Faculty of Information Technology

FIT3162 Computer Science Project Part 2
FIT3164 Data Science Project Part 2

Assessment details
Prepared by Varun Mathur, Alan Dorin
Version 23082024

Test Report (Group submission)
10%, Due Friday Week 11, 11:55 pm

Learning Outcomes

1. Work effectively in collaborative teams;
2. Develop and test a substantial piece of software or hardware;
3. Explain and reflect upon the purpose, operation, success and value of the developed project in writing and orally;
4. Write a report explaining the methodology, outlining your contributions and the contributions of others, and documenting the developed project from appropriate perspectives (e.g., that of a user, researcher or developer).

Introduction

The purpose of a test report is to show that:

1. the software satisfies the initial and/or subsequent set of requirements. If the requirements were substantially changed from the initial set, the reasons for the changes and possible re-negotiation will be clearly discussed in the Final Report, to be submitted separately.

2. special conditions are properly handled (e.g., erroneous input data are handled correctly and do not lead to unexpected results, behaviour or crashes).

3. the main interface between your code and external libraries (i.e., API calls) works as expected (see seminar materials on black-box testing).

4. most software execution paths are traversed correctly and produce sensible output or meaningful error messages (Test coverage: see seminar materials on white-box testing).

5. robustness is considered (e.g., before writing to a file, the file exists and is accessible). All potential failure points outside of the software control have failure handling code.

6. the user interface (either command line or GUI) is user-friendly and simple.

7. functions (or most of the main functions) are tested separately and work correctly with a range of input parameters including boundary condition data. (Unit testing frameworks such as PyTest or PyUnit for Python or equivalent in other languages may be useful here).

Since all team members will be contributing to this report, it is important to ensure that the writing and formatting are consistent throughout the document.

Continue next page..

Size of report

It is expected that the report will showcase a good sample of completed tests. Size will depend on the specific software design and programming for each project. You are encouraged to review the software testing seminar and slides to plan your tests and to produce a test report.

Suggested report structure

1. Introduction and a brief description of the testing plan.

2. Description of the test approach used: either a manual test or using a framework such as PyTest, Jest or other similar frameworks.

3. The test should show:
   a) 2 examples of white-box testing carried out on different parts of your software (Data Science FIT3164 students are not required to do white box testing).
   b) 2 examples of black-box testing carried out on different parts of your software.
   c) Testing of the software as a whole (Integration Testing).
   d) Usability testing. Testing by a third party is highly encouraged where possible.

4. For each test, describe:
   a) What is being tested;
   b) How it is being tested;
   c) What are the inputs to the code or part of the code being tested;
   d) What are the expected outputs;
   e) What are the actual outputs being observed;
      i. If different from the expected output, is the test repeated after code modification was undertaken and what results are obtained? Or,
      ii. Is it left as a software "Bug"?
      iii. Note. Some software bugs may be acceptable if they are minor and are properly documented within the Limitations of the Software section of your report.

5. Recommendation for improvements, based on test results if possible.

6. Discussion of the limitations of the testing process. It is also important to mention tests that were NOT conducted. Explain why these tests were not conducted, how the operation of the software could be affected, and how these tests could be (have been) conducted? (This is not about limitations of the software. Limitations of the software are discussed in the Final Report).

7. Reports must be presented in a structured, consistent and easy-to-read format. A clear content structure requires sections with relevant subtitles, numbered and labeled figures and diagrams.

Faculty of Information Technology

FIT3162 Computer Science Project Part 2
FIT3164 Data Science Project Part 2

Team Members' Contribution Declaration

Team members' contributions must be declared in the Team Members Contribution declaration spreadsheet.

IMPORTANT: Failure to include a final Team Members Contribution declaration will incur an automatic 10% penalty on the final mark for the assessment.
Make sure that ALL team members need to click the SUBMIT button for the submission to be marked for grading.

1. 1 week before submission or earlier: Pre-declaration
   a. Download the contribution declaration spreadsheet from Moodle.
   b. Fill it in and submit it on Moodle at least 1 week before the assessment submission date.
2. On submission day: Final declaration
   a. Update the Pre-declaration spreadsheet with actual contributions.
   b. Submit the updated declaration form on Moodle on the 'FINAL- Team contribution (User Guide & Test report) link by the same deadline.

Marking

All Team Members are expected to make equitable contributions to this team task. However, marks are adjusted if the contributions of individual team members differ significantly.

Marking Guide

This assignment will be marked over 100 points. The maximum points allocated to each section is shown below:

Criteria	Marks(3162)	Marks (3164)
Introduction/Test plan	10	10
White-box testing (*Not required by FIT3164 students) Whitebox testing 1: - Test process (What is tested and how) (5) - Test result (5) White-box testing 2: - Test process (What is tested and how) (5) - Test result (5)	20	0*
Black-box testing (10x2) Black-box testing 1 (10) Black-box testing 2 (10)	20	20
Integration testing	10	10
Usability Testing (What & how)	10	10
Test Limitations/Discussion (note, Limitations are about limitations of the testing, not limitations of the software)	20	20
Presentation	10	10
Total	**100**	**80**

*For FIT3164 students, marks will be scaled to 100, to compensate for omitting white-box testing.

Late Submission

1. Submission must be made by the due date. Late submission penalties will be deducted according to Monash University policy. (See Moodle, "Additional information and resources" section).
2. If you believe that your assignment will be delayed because of circumstances beyond your control such as illness, apply for an extension prior to the due date. Extensions and individual alterations to the assessment regime will only be considered using the University Special Consideration Policy (See Special Consideration website, and follow the procedure).

Use of Generative AI tools in Projects and Assignments
See this assessment's Moodle module for the policy regarding use of GenAI in this unit.