National 4
Added Value
Unit Specification

Computing Science Assignment (National 4)

SCQF: level 4 (6 SCQF credit points)

Unit code: H227 74

Unit outline

This is the Added Value Unit in the National 4 Computing Science Course. The general aim of this Unit is to enable the learner to provide evidence of added value for the National 4 Computing Science Course through the successful completion of an assignment which will allow the learner to demonstrate challenge and application.

Learners who complete this Unit will be able to:

1 Develop, with guidance, a digital solution which will draw on and apply skills and knowledge of software and information system design and development

This Unit is a mandatory Unit of the National 4 Computing Science Course and is also available as a freestanding Unit. The Unit Specification should be read in conjunction with the *Course Support Notes,* which provide advice and guidance on delivery and assessment approaches. Exemplification of the standards in this Unit is given *in Unit Assessment Support*.

Recommended entry

Entry to this Unit is at the discretion of the centre. It is recommended that the learner should be in the process of completing, or have completed, the Units of the National 4 Computing Science Course:

- Software Design and Development (National 4)
- Information System Design and Development (National 4)

Equality and inclusion

This Unit Specification has been designed to ensure that there are no unnecessary barriers to learning or assessment. The individual needs of learners should be taken into account when planning learning experiences, selecting assessment methods or considering alternative evidence. For further information, please refer to the *Course Support Notes.*

Standards

Outcomes and assessment standards

Outcome 1

The learner will:

1 Develop, with guidance, a digital solution which will draw on and apply skills and knowledge of software and information system design and development by:

1.1 Analysing a straightforward problem
1.2 Designing a solution to the problem
1.3 Creating a program or application as a solution to the problem
1.4 Testing and reporting on the solution to the problem

Evidence Requirements for the Unit

This Unit will be assessed through controlled assessment which meets the Evidence Requirements below:

The assessment method for this Unit will be an assignment in which the learner will draw on and apply skills and knowledge of software and information systems design and development.

The assignment will assess learners' skills in analysing a problem, designing and implementing a solution using an appropriate programming or software application, and testing and reporting on that solution.

The assignment is:

- set by centres within the SQA guidelines described below
- conducted under some supervision and control

Evidence will be internally marked by centre staff in line with SQA guidelines.

All assessment is subject to quality assurance by SQA.

Setting the assessment
The assignment will be set by centres within the following guidelines:

- The specification for the assignment will be agreed between the learner and the teacher/lecturer.
- The assignment will be a meaningful and appropriately challenging task, which should clearly demonstrate application of knowledge and skills, at an appropriate level, from both the *Software Design and Development* Unit and the *Information System Design and Development* Unit (as defined in the 'Further mandatory information on Course coverage' section of this document).
- The teacher/lecturer will provide overall guidelines for the assignment and a list of questions/tasks/prompts which will lead learners through the assignment in clear stages.

Conducting the assessment

The assignment will be conducted under some supervision and control. This will take the form of the following:

- The assignment will be carried out under supervised open book conditions.
- The teacher/lecturer may also give learners some support and guidance, as appropriate to National 4 level, to help them progress through each stage of the assignment. The amount of support provided should be reflected in the assessment judgement.

Judging the evidence

Evidence will be internally marked and verified by centre staff in line with SQA guidelines.

All assessment is subject to quality assurance by SQA.

Evidence should include:

- the completed solution
- a record of progress through the assignment (such as an informal electronic log, blog or diary produced by the learner)
- a short report on the solution (in written, electronic and/or oral form)

Re-assessment

In relation to Unit assessment, SQA's guidance on re-assessment for Units applies.

Further information is provided in the exemplification of assessment in *Unit Assessment Support*. Advice and guidance on possible approaches to assessment is provided in the *Course Support Notes.*

Development of skills for learning, skills for life and skills for work

Please refer to the *Course Specification* for information about skills for learning, skills for life and skills for work.

Further mandatory information on Course coverage for the National 4 Computing Science Course

The following gives details of mandatory skills, knowledge and understanding for the National 4 Computing Science Course. Assessment of this Added Value Unit will involve selecting appropriate skills, knowledge and understanding from those listed below, in line with the Evidence Requirements above. This list of skills, knowledge and understanding also provides the basis for the assessment of all the Units in the Course:

Software Design and Development	
Computational constructs	Exemplification and implementation of the following constructs: ♦ expressions to assign values to variables ♦ expressions to return values using arithmetic operations (+, -, *, /, ^) ♦ execution of lines of code in sequence demonstrating input – process – output ♦ use of selection constructs including simple conditional statements ♦ iteration and repetition using fixed and conditional loops
Data types and structures	string numeric (integer) variables graphical objects
Testing and documenting solutions	♦ normal, extreme and exceptional test data ♦ readability of code (internal commentary, meaningful variable names)
Design notations (for both software and information system development)	♦ graphical to illustrate selection and iteration ♦ other contemporary design notations
Low-level operations and computer architecture	Use of binary to represent and store ♦ positive integers ♦ characters ♦ instructions (machine code) Units of storage (bit, byte, Kb, Mb, Gb, Tb, Pb)

NB. The following mandatory generic concepts and vocabulary may be applicable to a range of information systems types and contexts (including databases, websites, games, mobile applications, kiosk systems).

Information Systems Design and Development	
Structures and links (database)	♦ database structure: field, record, file ♦ field types (text, numbers, date, time, graphics, calculated) ♦ database operations (search, sort)
Structures and links (web-based)	♦ website, page, URL ♦ hyperlink
Media types (for both software and information design and	Sound, graphics, video, text

development)	
Purpose, features, functionality and users	Simple descriptions of main features and functionality
Technical implementation (hardware requirements)	♦ input and output devices ♦ processor clock speed (Hz) ♦ memory (RAM, ROM)
Technical implementation (software requirements)	♦ operating system platform required
Technical implementation (storage)	Storage devices ♦ built-in, external, portable ♦ magnetic, optical ♦ capacity, speed ♦ rewritable, read-only
Technical implementation (networking/ connectivity)	♦ stand-alone or networked ♦ LAN/internet ♦ wired/wireless
Security risks	♦ viruses, worms, Trojans ♦ hacking

Administrative information

Published: June 2013 (version 1.1)

Superclass: CB

History of changes

Version	Description of change	Authorised by	Date
1.1	Change to content tables and description of Assignment	Qualifications Development Manager	June 2013