



Module Specification

Advanced Software Development

Version: 2021-22, v1.0, 14 Jul 2020

Contents

Module Specification	1
Part 1: Information	2
Part 2: Description	2
Part 3: Teaching and learning methods	3
Part 4: Assessment.....	5
Part 5: Contributes towards	6

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

Department: FET Dept of Computer Sci & Creative Tech

Partner institutions: None

Delivery locations: Frenchay Campus

Field: Computer Science and Creative Technologies

Module type: Standard

Pre-requisites: Foundations of Computing 2020-21

Excluded combinations: None

Co-requisites: None

Continuing professional development: No

Professional, statutory or regulatory body requirements: None

Part 2: Description

Overview: 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.

Features: Not applicable

Educational aims: This module aims to build and the underlying skills developed at L4 to bring them together in the context of greater software development complexity.

Outline syllabus: 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 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.

Part 3: Teaching and learning 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 undertaken. Group working will be used to mirror how software development is managed in industry.

Module Learning outcomes:

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, working in a group , considering database and GUI components. (Assessed in Component B)

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.

Hours to be allocated: 300

Contact hours:

Independent study/self-guided study = 228 hours

Face-to-face learning = 72 hours

Total = 300

Reading list: The reading list for this module can be accessed at [readinglists.uwe.ac.uk](https://rl.talis.com/3/uwe/lists/939B5869-07ED-B7EC-90C9-B237896D16B9.html?lang=en-US) via the following link <https://rl.talis.com/3/uwe/lists/939B5869-07ED-B7EC-90C9-B237896D16B9.html?lang=en-US>

Part 4: Assessment

Assessment strategy: The coursework comprises a group project assignment with three parts. 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. The mark is a group mark, adjusted according to department group work guidelines following peer assessment of individual contributions to the group. The final part is a short individual reflection (approx. 500 words) on the benefits and challenges of conducting this as a group process.

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

The resit 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 reflection, as above, on how the individual project would have benefited or been hindered by being undertaken and managed as a group project.

Assessment components:

Group work - Component A (First Sit)

Description: Group project: including problem analysis, requirements specification, design of a solution, coding and testing. Work to be split among group members.

Report to include a 500 word reflection on the benefits and challenges of completing this as a group.

Weighting: 100 %

Final assessment: No

Group work: Yes

Learning outcomes tested: MO1, MO2, MO3, MO4, MO5, MO6

Project - Component A (Resit)

Description: 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. The individual reflection is longer, as the student may not actually have undertaken this in a group, and therefore needs to spend more time on reflection.

Weighting: 100 %

Final assessment: No

Group work: No

Learning outcomes tested: MO1, MO2, MO3, MO4, MO5, MO6

Part 5: Contributes towards

This module contributes towards the following programmes of study:

Computer Science [Sep][FT][Frenchay][3yrs] BSc (Hons) 2020-21

Computer Science [Sep][FT][Frenchay][3yrs] BSc (Hons) 2020-21

Computer Science [Sep][SW][Frenchay][4yrs] BSc (Hons) 2020-21

Computer Science [Sep][SW][Frenchay][4yrs] BSc (Hons) 2020-21