



MODULE SPECIFICATION

Part 1: Information			
Module Title	Collaborative Software Development Project		
Module Code	UFCFCR-30-3	Level	Level 6
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		
Module type:	Standard		
Pre-requisites	None		
Excluded Combinations	None		
Co- requisites	None		
Module Entry requirements	None		

Part 2: Description
<p>Overview: The primary role of a software engineer is to be able to design, build and test high-quality software solutions following best practices and industry standards. They will typically be working as part of a larger collaborative team, in which they will have responsibility for significant elements of the overall project. The developer will need to be able to interpret requirements, specification documentation and designs in order to develop and test software that meets its requirements, even when these requirements may change. Using a collaborative environment allows developers to bring together independent modules together to create a complete solution</p> <p>Educational Aims: Undertake the roles and responsibilities of a software engineer</p> <ul style="list-style-type: none"> Collaborate as a team Interpret and implement a design Collaboratively create a program Create effective and secure software solutions Participate in code reviews, debugging and refactoring processes

STUDENT AND ACADEMIC SERVICES

Thoroughly test a solution

Outline Syllabus: The roles and responsibilities that are required from a software engineer at every stage of the development lifecycle

Collaboration as a team to apply systems analysis and design to a project specification creating artefacts (e.g. use case)

Interpreting and implementing a design that's compliant security requirements (e.g. functional and nonfunctional)

Collaboratively creating a program based on user requirements, embracing an industry based methodology

Industry standard build processes, and tools for configuration management, version control and software build (e.g. GitHub, Bitbucket, Tortoise SVN), release and deployment

Contemporary software development languages

Quality metrics

Code reviews, debugging and refactoring processes to improve code quality and efficiency

Testing to ensure resilience of code and that it meets the functional and non-functional requirements (e.g. black box, white box, unit testing)

Teaching and Learning Methods: Introductory lectures covering the fundamentals and technical underpinning of the module for the first assessment before progressing onto practical delivery through a series of lessons, workshops and practical tasks in the classroom to develop the tools and techniques required to complete the practical assessment for this module. Students are also provided with access to a suitable collaborative software development tools to aid the completion of this module.

Part 3: Assessment

This module is assessed by a combination of techniques: a practical portfolio and a presentation.

The practical portfolio is used to evidence the student's software project, assessing their collaborative software developments, planning, testing, and team working.

The final presentation requires students to identify, justify and communicate all stages of their software development lifecycle critically review the stages their software solution and evaluation of team performance.

The first sit presentation will be delivered as a group with individual grades and feedback issued. To accommodate individual circumstances the re take opportunity will be an individual presentation, with a shorter time allocation.

Tutor-lead formative feedback will be available throughout the module.

First Sit Components	Final Assessment	Element weighting	Description
Presentation - Component A	✓	30 %	Group Presentation (30 Minutes)
Portfolio - Component B		70 %	Portfolio - Design, Build and Test a Software Project built using a Collaborative Approach

STUDENT AND ACADEMIC SERVICES

Resit Components	Final Assessment	Element weighting	Description
Presentation - Component A	✓	30 %	Individual Presentation (20 Minutes)
Portfolio - Component B		70 %	Portfolio - Design, Build and Test a Software Project built using a Collaborative Approach

Part 4: Teaching and Learning Methods																	
Learning Outcomes	<p>On successful completion of this module students will achieve the following learning outcomes:</p> <table border="1"> <thead> <tr> <th>Module Learning Outcomes</th> <th>Reference</th> </tr> </thead> <tbody> <tr> <td>Identify, justify and review all stages of the software development lifecycle for a completed project.</td> <td>MO1</td> </tr> <tr> <td>Analyse, interpret and implement a design compliant with functional, nonfunctional and security requirements</td> <td>MO2</td> </tr> <tr> <td>Perform code reviews, debugging and refactoring to improve code quality and efficiency</td> <td>MO3</td> </tr> <tr> <td>Select appropriate testing techniques to ensure that functional and non-functional requirements are met.</td> <td>MO4</td> </tr> <tr> <td>Deliver software solutions in a complex environment.</td> <td>MO5</td> </tr> <tr> <td>Evaluate and communicate the effectiveness of teamwork and roles in completing a collaborative software development project</td> <td>MO6</td> </tr> </tbody> </table>	Module Learning Outcomes	Reference	Identify, justify and review all stages of the software development lifecycle for a completed project.	MO1	Analyse, interpret and implement a design compliant with functional, nonfunctional and security requirements	MO2	Perform code reviews, debugging and refactoring to improve code quality and efficiency	MO3	Select appropriate testing techniques to ensure that functional and non-functional requirements are met.	MO4	Deliver software solutions in a complex environment.	MO5	Evaluate and communicate the effectiveness of teamwork and roles in completing a collaborative software development project	MO6		
Module Learning Outcomes	Reference																
Identify, justify and review all stages of the software development lifecycle for a completed project.	MO1																
Analyse, interpret and implement a design compliant with functional, nonfunctional and security requirements	MO2																
Perform code reviews, debugging and refactoring to improve code quality and efficiency	MO3																
Select appropriate testing techniques to ensure that functional and non-functional requirements are met.	MO4																
Deliver software solutions in a complex environment.	MO5																
Evaluate and communicate the effectiveness of teamwork and roles in completing a collaborative software development project	MO6																
Contact Hours	<table border="1"> <thead> <tr> <th colspan="2">Independent Study Hours:</th> </tr> </thead> <tbody> <tr> <td>Independent study/self-guided study</td> <td>192</td> </tr> <tr> <td>Total Independent Study Hours:</td> <td>192</td> </tr> <tr> <th colspan="2">Scheduled Learning and Teaching Hours:</th> </tr> <tr> <td>Face-to-face learning</td> <td>108</td> </tr> <tr> <td>Total Scheduled Learning and Teaching Hours:</td> <td>108</td> </tr> <tr> <td>Hours to be allocated</td> <td>300</td> </tr> <tr> <td>Allocated Hours</td> <td>300</td> </tr> </tbody> </table>	Independent Study Hours:		Independent study/self-guided study	192	Total Independent Study Hours:	192	Scheduled Learning and Teaching Hours:		Face-to-face learning	108	Total Scheduled Learning and Teaching Hours:	108	Hours to be allocated	300	Allocated Hours	300
Independent Study Hours:																	
Independent study/self-guided study	192																
Total Independent Study Hours:	192																
Scheduled Learning and Teaching Hours:																	
Face-to-face learning	108																
Total Scheduled Learning and Teaching Hours:	108																
Hours to be allocated	300																
Allocated Hours	300																
Reading List	<p>The reading list for this module can be accessed via the following link:</p> <p>https://rl.talis.com/3/uwe/lists/AD653781-0CB7-33AB-46FD-7EE27F7B2D76.html</p>																

Part 5: Contributes Towards
This module contributes towards the following programmes of study: