# **Milestone 2 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**: B

**Members Present**:

|  |  |
| --- | --- |
| 1. Isabela Jorge Bulla | 4. Ketia Teta |
| 2. Abdullah Al Mahfuz | 5. |
| 3. Devarsh Patel | 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, updated in the project, and added to the repository) | 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** |
| **Isabela Jorge Bulla** | **Managing Jira, scrum report and reflection question 2. Committing test plan and scrum report to GitHub repository.** | **-** |
| **Abdullah Al Mahfuz** | **Making header files and reflection question 1. Committing source code files to GitHub repository.** | **-** |
| **Ketia Teta** | **Making test plan.** | **-** |
| **Devarsh Patel** | **Reviewing test plan and answering reflection question 3.** | **-** |
|  |  |  |
|  |  |  |
|  |  |  |

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 |
| Analysis | **Analysis of the project requirements and deliverables.** | **Better understanding of the project** |
| Header files | **Discussion about the header files.** | **-** |
| Reflection questions | **Discussion about reflection questions and how they should be answered.** | **-** |
| Scrum Report | **Filling scrum report** | **Scrum report 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 |
| Jira tasks posted early in the week | Making sure that every member of the team has enough time to review the tasks and complete them. |
| Meeting every Tuesday 7:30pm | Have questions prepared for the class every Thursday. |
| Reflection questions | Each week the reflection questions will be posted separately on Jira, so that the work is equally divided between the team members. |
| Developing of the code | All the team members are responsible for developing the code and test cases. |
|  |  |
|  |  |
|  |  |

**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? |
| Isabela Jorge Bulla | **Scrum report** | **10min** | **Yes** |
| ALL | **Discussion about the problem and program requirements** | **25min** | **Yes** |
| ALL | **Discussion about Test Plan** | **15min** | **Yes** |
| ALL | **Discussion about Reflection** | **10min** | **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 |
| Isabela Jorge Bulla | Management of Jira, filling the scrum report. |
| ALL | Meeting every Tuesday at 7:30pm. |
| ALL | Reflection. |
| ALL | Function Specifications. |
| ALL | Creation of black box tests. |
| Abdullah Al Mahfuz | Update source files in GitHub. |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

**Major Outcomes of Meeting:**

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

|  |  |
| --- | --- |
| Outcome | Impact on Project |
| Completed scrum report | **Deliverables ready on time.** |
| Discussion of the entire project | **Guaranteeing that all the team members are engaged in the project and aware of their responsibilities.** |
| Reflection | **Every team member cleared up their doubts regarding the reflection’s questions and how they should be answered.** |
| Test Plan | **Better understanding of the project.** |
|  |  |
|  |  |
|  |  |

**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 |
| Test Plan discussion | **Great communication about the ideas and expectations regarding the test plan.** |
| Contribution | **All the group members were present and contributed to the project.** |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

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

**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 were asked to design the data structure for the project. Print the data structure below then explain each item.  
     
   **The data structure:**

//Struct that tracks status of trucks

struct Truck {

int weight; // A maximum weight of 2500 kg cargo carying capacity

double volume; // A maximum size of 100 cubic meter box

struct Shipment\* shipment;

};

//Structure based on required input needed for the Truck Delivery software

struct Shipment {

int c\_weight; // A cargo can weight from 1 to 2500 kg

int c\_size; // A valid cargo come in 1, 3, 5 cubic meters in size

char c\_destination[DEST\_COORDINATE + 1]; // Max 3 characters to define where the box needs to go from the city grid

};

**Explanation**: Based on the instructions for the Truck delivery software-

the structure “**struct Truck**” defines the properties of the delivery truck. The Truck structure has 3 fields (weight, volume, shipment ). The **weight** field represents the maximum weight carrying capacity of the truck, specified as an integer. The maximum weight capacity is 2500 kg. the **volume** field represents the maximum volume capacity of the truck, specified as a double. The maximum volume capacity is 100 cubic meters. The **shipment** field is a pointer to the **struct Shipment**. It represents the shipment that the truck is currently carrying or scheduled to carry.

The structure “**struct Shipment**” defines the properties of the shipment/cargo that needs to be delivered to a certain destination. It has 3 fields (c\_weight, c\_size, c\_destination). The **c\_weight** field represents the weight of the cargo in Kilograms. A valid cargo can range from 1 to 2500 kg. the **c\_size** field represents the size of the cargo in cubic meters. The valid cargo sizes are- 1, 3, 5 cubic meters. The c\_destination represents the destination of the shipment within the city grid ( which is a number followed by a letter).

Its defined as a character array (char c\_destination[DEST\_COORDINATE + 1]), where DEST\_COORDINATE is a constant representing the maximum size of the destination coordinate string (up to 3 characters plus one for the null terminator).

1. Describe the process you used to analyze and understand the existing software code.  
     
   First, we examined the PDF file containing the project description and program expectations. This allowed us to understand the project's main idea and its requirements. We identified the important data structures and functions necessary to successfully complete the project and implement a working program.

Subsequently, we conducted a thorough review of mapping.h, identifying the existing data structures and their contributions to the program. Additionally, we analyzed the .c files: mapping.c and main.c. In mapping.c, we studied the behavior of each function and their intended usage within the main program.

During program execution and subsequent comparison with sample outputs, we observed that certain functions, such as printMap, while unnecessary for the final output, remain crucial for testing purposes.

1. What aspects did you consider when creating the test plan? What were the milestones you identified in the test plan?

First, we have gone through the material that is already provided the header file, and the source code files, and identified the potential areas which are taken into consideration for improvements or issues.

While creating a test plan we considered aspects like objective and scope which help the algorithm assign packages to trucks based on capacity, destination, and route availability. Test strategy which helps functional testing for validating key functions, and system testing for overall functionality testing. Execution helps define criteria for starting and completing tests. Control procedures for common reviews, bug reviews, etc. Resources and responsibilities are divided among all the members to ensure the smooth execution of the test plan. And there are some more of it which we considered while creating a test plan.

As mentioned above, milestones of the project like analyzing truck capacity, package dimensions, shipment allocation, and route are identified. Some other factors like designing different test scenarios for different package sizes, weights, and route complexities are looked after. Also ensured the right environment for the project by getting the required tools installed. These were the milestones that were identified in the test plan and taken into consideration.