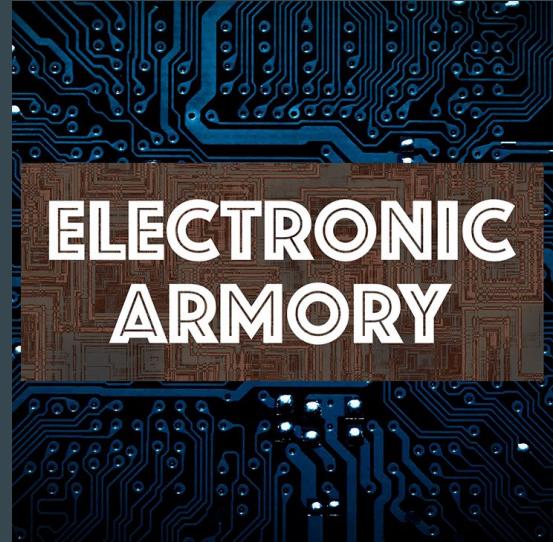


# Introduction to 2D/3D Game Development

...

[ElectronicArmory.com](http://ElectronicArmory.com)  
3D Game Development Course

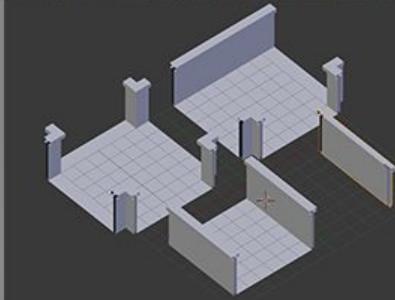
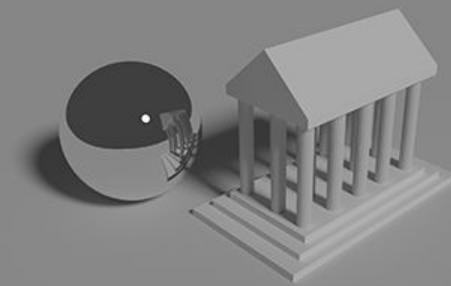


# Question of the Day

Favorite game from 90s, 00s, 10s, current

# ELECTRONIC ARMORY

## 3D GAME DEVELOPMENT CLASS



```
pragica once
#include "GameFramework/Character.h"
#include "DungeonCharacter.generated.h"

UCLASS(Blueprintable)
class ADungeonCharacter : public ACharacter
{
    GENERATED_BODY()

public:
    ADungeonCharacter();

    // Called every frame.
    virtual void Tick(float DeltaSeconds) override;

    /** Returns TopDownCameraComponent subobject */
    FORCEINLINE class UCameraComponent* GetTopDownCameraComponent() const
    /** Returns CameraBoom subobject */
    FORCEINLINE class USpringArmComponent* GetCameraBoom() const { return
    /** Returns CursorToWorld subobject */
    FORCEINLINE class UDecalComponent* GetCursorToWorld() { return CursorT
    /** Returns collection sphere component */
    FORCEINLINE class USphereComponent* GetSphereComponent() const { return
```



LEARN 3D CONCEPTS  
AND ANIMATION

CREATE 3D  
GAME ASSETS

PROGRAM IN  
C++ AND BLUEPRINTS

CONSTRUCT GAMES  
IN UNREAL ENGINE

# Your Instructor

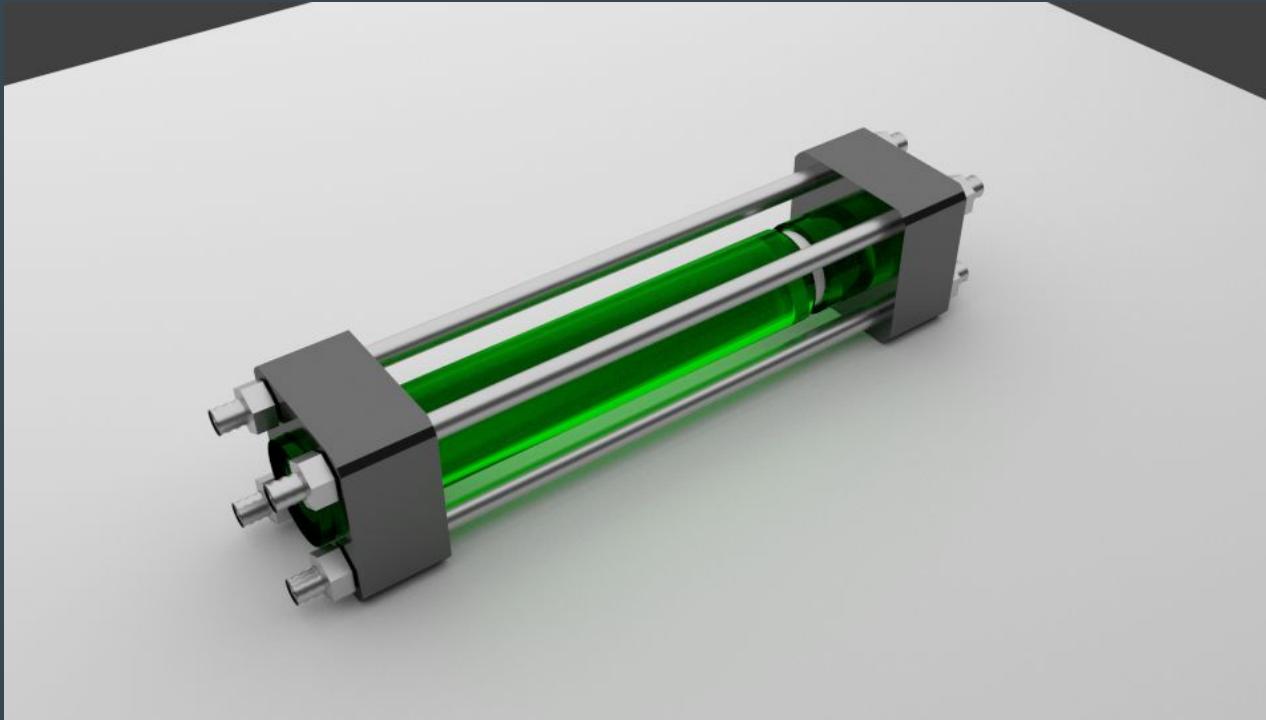
- Mike Ziray
- Made well over \$100,000 in games
- 6+ published games
- 3 currently in the works
- 3D artist
- Been making games since 1997 (Duke Nukem 3D mods)
- Idaho Game Developers Meetup

# My Start in 3D

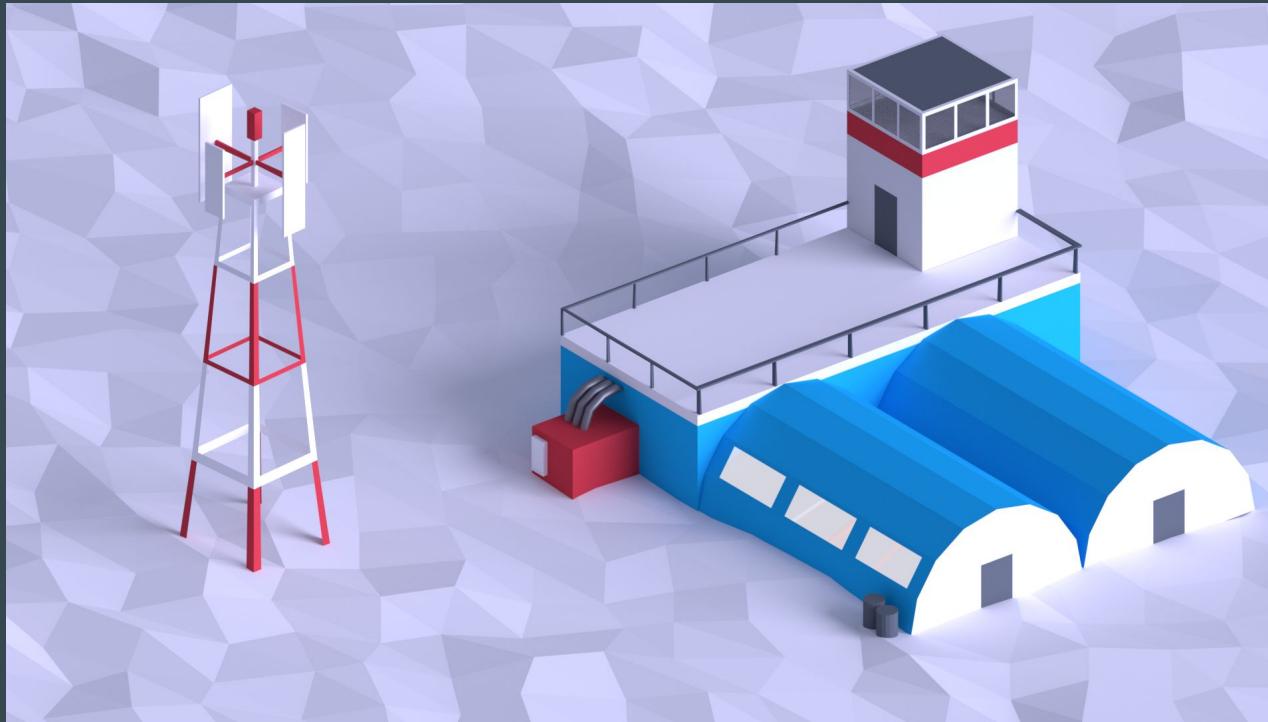
# My Former Employer Was Creating Things Like This:



# I Started With This



# Then I Kept Going



# Create Anything We Want in 3D

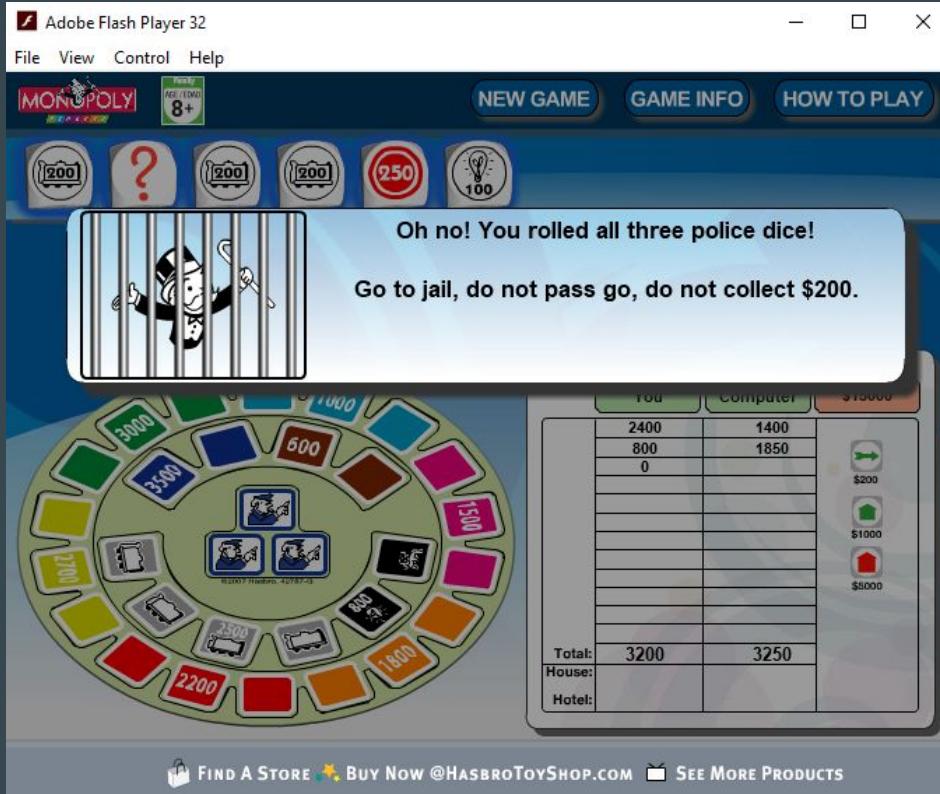




# My First Professional Games

# Monopoly Express (Flash)





# Sorry Express (Flash)

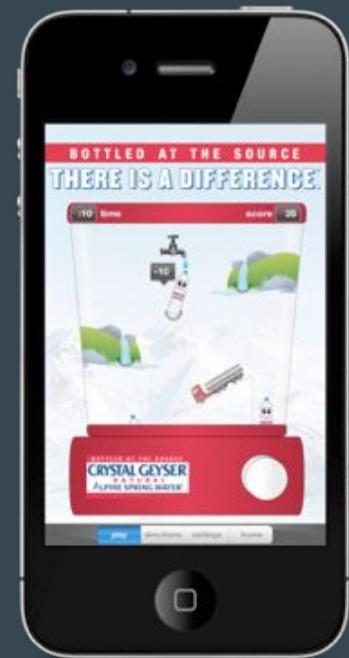
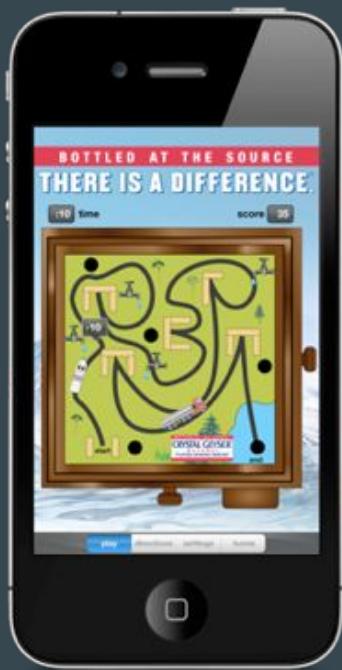




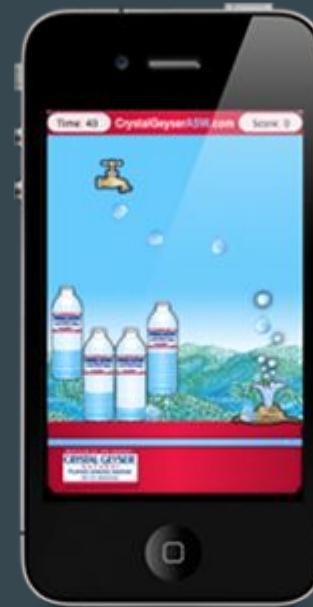
# Pictureka (Flash)



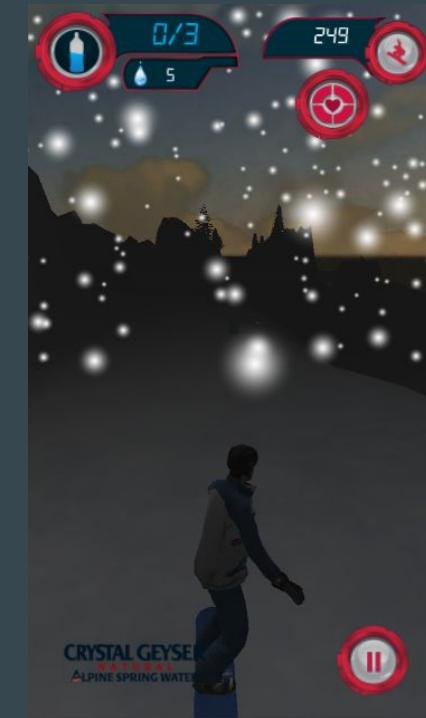
# Concepts We've Pitched In the Past



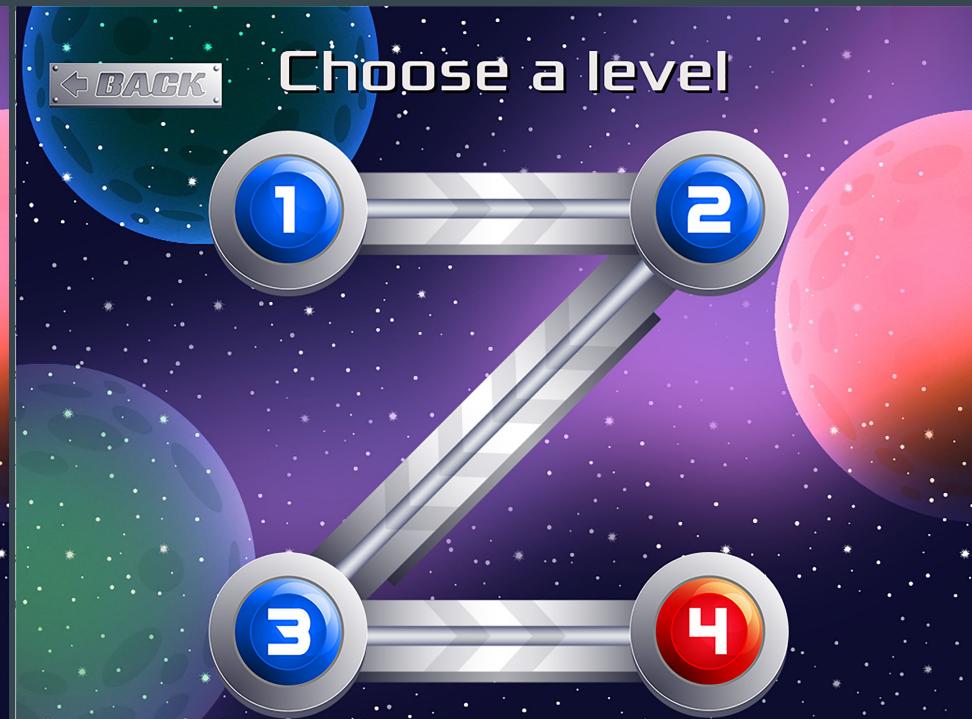
# Crystal Geyser Water (Corona SDK)



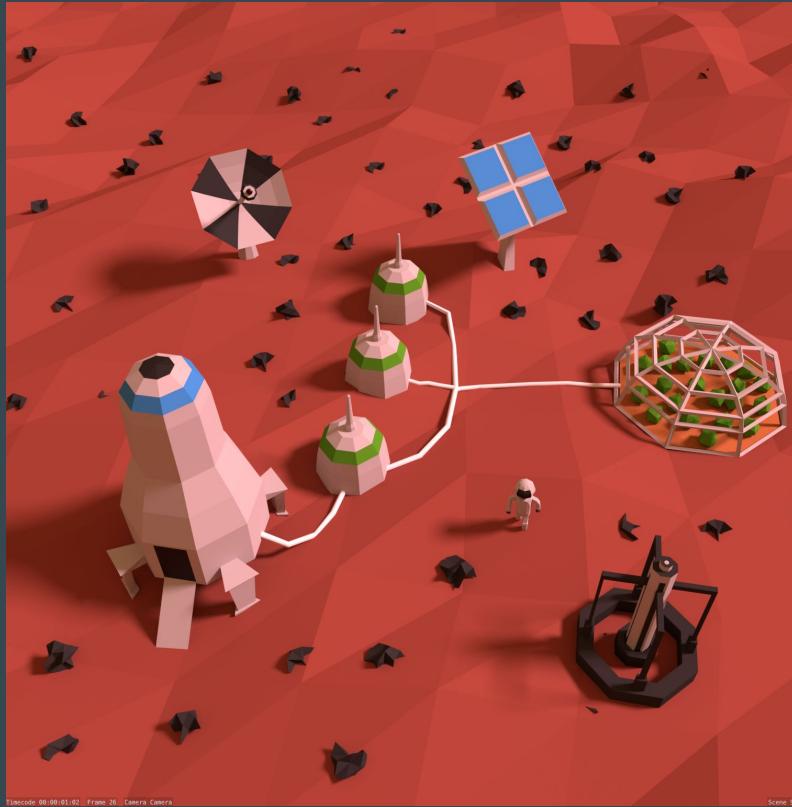
# Crystal Geyser Water: Power of Nature (Unity)



# Fraction Blast (Godot)



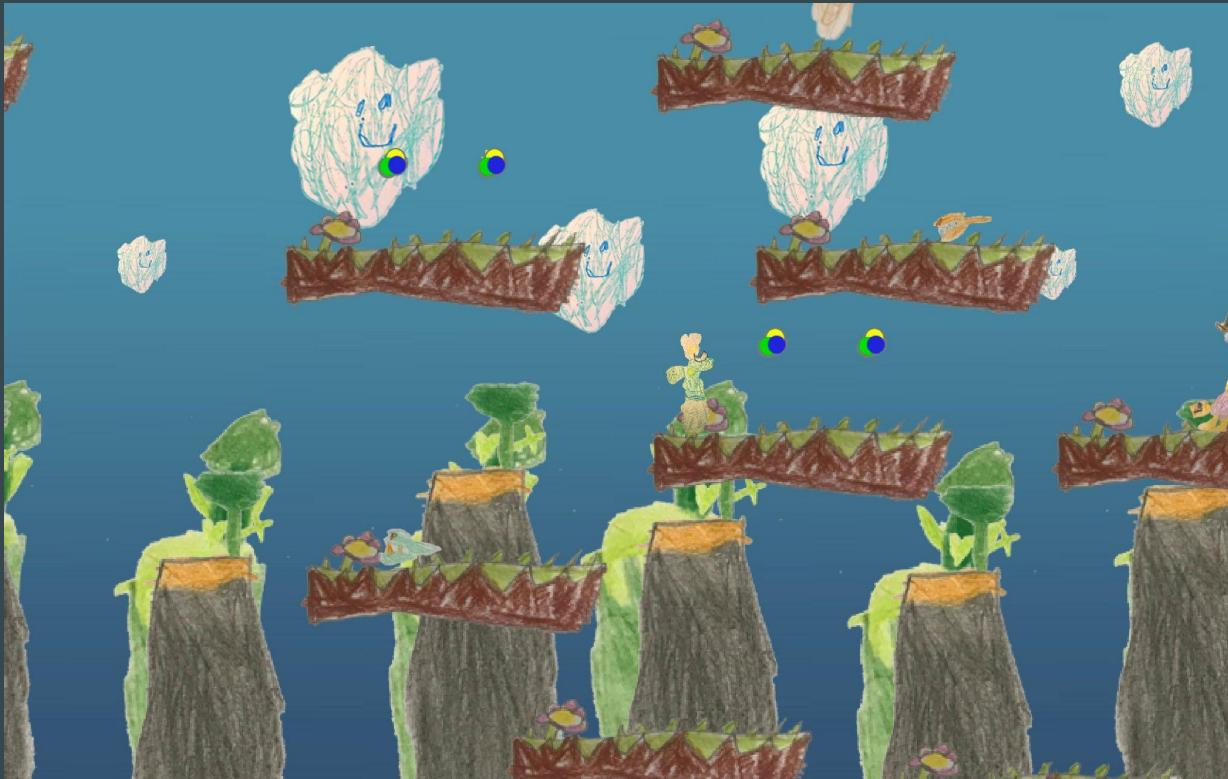
# Mars Survivor (WIP, Unreal Engine)



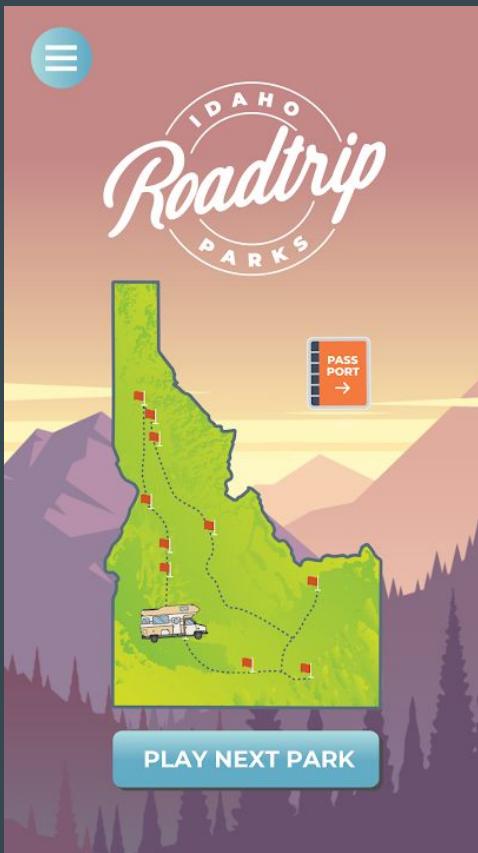
# Mobile Card Game



# Game Modding



# Idaho Lottery - Mobile

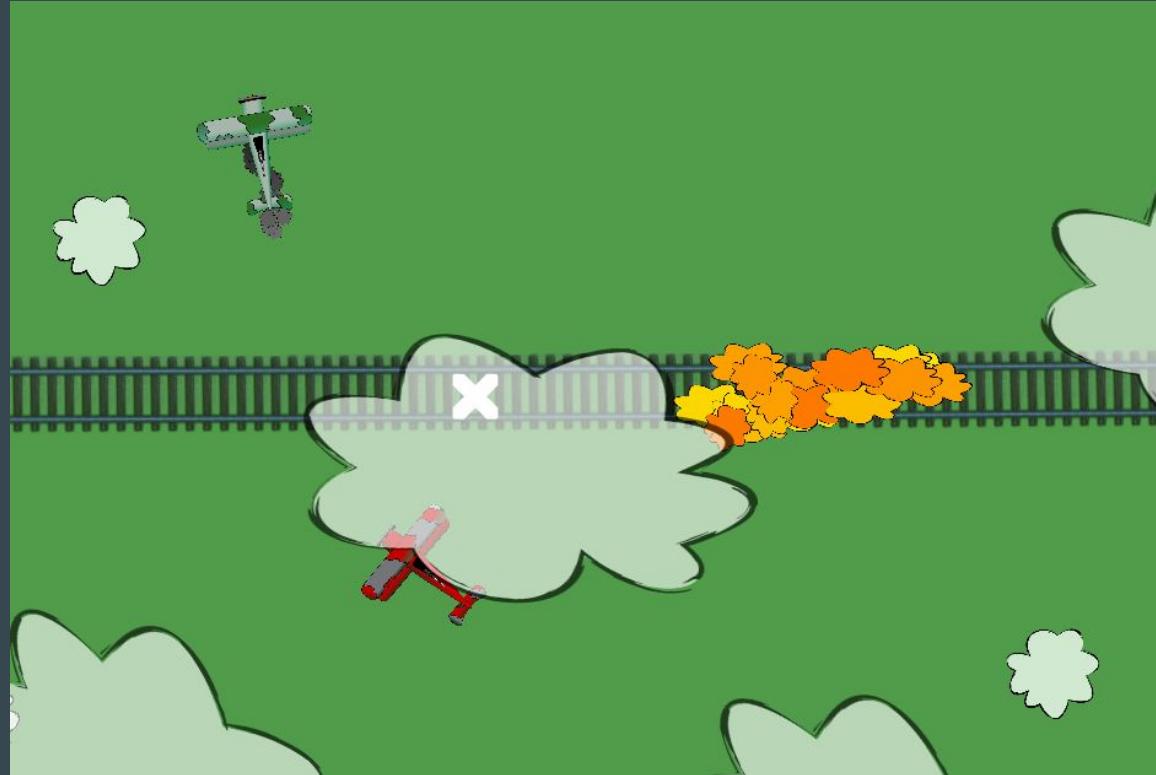


# Astral Wars - Mobile





# Great War Aces

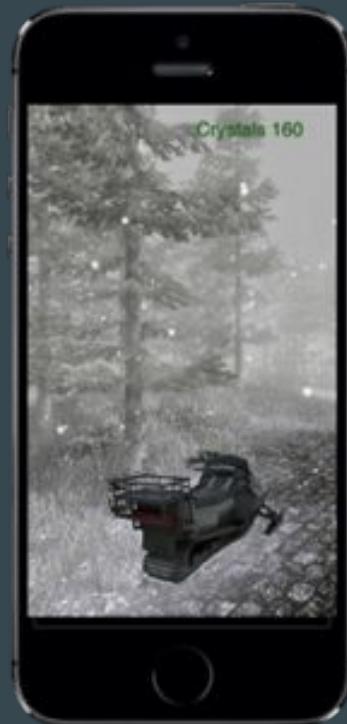
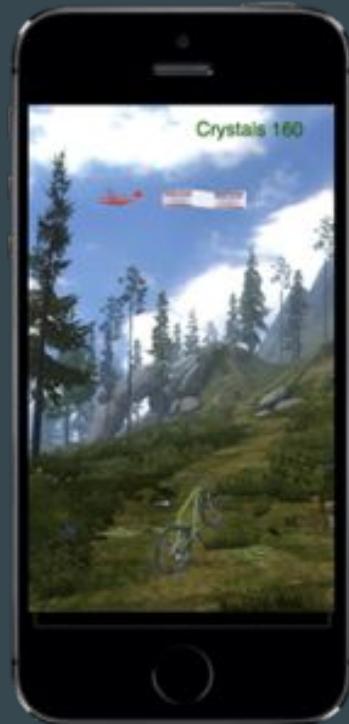


# Our Great War

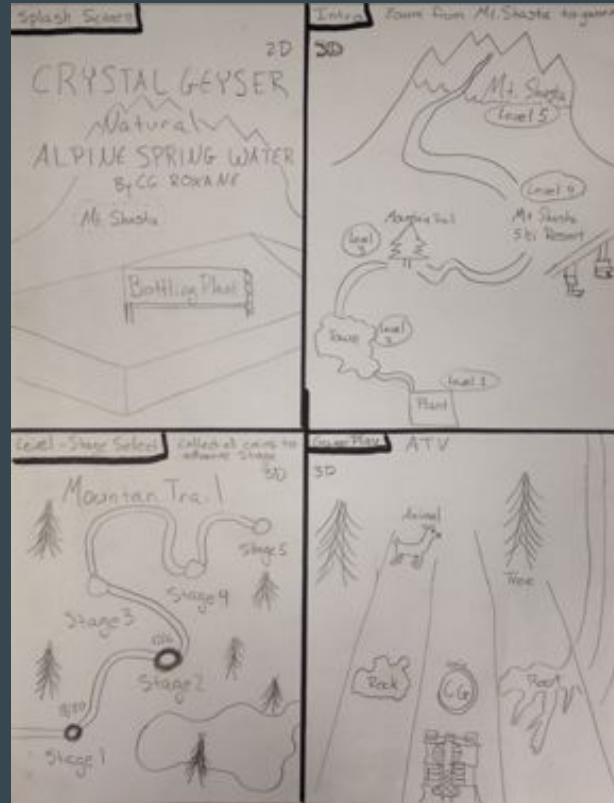


# Creating a Game - A Post Mortem

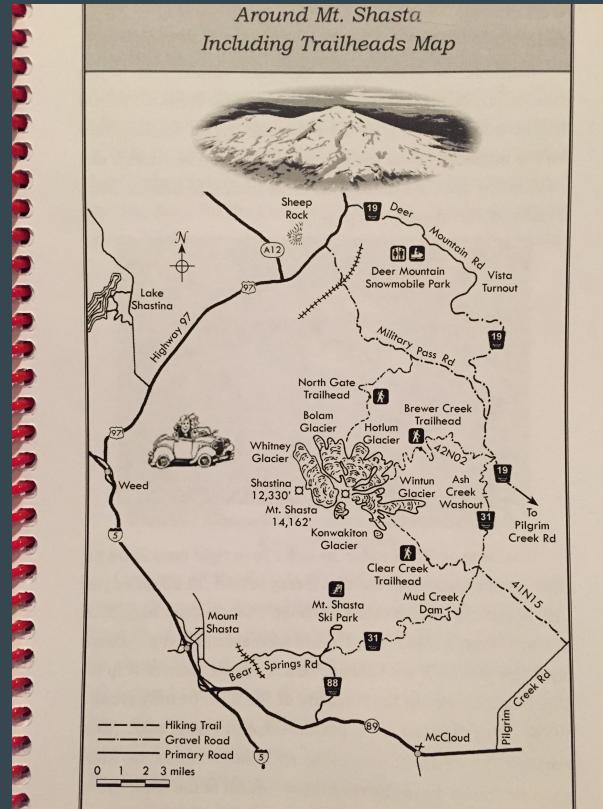
# The Pitch



# Concept Art - Seriously!



# Lots of Research



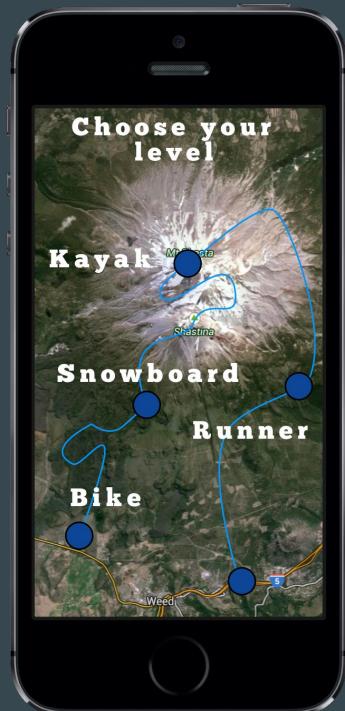
# References



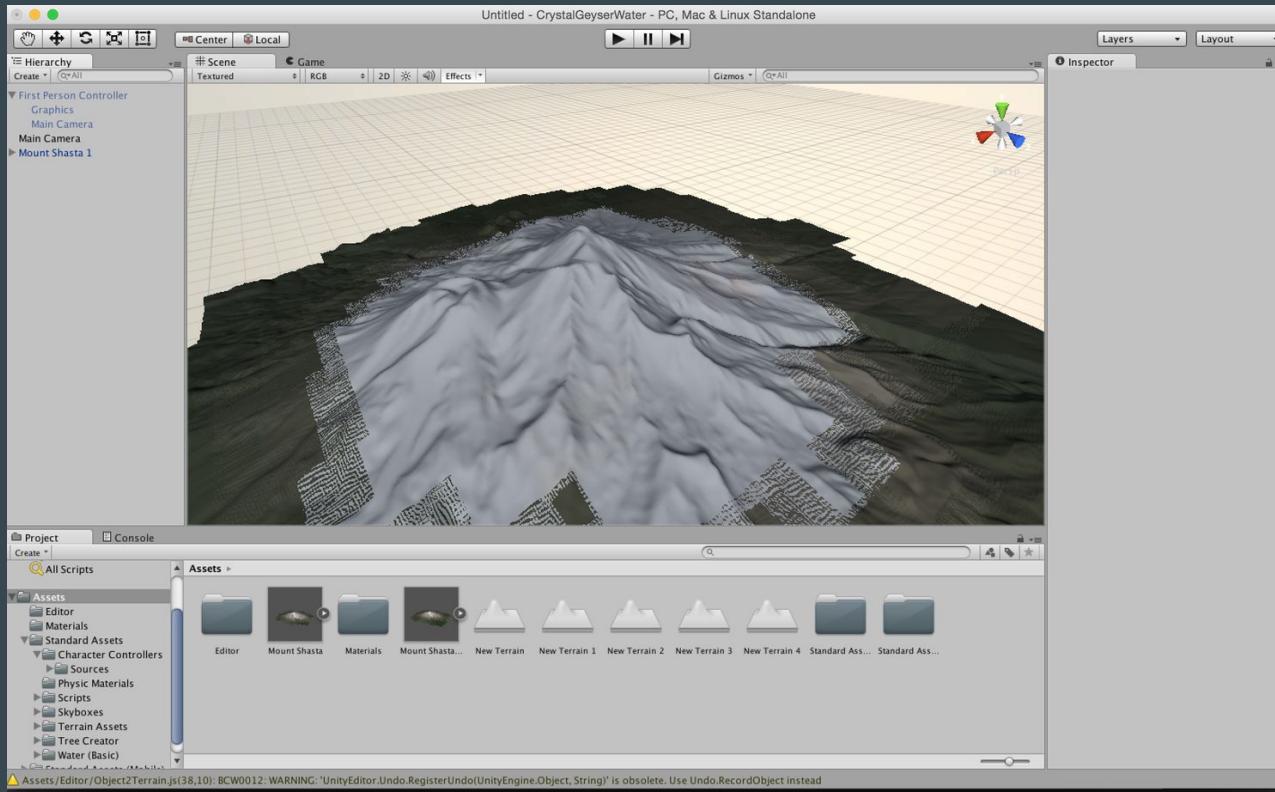
# Better Concept Art



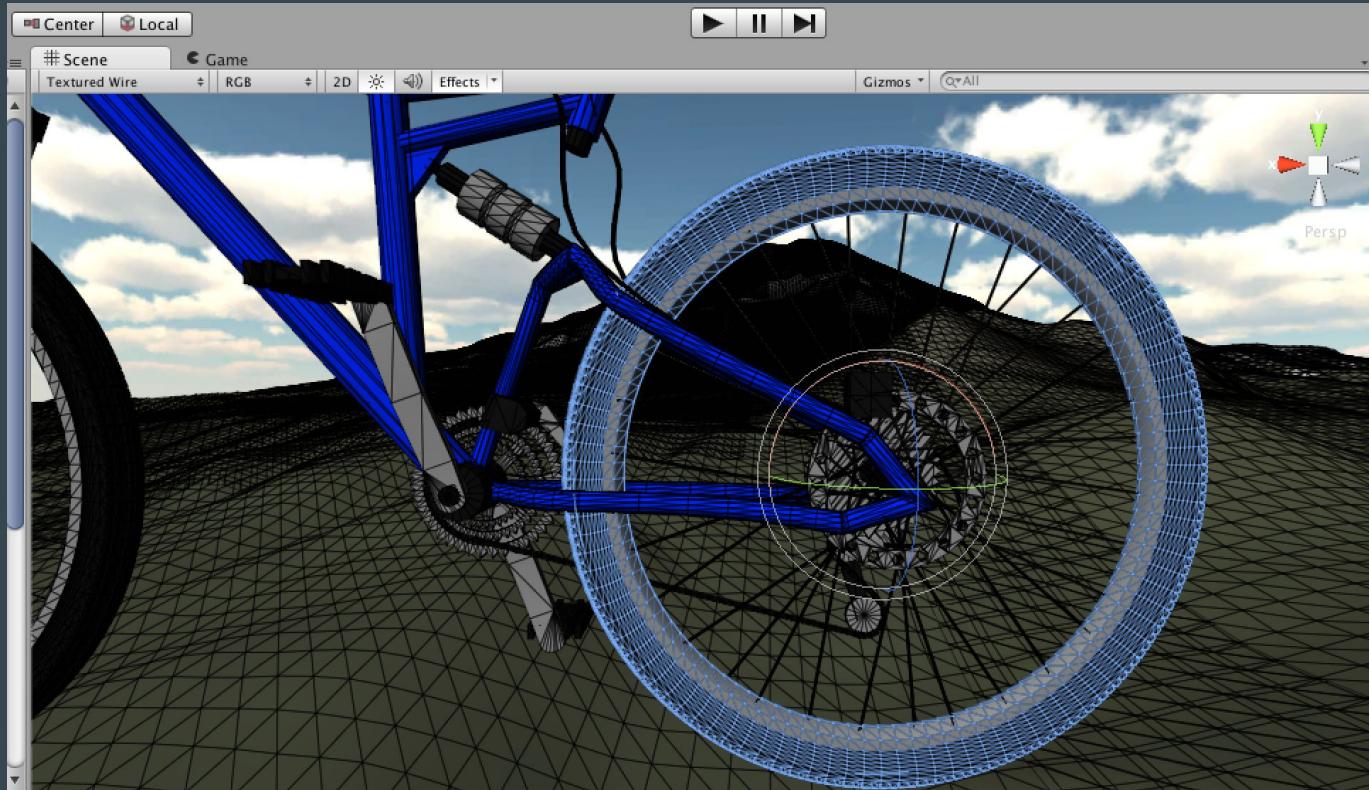
# More Concept Art



# Proofs of Concept - Too Much Geometry



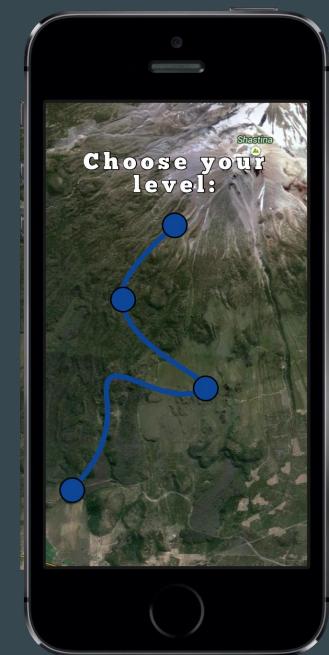
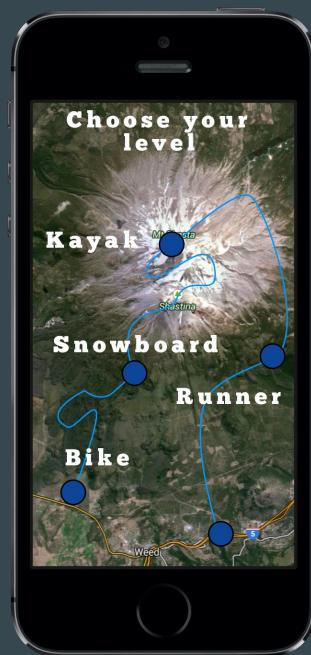
# Proofs of Concept - Too Much Geometry



# Final Result



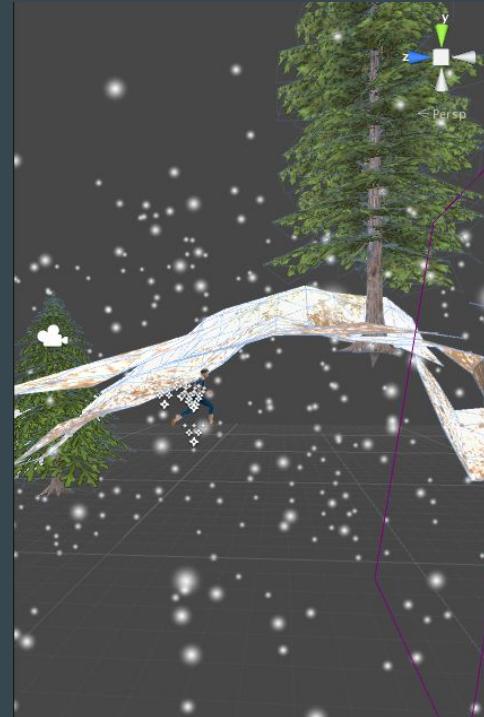
# More Concept Art



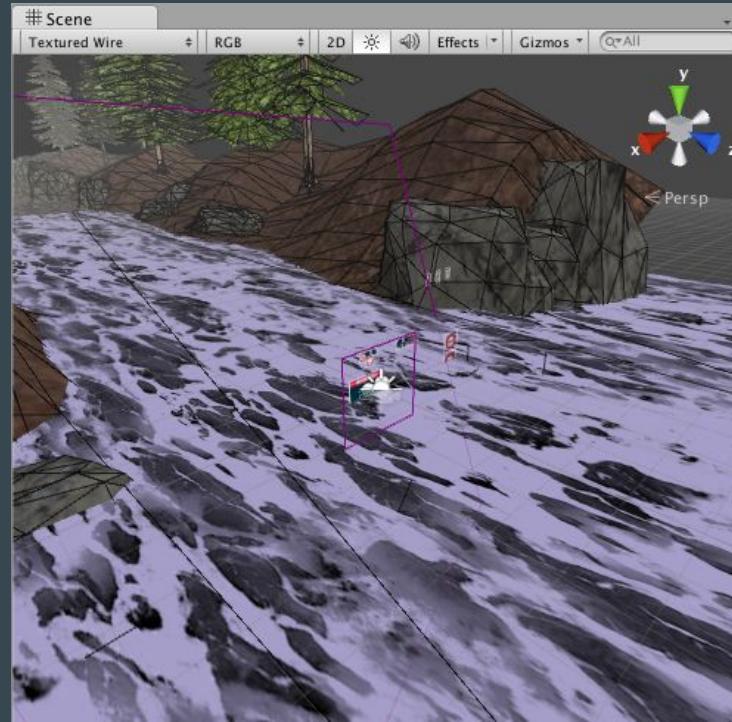
# Final Result



# In Game Editing



# More Editing



# Reuse 3D Assets



# Reuse Assets for Marketing



# Reuse Assets for Marketing



# Reuse Assets for Marketing



# Reuse Assets for Marketing



# Reuse Assets for Marketing



# Reuse Assets for Marketing



# Reuse Assets for Marketing



# Reuse Assets for Marketing



# Reuse Assets for Marketing



# Reuse Assets for Marketing



# Reuse Assets for Marketing



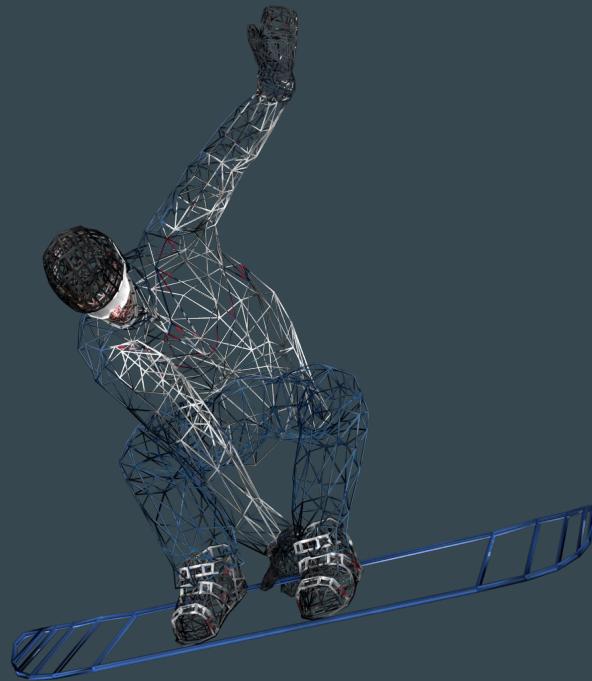
# Reuse Assets for Marketing - Get Artsy!



# Reuse Assets for Marketing - Get Artsy!



# Reuse Assets for Marketing - Get Artsy!



# Reuse Assets for Marketing - Get Artsy!



# Reuse Assets for Marketing - Get Artsy!

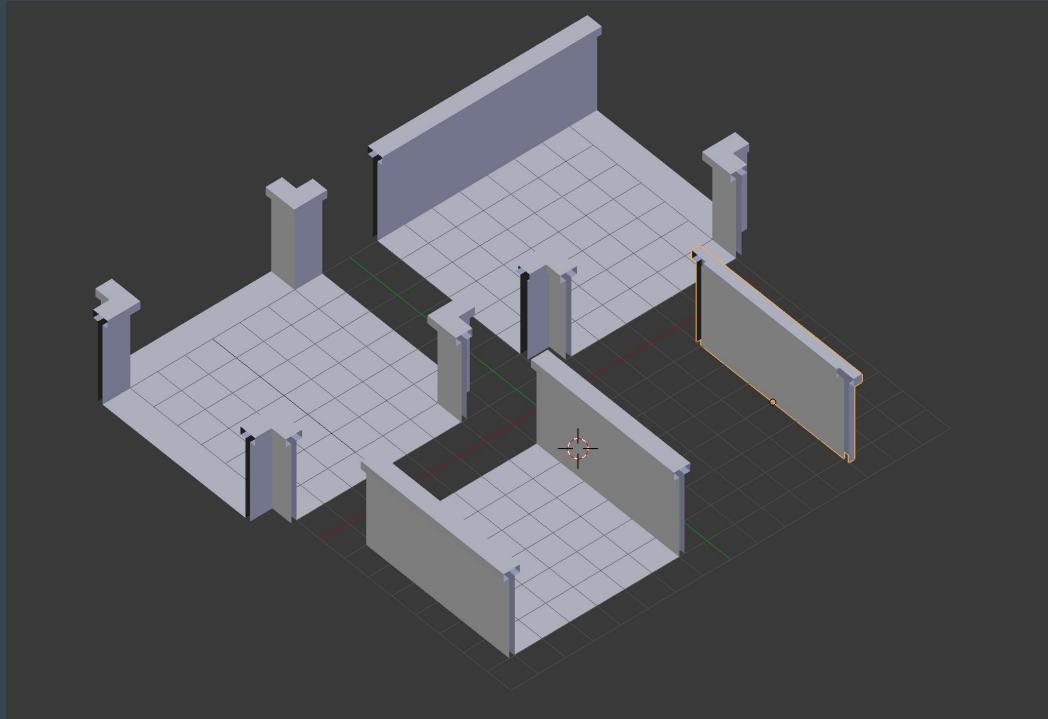


# What We'll Learn in this Course

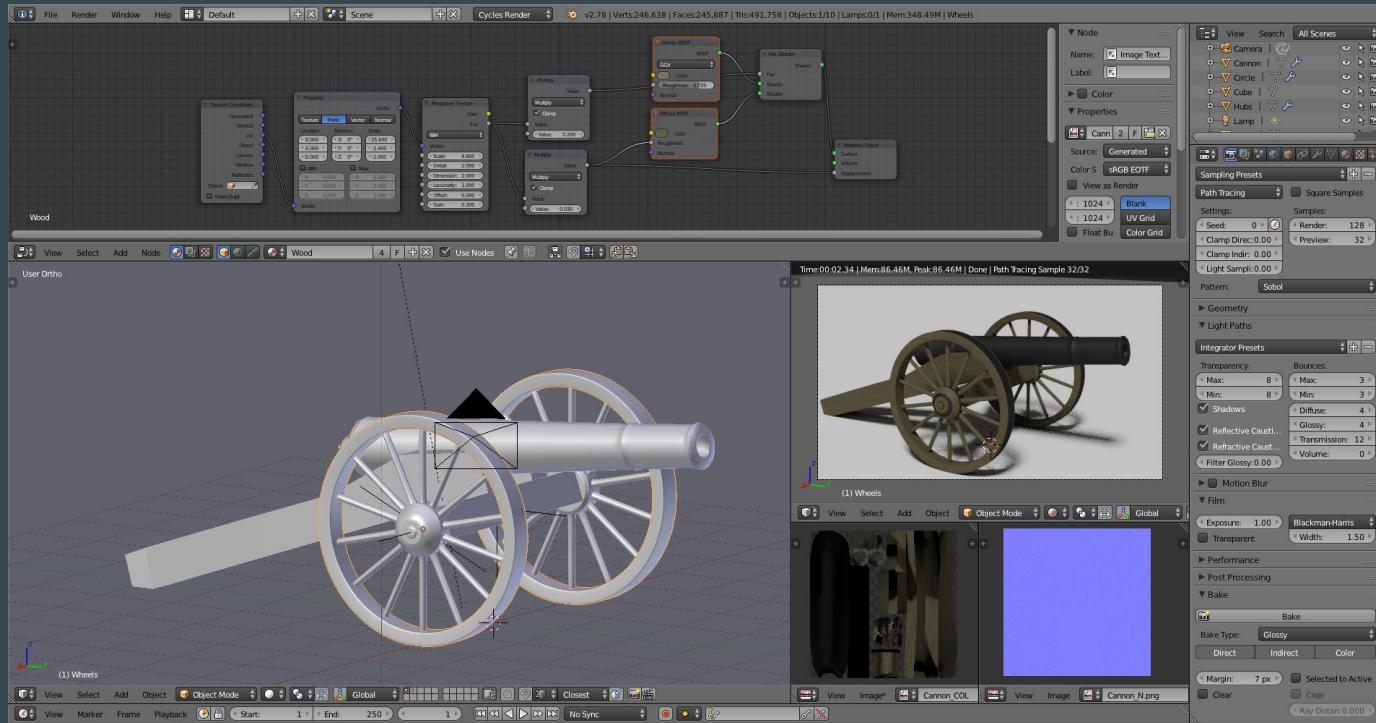
# 3D Modeling



# Level Creation in Blender



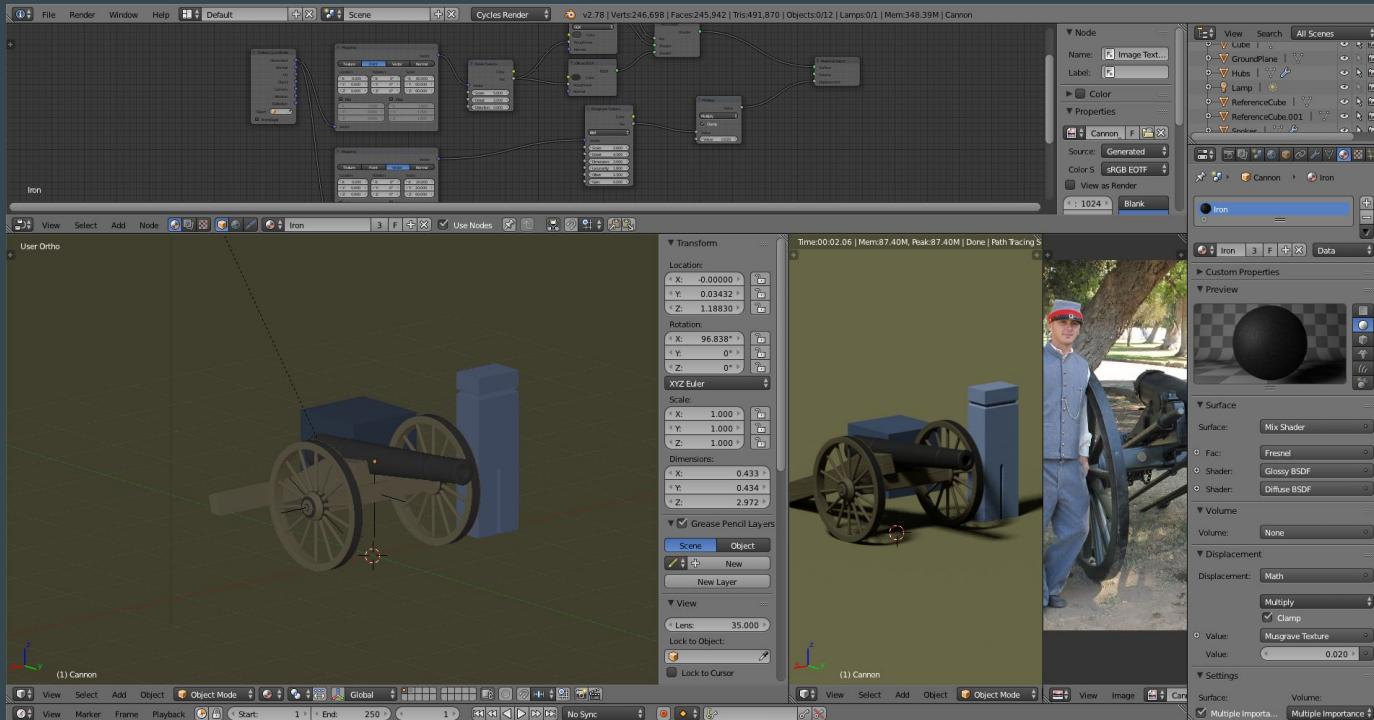
# Model and Texture Assets



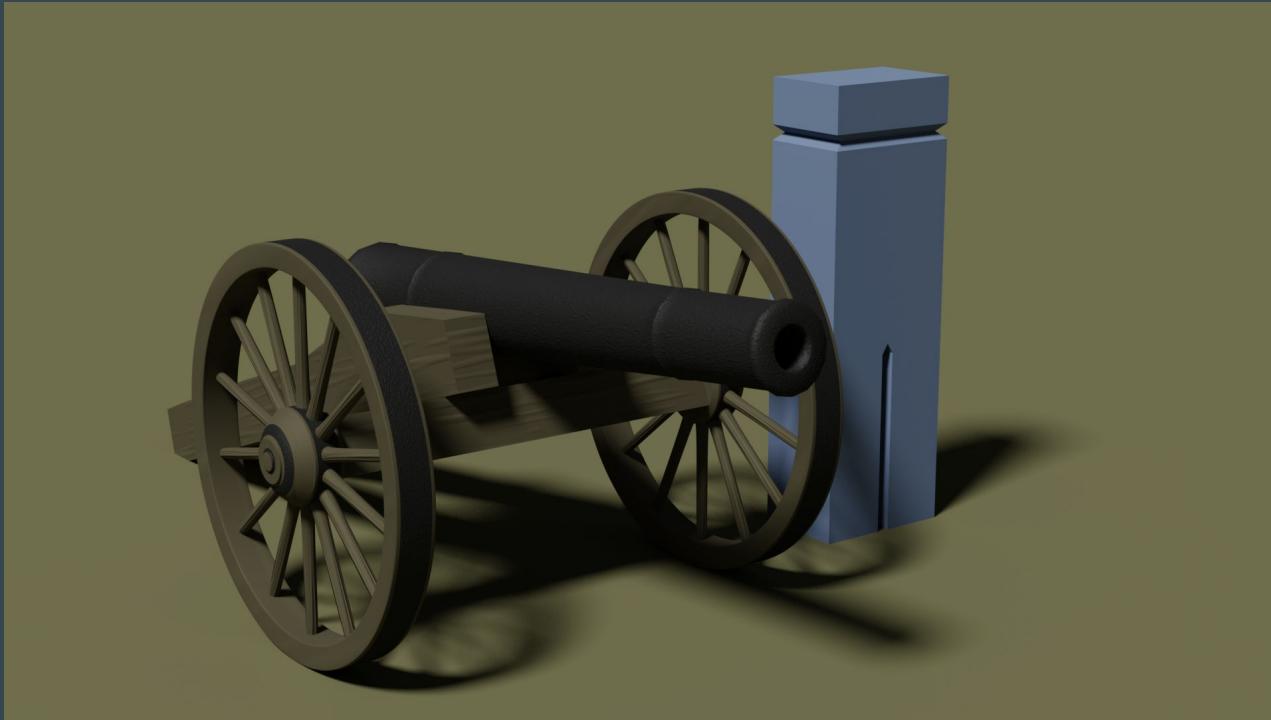
# Make Those Assets Look Good



# Prepare Asset for Export - Size, Orientation, Format



# Final Product



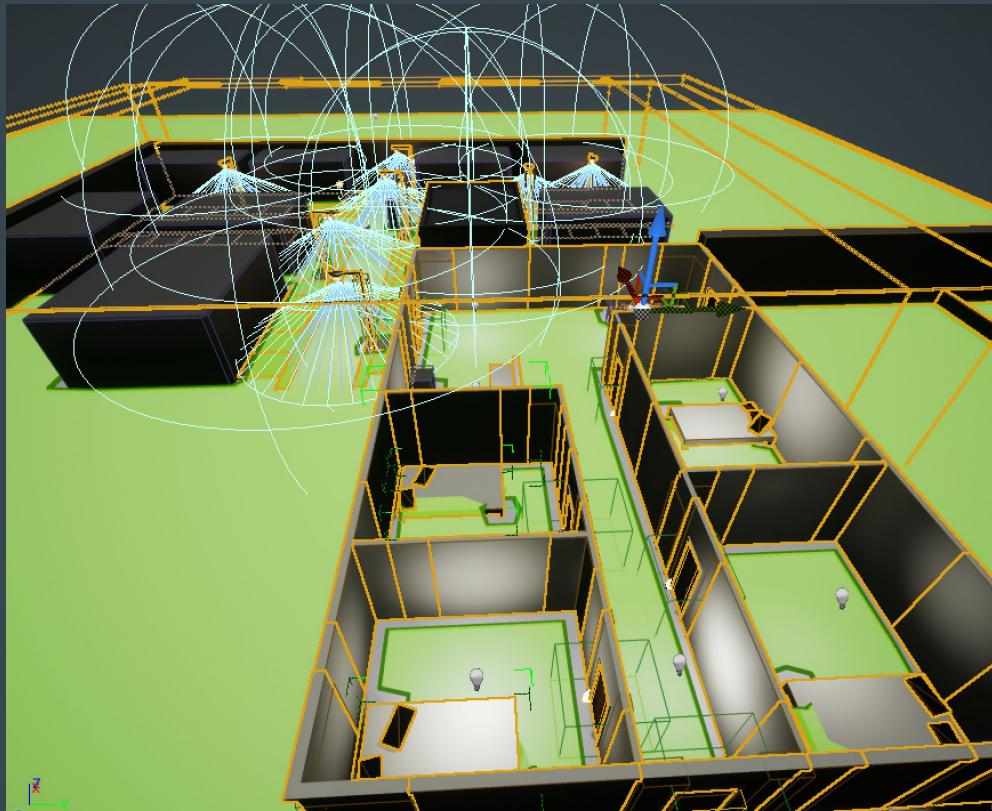
# Bring Those Into Unreal Engine



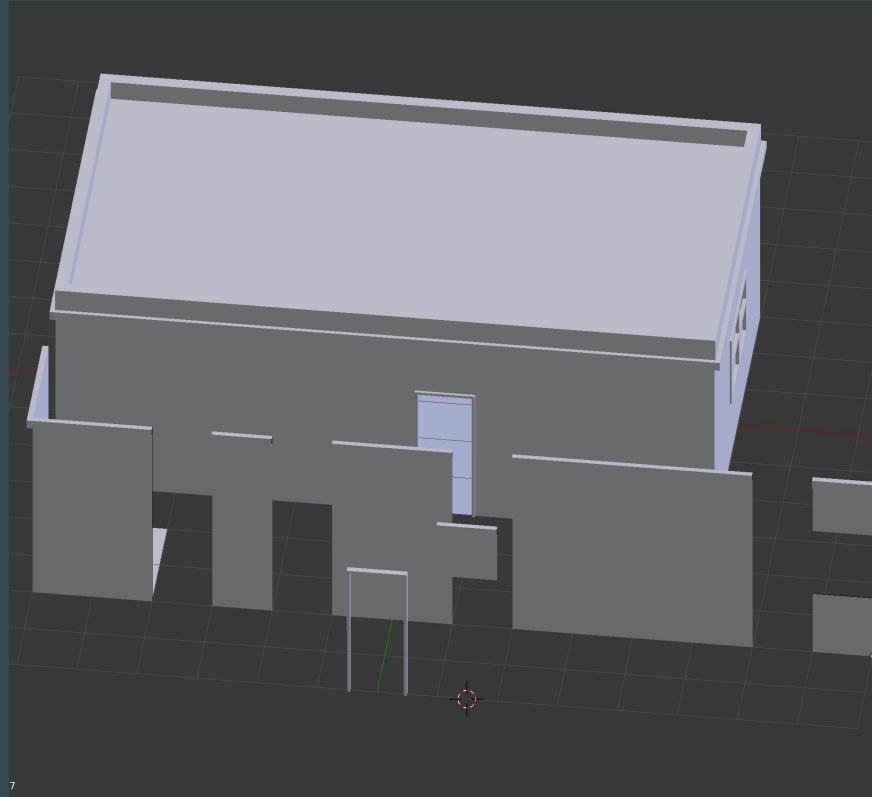
# Set the Lighting and Environment



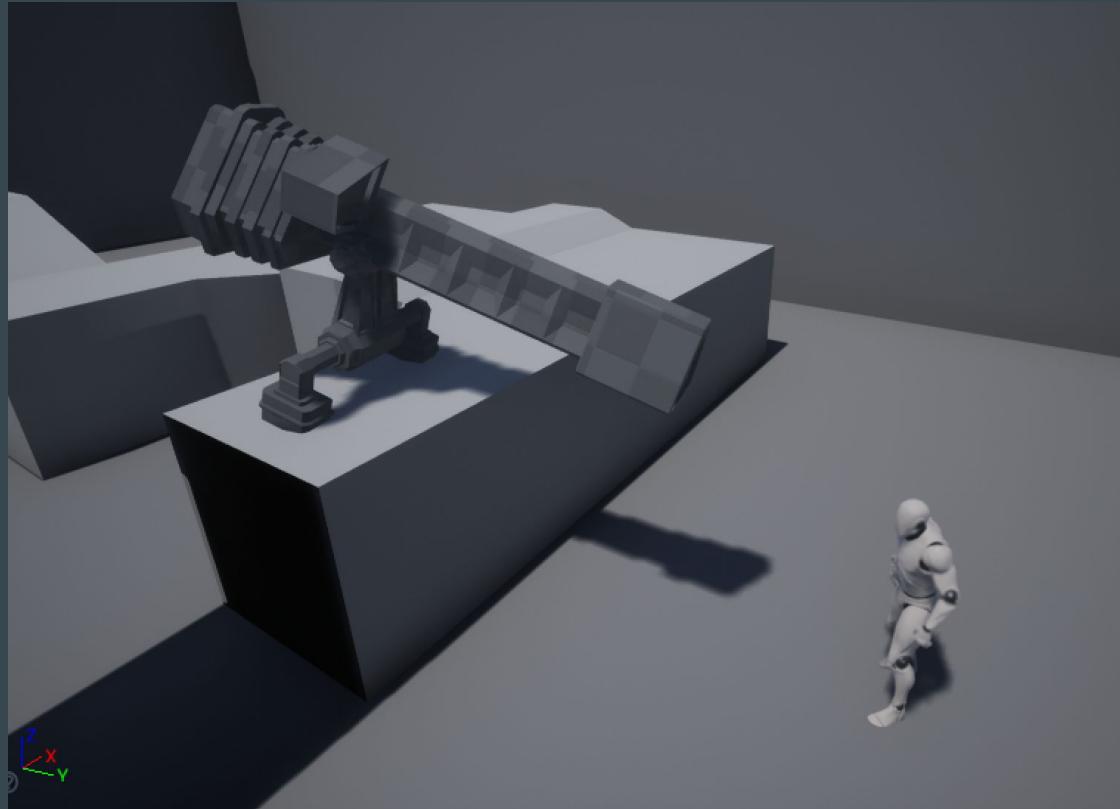
# Level Design



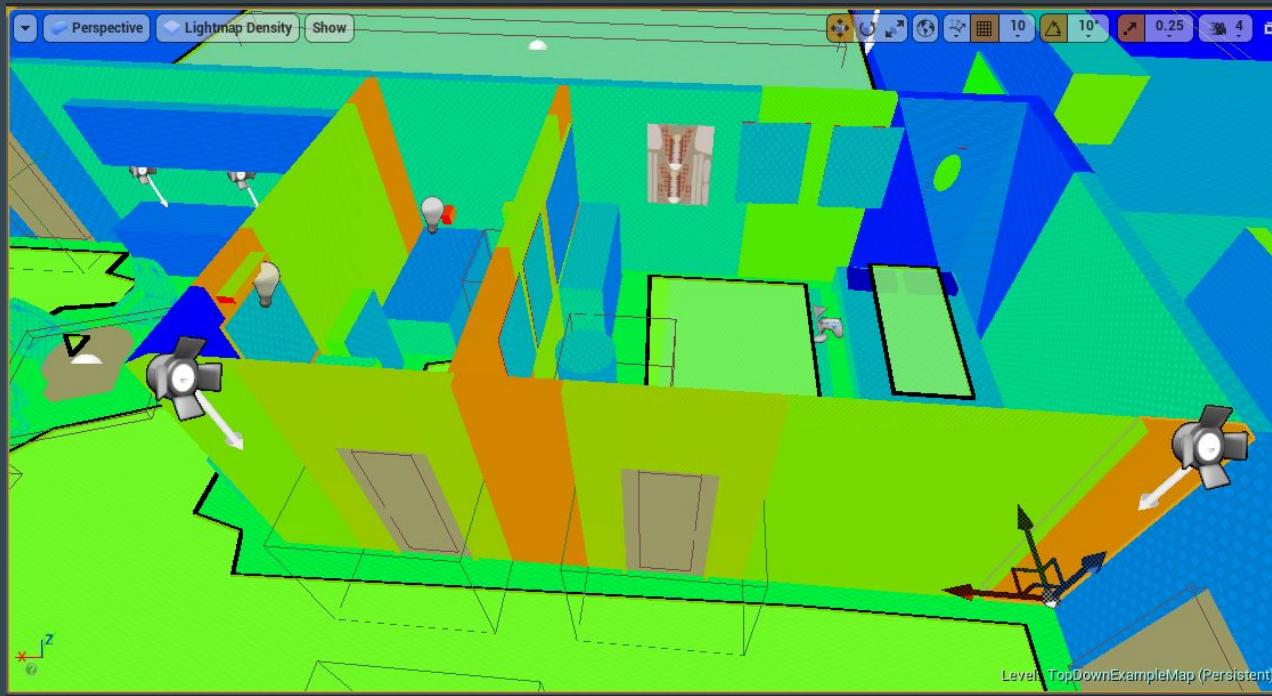
# Modular Level Design

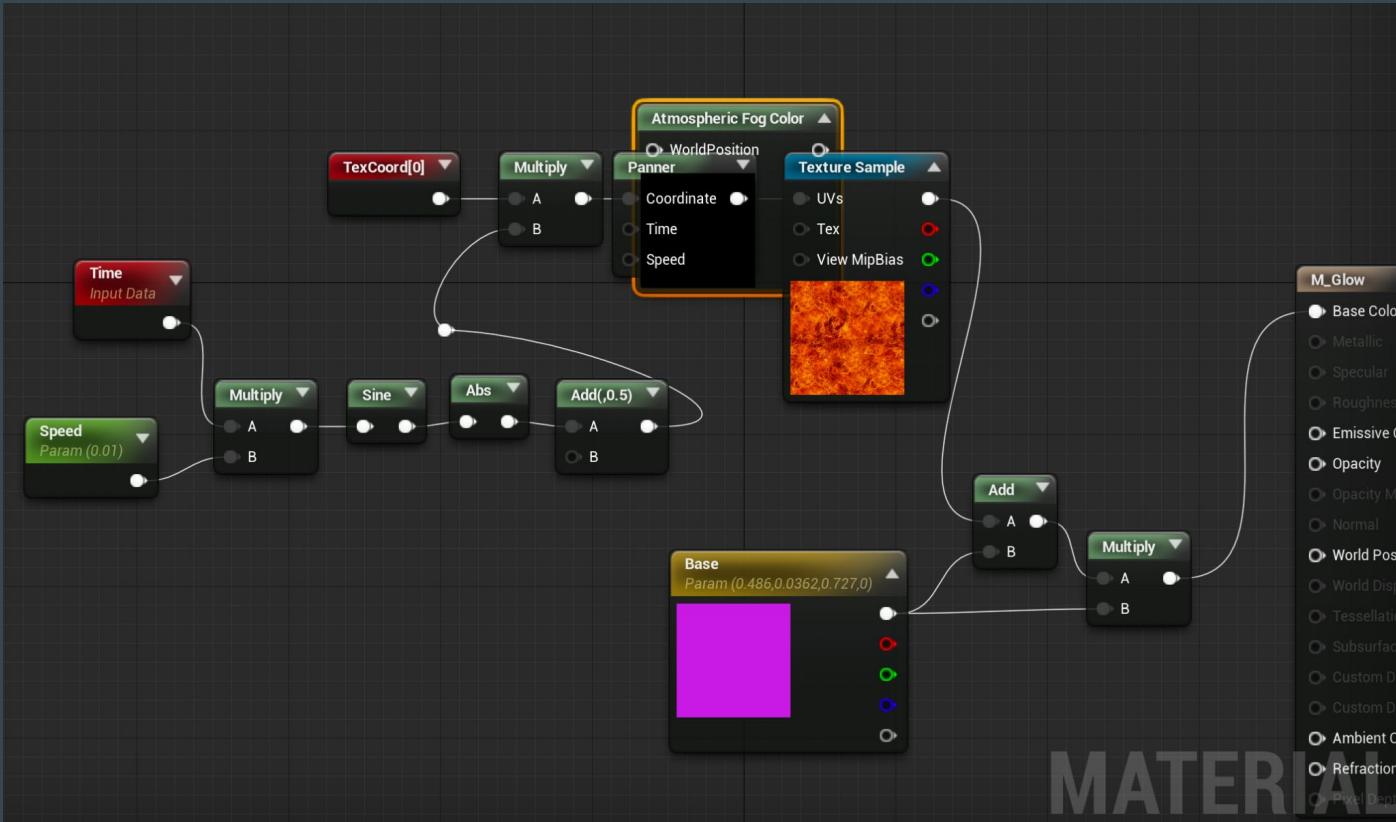


# Enemies and AI



# Engine Optimizations





# Learn Some C++ in Unreal Engine

```
#pragma once
#include "GameFramework/Character.h"
#include "DungeonCharacter.generated.h"

UCLASS(Blueprintable)
class ADungeonCharacter : public ACharacter
{
    GENERATED_BODY()

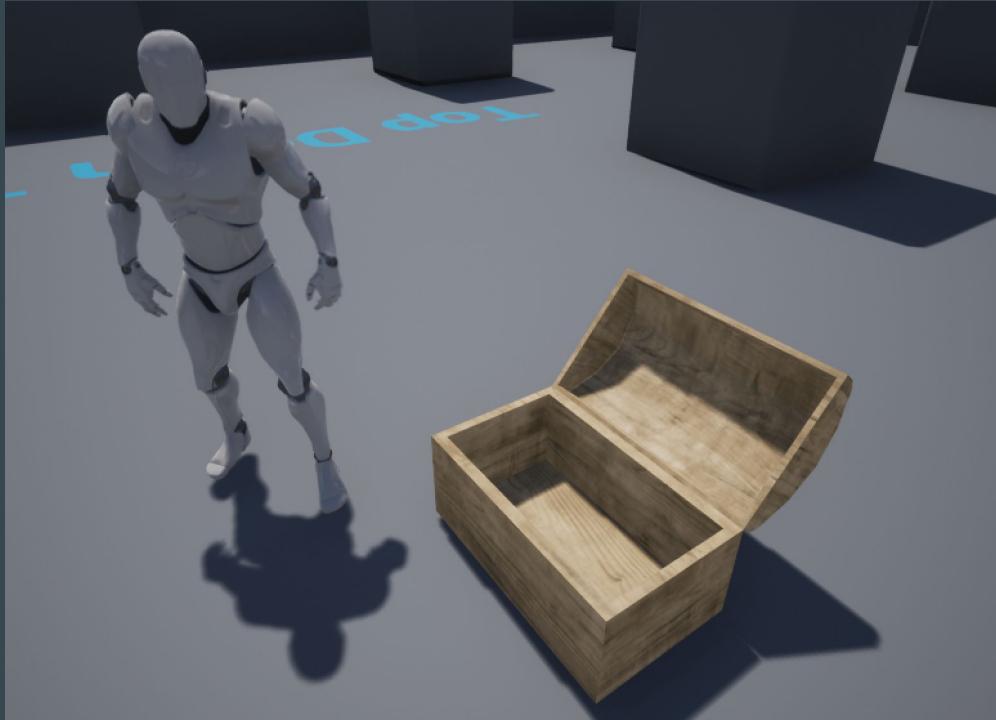
public:
    ADungeonCharacter();

    // Called every frame.
    virtual void Tick(float DeltaSeconds) override;

    /** Returns TopDownCameraComponent subobject */
    FORCEINLINE class UCameraComponent* GetTopDownCameraComponent() const { return TopDownCameraComponent; }
    /** Returns CameraBoom subobject */
    FORCEINLINE class USpringArmComponent* GetCameraBoom() const { return CameraBoom; }
    /** Returns CursorToWorld subobject */
    FORCEINLINE class UDecalComponent* GetCursorToWorld() { return CursorToWorld; }

    /** Returns collection sphere component */
    FORCEINLINE class USphereComponent* GetSphereComponent() const { return CollectionSphereComponent; }
```

# Play Test



# And We'll End With A Game



We're cover everything in between and more!

# Tools

## 1. Blender

1. Other 3D modeling apps
2. Intro to Blender

## 2. Unreal

1. Other Game Engines
2. Intro to Unreal

## 3. Other apps

1. Photoshop (Gimp)
2. Audacity (Sound)

# Resources

Websites for textures, sounds, images, fonts, and much much more.

# After this course

- We'll cover the major topics in making games
- This is an intro course, so while we'll cover a lot of different topics
- It takes a lot of time to become an expert in any one of them
- You'll know and will have put into practice the major concepts but like any art form it takes practice.

# Course Format

1. Homework each week
2. Published game each week (choose among a list)
3. Game design concepts
4. Game business topic

This is an art form. You'll be heavily graded on your creativity but technical ability counts for a lot too!

# Format

- First ~15 minutes: Game news, general game discussion, game business
- Game Design: ~15 minutes
- Lecture: ~30 minutes
- Building games: ~45 minutes - 1 hour 15
- Questions and coding help: ~15 minutes or as long as needed

Other things we'll do:

- Peer critiques: Show your progress, get critiqued
- Game discussions: Understand games, live games, play games

# Resources

Posted in Discord

Notes and other course materials will be in the GitHub repo