Finished function implementations | **Successfully finished function implementations, allowing the start of white box test creation.** |
| Finished black box test cases implementation | **Successfully finished implementing black box test cases, providing additional testing coverage.** |
| Finished white box test cases for 6 functions | **Successfully added white box test cases for additional testing coverage.** |
| Finished white box test implementation (3 functions) | **Successfully implemented a set of white box test cases for three functions, showing test progress and results.** |
| Updated requirements traceability matrix | **Successfully updated the status of test traceability, showing progress in testing.** |
| Scrum report + Jira | **Successfully summarized all activities, decisions, outcomes of milestone 4. Updated Jira project with new issues.** |
|  |  |

**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 |
| Black box test implementation | **All black box test cases were successfully implemented and executed with results** |
| White box test cases creation | **All white box test cases were successfully created in the excel file** |
| Updated requirements traceability matrix | **The requirements traceability matrix has been updated with results from black box and white box test cases** |
| Scrum report + Jira update | **All members contributed to / gave feedback on the scrum report and Jira updates** |
|  |  |
|  |  |
|  |  |

**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 |
| Linking unit testing project to solution | **Some members could not link the unit test project to the main source code solution. Next week we will discuss how to fix this so all members can link the testing project to the solution and execute the tests themselves.** |
| Hook Files | **Since some members couldn’t link the test and source project, they were not able to upload their hook files. Next week we will discuss how to link both projects so the hook files can be created and posted on Git.** |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

**Reflections**:

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

1. How did analyzing the internal logic and structure of the code help you design effective white-box test cases?

Analyzing the internal logic of the code is necessary to start creating white box test cases because it allowed the testers to create more test scenarios to ensure that there is more code coverage than just relying on the black box test cases. Additionally, it allowed the testers to further understand the logic of the implemented functions as well as the logic of the already-given code in the mapping modules. Seeing the internal structure allowed us to test scenarios which are uncommon but can still occur, such as the edge case testing, error handling, and identifying possible bugs that might not be found through black box test cases. White box testing also ensures that the inner behavior of the code aligns with the expected functionality of the overall program, making the program more reliable and easily maintainable.

1. How did using automated unit testing tools simplify or enhance your testing process? Reflect on the advantages and potential limitations of automation compared to manual testing methods.

Using automated unit testing tools greatly simplified and enhanced our testing process because by using the hook files and TortoiseGit, we are able to push the source code immediately into the Git repository if all test cases are passed. This means that we do not need to open Visual Studio 2022 and click multiple buttons to check if all tests are passed. We can simply do testing from the file explorer. This gives us more time to focus and prioritize other tasks. However, some disadvantages of automated unit testing is that if we fail to pull the most recent version of the repository, source code, or testing files, we might accidentally use automated testing on the wrong code and therefore push an outdated version of the files that we needed to test. Fortunately, this can easily be remedied by making sure that our local repository is up to date before working on the issues.

1. How did you document and communicate the bugs you identified? Reflect on the importance of clear and detailed bug reports in ensuring that issues are effectively resolved by the development team.

As instructed in milestone 4, we documented the identified bugs by updating our black box and white box test cases documents. We tested the unit testing project on the source code and updated the test cases on whether or not the test passed or failed. After that, we took note of the bugs to be fixed and added them on Jira as bugs to be fixed in the next milestone. Additionally, we updated the requirements traceability matrix to mark which tests passed and which ones failed. Updating these files and documents of the current status of the test cases allow us to be efficient in debugging the source code for the next milestone and also helps us to prioritize fixing the bugs rather than unnecessarily checking each test status over and over again. In this way, we are able to properly document all test cases and give us more room for improvement on whether or not we should create more test cases for additional coverage.