# **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**: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_11\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Members Present**:

|  |  |
| --- | --- |
| 1.Richa Koirala | 4.Ammar |
| 2.Akansha | 5. |
| 3.Manjot Singh | 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** |
| **Akansha** | **Git control, Scrum Report, Jira** | **N/A** |
|  |  |  |
| **Richa** | **Git control, Scrum Report, Jira** | **N/A** |
|  |  |  |
| **Manjot** | **Git control, Scrum Report, Jira** | **N/A** |
|  |  |  |
| **Amaar** | **Git control, Scrum Report, Jira** | **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** |  |
| **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 discussed in the meeting and the outcomes of the discussions.

|  |  |  |
| --- | --- | --- |
| Topic | Discussion Summary | Outcome |
| GitHub | **Updating git branches** | **DONE** |
| Test plan | **Test plan done** | **DONE** |
| Jira | **Task schedule setup in Jira** | **DONE** |
| .h file | **Using header file done** | **DONE** |
| Scrum Report | **Discussing task completed, not completed and reflection questions** | **DONE** |
|  |  |  |
|  |  |  |

**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 |
| Testing decision | Using integration test cases, unit test cases, traceability matrix |
|  |  |
| Prioritization of tasks | Equal work to be divided among all the group member |
|  |  |
| Using new header file | Using header file to get desired output |
|  |  |
|  |  |

**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? |
| Akansha | **held the meeting and made data structures in header file, reflection questions** | **15 mins** | **Yes** |
|  |  |  |  |
| Ammar | **Worked with Jira** | **15 mins** | **Yes** |
|  |  |  |  |
| Richa | **Prepared SCRUM report and reflection questions** | **15 mins** | **Yes** |
|  |  |  |  |
| Manjot | **Worked for test plan** | **15 mins** | **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 |
|  |  |
|  |  |
| Akansha | Looking after GitHub |
|  |  |
| Richa | Matching the availability and get a time schedule for the meeting in person and the location for the meeting. |
|  |  |
| Manjot | Manjot will conduct the meeting with tasks for milestone 2 |
|  |  |
| Ammar | Looking after Jira |
|  |  |
|  |  |
|  |  |

**Major Outcomes of Meeting:**

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

|  |  |
| --- | --- |
| Outcome | Impact on Project |
| Test plan | **Testing algorithm was set according to the group member** |
|  |  |
| .H file | **Created .h file will help to get the desired output and identify any potential bugs and errors** |
|  |  |
| Attendance | **Everyone attended the meeting** |
|  |  |
|  |  |

**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 REPORT | **Everyone contributed to scrum report** |
| GIT | **Discussed more about git** |
| Jira | **Discussed more about Jira** |
|  |  |
|  |  |
|  |  |
|  |  |

**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?  
      Yes, this process is more difficult than just writing the software to complete the project. It is more difficult because there will be a high probability of missing important details. If immediate feedback is not provided by writing actual code, it can be hard to ensure that the design adequately captures all aspects of problem and solution.to design software functions, identification of risk and dependencies is important.
   2. Describe two advantages of developing software in this manner rather than just moving on to writing the functions without writing specifications first.

The two advantages of developing software in this manner rather than just moving on to writing the functions without writing specifications first are:

1. If time is taken to analyze the problem thoroughly and design software functions before writing code, a clear and deeper understanding of the problem domain can be gained.
2. If time is taken to analyze the problem thoroughly and design software functions before writing code, one can have better alignment with the stakeholder expectations and priorities. With this, developers can ensure that the proposed solution meets the needs and preferences of end users, customers, and other project stakeholders.

1. Why is it a good idea to create a test plan? Describe at least 3 advantages of test plans.  
   A test plan is blueprint for the testing process, providing guidance to testers for conducting testing activities. It is a good idea to create a test plan. Some of its advantages are:
2. A test plan provides a structured framework for conducting testing activities throughout the software development lifecycle.
3. A test helps in risk identification and mitigation of risk associated with software development and testing.
4. Prioritizing project, a test plan helps allocate resources effectively and it also prioritizes testing activities based which helps allocate resources.
5. Describe the process you used to analyze and understand the existing software.

Analyzing and understanding the existing software was very important. So, We first created header file, examined the header file and configuration files to understand the implementation details and identifying if it gives desired output and potential areas for improvement or issues.

We also analyzed software and pointed out the strengths, weakness, and field of improvement areas of the software.

We also conducted meeting to clarify doubts and collect more information and identify field of improvements.