

# Next Generation Higher National Unit Specification

## Software Development (SCQF level 7)

**Unit code:** J68V 47

**SCQF level:** 7 (16 SCQF credit points)

**Valid from:** session 2022–23

**Prototype unit specification for use in pilot delivery  
only (version 1.0) May 2022**

This unit specification provides detailed information about the unit to ensure consistent and transparent assessment year on year.

This unit specification is for teachers and lecturers and contains all the mandatory information required to deliver and assess the unit.

The information in this unit specification may be reproduced in support of SQA qualifications only on a non-commercial basis. If it is reproduced, SQA must be clearly acknowledged as the source. If it is to be reproduced for any other purpose, written permission must be obtained from [permissions@sqa.org.uk](mailto:permissions@sqa.org.uk).

This edition: May 2022 (version 1)

© Scottish Qualifications Authority 2022

## Unit purpose

This unit introduces learners to the principles and practice of software development. It is a non-specialist unit, intended for learners studying a Higher National Qualification in Computing or a related field. It may also be appropriate for learners with an interest in any STEM subject. No previous knowledge or experience of programming is required.

The unit covers software development concepts by allowing learners to build, test and deploy a software application. It examines the differences between software development life cycle models, and learners understand the importance of each development stage in producing robust applications. The unit explores practical programming concepts and gives learners hands-on experience of building software applications following a suitable design solution. It examines different testing strategies and highlights the importance of testing, while developing software applications and preparing them for deployment.

This unit focuses on learners developing software individually. Developing software as part of a team is the focus for the Software Development unit at SCQF level 8.

On completion of this unit, learners gain knowledge and skills using various software development methods, techniques and tools at an intermediate level, and show they can apply them by analysing, designing, implementing, testing and deploying a software solution. This provides them with a good foundation to progress to the Software Development unit at SCQF level 8.

## Unit outcomes

Learners who complete this unit can:

- 1 explain the software development life cycle
- 2 build a software application from a design
- 3 test a software application
- 4 deploy a software application

## Evidence requirements

Learners must provide product evidence.

The product evidence relates to outcomes 1, 2, 3 and 4. Competence in outcome 1 is inferred from the product evidence.

The product evidence must demonstrate that learners can:

- ◆ write a requirements specification
- ◆ design a software solution
- ◆ implement the solution in code
- ◆ carry out testing
- ◆ document the solution
- ◆ deploy the software

They must create at least one complete, working, error-free program. The problem should be non-trivial, with the focus on the software development life cycle rather than the complexity of code. The code must be sufficiently complex to illustrate data structures, control structures, modular programming, parameter passing, program I/O and software libraries.

Learners must produce both maintenance and user documentation.

They can produce evidence over an extended period of time, in lightly controlled conditions, however, authentication is required.

The SCQF level of this unit provides additional context relating to the quality of evidence required.

## Knowledge and skills

The following table shows the knowledge and skills covered by the unit outcomes:

Knowledge	Skills
<p>Learners should understand:</p> <ul style="list-style-type: none"> <li>◆ the software development life cycle</li> <li>◆ the software development method</li> <li>◆ deliverables for each stage in the software development life cycle</li> <li>◆ good programming practice</li> <li>◆ software security</li> <li>◆ requirements analysis</li> <li>◆ software development tools and environments</li> <li>◆ software design, including algorithms</li> <li>◆ the syntax and semantics of a programming language</li> <li>◆ programming techniques, including structured programming</li> <li>◆ program control structures</li> <li>◆ parameter passing between modules</li> <li>◆ basic data structures</li> <li>◆ input/output file operations</li> <li>◆ testing strategies and methods used in software development</li> <li>◆ types of errors in software development</li> <li>◆ error handling</li> <li>◆ software documentation</li> <li>◆ software deployment process</li> </ul>	<p>Learners can:</p> <ul style="list-style-type: none"> <li>◆ create requirements specification</li> <li>◆ write algorithms</li> <li>◆ desk-check algorithms</li> <li>◆ create design documents</li> <li>◆ write code following a design</li> <li>◆ apply structured programming</li> <li>◆ use modular code</li> <li>◆ use arithmetic, logic and Boolean operators</li> <li>◆ use control constructs</li> <li>◆ use parameter passing to transfer values between modules</li> <li>◆ use standard libraries</li> <li>◆ use array data structures</li> <li>◆ use input/output file operations</li> <li>◆ use error handling techniques</li> <li>◆ create a test plan using a defined strategy</li> <li>◆ perform software testing</li> <li>◆ create test logs</li> <li>◆ create error logs</li> <li>◆ write maintenance documentation</li> <li>◆ write user documentation</li> <li>◆ deploy a software application</li> </ul>

## Meta-skills

Throughout the unit, learners develop meta-skills to enhance their employability in the computing sector.

### Self-management

This meta-skill includes:

- ◆ focusing: attention
- ◆ adapting: critical reflection, self-learning
- ◆ initiative: independent thinking, self-motivation, responsibility

### Social intelligence

This meta-skill includes:

- ◆ communicating: receiving information, listening, giving information
- ◆ leading: change catalyst

### Innovation

This meta-skill includes:

- ◆ curiosity: information sourcing, problem recognition
- ◆ creativity: imagination, visualising, maker mentality
- ◆ sense-making: pattern recognition, holistic thinking, analysis
- ◆ critical thinking: deconstruction, logical thinking, judgement, computational thinking

## Literacies

### Numeracy

Learners can develop numeracy skills through coding tasks. This allows them to develop small applications such as profit loss calculators, conversion calculators or mathematical programs.

### Communication

Learners can develop communication skills through a range of activities, including the production of documentation and various design tools.

### Digital

Learners can develop digital skills and computer literacy throughout the duration and stages of the project.

## **Delivery of unit**

We suggested the following distribution of time:

**Outcome 1** — Explain the software development life cycle  
(10 hours)

**Outcome 2** — Build a software application from a design  
(40 hours)

**Outcome 3** — Test a software application  
(20 hours)

**Outcome 4** — Deploy a software application  
(10 hours)

## **Additional guidance**

The guidance in this section is not mandatory.

The general context of this unit is to introduce learners to software development. When you address complex technological topics, you should use high-level terms and concepts, and teach terminology in a hands-on and problem-solving context throughout the unit.

## **Assessment approaches**

We recommended that you use a programming assignment for learners to produce the required evidence (see 'Evidence requirements' section). Alternatively, you could maintain a portfolio of code, produced over the life of the unit. If you use this approach, you must take care to ensure that learners satisfy all the evidence requirements.

## **Equality and inclusion**

This unit is designed to be as fair and as accessible as possible with no unnecessary barriers to learning or assessment.

You should take into account the needs of individual learners when planning learning experiences, selecting assessment methods or considering alternative evidence.

Guidance on assessment arrangements for disabled learners and/or those with additional support needs is available on the assessment arrangements web page:

[www.sqa.org.uk/assessmentarrangements](http://www.sqa.org.uk/assessmentarrangements).



## Information for learners

### Software Development (SCQF level 7)

This section explains:

- ◆ what the unit is about
- ◆ what you should know or be able to do before you start
- ◆ what you need to do during the unit
- ◆ opportunities for further learning and employment

### Unit information

This unit introduces you to the principles and practices of software development. It is a non-specialist unit that needs little or no previous knowledge or experience. It is particularly suitable if you are studying an HND in Computing or a related subject. However, it may be appropriate for other qualifications that require a general knowledge of software development.

This unit gives you hands-on experience in using software development concepts, by allowing you to design, build, test and deploy software solutions. You learn about various software development life cycle models and the importance of each software development stage in producing robust software. It explores practical programming concepts and gives you hands-on experience of building software applications following a suitable design solution. You learn to use different testing strategies and methodologies to ensure that your applications run error-free and are ready for deployment. Although the unit is intended to be primarily practical, it does contain a theoretical section. You are introduced to concepts and terminologies using real-world examples of software development solutions.

The assessment involves designing, writing, testing, documenting and deploying a software solution.

During the unit, there are many opportunities to develop or enhance your self-management, social intelligence, and innovative skills.

# Administrative information

---

**Published:** May 2022 (version 1.0)

**Superclass:** CB

---

## History of changes

Version	Description of change	Date

Note: please check [SQA's website](#) to ensure you are using the most up-to-date version of this document.