# **Milestone 6 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**: **B**

**Members Present**:

|  |  |
| --- | --- |
| 1. Aung Moe Thwe | 4. Thiri Aung |
| 2. Jhonatan Lopez Olguin | 5. |
| 3. Kashish Verma | 6. |

## Milestone 6 Tasks

This is the final milestone where you will run the acceptance tests and fix any remaining bugs found. In addition, you will produce a testing report which lists all the tests conducted, the results and whether the bugs were fixed, and the final test passed. You will also review the test matrix to ensure every test has been performed and passed. You can change the colour of the test in the matrix to show it was run and passed. At the end, all tests in the matrix should have been passed.

The final test report can be tabular like this:

|  |  |  |  |
| --- | --- | --- | --- |
| Function/acceptance/requirement | Test Run | Bugs Fixed | Passed |
| Distance | TF001 | Did not handle negative coordinates | 🗹 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

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

* Final testing report listing tests conducted, bugs fixed, and the final tests passed.
* Execute acceptance tests (results in Jira), and debug.
* Updated requirements traceability matrix 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** | Complete solution code running and executing successfully | 15% |
| Test execution (performed, results recorded, issues created) | 10% |
| Updated requirements traceability matrix | 5% |
| Final test report | 30% |
| Debugging (bugs fixed, documented, Jira updated) | 5% |
| Git usage (used properly with good structure) | 5% |
| Jira usage (creates issues, tracks progress) | 15% |
| Scrum report & reflections | 15% |
| **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** |
| Kashish Verma | Test Report Finalization | **N/A** |
| Aung Moe Thwe | Source Code Implementation | **N/A** |
| Aung Moe Thwe | Complete Solution | **N/A** |
| Thiri | Debugging of the remaining bugs | **N/A** |
| Kashish Verma | Test Report Finalization | **N/A** |
| Aung Moe Thwe | Source Code Implementation | **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** | **N/A** |  |
| **Impact on Project** | **N/A** |  |
| **Solution or work-around** | **N/A** |  |
|  |  |  |
| **Delayed or Blocked Task** | **N/A** |  |
| **Reason for delay or block** | **N/A** |  |
| **Impact on Project** | **N/A** |  |
| **Solution or work-around** | **N/A** |  |

**Summary of Meeting:**

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

|  |  |  |
| --- | --- | --- |
| Topic | Discussion Summary | Outcome |
| Finding bugs | **How to start debugging** | **By entering the different inputs** |
| Fixing compile errors | **How to fix the errors and why are the occurring** | **By following the error messages and debug the code line by line** |
| Updating test cases | **How to make modifications according to the requirements** | **N/A** |
| Fixing bugs | **Discussed about the reason of the bugs and how can we fix them** | **N/A** |
|  |  |  |
|  |  |  |
|  |  |  |

**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 |
| Made Changes to the acceptance testing | Needed to fix this from the feedback we got during presentation |
| Made Changes to integration testing | Similar to acceptance. |
| Fixed and implemented integration testing | This was to integrate it into the program as a whole rather than a separate source file |
| Test Report updates | Fixed and combined all the test cases. |
|  |  |
|  |  |
|  |  |

**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? |
| Kashish Verma | **Final Text report** | **35** | **No. Needed more time** |
| Aung Moe Thwe | **Fixing the compile errors and integration test cases** | **30** | **Not all cases pass, but completed** |
| Thiri Aung | **Reflections and debugging** | **40** | **Yes** |
| Jhonathan Lopez Olguin | **Updating traceability matrix** | **30** | **Yes** |
| All | **Scrum Report** | **30** | **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 |
| Kashish Verma | Finish the final test report |
| Aung Moe Thwe | Fixing the compile errors and make sure that the test cases are working |
| Thiri Aung | Debugging and updating test cases |
| Jhonathan Lopez Olguin | Improvising the test cases from the user’s point of view |
| All | Scrum Report and Reflection |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

**Major Outcomes of Meeting:**

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

|  |  |
| --- | --- |
| Outcome | Impact on Project |
| Deadlines Update | **We can predict when we can finish the project.** |
| Finalized solution | **This makes it easier to focus on the end-user perspective side of the code. Especially the acceptance and fulfillment business requirements** |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

**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 |
| Fixed integration tests | **Worked together** |
| Fixed Acceptance tests | **Discussed in detail** |
| Implemented proper integration tests | **Debug and reflect** |
| Fixed Bugs | **Debug** |
|  |  |
|  |  |
|  |  |

**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 |
| Logic Error | **We had a mistake in the core logic of the program where it works for the routes but not to buildings itself. Everything else works just fine** |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

**Reflections**:

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

1. Although we wrote a report on the testing that shows which tests were run and passed or failed, we also updated the traceability matrix. What are the advantages of updating the traceability matrix in addition to writing the test report?  
     
   The advantages of updating this traceability matrix in addition to writing the test report is because it guarantees that all criteria have been verified and to give a clear overview of coverage as well as the traceability matrix connecting the requirements to test cases. Moreover, you can also make sure that all tests are in line with the project's goals, confirm that every requirement has been validated, and rapidly spot any testing gaps by updating it.  Furthermore, by giving stakeholders a thorough understanding of the testing scope and status, a well-maintained traceability matrix facilitates improved stakeholder communication and makes evaluating the project's overall quality and readiness simpler.
2. Teamwork on a project like this is vital to its success. How well did your team work together? If you worked well, what contributed to its success? If it did not work well, what contributed to the problems?  
     
   On this project, our team collaborated incredibly well which is one of the things that helped us succeed. A key component was effective communication, which was ensured by frequent jira updates and weekly meetings that kept everyone updated and keep track of tasks. Each member showed a great sense of accountability and respect for one another by being dedicated to their roles and responding well. We also made use of one another's advantages by delegating work according to individual specializations, which increased productivity and product quality. The group worked together to solve problems as they came up, creating a positive atmosphere that promoted innovation and problem-solving.
3. In every milestone you were asked what worked and did not work along the way. Were you able to incorporate what you learned to improve your team’s performance on the next milestone? Did your team learn from their mistakes and improve? If so, why? If not, why?  
     
     
   Yes we were. As we accomplished each milestone, reflecting on what we did and did not, we found out areas that we could have improved and tried to improve, making adjustments accordingly. We tried to implement more efficient task delegations based on strengths of each individuals. When mistakes were made, we looked through them as a team, making sure that everyone is on the same page and how to fix on them in the future. The process of learning and cooperating together fostered a sense of continuous achievement and improvement, allowing us to refine our approach after every milestone. Our ability to learn from each mistakes and adapt quickly was very important in enhancing our overall performance and achieving our goals of the project itself.
4. Did you end up testing the code to the point where you were convinced it worked correctly? Were there any tests that had not pass at the end? If so, what was the impact of this on the project?

We did as much tests as we could but as always, no programs can be entirely bug free. There are some bugs that did not pass, but most of it is due to misinterpretation of the logic of the main program since starting the project. The main impact is on the end-user side. When they try to put in a certain value, the program might crash. Although it does not affect the main features, they highlight potential vulnerabilities that could impact the experiences of the user in specific conditions. There are still further refinements and potential updates that we could still work on.