# **MILESTONE 2** -- 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.Thomas Lewis | 4.Kiki |
| 2.Asem M | 5. |
| 3.Hla Myint Myat | 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) | 20% |
| Test plan (complete, well-written) | 20% |
| Git usage (used properly with good structure) | 10% |
| Jira usage (creates issues, tracks progress) | 15% |
| Scrum report & reflections | 25% |
| Meets deadlines | 10% |

**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** |
| **Thomas Lewis** | **GitHub Contribution** | **N/A** |
| **Asem M** | **Jira** | **N/A** |
| **Hla Myint Myat** | **Scrum and Reflection** | **N/A** |
| **All** | **Group 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 |
| Create New Data Structures | **Create data structures to do the project** | **Created new data structures** |
| Create Testing Plan | **Create a testing plan and reviewing errors** | **Created testing plan** |
| Scrum Report and Reflections | **Before starting project, analyzing problem and design software** | **Learned the idea if software design** |
| TESTING, GITHUB,JIRA | **Updating Gits, Updating Jira Tasks Schadules** | **Completed** |
|  |  |  |
|  |  |  |
|  |  |  |

**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 |
| Breaking Down Tasks | Equal amount of works assigned to each member of team. |
| Testing Plan | Optimized Testing plans |
| Scrum | Analysis discussion of problems in program |
|  |  |
|  |  |
|  |  |
|  |  |

**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? |
| KiKi | **Creating Test Plan** | **1hr** | **Yes** |
| Thomas Lewis | **Creating New Data Structures** | **2hrs** | **Yes** |
| Asem M | **Testing , GH, Jira** | **1.5hrs** | **Yes** |
| Hla Myint Myat | **Scrum Report & Reflections** | **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 |
| Thomas Lewis | Git Control |
| Asem M | Jira Control |
| Kiki | Project Discussion |
| Hla Myint Myat | Scrum and Reflection |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

**Major Outcomes of Meeting:**

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

|  |  |
| --- | --- |
| Outcome | Impact on Project |
| Test Planning | **Testing specification has been set** |
| Data Structure | **Created Data Structures** |
| Updating | **Updated GitHub and Jira** |
| Scrum Report | **Completed Reflections** |
|  |  |
|  |  |
|  |  |

**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 |
| GitHub, Jira | **Useful for version control and keeping track of changes** |
| Meeting | **Contribution of all members** |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

**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?  
        
      Analyzing the problem and designing software functions without implementing the software can be more challenging than simply writing the software to complete the task.

There is a risk of missing important information if the software is not written. There is a risk of overlooking some complexity that may surface during development if the implementation procedure is skipped. Second, without the program, it is more difficult to understand the problem. It required abstract and intellectual thinking to tackle the problem and potential solutions.

* 1. Describe two advantages of developing software in this manner rather than just moving on to writing the functions without writing specifications first.  
       
     i) We can establish the exact needs and requirements for the software by first analyzing the problem and designing the software functionality. This ensures that team members have a common idea of what the program needs to do.  
       
     ii) The process of analyzing and designing software provides for better project management and organization. It helps in getting ahead of upcoming challenges, dependencies, and risks, allowing for more effective allocation of resources, scheduling, and task prioritization.

1. Why is it a good idea to create a test plan? Describe at least 3 advantages of test plans.  
     
   i) Effective Communication: Test plans represent a way for communication among the group members involved in testing. They improve communication and understanding among team members, such as developers, project managers, and clients.

ii) Comprehensive Coverage: Test plans allow in identifying and resolving of all functional requirements, features, and scenarios that must be tested. This ensures thorough testing and reduces the possibility of missing essential parts.  
  
iii) Structured approach: provides a clear and organized framework for testing. It enables testers to take a systematic approach to testing and verify that all relevant aspects are covered.

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

We reviewed the header and configuration files to better understand the implementation and find any potential spots for improvement or concerns.

We gathered software-related support. Analyzing reported difficulties gave useful information about the software's strengths, limitations, and places for improvement.

We maintained conversations with the group to clear up any misunderstandings and obtain more details.