Module Specification

Object-Oriented Analysis, Design and Programming

Version: 2023-24, v2.0, 20 Mar 2023

Contents

Part 1: Information

Module title: Object-Oriented Analysis, Design and Programming

Module code: UFCFPD-15-M

Level: Level 7

For implementation from: 2023-24

UWE credit rating: 15

ECTS credit rating: 7.5

Faculty: Faculty of Environment & Technology

Department: FET Dept of Computer Sci & Creative Tech

Partner institutions: None

Delivery locations: Not in use for Modules

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: Analysis and Design:

There is a strong emphasis on the critical evaluation aspect of the design instead of simply being able to do designs. The specific areas include:

Design criteria
Use cases
Classes and objects
Inheritance, Abstract classes and Interfaces
Class relationships
Interaction modelling

Programming: There is an emphasis on the evaluation of language features. Specific features include:

Fundamental:
Classes and objects, message passing
Inheritance, Abstract classes and Interfaces, polymorphism

Concurrency and Networking:
Thread, Thread synchronisation
Client server programming

Database:
Fundamental concepts and implementation

Part 3: Teaching and learning methods

Teaching and learning methods: Scheduled learning There are 3 hours scheduled “lectorial”-style classes weekly which include interactive lectures, tutorials and practical sessions, where in the theory and practice of object-oriented systems design and development are demonstrated; questions are invited and freely discussed. Students are encouraged to articulate and present their analysis and design models of some case studies, as well as carrying out programming tasks. All lecture slides and other relevant learning materials are available on the Blackboard VLE.

Independent learning Students are expected to spend about 6 hours per week to engage with essential reading, case study preparation, assignment work and completing the weekly tasks set by the module.

Contact Hours:

Activity:
Contact time: 36 hours
Assimilation and development of knowledge (independent learning): 74 hours
Coursework and exam preparation: 40 hours
Total study time: 150 hours

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

MO1 Understand and have knowledge of the typical characteristics of object-oriented software systems

MO2 Understand and have knowledge of software design criteria

MO3 Apply object-oriented analysis and design techniques to analyse and design object-oriented system

MO4 Critically evaluate Java language features and apply Java-programming skills to effectively design and implement object-oriented software solutions

MO5 Critically evaluate software design with respect to design criteria

Hours to be allocated: 150

Contact hours:

Independent study/self-guided study = 114 hours

Face-to-face learning = 36 hours

Total = 150

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/ufcfpd-15-m.html

Part 4: Assessment

Assessment strategy: The assessment strategy will consist of one coursework assessment and one examination. The coursework assessment will be based on work covered in lectures and tutorials. The examination will be based on topics covered in the lectures relating to the learning outcomes.

Summative assessment:
Examination, 2 hours, questions will focus on the design aspect involving students demonstrate their understanding of the concepts, their ability to apply the concepts to produce designs and their ability to critically evaluate the designs.

Coursework will be around a case study. Each student will be expected to design and implement a solution based on some given cases studies and to explain their design and implementation choices.

Formative assessment:
Each week, tasks will be set to re-enforce the concepts covered in the lectures. The weekly classes will be used to explore and discuss software design concepts and students are expected to complete programming tasks. Feedback will be given.

Assessment components:

Examination (First Sit)
Description: Written examination (2 hours final assessment)
Weighting: 50 %
Final assessment: Yes
Group work: No
Learning outcomes tested: MO1, MO2, MO3, MO4, MO5

Written Assignment (First Sit)
Description: Coursework (design and implement a software system)
Weighting: 50 %
Final assessment: No

Group work: No

Learning outcomes tested: MO3, MO4

Examination (Resit)

Description: Written examination (2 hours final assessment)

Weighting: 50 %

Final assessment: Yes

Group work: No

Learning outcomes tested:

Written Assignment (Resit)

Description: Coursework (design and implement a software system)

Weighting: 50 %

Final assessment: No

Group work: No

Learning outcomes tested:

Part 5: Contributes towards

This module contributes towards the following programmes of study: