Module specification

When printed this becomes an uncontrolled document. Please access the Module Directory for the most up to date version by clicking on the following link: Module directory

Module Code	COM559
Module Title	Games Programming
Level	5
Credit value	20
Faculty	FAST
HECoS Code	101020
Cost Code	GACP

Programmes in which module to be offered

Programme title	Is the module core or option for this programme
BSc (Hons) Computer Game Development	Core
BSc (Hons) Computer Game Development (with Industrial Placement)	Core

Pre-requisites

None

Breakdown of module hours

Learning and teaching hours	30 hrs
Placement tutor support	0 hrs
Supervised learning e.g. practical classes, workshops	0 hrs
Project supervision (level 6 projects and dissertation modules only)	0 hrs
Total active learning and teaching hours	**30**hrs
Placement / work based learning	0 hrs
Guided independent study	170 hrs
Module duration (total hours)	200 hrs

For office use only	
Initial approval date	10/06/2021
With effect from date	September 2023

For office use only	
Date and details of revision	10/05/2023 AB approval of revalidated Games suite
Version number	2

Module aims

This module aims to revisit the essential pillars of Object-Oriented Programming (OOP) through the scope of game development. Early work through the module will ensure that students have the understanding the concepts of objects and data structures whilst continuing to expand on more complex techniques and strategies.

This module will further students understanding of the OOP paradigm through experimentation and development within a game engine environment. Within this there will be a specific focus on visual and non-visual tools and strategies, how they relate to one another and when it is best to use them within game development. Students will finish this module with portfolio work that demonstrates their learning and programming acumen to date.

Module Learning Outcomes - at the end of this module, students will be able to:

1	Apply concepts of Object-Oriented Programming within game development.
2	Utilise Data Structures and Object Communication to solve project-based problems.
3	Differentiate Visual and Non-visual programming strategies within game engines and relate them to appropriate uses.
4	Produce technical portfolio pieces that demonstrate the Object-Oriented Programming paradigm.

Assessment

Indicative Assessment Tasks:

This module will indicatively be made of several coursework pieces that build on/focus on individual areas of expertise within game development. This may include smaller, sequential activities for students to build up skills and self-efficacy at the start of the module and finalise with a larger piece that demonstrates their cumulative skill learned throughout. These may not all represent completed games, but they will be tightly associated with game development and at least some will represent a portfolio-worthy technical project.

Assessment number	Learning Outcomes to be met	Type of assessment	Weighting (%)
1	1, 2, 3, 4	Coursework	100

Derogations

N/A

Learning and Teaching Strategies

In line with the Active Learning Framework, this module will be blended digitally with both a VLE and online community. Content will be available for students to access synchronously and asynchronously and may indicatively include first and third-party tutorials and videos, supporting files, sections of code/diagrams or any additional content that supports their learning.

As this module progresses, a structured strategy will be used to support the students engaging with the key threshold concepts relating to the learning outcomes. The module will include a balanced mixture of engaging tutor-led lectures, demonstrations, and facilitation. As the module continues experiential and peer learning strategies will be encouraged as the students' progress with their coursework.

Indicative Syllabus Outline

Depending on the relevance to current industry trends a programming language will be chosen that aligns with a contemporary Game Engine. Indicatively C++ and Unreal Engine 4/5 will be the focus of teaching and assessment, and students will be required to demonstrate the module outcomes to the specified language and engine chosen. The following essential topics will be delivered through the syllabus as core programming concepts. These may be contextualised through a selection of the second list of indicative subject areas.

Indicative essential topics:

- Object-Oriented Programming paradigm.
- Visual Programming (e.g. UE4 Blueprint)
- Non-visual Programming (e.g. C++)
- Classes and Encapsulation
- Inheritance and Derived Classes
- Abstraction and Polymorphism
- Pointers, References and Memory Allocation

Indicative subject areas:

- Gameplay Mechanics
- User Interface Development
- Artificial Intelligence and Finite State Machines
- Procedural Techniques
- Creation of Components
- Technical Portfolio Development

Indicative Bibliography:

Essential Reads

Dawson, M. (2014), *Beginning C++ Through Game Programming*, Fourth Edition, Boston: Cengage Learning.

Other indicative reading

Ferrone, H. (2022), *Learning C# by Developing Games with Unity 2022*, Seventh Edition, Birmingham: Packt Publishing.

Kelly, S. (2019), *Python, PyGame, and Raspberry Pi Game Development*, Second Edition, Niagra Falls: Apress.

Ulibarri, S. S. (2020), *Unreal Engine C++ the Ultimate Developer's Handbook*, London: Druid Mechanics.

Employability skills – the Glyndŵr Graduate

Each module and programme is designed to cover core Glyndŵr Graduate Attributes with the aim that each Graduate will leave Glyndŵr having achieved key employability skills as part of their study. The following attributes will be covered within this module either through the content or as part of the assessment. The programme is designed to cover all attributes and each module may cover different areas.

Core Attributes
Engaged
Enterprising
Creative

Key Attitudes
Commitment
Curiosity
Resilience
Confidence
Adaptability

Practical Skillsets
Digital Fluency
Organisation
Critical Thinking
Communication