SCOTTISH
QUALIFICATIONS
AUTHORITY

Higher National Unit Specification

General information for centres

Unit title: Software Development: Fourth Generation Environments

Unit code: D76S 35

Unit purpose: This Unit is designed to expose the candidate to the study of the features of a relational database management system (RDBMS) and to develop a critical understanding of the capabilities of such systems. In addition, the candidate will develop database design and implementation skills in a fourth generation environment. This Unit is aimed at candidates who have completed the first year of the HND Computing: Software Development.

On completion of this Unit, the candidate should be able to:

1. Demonstrate an understanding of the features and uses of an RDBMS and fourth generation language environment
2. Use a data definition language (DDL) and a data manipulation language (DML) to define and interrogate tables
3. Apply a systematic approach to designing applications/systems interfaces
4. Evaluate the design and functionality of an RDB application

Credit value: 2 HN credits at SCQF level 8: (16 SCQF credit points at SCQF level 8)

SCQF (the Scottish Credit and Qualifications Framework) brings Scottish qualifications into a single framework of 12 levels ranging from SQA Access 1 to doctorates. The SCQF includes degrees; HNC/Ds; SQA National Qualifications; and SVQs. Each SQA Unit is allocated a number of SCQF credit points at a specific level. 1 SCQF point = 10 hours of learning. HN candidates are normally expected to input a further number of hours, matched to the credit value of the Unit, of non-contact time or candidate-led effort to consolidate and reinforce learning.

Recommended prior knowledge and skills: Access to this Unit will be at the discretion of the Centre, however it is recommended that the candidate has achieved one of the following HN Units: HN Unit D76V 35: *Software Development: Object Oriented Programming*, HN Unit D76R 35: *Software Development: Event Driven Programming*, HN Unit D76X 35: *Software Development: Procedural Programming*, or HN Unit (Code): *Software Development*: *Applications Development*. Alternatively, candidates should have considerable practical work experience and some appreciation of the role of program design and implementation.

Core skills: There may be opportunities to gather evidence towards core skills in this Unit, although there is no automatic certification of core skills or core skills components.

Higher National Unit specification: General information for centres (cont)

Context for delivery: If this Unit is delivered as part of a group award, it is recommended that it should be taught and assessed within the subject area of the group award to which it contributes. It is recommended that the candidate has taken HN Unit D77D 34: *Systems Development: Introduction* before starting this Unit.

Assessment: This Unit can be assessed by one assessment in the format of a report.

It is envisaged that Outcome 1 can be assessed by including a section within the final report.

Outcomes 2 to 4 would involve the analysis and design of a small-scale application.

The candidate will be required to carry out some basic data analysis which will lay down the basis for the application.

Before embarking on the design of the application (menu, form, report) and any redesign the candidate will need to evaluate each component of the system by applying the principles laid down within the field of Human Computer Interaction (HCI).

The candidate will then be expected to provide a demonstration of the application to either the other candidates within the group and/or the candidate's lecturer.

As part of the submission, the candidate should also submit a report covering the knowledge and/or skills required for the successful completion of Outcome 4. These requirements will be based on the application delivered. The sections will include system definition (short narrative outlining the aims and objectives of the application), user interface (evaluation of the systems interface based on HCI concepts), security (evaluation of the application of relational database principles and security issues) and bibliography (outlining sources of information).

An Assessment Exemplar has been produced to indicate the national standard of achievement at SCQF level 8.

Higher National Unit specification: statement of standards

Unit title: Software Development: Fourth Generation Environments

Unit code: D76S 35

The sections of the Unit stating the Outcomes, knowledge and/or skills, and evidence requirements are mandatory.

Where evidence for Outcomes is assessed on a sample basis, the whole of the content listed in the knowledge and/or skills section must be taught and available for assessment. Candidates should not know in advance the items on which they will be assessed and different items should be sampled on each assessment occasion.

Outcome 1

Demonstrate an understanding of the features and uses of a RDBMS and fourth generation language environment

Knowledge and/or skills

- Identification of code generator
- Identification of data manipulation language
- Identification of data definition language
- Identification of data dictionary
- Identification of report generator
- Identification of form generator

Evidence requirements

The candidate must produce a written explanation of the features and uses of each of the knowledge and skills to be included as part of the final report produced for the assessment of the Unit.

The assessment of this Outcome will be by means of an integrated assessment. Details of the evidence requirements are given under Outcome 4.

Assessment guidelines

The assessment of this Outcome is combined with Outcomes 2, 3 and 4. Details of the assessment guidelines are given under Outcome 4.

Higher National Unit specification: statement of standards (cont)

Unit title: Software Development: Fourth Generation Environments

Outcome 2

Use a Data Definition Language (DDL) and a Data Manipulation Language (DML) to define and interrogate tables

Knowledge and/or skills

- Identification and implementation of table creation statements
- Identification of key fields
- Identification of linking fields
- Population of tables
- Identification of simple queries
- Identification of complex queries involving more than one table which are linked

Evidence requirements

The assessment of this Outcome will be by means of an integrated assessment. Details of the evidence requirements are given under Outcome 4.

Assessment guidelines

The assessment of this Outcome is combined with Outcomes 1, 3 and 4. Details of the assessment guidelines are given under Outcome 4.

Outcome 3

Apply a systematic approach to designing applications/systems interfaces

Knowledge and/or skills

- How to use and modify data using a variety of 4GL tools, ie form generator, report generator.
- The advantages and disadvantages of the objects used to produce forms, reports or menus
- How to apply HCI concepts to the design of user interfaces

Evidence requirements

The assessment of this Outcome will be by means of an integrated assessment. Details of the evidence requirements are given under Outcome 4.

Assessment guidelines

The assessment of this Outcome is combined with Outcomes 1, 2 and 4. Details of the assessment guidelines are given under Outcome 4.

Higher National Unit specification: statement of standards (cont)

Unit title: Software Development: Fourth Generation Environments

Outcome 4

Evaluate the design and functionality of an RDB application

Knowledge and/or skills

- Evaluation of the interface using design heuristics criteria (for example, flexibility, learnability, user satisfaction, speed of performance)
- Evaluation of the system with regards to security, data integrity, referential integrity

Evidence requirements

The evidence requirements stated under this section cover Outcomes 1, 2, 3 and 4.

The candidate will be required to produce a report that includes the following sections:

1. Title
2. From a given System Definition produce a short narrative outlining the aims and objectives of the given application in approximately 50 words.
3. Components of a fourth generation environment. This section must include a definition of the features and uses of each of the following components: code generator, DML, DDL, data dictionary, report generator and form generator
4. From a manual table definition produce a script that will create and populate table structures. The required system must utilise a minimum of three tables that are all linked. At least one example of a text, date and numerical field must be included in the definition. The candidate must provide evidence of using primary keys and referencing. Each table must include at least three fields. The candidate must implement the script correctly.
5. The candidate must include evidence of at least five query definitions which show evidence of utilising the following constructs: selection, logical operators (for example, AND, OR, NOT), arithmetic operators (for example, SUM, AVERAGE), comparators (for example, LIKE, IN) and grouping operators.
6. The candidate must produce an evaluation of the user interface from a system of one other candidate in the peer group. This evaluation must be based upon how the principles of HCI have been applied to the other candidates' system. This evaluation will be presented in a written format of approximately 200 words.
7. The required system must use at least two forms and produce at least two reports. One form must utilise a one-to-many relationship. The candidate will need to ensure that the forms created use at least the following objects: command button, text box and label. The system must have at least one security device such as a password on a form. One of the reports must group the data and utilise a calculated field.

Higher National Unit specification: statement of standards (cont)

The candidate will need to ensure that the reports contain at least four fields, one which must be numeric and one a calculated field (utilising for example, sum or count or average). The candidate will be required to produce printouts for all forms and reports produced.

8. Security – an evaluation of the system in relation to security, data integrity and referential integrity in approximately 200 words.
9. Bibliography (outlining sources of information).

Assessment guidelines

Part 3 of the report covers Outcome 1. Part 4 and 5 cover Outcome 2. Part 8 and 9 cover Outcome 3. Part 2, 6 and 7 cover Outcome 4.

Although the candidate may not implement all of the components in the implementation of the system, s/he should be able to understand the use of each component in a fourth generation environment.

The user interface section should provide evidence of the candidate’s conceptual understanding of the area of concern along with any interpretations of those concepts to the implementation.

Comment on design heuristics as laid down, for example, by Morlich and others (Human Computer Interaction), should be found to be useful in the evaluation of the principles of HCI as they apply to user interfaces.

Security issues such as data validation and data integrity may be commented on within the context of the implementation. The candidate may wish to highlight a validation event that they have included.

The candidate should be expected to create and populate at least three linked tables for an application from a given problem. The problem can be annotated by using any appropriate data modelling techniques, eg E-R diagrams.

The candidate should be expected to produce a variety of forms, reports and menus as an interface to a given problem. It is envisaged that the same problem will be used throughout the assessment of this Unit.

The system of forms and reports should be held together by a series of menus. These may take the form of pull-down menus, navigation buttons or a combination of both.

Administrative Information

Unit code: D76S 35

Unit title: Software Development: Fourth Generation Environments

Superclass category: CB

Date of publication: June 2001

Source: SQA

Additional copies of this Unit specification can be purchased from the Scottish Qualifications Authority. The cost for each Unit specification is £2.50 (minimum order £5.00). Additional copies of the Assessment Exemplar can also be purchased from Scottish Qualifications Authority. The cost is £15.00.

Higher National Unit specification: support notes

Unit title: Software Development: Fourth Generation Environments

This part of the Unit specification is offered as guidance. The support notes are not mandatory.

While the exact time allocated to this Unit is at the discretion of the Centre, the notional design length is 80 hours.

Guidance on the content and context for this Unit

Fourth generation environments are increasingly being utilised to create solutions to complex every day problems in a fast, efficient way. Their flexibility is fundamental to the success of systems and application development.

It is recommended that centres offer this Unit in collaboration with other Units such as HN Unit D76Y 35: *Software Development: Rapid Applications Development and Prototyping*.

It is recommended that this Unit be taught using a modern fourth generation medium such as ORACLE, INGRES or such like.

The candidate will be able to reference material from a variety of sources, including the Internet, journals and recommended textbooks.

The Unit has been developed with a view towards software development using a fourth generation language (4G). It is probable that centres will adopt ORACLE and with it the corresponding SQL language as the vehicle for learning and assessment

The candidate will be encouraged to develop research skills and investigate the development of 4GL's and their environment within the context of modern software development.

The Unit is geared towards candidates wishing to experience the development of a software product which links the knowledge of databases with the development of an application utilising modern development tools.

As such the candidate may use text books and the Internet to discover:

- Methods of interaction (forms, events and objects)
- Principles of interaction (http://www.useit.com/papers/heuristic/heuristic_list.html)
- Development of software products using a non-traditional language

Candidates should be able to develop a natural understanding of the use and capability of the different features of a 4GL through practical experience. However, they should also experience the theoretical background to the development and utilisation of 4GL's.

The candidate will have to carry out some data analysis on the case study. This would take the form of the identification of keys (primary and foreign) and related tables. It is therefore recommended that candidates take HN Unit D77D 34: *Systems Development: Introduction* before taking this Unit.

Unit title: Software Development: Fourth Generation Environments

Guidance on the delivery and assessment of this Unit

Delivery of the Unit would be best served by completing Outcome 1 first. Outcome 1 should be assessed by means of a series of short answers that will form part of the final report. The remaining Outcomes should be assessed via a case study. Integration of Outcomes 2 to 4 will provide a more holistic approach more akin to the production of a software product.

The case study should take up approximately one third of the allotted time. The case study should be based on a realistic situation or preferably on a local work based problem. The teaching component of this Unit should be completed before the candidate embarks on the assessment. If this is not possible the candidate should have been exposed to the features of a 4GL, have designed and implemented the scripts, forms and reports before starting to complete the assessment.

Before attempting the assessment the candidate should be exposed to the 4GL components through a series of small exercises or case studies which are logically broken down in to realistic steps.

Discussions and presentations on the components of a 4GL could be held in parallel with practical exercises in producing scripts for the implementation of stable data structures. For example, a set of prescribed normalised tables could be used as the basis for the implementation and population of tables within the 4GL. The candidate should be aware of the need to create/analyse data before implementation begins.

These scripts could be used for the development of user interfaces. The candidate could utilise the experiences of using screen generators and toolboxes which were observed in the development of event driven applications (for example, Visual Basic or Delphi) to aid the candidates production of the form/screen interface.

The candidate should be exposed to a variety of techniques to aid her/him in the understanding of using and designing forms and reports. Sample forms could be used to allow the candidate to evaluate the interface. The candidate should be encouraged to present solutions to class exercises in interface design to their peers, and to be exposed to constructive criticism and praise. These presentations could also be used to encourage self-evaluation. Interface exercises with no validation/data verification could be given to the candidate in order to enable her/him to concentrate on the development of security and validation procedures.

The key to all exercises described would be the avoidance of lengthy repetition. If the exercise is based on form design then the underlying data and structures supporting that data should be in place.

If the exercise is in validation, then the form should be in place so that the exercise is focussed on writing code, which will create the password, or range check.

Suggested delivery schedule. This schedule is based on at least three hours exposure per week.

Higher National Unit specification: support notes (cont)

Week 1: presentation of features and components of a 4GL, introduction of DDL and centre's 4GL.

Week 2 and 3: application of DDL from exercises. Creation of applications utilising at least two linked tables, introduction to DML.

Week 4: exercises using DML, introduction and presentation of forms and/or 'replace text as you type' generators.

Week 5 to 7: exercises on implementation of forms based on single tables and/or multiple tables, introduction of other form generator tools (such as combo boxes, option buttons, repeater, OLE objects and images).

Week 8: exercises taking data and constructing scripts, implementing scripts and creation of interfaces (forms), presentation and introduction to reports and/or pull-down menus.

Week 9: presentation and introduction to HCI concepts and evaluation processes.

Week 10: exercises on HCI concepts, re-evaluation of previous exercises involving form/report design and implementation, peer group evaluation.

Week 11 to end: assessment.

The assessment brief should include a description of the system and the required output of the system including a description of the forms and reports to be produced. The brief should include a description of the relationships between the proposed tables (possibly in the form of an Entity-Relationship diagram, showing all primary keys and foreign keys). Table descriptions should be included showing the data which will eventually populate these tables.

The section of the assessment brief covering Outcome 1 should consist of a series of headings which the candidate will expand upon. These headings will correspond to the knowledge and skills identified in Outcome 1.

The candidate should be encouraged to complete this section before attempting to complete the other parts of the report.

The candidate should create the script, and ensure that no errors occur before embarking on the user interface section. Using the form and report generator the candidate will complete the pre-subscribed tasks and evaluate his/her own work.

The candidate will then be expected to evaluate another candidate's work using the same criteria as used on their own application. Both evaluations will be written up in the form of a short report. It is advised that these reports should include not only criticisms but also constructive recommendations. This should alleviate the problem of unsubstantiated criticism.

This Unit can only be suitable for candidates who will have access to a fourth generation environment. The use of PERSONAL ORACLE and POWER OBJECTS would allow all Outcomes to be covered.

To ensure that peer evaluation takes place it is envisaged that the centre responsible for the delivery of this material use the exemplar material provided by SQA as a means for carrying out the peer group evaluation part of the report.

Higher National Unit specification: support notes (cont)

Candidate authenticity can be guaranteed using the same assessment regulations set. However it would be envisaged that Outcome 1 would be assessed by the use of a report covering the criteria.

Open learning

If this Unit is delivered by open or distance learning methods, additional planning and resources may be required for candidate support, assessment and quality assurance.
A combination of new and traditional authentication tools may have to be devised for assessment and re-assessment purposes. For further information and advice, please see *Assessment and Quality Assurance for Open and Distance Learning* (SQA, February 2001 — publication code A1030).

Special needs

This Unit specification is intended to ensure that there are no artificial barriers to learning or assessment. Special needs of individual candidates should be taken into account when planning learning experiences, selecting assessment instruments or considering special alternative Outcomes for Units. For example, some candidates may require a longer period for the single assessment or may require that it be split into more than one event.

For information on these, please refer to the SQA document *Guidance on Special Assessment and Certification Arrangements for Candidates with Special Needs/Candidates for Whom English is an Additional Language* (SQA, 2000).

General information for candidates

Unit title: Software Development: Fourth Generation Environments

You will learn about the structure and the facilities offered within a fourth generation environment.

You will be expected to understand and use some of the features common to fourth generation language (4GLs) including form and report generators, data dictionaries, and data manipulation and definition languages.

As well as understanding the features of a 4GL you will demonstrate this understanding by constructing a small-scale application using some of the above features.

It is expected that you will use either ORACLE or INGRES S to implement the small-scale application, although your Centre may wish to adopt one of the other 4GLs that are currently available.

You will be exposed to a variety of tasks, which will develop your skills in the design of database components within the 4GL. Construction of tables, the identification of key fields, and the definition of links will allow you to construct the building blocks upon which your subsequent application may be based.

A variety of design tools and/or components may be utilised to allow you to produce usable forms and reports which are acceptable to users and/or peer groups. These features may be evaluated using modern techniques and incorporate heuristics generic to human computer interface design.

You will experience evaluation of your own and other candidates’ work, and learn to use this constructive praise and/or criticism to aid further development.

The assessment is the production of a report in which you will display a theoretical understanding of the features of a 4GL, construct a database using DDL and DML components, and the design and implementation of a small-scale application. Form, report and code generators will be used to construct a working application.

The report also acknowledges the need for modern software applications to be evaluated. Therefore you will be expected to evaluate, not only your own work but that of other peer group members.