Milestone 4 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.Syed Abdullah
2.Ahnaf Tahmid Khan	5.
3.Huynh Huy Hoang	6.

Milestone 4 Tasks

- Finish implementing/coding the functions.
- Finish implementing/coding blackbox tests. Store in repo, executed, results in Jira (and on corresponding test documents, and debugged.
- A set of whitebox tests as test documents (in an Excel file) with test data for the functions you created. At least 4 sets of test data are required for each function. You must have test cases for at least 6 functions (including all your custom function). Stored in the repository.
- Whitebox tests implemented (in the C++ testing project), stored in repository, executed, results in Jira and on corresponding test documents, and debugged (at least 1 SET is required).
- Updated requirements traceability matrix in the repository, ensuring it shows both passed (green) and failed (red) tests.
- Completed hook file (for EACH team member) for test automation 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	Implemented functions and main (well-designed, and documented)	10%
	Finish coding blackbox code (well-designed, written, and documented)	5%
	Whitebox test case document (well written, complete, good test data)	10%
	Whitebox test code (well designed and documented)	10%
	Updated requirements traceability matrix	10%
	Test execution (performed, results recorded, issues created)	5%
	Debugging (bugs fixed, documented, Jira updated)	5%
	Hook files	15%
	Git usage (used properly with good structure)	5%
	Jira usage (creates issues, tracks progress)	15%
	Scrum report & reflections	10%
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 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
Duong Truong Phuc Nguyen	create set of functions and add to new header file and add to GitHub, 2 test data with 20 test cases, implementing the function and store in repository, created and add C++ testing project	N/A
Ahnaf Tahmid Khan	reflection 1 , reflection 2 , reflection 3	n/a
Huynh Huy Hoang	Create black box testing cases, test functions logic and manage the github respository	N/A
Syed Abdullah	BlackBox tests, requirement matrix traceability updated, updated jira project, hook files	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	Creating native test project on VS for unit tests
Reason for delay or block	Not sure how to set it up
Impact on Project	Needed to ask for extension
Solution or work-around	Worked on it during entension
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 discusses in the meeting and the outcomes of the discussions.

Topic	Discussion Summary	Outcome
Division of labor	**Discussed who would do what**	**Decided to let members pick task any time during the week**
Project details	**Discussed what would be needed to be created for submission**	**Clear understanding of task at hand**

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
Ask for a extension	We were not able to set up the Project for the unit tests and had to troubleshoot more.

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 could not be completed, the student should indicate why this was not possible.

Member	Task Attempted	Time Spent	Complete ?
Duong Truong Phuc Nguyen	Finish implementing/coding the functions, Completed hook file Test data are required for 2 functions Whitebox tests implemented (in the C++ testing project), stored in repository		**yes**
Ahnaf Tahmid Khan	**reflection 1 , reflection 2 , reflection 3**	**40 mins**	**yes**
Huynh Huy Hoang	**Create white box testing cases, manage github repository and submit the milestone.**		**yes**
Syed Abdullah	**Create blackbox tests on visual studio, handled project on jira, updated traceability matrix, hook files**		**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
Duong Truong Phuc Nguyen	
Syed Abdullah	Manage Jira and participate in group meetings

Major Outcomes of Meeting:

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

Outcome	Impact on Project
Clear objectives	**Everyone had a task we were able to build up a complete 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
Attendance	⅘ **members attended**

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
1 member did not attend the meeting	**Try to communicate more effectively**

Reflections:

Answer the following questions using your own words. Make sure that each answer comprises a minimum of 100 words.

1. Why did we wait until the fourth milestone to write the whitebox tests?

   We decided to develop the whitebox tests at the fourth milestone since the logic and architecture of the system were completely solidified and stable. Implementing whitebox tests earlier could have resulted in a large amount of rework if the codebase had undergone major changes during the prior phases. We were able to create more comprehensive and efficient tests that matched the finished code by the fourth milestone as we had a better grasp of the system's architecture, functionality, and possible edge cases. In addition to ensuring that the tests were pertinent and focused on the internal workings of the system, this approach reduced effort waste.

2. How does the Agile methodology ensure that all team members are consistently engaged throughout the software development process, avoiding downtime due to dependencies on others? Provide an example to illustrate your point.

   The agile technique maintains team members' constant engagement by encouraging cooperation, iterative development, and cross-functional roles. Team members identify challenges, modify priorities, and provide updates on their work at daily stand-up meetings, which promotes openness and minimizes downtime. Delivering incremental value is highly valued in agile, which frees people up to concentrate on smaller, autonomous tasks while simultaneously managing interdependencies. For example, while developers work on backend APIs during a web application development sprint, testers can develop test cases and designers can enhance the user experience. Everyone is able to contribute consistently without waiting for others to complete their work thanks to this parallel collaboration.

3. What is a shell script and how are we going to utilize a hook script in this project?

   A shell script is a text file that contains a series of commands designed to automate tasks using the shell, which is a command-line interpreter. In this project, we will use hook scripts to automate processes and establish workflow policies; these are specific shell scripts that activate in response to Git events like commits, pushes, or merges. For instance, a pre-commit hook script can be employed to automatically execute tests or verify code formatting prior to allowing a commit, ensuring code quality and consistency throughout the development process. This method minimizes errors and improves collaboration by identifying issues at an early stage.