

Unit 10: Software Development Project

Learning hours: 60

NQF level 4: BTEC Higher National — H2

Description of unit

This unit will form a central part in the development of the learner's ability to link and integrate the knowledge and skills acquired during the programme to produce a practical solution to a realistic problem. This is a major piece of work that should demonstrate the performance expected at a Higher National level.

The chosen problem may be work-based, college-based or a learner interest but it must be developed in accordance with the unit's frameworks and constraints. This will develop the ability to produce a suitable, realistic software solution to an agreed specification within a defined timescale.

Summary of learning outcomes

To achieve this unit a learner must:

- 1 Develop a **plan for a project to an agreed specification**
- 2 Develop a **solution** for the project
- 3 **Implement the solution** to the system
- 4 **Present and evaluate** the project.

Content

1 Plan for a project to an agreed specification

Project selection: research and review areas of interest, selection of project, design of structure and production of aims, estimation and list of required resources

Project plan: production of computer-based project plan, to include timescales, deliverables, milestones, resource allocation, quality assurance systems, quality plans

Design/structure: consideration of alternative design methods/techniques and environment (use of preliminary design, detailed design, data design, architectural design, procedural design, modular design, structured design, object design, real time design, graphical displays, storyboarding, data analysis)

2 Solution

Programming languages: selection of languages for the project, specification of language to include identification of area of project used

Programming techniques: techniques to be implemented, where techniques will be used

Program requirement: data type, data structure (arrays and records, declarations, assignments, expressions); control structures, subprograms, algorithms including recursion

Program testing: strategy, functional testing (black box), structural testing (white box)

3 Implement the solution

Specification: operating environment

Software installation: software installation procedures

Testing activities: component testing, case testing, unit testing, integrating testing, acceptance testing

4 Present and evaluate

Demonstration: well organised, structured, giving clear explanations, justify overall design

Audience: peer groups, tutors, clients, employers

Documentation: instruction manuals, user guides, evidence to support designs, printed screen shots where applicable, printed examples of input screens

Outcomes and assessment criteria

Outcomes	Assessment criteria for pass To achieve each outcome a learner must demonstrate the ability to:
1 Develop a plan for a project to an agreed specification	<ul style="list-style-type: none">• design a plan for an agreed project• produce a project specification including complete user requirements and design specifications• select and use appropriate software planning tools
2 Develop a solution for the project	<ul style="list-style-type: none">• select a suitable programming language• produce all algorithms, pseudo code, flowchart, data dictionary, programme coding, classes, methods as required• review, select and use appropriate testing techniques to validate the project• use appropriate software tools to develop project coding
3 Implement the solution to the system	<ul style="list-style-type: none">• identify and implement system requirements• apply verification and testing required at all levels of the system• produce a clear and structured implementation plan
4 Present and evaluate the project	<ul style="list-style-type: none">• present your solution in a structured and well organised format• produce documentation for all stages of a project and a full report• assess the quality of the product compared to the clients' original requirements

Guidance

Delivery

Learners must work individually. Once a specific project has been agreed, the tutor's role is verifying, encouraging and counselling rather than directing. Part of the unit should be formally devoted to verifying the project stages. During the development it would be useful to feed back to the learner group so that the learners can benefit from comments on good and bad practice etc. It is considered good practice to involve a few employers in the demonstration and/or the review.

It is important that learners document each stage of the project. The content will differ according to the specific project but in each case it should include complete, relevant documentation of all stages of the project to agreed standards, and critical evaluation of the project. It is advisable for learners to keep a log of the project as this can be used as part of their assessment evidence.

Assessment

Evidence will be obtained by validating the deliverables produced at each milestone against the agreed plan ie project selection, planning and outline specification or project selection, planning and detailed design, implementation, and project demonstration. If milestones are missed learners should demonstrate the remedial action taken. Demonstration should be to a known audience.

Links

The unit is intended to integrate skills and knowledge from the core units with other units from the rest of the programme. Learners should be aware of the significance of knowledge and experience gained from earlier work. In particular, knowledge and skills could be drawn on from *Unit 7: Quality Systems*.

Resources

Learners should have access to a wide variety of relevant software tools including management tools and resources depending on the specific project. Acknowledgement and support should be given for the use of relevant software tools not usually used in the centre. Centres should try to involve industrial organisations to bring realism and relevance to the project.

Support materials

Textbooks

Tutors should be aware that textbooks are frequently updated and that they should use the latest editions where available. This is a practical unit and textbook materials should be used for reference purposes. There is a range of general textbooks relating to this unit, including the following:

- Hughes, B and Cotterell, M — *Software Project Management* — (McGraw Hill, 2002) ISBN: 0077095057
- Pressman, R S — *Software Engineering — A Practitioner's Approach* — (McGraw Hill, 2000) ISBN: 0077096770