Software Testing 2024: Your Portfolio

[Text in grey is guidance and should be deleted in your final portfolio. Text in blue is the statement of the learning outcomes you should retain this. The text of your portfolio should be in black. To structure your self-evaluation, you should retain the sub-sections of each learning outcome. Your text should evidence how well you have done is reaching the learning outcome.]

The aim of the portfolio is to provide evidence that you have taken the elements of the learning outcomes into account and have done some work that demonstrates you have achieved the learning outcome to some extent. The idea here is to point to the work you have done that evidences your having achieved the learning outcome. You should try to be as specific as possible, pointing out how particular pieces of work illustrate your learning.   Notice that you are not expected to produce a "perfect" answer. It is important to remember that you have a time budget (approx. 100 hours for the course in total) and you need to manage that carefully to produce a balanced portfolio.

For each of the learning outcomes a **quiz will be available that tests basic knowledge**. You can choose to use your performance in the quiz as evidence of knowledge or not as you choose. If you have passed the quiz for a particular Learning Outcome you should not mark yourself below 2 on any sub-criterion in the marking scheme for that Learning Outcome. The quizzes are not compulsory and if you choose not to use them as evidence they are not included in the assessment. Quizzes will be available 08:00 to 20:00 on Thursday of weeks 3,5,7,9, and 11. The quizzes should take about 60 minutes or less to complete.

Outline of the Software Being Tested

This section should have a brief overview of the software being tested. It should also have a link to the repo with your work on the software in a form that is readable by the auditors. If your chosen software is confidential, you do not need to make source code available

Learning Outcomes

This section demonstrated how well you have reached the learning outcomes. Each of the learning outcomes is decomposed into four sub-criteria that are used to structure the assessment. In the marking scheme you are asked to rate your work for that sub-criterion on a scale of 0-5. The marking scheme has guidance on how to interpret these scores.

1. Analyze requirements to determine appropriate testing strategies [default 20%]
    1.1. Range of requirements, functional requirements, measurable quality attributes, qualitative requirements, …
    1.2. Level of requirements, system, integration, unit.
    1.3. Identifying test approach for chosen attributes.
    1.4. Assess the appropriateness of your chosen testing approach.
2. Design and implement comprehensive test plans with instrumented code [default 20%]
    2.1. Construction of the test plan
    2.2. Evaluation of the quality of the test plan
    2.3. Instrumentation of the code
    2.4. Evaluation of the instrumentation

3. Apply a wide variety of testing techniques and compute test coverage and yield according to a variety of criteria [default 20%]
    3.1. Range of techniques
    3.2. Evaluation criteria for the adequacy of the testing
    3.3. Results of testing
    3.4. Evaluation of the results
4. Evaluate the limitations of a given testing process, using statistical methods where appropriate, and summarise outcomes. [default 20%]
    4.1. Identifying gaps and omissions in the testing process
    4.2. Identifying target coverage/performance levels for the different testing procedures
    4.3. Discussing how the testing carried out compares with the target levels
    4.4. Discussion of what would be necessary to achieve the target levels.
5. Conduct reviews, inspections, and design and implement automated testing processes. [default 20%]
    5.1. Identify and apply review criteria to selected parts of the code and identify issues in the code. [default 20%]
    5.2. Construct an appropriate CI pipeline for the software
    5.3. Automate some aspects of the testing
    5.4. Demonstrate the CI pipeline functions as expected.