Office Solutions Development

Topic11:Testing Software Development

Learning Outcomes - 1

- By the end of this topic, students will be able to:
  - Explain why software testing is needed
  - Discuss the different types of testing
  - Explain why a test plan and report are required
  - Produce a test plan
  - Determine expected test results
  - Record actual test results to enable comparison with expected results

Learning Outcomes - 2

- By the end of this topic, students will be able to:
  - Analyse actual test results against expected results to identify discrepancies
  - Investigate test discrepancies to identify and rectify their causes
  - Produce a testing checklist

Why Testing is Needed - 1

- The purpose of software testing is to ensure that software does exactly what the user needs it to do, ensuring quality.

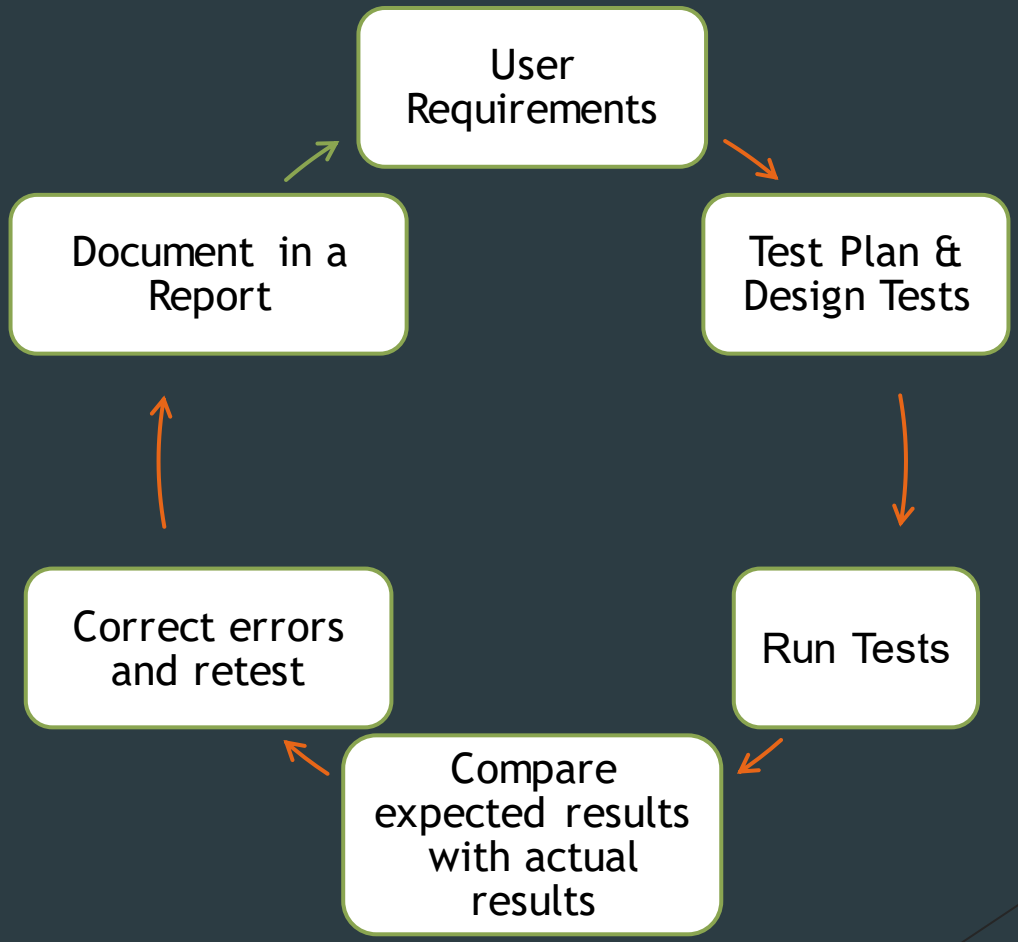
- It can be used to check that software is functioning correctly, reliably, efficiently, securely and is usable.

Why Testing is Needed - 2

- Errors in software as a result of non-testing or inefficient testing can cost a business money, time and reputation.

- It must be planned carefully in stages, such as those in the following diagram:

Stages of Software Testing



Software Quality

- The Standard Glossary of Software Engineering Terminology IEEE610.12 defines software quality as:
  - The degree to which a system, component, or process meets specified requirements.
  - The degree to which a system, component, or process meets customer or user needs or expectations.

Different Types of Testing

- There are many different types of testing and we will focus on the following:
  - Acceptance Testing
  - Black Box Testing
  - Functional Testing
  - Graphical User Interface Testing (GUI)
  - Usabilty Testing

Acceptance Testing

- Also referred to as UAT testing (User Acceptance Testing) and EU (End User Testing).
- It is undertaken by a user/customer to help them decide whether to accept the software as it is or whether enhancements or improvements need to be made.
- It marks the boundary between software being developed for use to software in use.

Black Box Testing

- This type of testing tests that the features and operations of the software function correctly.
- It is concerned with ***what*** the software is supposed to do and not ***how*** it works .
- Technical knowledge of the coding of the software is not required (White Box Testing).
- It is also referred to as ***Functional Testing***.

GUI Testing

- Graphical User Interface testing – refers to testing of all or aspects of graphical user interfaces.
- It can be difficult to undertake as although there are guidelines for good GUI design, a user's judgement of what they consider to be good design can be subjective, e.g. the use of colour, sound, etc.

Usability Testing

- This is undertaken to discover if the software is appropriate to users and how easy it is for them to use and learn. Tests can include the following:

  - Level of use
  - Ease of use
  - Navigation of the software
  - Meaningful instructions
  - Meaningful warnings
  - Consistent style, etc.

Test Plan - 1

- When preparing to test software, a ***Test Plan*** needs to be produced.
- Such plans are very detailed if they are testing whole systems and parts of systems.
- Our test plan will need the following clear structure that sets out:
  - a description of each test
  - the expected results of each test
  - the actual results
  - whether the test passed or failed (include the date)
  - if the test failed, what action needs to be taken
  - whether the re-test passed or failed (include the date)

Test Plan - 2

Test Number	Description	Expected Results	Actual Results	Test Passed (Date)	Action Taken	Test Passed (Date)
1	A test to ensure that a cell is correctly formatted to accept data	The currency amount (2 decimal places) should be displayed in cell D3	The amount was displayed as a whole number, no decimal places	No (21/04/11)	Check the formatting of cell D3 and re-test	Yes (22/04/11)

Expected Test Results

- These are to be clearly set out in the expected results section of the Test Plan.

Actual Test Results Compared with Expected Results

- If the actual test results are different when compared to the expected test results, these details need to be recorded on the Test Plan.
- All the discrepancies must be identified.
- The tests must be re-done and all the discrepancies rectified before each test can be passed.

Test Report

- Test report plans need to be very detailed if they are testing whole systems and parts of systems.
- As a minimum, it should include the following information:
  - the name, version and description of the software program that is being tested
  - the date that the testing is starting
  - the date for completion of all testing
  - who is undertaking the testing and to who they report
  - a description of the type(s) of testing
  - a copy of the Test Plan
  - details of problems

Test Documentation Standards

- For details of the IT industry testing documentation standards, refer to the ***IEEE 229 Standard for Software Test Documentation*** at:

http://tinyurl.com/6dzhhn

A Testing Checklist

- This should check the following:
  - Have all aspects of the software that has been developed been tested?
  - If any tests have failed, have they been redone and passed?
  - Is the Test Plan structured correctly?
  - Is the Test Report structured correctly?