

H6SWE: Software Engineering

Module Code:	H6SWE
Long Title	Software Engineering APPROVED
Title	Software Engineering
Module Level:	LEVEL 6
EQF Level:	5
EHEA Level:	Short Cycle
Credits:	5
Module Coordinator:	
Module Author:	Alex Courtney
Departments:	School of Computing
Specifications of the qualifications and experience required of staff	Master's degree in computing or cognate discipline.
Learning Outcomes	
<i>On successful completion of this module the learner will be able to:</i>	
#	Learning Outcome Description
LO1	Describe the theory, concepts and methods pertaining to Software Engineering such as Agile and UML
LO2	Create requirements using Use Case modelling concepts.
LO3	Collaboratively analyse, design, implement, test and present a software development solution
LO4	Employ tools and techniques for Object Oriented Software Engineering,
Dependencies	
Module Recommendations	
No recommendations listed	
Co-requisite Modules	
No Co-requisite modules listed	
Entry requirements	Learners should have attained the knowledge, skills and competence gained from stage 1 of the BSc (Hons) in Computing.

H6SWE: Software Engineering

Module Content & Assessment			
Indicative Content			
Software Engineering Principles of Software Engineering.. Software Product.. Software Process.			
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.			
Behavioural 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:	430	% of total:	0
Assessment Date:	n/a	Outcome addressed:	
Non-Marked:	No		
Assessment Description: n/a			
Assessment Type:	Formative Assessment	% of total:	Non-Marked
Assessment Date:	n/a	Outcome addressed:	1,2,3,4
Non-Marked:	Yes		
Assessment Description: Ongoing feedback on ongoing tutorial activities. Feedback on regular reflection.			
Assessment Type:	Project	% of total:	50
Assessment Date:	n/a	Outcome addressed:	1,2,3,4
Non-Marked:	No		
Assessment Description: Team project requiring learners to apply UML diagrams to the resolution of a real-life problem.			
Assessment Type:	Project	% of total:	50
Assessment Date:	n/a	Outcome addressed:	1,2,3,4
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			
Repeat examination <i>Reassessment of this module will consist of a repeat examination. It is possible that there will also be a requirement to be reassessed in a coursework element.</i>			
Reassessment Description Coursework Only 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.			

H6SWE: Software Engineering

Module Workload				
Module Target Workload Hours 0 Hours				
Workload: Full Time				
Workload Type	Workload Description	Hours	Frequency	Average Weekly Learner Workload
Lecture	Classroom & Demonstrations (hours)	24	Every Week	24.00
Tutorial	Other hours (Practical/Tutorial)	12	Every Week	12.00
Independent Learning	Independent learning (hours)	89	Every Week	89.00
Total Weekly Contact Hours				36.00

Module Resources	
Recommended Book Resources	
<p>Kent Beck. (2003), Test-driven Development, Addison-Wesley Professional, p.220, [ISBN: 9780321146533].</p> <p>Alan Dennis, Barbara Haley Wixom, David Tegarden. (2015), Systems Analysis and Design, John Wiley & Sons, p.546, [ISBN: 1118804678].</p> <p>James Shore, Shane Warden. (2014), Systems Analysis and Design, 3rd. John Wiley & Sons, p.p.546, [ISBN: 978-059652767].</p>	
Supplementary Book Resources	
<p>Fowler, M. (2003), UML Distilled, Addison-Wesley.</p> <p>Eriksson, H.E., Penker, M., Lyons, B., and Fado, D. (2003), UML 2 Toolkit, Wiley.</p> <p>Dragan Milicev. (2009), Model-Driven Development with Executable UML, Wiley.</p> <p>Axel van Lamsweerde. (2009), Requirements Engineering: From System Goals to UML Models to Software Specifications.</p> <p>Roger S. Pressman. (2010), Software Engineering: A practitioners approach, 7th Edition, McGraw and Hill.</p> <p>Eric Baude, Michael Bernstein. (2011), Software Engineering Modern Approaches, 2nd edition, Wiley.</p>	
This module does not have any article/paper resources	
This module does not have any other resources	
Discussion Note:	