Assignment 1: Test Preparation and Manual Testing
Testing Techniques 2024 – 2025

In this assignment you will do the test preparation and apply manual, black-box testing to your SUT (System Under Test). Follow the numbered questions and sub-questions provided below in the given order, and write your answers in a report. Your answers shall be well-motivated, yet concise and to the point; use pictures and diagram where appropriate. If you think that some important information about preparation for your testing does not fit under these questions, then you should add it. Please make sure that the text is well readable and consistent throughout. The total length of the core of the report should not exceed 7 pages (appendices with test cases and test results are allowed).

Test Preparation: SUT Description

1. Description of your SUT

   a) what is your SUT;

   b) what functionality does your SUT provide for its users;

   c) external perspective: functionality, what the SUT shall do, its interfaces, the inputs it accepts, the outputs that can be observed, possible and necessary environments, how to start and stop the SUT, . . . , including a picture of the external view;

   d) internal perspective: structure and implementation details as far as they matter for black-box testing, . . . , including a picture of the structural view;

   e) the software/hardware platform(s) on which it is running, required additional software for running it, version number, configuration, and any other specific information about the SUT;

   f) references to documentation about your SUT.

2. Quality aspects that matter for your SUT

   a) mention three quality characteristics that matter for your SUT, and why;

   b) discuss which particular risks are related your SUT (occurrence of failures, impact of failures);

   c) discuss three other quality assessment techniques that could be used for your SUT in addition to black-box testing.

Test Preparation: Test Goal

3. What are you going to test, on a global level?

    a) which part(s) of the SUT, which functionalities, which interfaces;

    b) which kind of properties and behaviour.

4. Give the requirements/specification of your SUT, with emphasis on what you are going to test. Refer to available documentation, where applicable, and give a summary yourself.

Test Preparation: Test Method

5. Provide a *test architecture*:

    a) give a diagram in which you position the SUT, its (distributed) structure, interfaces, environment, stubs, drivers, necessary test tools, . . . ;

    b) describe the SUT input interfaces that you will use, i.e., where and how will your tests will trigger the SUT;

    c) describe the SUT output interfaces, i.e., where and what you will observe during testing;

    d) describe assumptions on the environment of the SUT;

    e) describe which tools, stubs, drivers, . . . , you need for testing, consistent with the diagram that you made above.

6. How will a typical test case look like:

    a) what is a typical structure of your tests: initial state, test inputs, test outputs, observations, conditions, . . . ;

    b) how will you write down your test cases, i.e., the test notation or language to document your test cases.

Test Preparation: Test Implementation

7. Describe how you implemented the *test architecture* described above, including:

    a) concrete implementation of input and output interfaces;

    b) implementation of stubs, drivers, used test tools;

    c) any additional tools or implementation details that are necessary;

    d) are all test interfaces in the architecture accessible?

Manual Testing

Develop and execute manual, black-box functionality test cases for your SUT.

8. What are the domains of test inputs and outputs, over which interfaces are they communicated, and what are valid and invalid inputs?

9. Which test generation techniques will you use for making manual test cases (EP, BVA, state-based, . . .)

10. Develop (at least) 10 black-box functionality test cases to test your SUT manually. Motivate your choice for these test cases, and indicate which test generation technique you used for each test.

11. Test your SUT manually with the developed test cases, or, if real manual testing is impossible for your SUT, use some tool to execute the tests.

12. Analyze and discuss the test results, and draw some conclusions from testing, e.g., considering correctness, completeness of testing, coverage.