



ORIGINAL COURSE IMPLEMENTATION DATE: September 2022
 REVISED COURSE IMPLEMENTATION DATE:
 COURSE TO BE REVIEWED (six years after UEC approval): February 2028
 Course outline form version: 06/18/2021

OFFICIAL UNDERGRADUATE COURSE OUTLINE FORM

Note: The University reserves the right to amend course outlines as needed without notice.

Course Code and Number: MEDA 280	Number of Credits: 3 Course credit policy (105)										
Course Full Title: Game Engines I Course Short Title:											
Faculty: Faculty of Humanities	Department (or program if no department): Media Arts										
Calendar Description: Students explore the use of game engines for game creation and for designing VR experiences, filmmaking, animation, training, and simulation. The course introduces basic software engineering concepts through graphic manipulation of programming elements. Students develop, package, and deliver software without writing code.											
Prerequisites (or NONE):	MEDA 110.										
Corequisites (if applicable, or NONE):											
Pre/corequisites (if applicable, or NONE):											
Antirequisite Courses <i>(Cannot be taken for additional credit.)</i> Former course code/number: Cross-listed with: Equivalent course(s): <i>(If offered in the previous five years, antirequisite course(s) will be included in the calendar description as a note that students with credit for the antirequisite course(s) cannot take this course for further credit.)</i>	Course Details Special Topics course: No <i>(If yes, the course will be offered under different letter designations representing different topics.)</i> Directed Study course: No Grading System: Letter Grades Delivery Mode: May be offered in multiple delivery modes Expected frequency: Annually Maximum enrolment (for information only): 36										
Typical Structure of Instructional Hours <table border="1" style="width: 100%; border-collapse: collapse; margin-top: 10px;"> <tr> <td style="width: 80%;">Lecture/seminar</td> <td style="width: 20%; text-align: center;">15</td> </tr> <tr> <td>Tutorials/workshops</td> <td style="text-align: center;">15</td> </tr> <tr> <td>Supervised laboratory hours (computer lab)</td> <td style="text-align: center;">15</td> </tr> <tr> <td> </td> <td> </td> </tr> <tr> <td style="text-align: right;">Total hours</td> <td style="text-align: center;">45</td> </tr> </table>	Lecture/seminar	15	Tutorials/workshops	15	Supervised laboratory hours (computer lab)	15			Total hours	45	Prior Learning Assessment and Recognition (PLAR) PLAR is available for this course.
Lecture/seminar	15										
Tutorials/workshops	15										
Supervised laboratory hours (computer lab)	15										
Total hours	45										
Labs to be scheduled independent of lecture hours: <input checked="" type="checkbox"/> No <input type="checkbox"/> Yes	Transfer Credit <i>(See bctransferguide.ca.)</i> Transfer credit already exists: No Submit outline for (re)articulation: No <i>(If yes, fill in transfer credit form.)</i>										
Department approval	Date of meeting: November 26, 2021										
Faculty Council approval	Date of meeting: December 17, 2021										
Undergraduate Education Committee (UEC) approval	Date of meeting: February 25, 2022										

Learning Outcomes

Upon successful completion of this course, students will be able to:

1. Use a game engine for real-time interactive design.
2. Apply user experience design concepts to human-computer interaction (HCI) systems.
3. Engineer games, software, or interactive experiences, using a visual scripting system.
4. Produce 2D and 3D computer-generated imagery and animation for use in interactive experiences.
5. Apply principles of visual organization, visual language, and theory to visual communication problems.
6. Recognize how a game engine is used within a larger production context.
7. Use appropriate terminology to communicate the technical concepts of real-time interactive design.

Recommended Evaluation Methods and Weighting (*Evaluation should align to learning outcomes.*)

Assignments:	100%	Project:	%	Quizzes/tests:	%
	%		%		%

Details: Assignment 1 (25%): Creating a 1st Person Level. Assignment 2 (25%): Environmental Design. Assignment 3 (40%): Using Blueprints to Engineer Gameplay. Assignment 4 (10%): Animation in Unreal Engine.

NOTE: The following sections may vary by instructor. Please see course syllabus available from the instructor.

Texts and Resource Materials (*Include online resources and Indigenous knowledge sources. [Open Educational Resources](#) (OER) should be included whenever possible. If more space is required, use the [Supplemental Texts and Resource Materials form](#).)*)

Type	Author or description	Title and publication/access details	Year
1. Textbook	Cookson A, DowlingSoka R, Crumpler C, Johnson T	Unreal Engine 4 Game Development in 24 Hours, Sams Teach Yourself	2016
2. Textbook	Shannon Tom	Unreal Engine 4 for Design Visualization: Developing Stunning Interactive Visualizations, Animations, and Renderings.	2017
3. Textbook	McCaffrey M	Unreal Engine VR Cookbook: Developing Virtual Reality with UE4	2017
4.			
5.			

Required Additional Supplies and Materials (*Software, hardware, tools, specialized clothing, etc.*)

Adobe Photoshop CC, Unreal Engine.

Course Content and Topics

Unit 1: Introduction to game engines.

- What is a game engine?
- Common engines.
- How game engines are used for real-time content creation outside of the game industry.
- Introduction to Unreal Engine.
- Using a 3D interface.
- 3D prototyping.
- Introduction to game engine look development.
- Materials and textures.
- Importing objects from 3D DCC applications.
- Lighting.
- Packaging (exporting) and delivery of games.

Unit 2: Components and assets.

- Component types.
- Meshes, capsules, cameras.
- Player Inputs
- Importing Assets
- Asset store.
- 3D assets (alembic, obj, fbx).
- 2D images.
- Lighting and rendering.
- Rendering methods.

Unit 3: Introduction to blueprints.

- Classes and instances.
- Compiling.
- Creating interactive objects and volumes.
- Building logic using blueprints.
- Types of variables.
- Actors.
- References.
- Getters and setters.
- Using arithmetical, relational, and logical operators.
- Input and output parameters.
- Macros, events, and functions.
- For loops.

Unit 4: Animation

- Importing animation.
- Animating in Unreal Engine
- Dynamics.