# **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**: 5

**Members Present**:

|  |  |
| --- | --- |
| 1.Jubin Verma | 4.Photswat Boonme |
| 2.Sandhya Timsina | 5.Udhav Tamyal |
| 3.Susinta Bastola | 6. Parshv Nileshbhai Gandhi |

## 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** |
| **Jubin verma** | **Scrum report, Jira management** | **No delays or blocks** |
| **Sandhya Timsina** | **Code** | **No delays or blocks** |
| **Susinta Bastola** | **Test plan** | **No delays or blocks** |
| **Photswat Boonme** | **Reflection questions** | **No delays or blocks** |
| **Udhav Tamyal** | **Code** | **No delays or blocks** |
| **Parshv Nileshbhai Gandhi** | **Test plan** | **No delays or blocks** |

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 |
| Task assignment and distribution | **Who wants to and is assigned which tasked** | **Assignment of tasks effectively with regards to each members strengths and weaknesses** |
| Test plan ms2 | **Understanding the given test plan template and further the requirements** | **Well understanding of test plan and who is to do it** |
| Scrum report for ms 2 | **Understanding the given scrum report template and further the requirements** | **Assignment of whom to do the scrum report** |
| Ms 3 | **Discussing what is required in ms3** | **Better understanding of ms3** |
| Code of ms 2 | **Understanding and discussing what is needed to be made in ms 2 according to the requirements and further according with future milestones** | **Better understanding of milestones and its code** |
| Reflection questions | **What the reflections questions are and who is to do it** | **Better understanding of project entirely** |
|  |  |  |

**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 |
| Code structure | To assign the coding structure according to ms2 and ms3 to two members rather than individual |
| Test plan structure | To assign test plan to two members rather than 1 individual according to ms2 and more. |
| Task assignment and decision | Tasks were assigned according to their strengths and weaknesses |
|  |  |
|  |  |
|  |  |
|  |  |

**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? |
| Jubin verma | **Scrum report, Jira management** | **1hr** | **Completed** |
| Sandhya Timsina | **Code** | **1hr** | **Completed** |
| Susinta Bastola | **Test Plan** | **1hr** | **Completed** |
| Photswat Boonme | **Reflection questions** | **1hr** | **Completed** |
| Udhav Tamyal | **Code** | **1hr** | **Completed** |
| Parshv Nileshbhai Gandhi | **Test Plan** | **1hr** | **Completed** |

**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 |
| Jubin verma | Checking everyone’s work, scrum report creation. |
| Sandhya Timsina | Coding, bringing snacks for next meeting. |
| Susinta Bastola | Testing |
| Photswat Boonme | Reflection questions |
| Udhav Tamyal | Coding |
| Parshv Nileshbhai Gandhi | Testing |
|  |  |
|  |  |

**Major Outcomes of Meeting:**

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

|  |  |
| --- | --- |
| Outcome | Impact on Project |
| Better understanding of MS\_2 Coding | **Faster and more efficient Completion of project(Milestone2)** |
| Better understanding of project | **Faster and more efficient Completion of project(Milestone2)** |
| Better understanding of GitHub | **Faster and more efficient Completion of project(Milestone2)** |
| Better understanding of MS\_2 Test Plan | **Faster and more efficient Completion of project(Milestone2)** |
|  |  |
|  |  |
|  |  |

**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 |
| Code | **Good communication, Teamwork, overall great team synergy.** |
| Test plan | **Good communication, Teamwork, overall great team synergy.** |
| Reflection questions | **Good communication, Teamwork, overall great team synergy.** |
| Assignment of tasks | **Good communication, Teamwork, overall great team synergy.** |
|  |  |
|  |  |
|  |  |

**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.

struct Map

{

                  int squares[MAP\_ROWS][MAP\_COLS];

                  int numRows;

                  int numCols;

};

typedef struct Map Map;

struct Point

{

                 char row;

                  char col;

};

struct Route

{

                  struct Point points[MAX\_ROUTE];

                  int numPoints;

                  char routeSymbol;

};

struct Truck

{

                  char truckID;

                  double currentWeight;

                  double currentVolume;

                  struct Route route;

};

typedef struct Truck Truck;

struct Shipment

{

                  double weight;

                  double volume;

                  struct Point destination;

};

typedef struct Shipment Shipment;

To efficiently handle and arrange related data, the struct keyword is used in the provided C code to define a variety of data structures. The ***struct Map*** is a representation of a two-dimensional map that has integers numRows and numCols to indicate the map's dimensions and a 2D array named squares to store map data. With two members, row and col, denoting the row and column indices, respectively, the ***struct Point*** defines a particular location on the map. An array of Point structures, or points, that make up the route, together with numPoints to keep track of the number of points and routeSymbol to designate the type of route (e.g., BLUE, GREEN, YELLOW), comprise the ***struct Route***, which represents a path on the map. The ***struct Truck*** contains the following data about a truck: truckID, currentWeight, currentVolume, and a Routestructure that describes the truck's itinerary. When a shipment is being transported, its weight, volume, and destination point are all represented by the ***struct Shipment***. When combined, these data types allow a mapping application to be effectively modelled, making activities like cargo tracking and route planning easier.

1. Describe the process you used to analyze and understand the existing software code.  
     
   To analyze and understand the existing code and then create a new fixing code, I followed a step-by-step process to identify and resolve the issues. Here’s how I approached it:

**Read the Existing Code**: First, I went through the entire code to understand its structure and functionality. I identified the purpose of the code, the classes or functions involved, and the overall logic behind it. Since we a beginner to intermediate level of coding in C, I took my time with each line to understand what each statement does.

**Identify Problems**: While reviewing the existing code, I looked for areas where the program might not work correctly. I checked for common issues such as incorrect variable declarations, improper use of functions, or logical mistakes that could lead to wrong outputs or errors.

**Research Solutions**: Once I found possible issues, I researched the correct ways to fix them. This might include looking up documentation, revisiting C concepts, or learning about better coding practices.

**Plan Fixes**: I planned how to modify the code to fix the issues I found. This involved thinking about how changes in one part of the code might affect the overall program.

**Implement Fixes**: I then wrote the second version of the code by correcting errors, simplifying logic, and improving readability. My goal was to ensure the new code works more efficiently without causing new issues.

**Test and Compare**: After writing the fixing code, I compared it with the existing code l to see if it solves the problems I identified. I also ran both versions to ensure the second code worked better.

This process helped me systematically understand and improve the existing code, leading to a more functional second version.

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

When creating the test plan for the package-delivery application, I considered various critical aspects. First, the primary objective was to ensure the system's reliability, focusing on functionality, performance, capacity, stability, and user satisfaction. The test plan aimed to confirm that the system correctly assigns packages to trucks, optimizes routes, and handles various package sizes and weights.

Key milestones in the test plan included designing and documenting test cases, executing tests for functionality and performance, conducting regression tests, and logging and resolving defects. Testing was divided into stages, with each milestone focusing on different aspects such as system functionality, performance, security, and documentation. Finally, stress and recovery testing ensured the system's robustness under extreme conditions, leading to a stable and efficient solution.