# **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: 5**

**Members Present**:

|  |  |
| --- | --- |
| 1. Gabriel Yeung | 4.Sheng Chieh Lin |
| 2.Ali Riza Sevgili | 5.Sheida Hashem Dabbaghian |
| 3.Rong Gang Xu | 6. Hon Kit Mok |

**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 4 days after your lab day:**

* Completed team contract.
* Fully initialized Git repository. **Be sure to send your professor the link to your GitHub repository and a screenshot of the GitHub users.**
* Fully setup Jira project. **Be sure to send your professor the link to your Jira Project.**
* Completed scrum report including reflection questions answered.

**Rubric**

|  |  |  |
| --- | --- | --- |
| **Individual** | Group participation | 80% |
| Teamwork | 20% |
| **Group** | Contract | 15% |
| Git repository | 25% |
| Jira project | 25% |
| SCRUM report & reflections | 25% |
| Meets deadlines | 10% |
| **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** |
| **Gabriel Yeung** | **Setup github repository** | **0 ( No delay)** |
| **Gabriel Yeung** | **Setup jira board** | **0** |
| **Gabriel Yeung** | **Share git experience** | **0** |
| **Gabriel Yeung** | **Discuss tasks distribution** | **0** |
| **Gabriel Yeung** | **Add teammate to github repo** | **0** |
| **Gabriel Yeung** | **Add teammate to jira board** | **0** |
| **Hon Kit Mok** | **Register in the project git repository and jira board** | **0** |
| **Hon Kit Mok** | **Sign the project contract and contribute in the scrum report** | **0** |
| **Rong Gang Xu** | **Join the github repository** | **0** |
| **Rong Gang Xu** | **Join the jira board project** | **0** |
| **Rong Gang Xu** | **Learn how to use git** | **0** |
| **Rong Gang Xu** | **Learn how to use jira** | **0** |
| **Rong Gang Xu** | **Discuss tasks distribution** | **0** |
| **Ali Riza Sevgili** | **Share git experience** | **0** |
| **Ali Riza Sevgili** | **Discuss tasks distribution** | **0** |
| **Sheng Chieh Lin** | **Join the Github repository and the Jira board project** | **0** |
| **Sheng Chieh Lin** | **Discuss tasks distribution** | **0** |
| **Sheida Hashem Dabbaghian** | **Discuss tasks distribution** | **0** |
| **Sheida Hashem Dabbaghian** | **Join Jira board project and the GitHub repository** | **0** |
| **Sheida Hashem Dabbaghian** | **Learning how to use GitHub** | **0** |

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** | **0** |
| **Reason for delay or block** | **0** |
| **Impact on Project** | **0** |
| **Solution or work-around** | **0** |
|  |  |
| **Delayed or Blocked Task** | **0** |
| **Reason for delay or block** | **0** |
| **Impact on Project** | **0** |
| **Solution or work-around** | **0** |

**Summary of Meeting:**

A summary of the main points discusses in the meeting and the outcomes of the discussions.

|  |  |  |
| --- | --- | --- |
| Topic | Discussion Summary | Outcome |
| Create git repo | **Gabriel would be the group leader and in charge of committing git repo** | **All members agreed the abovementioned topic. Git repo was created.** |
| Create Jira | **Gabriel would be the person in charge of jira** | **All members agreed the abovementioned topic. Jira board was created.** |
| Finish scrum report | **Everyone in the group will be contribute their own part of scrum report** | **All members agreed and participating in the scrum report.** |
| Commit the agenda of next meeting | **Every members agree the next group meeting would be scheduled on next week Friday** | **Next meeting would be held on 10 Nov.** |
|  |  |  |
|  |  |  |
|  |  |  |

**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 |
| No development would be done in the first week | The first week will be setup week. |
| Roles of members would be confirmed during the next meeting | Work distribution is essential for completing the following progress of the project. |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

**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? |
| Gabriel Yeung | **Create git repo** | **15 min** | **Yes** |
| Gabriel Yeung | **Create jira board** | **15 min** | **Yes** |
| Gabriel Yeung | **Create ssh key for git** | **30 min** | **No** |
| Other members | **Register and join the Git repo and group Jira board.** | **30 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 |
| Gabriel Yeung | Provide insight on project management |
| Gabriel Yeung | Provide support on jira and git |
| Gabriel Yeung | Write scrum report |
| Gabriel Yeung | Commit and push works to git |
| Hon Kit Mok | Attend the git repo and request to join the jira board |
| Hon Kit Mok | Taking part into completing the scrum report |
| Rong Gang Xu | Learn how to do project management |
| Rong Gang Xu | Get more familiar with jira and git and start using them |
| Rong Gang Xu | Write the assigned part of the scrum report |
| Rong Gang Xu | Commit and push my works to git |
| Rong Gang Xu | Go through the project again and get more insights and understanding to prepare for the following milestones. |
| Ali Riza Sevgili | Provide insight on project management and discuss what the group should work on it during the later process |
| Ali Riza Sevgili | Discussion supported knowledges on Jira and git |
| Sheng Chieh Lin | Learn how to use jira and git. Familiarize yourself with the features inside. |
| Sheng Chieh Lin | Learn project management skills and discuss them with fellow students during the next meeting. |
| Sheng Chieh Lin | Study the next week's milestones beforehand. Understanding the content clearly in advance can improve the efficiency of the meeting and reduce the time of the next meeting. |
| Sheida Hashem Dabbaghian | Learn more about Jira and GitHub |
| Sheida Hashem Dabbaghian | Doing the scrum report |
| Sheida Hashem Dabbaghian | Learn how to do project management and working with group |
| All members | Taking up the designated roles of the projects and make preparation of the following progress. |

**Major Outcomes of Meeting:**

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

|  |  |
| --- | --- |
| Outcome | Impact on Project |
| Git repo creation | **Provided a space to store project process** |
| Jira board creation | **Provided a place to share the progress status** |
| Create the platforms of communication during the project | **Aparts from the Git and Jira board, members can communicate via group whatapps and MS Teams channel** |
| Provide Git operation procedures | **Increase group members' familiarity with github to improve the progress of milestone completion.** |
|  |  |
|  |  |
|  |  |

**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 |
| Git repo creation | **Instruction is clear** |
| Jira board creation | **Instruction is clear** |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

**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)**:

Answer the following questions using your own words. Make sure that each answer comprises a minimum of 100 words.

1. GIT is an example of a version control system. List and explain 3 benefits of using a version control system.  
   i. Version control system its name speaks itself. It keeps tracks of versions of your works. Whenever changes are made, it would be recognized and recorded by the system. Users can input message for each change and then commit to the system. There are a lot of information being collected and stored, for example, the change itself, change date, change by which person, etc. And thus users can trace and revert changes whenever things go wrong.

ii. GIT supports branch. Branch is a technology which is beneficial for collaboration and separation between production and development version control. Users can create many branches other than the main branch for different usage. It can be separation of development work copy and production work copy and also can be used to separate works between different users.  
  
iii. Conflict detection is another great benefit from version control system. From time to time, users may create a lot of branches. To a certain stage of development, users may merge different branches together. Different versions from different branches may have conflicts between them. Version control systems often comes with a function to compare versions and detect these conflicts, so that users can locate the problem in the works.

1. Jira is a modern, web-based tool for managing software projects. Describe 3 advantages of using a project management tool like Jira.  
     
   i. Jira is basically constructed for the Agile style project management. The functionality fits perfectly for Agile. One of the core ideas of Agile is to break down work into smaller portions, determine what is achievable in one sprint and then fulfill the target afterwards. Jira is designed for this style of project management.

ii. Jira is excellent for users and managers to track the progress of works. There are stages of tasks predefined for users. And then users and create tasks on those stages. Project managers and developers can move the tasks to different stages to indicate the progress of tasks. These changes will be shown to all jira board users so that everyone can see the progress of tasks which is great for project management.

iii. As being a web-based tool, jira supports real-time and instant update from users. When users update task detail, change progress status and leave comments to task, these can be shown instantly on other users on the board. This allows users solve problem and collaborate with others easily.

1. Write a brief history of the Kanban board. Describe why it is useful in a project like this one.  
     
   Kanban is a Japanese word which basically means board. So Kanban board actually means board board. It was first developed by a Japanese Engineer Taiichi Ohno in the early 1940s. The aim of Kanban board is to control and manage work at every stage of production at Toyota. It was found that the productivity and efficiency of Toyota are slower than the American competitors. So they developed this system to better manage tasks and materials needed at different stages. In result, the productivity of Toyota in car making increased and also reduced wastes of money in inventories. The modern Kanban board system is developed from the Toyota’s one. The idea of managing small details in different stages in the core of it. It is very suitable for IT project management.