

| | | DOMAIN | | |
|------|-----------------|--|-----------|--------|
| CUTM | ARCU2060 | Gaming and Immersive Learning (AR & VR) | 20 | 5+5+14 |
| | CUAR2060 | Introduction to Gaming & Simulation | 2 | 1+1+0 |
| | CUAR2061 | Game Assets & Game Objects | 3 | 1+1+1 |
| | CUAR2062 | Building Game Environment | 3 | 1+1+1 |
| | CUAR2063 | Game Animation, Scripting & UI | 3 | 1+1+1 |
| | CUAR2064 | Binary Deployment and Cross-Platform Controls | 3 | 1+1+1 |
| | CUAR2065 | Project | 10 | 0+0+10 |

Objective:

- Students will know about the History of Computer Graphics
- Know about Gaming Industry and get a job in AR/VR field
- Understanding of Individual Roles in a Gaming Industry
- End to End Game Development skill

Course outcome:

- Storyboarding
- Game Assets Development, efficient Import/ Export of assets for Games.
- Animation in Games
- C# Programming
- Audio Pipeline
- User Interface
- Cross-Platform Support of a Game
- Technical and Specification Document of a Game

Course content

1. Introduction to Gaming & Simulation

Module I: Welcome to Game Engine

(2+2.5)

Importance of Storyboarding a Game Idea, The Economics of Game Development, Assessing Game Markets and Platforms, Marketing Methods for Games, Monetizing Games and Upgrades.

Module II: Introduction to Game Production

(1)

Video Game Platforms and Genres, Describing the Game Production Pipeline, Game Development Jobs and Roles, The Game Design Document, The Technical Design Document,

Getting Started in Unity, Creating a New Unity Project, Using the Unity Asset Store, Source Control for Working in Team.

Module III: The Game Engine User Interface (2+2)

Introduction to the Unity Editor Interface, Analyzing the Unity Editor User Interface, Utilizing the Unity Editor User Interface, Navigating the Scene View Window, Utilizing the Game View Window, Navigating the Hierarchy Window, Using the Inspector Window, Managing Assets in the Project Window, Searching and Filtering in the Project Window, Organizing the Scene with Layers.

Module IV: Using Game objects and assets (1+2)

Creating and Modifying Game Objects, Defining Unity Editor Units, Describing Assets in the Production Pipeline, Review: Defining an Asset, Organizing Assets in the Unity Editor, Defining a Game Object.

Module V: Defining a Game Object (2+2)

Creating Unity-native Game Objects, Manipulating Game Objects in the Unity Editor, Describing What is a Unity-native Game Object, The Role of Components in the Unity Editor, Defining Prefabs and Scene Structure, Defining the Role of the Prefab in Unity, Creating and Saving a Scene.

Module VI: The Hierarchy of Scenes within a Game (1+2)

Importing Assets into a Project, Importing and Configuring a 3D Model, Importing Textures for Use in Materials, Importing FBX Files with Animation, Working with Sprites, Introduction to Sprites in Game Development.

Module VII: Managing Projects and Assets (1+2)

Project Management in Unity, Introduction to Game Project Management, Managing Assets, Using the Unity Asset Store (Reprise), Importing Offline Content, Creating Project Structure Based on Assets, Sorting the Zombie Toys Prop Model Assets, Setting Resolution and Type of Texture Files.

2. Game Assets and Objects

Module I: Preparing Assets for implementation (1+2.5)

Best Practices in 3D Content Creation, Modelling for Games, Animating for Games, UV Mapping and Texturing Techniques, and Exporting to Unity, Importing into Unity, Materials in Unity, The Interaction of Lighting and Materials.

Module II: Discovering the Standard Shader in Unity (1+2)

Exploring other Material Types, Analyzing the Benefits of Custom Shaders, Creating the Materials for Zombie Toys Props, Duplicating and Modifying Materials, Case Studies in Material Creation, Managing and Using Textures in the Unity Editor, Texturing for Game Development, Optimization and Reuse of Textures.

Module III: Assembling the Game Level (2+2)

Branching and Hierarchies, Creating Hierarchies in Unity, Using Empty Game Objects as Pivots, Introduction to Physics in Unity, Understanding the Physics System in Unity, Introduction to the Rigid body Component.

Module IV: Introduction to Colliders (1+2)

Creating the Colliders for Zombie Toys Props, Introduction to Game Level Design, Introduction to Game Level Design, The Level Design in Zombie Toys, Placing Objects in a Scene, Importing the Prop Prefabs into the Scene, Cloning the Stars, Creating the Level Boundaries.

Module V: Lighting in Games (1+2)

Introduction to Game Lighting, Introduction to Game Lighting, Differences in Lighting for Games and for Film, Placing and Adjusting Lights in a Scene, Analyzing the Different Lights and Properties, Light Types and Behaviors, Using Layers to Exclude Objects from Lighting, Casting and Modifying Shadows, Mesh Renderer Attributes for Shadows.

Module VI: Differentiating Shadow Types (1+2)

Creating Cookies to Shape Lights, Faking Shadows for Better Performance, Benefits of Faking Shadows in Games, Utilizing Painted Shadows, Using Projectors to Project Shadow Cookies, Lighting the Zombie Toys Game, Lighting the Zombie Toys Scene, Lighting Variations for Changing the Mood.

Module VII: Baking Lighting in Game Production (1+2)

Light Baking in Video Games, Introduction to Light Baking in Video Games, Setting Objects to Participate in Light Baking, Marking Objects as Static for Light Baking, Creating UV Coordinates for Light Baking, Baking Lighting, Continuous and Manual Light Baking, Placing Light Probes for Moving Objects, Creating Reflection Probes, Baking the Lighting in Zombie Toys, Creating the Light Probes in Zombie Toys.

3. Building Game Environment

Module I: Building the Player and Allies (2+2)

Creating a Player Controller, Examining Why to Use a Custom Controller, Creating the Player Controller Game Object, Adding a Game Manager, Explaining the Purpose of the Game Manager, Making the Controller Functional, Adding Scripts for Behavior, Configuring the

Camera, Creating the Sheep Ally, Building the Sheep Ally From a Model, Creating the Dog Ally, Building the Dog Ally From a Model.

Module II: Building the Enemies (2+2)

Creating an Enemy, Designing the Enemy Behaviors, Creating the First Enemy Character, Creating the Enemy Animator Controller, Creating Additional Enemies, Creating the Zombear Enemy, Creating the Zombie Duck Enemy, Creating the Other Enemies, Integrating Enemies into the Game, Placing the Spawn Points, Spawning the Enemies.

Module III: Introduction to Unity's Particle System (1+1)

Analyzing Existing Particle Effects, Setting Up the Interface for Effects, Case Study: Developing the Lightning Attack, Overview of the Lightning Attack, Building the Lightning Attack Hit, Building the Lightning Attack Emitter, Building the Lightning Bolt, Integrating the Lightning Attack into the Game.

Module IV: Creating Particle Systems (1+2)

Intro to the Particle Systems in the Unity Editor, Examples of Unity Particles in Video Games, the Role of the Effects Artist in Video Games, Comparing Game Effects with Other Media, And Production Best Practices for Particle Systems.

Module V: Case Study (1+2)

Developing the Frost Attack, Introduction to the Frost Attack, Building the Frost Debuff, Building the Frost Attack Emitter, Building the Frost Cone Effect, Integrating the Frost Attack into the Game, Case Study: Developing the Stink Bomb Attack, Introduction to the Stink Bomb Attack, Creating the Stink Bomb Hit Effect.

Module VI: Case Study (1+2)

Developing the Slime Attack, Introduction to the Slime Attack, Creating the Slime Hit Effect, Creating the Slime Debuff, Creating the Slime Attack Reticule, Building the Slime Attack Emitter, Building the Slime Projectile, Integrating the Slime Attack into the Game, Finalizing Player Attacks, Adding the Ally Manager.

Module VII: Adding Audio to Game Levels (1+2.5)

Introduction to Audio in Game Development, Introduction to Audio in Game Development, Importing Audio into Unity, Introduction to Importing Audio in Unity, Supported Audio Formats in Unity, Playing Audio in the Unity Editor, Testing Audio Sources in the Scene, Mixing Audio in Unity, Using Audio Mixers and Audio Mixer Groups, Setting up the Zombie Toys Audio Mixers, Creating Audio Effects, Introduction to Audio Effects.

4. Game Animation, Scripting & UI

Module I: Animating Game Objects in the Unity Editor (1+2)

Introduction to Animation in Game Development, Introduction to Animation in Game Development, Animating in the Unity Editor, Creating Animation in the Unity Editor, Refining Animation in the Unity Editor.

Module II: Bringing Animation into the Game (1+2)

Importing Animated Characters, Introduction to Rigging and Imported Animation, Recognizing Asset Data when Importing, Differentiating Available Rig Animation Types.

Module III: Animation Creation and Controlling (1+2)

Creating and Naming AnimationClips, Creating an Animator Controller, Introduction to the Animator Controller, Creating and Modifying Animation States, Creating Parameters to Control Transitions, Creating an Animator Override Controller.

Module IV: Scripting in Game Development (2+2.5)

Intro to Scripting in Game Development, Intro to Scripting in Game Development, Creating Scripts in Unity, Creating and Saving a Script in Unity, Analysing the Default, Script Methods, Scripting Primer and Best Practices, Attaching a Script to a Game Object. Declaring Variables, List of Variable Types, Creating Conditions in Scripting, Introduction to Conditions, The “if” Condition, The “if else” Condition, Complex Conditions, Looping.

Module V: Designing User Interfaces for Games (1+2)

Introduction to Designing the User Interface, Assessing User Interface Design Needs, Examining the UI Tools in the Unity Editor, Creating a User Interface, Investigating the Canvas Functionality, Utilizing the Power of the Rect Transform, Creating a UI Button, Creating a UI Image, Creating UI Text, Creating Interaction in the UI with Events.

Module VI: Introduction to Looping (1+2)

The “while” Loop, The “for” Loop, Creating Custom Methods, The Benefits of Using Custom Methods, Utilizing Arguments, Utilizing Method Return Types, Coroutines, Introduction to Coroutines, Accessing Components via Script, Utilizing the GetComponent() Function, Common Code Cases, Common Pieces of Zombie Toys Code..

Module VII: Implementing Navigation and Path Finding (1+2)

Introduction to Navigation and Path Finding, Introduction to Navigation in Unity, Describing a NavMesh, Defining a NavMesh Agent, Describing a NavMesh Obstacle.

5. Binary Deployment and Cross-Platform Controls

Module I: Building the Camera and Player Selection System (1+3)

Intro to the Camera and Player Selection Behaviors, Analyzing the Player Selection System.

Module II: Creating another Player Option (2+2.5)

Making the Player Selectable, Adding Another Player, Finalizing the Camera.

Module III: Adding Camera Animations (2+2)

Configuring the Camera Animator Controller, Applying Behavior to the Camera, Adding Character Selection Spotlights.

Module IV: Building and Deploying the Game (1+1)

Building the Game, Introduction to the Build Process, Adjusting the Player Settings, Building the Game.

Module V: Protecting Your Creation (1)

Legal Considerations for Your Game, Unity Services, Unlocking the Unity Platform Potential, Surveying Unity Services

Module VI: Understanding of Cross-Platform Inputs (1+2)

Different Input types like, Mobile, WebGL, OpenVR & other unity supported platforms.

Module VII: Preparing for Mobile Deployment (2+2)

Modifying Zombie Toys for Mobile, Introduction to Mobile Development in Unity. Changing the Build Platform to Mobile, Adding the Mobile Interface UI, Implementing Mobile Input Behaviors.

Text Books:

1. *Jared Halpern, Developing 2D Games with Unity: Independent Game Programming with C#, Apress, Final Edition*
2. *Jon Manning, Paris Buttfield-Addison, and Tim Nugent, Unity Game Development Cookbook: Essentials for Every Game, O'Reilly Media, Inc.*

Reference Books:

3. *Jason Gregory, Game Engine Architecture, CRC Press, Third Edition*
4. *Linowes Jonathan, Unity Virtual Reality Projects, Packt, Second Edition*

Course outline Prepared by: Abhi Mitra

Date: July 6, 2020

Source of reference: UCA Courseware.

*Note: 1 credit theory=10 hrs lecture, 1 credit practice/project=12.5 hrs lab/workshop/field work
in a semester*