# **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**: \_\_\_\_\_\_\_\_5\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Members Present**:

|  |  |
| --- | --- |
| 1. Shuja Lashkari | 4. Cristian David Vargas Marin |
| 2. Phuong Bac Nguyen | 5. |
| 3. Maryam Jawed | 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.

**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 in the repository, ensuring it shows both passed (green) and failed (red) tests.
* 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 cases code (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** |
| Maryam | Final Test Report |  |
| Phuong Bac | Bug Found |  |
| Maryam | Execute acceptance tests (results in Jira), and debug. |  |
| Cristian | Scrum Report |  |
|  |  |  |
|  |  |  |
|  |  |  |

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 |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

**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 |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

**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? |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**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 |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

**Major Outcomes of Meeting:**

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

|  |  |
| --- | --- |
| Outcome | Impact on Project |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

**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**:

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

1. How did creating the Quality Assurance report help you summarize and communicate the overall testing outcomes? Reflect on how this document supports decision-making at the project closure stage.  
     
   Creating the Quality Assurance report was crucial for summarizing all testing activities and communicating the project's test results clearly. It allowed us to document the test categories (unit, whitebox, integration, acceptance), report bugs like CD-02, and provide evidence of fixes. This report helped the entire team understand the progress and ensured that every requirement was tested and passed. At the closure stage, this document supports decisions such as project approval, system readiness, and future improvement opportunities. It also demonstrates testing traceability and makes the project defendable if questions arise during review.
2. How did updating the traceability matrix ensure that all project requirements were adequately tested? Reflect on the role of the matrix in maintaining accountability and completeness in the testing process.  
     
   The traceability matrix was essential to confirm that all system requirements were covered with at least one corresponding test case. By updating the matrix during Milestone 6, we verified which requirements were tested using unit, integration, or acceptance tests. The matrix helped maintain accountability, especially when checking which requirements had passed or failed. It ensured nothing was left out and gave a clear visual summary of our testing progress. As a result, we could confidently say that 100% of the requirements were tested and passed by the end of the project.
3. How did the process of preparing the project closure report help you evaluate the overall success of the project? Reflect on how documenting achievements, challenges, and unresolved issues can guide future projects.  
     
   Preparing the closure report helped us evaluate the entire testing lifecycle and the system’s reliability. We were able to document all the bugs that were found and fixed (e.g., CD-02 in calculateDiversionPath), the testing coverage (61 tests across all types), and lessons learned. It helped us reflect on our strengths, like collaborative debugging and test planning, and areas that need improvement, such as edge case handling. This level of documentation is useful not just for this project but also for future projects, as it builds a testing culture focused on traceability, continuous improvement, and transparency.
4. During the project closure process, what lessons did you identify that could improve future software testing efforts? Reflect on how evaluating the testing and development cycle contributes to continuous improvement.  
     
   One key lesson we learned is to test edge cases earlier. The issue with adjacent points in CD-02 only appeared in Milestone 6, delaying full validation. Another lesson was how valuable early traceability tracking is; updating it late increased the workload. We also realized the benefit of logging bugs and resolutions in Jira from the beginning. Overall, this experience showed us that continuous testing and early documentation are critical. Evaluating our cycle helped us identify gaps and improve our strategies, which we’ll apply in future testing efforts for better quality and efficiency.