# **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**: 6

**Members Present**:

|  |  |
| --- | --- |
| 1. Nevan Sargeant | 4. Cynthia Fotso |
| 2. Tsz Wa Wong (Locus) | 5. Shwe Yee Lin Aung |
| 3.Jackey Zhou | 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** |
| **Nevan** | **Jira, Test Plan** | **None** |
| **Cynthia** | **Reflection Q2, Test Plan** | **None** |
| **Jackey** | **Github, Reflection Q3, Test Plan** | **None** |
| **Locus** | **Fixing code, Test Plan** | **None** |
| **Shwe Yee** | **Scrum Report, Reflection Q1, Test Plan** | **None** |
|  |  |  |
|  |  |  |

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** | **Fixing the code** |
| **Reason for delay or block** | **We needed to understand how the code worked first before attempting to fix it** |
| **Impact on Project** | **There was some delay in completing the task** |
| **Solution or work-around** | **First, we tried to understand the code, then, fixing the code went smoothly accordingly.** |
|  |  |
| **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 |
| Task Allocation | **Tasks required to be completed are identified and allocated to each member according to availability. The tasks were allocated based in urgency and importance.** | **A general allocation of tasks for each member was completed by collaboration and discussion among the group about which person was allocated to which task.** |
| Documentation of tasks | **To keep a through documentation of task assignments, progress updates and any decisions made during the task allocation process.** | **It serves as a reference point and help maintain accountability.** |
| Feedback Mechanisms | **Establish mechanisms for providing feedback on task progress and addressing any challenges or obstacles that arise.** | **Regular check-ins or status updates helps keep tasks on track and identify issues early.** |
| Task Completion | **Discussion is done to have a general summary and confirmation of completed tasks by each group member.** | **Acknowledge the contributions of each team member to the project or initiative.** |
|  |  |  |
|  |  |  |
|  |  |  |

**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 |
| Everyone does the Task Plan | There are quite a number of questions to contribute to which is essential in the steps that are to be taken towards the completion of the milestone and hence everyone contributes. |
| Nevan contributes to Jira overall | He does this to ensure that all the tasks and communication being done/ taking place between the members are being well documented |
| Jackey updates Gitbub and does Reflection Question 2 | He does this to ensure that all the files required to be uploaded will be uploaded before the deadline and then, contributes by sharing his thoughts and ideas toward the completion of Reflection 2 |
| Cynthia does Reflection Question 3 | She does this by reflecting on the steps she took and the methods she used and the decisions she made during the analysis process and then identifying effective strategies and discarding ineffective ones |
| Locus fixes the code | He does this by correcting errors and defects in the code that may exist which will then ensure the stability of the code which will in turn enhance its performance |
| Shwe does the Scrum Report and reflection 1 | I did this to promote transparency by documenting the group’s progress and the type of communication and decision making that is occurring during group meetings and tasks. |
|  |  |

**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? |
| Jackey | **Github, Reflection Q2, Test Plan** | **45 mins** | **Yes** |
| Cynthia | **Reflection Q3, Test Plan** | **45 mins** | **Yes** |
| Nevan | **Jira, Test Plan** | **30 mins** | **Yes** |
| Locus | **Fixing the code, Test Plan** | **90 mins** | **Yes** |
| Shwe Yee | **Scrum Report, Reflection Q1, Test Plan** | **70 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 |
|  | Will be allocated at a later date during a group call |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

**Major Outcomes of Meeting:**

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

|  |  |
| --- | --- |
| Outcome | Impact on Project |
| Code was fixed according to the instructions given | **The overall process to finish the milestone using the fixed code in the upcoming week’s tasks are going smoothly** |
| Jira updates | **There is a documentation of who does what for each week tracked which makes it easier for the professor to see who made each contributions** |
| Github | **The files completed by each group members are uploaded for easier access by the professors and easier collaboration of the team going forward** |
| Scrum Report | **Help establish accountability within the team by documenting commitments and progress towards those commitments. Team members are accountable for completing their assigned tasks within the agreed-upon time frame** |
| Reflections | **Reflection encourages the group members to enhance their self-awareness, communication and collaboration which leads to overall improved performance and productivity within a team** |
|  |  |
|  |  |

**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 |
| Allocation of tasks | **Group members identified what they were comfortable with doing and set off to do what was required for each task.** |
| Completion of tasks | **Since group members did what they were comfortable with and asked for clarifications during the group call, the progress went smoothly for each deliverable.** |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

**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 |
| Time Constraints | **Some of the deliverables in the project took longer time than expected** |
| Attempting to create new data structures | **There was some general confusion about how to complete this part of the project, we were able to figure it out after some time to discuss and think** |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

**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 a problem and designing software functions without writing the software is more challenging than writing the software to complete the project. This can be due to several factors such as clarity of requirements, complexity assessment, design considerations and risk management. Without writing the software, it can be challenging to fully understand and clarify all the requirements of the project. Ambiguities or misunderstandings in the problem statement can lead to inaccuracies or inefficiencies in the design of the software functions. It can be challenging to accurately assess the level of complexity and anticipate potential pitfalls without hands-on experience in implementing the software. Designing software functions involves making numerous decisions regarding architecture, data structures, algorithms, and interfaces. Without writing the software, it can be challenging to evaluate the effectiveness and feasibility of different design choices and their implications on performance, scalability, and maintainability. Developing software without writing it involves inherent risks, such as overlooking critical requirements, underestimating complexity, or making incorrect assumptions. Managing these risks effectively requires careful analysis, planning, and mitigation strategies.

* 1. Describe two advantages of developing software in this manner rather than just moving on to writing the functions without writing specifications first.  
       
     Two advantages of developing software in this manner rather than immediately writing the functions without writing specifications first are:

Improved Clarity and Understanding: Analyzing the problem and designing software functions allow for a deeper understanding of the project requirements and objectives. By clarifying the specifications and identifying potential challenges upfront, developers can reduce the risk of misunderstandings and ensure that the software solution aligns closely with stakeholders' expectations.

Efficient Resource Utilization: Investing time in analyzing the problem and designing software functions upfront can lead to more efficient resource utilization in the long run. By identifying potential issues early in the development process, developers can avoid costly rework, reduce the likelihood of project delays, and allocate resources more effectively to address critical aspects of the software solution.

1. Why is it a good idea to create a test plan? Describe at least 3 advantages of test plans.  
     
   With test plans being the most fundamental documents for software testing, it is a good idea to always create one because it provides a structured approach to testing, and at the same time ensures the reliability and quality of the software product, thereby giving an overview of how all the software testing should be conducted.

The advantages of test plans are :

i. Collaboration and communicatoin: Test plans usually serve as a communiaction tool among project stakeholders, project managers, developers, testers, and finally clients. Documenting testing objectives, the methodologies to be used, and criterias for project success enables ensure that all stakeholders have a clear understanding of the testing process and the expectations. This fosters collaboration and alignment among team members, enabling them to work towards a common goal. Test plans also provide project status updates which enables track progress, identify potential difficulties and make decisions throughout the testing phase.

ii. Structured Approach: Test plans outline an approach to testing by defining the list of constraints and assumptions on the testing, scope, resources (test tools required), schedules, responsibilities and software components and business processes to be validated. This ensures that testing activities are organized and executed efficiently, preventing redundant efforts and ensuring a comprehensive coverage of the software's functionality. Following a predefined and well structured plan enables the team to prioritize their tasks effectively.

iii. Risk identification and mitigation: Test plans also help in identifying potential risks associated with the project early in the development process. Testers can anticipate potential issues like bugs, compatibility issues across different devices or platforms by defining test scenarios and test cases. Adressing risks during the testing phase will significantly reduce the likelihood of encountering critical issues, thereby enhancing the quality and reliability of the software.

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

The process we used to analyze and understand the existing software is by reviewing the structure and header configuration files to better understand the implementation and to locate any potential errors or improvements.

The first step is to gather and obtain available documentation that is related to software. By reviewing the document it provides insights into an software's intended purpose, feature, and usage instructions. It's practically a guide that allows us to understand the software's context and objectives.

The second step would be to evaluate the software's structure, by identifying the modules, layers, and components. We do this to understand the architecture as it also helps comprehend how the different parts of the software interact with each other and contribute to the overall functionality.

The third step would be to examine the source code to understand and delve deeper into the software's implementation. We would review the codebase, patterns, coding, and organization to gain insights into the development.

As well identify external libraries, frameworks, APIs that the software relies on.

We also maintained conversations within the group to clear up misunderstandings. We documented our findings to share with each other.

Please let me know if you want to change anything.