# **Milestone 1 Scrum Report**

All students are expected to attend the scrum meetings and to participate. Failure to do so will result in greatly reduced grades.

**GROUP**: \_\_\_\_\_\_\_\_\_\_\_\_\_SFT211-ZFF\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Members Present**:

|  |  |
| --- | --- |
| 1. Sidhart Variyath | 4. Sofiia Parkhomenko |
| 2. Fahad Bin Adbullah | 5. Mihir Bakulbhai Patel |
| 3. Anurag K C | 6. Jaskaran Singh |

**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 | 25% |
| Git repository | 25% |
| Jira project | 25% |
| Scrum report & reflections | 25% |
| **Deadline** | 20% deduction for each day you are late |  |
| **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** |
| **Sidhart Variyath** | **Group Contract, SCRUM** | **N/A** |
| **Fahad Bin Adbullah** | **Group Contract, SCRUM** | **N/A** |
| **Anurag K C** | **Group Contract, SCRUM** | **N/A** |
| **Sofiia Parkhomenko** | **Group Contract, SCRUM** | **N/A** |
| **Mihir Bakulbhai Patel** | **Group Contract, SCRUM** | **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** | **None, project begins** |
| **Reason for delay or block** | **N/A** |
| **Impact on Project** | **N/A** |
| **Solution or work-around** | **N/A** |
|  |  |
| **Delayed or Blocked Task** | **N/A** |
| **Reason for delay or block** | **N/A** |
| **Impact on Project** | **N/A** |
| **Solution or work-around** | **N/A** |

**Summary of Meeting:**

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

|  |  |  |
| --- | --- | --- |
| Topic | Discussion Summary | Outcome |
| Contract | Finished Contract | Signed |
| Scrum | **SCRUM done** | **finished** |
| Jira | **Jira setup** | **complete** |
| Git | **Git setup** | **complete** |
|  |  |  |
|  |  |  |
|  |  |  |

**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 |
| Limited Team | Compromise and adapt to situation of a 6-member team |
| Prioritization of the tasks | Equal number of tasks distributed to limited member of the team |
| Division of the tasks | Tasks distributed of corresponding ability to limited member of the team |
|  |  |
|  |  |
|  |  |
|  |  |

**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? |
| All | Signing contract via digital method | 4 hour 30 mins | Yes |
| All | **Writing SCRUM report** | **2 days** | **Yes** |
| All | **Jira steps** | 2 days | **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 |
| Sidhart Variyath | Scrum and reflection |
| Fahad Bin Adbullah | Handle files, gave instructions, test plan, |
| Anurag K C | Complete milestone 2 (11/5/2024) |
| Sofiia Parkhomenko | Create test plan |
| Mihir Bakulbhai Patel | git commit and control, Jira, Assist with test plan |
| Jaskaran singh | contract |
|  |  |
|  |  |
|  |  |
|  |  |

**Major Outcomes of Meeting:**

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

|  |  |
| --- | --- |
| Outcome | Impact on Project |
| Finished Contract | Agreed on the requirements from all group member |
| SCRUM done | **Milestone 1 SCRUM report completed** |
| Jira setup | **Assigning and tracking project progress** |
| Git setup | **Setup done** |
|  |  |
|  |  |
|  |  |

**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 |
| Contract | All signed |
| SCRUM | **All contributed** |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

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

**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.**   
   A version control system (VCS) like Git helps teams work together by letting each person have their own copy of the code, which they can merge without conflicts. It keeps track of every change made, who made it, and why, so you can see the history of the project and go back if needed. It’s also a safe backup for your code; if something breaks or you accidentally delete something, you can easily restore a previous version. In short, a VCS makes teamwork, tracking changes, and code safety easy and reliable.
2. **What is a version control system? Why does GitHub qualify as a version control system?**

A version control system (VCS) is a tool that helps manage changes to files over time. It keeps track of every change made, who made it, and when. This allows users to go back to previous versions, collaborate with others without overwriting work, and understand the history of a project.

GitHub qualifies as a version control system because it uses Git, a popular VCS, to track and manage changes in code. It also provides an online platform where multiple people can work together on the same codebase, making it easy to collaborate, review changes, and keep a complete history.

1. **What is Jira? How are we going to use Jira for this project?**

Jira is a tool used to manage projects, tasks, and workflows, especially in software development. It helps teams organize and track their work by creating tasks, assigning them, setting deadlines, and tracking progress. Jira is often used to follow agile methods, like Scrum or Kanban, where tasks are divided into small, manageable pieces called "issues" (such as bugs, features, or improvements).

We us jira to assign the work to the team members an individual task. So that and see weather it is in progress or it in the to do list or it is done.

1. **Why is a Kanban board useful in software development. What are the advantages of using Kanban board?**

A Kanban board is useful in software development because it visually organizes tasks, making it easy to see the status of each one. It typically has columns like "To Do," "In Progress," and "Done," which give a quick overview of what’s happening in the project.

Advantages of Using a Kanban Board:

Improved Visibility and Clarity

Team members can see what tasks are pending, ongoing, or completed at a glance, which helps everyone stay on the same page.

Better Workflow Management

Kanban boards limit the number of tasks in progress, reducing bottlenecks and helping teams focus on completing tasks before starting new ones.

Flexibility

It’s easy to add, remove, or reprioritize tasks on a Kanban board. This adaptability is useful in development, where priorities often change.