# **MILESTONE 1** -- 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**: **E**

**Members Present**:

|  |  |
| --- | --- |
| 1. Jay Vijaykumar Vakil | 4. Srujal Patel |
| 2. Mohit Sheth | 5. Viren Vaishnav |
| 3. Harsh Dugar | 6. Hetav Mamtora |

**Milestone 1 Tasks**

In this phase of the project you will:

* Setup teams of about 3-5 developers (6 is too large)
* Write and sign a team contract
* Create a GIT account
* Create a Jira account
* Add your professor to the GIT and Jira accounts
* Update Jira with the work performed and planned

**Deliverables Due at End of Lab**

* Completed SCRUM report & reflections

**Deliverables Due 24 hours after lab**

* Completed team contract
* Fully initialized Git repository
* Fully setup Jira project

**Rubric**

|  |  |  |
| --- | --- | --- |
| **Individual** | Group Participation | 75% |
| Teamwork | 25% |
| **Group** | Contract | 15% |
| Git Repository | 25% |
| Jira Project | 25% |
| SCRUM Report & Reflections | 35% |
| **NOTE** | Both the individual and group marks are calculated separately. Each member of the group will have their mark calculated based on their contribution to the group work and their contributions to the team. The group participation is a percentage that your professor feels you contributed to the group work. This is multiplied by the weight of the group participation component to determine your grade. |  |

**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** |
| **Jay Vijaykumar Vakil** | **Github Account, Github Repository, Jira Account, Jira Account** | **NA** |
| **Mohit Sheth** | **Github Account, Jira Account** | **NA** |
| **Harsh Dugar** | **Github Account, Jira Account** |  |
| **Hetav Mamtora** | **Github Account, Jira Account** |  |
| **Viren Vaishnav** | **Github Account, Jira Account** |  |
| **Srujal Patel** | **Github Account, Jira Account** |  |
|  |  |  |
|  |  |  |

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 |
| How the tasks will be divided |  | **The activity will be divided according to the extent of the milestone.** |
| Due to submit project files to the group leader | **1 Day Before The Group Assignment Is Due.** | **Everybody Agreed Upon this rule.** |
| How Will the upcoming milestones will be done? | **Meeting platforms, reading and understanding the tasks,etc** | **Will follow a procedure**  **Meeting -> Analysing -> implementing -> reviewing -> submitting.** |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

**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 |
| Prioritization of Tasks | Putting tasks in order of priority and urgency to guarantee that they are finished on schedule. |
| Addressing Problems Encountered | promptly identifying and resolving any issues that arise throughout the process. |
| Documentation of Architecture and Design | creating a record of important architectural and design choices. |
| Testing Approach | completing the testing strategy to guarantee software quality. |
|  |  |
|  |  |
|  |  |

**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? |
| Jay Vijaykumar Vakil | **Github Acc, Git Repo, Jira Account** | **15 minutes** | **YES** |
| Mohit Sheth | **Github Account, Jira Account** | **5 Minutes** | **YES** |
| Viren Vaishnav | **Github Account** | **5 Minutes** | **Yes** |
| Srujal Patel | **Github Account, Jira Account** | **5 Minutes** | **YES** |
| Harsh Dugar | **Github Account, Jira Account** | **5 Minutes** | **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 |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

**Major Outcomes of Meeting:**

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

|  |  |
| --- | --- |
| Outcome | Impact on Project |
| Problem resolution | **The team reviewed and decided on a proactive strategy to handle issues that arose throughout the project. This dedication to problem-solving will support project development and lessen any possible roadblocks.** |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

**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 |
| Active participation | **Seriousness towards Project and Participation encouraged teamwork and made sure that everyone was heard during the conversations.** |
| Open communication | **The group promoted an atmosphere of open communication where members felt free to express their ideas, worries, and feelings. This promoted openness and aided in efficient decision- and problem-making.** |
| Clear decision-making | **Regarding job prioritisation, issue solving, documentation, testing strategy, and communication, the team made judgements that were clear and well-informed.** |
|  |  |
|  |  |
|  |  |
|  |  |

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

**Reflections (to be answered by the group)**:

1. GIT is an example of a version control system. List and explain 3 benefits of using a version control system.  
     
   Version management tools can improve various effective communication amongst teammates who are working on the same projet assigned to them. Multipleteammates collabarate for working on sevral parts of the program paralley wihthout interfying with teammates work.Each and every developer is able to use Github to create their own section for combining a specific roles or probelm solving area.They can also execute various modifications on their own as per the system requirement.Version control system allows indivuals to track the changes that are been made in the code over a certain period of time.Everytime whenever a code is executed its snapshot is captured for future use.Some changes in the history can be also very helplful while troubleshooting and understanding the code while implementing it if any bug had occured iy can be easily identfied.This fucntion helps to roll back the changes and see teh project history for providing safety and helps to recover easily from the errors.Programmers can sometimes experiement with various new featurs for settign up additonal different versions with the help of Github's meachanism.Until it is ready to get merged developers do their work indiviually from the main code diividng their works in divisons that helps in faciltiating the execution of new features freely to implement and executed.
2. Jira is a modern, web-based tool for managing software projects. Describe 3 advantages of using a project management tool like Jira.  
     
   Jira has several advantages to offer for managing software projects. These are the three main advantages:

1. Collaboration: Jira provides us with a platform where we can collaborate, communicate, and access all the project-related information. It allows us to set requirements and the tasks to complete, and stores all the documentation and discussion in one place. This approach ensures that all the team members have access to all the updated information, reducing the miscommunication between the team members.

2. Workflow Management: this tool provides us with features for managing tasks and workflows. As stated in the above point it allows us to create tasks and also prioritize them accoringlly, assign them to a particular team member, and even track their progress. It adopts to different project methods such as Scrum or Kanban. This improves the overall efficiency of the team and streamline the task management.

3. Issue tracking: Jira excels in issue tracking and bug management, which is a very crucial step in software development projects. It allows teams to log and track issues such as bugs, feature requests or improvement. It has features like issue creation, assignment, status updates, attachments and commenting. This helps in resolving the issues efficiently and ensuring software development is up to the mark and efficient.

1. Write a brief history of the Kanban board. Describe why it is useful in a project like this one.  
     
   Developed by Toyota in the 1940s, the Kanban board is an effective visual project management tool. It provides a clear visual picture of the procedure, fostering team member participation and transparency. Prioritisation and effective job completion are aided by the board's structuring of tasks and their progression. By limiting the amount of work in progress, the Kanban board helps teams stay focused and avoid member overload. Furthermore, by detecting bottlenecks and promoting experimenting with fresh ideas, it promotes a culture of continual development. The Kanban bodard is useful for projects of any size or complexity because to its adaptability. Teams can visualise and monitor their progress, spot possible problems, and make timely modifications thanks to it. The Kanban board has gained popularity for its efficiency and simplicity.