Computing and Information Science (National 4)

Draft Course Support Notes

For general advice and guidance on the Course.

This edition: January 2012, draft version 1.0

Please refer to the note of changes at the end of this document for details of changes from previous version (where applicable).

Contents

Introduction

These support notes are not mandatory. They provide advice and guidance on approaches to delivering and assessing the Computing and Information Science (National 4) Course. They are intended for teachers and lecturers who are delivering the Course and its Units. They should be read in conjunction with the *Course Specification*, the *Added Value Unit Specification*, and the *Unit Specification*s for the Units in the Course.

General guidance on the Course

Aims

As stated in the *Course Specification*, the aims of the Course are to enable learners to:

- develop knowledge and understanding of key facts and ideas in computing and information science
- introduce and develop aspects of computational thinking in a range of contexts
- apply analysis, design, modelling and evaluation to a range of problems
- communicate clearly and concisely, using appropriate terminology
- develop an understanding of the impact of computing and information technology in changing and influencing our environment and society

This Course will also give learners the opportunity to develop thinking skills as well as skills in numeracy, employability, enterprise and citizenship.

Progression into this Course

Entry to this Course is at the discretion of the centre. However, learners would normally be expected to have attained some relevant skills and knowledge through prior experience.

Skills and knowledge developed through any of the following, **while not mandatory**, are likely to be helpful as a basis for further learning for this Course:

- other SQA qualifications
- Access 3 Computing and Information Science Course
- Numeracy (SCQF level 3)

Experiences and outcomes
New National Courses have been designed to draw on and build on the curriculum experiences and outcomes as appropriate. Qualifications developed for the senior phase of secondary education are benchmarked against SCQF levels. SCQF level 4 and the curriculum level 4 are broadly equivalent in terms of level of demand although qualifications at SCQF level 4 will be more specific to allow for more specialist study of subjects.

Learners who have completed Curriculum for Excellence experiences and outcomes will also find these an appropriate basis for doing the Course.

In this Course, any of the following may be relevant.

- I can debate the possible future impact of new and emerging technologies on economic prosperity and the environment. (TCH 4-01c)
- I can approach familiar and new situations with confidence when selecting and using appropriate software to solve increasingly complex problems or issues. (TCH 4-03a)

- I can use ICT effectively in different learning contexts across the curriculum to access, select and present relevant information in a range of tasks. (TCH 4-04a)
- I can work with others to plan and use a learning group for sharing experiences, ideas and information within a secure online environment. (TCH 4-08a)
- I can compare different forms of security software to gain knowledge and understanding of their functions in protecting contemporary technologies. (TCH 4-08b)
- I can integrate different media to create a digital solution which allows interaction and collaboration with others. (TCH 4-08c)
- Through research, I can gain knowledge of computer systems or emerging technologies to understand their differing features and consider their suitability for the world of work. (TCH 4-08d)
- By learning the basic principles of programming language or control technology, I can design a solution to a scenario, implement it and evaluate its success. (TCH 4-09a)

Other experience
Learners may have also relevant skills and knowledge gained through other prior learning, life and work experiences.

Skills, knowledge and understanding covered in this Course

This section provides further advice and guidance about skills, knowledge and understanding that could be included in the Course.

Note: teachers and lecturers should refer to the *Added Value Unit Specification* for mandatory information about the skills, knowledge and understanding to be covered in this Course.

The mandatory skills may be developed throughout the Course. The table below shows where there are significant opportunities to develop these in individual Units.

Mandatory skills and knowledge	**Software Design and Development**	**Information Systems Design and Development**	**Added Value**
Applying, with guidance, aspects of computational thinking across a range of straightforward contexts.	✓	✓	✓
Analysing, with guidance, straightforward problems within computing and information science across a range of contemporary contexts.			✓
Designing, implementing and testing, with guidance, digital solutions (including computer programs) to straightforward problems across a range of contemporary contexts.	✓	✓	

Developing skills in computer programming and the ability to communicate how a program works by being able to read and interpret code.	✓		
Communicating basic understanding of key concepts related to software design and development clearly and concisely using appropriate terminology.	✓		
Communicating basic understanding of key concepts related to information system design and development clearly and concisely using appropriate terminology.		✓	
Basic knowledge of the role and impact of contemporary technologies on the environment and society.		✓	
Applying basic computing and information concepts and techniques to create solutions.			✓

Progression from this Course

This Course or its components may provide progression to:

- Computing and Information Science (National 5)
- National Certificate Group Awards (NCGAs) in computing
- other technological subjects at National 4
- employment, apprenticeships and/or training in IT and related fields

and ultimately, for some, to:

- Higher and Advanced Higher Computing and Information Science
- a range of computing and digital media-related Higher National Certificates (HNCs) and Higher National Diplomas (HNDs)
- degrees in Computing, IT and related disciplines
- careers in computing and information science

Hierarchies

Hierarchy is the term used to describe Courses and Units which form a structured sequence involving two or more SCQF levels.

It is important that any content in a Course and/or Unit at one particular SCQF level is not repeated (unless required for consolidation) if a learner progresses to the next level of the hierarchy. The skills and knowledge should be able to be applied to new content and contexts to enrich the learning experience. This is for centres to manage.

The Course is designed in hierarchy with the corresponding Courses at SCQF levels 5 and 6 (National 5 and Higher). The Computing and Information Science Courses at all three levels have the same structure of two Units with corresponding titles.

The table in Appendix 2 shows the relationship between the mandatory National 4 and National 5 knowledge and understanding. This may be useful for:

- designing and planning learning activities for bi-level classes
- ensuring seamless progression between levels
- identifying important prior learning for learners at National 5

Teachers should also refer to the Outcomes and Assessment Standards for each level when planning delivery.

Further advice on bi-level delivery is given in the next section of these support notes, with additional detailed guidance in the *Unit Support Notes*.

Approaches to learning and teaching

Computing and Information Science, like all new and revised National Courses, has been developed to reflect Curriculum for Excellence values, purposes and principles.

The approach to learning and teaching developed by individual centres should reflect these principles. Learners should be encouraged to participate fully in active learning and practical activities by working together, talking, listening, reading or reflecting on a topic while the teacher acts as a facilitator.

An appropriate balance of teaching methodologies should be used in the delivery of the Course. Whole-class, direct teaching opportunities should be balanced by activity-based learning on practical tasks. An investigatory approach is encouraged, with learners actively involved in developing their skills, knowledge and understanding by investigating a range of real-life and relevant information systems, problems and solutions.

Learning should be supported by appropriate practical activities, so that skills are developed simultaneously with knowledge and understanding. Practical activities and investigations lend themselves to group work, and this should be encouraged. While 'working as a team' is not specifically identified as one of the skills for life, learning and work for this Course and therefore not assessed, it is a fundamental aspect of working in the IT and related industries and so should be encouraged and developed by teachers.

Learning and teaching activities should be designed to develop:

- skills and knowledge to the standard required by the Units and to the level defined by the associated Outcomes and Assessment Standards
- knowledge and understanding as defined by the content statements listed in the *Added Value Unit Specification*, to ensure the required breadth of knowledge is covered

A range of approaches to learning and teaching are appropriate to the delivery of the Course, including the following:

Self-paced learning
In this approach, progression, support and direction are embedded in carefully written notes, and the teacher's role is primarily that of providing learners with support.

Didactic teacher-led approaches

- exposition is particularly useful when introducing key concepts such as selection or iteration
- demonstration, eg of key aspects of a software environment, such as the user interface
- lock step: in which groups of learners are taken through a complex process by a series of instructions issued by the teacher

Flexible grouping strategies

- using group work, eg to discuss approaches to problems
- providing individual support, particularly when pupils are engaged in practical tasks

Problem-based learning (PBL) strategies

- useful at the end of an Outcome or a topic to consolidate knowledge and understanding
- provides additional challenge in applying knowledge and skills in less familiar contexts
- offers potential for group or individual tasks

Research strategies

- effective use of internet search engines
- collation and storage of research materials
- evaluation of research materials

ICT-based strategies

- use of multimedia presentation and web-based software to support teacher-led approaches
- provision of interactive multimedia learning resources
- use of interactive quizzes

Involvement with outside agencies

- to provide guest speakers
- to provide information and contexts from the world of work

Formative assessment
The use of formative assessment should be a natural part of all learning activities, continuing from practice in the broad general education, with assessment activities blended with learning activities throughout the Course.

Formative assessment should be used to support learning by:

- sharing learning intentions/success criteria
- using assessment information to set learning targets and next steps
- adapting teaching and learning activities based on assessment information
- boosting learners' confidence by providing supportive feedback

Self- and peer-assessment techniques should be encouraged wherever appropriate.

Working towards Units and Course
Learning and teaching activities should be designed to develop both:

- skills and knowledge to the standard required by each Unit and to the level defined by the associated Outcomes and Assessment Standards

- ability to apply the breadth of knowledge and understanding listed in the *Course Assessment Specification*, as required to complete the Course assessment successfully

Catering for individual needs
Every class of learners will have individual strengths and weaknesses.

Formative assessment strategies allow teachers to cater for individual needs by supporting and building on each individual learner's knowledge, skills and confidence and identifying their next steps.

It is also possible to support individual learning needs by providing a range of differentiated materials that have an embedded series of examples and problem solving tasks which progress from the simple to the complex, from the familiar to the unfamiliar.

Guidance on delivery of bi-level teaching
Within a National 4 class, there may be learners capable of achieving National 5 standards in some aspects of the Course. Where possible, they should be given the opportunity to do so.

Teachers need to consider both the Outcomes and Assessment Standards, and the tables of concepts in the previous section of these notes, to identify the differences between National 4 and National 5.

In some aspects of the Course, the difference between National 4 and National 5 is defined in terms of a higher level of skill. For example, in *Information Systems Design and Development*, Outcome 1 requires National 4 learners to develop an information system using appropriate development tools, while at National 5 learners are also required to write or edit simple code. In other aspects of the Course, the difference between National 4 and National 5 is defined by additional or more in-depth knowledge. For example, in *Software Design and Development*, National 4 learners need to use one software development environment, while National 5 learners need to compare two.

Where a class is formally a bi-level class, with some learners working towards National 4 and others towards National 5, a core and extension approach may assist the teacher to plan activities, with core activities covering National 4 for all learners, and extensions for National 5 learners. This is particularly appropriate where the National 5 learners have come direct from the broad general education without previously studying National 4.

However, where National 5 learners have studied National 4 in a previous year, it is important to provide them with new and different contexts for learning to avoid de-motivation.

Sequence of delivery
The decision of how to deliver the Units is a matter of professional judgement and is at the discretion of the centre.

Units could be delivered in sequence or in parallel. One approach might be to alternate delivery of the Units' content to give variety to teaching and learning activities, eg programming alternating with application development, but other possibilities exist.

Whether the decision is taken to deliver Units sequentially or in parallel, it would be good practice to complete the two other Units before attempting the Added Value Unit. This approach will give learners the opportunity to develop the skills and knowledge to enable them to successfully attempt the added value assignment. However, it may be possible to begin work on the Added Value Unit at an earlier stage, but only where it is clear that learners have already gained the required skills and knowledge.

Advice on distribution of time
The distribution of time between the Units is a matter for professional judgement and is entirely at the discretion the centre. Each Unit is likely to require an approximately equal time allocation, although this may depend on the learners' prior learning in the different topic areas.

Suggested allocation of total available Course time is as follows:

Software Design and Development	40%
Information Systems Design and Development	40%
Computing and Information Science Assignment	20%

Within the framework for completing the Added Value Unit (*Computing and Information Science Assignment*) time will be required for:

- preparation for the assignment, which could include considering exemplar assignments and practising the application and integration of skills
- carrying out the stages of the assignment, with teacher guidance and support
- assessing the process and completed solution
- providing opportunities for re-assessment if required
- completing a record of progress

Developing skills for learning, skills for life and skills for work

Guidance on the development of skills for life, skills for learning and skills for work, is to be found in the support notes for each of the component Units.

Approaches to assessment

To gain the award of the Course, the learner must pass all the Units. All Units are internally assessed against the requirements shown in each *Unit Specification*, with further details supplied in the individual Unit Support Notes.

Units are assessed on a pass/fail basis within centres. SQA will provide rigorous external quality assurance, including external verification, to ensure assessment judgments are consistent and meet national standards.

See the Unit Support Notes for guidance on approaches to assessment of the Units of the Course.

Added value

Courses from National 4 to Advanced Higher include assessment of added value.

At National 4 the added value will be assessed in the Added Value Unit.

Information given in the *Course Specification* about the assessment of added value is mandatory. Full details of assessment of the assignment are included in the *Added Value Unit Specification.*

The Added Value Unit will assess the application of skills and knowledge which learners will have developed through the other Units. Evidence for this Unit will be generated through an assignment in which learners will be required to solve an appropriately challenging computing and information science problem.

Combining assessment across Units

If an integrated approach to Course delivery is chosen, then there may be opportunities for combining assessment across Units.

E-assessment

E-assessment can play an important role in the design and delivery of National Courses and Units by supporting integration and learners' personalisation and choice. While it is important not to introduce new, additional ICT skills or knowledge, it may be that learners may be using ICT in working towards their assessment.

In this Course, learners will be constantly engaged in using ICT as they work towards their Unit and Course assessment by demonstrating their skills in using software to create a range of digital solutions, such as database and web-based information systems.

The knowledge and understanding elements of the Course lend themselves to being assessed through e-assessment via an e-portfolio or similar. Online or electronic tests can take the form of short answer, multiple choice or true/false. One advantage of this approach is that feedback can be almost instantaneous and reduces the time required by the teacher/lecturer to mark the completed work.

Learners will engage with web-based research to gather information on their Outcome 2 topics within the Units and will have the choice to deliver this information in a variety of formats, including digital ones such as electronic presentations, web pages, video, podcasts, blogs, etc.

Equality and inclusion

The following is a list of the aspects of the qualification identified as potentially having an adverse impact on learners on grounds of one or more of the protected characteristics:

- practical work in implementing web-based information systems and multimedia information systems
- using media types such as graphics, video and animation in the development of these systems

The potential adverse impact of the particular aspect of the qualification on any learner on grounds of one or more of the protected characteristics is as follows:

- some visually impaired learners may be unable to edit and manipulate graphics, videos and animations due to limitations in vision

The following assessment arrangements could remove the adverse impact:

- partially sighted learners may be assisted by technologies such as screen magnifiers or larger screens

It is recognised that centres have their own duties under equality and other legislation and policy initiatives. The guidance given in these *Course Support Notes* is designed to sit alongside these duties but is specific to the delivery and assessment of the Course.

It is important that centres are aware of and understand SQA's assessment arrangements for disabled learners, and those with additional support needs, when making requests for adjustments to published assessment arrangements. centres will find more guidance on this in the series of publications on Assessment Arrangements on SQA's website: www.sqa.org.uk/sqa/14977.html.

Appendix 1: Reference documents

The following reference documents will provide useful information and background.

- Assessment Arrangements (for disabled candidates and/or those with additional support needs) — various publications are available on SQA's website at: www.sqa.org.uk/sqa//14977.html.
- *Building the Curriculum 4: Skills for learning, skills for life and skills for work*
- *Building the Curriculum 5: A framework for assessment*
- *Course Specifications*
- Design Principles for National Courses
- *Guide to Assessment* (June 2008)
- Overview of Qualification Reports
- Principles and practice papers for curriculum areas
- *SCQF Handbook: User Guide* (published 2009) and SCQF level descriptors (to be reviewed during 2011 to 2012): www.sqa.org.uk/sqa/4595.html
- *SQA Skills Framework: Skills for Learning, Skills for Life and Skills for Work*
- *Skills for Learning, Skills for Life and Skills for Work: Using the Curriculum Tool*

Appendix 2: Comparison of National 4 and National 5

This table shows the relationship between the mandatory National 4 and National 5 knowledge and understanding.

This table may be useful for:

- designing and planning learning activities for bi-level National 4/National 5 classes
- ensuring seamless progression between levels
- identifying important prior learning for learners at National 5

Teachers should also refer to the Outcomes and Assessment Standards for each level when planning delivery.

Software Design and Development		
Topic	**National 4**	**National 5**
Exemplification and implementation of:		
Programming constructs	The following constructs in a programming development environment: ♦ 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 (IF) ♦ iteration using fixed and conditional loops	The following constructs in programming development environment: ♦ expressions to assign values to variables ♦ expressions to return values using arithmetic operations (+, -, *, /, ^) ♦ use of selection constructs including simple and complex conditional statements (IF) and logical operators (AND, OR, NOT) ♦ pre-defined functions
Data types	String ♦ numeric (integer) ♦ variables	String ♦ numeric (integer and real) ♦ Boolean variables ♦ 1D arrays
Testing and documenting solutions	♦ Testing digital solutions using supplied normal, extreme and exceptional test data	♦ Testing digital solutions using own normal, extreme and exceptional test data ♦ Syntax, execution and logic errors in programs ♦ Readability of code (internal commentary, meaningful identifiers, indentation)

Design notations	Contemporary design notations: ♦ graphical to illustrate selection and iteration	Contemporary design notations: ♦ pseudocode to exemplify programming constructs
Low-level operation and structure	Use of binary to represent and store ♦ positive integers ♦ characters ♦ instructions (machine code) — units of storage (bit, byte, Kb, Mb, Gb, Tb)	Translation of high-level program code to binary (machine code): interpreters and compilers Use of binary to represent and store ♦ integers and real numbers ♦ characters ♦ instructions (machine code) ♦ graphics (bit-mapped and vector) Basic computer architecture: processor, memory, buses, interfaces

Information Systems Design and Development		
Topic	**National 4**	**National 5**
The following mandatory generic concepts and vocabulary may be applicable to a range of information systems types		
Structures and links	♦ database structure: field, record, file ♦ field types (text, numbers, date, time, graphics, calculated) ♦ web site, page, URL ♦ hyperlink	♦ database structure: flat file, linked tables ♦ field types (text, numbers, date, time, graphics, calculated, link, Boolean) ♦ field lengths and range ♦ key field ♦ website, page, URL ♦ hyperlink (internal, external), navigation ♦ frames, sequence
User interface	♦ target audience	User requirements (visual layout, navigation, selection, consistency, interactivity, readability)
Media types	Sound, graphics, video, text	Standard file formats: ♦ text: txt, rtf ♦ audio: wav, mp3 ♦ graphics: jpeg, bmp, gif, png ♦ video: mpeg, avi ♦ pdf — factors affecting file size and quality, including resolution, colour depth, sampling rate — need for compression

Coding	Macros	♦ scripting languages (including javascript) ♦ mark-up languages (including HTML)
Testing		♦ links and navigation
Note: where similar concepts are used at more than one level, the Outcomes distinguish the level of treatment.		
Purpose, features, functionality, users	Simple descriptions of main features and functionality	Description of purpose Users: expert, novice, age-range
Technical implementation (hardware and software requirements)	♦ input and output devices ♦ processor clock speed (Hz) ♦ memory (RAM, ROM)	♦ input and output devices ♦ processor type and speed (Hz) ♦ memory (RAM, ROM) ♦ operating system ♦ device type (including supercomputer, desktop, laptop, tablet, smartphone)
Technical implementation (storage)	♦ storage devices ♦ built-in, external, portable ♦ magnetic, optical ♦ capacity, speed ♦ rewritable, read-only	♦ storage devices ♦ built-in, external, portable ♦ magnetic, optical ♦ capacity, speed ♦ rewritable, read-only ♦ interface type and speed
Technical implementation (networking/ connectivity)	♦ stand-alone or networked ♦ LAN/internet ♦ wired/wireless	♦ peer-to-peer versus client/server ♦ media: wired, optical, wireless ♦ comparison of local versus web/cloud
Security risks	♦ viruses, worms, trojans	♦ spyware, phishing, keylogger ♦ online fraud, identity theft, pharming ♦ hacking
Security precautions	Anti-virus software	♦ passwords/encryption ♦ biometrics ♦ security protocols and firewalls ♦ use of security suites
Legal implications		♦ computer misuse act ♦ data protection act ♦ health and safety regulations
Environmental implications		♦ energy use ♦ disposal of IT equipment

Administrative information

Published: January 2012 (draft version 1.0)

Superclass: to be advised

History of changes to Course Support Notes

Course details	Version	Description of change	Authorised by	Date

Draft Unit Support Notes — Software Design and Development (National 4)

For general advice and guidance on the Unit.

This edition: January 2012, draft version 1.0

Please refer to the note of changes at the end of this document for details of changes from previous version (where applicable).

Introduction

These support notes are not mandatory. They provide advice and guidance on approaches to delivering and assessing the Software Design and Development (National 4) Unit. They are intended for teachers and lecturers who are delivering this Unit. They should be read in conjunction with:

- the *Unit Specification*
- the *Course Specification*
- the *Added Value Unit Specification*
- the *Course Support Notes*
- appropriate assessment support materials

If the *Unit Support Notes* have been developed for a Unit which is not part of a Course, then it is only necessary to read them in conjunction with the *Unit Specification.*

General guidance on the Unit

Aims

The purpose of this Unit, as stated in the *Unit Specification*, is to develop, in the learner, basic knowledge, understanding and practical problem solving skills in software development through appropriate software development environments. Learners will develop programming skills by implementing practical solutions and explaining how these programs work. They will also develop an understanding of how data and instructions are stored in binary form, how programming underpins all computer applications, and an awareness of some of the implications of software on society and the environment.

This Unit will give learners the opportunity to develop their thinking skills as well as their skills in numeracy, employability, enterprise and citizenship.

This Unit can be delivered:

- as a standalone Unit
- as a component of the Computing and Information Science (National 4) Course

Progression into this Unit

Entry to this Unit is at the discretion of the centre. However, learners would normally be expected to have attained some useful skills and knowledge from prior learning, such as:

- Access 3 Computing and Information Science or relevant component Units
- Numeracy (SCQF level 3)

Learners may also have gained relevant skills and knowledge through other education systems or from their own interests and informal learning.

Skills, knowledge and understanding covered in this Unit

Information about skills, knowledge and understanding is given in the Computing and Information Systems (National 4) *Course Support Notes*.

If the Unit is being delivered as part of the Computing and Information Science (National 4) Course, the teacher should refer to the 'Further mandatory information on Course coverage' section in the *Added Value Unit Specification* for detailed content.

If this Unit is being delivered on a free-standing basis, teachers and lecturers are free to select the skills, knowledge, understanding and contexts which are most appropriate for delivery in their centres.

Progression from this Unit

On successful completion of this Unit, the following Units and Courses provide a selection of progression pathways available to learners:

- Software Design and Development (National 5) Unit
- Computing and Information Science (National 5) Course
- National Certificate Group Awards (NCGAs) in a range of computing and IT disciplines
- National Progression Awards (NPAs) in Digital Media
- other technological subjects at National 4
- employment, apprenticeships and/or training in computing and related fields

Approaches to learning, teaching and assessment

Learning and teaching
The Unit is designed to provide flexibility, personalisation and choice for both the learner and the teacher.

Learning and teaching activities should be designed to stimulate learners' interest, and to develop skills and knowledge to the standard required by the three Outcomes and to the level defined by the associated Assessment Standards. Learning should be supported by appropriate practical activities so that skills are developed simultaneously with knowledge and understanding.

An investigatory approach is encouraged, with learners actively involved in developing their skills, knowledge and understanding by investigating a range of real-life and relevant software development environments, problems and solutions.

Another strategy might be to introduce problem-solving opportunities by setting tasks in a progression of unfamiliar contexts. This could be further extended by increasing the complexity of the problem presented to the learner.

Teacher-led approaches can be useful when introducing key concepts in areas such as programming, ie when demonstrating using a combination of expressions, sequence, selection and iteration, or when introducing aspects of a software environment.

Practical activities and investigations lend themselves readily to group work, and this should be encouraged. Individual, paired or group problem-solving tasks could be related to authentic and relevant contexts.

The Computing and Information Science (National 4) *Course Support Notes* provide further broad guidance on approaches to learning and teaching which may apply to all three of the component Units of the Course and should be read before delivering this Unit.

When delivering the Unit as part of the Computing and Information Science (National 4) Course, reference should be made to the appropriate content statements within the *Added Value Unit Specification* to ensure the required breadth of knowledge is covered.

Sequence of delivery of Outcomes
The sequence of delivery of the Outcomes is a matter of professional judgement and is entirely at the discretion of the centre. Some suggested approaches are listed below.

Outcome 1 and Outcome 2 simultaneously
Teachers may consider it good practice to combine the development of knowledge and understanding of how programs work at the same time as learners progress through the practical tasks involved in developing their own short programs.

For example, when learners are selecting and using appropriate constructs and variables when developing their programs, they will need to understand the purpose and function of these constructs and the purpose and roles of variables. As learners develop their programming skills by selecting and using appropriate constructs and assigning values to variables, they will also develop a clear understanding of how these constructs and variables work and what they can be used for. This will enable them to read, interpret and explain the code in their programs.

Outcome 1 before Outcome 2
It is possible to explain how simple software development programs work, and meet the requirements of Outcome 1 before carrying out the practical task of developing short programs in Outcome 2.

For example, learners could gain an understanding of the basic concepts involved in software development before creating their own programs. They could select different programs and then find out the following:

- What is the purpose of the programming constructs?
- How do they work?
- How is data stored?
- How are instructions stored?
- How can the underlying code be read and explained?

Once the learners have gained a sound understanding of the purpose of a range of programming constructs, how they function and how data and instructions are stored within a program, they should be well placed to develop their own simple programs in their chosen software development environment, such as games, mobile application or text-based. They should be able to apply their knowledge gained in Outcome 1 to select and use the appropriate programming constructs to produce a complete, working program.

Outcome 2 before Outcome 1
It is possible to develop the learner's practical skills in developing short programs and meet the requirements of Outcome 1 before moving to formally develop the knowledge and understanding associated with explaining how programs work in Outcome 2.

For example, learners would develop their programming skills in a software development environment by learning the functions of certain programming constructs and using a range of simple data types. They would be coding solutions to problems using these constructs and data types and building up valuable experience in programming. These activities would allow the learners to fully understand the purpose and function of certain constructs and data types and how data and instructions are stored within programs. Consequently, when learners are then asked to read and explain sections of code or determine the value of a variable within a loop construct within a program in Outcome 1, they will have the necessary knowledge and skills to do this from their experience in coding in Outcome 2.

Outcome 3
It would seem to be good practice to meet the requirements of Outcomes 1 and 2, thus equipping learners with useful background knowledge and skills which will be useful in addressing Outcome 3, but again, this is a matter of professional judgement and is entirely at the discretion of the centre.

For example, learners will have experienced a software development environment and will be familiar with its main features in Outcomes 1 and 2. It would therefore seem sensible to expose learners to this first before going on to research a contemporary software-based application.

Guidance on bi-level teaching
When delivering this Unit to a bi-level class, with some learners working towards National 4 and others towards National 5, a core and extension approach may assist the teacher to plan activities, with core activities covering National 4 for all learners, and extensions for National 5 learners.

For example, when National 4 and National 5 learners are designing their programs, both sets of learners will be taught the principles and importance of design, but National 4 learners will use a graphical notation to illustrate selection and iteration whereas the National 5 learners will use pseudocode to exemplify programming constructs.

When learners are developing their skills and knowledge in computational constructs, both National 4 and Nation 5 learners will undertake tasks where they have to:

- use expressions to assign values to variables
- use expressions to return values using arithmetic operations
- execute lines of code in sequence demonstrating input – process – output
- use IF statements
- use fixed and conditional loops

The National 5 learners will then be given extension tasks where they will have to use logical operators and pre-defined functions.

For data types and structures, both National 4 and National 5 learners will cover string and numeric variables but National 5 learners will be given extension exercises covering Boolean variables and 1-D arrays.

When it comes to testing and documenting solutions, both sets of learners will test their solutions with normal, extreme and exceptional test data. The National 5 learners will then document their programs using internal commentary, meaningful identifiers and indentation.

In line with the underlying principles of Curriculum for Excellence, learners should be encouraged, and expected, to take an active role in their own learning. Where Course activities and materials allow them to progress in an independent manner, this will allow teaching of the two groups to happen most effectively.

Useful resources
Online resources (websites, microsites, wikis, newsfeeds, databases, etc) can provide a valuable source of easily accessible and up-to-date information on a wide range of software development tools and topics. In addition, the internet can act as a rich source of information for research into the impact of a software-based application on the environment or society.

Some suggested online resources:

- Technology Student — www.technologystudent.com
- Teaching Education Scotland — www.tes.co.uk/teaching-resources/
- STEM — www.ltscotland.org.uk/stemcentral
- Khan Academy — www.khanacademy.com
- Scottish Government — www.scotland.gov.uk
- How Stuff Works — http://computer.howstuffworks.com/
- Scratch — http://scratch.mit.edu/
- Scratch projects — http://scratch.mit.edu/channel/featured
- Kodu — http://www.kodugamelab.com/
- BYOB (Build Your Own Blocks) — http://byob.berkeley.edu/
- Drawplus tutorials — www.serif.com/appresources/DPX5/Tutorials/en-gb/DrawPlus.htm
- Dynamic Learning — www.dynamiclearning.co.uk
- MonkeyJam PDF tutorial — www.brickshelf.com/gallery/brenden17/Tutorials/monkeyjam.pdf
- You Tube — www.youtube.com
- MonkeyJam Tutorial — www.youtube.com/watch?v=omZ1wUoNf88
- Trackmania tutorial — www.youtube.com/watch?v=vB3MFOUCYhs
- MIT Centre for Mobile Learning — http://mitmobilelearning.org/

Unit assessment
The learner must demonstrate attainment of all of the Outcomes and their associated Assessment Standards. Assessment must be valid, reliable and fit for purpose.

SQA does not specify the methods of assessment to be used; teachers should determine the most appropriate method for their learners. In many cases, evidence (which may be oral or observational) will be gathered during normal classroom activities, rather than through formal assessment instruments. Centres are expected to maintain a detailed record of evidence, including oral or observational evidence. Evidence in written or presentation format should be retained by the centre.

Authentication of evidence
All evidence should be gathered under supervised conditions.

In order to ensure that the learner's work is their own, the following strategies are recommended:

- personal interviews with learners where teachers can ask additional questions about the completed work
- asking learners to do an oral presentation on their work
- ensuring learners are clear about acknowledging sources
- using checklists to record the authentication activity

Approaches to delivering and assessing each Outcome

Preparing learners for their assessment activities
In order to ensure that learners are prepared in advance for their assessment activities, it is good practice for teachers to make learners aware of the Assessment Standards required and to provide a range of feedback designed to improve learners' knowledge and skills as they progress through the Unit.

It is accepted as good practice that the evidence to meet the Assessment Standards for Outcome 1 and Outcome 2 be generated throughout the Course as an integral part of classroom activities.

Outcome 1

The learner will:

1 Explain how simple programs work, drawing on understanding of basic concepts in software development, by:

1.1 Reading and explaining code
1.2 Describing the purpose of a range of programming constructs and how they work
1.3 Explaining how data and instructions are stored

The range of programming constructs should include expressions, sequence, selection and iteration.

Notes on delivery of Outcome 1
In order to meet Outcome 1, learners are expected to develop their knowledge and understanding of how programs work to the point where they have the ability to describe the purpose of the following constructs and to describe how they work:

- expressions to assign values to numeric and string variables
- expressions to return values using arithmetic operations (+, -, *, /, ^)
- sequence to determine the order in which statements are executed demonstrating input – process – output
- use of selection constructs, including simple conditional statements (IF)
- iteration using fixed and conditional loops

Learners should also develop the ability to read and interpret code by:

- identifying errors and suggesting corrections in a given piece of code
- interpreting and identifying the input, process and output in a given piece of code

Learners also need to be able to explain how data and instructions are stored:

- how assigning values to declared variables allows the value to be stored and used again within the program

Opportunities for different learning and teaching activities might include the following:

- Learners could be provided with completed working programs in a software development environment in order to be taught the purpose of a range of programming constructs, how they work and how data and instructions are stored.
- Teachers and lecturers could go over these constructs and how values are assigned to variables and then demonstrate how they work inside a program. This will develop the knowledge and understanding required in the learner to be able to read and explain sections of code within a program.

Notes on assessment of Outcome 1
Evidence for Outcome 1 can be generated throughout the Unit as an integral part of classroom activities.

Learners will use understanding of basic concepts in software development to explain how programs work. Written evidence may take the form of responses to a series of assignments or a short test in which the learner demonstrates the ability to:

- read and interpret code
- describe the purpose of a range of programming constructs
- describe how these constructs work
- explain how data and instructions are stored

A valid and reliable form of assessment would, for example, present the learner with a section of code which contains, eg, examples of selection and the use of string and numeric variables. It would then require the learner to:

- describe the purpose of the selection constructs in the coding
- describe how those constructs work
- explain how the data and instructions are stored

The learner would be expected to produce an accurate series of descriptions in the form of extended responses.

Evidence of the Assessment Standards for Outcome 1 may be oral or written.

Evidence in written form should be retained by the centre.

Where learners' responses have been in oral form, centres are expected to keep either a recording of candidates' performance as evidence and/or an observation checklist.

Authentication
All evidence should be gathered in supervised conditions.

Note that the coding presented to the learner should include a representative sample of the programming constructs and variable types set out in the mandatory content.

Outcome 2

The learner will:

2 Develop short programs using a software development environment by:

2.1 Selecting and using expressions, sequence, selection and iteration
2.2 Selecting and using appropriate simple data types, such as numeric (integer) and string
2.3 Testing digital solutions using supplied test data
2.4 Identifying and rectifying errors in programs

Programs should include at least one construct and one data type.

Notes on delivery of Outcome 2
To meet Outcome 2, learners are expected to gain experience of using different software development environments, although they may produce evidence from only one if they so wish.

The advantage of using more than one development environment is that it enables comparison between (say) the operation of constructs in two or more environments, providing opportunities to deepen the learner's understanding. Using just one development environment, however, might secure deeper development of knowledge and skills.

Software development environments to choose from might include:

- graphical environments in which code is assembled by combining graphical objects which represent instructions, variables and constructs
- software development environments which are specifically suited to games development
- apps development environments which are suited to producing applications
- software development environments for handheld devices and smartphones
- text-based programming environments

Some of the software development environments available have a more extensive range of programming features than others. A software development environment with a narrowly restricted range of features would not be suitable for meeting the demands of this Unit.

Below is a non-restrictive list of possible examples of software development environments which might be suitable. The choice of software development environment is entirely at the discretion of the centre and should be based on the suitability of the chosen environment to support the delivery of the mandatory content of the Unit.

Non-restrictive examples of current software development environments
Graphical environments in which code is assembled by combining graphical objects which represent instructions, variables and constructs, such as Scratch
Software development environments which are specifically suited to games development, such as C++, Gamemaker, Greenfoot and Dark Basic
Apps development environments which are suited to producing applications for handheld devices and smartphones, such as Java development tools
Text-based programming environments, such as Visual Basic and Truebasic

Developing short programs
Learners should develop short programs using the following constructs in a software development environment:

- 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 (IF)
- iteration using fixed and conditional loops. Conditional loops should contain simple conditions

Level of complexity of programs
In this National 4 Unit it is expected that learners' programs should contain at least one data type, either string or numeric where appropriate, and at least one construct.

Testing digital solutions
Learners should develop skills in testing their programs and learn to use supplied sets of test data. They should:

- test their programs using normal, extreme and exceptional test data which is supplied by the teacher
- identify and rectify errors in programs

When testing programs
The test data which is provided should be composed of three types:

- **normal:** data that would fall within the range of data a program is expected to process
- **extreme:** data that tests the boundaries of the range of data a program is expected to process
- **exceptional:** data that is outside the range of data a program is expected to process

Using the test data and any error reporting features of their development environments, learners should develop the skills required to identify and rectify errors in programs.

Notes on assessment of Outcome 2
Evidence is only required of one successful example for each Assessment Standard and can be generated throughout the Unit as an integral part of classroom activities, using a single problem solving task or by a series of separate activities.

Learners will develop a number of short programs in a software development environment and evidence can be gathered from any of these throughout the duration of the Unit.

- Learners can decide which type of software development environment to work in.
- Learners can have access to books, the internet and other materials during the assessment.
- Learners should receive accurate and regular feedback from teachers and lecturers and be actively involved in the assessment process.

The short programs should be completed and each should include both of the following:

- at least one construct
- one data type

Teachers and lecturers will select the appropriate evidence from the learner on the following aspects:

- The solutions to the tasks set should be straightforward and, for example, not involve complex combinations of selection and iteration.
- Any conditions which are part of selection or iteration constructs are expected to be simple in structure and not to require the use of logical operators (AND, OR, NOT).

Learners should:

- use supplied test data which contains normal, extreme and exceptional data
- identify and correct errors

Authentication
Users should produce their programs in supervised conditions.

Evidence of the Assessment Standards for Outcome 2 may be derived from several software development tasks; formal documentation is not expected or required. However, an observation checklist should be maintained by the teacher to record learners' performance.

While formal documentation is not required, it is recognised good practice to keep digital copies of all learners' work in order to support the learning process by acting as a source of reference and revision activities, particularly if the Unit is being taken as part of the Computing and Information Science (National 4) course.

Outcome 3

The learner will:

3 Produce a short factual report on a contemporary software-based application by:

3.1 Describing the application
3.2 Explaining how its features relate to programming constructs and data types
3.3 Describing its impact on the environment or society

Notes on the delivery of Outcome 3
The learner should investigate some 'contemporary software-based applications' to develop an understanding of the fact that software and programs drive all computer applications. The applications investigated could include commercial office applications, games, educational software, multimedia tools or any other software. For example, learners may want to research some of the contemporary app development applications, such as Xcode for Apple devices or Android for HTC devices. This is a clear opportunity for the learner, in consultation with their teacher, to exercise a degree of choice in the content of their learning.

The learner should relate aspects of the software to the constructs and data types they have been using for Outcomes 1 and 2. For example, in a game environment, they could relate menu systems to selection constructs (eg if then…), and realise that their game identity will be stored as a string, while their high score will be a numeric variable.

This task will involve a degree of research and will provide opportunities for the learner to develop skills related to the focused and effective use of internet search engines as well as the collation, evaluation and organisation of research material.

Notes on assessment of Outcome 3
To meet Outcome 3, learners are required to produce a short factual report on a contemporary software-based application.

- Learners can have access to books, the internet and other materials during the assessment.
- Learners should receive accurate and regular feedback from teachers and lecturers and be actively involved in the assessment process.

Choice of format
There is no set format for the report. It may be produced, for example, as a text document, a visual presentation, a video clip, a website or any other appropriate format. The format of the report is at the discretion of the centre and is an opportunity for learners to exercise an element of choice. The decision on the selected format is likely to emerge as the result of discussion between staff and learner.

Authentication
The process of producing the report which constitutes evidence of the Assessment Standards for Outcome 3 will necessarily involve a degree of research which may be carried out by the learner in the centre or at home. Centres should ensure that report is the work of the candidate, for example by asking the candidate questions about the content of the completed report or to give an oral presentation of their work.

Length of the report
There is no set length for the report but centres may be guided by the suggested range of activities required to complete the work. This might include:

- discussion of proposals
- planning including setting timescales and deciding on sources of information
- research activities
- drafting/revising the report
- final submission of the report

Developing skills for learning, skills for life and skills for work

Learners are expected to develop broad generic skills as an integral part of their learning experience. The *Unit Specification* lists the skills for learning, skills for life and skills for work that learners should develop through this Course. These are based on SQA's Skills Framework: Skills for Learning, Skills for Life and Skills for Work and must be built into the Unit where there are appropriate opportunities. The level of these skills will be appropriate to the level of the Unit.

The table below highlights opportunities to develop these skills during this Unit.

2 Numeracy	
2.1 Number processing	Learners can be given opportunities to develop their number processing skills by gaining practice in problem solving in numeric based contexts which involve eg multiplication, division or calculating percentages. Problem solving contexts could then be set in which software would take decisions and vary the output based on the results of calculations.
2.3 Information handling	Information handling skills could be developed by setting problem solving contexts in which learners are required to use data set out in tables or a graphical format as the basis for input to their programs which then process the data to produce required output.

4 Employability, enterprise and citizenship	
4.2 Information and communication technology	Throughout the Unit learners will be continuously interacting with the technology around them and will be given abundant opportunities to extend their ICT skills. When producing their report for Outcome 3, they will be given opportunities to specifically develop their ICT based research, evaluation and decision making skills

♦

5 Thinking skills	
5.3 Applying	Learners will be given opportunities to apply their software-based knowledge and skills and then test their solutions.

Added value

Courses from National 4 to Advanced Higher include assessment of added value. At National 4 the added value will be assessed in the Added Value Unit.

Information given in the *Added Value Unit Specification* about the assessment of added value is mandatory.

If this Unit is being taken as part of the Computing and Information Science (National 4) Course then the knowledge and understanding as well as the skills developed within this Unit will contribute towards the assessment of the Added Value of the Course.

In the Computing and Information Science (National 4) Course, added value will focus on challenge and application.

The learner will be required to draw on, extend and apply the skills and knowledge they have developed during this Unit as well as the *Information Systems Design and Development* (National 4) Unit. These will be assessed through an assignment which involves the application of skills and knowledge

from both Units to solve an appropriately challenging computing and information science problem.

It is therefore important to emphasise that if this Unit is being taken as part of the Computing and Information Science (National 4) Course then all of the mandatory content specified in the *Added Value Unit Specification* should be covered in the course of this Unit.

Combining assessment within Units

Holistic assessments could be developed which cover some of the Assessment Standards from Outcomes 1 and 2.

For example: an assessment could be devised that asks the learner to create a program with at least one construct, at least one data type and some variables. Once they have created the program, tested it, and rectified any errors, they could then be asked to describe the purpose of the constructs and how they work. The assessor could then select a segment of code from the program and ask the learner to explain what the code actually does. The assessor could also ask the learner to explain how data and instructions are stored.

This would cover the following Assessment Standards:

1.1 Reading and explaining code
1.2 Describing the purpose of a range of programming constructs and how they work
1.3 Explaining how data and instructions are stored

2.1 Selecting and using expressions, sequence, selection and iteration
2.2 Selecting and using appropriate simple data types such as numeric (integer) and string
2.3 Testing digital solutions using supplied test data
2.4 Identifying and rectifying errors in programs

E-assessment

E-assessment can play an important role in the design and delivery of National Courses and Units by supporting integration and learners' personalisation and choice. While it is important not to introduce new, additional ICT skills or knowledge, learners may make use of ICT while working towards their assessment.

In this Unit, learners will be constantly engaged in using ICT as they work towards their assessment by demonstrating their skills in using software to develop short programs. They will also develop skills in programming and testing digital solutions.

Learners will also use web-based research to gather information on specific topics, such as the environmental or social impact of a software-based application. They will be able to deliver this information in a variety of digital formats, including presentations, web pages, video, podcasts and blogs.

Equality and inclusion

The following is a list of the aspects of the qualification identified as potentially having an adverse impact on learners on grounds of one or more of the protected characteristics:

- practical work in completing software development and programming tasks
- using media types such as graphics, video and animation in the development of these systems

The potential adverse impact of the particular aspect of the qualification on any learner on grounds of one or more of the protected characteristics is as follows:

- some visually impaired learners unable to edit and manipulate graphics, videos and animations due to limitations in vision

The following assessment arrangements could remove the adverse impact:

- partially sighted learners may be assisted by technologies such as screen magnifiers or larger screens

It is recognised that centres have their own duties under equality and other legislation and policy initiatives. The guidance given in these *Unit Support Notes* is designed to sit alongside these duties but is specific to the delivery and assessment of the Unit.

Alternative approaches to Unit assessment to take account of the specific needs of learners can be used. However, the centre must be satisfied that the integrity of the assessment is maintained and that the alternative approach to assessment will, in fact, generate the necessary evidence of achievement.

Appendix 1: Reference documents

The following reference documents will provide useful information and background.

- Assessment Arrangements (for disabled candidates and/or those with additional support needs) — various publications on SQA's website: http://www.sqa.org.uk/sqa/14976.html
- *Building the Curriculum 4: Skills for learning, skills for life and skills for work*
- *Building the Curriculum 5: A framework for assessment*
- *Course Specifications*
- Design Principles for National Courses
- *Guide to Assessment* (June 2008)
- Overview of Qualification Reports
- Overview of Qualification Reports
- Principles and practice papers for curriculum areas
- Research Report 4 — Less is More: Good Practice in Reducing Assessment Time
- Coursework Authenticity — a Guide for Teachers and Lecturers
- *SCQF Handbook: User Guide (published 2009)* and SCQF level descriptors (to be reviewed during 2011 to 2012): www.sqa.org.uk/sqa/4595.html
- *SQA Skills Framework: Skills for Learning, Skills for Life and Skills for Work*
- *Skills for Learning, Skills for Life and Skills for Work: Using the Curriculum Tool*
- SQA Guidelines on e-assessment for Schools
- SQA Guidelines on Online Assessment for Further Education
- SQA e-assessment web page: www.sqa.org.uk/sqa/5606.html

Administrative information

Published: January 2012 (draft version 1.0)

Superclass: to be advised

History of changes to Unit Support Notes

Unit details	Version	Description of change	Authorised by	Date

Draft Unit Support Notes — Information Systems Design and Development (National 4)

For general advice and guidance on the Unit.

This edition: January 2012, draft version 1.0

Please refer to the note of changes at the end of this document for details of changes from previous version (where applicable).

Introduction

These support notes are not mandatory. They provide advice and guidance on approaches to delivering and assessing the *Information Systems Design and Development* (National 4) Unit. They are intended for teachers and lecturers who are delivering this Unit. They should be read in conjunction with:

- the *Unit Specification*
- the Computing and Information Science (National 4) *Course Specification*
- the Computing and Information Science (National 4) *Course Assessment Specification*
- the Computing and Information Science (National 4) *Course Support Notes*
- appropriate assessment support materials

General guidance on the Unit

Aims

The purpose of this Unit, as stated in the *Unit Specification*, is to develop in the learner basic knowledge, understanding and practical problem solving skills related to the design and development of information systems. Learners will implement practical solutions using a range of development tools. They will also develop an understanding of hardware, software, connectivity and security issues, and relate these to one or more information systems.

This Unit will give learners the opportunity to develop their thinking skills as well as their skills in numeracy, employability, enterprise and citizenship.

This Unit can be delivered:

- as a stand-alone Unit
- as a component of the Computing and Information Science (National 4) Course

Progression into this Unit

Entry to this Unit is at the discretion of the centre. However, learners would normally be expected to have attained some useful skills and knowledge from prior learning, such as:

- Access 3 Computing and Information Science Course or relevant component Units
- Numeracy (SCQF level 3)

Learners may also have gained relevant skills and knowledge through other education systems or from their own interests and informal learning.

Skills, knowledge and understanding covered in this Unit

Information about skills, knowledge and understanding is given in the (Computing and Information Systems (National 4) *Course Support Notes.*

If the Unit is being delivered as part of the Computing and Information Science (National 4) Course, the teacher should refer to the 'Further mandatory information on Course coverage' section within the *Added Value Unit Specification* for detailed content.

If this Unit is being delivered on a free-standing basis, teachers and lecturers are free to select the skills, knowledge, understanding and contexts which are most appropriate for delivery in their centres.

Progression from this Unit

On successful completion of this Unit, the following Units and Courses provide a selection of progression pathways available to learners:

- Information System Design and Development (National 5) Unit
- Computing and Information Science (National 5) Course
- National Certificate Group Awards (NCGAs) in a range of computing and IT disciplines
- National Progression Awards (NPAs) in Digital Media
- other technological subjects at National 4
- employment, apprenticeships and/or training in computing and related fields

Approaches to learning, teaching and assessment

Learning and teaching
The Unit is designed to provide flexibility, personalisation and choice for both the learner and the teacher.

Learning and teaching activities should be designed to stimulate learners' interest, and to develop skills and knowledge to the standard required by the two Outcomes and to the level defined by the associated Assessment Standards. Learning should be supported by appropriate practical activities so that skills are developed simultaneously with knowledge and understanding.

An investigatory approach is encouraged, with learners actively involved in developing their skills, knowledge and understanding by investigating a range of real-life and relevant information systems, problems and solutions.

Practical activities and investigations lend themselves readily to group work, and this should be encouraged. Individual, paired or group problem-solving tasks could be related to authentic and relevant contexts. For example, learners could be asked to:

- create a database structure to hold information such as a CD or MP3 collection
- create a website that appeals to the viewer with appropriate use of text, graphics, sound and video similar to the glossy magazine websites
- work in pairs or groups to research the different types of user interface for information systems aimed at different users
- work in pairs or groups to compare the security risks of particular types of information system, eg office information systems, business intelligence systems, transaction processing systems, management reporting systems

The Computing and Information Science (National 4) *Course Support Notes* provide further broad guidance on approaches to learning and teaching which may apply to both of the component Units of the Course and should be read before delivering this Unit.

When delivering the Unit as part of the Computing and Information Science (National 4) Course, reference should be made to the appropriate content statements within the 'Further mandatory information on Course coverage' section to ensure the required breadth of knowledge is covered

Sequence of delivery of Outcomes
The sequence of delivery of the Outcomes is a matter of professional judgement and is entirely at the discretion of the centre. Some suggested approaches are listed below.

Outcome 1 and Outcome 2 simultaneously
Teachers may consider it good practice to combine the development of knowledge and understanding of the factors involved in the design and implementation of an information system at the same time as learners progress through the practical tasks involved in developing simple information systems.

For example, when learners are designing their information systems, whether databases, multimedia applications, websites, and/or hybrids of these, they will have to consider the functionality and features of the products they are designing. Learners may experiment with different types of software applications to create their products and they should be made aware of the hardware and software requirements for these different types of applications. Although learners will probably use software that is available to them on the network, they should also have an idea of the type of similar applications that are available from cloud systems. File sizes and types of storage media should be considered when learners are storing their products on the network, which highlights the importance of storage and capacity for any information system.

Outcome 1 before Outcome 2
It is possible to develop the learner's practical skills in creating information systems and meet the requirements of Outcome 1 before moving to formally develop the knowledge and understanding associated with the factors involved in the design and implementation of an information system in Outcome 2.

For example, learners could develop the practical skills associated with creating information systems by working with databases, websites, multimedia applications and/or hybrids of these. This would allow them to become familiar with the features of different software applications used to build these systems; how to integrate different media types and how to identify and rectify errors and bugs that occur during implementation. Following this process could give them a solid understanding of the main factors involved in developing a 'real-life' information system and the practical skills required to actually create it. By investigating the technical specification and the security risks of a 'real-life' information system, learners should be able to relate the factors and concepts to the smaller-scale information systems that they have created.

Learners should be encouraged to make connections between the processes they went through in developing their own information systems and the factors that have to be considered in the design and implementation of information systems used in the real world.

Outcome 2 before Outcome 1
It is also possible to consider the factors involved in the design and implementation of an information system and meet the requirements of Outcome 2 before actually carrying out the practical task of creating information systems.

Outcome 1
For example, learners could develop an understanding of the basic factors involved in the design and implementation of a 'real-life' information system before developing their own. They could select a particular information system that is used in their school, a local company or business and then find out the following:

- What are its main features?
- What functionality does it offer?
- What are the hardware, software and connectivity requirements?
- What are the main security risks with the system?

With this overview in mind, learners could then proceed to create their own simple information systems, such as databases, websites, multimedia applications and/or a hybrid of these.

By doing Outcome 2 first, learners should be able to relate the basic issues and factors they discovered when researching a 'real-life' information system to their own products, albeit on a much smaller scale. Learners could then develop their practical skills by discovering the functionality of the software they are using to create structures with links and integrating different media types. They might look at the issues related to the importance of storage requirements in real-life information systems.

Guidance on bi-level teaching
When delivering this Unit to a bi-level class, with some learners working towards National 4 and others towards National 5, a core and extension approach may assist the teacher to plan activities, with core activities covering National 4 for all learners, and extensions for National 5 learners.

For example, both National 4 and National 5 learners will be creating information systems, such as databases, websites, multimedia applications or hybrids of these, but the National 5 learners will have different specifications. Both sets of learners can decide on the type of information systems they are going to create and the types of software they are going to use to build the systems. Teacher-directed lessons or self-supported materials could be used to guide the learner through the main functions of selected software with extension exercises built in for the National 5 learners dealing with such aspects as linked tables; link and Boolean data types; field lengths and range checks; key fields; navigation and frames.

Core lessons in the importance of the design of a user interface could be delivered to both sets of learners, with the National 5 learners studying additional criteria for good user interface, such as visual layout, navigation, selection, consistency, interactivity and readability.

Both sets of learners will need to integrate text, graphics, sound and video into their information systems, but National 5 learners should be made aware of the properties of standard file formats, factors affecting file size and quality, and the need for compression.

Both sets of learners should learn the importance of customising their information systems, although at National 4 this will take the form of implementing macros, whereas at National 5, learners should learn the basic constructs of a scripting language such as JavaScript and a mark-up language such as HTML.
National 4 and National 5 learners should both study the technical specifications of real-life information systems covering the same aspects in terms of processor type, speed, memory (RAM and ROM), input and output devices; however, the National 5 learners are also required to consider security precautions, legal aspects and impact on the environment.

On the topic of networking and connectivity, National 4 learners might compare stand-alone computers with networked computers, LAN with the internet and wired with wireless connections. National 5 learners should also be comparing networks but they would be looking at peer-to-peer versus client/server and LAN versus web/cloud systems.

Where National 5 learners have studied National 4 in a previous year, it is important to provide them with new and different contexts for learning to avoid de-motivation. It is particularly important that learners do not feel that they are simply doing the same work over again, albeit at a deeper level.

For example, at National 4, learners will have built simple information systems, such as databases, websites, multimedia applications and/or hybrids of these. At National 5, they will be able to add more functionality to one or more products or build new systems using the techniques and skills they have acquired throughout the National 5 Course. They will be able to include internal and external hyperlinks and navigation panes or frames. Learners may have focused mainly on (say) multimedia and web applications at National 4, and then could spend more time studying (say) database systems at National 5.

National 5 learners should consider specific aspects of user interface design while at National 4 learners only need to be aware of the target audience of a system; when studying media types, National 5 learners should focus on the properties of the standard media file formats and how they affect file size.

At National 4, learners will have been introduced to coding using macros. At National 5, they could continue to develop their coding skills in scripting and in the use of a mark-up language.

In line with the underlying principles of Curriculum for Excellence, learners should be encouraged, and expected, to take an active role in their own learning. Where Course activities and materials allow them to progress in an independent manner, this will allow teaching of the two groups to happen most effectively.

Useful resources
Online resources (websites, microsites, wikis, newsfeeds, databases, etc) can provide a valuable source of easily accessible and up-to-date information on a wide range of information systems design and develop topics. In addition, the internet can act as a rich source of information for research into the legal implications and environmental impact of IT systems.

Some suggested online resources:

- Technology Student — www.technologystudent.com
- Teaching Education Scotland — www.tes.co.uk/teaching-resources/
- STEM — www.ltscotland.org.uk/stemcentral
- Khan Academy — www.khanacademy.com
- Scottish Government — www.scotland.gov.uk
- How Stuff Works — http://computer.howstuffworks.com/
- Hackasaurus — http://hackasaurus.org/en-US/

Unit assessment
The learner must demonstrate attainment of **all** of the Outcomes and their associated Assessment Standards. Assessment must be valid, reliable and fit for purpose.

SQA does not specify the methods of assessment to be used; teachers should determine the most appropriate method for their learners. In many cases, evidence (which may be oral or observational) will be gathered during normal classroom activities, rather than through formal assessment instruments.

Centres are expected to maintain a detailed record of evidence, including oral or observational evidence. Evidence in written or presentation format should be retained by the centre.

Authentication of evidence
All evidence should be gathered under supervised conditions.

In order to ensure that the learner's work is their own, the following strategies are recommended:

- personal interviews with learners where teachers can ask additional questions about the completed work
- asking learners to do an oral presentation on their work
- ensuring learners are clear about acknowledging sources
- using checklists to record the authentication activity

Approaches to delivering and assessing each Outcome

Preparing learners for their assessment activities
In order to ensure that learners are prepared in advance for their assessment activities, it is good practice for teachers to make learners aware of the Assessment Standards required and to provide a range of feedback designed to improve learners' knowledge and skills as they progress through the Unit.

It is accepted as good practice that the evidence to meet the Assessment Standards for Outcome 1 and Outcome 2 be generated throughout the Course as an integral part of classroom activities.

Outcome 1

The learner will:

4 Develop information systems using appropriate development tools by:

4.1 Creating a structure and links
4.2 Integrating different media types
4.3 Identifying and rectifying errors

Notes on delivery of Outcome 1
It is envisaged that learners will develop a number of different information systems, which might include databases, web pages, multimedia products, web-based applications or other current information systems. Developing a range of these would allow the learner to develop the broad knowledge and understanding required for Course assessment. Tasks and activities throughout the Unit should be linked to relevant contexts.

Learners could work in pairs or groups to consider and discuss the basic features and functionality of a good information system from both the operator's and end-user's perspectives.

Learners should be given instruction or resources to enable them to use software applications to incorporate and integrate different media types into an information system. They should also develop skills in using macros. Practical tutorials or teacher-led instruction may be required to teach the basics, followed by a series of tasks or exercises to reinforce learners' understanding. Learners should be

encouraged to identify any errors and seek solutions during the development of their information systems.

Notes on assessment of Outcome 1
Evidence is only required of one successful example for each assessment standard and can be generated throughout the Unit as an integral part of classroom activities, using a single problem solving task or by a series of separate activities.

- Learners can decide which type of information systems to develop, eg databases, a series of web pages, multimedia products, web-based applications or other current information systems.
- Learners can have access to books, the internet and other materials during assessment activities.
- Learners should receive accurate and regular feedback from teachers and lecturers and be actively involved in the assessment process.

Teachers will select the appropriate evidence from the learner. Evidence can be gathered from observation; however, centres should keep a record, either in digital format or as an observational checklist, recording, video, etc, for verifying purposes.

All evidence should be gathered under supervised conditions.

Outcome 2

The learner will:

5 Consider a number of basic factors involved in the design and implementation of an information system by describing in simple terms:

5.1 Its basic features and functionality
5.2 Its hardware, software and connectivity requirements
5.3 The security risks involved in digital communications

Notes on delivery of Outcome 2
It is envisaged that learners will investigate a real-life information system (this might be an office information system, a business intelligent system, a transaction processing system, management reporting system, a website or other), and by focusing on the basic factors that should be considered in the design and implementation of this information system they will meet the Assessment Standards for Outcome 2.

Teachers and lecturers can provide some background information relating to aspects of the information system to be investigated. Learners have to demonstrate that they have a detailed understanding of the underlying concepts of this information system, including hardware and software, connectivity and the security risks involved in digital communication.

Opportunities for different learning and teaching activities might include:

- comparing different input and output devices, such as desktops, laptops, tablet PCs and smartphones in terms of processor speed and memory (RAM, ROM

- utilising websites such as Dell, PC World, etc, to compare the range of different storage devices available
- researching the news online to look for instances when security risks have occurred through digital communication modes

Outcome 2 lends itself to more pair and/or group work. Aspects of existing information systems solutions to real-world problems can be analysed in simple terms to aid understanding, and a number of different types of 'real-life' information systems should be provided to the learners. The tasks and activities here will be more of the research and investigation type, with learners working in pairs or groups.

For example, each group could be asked to analyse the features of the information system in their school or college to obtain an overview of the many tasks that it performs.

Alternatively, each group could be given an information system to investigate and report on, under the following headings:

- features and functionality
- hardware, software and connectivity
- security risks

Notes on assessment of Outcome 2
Evidence for Outcome 2 can be generated throughout the Unit as an integral part of classroom activities.

- Learners can decide on the information system to investigate.
- Learners can have access to books, the internet and other materials during the assessment.
- Learners should receive accurate and regular feedback from teachers and lecturers and be actively involved in the assessment process.

The learner would be expected to produce an accurate series of descriptions in the form of extended responses. However, the evidence for Outcome 2 need not be written, but may be presented in another format, such as a text document, a series of web pages, a visual presentation, a video, a podcast, a blog or any other appropriate format.

Teachers and lecturers will select the appropriate evidence from the learner on the following aspects of the information system:

- statement relating to features and functionality of the information system
- description of the hardware, software and connectivity required to run the information system
- list of the security risks involved in digital communication to operator and end-users

Where a learner's responses have been in oral form, centres should keep either a recording of a candidate's performance as evidence and/or an observation checklist.

All evidence should be gathered in supervised conditions.

Developing skills for learning, skills for life and skills for work

Learners are expected to develop broad generic skills as an integral part of their learning experience. The *Unit Specification* lists the skills for learning, skills for life and skills for work that learners should develop through this Course. These are based on SQA's *Skills Framework: Skills for Learning, Skills for Life and Skills for Work* and must be built into the Unit where there are appropriate opportunities. The level of these skills will be appropriate to the level of the Unit.

The table below highlights opportunities to develop these skills during this Unit.

2 Numeracy	
2.1 Number processing	Learners can be given opportunities to develop their number processing skills by gaining practice in problem solving in numeric-based contexts, eg calculation of main memory, storage requirements, and then making informed decisions based on the results of these calculations.
2.3 Information handling	Information handling skills could be developed by setting problem solving contexts in which learners are required to interpret data in different structures, eg fields, records and files in databases, URLs and hyperlinks in web pages, and comparison of current input/output devices using the criteria of speed and memory.

4 Employability, enterprise and citizenship	
4.2 Information and communication technology	Throughout the Unit, learners will be continuously interacting with the technology around them and will be given abundant opportunities to extend their ICT skills. When investigating information systems for Outcome 2, they will be given opportunities to specifically develop their ICT-based research skills.

5 Thinking skills	
5.3 Applying	Learners will be given ample opportunities to analyse a wide range of problems, apply the knowledge and skills they have acquired and then test their solutions.

Added value

Courses from National 4 to Advanced Higher include assessment of added value. At National 4 the added value will be assessed in the Added Value Unit.

Information given in the *Added Value Unit Specification* about the assessment of added value is mandatory.

If this Unit is being taken as part of the Computing and Information Science (National 4) Course then the knowledge and understanding as well as the skills developed within this Unit will contribute towards the assessment of the added value of the Course.

In the Computing and Information Science (National 4) Course, added value will focus on challenge and application.
The learner will be required to draw on, extend and apply the skills and knowledge they have developed during this Unit as well as during the *Software Design and Development* (National 4) Unit. These will be assessed through an assignment which involves the application of skills and knowledge from both Units to solve an appropriately challenging computing and information science problem.

It is therefore important to emphasise that if this Unit is being taken as part of the Computing and Information Science (National 4) Course then all of the mandatory content specified in the *Added Value Unit Specification* should be covered in the course of this Unit.

Combining assessment within Units

See *Course Support Notes*.

E-assessment

E-assessment can play an important role in the design and delivery of National Courses and Units by supporting integration and learners' personalisation and choice. While it is important not to introduce new, additional ICT skills or knowledge, learners may make use of ICT while working towards their assessment.

In this Unit, learners will be constantly engaged in using ICT as they work towards their assessment by demonstrating their skills in using software to create a range of digital solutions, such as databases and web-based information systems. They will also develop skills in creating digital structures with links, integrating different media types, and testing them for errors.

They will also use web-based research to gather information on specific topics, such as legal implications and environmental impact. Learners will be able to deliver this information in a variety of digital formats, including presentations, web pages, video, podcasts and blogs.

Equality and inclusion

The following is a list the aspects of the qualification identified as potentially having an adverse impact on learners on grounds of one or more of the protected characteristics:

- practical work in implementing web-based information systems and multimedia information systems
- using media types such as graphics, video and animation in the development of these systems

The potential adverse impact of the particular aspect of the qualification on any learner on grounds of one or more of the protected characteristics is as follows:

- some visually impaired learners unable to edit and manipulate graphics, videos and animations due to limitations in vision

The following assessment arrangements could remove the adverse impact:

- partially sighted learners may be assisted by technologies such as screen magnifiers or larger screens

It is recognised that centres have their own duties under equality and other legislation and policy initiatives. The guidance given in these *Unit Support Notes* is designed to sit alongside these duties but is specific to the delivery and assessment of the Unit.

Alternative approaches to Unit assessment to take account of the specific needs of learners can be used. However, the centre must be satisfied that the integrity of the assessment is maintained and that the alternative approach to assessment will, in fact, generate the necessary evidence of achievement.

Appendix 1: Reference documents

The following reference documents will provide useful information and background.

- Assessment Arrangements (for disabled candidates and/or those with additional support needs) — various publications on SQA's website: http://www.sqa.org.uk/sqa/14976.html
- *Building the Curriculum 4: Skills for learning, skills for life and skills for work*
- *Building the Curriculum 5: A framework for assessment*
- *Course Specifications*
- Design Principles for National Courses
- *Guide to Assessment* (June 2008)
- Overview of Qualification Reports
- Overview of Qualification Reports
- Principles and practice papers for curriculum areas
- Research Report 4 — Less is More: Good Practice in Reducing Assessment Time
- Coursework Authenticity — a Guide for Teachers and Lecturers
- *SCQF Handbook: User Guide (published 2009)* and SCQF level descriptors (to be reviewed during 2011 to 2012): www.sqa.org.uk/sqa/4595.html
- *SQA Skills Framework: Skills for Learning, Skills for Life and Skills for Work*
- *Skills for Learning, Skills for Life and Skills for Work: Using the Curriculum Tool*
- SQA Guidelines on e-assessment for Schools
- SQA Guidelines on Online Assessment for Further Education
- SQA e-assessment web page: www.sqa.org.uk/sqa/5606.html

Administrative information

Published: January 2012 (draft version 1.0)

Superclass: to be advised

History of changes to Unit Support Notes

Unit details	Version	Description of change	Authorised by	Date