**Extracting the carrying object**

We will create an actor with collision box to extract.

First create an actor called FPSExtraction

Then create a box collision in it through C++.

Add following to the Header file (Under protected).

UPROPERTY(VisibleAnywhere, Category = "Components")

UBoxComponent\* OverlapComp;

forward decleration required.

class UBoxComponent;

#include "Components/BoxComponent.h"

Then go to the C++ file and in constructor add the following

OverlapComp = CreateDefaultSubobject<UBoxComponent>(TEXT("OverlapComp"));

// create the box

OverlapComp->SetCollisionEnabled(ECollisionEnabled::QueryOnly);

// set collision on spatial queries (raycasts, sweeps, and overlaps)

OverlapComp->SetCollisionResponseToAllChannels(ECR\_Ignore);

// Ignore all collisions

OverlapComp->SetCollisionResponseToChannel(ECC\_Pawn, ECR\_Overlap);

// Setting collision to only Pawn overlap

OverlapComp->SetBoxExtent(FVector(200.0f));

// Setting box scale

RootComponent = OverlapComp;

// Making it root component

Now we need an event for overlap detection. Alternative to the way we did in actornotifybeginoverlap used in ObjectiveActor is as follows

we are going to bind specifically to 'OnComponentBeginOverlap' method. Type the following

OverlapComp->OnComponentBeginOverlap.AddDynamics(this, &AFPSExtractionZone::HandleOverlap); (intellisence will not work at this point) HandleOverlap function is not yet declared. So declare the function in header file as follows.

UFUNCTION()

void HandleOverlap();

Function requires some parameters. To get it do the following.

Right click on 'OnComponentBeginOverlap' and go to definition. you will be taken to 'PremitiveComponent.h' to the following line of code

FComponentBeginOverlapSignature OnComponentBeginOverlap;

Right click on 'FComponentBeginOverlapSignature' and go to definition. You will be taken to the following line in the same file.

DECLARE\_DYNAMIC\_MULTICAST\_DELEGATE\_SixParams( FComponentBeginOverlapSignature, UPrimitiveComponent\*, OverlappedComponent, AActor\*, OtherActor, UPrimitiveComponent\*, OtherComp, int32, OtherBodyIndex, bool, bFromSweep, const FHitResult &, SweepResult);

We need to copy the parameters in that method and paste it in the decleration of HandleOverlap function (remove the commas)

The final HandleOverlap function should look like this.

UFUNCTION()

void HandleOverlap(UPrimitiveComponent\* OverlappedComponent, AActor\* OtherActor, UPrimitiveComponent\* OtherComp, int32 OtherBodyIndex, bool bFromSweep, const FHitResult & SweepResult);

now in the C++ file create the new definitions of function HandleOverlap as follows(just to see the function work at the moment)

void AFPSExtractionZone::HandleOverlap(UPrimitiveComponent \* OverlappedComponent, AActor \* OtherActor, UPrimitiveComponent \* OtherComp, int32 OtherBodyIndex, bool bFromSweep, const FHitResult & SweepResult)

{

UE\_LOG(LogTemp, Log, TEXT("Overlapped with extraction zone"));

}

In the Editor add the FPSExtractionBox in the world and play and go over it to see the message in output log.

can add the following in constructor to see the box while playing;

OverlapComp->SetHiddenInGame(false);

**References**

https://docs.unrealengine.com/en-US/Programming/UnrealArchitecture/Delegates/index.html