SWE20001 – Development Project 1: Tools and Practices

Distinction Task <7.2D>

Duy Phuong Nguyen (101204984)

Lab: Friday 2:30pm

Tutor: Huai Liu

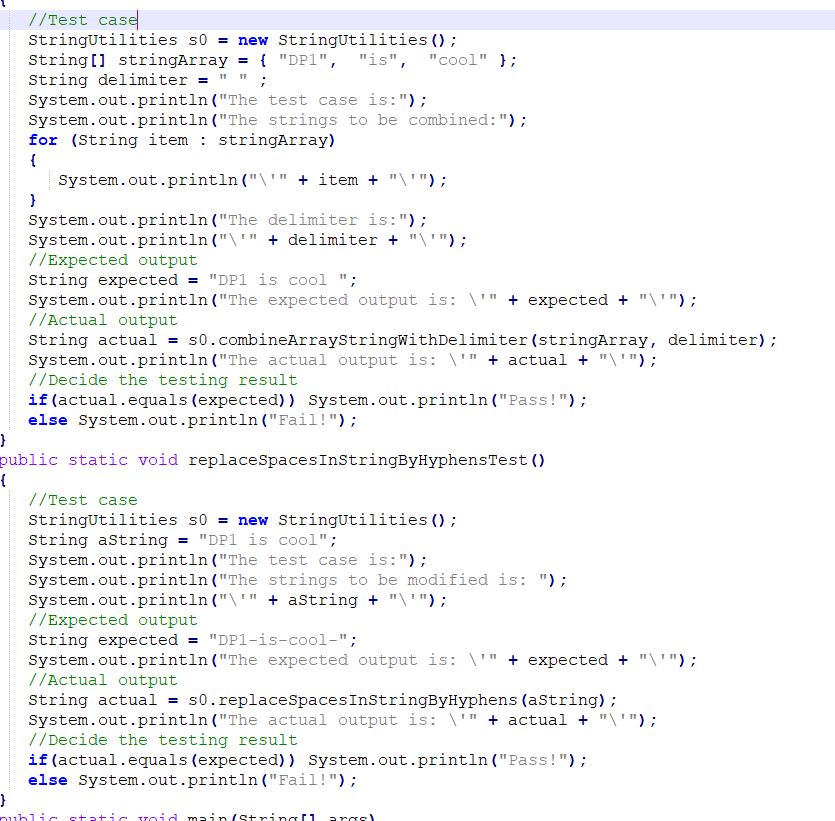
Team number: 3

Henry Tran 102075482

Edward Knight 102700043

Duy Phuong Nguyen 101204984

* Explain unit testing. Substantiate your description with examples from the software projects you were/are involved. Make sure that your explanation can be easily understood by a SWE20001 student.
  + UnitTesting is a process we test each method, functions of our program separately and see if it is working or not with some expected output. In this SWE20001 unit, in the Battleship project there are no Unit Testing used because that is not our code, so it is hard to understand what is going on. So, breakpoint is an easier method to find errors in the program. However, I have a chance to practices Unit Testing with 7.1P. It is using Java language and we must test the program with Unit Testing to get the expected output.
  + Below is the UnitTesting in 7.1P. We must check the function and match with the expected output to pass the UnitTesting. In below example, The Test case will create an input and run that input through the combineArrayStringWithDelimiter() function and replaceSpaceInStringByHyphens() function to get an output. The ouput then checked with the expected output. If they are matched, the program is correct and there are no errors.



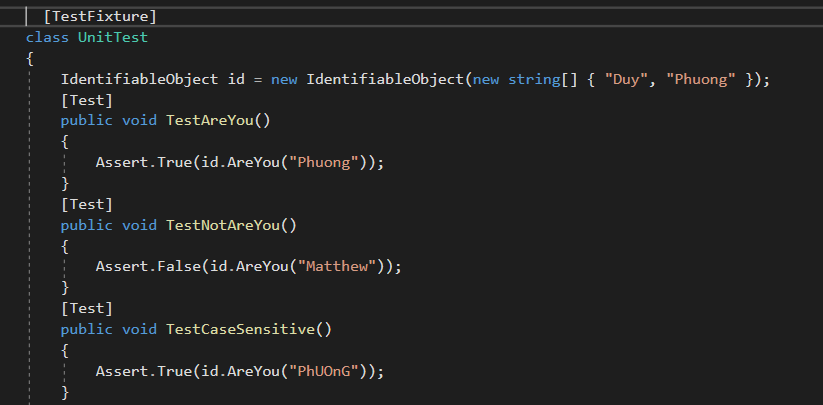
* Provide an in-depth analysis of the pros and cons of unit testing in the context of the projects you have been involved in (both in SWE20001 and in other contexts).

Pros of Unit Testing:

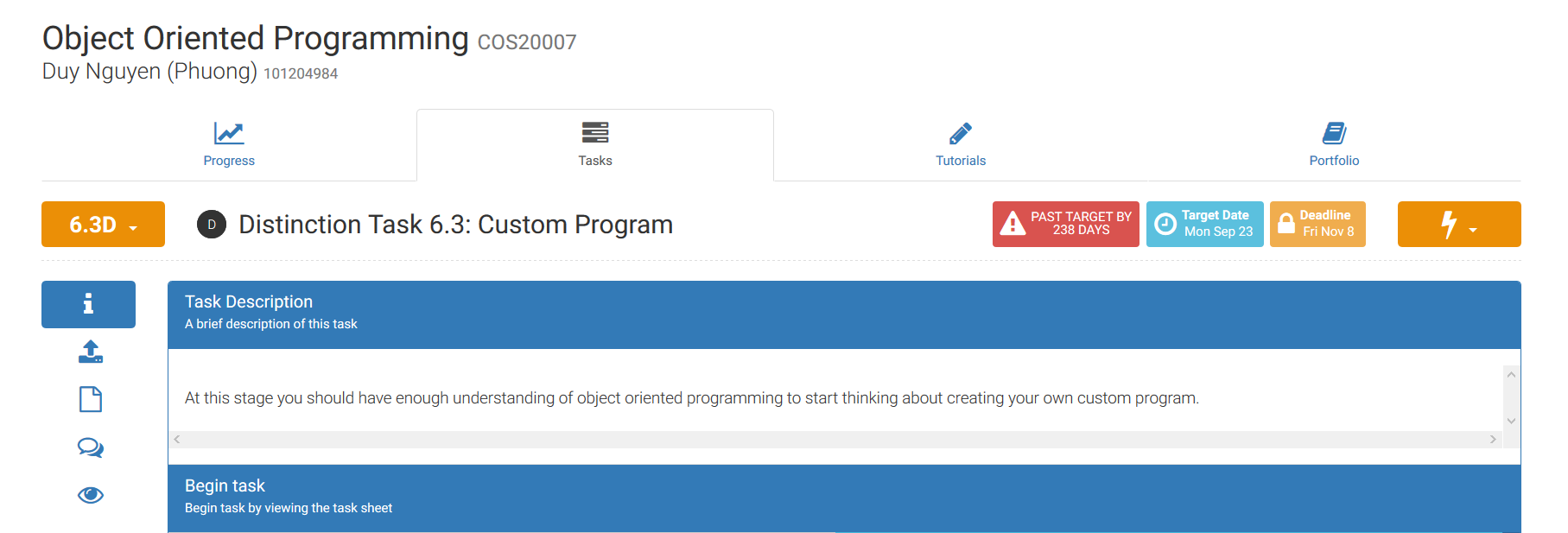
* + UnitTesting can help to reduce erros because when we add more code and features to the project, we have to make sure the old functions are working correctly. With the UnitTesting implemented beforehand, project developer can change/complete the code without worrying about any troubles and errors.
  + The quality of the code can be improved with Unit Testing.
  + Easy to find errors/bugs in the project. It provides hints to helps the developers fix the bugs.
  + Simplified the debugging process: When we implement new code and the testing is fail, we know that the latest code which the developer changes has caused the errors.

Cons of Unit Testing:

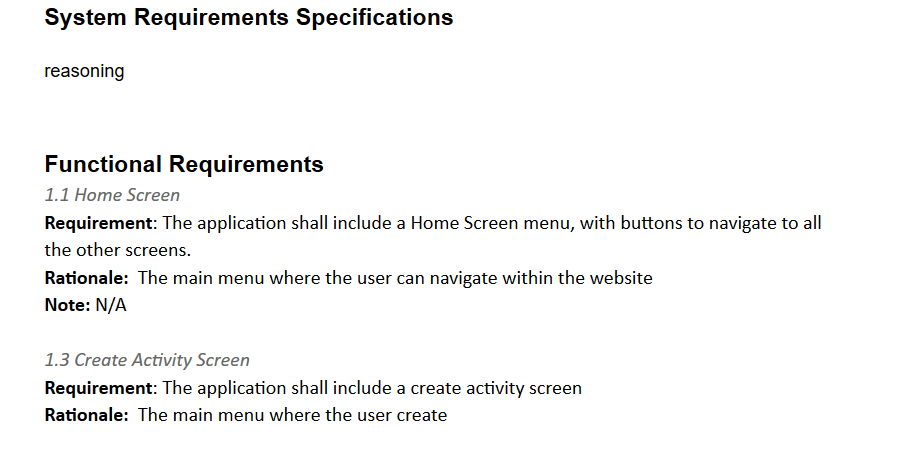
* + Time consuming: Unit Testing is required time to develop and complete, it may impact the completion of the main project. Sometimes, the project without UnitTesting can complete earlier than the project with UnitTetsing.
  + A bad quality UnitTesting might affect the whole project because it can pass a function with errors/bugs.
  + Unit Testing cannot prove the project success or not. It checks each unit separately as an independent unit.
  + With a big project, a good UnitTesting requires a lot of skills and time. This might caused the conflicts and confusion between developers.
* Describe how unit testing can have an impact on the project outcomes. What are the other practices you have done that either require unit testing or can work well with unit testing? Are there any practices that you have had experiences with and didn’t work well with unit testing? Again, it is important to substantiate your description with evidences from the projects you have been involved.
  + The UnitTesting can have a good impact on the project outcomes. Based on the Pros and Cons, we can clearly see that the UnitTesting makes the debugging process easier and it is also improving the quality of the code.
  + During the last semester, I enrolled in COS20007 - Object Oriented Programming which is a programming subject based on C# language. In order to complete subject tasks, I wrote and used a lot of UnitTesting. This method helped a lot in debugging the code and program. However, in my opinion UnitTesting requires a lot of times to use and implant. This creates a delay in project completion.



* + During the last semester, me and my friend worked in the same project and we tried to create an UnitTesting. The UnitTesting worked for my code but not worked for my friend. We tried to fix the UnitTesting but it is required advance programming skills, so we decided to remove UnitTesting for the project. This event effected a lot on our project progression, at the end we missed the deadline.



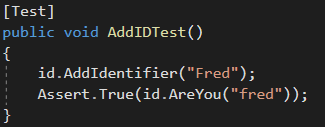
* + In this semester, we have a chance to create a group project for D/HD. Our group decided to create a website about Gym and Healthy lifestyle. It does not require any UnitTesting. However, with a group of 3, we have 2 developer and 1 product owner. The product owner will check the requirement and our product before completing the task. We follow the UnitTesting method which were introduced in 7.1P task to create a check list of requirements and rationales.



* Recommend guidelines for good practices with unit testing:

1. Determine with all the group members what components in the project need to be tested before start to write the UnitTesting.
2. UnitTesting can test and give all the expected outputs for the project. A good UnitTesting is self-sustaining, it is not depending on any outside variables.

For example, in this test method AddIDTest(). The UnitTesting will create an input “Fred” for AddID() method in the project and check with the expected output “fred”. This is a simple and independent test.



1. A good UnitTesting will create the same expected outputs whenever it runs. Its not depend on the system or where it is running.
2. Have a good naming system and in-line comments to help other developers understand the UnitTesting code. They can easy to understand what is being tested, what is the expected outcomes.

For example, in task 7.1P, we have to change the codes, make some improvement on the code to fix the bugs/errors. There are comments to helps student understand what is going on.

