G51FSE Coursework 4 – JUnit Testing

23-3-2017 – Max L. Wilson

Starting Premise: During the last meeting, Mayhem Grutton asked your team to begin devising tests to fix the bad code produced by Lax W. Milson and his team. Now, the new Lead Developer, Ace Jatkins, expects you to start fixing the code by first writing the tests, following proper Test Driven Development expectations. Ace Jatkins also hopes that you will follow some other good development principles, such as: detailed commenting, Paired Coding, and/or Version Control software.

Main Task: The main task for this page, for which you will be primarily assessed, is the development of good Tests in JUnit. You have three classes to build tests for, which can be downloaded from Moodle. You should try to build tests according to your test plan from the last lab, but it is normal to realise more tests as you start writing tests and debugging.

- You should document whether tests pass/fail.
- If failed, you should document the cause, what was fixed, and when they then passed.
- Further, using code comments, you should note who in your team authored each test.

Key Stages:

Stage 1: You should start by writing tests for the WorkshopPaper and WorkshopReview classes.

Stage 2: After finishing the first two classes, you should move onto Integration Testing, by building JUnit tests for the Main Class**.**

Secondary Task: You may also start improving and fixing the code.

- You should use comments in your code, to identify what changes were made, and by whom.

Submission: A **single zip file** to moodle, containing:

- a single java project folder, with java code and junit tests, that can be run by the marker. It should be clear in the junit and java code (from comments) who built which code/tests. When working in pairs, both persons should be listed, not just the person with the keyboard.
- a PDF of your test pass/fail log, identifying tests the date that tests passed. If a test fails at first, you should document why, and what was changed in the code to pass it, especially if tests were changed, or new tests were written, because of these failed tests.

Deadline: The deadline for this work is Thursday 6th April, 3pm.

PLEASE TURN OVER

Asynchronous Group Work & Timing: This is group work, and you are expected to work on this in your own time between now and the deadline, and must coordinate yourselves in your own time. You should do this as soon as possible, rather than wait until the next lab.

Labs: There are two labs during this time. You should attend these labs as a team, and this is a chance to ask questions to lab helpers. There will also be optional smaller exercises to help learn aspects of JUnit Testing.

1) Thursday 30th March – optional lab sheet to help with advanced JUnit tests.
2) Thursday 6th April – only a few hours before the deadline (!), you may wish to use it to get together and 'submit', rather than start new work.

Marking Criteria: In this work, you are being assessed primarily on your ability to write and use Unit tests.

1) The majority of marks are associated with the quality of your JUnit tests, including coverage and rigour of tests, and appropriate use of a range of different JUnit features, beyond just the basic @Test with AssertEquals().
2) Secondary marks are associated with the detail captured in your Test Report, about when and why tests fail, and what changed to make them pass.
3) Tertiary marks are associated with code that you fix – it is inevitable that some tests (especially at the integration stage) require you to fix code for other tests. Consequently, these tertiary marks are reserved for those that use the tests to improve the code. However, just fixing the code instead of writing tests, will not earn you any marks.

Peer Assessments: You will be expected to submit weekly peer-assessments, after the two labs. These should reflect the **amount of effort/work that each person did during the whole week**.