

## Module: Software Analysis & Design 261 (Deaf)

<b>Module name:</b>	Software Analysis & Design 261 (Deaf)
<b>Code:</b>	D-SWA261
<b>NQF level:</b>	6
<b>Type:</b>	Fundamental – Diploma in Information System (Software Development stream)
<b>Contact Time:</b>	34 hours
<b>Structured time:</b>	6 hours
<b>Self-directed time:</b>	40 hours
<b>Notional hours:</b>	80 hours
<b>Credits:</b>	8
<b>Prerequisites:</b>	None

### Purpose

This module, Software analysis and design deal with systems development processes, methods, techniques and tools focusing on the elicitation and initial modelling of information systems requirements that enable the identification of information problems and the subsequent analysis and modelling of an efficient solution to those problems. It critically examines the issues and professional responsibilities that need to be considered at different phases in the development of information systems for an organization; including the impact of the systems on intended users and the maintenance of quality.

### Outcomes

Upon successful completion of this module, the student will be able to:

- Demonstrate integrated knowledge of the types of business needs that can be addressed using information technology-based solutions.
- Understand a range of methods of enquiry in the discipline and their suitability to be applied in specific investigations to resolve problems related to the context of methodologies, writing clear, concise business requirements documents and converting them into technical specifications.
- Demonstrate the ability to evaluate, select and apply appropriate methods, procedures or techniques in investigation processes within a defined context.
- Demonstrate the ability to make decisions and act appropriately in familiar and new contexts.
- Demonstrating an understanding of the relationships between systems, and of how actions, ideas or developments in one system impact other systems.
- Develop and communicate their ideas and opinions in well-formed arguments, using appropriate academic discourse.
- Demonstrate the ability to evaluate different sources of information, select information appropriate to the task, and apply well-developed processes of analysis, synthesis and evaluation to that information.

- Demonstrate the ability to work effectively in a team or group, and to take responsibility for his or her decisions and actions and the decisions and actions of others within well-defined contexts, including the responsibility for the use of resources where appropriate.

## Assessment

- Continuous evaluation of theoretical work through written assignment, a formative, and a summative test.
- Continuous evaluation of project work, where the student must design, manage and report on the evaluation of testing methodologies and the selection of an appropriate methodology for a given scenario, justifying the choice made with well-formed arguments and evidence.
- Final assessment through a written examination.
- The assignments or projects collectively will count 30% of your class mark.
- All tests will collectively account for 70% of your class mark.
- Your class mark contributes 30% towards your final mark for the subject, while the final assessment accounts for 70% of your final mark.

## Teaching and Learning

### Learning materials

#### *Prescribed books (EBSCO)*

- 📖 **Kenneth Barclay and John Savage (2004) Object-Oriented Design with UML and Java. Oxford: Butterworth-Heinemann.**
- 📖 **Ambler, S. W. (2003) The Elements of UML Style. Cambridge, UK: Cambridge University Press.**
- 📖 **Satzinger, J.W., Jackson, R.B. and Burd, S.D. (2015). Systems analysis and design in a changing world. Cengage learning.**

### Learning activities

The teaching approach consists of a combination of formal lectures on theoretical and practical concepts, solving real-world problems through exercises, demonstrations of feasible solution in a specific context and discussions of high-level design specifications. One mandatory assignment and one project must be completed during the course. The experiences and progress on these practical components form the content of class discussions.

### Notional learning hours

Activity	Units	Contact Time	Structured Time	Self-Directed Time
Lecture		27.0		13.0
Formative feedback		3.5		
Project	1	3.5		9.0
Assignment	1			3.0
Test	2		4.0	8.0
Exam	1		2.0	7.0
		<b>34.0</b>	<b>6.0</b>	<b>40.0</b>

## Syllabus

- Software Development Fundamentals
- Identification of opportunities for IT-enabled organisational change
- Fundamentals of IS project management
- Analysis and specification of system requirements
- Different approaches to systems analysis & design: structured SDLC, unified process/UML, agile methods
- Different approaches to implementing information systems to support business Requirements
- Impact of implementation alternatives on system requirements specification
- Methodology (V-Process & Waterfall)