# **MILESTONE 2** -- SFT221 Scrum Report and Reflection

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

**GROUP: 2**

**Members Present**:

|  |  |
| --- | --- |
| 1.Johnny | 4.Theo |
| 2.Manny | 5.Renato |
| 3.Noah | 6. |

## Milestone 2 Tasks

Some of the software for the project has already been written for you and is available on Blackboard. You must use this in your project and every team should add it to the source code for their repository. Anything in the main function is simply for demonstration purposes and can be replaced. The software you are being given has not been tested and you will need to test it.

You need to study the problem and the code provided for you and then:

* Add any new data structures you will require This will require a thorough analysis of the problem and the existing software. This should be done by creating a new header file in the directory where the rest of the source code has been placed. You do not want to go back and modify it later if you can avoid it as it will slow the project.
* Create a test plan for the project by replacing the text in the supplied test plan template with your test plan.

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

* An analysis of the problem (no written artifacts produced).
* A series of data structures created as header files and stored in the repository.
* A test plan 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** | Data structures (complete, correct, and well-designed, & project updated) | 25% |
| Test plan (complete, well-written) | 25% |
| Git usage (used properly with good structure) | 10% |
| Jira usage (creates issues, tracks progress) | 20% |
| Scrum report & reflections | 20% |
| **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 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** |
| **Johnny** | **SCRUM, Contract** | **n/a** |
| **Manny** | **SCRUM, Contract** | **n/a** |
| **Noah** | **SCRUM, Contract** | **n/a** |
| **Theo** | **SCRUM, Contract** | **n/a** |
| **Renato** | **SCRUM, Contract** | **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 discussed in the meeting and the outcomes of the discussions.

|  |  |  |
| --- | --- | --- |
| Topic | Discussion Summary | Outcome |
| Scrum | **Scrum questions** | **Questions answered** |
| Test Plan | **Plan for testing** | **Created plan outline** |
| New .h File | **Create a header file ()** | **Created header file** |
| jira | **Getting used to program** | **complete** |
|  |  |  |
|  |  |  |
|  |  |  |

**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 Division | Equal division of work load to each member. |
| Testing Functions | Shortest Path Calculation, Capacity Calculation, Output Message Generation, Shipment Allocation |
| Testing Paths | Optimized for shortest path possible |
|  |  |
|  |  |
|  |  |
|  |  |

**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 cannot be completed, the student should indicate why this was not possible.

|  |  |  |  |
| --- | --- | --- | --- |
| Member | Task Attempted | Time Spent | Complete? |
| All | **Discussed problem and possible solutions/resolutions to the situation** | **30-45 min** | **Yes** |
| All | **Answering Scrum report** | **1 hr** | **Yes** |
| All | **Discussion and creating H file** | **1 hr** | **Yes** |
| All | **Discussion and creating Test Plan** | **1 hr** | **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 |
| All | Scrum/Reflection questions |
| All | Function specs |
| All | Blackbox testing |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

**Major Outcomes of Meeting:**

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

|  |  |
| --- | --- |
| Outcome | Impact on Project |
| H file created () | **Made for testing to locate/identify any possible bugs/errors** |
| Test planning created | **Set testing specifications as a group** |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

**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 |
| Scrum | **Questions answered through discussions** |
| Test planning | **Plan outline set** |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

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

**Reflection Questions:**

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

1. In this milestone you have been asked to analyze a problem and design software (functions) to complete the solution without writing the software.
   1. Is this process more difficult than just writing the software to complete the project? If so, why is it more difficult? If not, why is it easier than just writing the software?

We found that writing code is simpler and easier than creating functions that test code of an already made project.

Over looks and name errors are common as some functions or some variables made with the program could fall under different classes and accessing the data could be a struggle. As well understanding what each part of the code does, without truly understanding the reason behind all the variables/functions/classes its harder to make test cases and find an overall solution, but still possible.

* 1. Describe two advantages of developing software in this manner rather than just moving on to writing the functions without writing specifications first.
     1. To keep it simple, for both the code editors and those who examine the code. By understanding what the purpose of the code is you can determine the specifications required for the software to function efficiently.
     2. The process allows for better organization, so that a team can divide the work and each attack parts of the code and efficiently identify and solve any problems or errors early on.

1. Why is it a good idea to create a test plan? Describe at least 3 advantages of test plans.
   1. helps with identifying the possible errors or bugs that should be resolved, as well as specific scenarios that need to be tested. This ensures that the code should run smoothly and reduces the possibility of any bugs/errors/crashes that could occur.
   2. Test plans allow for all members involved to communicate and each share some possible tweaks or any extra testing or any missed areas, to overall bring out a better collaboration and a more efficient team and code
   3. It helps build an organized guideline for the testing, guiding the testers on the basic required aspects of testing.

1. Describe the process you used to analyze and understand the existing software.

Further reviewed header files, as well as configuration files, to grasp a better understanding of the issue at hand and locate and identify any issues or potentially weak code.

We collected data on the software and analyzed some of its issues and where it’s strengths and weaknesses were to undergo some improvements in the future.

Collaborated and worked as a group to discuss any extra information or points some other members had.