Unit 52: Software Testing

Unit code:	**L/601/3511**
QCF Level 3:	**BTEC Specialist**
Credit value:	**9**
Guided learning hours: 50	

Aim and purpose

This unit develops an understanding of testing strategies and techniques and the management of their application.

Unit introduction

Computer programs will never work if they are not thoroughly and fully tested. In this unit learners will investigate a range of testing strategies and techniques, including white box, black box, static and dynamic. They will practise the stages from planning to acceptance testing and produce software test plans.

Learners will understand how automation can be applied to software testing.

Learners will perform software testing on computer programs by developing and implementing test plans, identifying appropriate test data and recording their results.

Learning outcomes and assessment criteria

In order to pass this unit, the evidence that the learner presents for assessment needs to demonstrate that they can meet all the learning outcomes for the unit. The assessment criteria determine the standard required to achieve the unit.

On completion of this unit a learner should:

Learning outcomes	Assessment criteria	
1 Understand testing strategies and techniques	1.1	explain the purpose and scope of unit, integration and system testing of software
	1.2	describe the stages of system testing including alpha, beta, and acceptance testing
	1.3	describe how automation can be applied to software testing
	1.4	describe and differentiate functional (black box) and structural (white box) testing
2 Manage a test process for a software solution	2.1	develop a test plan including: • test specification (including functional and structural techniques) • test cases • test data and expected results • resources and scheduling • recording and checking of results • evaluation
	2.2	implement a test plan
	2.3	produce a test report

Unit content

---

1   Understanding testing strategies and techniques

Test methods: unit testing eg source code testing; integration testing eg big bang, top down-top up; system testing eg usability, performance, compatibility, error handling, security; black box testing eg test cases based on inputs and expected outputs; white box testing eg data flow, branch, path testing; purpose of each; static testing eg walkthrough without executing code; dynamic testing eg from a debugger environment

Test stages: eg planning, developing test procedures, carrying out tests, reporting (is software ready?), analysis of results, retesting; alpha eg white box testing; beta eg usability testing; acceptance eg black box testing; non-functional testing; performance testing; acceptance testing

Software test plan: contents eg introduction, features to be tested, features not to be tested, item pass/fail criteria, testing tasks, schedule, risks and contingencies; verification (does the software match the customer specification?); validation (does it actually do what the customer wants?); test methods

2   Manage a test process for a software solution

Test process: test specification; test cases; test data; expected results; resources required; time plan; recording documentation; evaluation of results

Test specification: setting minimum criteria for completion; functional eg black box testing, structural eg white box testing

Test cases: expected outputs from specified inputs; formal eg positive, negative testing; informal eg scenario testing

Test data: normal, erroneous, extreme (outside limits)

Record results: test eg branch test, test data, expected result, actual result, corrective action taken

Test report: contents eg test plan, test speciification, test cases, test procedure specification, test log/records, test incident report (actual v expected result)

Essential guidance for tutors

Delivery

This unit could be combined with one of the software development units to give learners a detailed insight into the testing of their own software. Learners could also be given access to a number of sufficiently complex sample programs to practise the various test strategies and techniques as outlined in the unit content or they should be able to watch these tests being demonstrated. Time should be given to ensuring that learners understand the difference between the various forms of testing and their purposes.

Developing comprehensive test plans is an important skill and learners should have the opportunity to practise putting test cases together and developing appropriate recording documentation.

Outline learning plan

The outline learning plan has been included in this unit as guidance and can be used in conjunction with the programme of suggested assignments. The outline learning plan demonstrates one way in planning the delivery and assessment of this unit.

Topic and suggested assignments/activities and/assessment
Introduction to the unit
Testing strategies and techniques   • Test methods – tutor led, demonstrations, examples, practical   • Test stages – tutor led, research, demonstrations   • Test plans – examples, practical.
Assignment 1 - How to test software effectively
Managing a test process   • Test process – tutor led, examples   • Developing test cases – tutor led, examples, practical   • Test data – identifying, examples, practical   • Test records – examples, practical.
Assignment 2 - Testing software

Assessment

It is suggested that this unit is assessed using the two assignments summarised in the *Programme of suggested assignments* table.

Learning outcome 1 assesses the learner's knowledge of testing strategies and techniques. Evidence could be presented in a variety of formats. For example, learners could be asked to produce a wall chart for reference in a test lab describing the stages and forms of testing. Alternatively, learners could produce a self-running presentation giving the same information. Note that for 1.1 the verb is 'explain' which requires more than just a description of each type of testing.

For learning outcome 2, learners will need to be provided with a scenario from which to develop a full software test plan, to include the items outlined in assessment criterion 2.1. There is a practical element to the assessment in 2.2 and 2.3, which requires learners to implement their test plan and produce a test report. Evidence will come from the learner's records of testing, a witness statement and the learner's report.

Programme of suggested assignments

The table below shows a programme of suggested assignments that cover the assessment criteria in the assessment and grading grid. This is for guidance and it is recommended that centres either write their own assignments or adapt any Edexcel assignments to meet local needs and resources.

Criteria covered	Assignment title	Scenario	Assessment method
1.1, 1.2, 1.3, 1.4	How to test software effectively	Produce a presentation (for reference when testing software) showing the stages of testing and their purpose.	Presentation.
2.1, 2.2, 2.3	Testing software	For a given piece of software you are to design a test plan, implement your plan and record the results.	Notes. Witness statements. Test plans. Test records. Test report.

Links to National Occupational Standards, other BTEC units, other BTEC qualifications and other relevant units and qualifications

This unit forms part of the BTEC in IT sector suite. This unit has particular links with:

Level 1	Level 2	Level 3
	Software Testing	Software Design Fundamentals

This unit maps to some of the underpinning knowledge from the following areas of competence in the Level 3 National Occupational Standards for IT (ProCom):

5.2   Software Development

5.3   IT/Technology Solution Testing.

Essential resources

Computer programs should be made available to learners so that they can undertake software testing.

Learners will require access to computer equipment to enable them to gain a practical awareness and enable them to apply their knowledge and understanding in a practical situation.

Employer engagement and vocational contexts

The use of vocational context is essential in the delivery and assessment of this unit.

There is a range of organisations that may be able help centres to engage and involve local employers in the delivery of this unit, for example:

* Learning and Skills Network – www.vocationallearning.org.uk
* Local, regional business links – www.businesslink.gov.uk
* National Education and Business Partnership Network – www.nebpn.org
* Network for Science, Technology, Engineering and Maths Network Ambassadors Scheme – www.stemnet.org.uk
* Work-based learning guidance – www.aimhighersw.ac.uk/wbl.htm
* Work experience/workplace learning frameworks – Centre for Education and Industry (CEI University of Warwick) – www.warwick.ac.uk/wie/cei

Indicative reading for learners

Textbooks

Hambling B – Software Testing: An ISEB Foundation (British Computer Society, 2008) ISBN-10 1902505794, ISBN-13 978-1902505794

Patton R – Software Testing: Second Edition (SAMS, 2005) ISBN-10 0672327988, ISBN-13 978-0672327988

Website

www.ece.cmu.edu/~koopman/des_s99/sw_testing

Functional Skills — Level 2

Skill	When learners are …
ICT - Using ICT	
Plan solutions to complex tasks by analysing the necessary stages	developing and implementing a test plan
Select, interact with and use ICT systems safely and securely for a complex task in non-routine and unfamiliar contexts	developing and implementing a test plan
ICT - Finding and selecting information	
Use appropriate search techniques to locate and select relevant information	describing and differentiating functional and structural testing
Select information from a variety of sources to meet requirements of a complex task	developing and implementing a test plan.