Liverpool John Moores University

Title:	ADVANCED COMPUTER GAMES DEVELOPMENT
Status:	Definitive
Code:	**6041COMP** (117453)
Version Start Date:	01-08-2019

Owning School/Faculty:	Computer Science
Teaching School/Faculty:	Computer Science

Team	Leader
Chris Carter	Y
David Tully	

Academic Level:	FHEQ6	**Credit Value:**	24	**Total Delivered Hours:**	72
Total Learning Hours:	240	**Private Study:**	168		

Delivery Options

Course typically offered: Standard Year Long

Component	Contact Hours
Lecture	24
Tutorial	24
Workshop	24

Grading Basis: 40 %

Assessment Details

Category	Short Description	Description	Weighting (%)	Exam Duration
Technology	AS1	Development of a 3D game prototype with the emphasis on the software development process and modern game engine structures.	50	
Technology	AS2	Full development of 3D game. Students are expected to provide full details of the software development process. Group based and must include peer assessment report.	50	

Aims

To develop skills and expertise in developing computer games.
To understand and use the structures and technologies of modern game engines.
To present advanced game programming techniques and technologies applicable to game development.
To identify, formulate and apply solutions to a diverse range of advanced computer game problems.
To teach the process and techniques of creating advanced computer/video games under simulated conditions of a real-world video game software development company.

Learning Outcomes

After completing the module the student should be able to:

1	Specify, design and implement game techniques using modern game engine methodologies and structures.
2	Apply an appropriate API (DirectX, OpenGL, OGRE, etc.) to implement a particular aspect of computer games development.
3	Apply suitable programming techniques and game technologies to solve problems specific to gaming applications.
4	Solve 3D game programming problems.
5	Apply Software Development Methodologies to game development.
6	Produce documentation for complete process of game development.

Learning Outcomes of Assessments

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

3D game prototype	1	2	3
3D game development	4	5	6

Outline Syllabus

Development methodologies, working collaboratively.
Win32 programming for games, C++ API usage, interaction event-driven software.
Algorithms and data structures for game development.
3D game engine architecture design and middleware.
Data-driven game development, scripting, localisation.
Game engine asset management.
Rendering middleware graphics programming.
Game UI; orthographic rendering for 2D in 3D environments.
GPU processing, parallelisation for games, pipelining.
Input and 3D audio.
CPU and GPU performance prediction, troubleshooting, optimisation.
Spatial data structures, bounding volumes (AABB, OBB, k-DOPs), potentially visible sets (BVH, BSP, quadtrees, octrees), occlusion, portals, anti-portals.

3D collision detection and interaction.
Motion in games (scripted behaviour, animation, dynamics).
Procedurally generated game data (Perlin noise, fractal, tessellation, Catmull-Clark, physics-based techniques).
Cellular Automata and water.
AI, pathfinding/route solving and flocking, rule-based systems, finite state machines.

Learning Activities

Lectures will be accompanied by hands-on practical laboratory sessions. Directed reading (Internet based) will be used to supplement course material.
Practical use of software engineering techniques, game APIs and programming language, in game development for individual and team-based assignments.

Notes

This module covers advanced techniques in modern games software implementation. The main objective of this course is to expose the students to the process and techniques of creating advanced computer/video games under simulated conditions of a real-world video game software development company.