COURSE SPECIFICATION DOCUMENT

Academic Department:	Dept Science, Innovation & Technology
Programme:	Software Engineering (AI)
***FHEQ* Level:**	6
Course Title:	Advanced Software Development with AI
Course Code:	SENG 6101
Student Engagement Hours:	160 (Standard 4- credit BA Course)
Total Hours:	160
Timetabled Hours:	45
Guided Learning Hours:	15
Independent Learning Hours:	100
Credits:	16 UK CATS credits 8 ECTS credits 4 US credits

Course Description:
This course builds on the course Intermediate Software Development with AI. It explores principles of machine learning and deep learning to understand AI and enables students to use advance C++ programming for AI combining manual and AI generated code. Further, the course will also explore profiles of a range of programmers from diverse backgrounds.

Prerequisites:
70 credits + SENG 5101 Intermediate Software Development with AI

Aims and Objectives:
By the end of this course, students will be able to use deep learning and machine learning to understand how AI works and create comprehensive software. They will be able to explain and analyse existing code and write advance C++ code for comprehensive applications. They will also be able to update and maintain code.

Programme Outcomes:

L6: AI, AII, BI, BIII, CII, DIII

A detailed list of the programme outcomes are found in the Programme Specification.

This is located at the archive maintained by Registry and found at:
https://www.richmond.ac.uk/programme-and-course-specifications/

Learning Outcomes:

By the end of this course, successful students should be able to:

- Critically analyse the profiles of different types of programmers and their roles within software development teams.
- Demonstrate an in-depth understanding of machine learning and deep learning principles, and evaluate their applications in solving complex problems.
- Critically analyse, extend, and refactor existing AI code to improve functionality and performance.
- Explain and critically evaluate existing coding frameworks and libraries, and apply them effectively to design solutions for complex AI problems.
- Develop, test, debug, and optimise AI applications, demonstrating a high level of proficiency in advanced programming techniques.

Indicative Content:

- Machine Learning
- Deep Learning
- Updating and maintaining code
- Advanced C++ eg. memory management, lamda functions

Diverse Programmers and their profiles

Assessment:

This course conforms to the University Assessment Norms approved at Academic Board and located at: https://www.richmond.ac.uk/university-policies/

Teaching Methodology:

This course will be delivered face to face through a combination of lectures and interactive sessions. In addition to classroom activities, there are guided learning elements that are tutor led and arranged through Blackboard. These activities can be asynchronous online sessions, flipped classrooms, set readings with discussion boards or set guest lectures for example. Set activities are monitored by the instructor to ascertain student engagement. Students are encouraged to prepare for class and to play an active part, to raise questions, following-up ideas and interact with a wide range of provided material.

Indicative Text(s):

Dooley, J.F., Kazakova, V.A., (2024) *Software Development, Design, and Coding: With Patterns, Debugging, Unit Testing, and Refactoring*. 3rd edn. Apress.

Kolodiazhnyi, K. (2025) *Hands-On Machine Learning with C++. Build, train, and deploy end-to-end machine learning and deep learning pilelines. 2nd edn.* <packt> Publishing.

Journals/Additional Texts

Weisfeld, M. (2018) *The Object-Oriented Thought Process, Fifth Edition.* New York: Addison-Wesley.

Websites

W3 Schools. Available at: https://www.w3schools.com/cpp/ (Accessed: November 2024).

See syllabus for complete reading list.

Change Log for this CSD:

Nature of Change	Date Approved & Approval Body (School or AB)	Change Actioned by Registry Services
First Edition	Nov 2024	