



## MODULE SPECIFICATION

Part 1: Information			
Module Title	Advanced Software Development		
Module Code	UFCF8S-30-2	Level	Level 5
For implementation from	2021-22		
UWE Credit Rating	30	ECTS Credit Rating	15
Faculty	Faculty of Environment & Technology	Field	Computer Science and Creative Technologies
Department	FET Dept of Computer Sci & Creative Tech		
Contributes towards	Computer Science BSc (Hons) 2020-21		
Module type:	Standard		
Pre-requisites	Foundations of Computing 2020-21		
Excluded Combinations	None		
Co- requisites	None		
Module Entry requirements	None		

Part 2: Description
<p>This module teaches students how to apply contemporary software development approaches in order to develop advanced software applications such as distributed and concurrent software intensive database systems featuring GUI front ends.</p> <p><b>Educational Aims:</b> This module aims to build and the underlying skills developed at L4 to bring them together in the context of greater software development complexity.</p> <p><b>Outline Syllabus:</b> Tools and techniques for problem analysis            Software solution approaches and the requirements for a chosen software solution approach.            Approaches to the creation of high- and low-level designs for the chosen solution approach.            Software development methodologies such as Agile and the incremental approach.            Design representations and modelling.            - Unified Modelling Language (UML)            - Security by design principles            Design principles: Design for resilience and system risk, the SOLID principles in object-oriented</p>

## STUDENT AND ACADEMIC SERVICES

software development - for example, the single responsibility principle, open-closed principle etc, design patterns and their use.

Object-oriented programming language features to support threading and distributed systems.

Distributed databases

User interface design, including secure transaction management and locking for the former.

Testing concepts and methods.

One Agile software development methodology will be taught in depth using, for example, Scrum or DevOps

Legal, ethical, social and professional issues.

Configuration management, software licensing, quality management, version control and contemporary.

Emerging software development tools, technologies and methods.

**Teaching and Learning Methods:** In this module, the focus is on learning by doing. Through a combination of lectures and practical work, students will pursue a software development life cycle, from problem analysis through to implementation, testing and maintenance. Along the way, a range of methods, tools and techniques will be introduced and experimented with.

Throughout there is an emphasis on enquiry. Students will be required to justify their development choices to both their peers and to teaching staff. They will also be required to consider legal, ethical, social and professional implications of the activities. Group working will be used to mirror how software development is managed in industry.

### Part 3: Assessment

A three hour examination assesses students' knowledge of the key concepts, principles and methods associated with the analysis of problems, specification of requirements, and design, coding and testing of solutions.

The coursework comprises of a two-part of a group project assignment. The first part covers the analysis of a problem, the specification of its associated requirements and the design of a solution. The second part covers the implementation and testing of the design created in the first part. Assessment of this component will be based both on portfolio submissions that include, analysis, design, coding and testing documents and also an in-class software demonstration. There will be a group mark and a mark for the assessment of an individual's contribution to group dynamics.

Students will have the opportunity for formative feedback during practical lab/tutorial sessions.

The resit for this component will be based on problem analysis, requirements specification, design, coding and testing – all resulting artefacts being submitted as an individual portfolio with supporting software. Assessment will be by an individual demonstration. The portfolio will include a report analysing how the individual project would be undertaken and managed as a group project.

As with the main sit, at resit, the exam assesses students' knowledge of the key concepts, principles and methods associated with the analysis of problems, specification of requirements, and design, coding and testing of solutions.

First Sit Components	Final Assessment	Element weighting	Description
Group work - Component B		50 %	Group project: including problem analysis, requirements specification, design of a solution, coding and testing. Work to be split among group members.
Examination - Component A	✓	50 %	Exam (3 hours)
Resit Components	Final Assessment	Element weighting	Description

## STUDENT AND ACADEMIC SERVICES

Project - Component B		50 %	An individual project, including problem analysis, requirements specification, design, coding and testing as well as a report (1,000 words) of an analysis of how the individual project would be undertaken and managed as a group project
Examination - Component A	✓	50 %	Exam (3 hours)

### Part 4: Teaching and Learning Methods

Learning Outcomes	On successful completion of this module students will be able to:	
	<b>Module Learning Outcomes</b>	
	MO1	Analyse problems in order to identify software-solution approaches and requirements for computer-based software-intensive systems.
	MO2	Compare and contrast software development methodologies and choose one suitable for a given application.
	MO3	Design, implement, test and manage reasonably sized concurrent and distributed software systems, considering database and GUI components.
	MO4	Develop the necessary transferable skills – e.g. communication, delegation, openness, group decision making, flexibility, tolerance - to enable them to work in a group
	MO5	Discuss the need for security in the context of system development
	MO6	Evaluate emerging software development tools, technologies and methods, e.g. cloud based development, devops and AI-based software development tools.
Contact Hours	<b>Contact Hours</b>	
	<b>Independent Study Hours:</b>	
	Independent study/self-guided study	228
	<b>Total Independent Study Hours:</b>	228
	<b>Scheduled Learning and Teaching Hours:</b>	
	Face-to-face learning	72
	<b>Total Scheduled Learning and Teaching Hours:</b>	72

## STUDENT AND ACADEMIC SERVICES

	<b>Hours to be allocated</b>	300
	<b>Allocated Hours</b>	300
Reading List	<i>The reading list for this module can be accessed via the following link:</i> <a href="https://rl.talis.com/3/uwe/lists/939B5869-07ED-B7EC-90C9-B237896D16B9.html?lang=en-US">https://rl.talis.com/3/uwe/lists/939B5869-07ED-B7EC-90C9-B237896D16B9.html?lang=en-US</a>	