Module Specification

Software Development Project

Version: 2024-25, v3.0, 20 May 2024

Contents

Module Specification **1**

Part 1: Information **2**

Part 2: Description **2**

Part 3: Teaching and learning methods **4**

Part 4: Assessment **5**

Part 5: Contributes towards **7**

Part 1: Information

Module title: Software Development Project

Module code: UFCFFF-30-3

Level: Level 6

For implementation from: 2024-25

UWE credit rating: 30

ECTS credit rating: 15

College: College of Arts, Technology and Environment

School: CATE School of Computing and Creative Technologies

Partner institutions: None

Field: Computer Science and Creative Technologies

Module type: Module

Pre-requisites: None

Excluded combinations: None

Co-requisites: None

Continuing professional development: No

Professional, statutory or regulatory body requirements: None

Part 2: Description

Overview: Not applicable

Features: Not applicable

Educational aims: See Learning Outcomes

Outline syllabus: There is no specific syllabus for this module as the project is an individual piece of work, exploring an idea from conception through to realisation. Nonetheless, elements of the project process are covered in a short lecture series at

the start of the academic year. The lectures will normally be delivered by the module leader or their nominee. and covers topics such as:

Choosing a project
Researching the project idea
Making use of your supervisor
Moving from research to requirements
Configuration and release management and version control
Quality planning and assurance
Writing up the project.

In parallel with the lecture series, students will be allocated a supervisor. They will then agree the subject of the project with the supervisor and the Module Leader. Suitable topics may stem from staff, the student, the student's employer or other outside organisations. The topic must lend itself to research followed by a software development process based on the research.

The research component will include the identification of a suitable topic and subsequent investigation from books, papers and other sources.

Requirements should be derived from the research. Software development will include the identification of suitable tools and methodologies to use, giving full and careful attention to the issues of quality and risk.

Students will be expected to demonstrate that they understand the importance of version control as it applies to software and systems; have defined suitable configuration management processes for use throughout the product development lifecycle in storing software deliverables and controlling and tracking changes to their software; have used configuration management and version control tools and release management frameworks and applied sound change management processes properly and effectively when modifying their software designs and deliverables.

Whatever the subject the student will be expected to treat material critically and to

demonstrate their understanding of the relevance of material both to their award to the project topic. They will also be expected to reflect on the tools and methodologies used and, at the project completion, comment on their suitability.

Part 3: Teaching and learning methods

Teaching and learning methods: Contact time for this module consists of two elements. At the start of the module, there will be a series of lectures to the whole group of Software Development Project students, amounting to approximately 4 hours of contact time. Subsequently, students
will meet with their individual supervisors. The precise nature and timing of the meetings will vary with the student's progress.

Each student will be assigned a supervisor who will meet them regularly to discuss progress and to give guidance on planning and managing the work. It is the student's responsibility to research material and techniques appropriate to the subject of the project.

Wherever possible students will be assigned a supervisor with an interest in the project topic but this cannot be guaranteed. The responsibilities of the tutor are primarily to provide guidance on the management of the project, the standard of work required, what can realistically be done in the available time and to give feedback on work done (including the writing of the report).

In the initial stages of the project the student and their tutor will discuss objectives which must be achieved if the project is to receive a pass grade. Criteria which must be met for a higher grade will also be identified. (Projects develop unpredictably, the objectives are only intended a guide to the level expected and details may change).

At the beginning of the academic year in which the project is undertaken, a short series of lectures will provide the student with the context in which the project is to be undertaken.

Scheduled learning therefore includes lectures and project supervision.

Independent learning includes hours engaged with essential research, the development of requirements, design, programme code, programme testing and debugging, preparation and completion of the project report etc.

Module Learning outcomes: On successful completion of this module students will achieve the following learning outcomes.

MO1 Work on a real-life problem, synthesising and critically evaluating information from multiple sources including academic and commercial documents.

MO2 Follow a development lifecycle from an initial idea through to the realisation of a software artefact using appropriate tools and methodologies.

MO3 Communicate the artefact developed and the process by which it was produced.

MO4 Demonstrate the artefact developed and discuss its design, attributes and shortcomings.

Hours to be allocated: 300

Contact hours:

Independent study/self-guided study = 280 hours

Face-to-face learning = 20 hours

Reading list: The reading list for this module can be accessed at readinglists.uwe.ac.uk via the following link https://uwe.rl.talis.com/modules/ufcfff-30-3.html

Part 4: Assessment

Assessment strategy: There are two assessments on the Software Development project. Assessment will be equally divided between reporting and demonstration of product functionality. The final report (6000 words) will be supported by additional material in the form of software and documentation as appropriate. The report is

submitted upon completion of the project and carries 50% of the available marks. The report will typically be assessed on the following criteria:

Extent, level and relevance of research
Requirements analysis
Identification and application of an appropriate development/ design methodology
Choice and application of technology to implementation.
Evidence of self-management and critical reflection on the project content and process
Clarity of exposition within the report

After submission of the project report, students will demonstrate, discuss and defend their software to their supervisor and second assessor. 50% of the available marks are attached to this demonstration which will be accompanied by a short plan of the demonstration session laid out in a short series of MS PowerPoint slides. The demonstration will typically be assessed on the following criteria:

Clear link between function and requirements analysis
Integration of design, function and intended audience
Clarity of code and modular design within the software

Students referred in the report assessment will submit a project report, with reworking based on feedback from their initial submission.

Students referred in the demonstration assessment may elect to demonstrate their work either in person, or using webcasting software (such as that employed for the Guru Lecture series: currently Panopto).

Both assessments are individual work; advice on plagiarism and how to avoid it together with regular formative feedback will be given as and when students meet with their individual supervisors. The precise nature and timing of the meetings will vary with the student’s progress.

The resit strategy is the same as the first sit with the project being reworked.

Assessment tasks:

Report (First Sit)
Description: Project report (6,000 words)
Weighting: 50 %
Final assessment: Yes
Group work: No
Learning outcomes tested: MO1, MO2, MO3

Practical Skills Assessment (First Sit)
Description: Demonstration (30 mins)
Weighting: 50 %
Final assessment: No
Group work: No
Learning outcomes tested: MO1, MO2, MO4

Report (Resit)
Description: Project report (6000 words)
Weighting: 50 %
Final assessment: Yes
Group work: No
Learning outcomes tested: MO1, MO2, MO3

Practical Skills Assessment (Resit)
Description: Demonstration (30 mins)
Weighting: 50 %
Final assessment: No
Group work: No
Learning outcomes tested: MO1, MO2, MO4

Part 5: Contributes towards

This module contributes towards the following programmes of study:

Software Engineering for Business {Foundation} [Sep][SW][Frenchay][5yrs] BSc (Hons) 2020-21

Software Engineering for Business {JEP}[Sep][FT][Neusoft][4yrs] BSc (Hons) 2021-22

Business Computing {Foundation} [Feb][FT][GCET][4yrs] BSc (Hons) 2021-22

Business Computing {Foundation} [Oct][FT][GCET][4yrs] BSc (Hons) 2021-22

Business Computing {Foundation} [Sep][FT][Frenchay][4yrs] BSc (Hons) 2021-22

Software Engineering for Business [Sep][SW][Frenchay][4yrs] BSc (Hons) 2021-22

Software Engineering for Business {Foundation} [Sep][FT][Frenchay][4yrs] BSc (Hons) 2021-22

Software Engineering for Business [Frenchay] BSc (Hons) 2022-23

Business Computing {Foundation} [Sep][SW][Frenchay][5yrs] BSc (Hons) 2020-21

Business Computing [Sep][SW][Frenchay][4yrs] BSc (Hons) 2021-22

Business Computing [Frenchay] BSc (Hons) 2022-23