

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: 10

Members Present:

1. Duong Truong Phuc Nguyen	4. Ahnaf Tahmid Khan
2. Syed Abdullah	5. Jasmin Grace Precioso Aro
3. Huynh Huy Hoang	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
Duong Truong Phuc Nguyen	reflection question 2 + create header file with data structures + update the test plan in 5. , 10. , 11., 12., 14., 15., 16., 17.	
Huynh Huy Hoang	Create respository in github, upload and manage files in github	
Ahnaf Tahmid Khan	reflection question 1	
Syed Abdullah	Set up and monitored Jira kanban board for Milestone 2 Completed a analysis of the problem reflection question 3	
Jasmin Aro	test plan initial	

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	N/A
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 discussed in the meeting and the outcomes of the discussions.

Topic	Discussion Summary	Outcome
Group tasks	Designated tasks for group members	Every group member knew what to do and when to complete it by
Separate tasks	Determine who will do which parts	Everyone know what to do in this milestone 2

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
Division of work	Divided work based on what work was left and first come first serve basis

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?
Duong Truong Phuc Nguyen	reflection question 2	25 mins	yes
Ahnaf Tahmid Khan	reflection question 1	40 mins	yes
Syed Abdullah	reflection question 3	30	yes

Scrum Tasks Selected for Next Week:

The tasks each member has selected to pursue for this class or the next week.

[illegible]

--	--

Major Outcomes of Meeting:

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

Outcome	Impact on Project
Division of labor	Everyone finished their part which led to a complete final 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
Division of labor	Set tasks for everyone, kicked off the progress for the project
Problem analysis	Helped members understand the program

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
-----------------	---

Attendance	Ideally everyone should attend the meeting. In order to improve we should decide on a date and time where everyone can attend

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 task for Milestone 2 was to design the data structures essential for the functioning of our project. To effectively store and access information pertinent to the project's needs—including the management of map, route, and truck data—we developed a data structure. Below is a summary of the components involved.

Map Configuration: This is a fundamental element where each node represents a location and each edge signifies a potential route. To visualize and link different areas of the map for route calculations, it contains fields for location coordinates and their identifiers.

Route Framework: This framework contains details about the starting and ending locations, the distance, and any potential pauses along the way. It facilitates the project's route coordination, allowing the program to efficiently manage multiple routes and choose the quickest or best routes.

Truck Configuration: This configuration includes all of the necessary features of each vehicle involved in the operation. Details including capacity, current location, and status (e.g., loaded or emptied) are included. This information ensures that route assignments are maximized for efficiency by supporting route distribution based on truck availability.

We took into account memory efficiency, data types, and compatibility with the existing code when designing these structures. In order to maintain modularity and facilitate testing, the design aims to clearly define the roles of each component and minimize the links between modules.

2. Describe the process you used to analyze and understand the existing software code.

I will describe the process that I used to analyze and understand the existing software code by the steps.

First, I assessed the requirements of the project by reading the project carefully to identify the main components like map, route, and truck. Therefore, I can understand the code needs to be implemented, and analyze the problems.

Second, I identified the module and main files to understand each file provided like mapping.h, mapping.c, main.c and their roles in the software. Also outlining the structure and the role of each file.

Third, I checked the data structure in the header file to understand the main data structures are used in the software because they identify how data like map, route, truck are organized.

Therefore, I can understand how the data of the project are stored and handled.

Fourth, I tried to read many times the definition and comments about the function to understand the function based on the name, parameter, and any comments that were provided or explained. Therefore, I read each function in mapping.c and check the input and output how the data that each function takes and returns to understand correctly the purpose of the function.

Fifth, I identified the location and how the functions were called from mapping.c and observed the flow execution like how the data of the map are initialized, manipulated, and used in calculating or output

Lastly, I identified which data structure or new function is necessary to meet the requirements of the project. Based on the requirement of the problem and code, I listed all the missing components like the structure for truck or the function to calculate the length of the path.

3. 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 we had to consider 17 aspects of how the process of testing the program would look like. This included the objective of the test plan, the scope of the test plan, the test strategy, the environmental requirements, the execution strategy, the schedule, the control procedures, the function tests, the resources and responsibilities, the deliverables, the suspension/exit criteria, the resumption criteria, the dependencies, the risks, the tools, the documentation and the approval. All of these aspects had to be considered in order for the program to have a comprehensive testing plan. With the objective we clarified what we were planning to achieve with our tests. With the scope we designated what parts of the program we would test. With the strategy we discussed which data would be tested and what type of test were to be used. The environmental requirements discussed what platform and program would be used in order to conduct the tests. The execution strategy discussed when the tests would take place, how their results would be divided and what reports would be created. The schedule was used to plan out how the program would be tested for that week. The control procedures

provided a way for reviewing the results of the test and bugs found, and adding changes. The function tests ensured all the functions created were running optimally. The deliverables discussed the actual data generated from the test. The exit criteria laid out how the test would be shut down. The resumption criteria would have the methods for resuming tests. The dependencies laid out what people, hardware and software would be required to run the tests. The risks discussed the problems that might arise with the testing process. The tools discussed if any additional help would be needed via hardware or software. The documentation dealt with the recording of tests and test data. Finally the approval sorted out if the tests and results would be approved at the end .

In terms of milestones all of the milestones of the project were recognized in the schedule. We clearly indicated which milestone would be tested during that week and what tests would be used.