Module Specification

Object-Oriented Development [TSI]

Version: 2023-24, v1.0, 09 Aug 2023

Contents

Part 1: Information

Module title: Object-Oriented Development [TSI]

Module code: UFCE3W-24-1

Level: Level 4

For implementation from: 2023-24

UWE credit rating: 24

ECTS credit rating: 12

College: College of Arts, Technology and Environment

School: CATE School of Computing and Creative Technologies

Partner institutions: Transport and Telecommunication Institute

Field:

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: This Object-oriented Development course offers a comprehensive study of object-oriented programming principles and techniques. It covers essential topics such as encapsulation, inheritance, and polymorphism, providing students with a strong understanding of designing and implementing software solutions using object-oriented concepts. Students will gain practical experience through hands-on exercises, utilizing industry-standard tools and frameworks. The course emphasizes critical thinking and problem-solving skills, enabling students to analyze complex software requirements and develop effective solutions. By the end of the course, students will be equipped with the knowledge and skills necessary to create robust and maintainable software applications using object-oriented development paradigms.

Features: Not applicable

Educational aims: The Object-oriented Development course aims to provide students with a strong foundation in object-oriented programming principles and techniques. Students will learn to design and implement software solutions using encapsulation, inheritance, and polymorphism, and develop critical thinking skills to create scalable and maintainable applications.

Outline syllabus: The module consists of two units:
UNIT 1:
-Overview of the OO Paradigm. Classes and Encapsulation Methods and polymorphism, Inheritance and Abstraction, Interfaces and abstract classes.
-Instances, Events and Delegates, Exceptions.
-Data Binding. Graphical User Interfaces Programming, I/O, Files, Serialisation.
-NET Framework Overview.
-OO Design with UML, Design Patterns.
-Reflection.
-ASP.NET Overview, ADO.NET Overview.
-Bit operations and strings, Operator Overloading.

UNIT 2:
-Technical writing, skills common to writers. Designing a technical document: situational analysis, arrangement.
-Drafting, revising and editing, copy marking and proofreading, tracking revisions and version control.
-Writing style, accuracy (factual and technical), usability, Document style guides.
-Writing and editing for specific audiences - readability, terminology, use of illustrations and other non-textual elements, layout suited to the purpose.
-Disability awareness in writing and editing.
-Writing and editing for an international audience: language considerations, cultural

considerations, technical differences.
-Writing and editing issues specific to different kinds of technical writing.

Part 3: Teaching and learning methods

Teaching and learning methods: Scheduled Learning
Materials will be introduced via lectures. Students will use lab time to develop practical programming exercises based on the lecture material.. During the lab time, students will receive feedback and support from the tutors and peers who will be available in the room to answer queries.

Independent Learning
Students are expected to learn independently by carrying out reading and practical exercises outside of taught classes.

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

MO1 Apply the core principles of object-oriented programming to implement solutions to problems.

MO2 Use industry-standard tools and frameworks for object- oriented development to create scalable and maintainable software applications.

MO3 Analyse software requirements and design effective solutions using object oriented development paradigms.

MO4 Show a detailed knowledge and understanding of technical writing and technical editing, including awareness of personal responsibility and relevant professional and ethical issues.

Hours to be allocated: 240

Contact hours:

Independent study/self-guided study = 192 hours

Face-to-face learning = 128 hours

Total = 320

Reading list: The reading list for this module can be accessed at readinglists.uwe.ac.uk via the following link https://rl.talis.com/3/uwe/lists/465FC014-91F0-779C-6318-48D261BC076F.html?lang=en&login=1

Part 4: Assessment

Assessment strategy: This module is assessed in three parts,
1. Students independently complete 3 time restricted labs which test their OOP abilities as the module develops,
2. Students will design and undertake a project where they develop OOP software for a given case study, (developers)
3. Students will produce a user manual and accompanying documentation for the developed project, (end users)

The resits will be like for like, with the exception of the project which will follow a new case study.

Assessment tasks:

Laboratory Report (First Sit)
Description: Series of 3 developed independent labs dedicated to OOP concept. Completed in specific time frames, without any support from teaching staff
Weighting: 20 %
Final assessment: No
Group work: No
Learning outcomes tested: MO1, MO2, MO3

Written Assignment (First Sit)
Description: To produce the technical report for the project. (max 3000 words)
Weighting: 30 %
Final assessment: Yes
Group work: No
Learning outcomes tested: MO2, MO3, MO4

Project (First Sit)
Description: Individual project, which is targeted on development software using OO paradigm, as well as document the software. Report around 3000 word is expected as deliverable. (Max words 3300 words).
Weighting: 50 %
Final assessment: No
Group work: No
Learning outcomes tested: MO1, MO2, MO3, MO4

Laboratory Report (Resit)
Description: Series of 3 developed independent labs dedicated to OOP concept. Completed in specific time frames, without any support from teaching staff
Weighting: 20 %
Final assessment: No
Group work: No
Learning outcomes tested: MO1, MO2, MO3

Written Assignment (Resit)
Description: To produce the technical report for the project. (Max 3000 words)
Weighting: 30 %
Final assessment: Yes
Group work: No
Learning outcomes tested: MO2, MO3, MO4

Project (Resit)
Description: Individual project, which is targeted on development software using OO paradigm, as well as document the software. Report around 3000 word is expected as deliverable. (Max 3300 words)
Weighting: 50 %
Final assessment: No
Group work: No
Learning outcomes tested: MO1, MO2, MO3, MO4

Part 5: Contributes towards

This module contributes towards the following programmes of study:

Computer Science and Software Development {Double Degree} {Foundation} [TSI] BSc (Hons) 2022-23

Computer Science and Software Development {Double Degree} {Foundation} [TSI] BSc (Hons) 2022-23