H9EEAI: Engineering and Evaluating Artificial Intelligence Systems

Module Code:	H9EEAI
Long Title	Engineering and Evaluating Artificial Intelligence Systems APPROVED
Title	Engineering and Evaluating Artificial Intelligence Systems
Module Level:	LEVEL 9
EQF Level:	7
EHEA Level:	Second Cycle

Credits:	5

Module Coordinator:	Rejwanul Haque

Module Author:	Shauni Hegarty

Departments:	School of Computing

Specifications of the qualifications and experience required of staff	

Learning Outcomes

On successful completion of this module the learner will be able to:

#	Learning Outcome Description
LO1	Comprehend, contrast, assess, and apply software architecture principles in the design of AI systems.
LO2	Theoretically evaluate the AI systems in terms of completeness, complexity, and admissibility.
LO3	Evaluate, summarise, critique, and present the quality and performance of AI systems.
LO4	Determine and critique the infrastructure for the deployment of AI systems.

Dependencies

Module Recommendations

No recommendations listed

Co-requisite Modules

No Co-requisite modules listed

Entry requirements	

H9EEAI: Engineering and Evaluating Artificial Intelligence Systems

Module Content & Assessment

Indicative Content

Artificial Intelligence Engineering
•Overview of AI Engineering•Software Development Life Cycle•Agile Development for AI

Software Architecture
Software Architecture•Architectural Structures and Views•What Makes a "Good" Architecture?•Importance of Software Architecture•Contexts of Software Architecture•Understanding Quality Attributes•Specifying Quality Attribute Requirements•Quality Attributes (i.e., Availability, Interoperability, Modifiability, Performance, Security, Testability, Usability, Deployability, Energy, and Safety)

Architectural Solutions
•Architectural Patterns•Architecting in the Cloud•Architecture for Machine Learning and Artificial Intelligence

Designing the Architecture
•Architecture in Agile Projects•Design Strategy•Attributed-Driven Design (ADD) Method•Steps of ADD

Documenting Architecture
•Notation for Architecture Documentation•Views•Documenting Behaviour•Architecture Documentation and Quality Attributes•Documenting in Agile Projects

Architecture Evaluation
•Evaluation Factors•The Architecture Trade-off Analysis Method•Lightweight Architecture Evaluation

Computational Complexity
•Basic Conventions•Big O Notation•Deterministic and Non-Deterministic Models of Computation•Class P, NP and NP Complete•Coping with NP Hardness•Time and Space Complexity

Evaluating Algorithms
•Admissibility of a Heuristic•Completeness•Evaluating Algorithm Correctness•Sensitivity Analysis

Software Quality Assurance
•Software Quality Characteristics•Software Testing Life Cycle•Types of Testing Techniques (e.g., Black-Box Testing, White-Box Testing)

Deployment Pipeline
•Introduction to DevOps•Building and Testing•Deployment Strategies

Infrastructure Design
•Infrastructure Design, Testing, and Reuse•Modularity and Infrastructure Churn•Scalability•Distributed System

Assessment Breakdown	%
Coursework	50.00%
End of Module Assessment	50.00%

Assessments

Full Time

Coursework

Assessment Type:	Formative Assessment	**% of total:**	Non-Marked
Assessment Date:	n/a	**Outcome addressed:**	1,2,3,4
Non-Marked:	Yes		

Assessment Description:
Formative assessment will be provided on the in-class individual or group activities. Feedback will be provided in written or oral format, or on-line through Moodle. In addition, in class discussions will be undertaken as part of the practical approach to learning.

Assessment Type:	Continuous Assessment	**% of total:**	50
Assessment Date:	n/a	**Outcome addressed:**	1,3
Non-Marked:	No		

Assessment Description:
This assessment will evaluate the learners’ comprehension of software architecture principles and skills for applying that knowledge to design AI systems. Learners will be provided with a description of an AI system or case study. Learners will be required to identify the minimum architectural and software quality requirements, design and document a software architecture for the proposed AI system.

End of Module Assessment

Assessment Type:	Terminal Exam	**% of total:**	50
Assessment Date:	End-of-Semester	**Outcome addressed:**	1,2,3,4
Non-Marked:	No		

Assessment Description:
The examination will be of two hours duration and may include a mix of: theoretical, applied and interpretation questions.

No Workplace Assessment

Reassessment Requirement

Repeat examination
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.

Module Workload

Module Target Workload Hours 0 Hours

Workload: Full Time

Workload Type	*Workload Description*	*Hours*	*Frequency*	*Average Weekly Learner Workload*
Lecture	Lectures	24	Per Semester	2.00
Independent Learning	Independent Learning	89	Per Semester	7.42
Tutorial	Tutorials	12	Per Semester	1.00
			Total Weekly Contact Hours	3.00

Module Resources

Recommended Book Resources

Bass, L., Clements, P., & Kazman, R. (2022). Software Architecture in Practice(4th ed.). Addison-Wesley Professional. SEI Series in Software Engineering. [ISBN: 978-0136886099].

Bass, L., Weber, I., & Zhu, L. (2016). DevOps: A Software Architect’s Perspective. Addison-Wesley Professional. SEI Series in Software Engineering. [ISBN: 978-9332570375].

Mahfuz, A. S. (2016). Software Quality Assurance: Integrating Testing, Security, and Audit. Auerbach Publications. [ISBN: 978-1498735537].

Supplementary Book Resources

Arora, S. & Barak, B. (2009). Computational Complexity: A Modern Approach. Cambridge University Press. [ISBN: 978-0521424264].

Cervantes, H. & Kazman, R. (2016). Designing Software Architectures: A Practical Approach. Addison-Wesley Professional. [ISBN: 978-0134390789].

Hulten, G. (2018). Building Intelligent Systems: A Guide to Machine Learning Engineering. Apress. [ISBN: 978-1484234310].

Humble, J. & Farley, D. (2010). Continuous Delivery: Reliable Software Releases through Build, Test, and Deployment Automation. Addison-Wesley Professional.[ISBN: 978-0321601919].

Richards, M. & Ford, N. (2020). Fundamentals of Software Architecture: An Engineering Approach. O’Reilly. [ISBN: 978-1492043454].

Tarlinder, A. (2016). Developer Testing: Building Quality into Software. Addison-Wesley Professional.[ISBN: 978-0134291062].

This module does not have any article/paper resources

This module does not have any other resources

Discussion Note:	