# **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**: Group 3

**Members Present**:

|  |  |
| --- | --- |
| 1. Bilal Umar | 4. Nolan Grossi |
| 2. Grace Gabrielle | 5. |
| 3. Kaitlyn Marino | 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 rdeplacing 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** |
| **Bilal** | **Group Contract** |  |
| **Kaitlyn** | **Github + reflections 1 & 2** |  |
| **Grace** | **Setup Jira + reflections 3 & 4** |  |
| **Nolan** | **Several scrum report tables** |  |
|  |  |  |
|  |  |  |
|  |  |  |

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 |
| Scrum report | **Discussed between who would work on scrum report, how it would look in terms of details** | **Bilal will work on total scrum report, including tables, comments + reflections, and also be a free agent for any other help** |
| Test plan | **Worked on who would work on the test plan in tandem with adding code** | **Grace/Kaitlyn will work on the test plan** |
| Adding data structure and new .h file | **Must work together with tester** | **We decided Kaitlyn will work with Grace for testing/coding** |
| Analysis of code | **Must work together with tester and coder, to find out issues/requirements the code needs** | **Nolan will work with Kaitlyn in describing errors and creating a data struct with this in mind** |
| Jira updates/Git Updates | **Worked on updating Jira/Github for MS2, adding tasks etc** | **Grace will do this** |
|  |  |  |
|  |  |  |

**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 |
| Scrum report | Scrum report will be completed in full by Bilal, with exception of getting some information from other group members |
| Test Strategy/Document | Grace and Kaitlyn will work on the test, while also working on the code addition |
| Code analysis | Although no written documentation is required, writing issues with the existing code greatly helps in testing as well as strategy for code writing. |
|  |  |
|  |  |
|  |  |
|  |  |

**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? |
| Kaitlyn | **Test plan** | **~45m** | **Yes** |
| Nolan | **Code analysis + new code** | **~45m** | **Yes** |
| Bilal | **Total scrum report + new tasks assigning** | **~40m** | **Yes** |
| Grace | **Test plan, Jira Updates** | **~45m** | **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 |
| Kaitlyn | 2nd half of test cases + scrum report |
| Bilal | Blackbox test documents/Function specs + function implementation help |
| Nolan | Function implementation + marix |
| Grace | C++ testing + jira update |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

**Major Outcomes of Meeting:**

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

|  |  |
| --- | --- |
| Outcome | Impact on Project |
| Test plan creation | **Greatly improves writing code to follow a format, have a certain output/result etc.** |
| Code analysis | **Helps testing/writing code to fix issues** |
| Scrum Report | **Summarized work done, collaboration and greatly aided in understanding of project** |
| New code/header | **Satisfied project requirements** |
|  |  |
|  |  |
|  |  |

**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 |
| Creation of new code, analysis and testing | **Combining all 3 of these activities to a team effort makes it easier to each module to succeed, if working alone on each it’s difficult to test without knowing any errors for example.** |
| Scrum Report | **Group contribution to scrum report, but made/edited by Bilal** |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

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

**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.

//weight : The current weight the Truck is carrying

//volume : The space currently taken up within the truck

//truckId: A string to uniquely identify a Truck

This data structure defines a struct for the truck, helping identify each truck and organizing them, into storing the weight of them, the volume, and truck id for their unique route (i.e 111, will always be green, so it must go this way)

struct Truck {

int weight;

float volume;

char truckId[10];

};

//weight : The weight of the package

//size : The size of the package in cubic meters

//truckId: A string corresponding to a trucks unique identifier for assignment

//address: A point on the map representing the address to be deliverd to

The same data structure is applied to a shipment, there is multiple shipments so they must be uniquely organized and identifiable, by weight, size, the truck they are stored in, and the point (from mapping) that they belong to, so for example 2,1 on the grid.

struct Shipment {

int weight;

float size;

char truckId[10];

Point address;

};

1. How did analyzing the project requirements and design before starting the coding process help you identify potential challenges or define a clear development strategy?  
     
     
   Analysis before implementation was crucial, because this helped point out our errors, bugs, changes needed as well as streamline the coding process to be faster. The requirements of what was needed and analyzing them made it easier to proceed, knowing what needed to be done/made, i.e. in this case data structures, types and functions of it. In terms of strategy, it helped determine what needed to be focused on, while considering the result, which again speeded up the coding process, because if no analysis/planning was done it was cause the coding to be much slower, and problems could arise as you would go about it.
2. How did creating a test plan help you ensure comprehensive test coverage for the project? Reflect on how defining objectives, scope, and test cases in advance influenced the effectiveness and efficiency of your testing process.

The test plan was extremely specific and vast in terms of comprehensiveness, scope and etc for the project. In terms of ensuring this, it helped define a clear set objective, to test the program through each milestone, and the types of tests that would be done on each iteration. This addition greatly aided the testing process, instead of simply adding testing cases it helped narrow down margins, as for what needed to be tested, how, why and who by, this again streamlines the whole process to make it more efficient rather than test-as-you-go to, which would be inefficient and take more time as you test each iteration.