Liverpool John Moores University

Title:	Object-Oriented Systems
Status:	Definitive
Code:	**5204COMP** (127983)
Version Start Date:	01-08-2021
Owning School/Faculty:	Computer Science and Mathematics
Teaching School/Faculty:	Computer Science and Mathematics

Team	Leader
Glyn Hughes	Y
Sorren Hanvey	
David Lamb	

Academic Level:	FHEQ5	**Credit Value:**	20	**Total Delivered Hours:**	44
Total Learning Hours:	200	**Private Study:**	156		

Delivery Options
Course typically offered: Semester 2

Component	Contact Hours
Lecture	22
Practical	22

Grading Basis: 40 %

Assessment Details

Category	Short Description	Description	Weighting (%)	Exam Duration
Artefacts	AS1	Design & Development of OO Systems	60	
Artefacts	AS2	Applying Principles & Patterns in OO Systems Design & Development (GRP)	40	

Aims

To investigate the underpinnings of object orientated systems.
To develop object oriented systems using the unified modelling language and object oriented program code.

To apply principles and patterns to improve the flexibility and maintainability of object oriented systems, with test driven development and source control playing a supporting role.
To investigate the various architectures that object oriented systems may embrace.

Learning Outcomes

After completing the module the student should be able to:

1	Specify object oriented designs using the unified modelling language.
2	Develop object oriented designs using object oriented program code.
3	Apply principles and patterns to improve the flexibility and maintainability of object oriented designs & program code.
4	Employ test driven development and source control in software engineering.

Learning Outcomes of Assessments

The assessment item list is assessed via the learning outcomes listed:

Design & Development	1	2
Applying Principles & Patterns	3	4

Outline Syllabus

OOAD & OOP
-Compositions, Aggregations & Associations
-Inheritance & Polymorphism
-Collections & Generics
-Interfaces
-Multi-Threading

Principles & Patterns
-e.g. Inversion of Control, Dependency Inversion, Factory, Strategy
-Applying SOLID

Object Relational Mapping
-Architectures
-e.g. event / data driven vs. responsibility driven

Test Driven Dev.
-Source Control

Learning Activities

Learning activities include hybrid lectures / tutorials where students are encouraged to ask questions / discuss scenarios and supported labs where students are encouraged to put theory gained though lectures / tutorials into practice. Directed

reading against appropriate industry and research sources further reinforces learning.

Notes

The module begins by exploring the object oriented paradigm from analysis and design through implementation and testing to maintenance. The UML (Unified Modelling Language) is employed alongside OOP (Object Oriented Programming) to demonstrate key concepts, resulting in mature, fully functioning object oriented systems. The module continues by applying principles and patterns to object oriented systems with test driven development and source control playing a supporting role. The module concludes by exploring the various architectures that object oriented systems may embrace. This module thusly represents the logical follow-on to NQF4's Introducing Programming module.