

CyberDefense a Unity Asset

by Justin Garza

Introduction

Hello Thanks for purchasing this product.

If you need to contact me for any reason you can find my contact information [here](#).

Description

A 3D tap to defend game with a cyber/computer sci-fi theme.

Features

Saving HighScore
Pool Manager
Optimized for mobile
ShockWave Effect
Digital Glitch Effect
Screen Shake
Code Comments

Note: If you use this asset in your game, please put my name in the credits/special thanks. :

Table of Contents

Open Source Components	Page2
C# Script List	Page3-5
Before Compiling	Page6

Open Source Components

This Assets contains some open source components, below is a list of all the open source components and where to download them.

This list is available in the read me file of the purchased asset.

C# Script List

Below is a list of the main C# Scripts that are used in this game.

BlurControl.cs

This script blurs the screen while on Pause, and removes the blur during play.

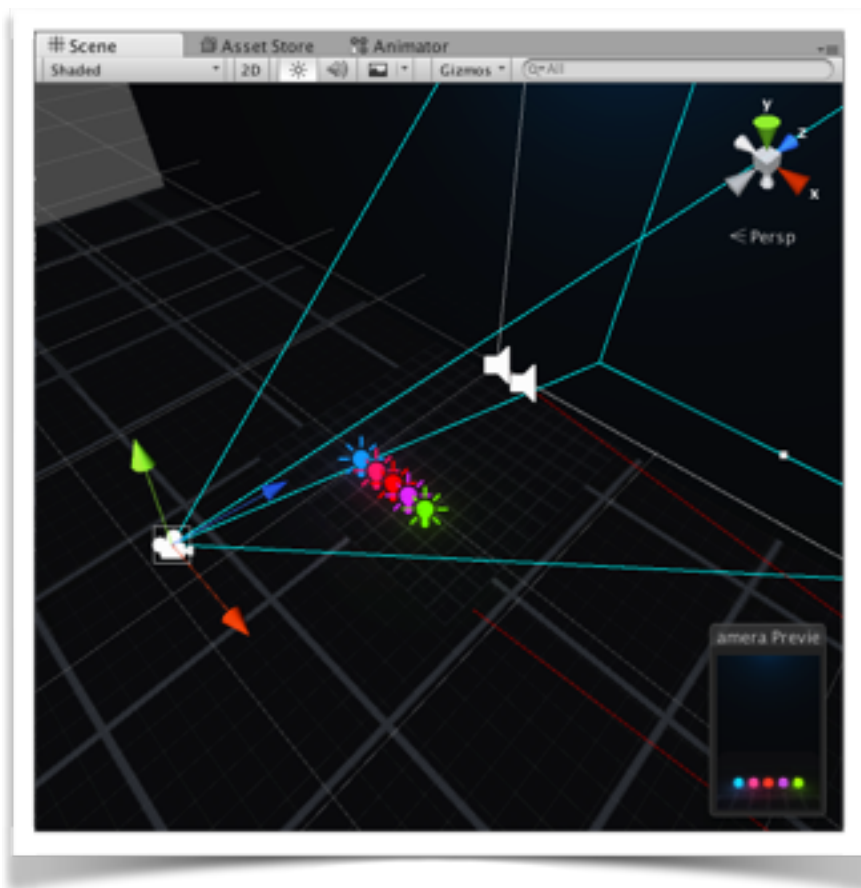
ButtonPlaySound.cs

This script plays a sound if a button is pressed

CameraFrustumGizmo.cs

This Script traces the Camera's guide lines.

This is just a tool for editing, and does not impact game play.



CameraShake.cs

This Script Shakes the camera.

Cannons.cs

This script controls the cannons, how they shoot the projectiles.

Cube.cs

This script is used to control the Cubes.

CustomEditorBase.cs

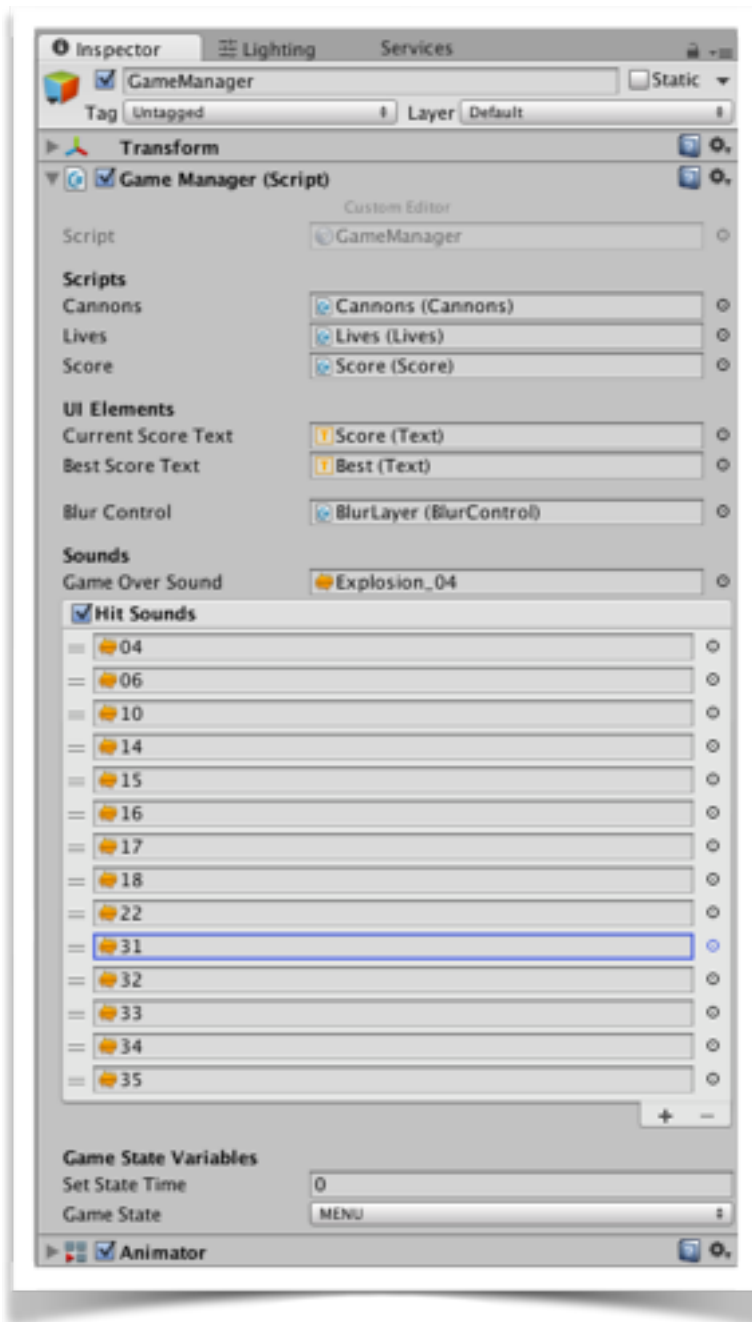
This script allows for the fancy Arrays and Lists in the inspector.

GameManager.cs

This script Manages the Game and switches the GameStates as needed.

GameStates include:

Menu	Menu or TitleScreen of the Game
Start	Starting a new Game
Play	While Playing the Game
Hit	If a cube get's hit
GameOver	During GameOver
Pause	During Pause



GlowColor.cs

This script leaps through colors in a gradient to animate the colors of Sprites, Text, Images, Shadows/Outlines.

Lives.cs

This script manages the Live UI Icons...thingies

ObjectPool.cs

This script contains a list of objects that can be reused.

OnTouch.cs

This script processes the user's touches to see if they touches a projectile.

PoolManager.cs

This script manages the ObjectPools for spawning and recycling.

Projectile.cs

This Manages the projectile's States.

RecycleAfter.cs

This script Recycles an Object after t seconds.

Score.cs

This script manages the Score.

Before Compiling

before compiling be sure to move (or copy) all the shaders to the Resource folder.

