H8OOS: Object Oriented Software Engineering

Module Code:	H8OOS
Long Title	Object Oriented Software Engineering APPROVED
Title	Object Oriented Software Engineering
Module Level:	LEVEL 8
EQF Level:	6
EHEA Level:	First Cycle
Credits:	5
Module Coordinator:	ANTHONY PAUL STYNES
Module Author:	ANTHONY PAUL STYNES
Departments:	School of Computing
Specifications of the qualifications and experience required of staff	Master's degree in computing or cognate discipline. Proposed lecturer: Dr Paul Stynes

Learning Outcomes	
On successful completion of this module the learner will be able to:	
#	**Learning Outcome Description**
LO1	Demonstrate the conceptual, practical and technical skills of planning and monitoring a project plan using an appropriate CASE tool
LO2	Describe in detail the theory, concepts and methods pertaining to Software Engineering such as Agile and UML.
LO3	Create requirements using use case modelling concepts.
LO4	Demonstrate conceptual and technical skills in the analysis, design, implementation and test of a software development solution individually or as part of a team.
LO5	Employ tools and techniques for Software Engineering,

Dependencies
Module Recommendations
No recommendations listed
Co-requisite Modules
No Co-requisite modules listed

Entry requirements	See section 4.2 Entry procedures and criteria for the programme including procedures recognition of prior learning

H8OOS: Object Oriented Software Engineering

Module Content & Assessment

Indicative Content
Software Engineering • Principles of Software Engineering. • Software Product. • Software Process.
Software Project Management • Project Management Concepts • Process, Project and Product Metrics • Software Project Organization • Recruiting and Staffing projects • Project roles and the project team • Project planning methods • Risk analysis and risk management • Project scheduling and Tracking • Project inspections and walkthroughs • Coping with change – change models • Software Quality Assurance
Requirements Engineering • Use case modelling. • Actors Use Cases. • Anatomy of use cases. • Advanced Use Case Concepts.
Structural Diagrams • Class Diagram. • Component Diagram. • Package Diagram. • Deployment Diagram.
Behavioral Diagrams • Sequence Diagram. • Communication Diagram. • Statechart.
Design Patterns • Expert Pattern. • Creator Pattern. • Controller Pattern. • Three tier Architectures. • Multi-tiered Architectures. • Model view separation pattern. • Mapping to code.
Testing • Software testing strategies. • Software Testing Techniques. • Unit Testing.
Test Driven Development • Test Driven Development (TDD) Approach.
Agile Process Models • Agility Agile Process. • Agile Process Models. • Overview of the SCRUM process.

Assessment Breakdown	%
Coursework	100.00%

Assessments

Full Time

Coursework

Assessment Type:	Project	**% of total:**	50
Assessment Date:	Week 8	**Outcome addressed:**	1,2,3,4,5
Non-Marked:	No		

Assessment Description:
Team project requiring learners to project plan a project and apply UML diagrams to the resolution of a real life problem.

Assessment Type:	Project	**% of total:**	50
Assessment Date:	Sem 1 End	**Outcome addressed:**	2,4,5
Non-Marked:	No		

Assessment Description:
Team project requiring learners to apply agile methods such as TDD and SCRUM to the resolution of a real life problem.

No End of Module Assessment

No Workplace Assessment

Reassessment Requirement

Coursework Only
This module is reassessed solely on the basis of re-submitted coursework. There is no repeat written examination.

Reassessment Description
This module is reassessed solely on the basis of re-submitted coursework. There is no repeat written examination. Learners who fail this module will be required to sit a repeat module assessment where all learning outcomes will be examined.

H8OOS: Object Oriented Software Engineering

Module Workload

Module Target Workload Hours 0 Hours

Workload: Full Time

Workload Type	*Workload Description*	*Hours*	*Frequency*	*Average Weekly Learner Workload*
Lecture	No Description	24	Per Semester	2.00
Tutorial	No Description	12	Per Semester	1.00
Independent Learning	No Description	89	Per Semester	7.42
			Total Weekly Contact Hours	3.00

Workload: Part Time

Workload Type	*Workload Description*	*Hours*	*Frequency*	*Average Weekly Learner Workload*
Lecture	No Description	24	Per Semester	2.00
Tutorial	No Description	12	Per Semester	1.00
Independent Learning Time	No Description	89	Per Semester	7.42
			Total Weekly Contact Hours	3.00

Module Resources

Recommended Book Resources

Alan Dennis, Barbara Haley Wixom, David Tegarden. (2015), Systems Analysis and Design with UML, 5. Wiley.

Shore, J., and Warden, S. (2014), The Art of Agile Development, 1. O'Reilly Media.

Kent Beck. (2003), Test-driven Development, Addison-Wesley Professional, p.220, [ISBN: 9780321146533].

Supplementary Book Resources

Axel van Lamsweerde. (2009), quirements Engineering: From System Goals to UML Models to Software Specifications, Wiley.

Dragan Milicev. (2009), Model-Driven Development with Executable UML, Wiley.

Eric Baude, Michael Bernstein. (2011), Software Engineering Modern Approaches, 2. Wiley.

Roger S. Pressman. (2010), Software Engineering: A practioners approach, 7. McGraw and Hill.

This module does not have any article/paper resources

Other Resources

[Website], http://www-01.ibm.com/software/rational/ uml/.

Discussion Note:	