# **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.Prabhjot Singh | 4. Siya Khanna |
| 2.Sampreet Klair | 5. Prince Prince |
| 3.Dhruv Kakadiya | 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 at End of Lab**

* Completed SCRUM report & reflections

**Deliverables Due within 48 hours of lab**

* 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.

**Rubric**

|  |  |  |
| --- | --- | --- |
| Individual | Group Participation | 75% |
| Teamwork | 10% |
| SCRUM Report | 15% |
| Group | Data structures (complete, correct and well-designed) | 20% |
| Test Plan (complete, well-written) | 20% |
| Git Usage (used properly with good structure) | 10% |
| Jira Usage (creates issues, tracks progress) | 10% |
| Meets Deadlines | 15% |
| SCRUM Report and Reflections | 25% |

**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** |
| **Prabhjot Singh** | **Problem Analysis, scrum report** | **-** |
| **Dhruv Kakadiya** | **Managing Jira and test cases** | **-** |
| **Sampreet Klair** | **Committing the changes to GitHub repo and test cases** | **-** |
| **Siya Khanna** | **Made the header files** | **-** |
| **Prince Prince** | **Making the Test cases** | **-** |
|  |  |  |
|  |  |  |

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 |
| 1. Problem Analysis | **Went through the project and made problem analysis** | **Problem Analysis Done** |
| 1. Header files | **Header files were made** | **Header files done** |
| 1. Test Cases | **Discussed about the test cases** | **Test cases were developed.** |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

**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 |
| Test Cases to be done | Success |
| Reflection | Success |
| Header Files making | Success |
|  |  |
|  |  |
|  |  |
|  |  |

**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? |
| Prabhjot Singh | **Scrum report** | **20 min** | **Yes** |
| Prabhjot Singh | **Reflection** | **15 min** | **Yes** |
| Siya Khanna | **Making header files** | **20 Min** | **Yes** |
| Prince Prince | **Test Cases** | **15 min** | **No** |
| Dhruv Kakadiya | **Test Cases and Jira management** | **15 min** | **Yes** |
| Sampreet Klair | **Test cases and git management** | **15 min** | **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 |
| Prabhjot Singh | MS3 analysis and allotting the work |
| Dhruv Kakadiya | Jira organization |
| Sampreet Klair | Git organization |
| Siya Khanna | Working on the code |
| Prince Prince | Working on the test cases |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

**Major Outcomes of Meeting:**

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

|  |  |
| --- | --- |
| Outcome | Impact on Project |
| Scrum report done | **success** |
| Reflection | **success** |
| Code management done | **Organized the code properly.** |
|  |  |
|  |  |
|  |  |
|  |  |

**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 |
| Everything went well as planned | **Team co-ordination** |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

**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 |
| nothing | **Team work** |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

**Reflections**:

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 feel like it is a little more complicated. Here is why,

* It is an essential part for the project because if things go wrong here it will make more complications.
* Design decisions made in this phase can have a significant impact on the quality, maintainability, and extensibility of the software in the long run.
* Identifying the issues early in the design phase is very difficult.
  1. Describe two advantages of developing software in this manner rather than just moving on to writing the functions without writing specifications first.
* Analysis and design help in clarifying requirements and system architecture, reducing any problems in the software development process.
* Well planned project makes the entire process easy and efficient.

1. Why is it a good idea to create a test plan? Describe at least 3 advantages of test plans.  
     
   The good test plans offers various advantages in overall quality and success of software project.

Here are some of the advantages of test plans:

1. A test plan outlines a systematic and structured approach to testing the software.
2. We can allocate the resources efficiently.
3. test plans serve as a valuable source of documentation and communication among project stakeholders, including developers, testers, project managers, and clients.
4. Describe the process you used to analyze and understand the existing software.
   1. We reviewed all the documentation associated with the software.
   2. We reviewed the code so that we can have a clear understanding of what exactly is happening in the code.
   3. Identify external dependencies and libraries used by the software if any.
   4. Discuss the software among the team members, testers, and checked the software as an end-user to have a clear idea.