GAME PROGRAMMING

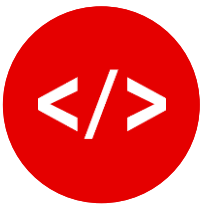
CS 3540

CAGLAR YILDIRIM

1

About the Course

UNITY 3D



C# PROGRAMMING

2

Transformations

Physics

Lighting

Collision Detection

Shooting Projectiles

Scores and Levels

Health and Damage

NPC Behavior

Player Control

Game AI

User Interface

3

Unity 3D as a Game Engine

Game engine:

- Software development environment or framework
- Supports and combines several core components of games
- Assets, lighting, audio, special effects, physics, animation, interactivity, and scripting

Unity 3D:

- Free for educational projects and small-scale projects
- Very popular among studios, independent developers, artists, etc.
- Great documentation, tutorials, and community support

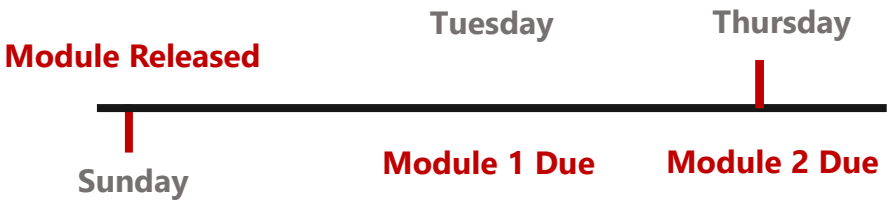
4

Course Requirements

5

Online Learning

Asynchronous content delivery



6

What will you do in this course?

Syllabus

Schedule

Canvas Page

Final Project

7

Final Project

GOAL

To give you hands-on experience applying the skills from this class to a life-like game development project in which you conceptualize and develop a 3D game from scratch

DELIVERABLES

- FP1: Game Design Document
- FP2: Level Design Prototype
- FP3: Alpha Game Release
- FP4: Beta Game Release
- FP5: Gold Master Release
- FP6: Game Demo
- FP7: Peer Evaluation

8

Final Project: Requirements

Be a 3D video game.

Feature high interactivity and appropriate visual and audio effects.

Use at least three different animations for visual effects.

Have a clear story and should support progression.

Have at least three levels with various game mechanics in each level.

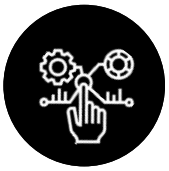
Have an intuitive user interface that displays progress/time/health/etc.

Use a game manager that keeps track of time/level/health/ammo/etc.

Implement game AI and use intelligent behaviors for enemies and NPCs.

Have a main menu

9



Foundations of Game Development

11

Games…

What is in a video game?

What are some game genres?

What are game mechanics?

12

Game Elements

Game world

Storytelling

Game logic and rules (gameplay)

Characters (players and enemies)

Game mechanics

Challenge and progress

Exploration and strategy

Feedback and rewards

Aesthetics

Interactivity

13

Game Mechanics

Sicart (2008)

> Methods invoked by agents, designed for interaction with the game state.

Järvinen (2008):

> “Game mechanics are best described with verbs“
>
> e.g., jump, look, switch weapons, slash boxes

14

Game Design Documentation

High Concept Document

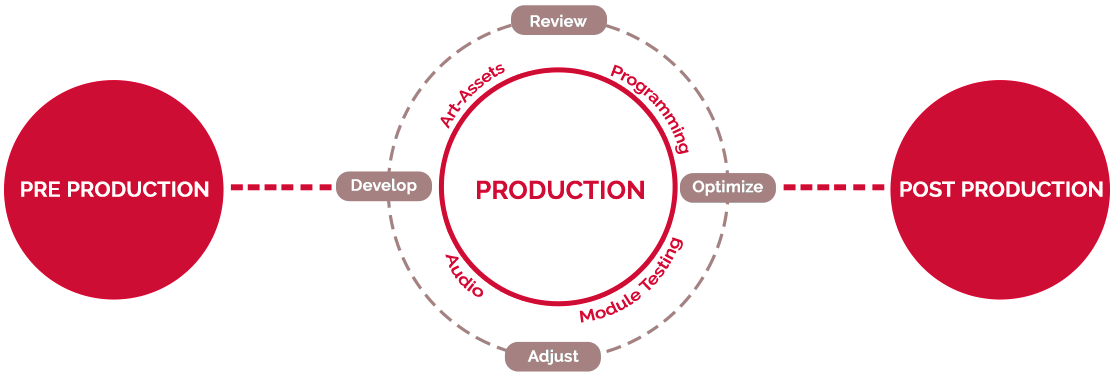
> To flush out your vision for the game
>
> Defines the scope of the game and basic game logic
>
> Intentionally brief

Game Design Document

> To flesh out your game with all the details
>
> e.g., world, story, characters, core mechanics, aesthetics, etc.
>
> Intentionally detailed

15

Game Development Life Cycle



16

Pre-Production

Game Concept

Ideation and brainstorming

Game Design Documentation

The core plan for features and game scope

Prototyping

Initial implementations such as sketches, grayboxing, etc.

17

Production

Initial Production

Main cycle of development and Content creation

Implementation of core gameplay features

Production Stages

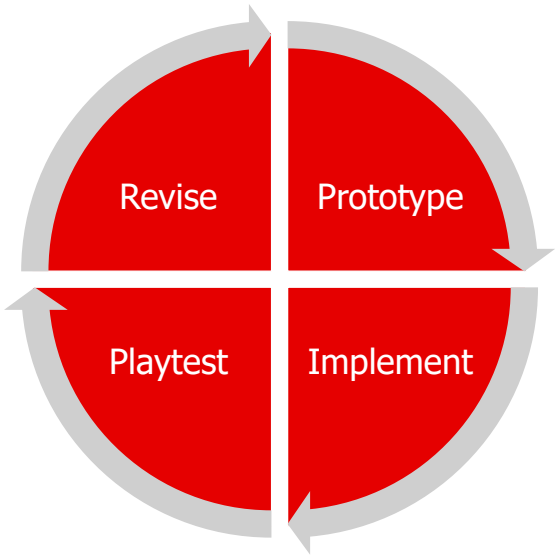
Alpha: Core gameplay features are implemented

Beta: Complete version of the game with fully functioning core game play features

Gold: Quality check and ready to be launched for the public release

18

Production: An Iterative Process



19

Post-Production

All actions after the game has been released

- Promotions
- Maintenance
- Patches
- Expansions
- Live content

20

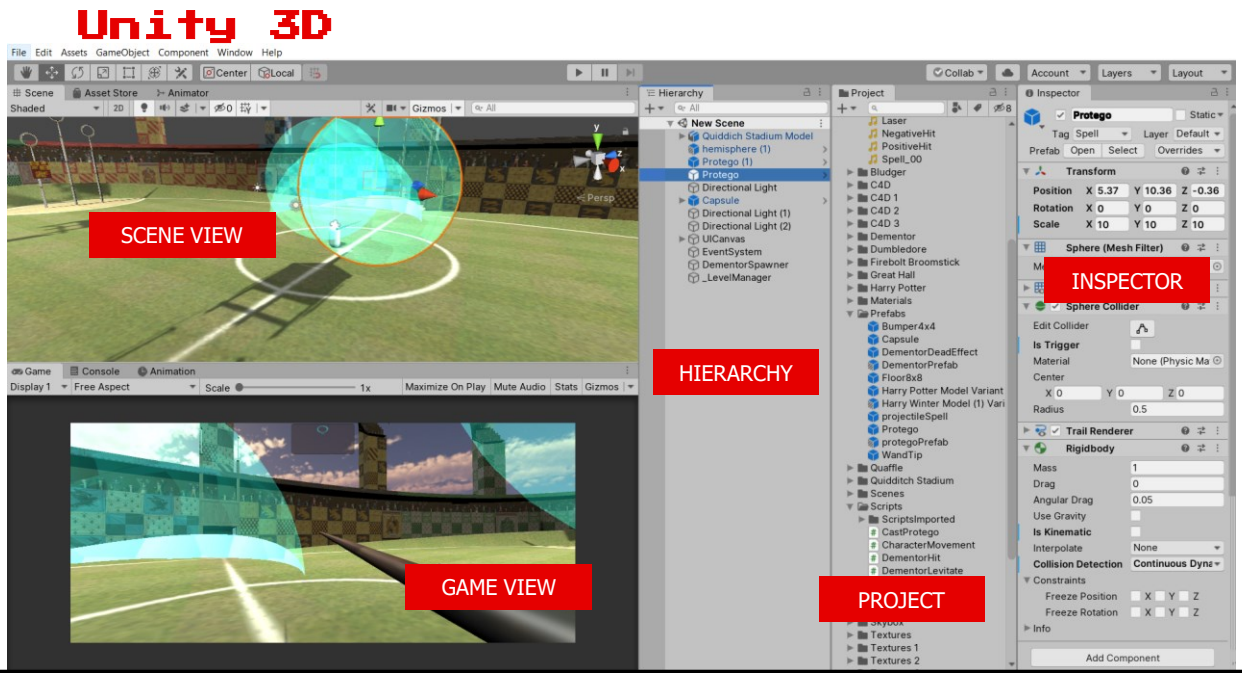
Installing Unity 3D

https://unity.com/download

21

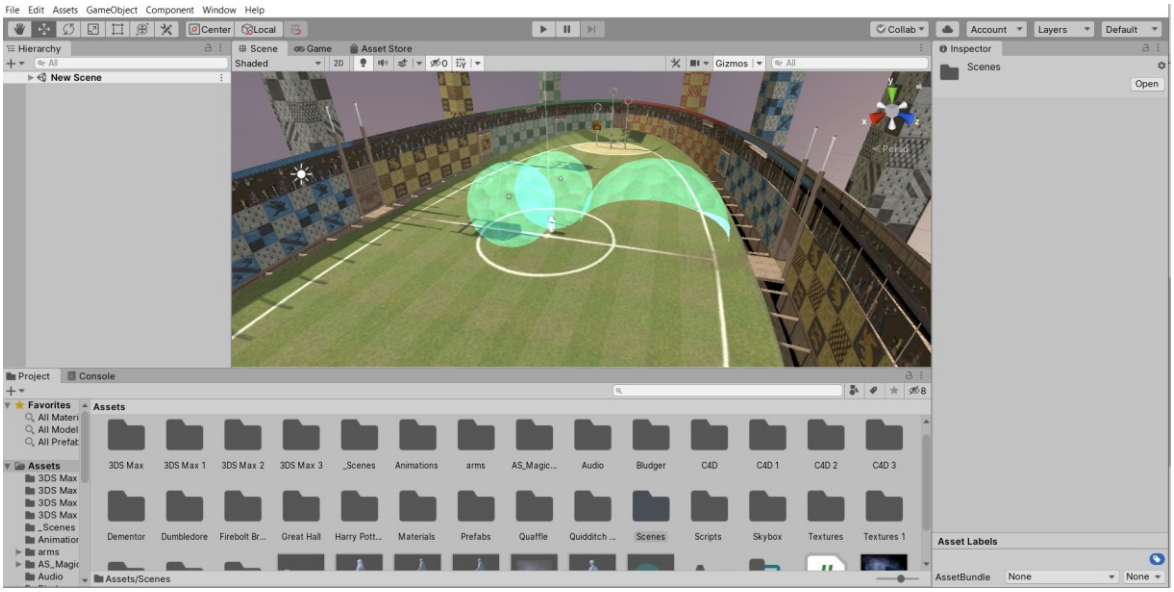
Unity 3D Environment

22



23

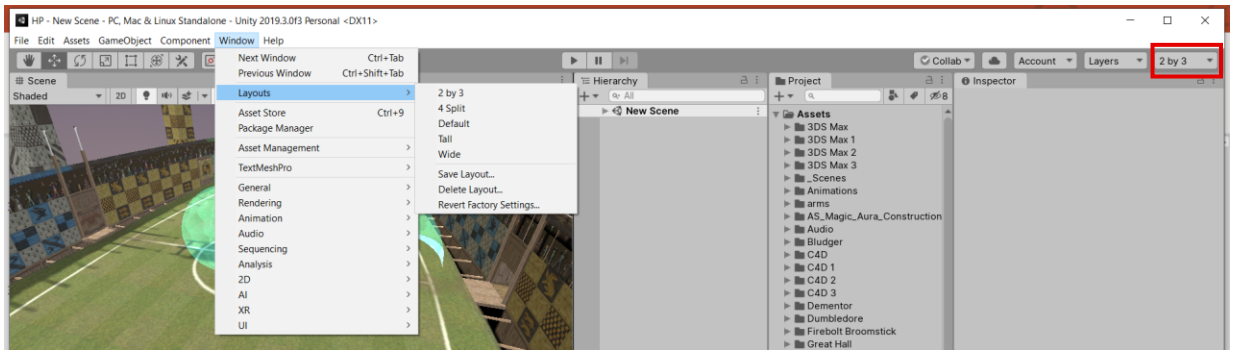
Unity 3D 2019.3



24

Unity 3D

Can change your Layout through Windows > Layouts



25

Hello Cube

26

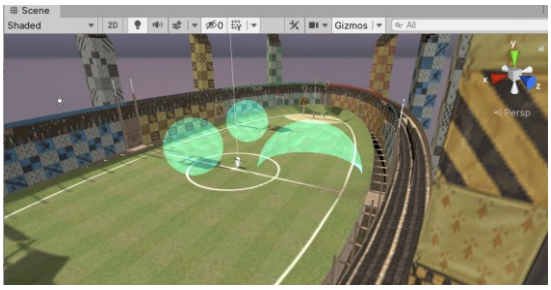
Unity 3D: Scenes

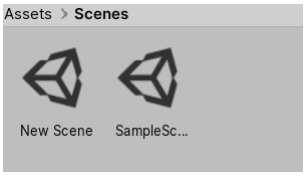
Scenes:

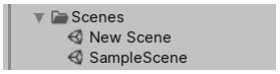
Contain the environments and menus of your game

A scene is essentially a level of your game

A project can (and usually does) contain multiple scenes





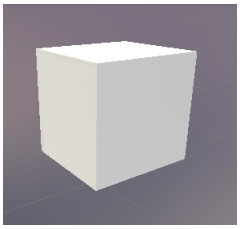


27

Unity 3D: GameObjects

Every object in your game is a **GameObject**

Players, collectibles, cameras, lights, and so on



GameObjects are more like containers or placeholders

They can't do anything on their own

Need to add some properties to GameObjects

Using **Components**

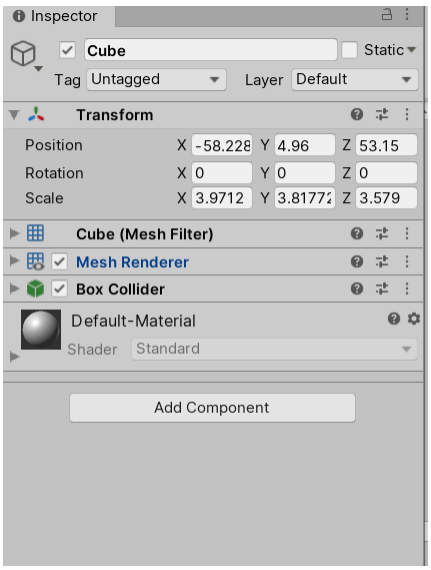
28

Unity 3D: Components

Define and control the behavior of GameObjects they are attached to

Implement the real functionality expected of the GameObjects

Provide a multitude of editable properties



29

Unity 3D: Materials

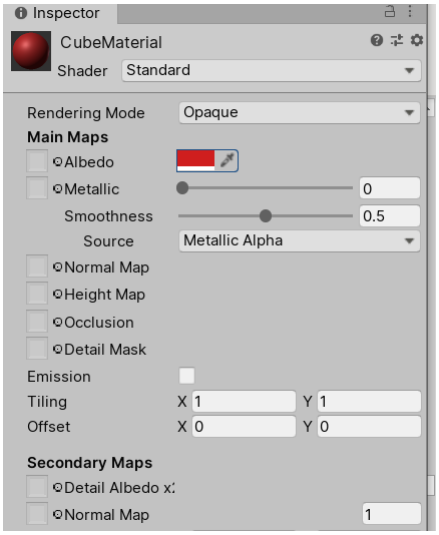
Define how a surface should be rendered

Can change the appearance of a GameObject's surface

Combines information from

- Textures,
- Tiling information,
- Color tints and more

The Albedo parameter controls the base color of the surface.



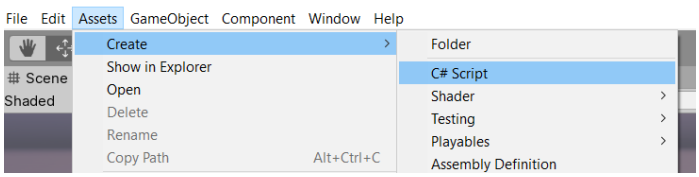
30

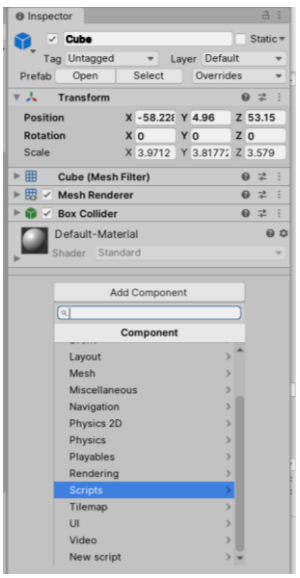
Unity 3D: Scripts

Scripts are Components too!

Allow you to write your own Components

Enable you to customize GameObjects

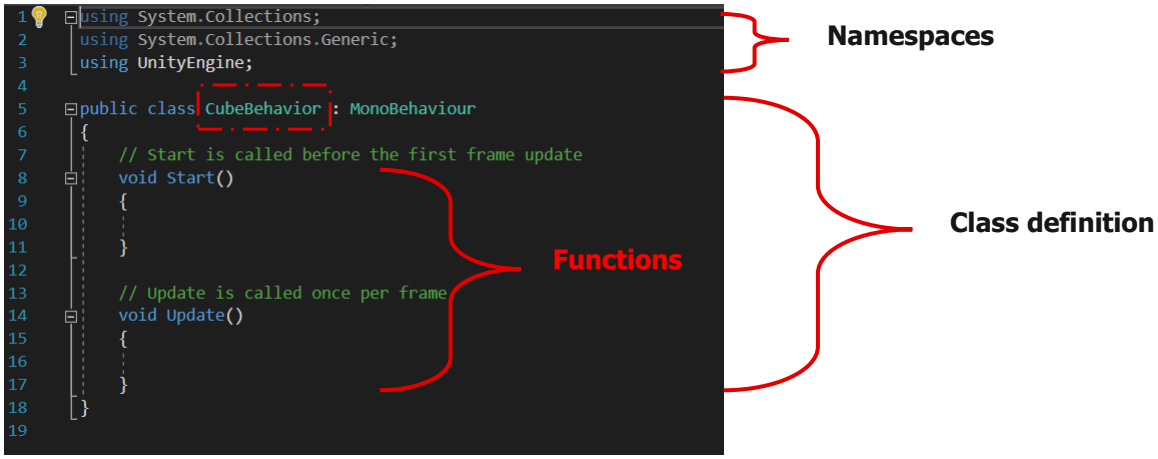




31

Anatomy of a Script File

All scripts, and thus classes, derive from the built-in class MonoBehaviour



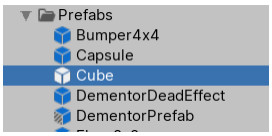
32

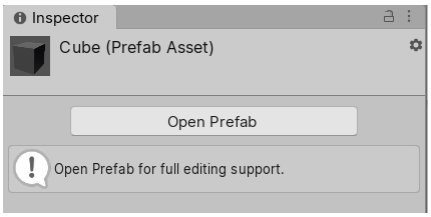
Unity 3D: Prefabs

Allows you to create, configure, and store a GameObject with all its components

Can be thought of as templates

Can use multiple instances of a prefab in your scene





33

Readings

https://docs.unity3d.com/Manual/LearningtheInterface.html
https://docs.unity3d.com/Manual/CreatingScenes.html
https://docs.unity3d.com/Manual/GameObjects.html
https://docs.unity3d.com/Manual/Components.html
https://docs.unity3d.com/Manual/Materials.html
https://docs.unity3d.com/Manual/ScriptingSection.html
https://docs.unity3d.com/Manual/Prefabs.html