Dungeon Architect Programming Guide (C++)

for Unreal Engine 4

1 Introduction

Dungeon Architect is a plugin for Unreal Engine 4 Editor that lets you quickly build procedural levels for your games. This guide helps you work with the plugin using C++

View in Youtube

2 Project Setup

Add Dungeon Architect to your game project's Build.cs file so you can access the plugin

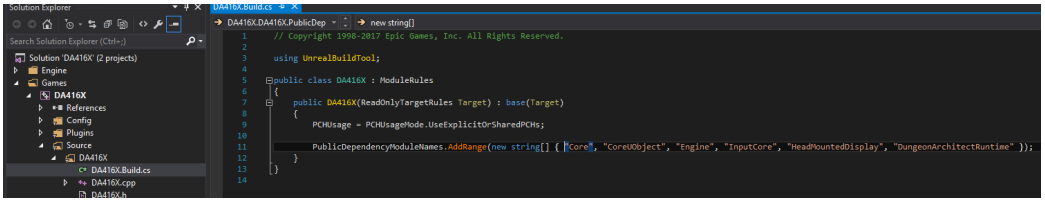

Figure 1: Setup Project.Build.cs

Open the file **Source/PROJECT_NAME/PROJECT_NAME.Build.cs**

Add **DungeonArchitectRuntime** to the list of PublicDependencyModuleNames

```
// Copyright 1998-2017 Epic Games, Inc. All Rights Reserved.

using UnrealBuildTool;

public class PROJECT_NAME : ModuleRules
{
    public PROJECT_NAME(ReadOnlyTargetRules Target) : base(Target)
    {
        PCHUsage = PCHUsageMode.UseExplicitOrSharedPCHs;

        PublicDependencyModuleNames.AddRange(new string[] { "Core", "CoreUObject", "Eng
            "DungeonArchitectRuntime" });
    }
}
```

3 Build at runtime

Build a random dungeon at runtime. This assumes that you have a dungeon in your level with the themes already setup.

The code sample is placed in the GameMode class on StartPlay

GameMode Header file

```
// Copyright 1998-2017 Epic Games, Inc. All Rights Reserved.

#pragma once

#include "CoreMinimal.h"
#include "GameFramework/GameModeBase.h"
#include "DA416XGameMode.generated.h"

UCLASS(minimalapi)
class ADA416XGameMode : public AGameModeBase
{
    GENERATED_BODY()

public:
    ADA416XGameMode();

    virtual void StartPlay() override;
};
```

GameMode Source file