cetb

Bord Oideachais agus
Oiliúna Chorcaí
*Cork Education and
Training Board*

Cork Education and Training Board

Programme Module for

Software Architecture

Leading to

Level 5 QQI

Software Architecture 5N2772

Introduction
This programme module may be delivered as a standalone module leading to certification in a QQI minor award. It may also be delivered as part of an overall validated programme leading to a Level 5 QQI Certificate.

The teacher/tutor should familiarise themselves with the information contained in Cork Education and Training Board programme descriptor for the relevant validated programme prior to delivering this programme module.

The programme module is structured as follows:

1. Title of Programme Module
2. QQI Component Title and Code
3. Duration in hours
4. Credit Value of QQI Component
5. Status
6. Special Requirements
7. Aim of the Programme Module
8. Objectives of the Programme Module
9. Learning Outcomes
10. Indicative Content
11. Assessment a. Assessment Technique(s) b. Mapping of Learning Outcomes to Assessment Technique(s) c. Guidelines for Assessment Activities
12. Grading
13. Learner Marking Sheet(s), including Assessment Criteria

Integrated Delivery and Assessment
The teacher/tutor is encouraged to integrate the delivery of content where an overlap between content of this programme module and one or more other programme modules is identified. This programme module will facilitate the learner to develop the academic and vocational language, literacy and numeracy skills relevant to the themes and content of the module.

Likewise the teacher/tutor is encouraged to integrate assessment where there is an opportunity to facilitate a learner to produce one piece of assessment evidence which demonstrates the learning outcomes from more than one programme module. The integration of the delivery and assessment of 5N0541 Fundamentals of Object Oriented Programming, 5N1367 Team working, level 5 Communications and level 5 Mathematics modules with that of other level 5 modules is specifically encouraged, as appropriate.

Indicative Content
The indicative content in Section 10 does not cover all teaching possibilities. The teacher/tutor is encouraged to be creative in devising and implementing other approaches, as appropriate. The use

of examples is there to provide suggestions. The teacher/tutor is free to use other examples, as appropriate. The indicative content ensures all learning outcomes are addressed but it may not follow the same sequence as that in which the learning outcomes are listed in Section 9. It is the teacher's/tutor's responsibility to ensure that all learning outcomes are included in the delivery of this programme module.

1. Title of Programme Module

Software Architecture

2. Component Name and Code

Software Architecture 5N2772

3. Duration in Hours

150 Hours (typical learner effort, to include both directed and self-directed learning)

4. Credit Value

15 Credits

5. Status

This programme module may be compulsory or optional within the context of the validated programme. Please refer to the relevant programme descriptor, Section 9 Programme Structure

6. Special Requirements

NONE

7. Aim of the Programme Module

This programme module aims to equip the learner with the knowledge, skill and competence to understand the software development life cycle for a range of projects and introduce industry-standard tools and techniques for implementing appropriate software solutions.

8. Objectives of the Programme Module

- To enable the learner to understand the history and development general systems theory
- To explore various with Software Development methodologies from traditional to more modern approaches.
- To understand best practice in relation to the development of software systems.
- Discover how to use software modelling tools such as UML to generate code and model systems based on specific requirements.
- Discover how to use appropriate testing techniques to test design specifications of Software.

9. Learning Outcomes of Level 5 Software Architecture 5N2772

Learners will be able to:

1. Demonstrate an understanding of general systems theory to the design of information systems in organisations

2. Discuss the impact of organisational structure on the design of information systems

3. Explain the various stages and associated roles, of the traditional software development lifecycle from feasibility/investigation through end-of- maintenance/support

4. Describe the role of modern methodologies in the software development process

5. Demonstrate an understanding of the management and technology issues, including risks and common pitfalls, involved in the design and construction of software systems

6. Use straightforward UML use-case diagrams to depict the interaction between an actor and system

7. Engage a modern methodology to aid the design of a system from initial requirements

8. Apply the main tools and techniques used in the gathering, recording and analysis of information relating to an existing information system

9. Generate Java (or suitable Object- Oriented language) classes from UML class diagram specifications

10. Test the design specification of a new information system based on a provided use-case

11. Explain the development of an information system using a traditional systems development life cycle model

10. Indicative Content

This section provides suggestions for programme content but is not intended to be prescriptive. The programme module can be delivered through classroom based learning activities, group discussions, one-to-one tutorials, field trips, case studies, role play and other suitable activities, as appropriate.

Section 1 : Introduction to Systems Development
1,2,5

Learning outcome 1 - Demonstrate an understanding of general systems theory to the design of information systems in organisations

Facilitate the learner to:

- Discover the history and aims of General Systems Theory
- Define Information Systems and the different types of information systems prevalent in Organisations
- Analyse in detail the history of Information Systems development for a period of time e.g. 1970 to present

Learning outcome 2 - Discuss the impact of organisational structure on the design of information systems

Facilitate the learner to:

- Explore the different types of Organisational Structure e.g. Hierarchical, Flat, Matrix, Networked
- Examine what type of organizational structure tends to be most willing to embrace technological change and sophistication and why.

Learning outcome 5 - Demonstrate an understanding of the management and technology issues, including risks and common pitfalls, involved in the design and construction of software systems

Facilitate the learner to:

- Examine the importance of having business case for starting the project
- Examine the importance of tool for analysing viability and feasibility of construction of a software system. e.g. Cost-Benefit analysis , Contingency plan.
- Outline the importance of setting project goals e.g., Time, Budget
- Examine the importance estimating resources needed
- Discover the impact of poorly defined system requirements
- Poor communication among customers, developers, and users

Section 2: Systems Development Methodologies
3,4,11

Learning outcome 3 - Explain the various stages and associated roles, of the traditional software development lifecycle from feasibility/investigation through end-of-maintenance/support

Facilitate the learner to:

- **Explore the main stages in the Traditional SDLC Namely**

  - Stage 1: Feasibility analysis
  - Stage 2: Requirement analysis and specification
  - Stage 3: Designing the product architecture
  - Stage 4: Building or Developing the Product
  - Stage 5: Testing
  - Stage 6: Deployment and Maintenance

- Discuss the different variations of the SDLC eg. Waterfall, V-Shaped Model, Incremental Model, Spiral Method (SDM)

Learning outcome 4 - Describe the role of modern methodologies in the software development process

Facilitate the learner to:

- Examine the history and evolution of the Agile Software Development methodologies
- Discover and discuss the main principles of the Agile manifesto
- Examine some of the main Agile Software Development methods e.g. Scrum, Kaban Method, Dynamic systems development method (DSDM) ,Extreme Programming (XP),
- Outline the advantages and disadvantages of Agile Development methodologies over more traditional SDLC methodologies e.g. Waterfall approach

Learning outcome 11 - Explain the development of an information system using a traditional systems development life cycle model

Facilitate the learner to:

- Using a traditional SDLC outline the stages in the development of a current information system using this approach e.g. Waterfall SDLC
- For each stage in the methodology explain the main activities that took place

Section 3:Systems Modelling
6,7,8,9,10

Learning outcome 6 - Use straightforward UML use-case diagrams to depict the interaction between an actor and system

Facilitate the learner to:

- Discover what a Use Case is.
- Discover and define the elements that make up a UML use-case diagram e.g
  - Actors
  - Use-Case
  - System boundary
  - Relationships (extends, includes)
- Understand the difference between the different elements and how to elicit them from a business use case
- Create Use-Case diagrams from a set of business rules or requirements.

Learning outcome 7 - Engage a modern methodology to aid the design of a system from initial

requirements

Facilitate the learner to:

- Design and develop a piece of Software using an Agile Software Development Methodology
- Document the stages, processes, procedures, meetings and tools used from your chosen modern methodology in Designing you piece of software

Learning outcome 8 - Apply the main tools and techniques used in the gathering, recording and analysis of information relating to an existing information system

Facilitate the learner to:

- **Discover the tools and techniques that are used gather and record information about a current information system e.g.**
  - Use Cases
  - Stakeholder identification
  - Stakeholder Interviews
  - Questionnaires
  - Data- Gathering
  - Observation
  - Document Analysis

- Use a relevant selection of the above tools and techniques to gather information about an existing information system

Learning outcome 9 - Generate Java (or suitable Object-Oriented language) classes from UML class diagram specifications

- Discover what a class diagram is and what the main elements are that make up a class diagram. e. g Attributes, operations, relationships between classes
- Examine the purpose of class diagrams and who in the SDLC uses them.
- Examine the difference between platform dependent and platform independent class diagrams.
- Use a suitable IDE e.g. Netbeans to Forward Engineer a class diagram to create the related Java class or classes
- Discuss the relationship between the class diagrams and the Java code and classes that are engineered from them.

Learning outcome 10 - Test the design specification of a new information system based on a provided use-case

Facilitate the learner to:

- Examine the importance and benefits of testing
- Derive a test case from a provided use case to include all or some of the following elements
  - ID

- Name
- Actor
- Description
- Priority
- Frequency of use
- Preconditions
- Typical workflow
- Exception workflows
- Post conditions

11. Assessment

11a. Assessment Techniques

Skills Demonstration	**70%**
Exam (Theory)	**30%**

11b. Mapping of Learning Outcomes to Assessment Techniques

In order to ensure that the learner is facilitated to demonstrate the achievement of all learning outcomes from the component specification; each learning outcome is mapped to an assessment technique(s). This mapping should not restrict an assessor from taking an integrated approach to assessment.

Learning Outcome	**Assessment Technique**
1. Demonstrate an understanding of general systems theory to the design of information systems in organisations	Exam
2. Discuss the impact of organisational structure on the design of information systems	Exam
3. Explain the various stages and associated roles, of the traditional software development lifecycle from feasibility/investigation through end-of- maintenance/support	Exam
4. Describe the role of modern methodologies in the software development process	Skills Demo
5. Demonstrate an understanding of the management and technology issues, including risks and common pitfalls, involved in the design and construction of software systems	Exam
6. Use straightforward UML use-case diagrams to depict the interaction between an actor and system	Skills Demo
7. Engage a modern methodology to aid the design of a system from initial requirements	Skills Demo
8. Apply the main tools and techniques used in the gathering, recording and analysis of information relating to an existing information system	Skills Demo
9. Generate Java (or suitable Object- Oriented language) classes from UML class diagram specifications	Skills Demo
10. Test the design specification of a new information system based on a provided use-case	Skills Demo
11. Explain the development of an information system using a traditional systems development life cycle model	Skills Demo

11c. Guidelines for Assessment Activities

The assessor is required to devise assessment briefs and marking schemes for the skills demonstration and an examination paper, marking scheme and outline solution for the examination. In devising the assessment briefs and examination paper, care should be taken to ensure that the learner is given the opportunity to show evidence of achievement of ALL the learning outcomes. Assessment briefs may be designed to allow the learner to make use of a wide range of appropriate media in presenting assessment evidence. Quality assured procedures must be in place to ensure the reliability of learner evidence.

Skill Demonstration	**70%**

The Skills Demonstration can be conducted throughout the duration of this programme module.

Engage and document Agile Software Development Methodology / Methodologies to develop a piece of software

- Skills Demonstration 1 **(12 Marks) (1 Hour) LO7**
  - Engage a modern methodology to aid the design of a system from initial requirements
- Skills Demonstration 2 **(12 Marks) (1 Hour) LO4**
  - Describe the role of modern methodologies in the software development process
- Skills Demonstration 3 **(12 Marks) (1 Hour) LO8**
  - Apply the main tools and techniques used in the gathering, recording and analysis of information relating to an existing information system
- Skills Demonstration 4 **(12 Marks) (1 Hour) LO7**
  - Generate Java (or suitable Object- Oriented language) classes from UML class diagram specifications
- Skills Demonstration 5 **(12 Marks) (1 Hour) LO6**
  - Use straightforward UML use-case diagrams to depict the interaction between an actor and system
- Skills Demonstration 6 **(12 Marks) (1 Hour) LO10**
  - Test the design specification of a new information system based on a provided use-case
- Skills Demonstration 7 **(10 Marks) (1 Hour) LO11**
  - Explain the development of an information system using a traditional systems development life cycle model

Evidence for this assessment technique may take the form of oral, written, graphic, visual or digital evidence, or any combination of these.

Any video or digital evidence must be provided in a suitable format. All instructions for the learner must be clearly outlined in an assessment brief.

Examination -theory	**30%**

The examination will be based on a range of specific learning outcomes and will be 2 hours in Duration **LOs 1, 2, 3, 5**	
The format for the examination will be as follows: **Section A** **10 Short questions** **Candidates are required to answer 10 (2 marks each)** **Section B.** **2 Structured questions** **Candidates are required to answer 2 (10 marks each)**	

11. Grading

Distinction:	80% - 100%
Merit:	65% - 79%
Pass:	50% - 64%
Unsuccessful:	0% - 49%

At levels 4, 5 and 6 major and minor awards will be graded. The grade achieved for the major award will be determined by the grades achieved in the minor awards.

Software Architecture 5N2772	**Learner Marking Sheet 1** **Skills Demonstration** **70%**

Learner's Name: _________________________________ Learner's PPSN:_________

Assessment Criteria	**Maximum Mark**	**Learner Mark**
Engage and document Agile Software Development Methodology / Methodologies to develop a piece of software		
● Skills Demonstration 1 **(12 Marks) (1 Hour)** ○ Engage a modern methodology to aid the design of a system from initial requirements	**10**	
● Skills Demonstration 2 **(12 Marks) (1 Hour)** ○ Describe the role of modern methodologies in the software development process	**10**	
● Skills Demonstration 3 **(12 Marks) (1 Hour)** ○ Apply the main tools and techniques used in the gathering, recording and analysis of information relating to an existing information system	**10**	
● Skills Demonstration 4 **(12 Marks) (1 Hour)** ○ Generate Java (or suitable Object- Oriented language) classes from UML class diagram specifications	**10**	
● Skills Demonstration 5 **(12 Marks) (1 Hour)** ○ Use straightforward UML use-case diagrams to depict the interaction between an actor and system	**10**	
● Skills Demonstration 6 **(12 Marks) (1 Hour)** ○ Test the design specification of a new information system based on a provided use-case	**10**	
Subtotal	**60**	
● Skills Demonstration 7 **(10 Marks) (1 Hour)** ○ Explain the development of an information system using a traditional systems development life cycle model	**10**	
Subtotal	**10**	
Total Mark	**70**	

Assessor's Signature: ___________________________ Date: ____________________

External Authenticator’s Signature: ___________________________ Date: ____________________

Mobile Technologies 5N2772	Learner Marking Sheet 2 Examination Theory 30%

Learner’s Name: __________________________________ Learner’s PPSN: _________

Assessment Criteria	Maximum Mark	Learner Mark
Section A: **Short answer questions. Answer all questions. (2 marks each)** **Question 1** **Question 2** **Question 3** **Question 4** **Question 5** **Question 6** **Question 7** **Question 8** **Question 9** **Question 10**	2 2 2 2 2 2 2 2 2 2	
Subtotal	20	
Section B: **Structured questions. Answer all questions (10 marks each)** **Question No. 1** **Question No. 2**	5 5	
Subtotal	10	
Total	30	

Assessor’s Signature: ___________________________ Date: ____________________

External Authenticator’s Signature: ___________________________ Date: ____________________