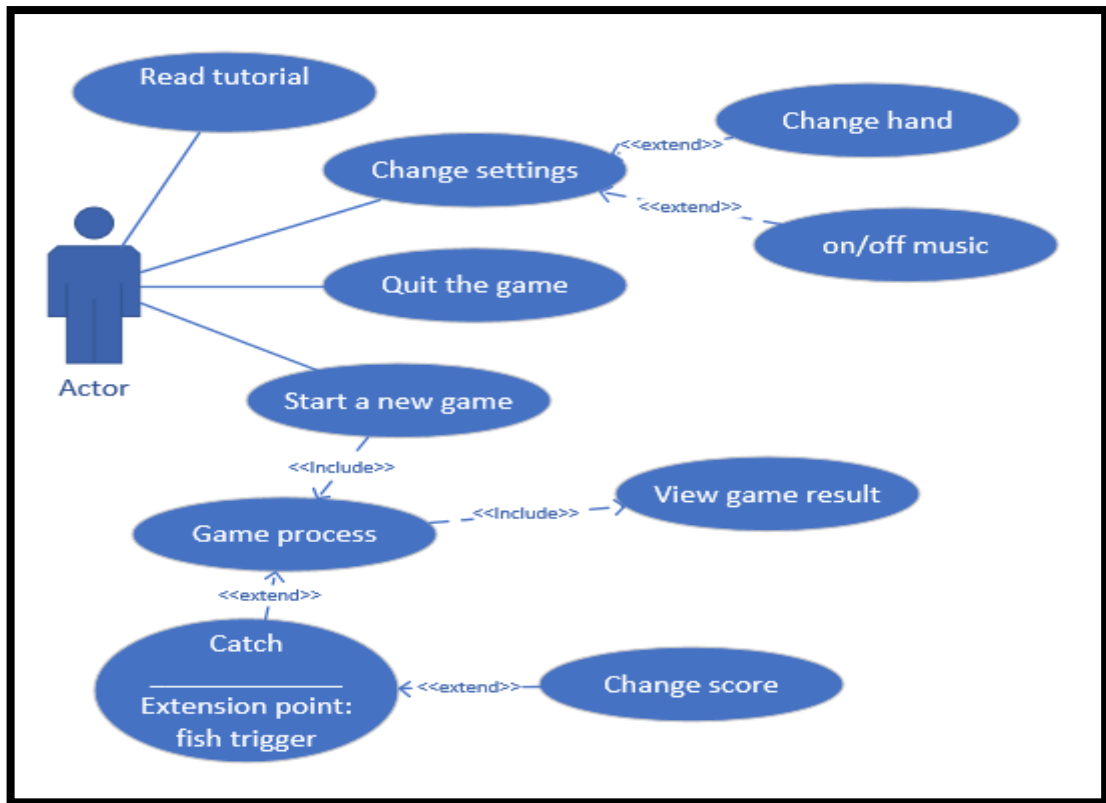


ДОДАТОК А

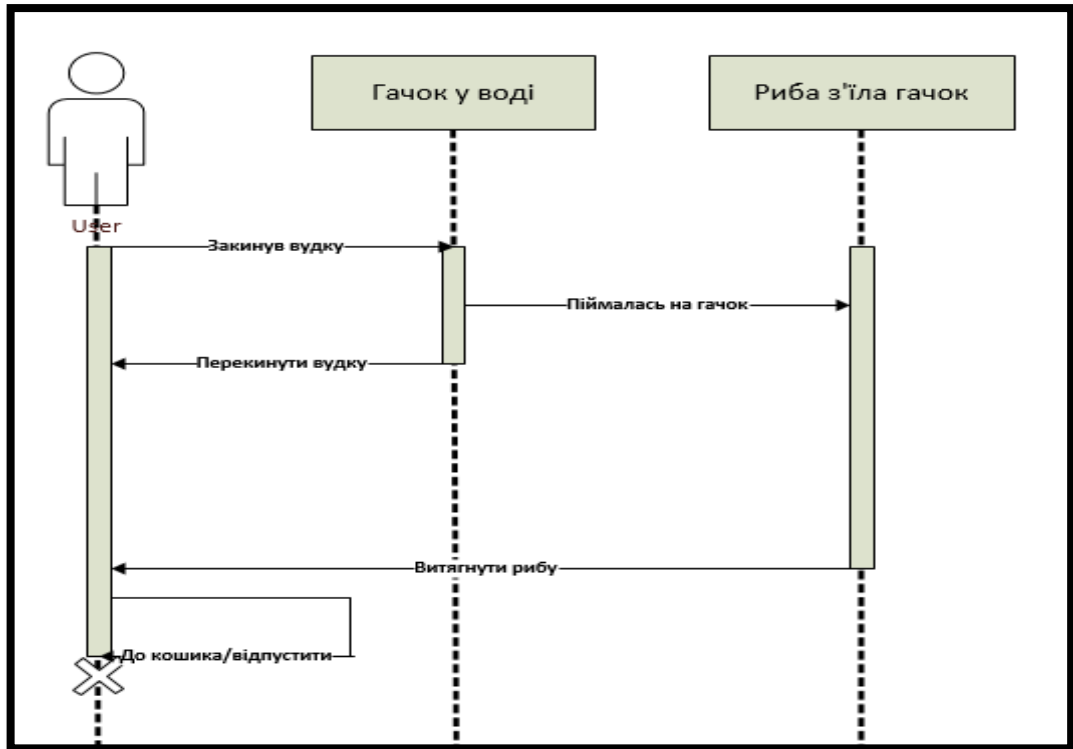
ГРАФІЧНИЙ МАТЕРІАЛ АТЕСТАЦІЙНОЇ РОБОТИ

## UML-МОДЕЛЮВАННЯ СИСТЕМИ



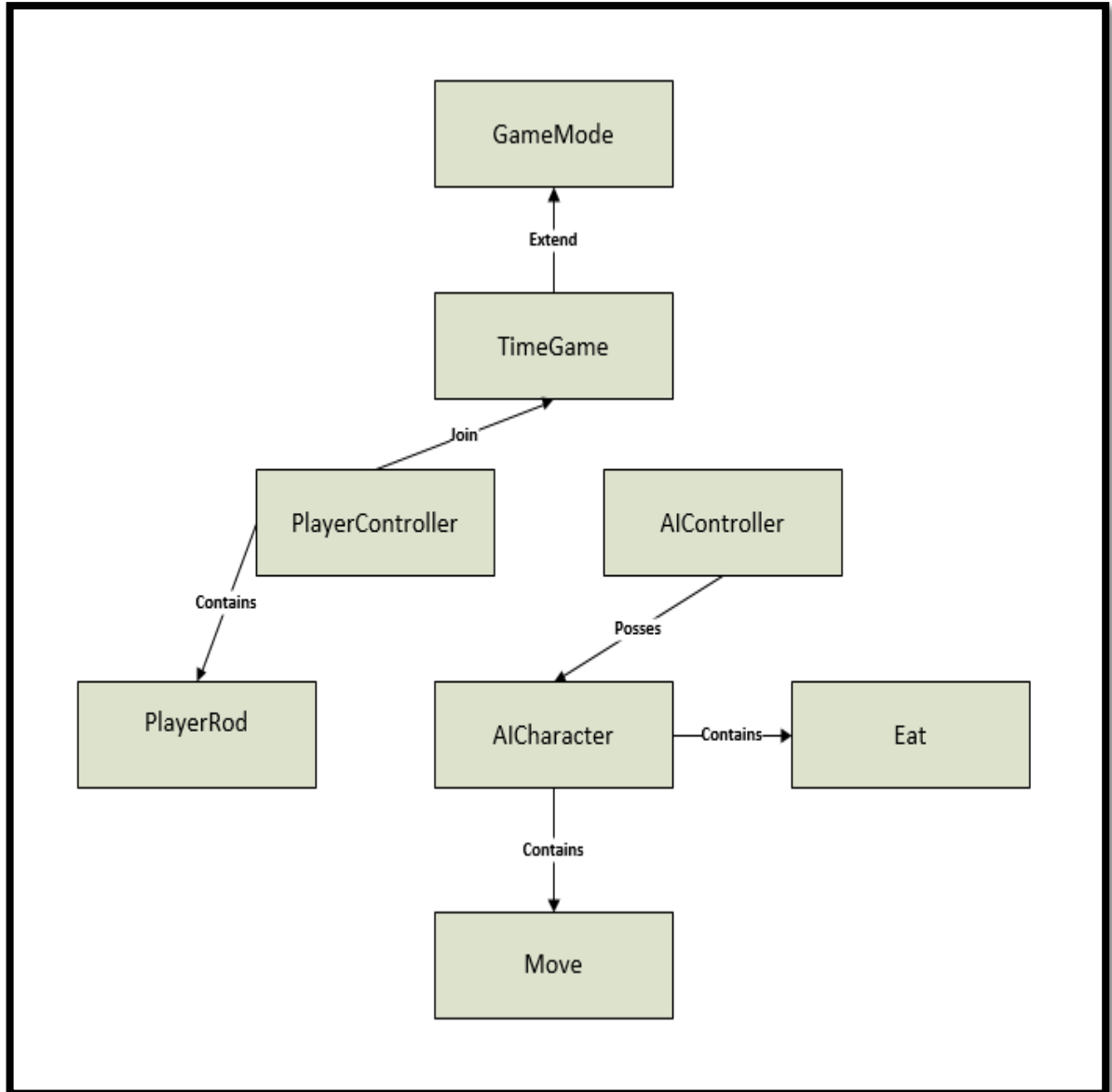
Діаграма Use case

Розробив	Орлов О.К.			Методи та технології розробки навчальних програм із застосуванням віртуальної реальності	Аркуш	Аркушів
Перевірила	Калита Н.І.				1	3
Н. контр.	Калита Н.І.				ХНУРЕ Кафедра СТ	
Затвердив	Гребеннік І.В.					



Діаграма послідовності

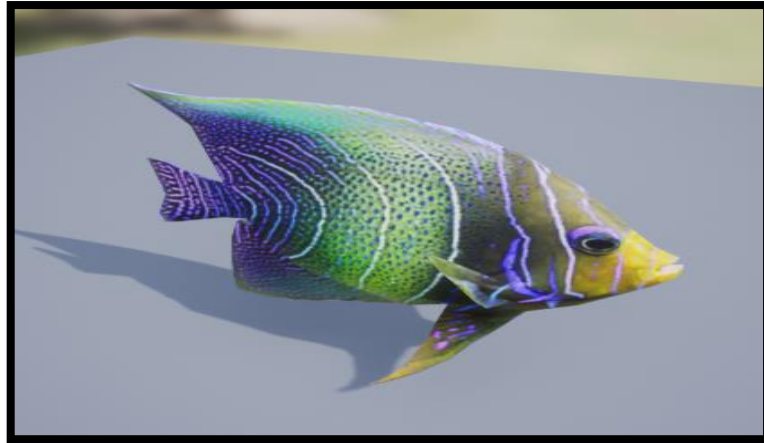
Розробив	Орлов О.К.			Методи та технології розробки навчальних програм із застосуванням віртуальної реальності	Аркуш	Аркушів
Перевірила	Калита Н.І.				2	3
Н. контр.	Калита Н.І.				ХНУРЕ Кафедра СТ	
Затвердив	Гребеннік І.В.					



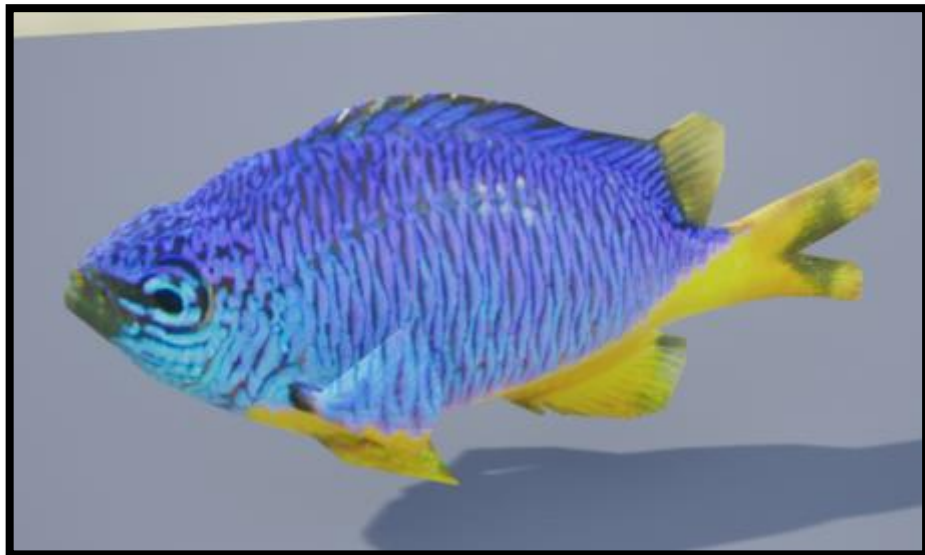
Діаграма класів

Розробив	Орлов О.К.			Методи та технології розробки навчальних програм із застосуванням віртуальної реальності	Аркуш	Аркушів
Перевірила	Калита Н.І.				3	3
Н. контр.	Калита Н.І.				ХНУРЕ Кафедра СТ	
Затвердив	Гребеннік І.В.					

## СХЕМИ ВІЗУАЛЬНОГО ПРОЕКТУВАННЯ

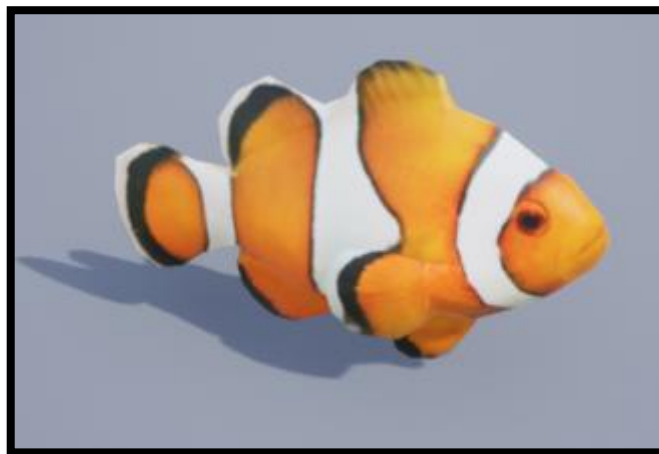


Модель риби №1



Модель риби №2

Розробив	Орлов О.К.			Методи та технології розробки навчальних програм із застосуванням віртуальної реальності	Аркуш	Аркушів
Перевірила	Калита Н.І.				1	3
Н. контр.	Калита Н.І.				ХНУРЕ Кафедра СТ	
Затвердив	Гребеннік І.В.					

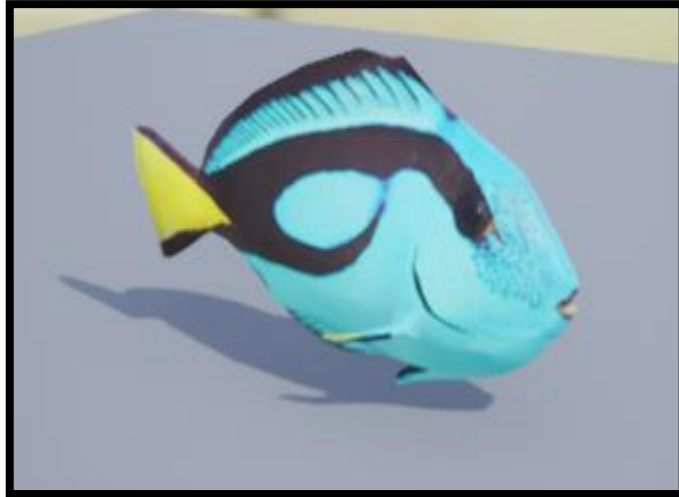


Модель риби №3

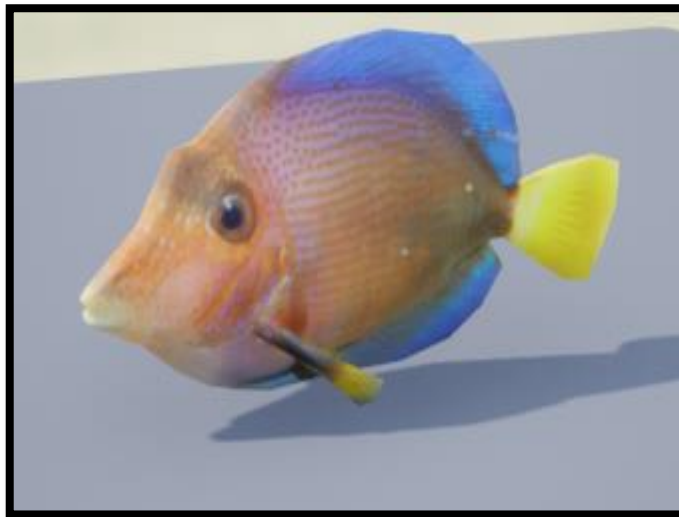


Модель риби №4

Розробив	Орлов О.К.			Методи та технології розробки навчальних програм із застосуванням віртуальної реальності	Аркуш	Аркушів
Перевірила	Калита Н.І.				2	3
Н. контр.	Калита Н.І.				ХНУРЕ Кафедра СТ	
Затвердив	Гребеннік І.В.					



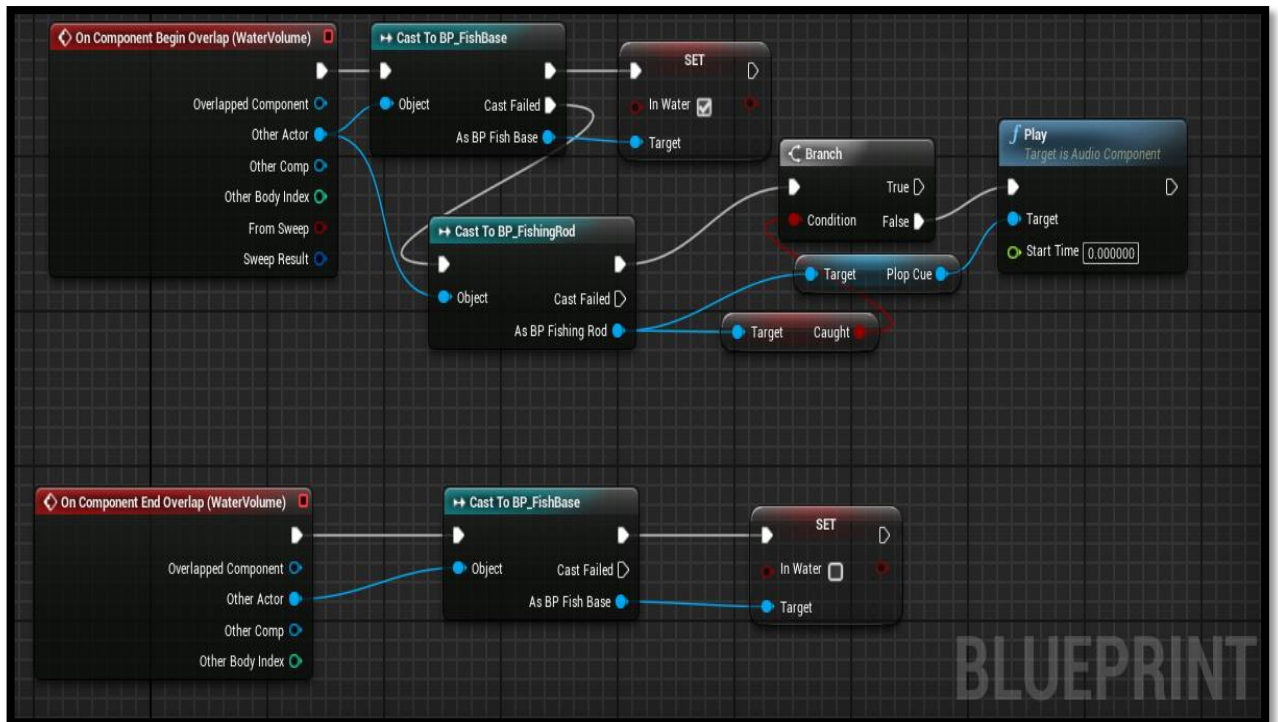
Модель риби №5



Модель риби №6

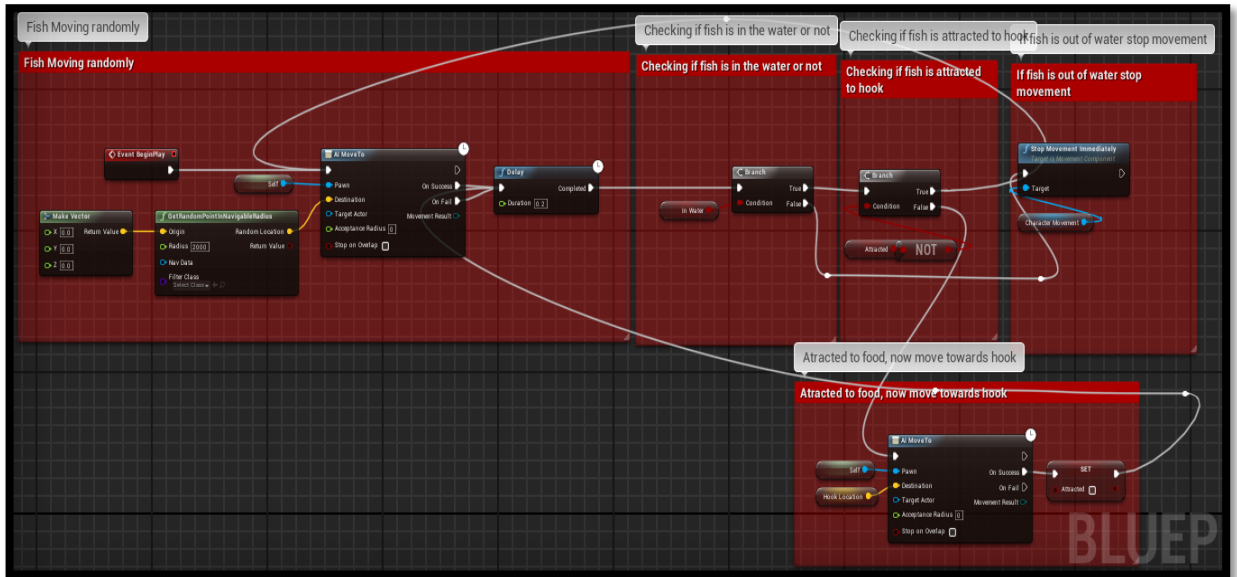
Розробив	Орлов О.К.			Методи та технології розробки навчальних програм із застосуванням віртуальної реальності	Аркуш	Аркушів
Перевірила	Калита Н.І.				3	3
Н. контр.	Калита Н.І.				ХНУРЕ Кафедра СТ	
Затвердив	Гребеннік І.В.					

## ВІЗУАЛЬНИЙ СКРИПТИНГ BLUEPRINT

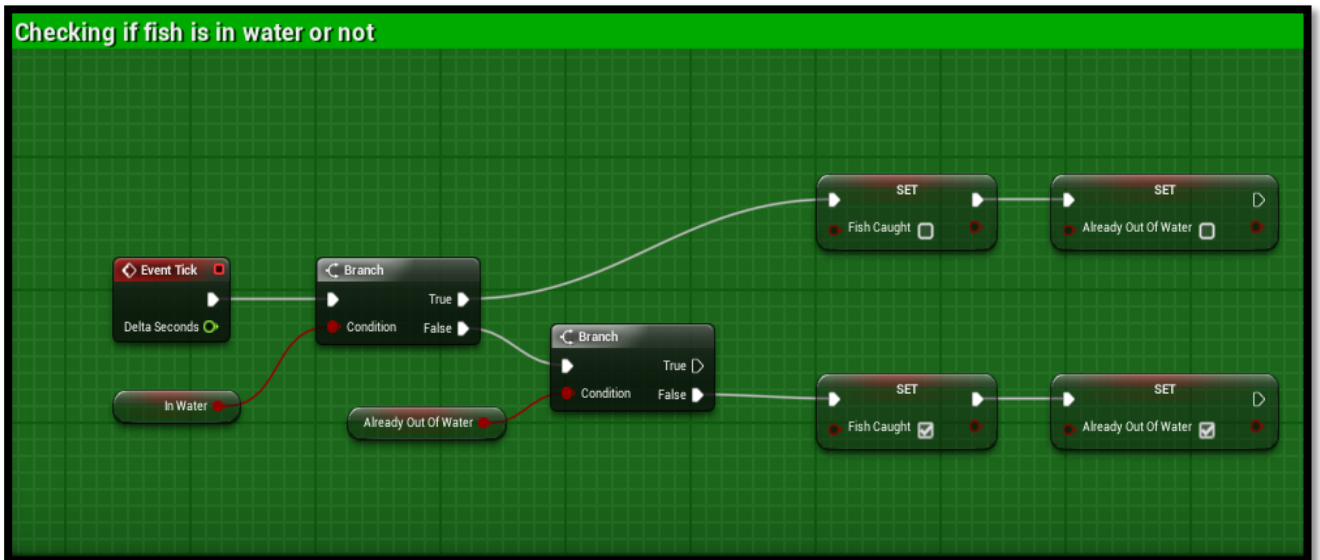


Взаємодія з водою

Розробив	Орлов О.К.		Методи та технології розробки навчальних програм із застосуванням віртуальної реальності	Аркуш	Аркушів
Перевірила	Калита Н.І.			1	4
Н. контр.	Калита Н.І.			ХНУРЕ Кафедра СТ	
Затвердив	Гребеннік І.В.				

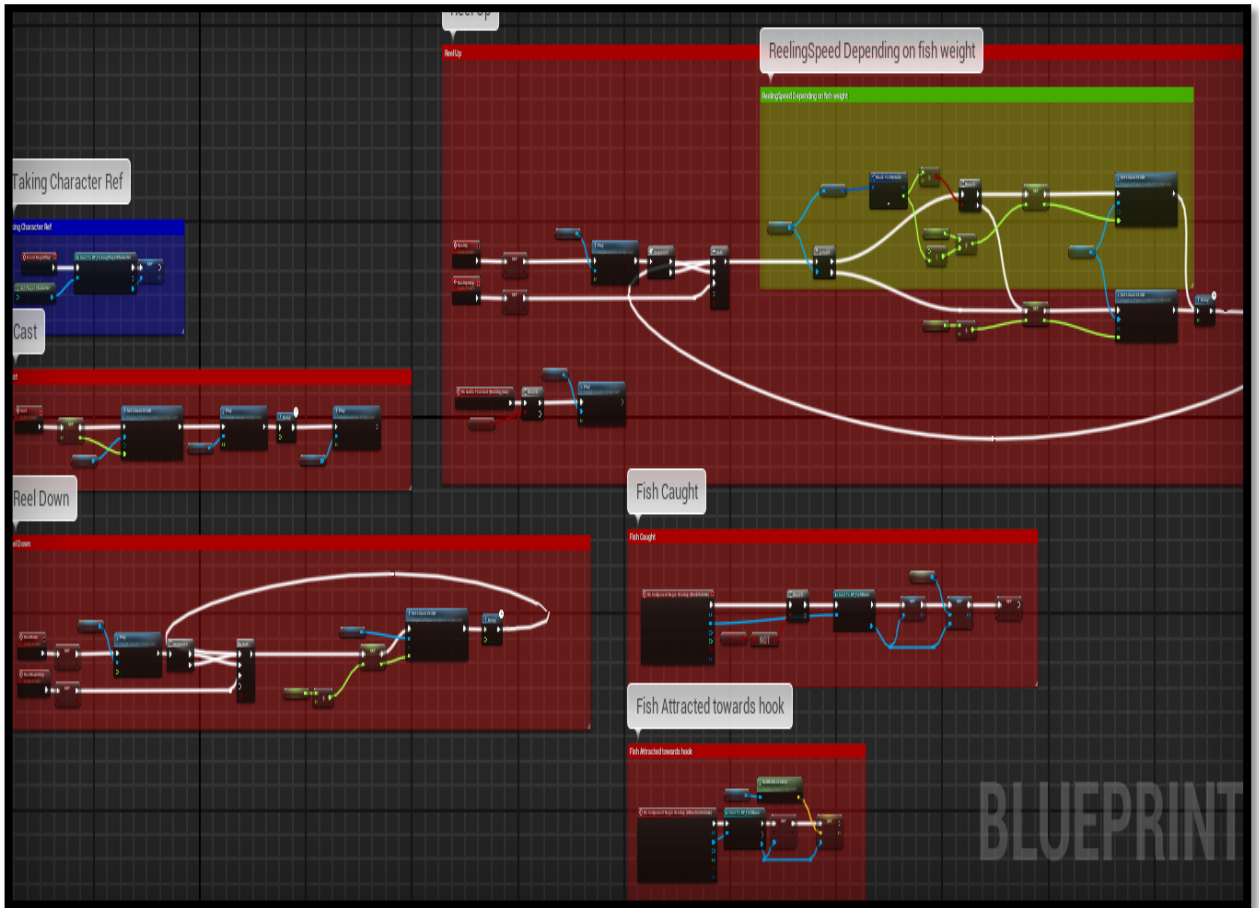


Штучний інтелект риби

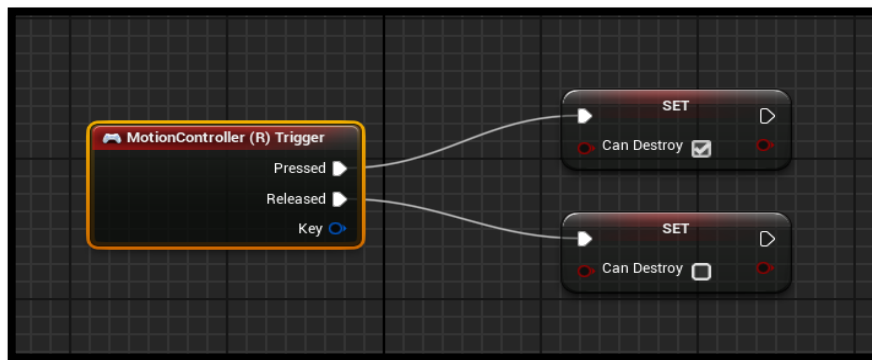


Перевірка риби «У воді чи ні»

Розробив	Орлов О.К.			Методи та технології розробки навчальних програм із застосуванням віртуальної реальності	Аркуш	Аркушів
Перевірила	Калита Н.І.				2	4
Н. контр.	Калита Н.І.				ХНУРЕ Кафедра СТ	
Затвердив	Гребеннік І.В.					

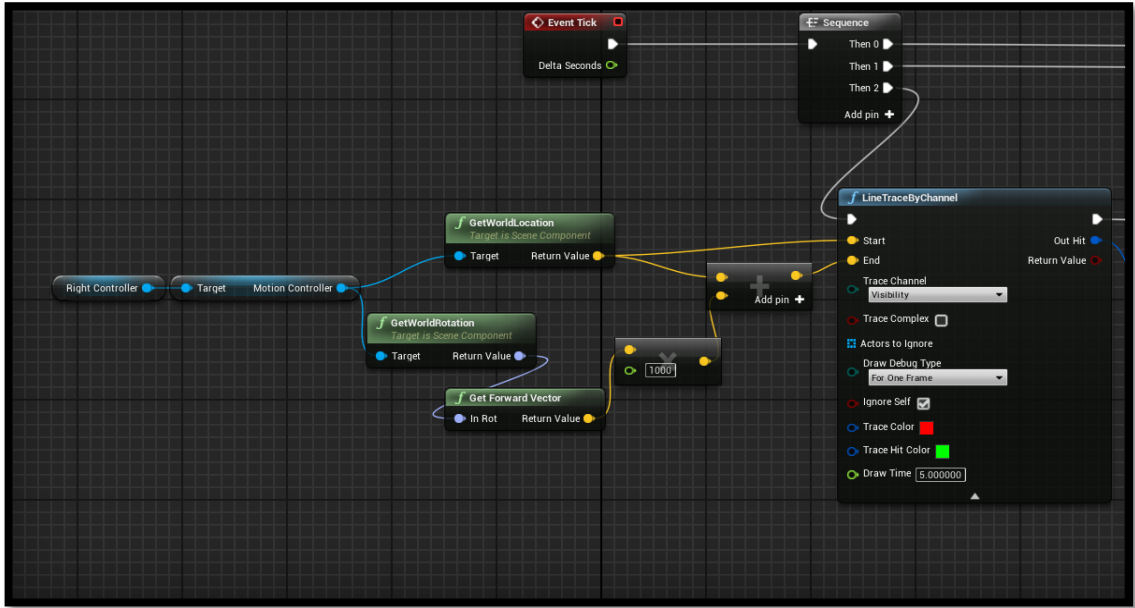


Логіка вудки

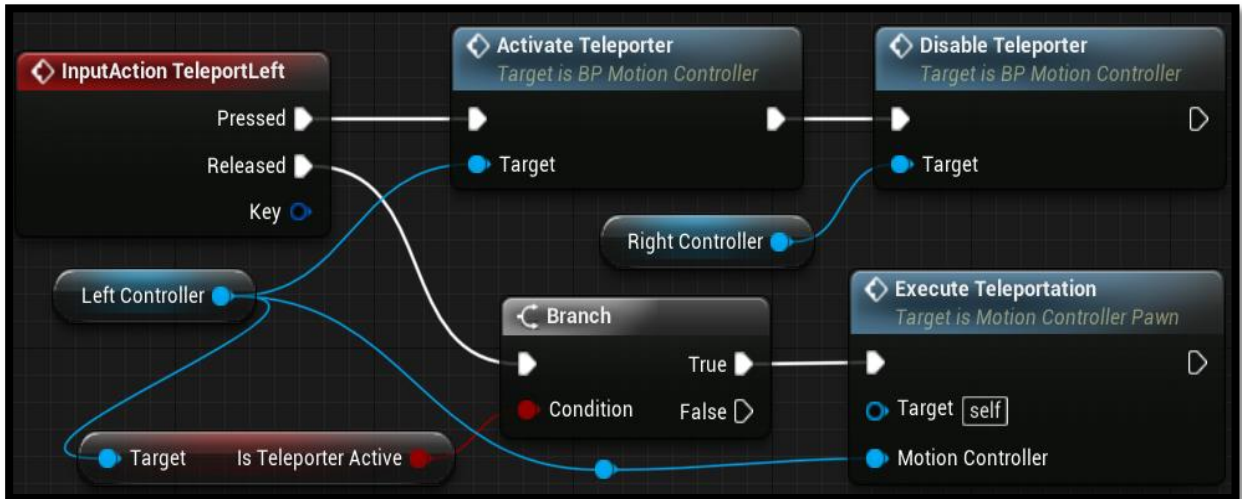


Реалізація однієї руки контролера

Розробив	Орлов О.К.		Методи та технології розробки навчальних програм із застосуванням віртуальної реальності	Аркуш	Аркушів
Перевірила	Калита Н.І.			3	4
Н. контр.	Калита Н.І.			ХНУРЕ Кафедра СТ	
Затвердив	Гребеннік І.В.				



Відрисовка лінії захвату цілі



Телепорт гравця

Розробив	Орлов О.К.			Методи та технології розробки навчальних програм із застосуванням віртуальної реальності	Аркуш	Аркушів
Перевірила	Калита Н.І.				4	4
Н. контр.	Калита Н.І.				ХНУРЕ Кафедра СТ	
Затвердив	Гребеннік І.В.					

## АНАЛІЗ ФАКТОРУ МОЖЛИВОСТЕЙ

№	Фактор	Зміст можливостей	Можлива реакція
1	Поява нових технологій за допомогою UE4	Розширення спектру можливостей для аналізу користувача і додавання методів забезпечення симуляції	Впровадження нових технологій, оптимізація і стабілізація роботи програми
2	Розширення можливостей віртуальної реальності	Додавання зручності і сегментів, де можна застосовувати нові ідеї	Розширення сегмента застосування
3	Збільшення кількості бізнес процесів	Новим гравцям потрібні такі додатки для моніторингу активності користувача	Створення та надання послуг
4	Поява нових технологій віртуальної реальності для симуляторів	Можливість впровадити ці технології в існуюче розширення або розробка нових рішень	Вивчення цих технологій і їх інтеграція з існуючими рішеннями

Розробив	Орлов О.К.			Методи та технології розробки навчальних програм із застосуванням віртуальної реальності	Аркуш	Аркушів
Перевірила	Калита Н.І.				1	1
Н. контр.	Калита Н.І.				ХНУРЕ Кафедра СТ	
Затвердив	Гребеннік І.В.					

## ПОРІВНЯННЯ ІГОР

	Fisherman	CrazyFishing	FancyFishingVR	AmigoFishing
Механіка	Основною цілю гри є ловля риби на час та зароблені як найбільше очок за одну ігрову сесію	Цілі тут немає. Гравець пересувається по локаціям та ловить що хоче. А трофеї складає у домі.	У грі подали нестандартний вид на рибалку, суміщено рибалку та тир. Гравець знаходиться під водою в якому має інвентар в якому може вибрати зброю. Ігровий процес складається з попаданням зброєю у рибу яка плаває навколо гравця. У грі присутні різні режими	Гравець знаходиться глибоко під водою у субмарині. Треба ловити рибу за допомогою пушок які зтягують рибу до підводної човен. Фактично, ця гра поєднує у собі рибалку та shooter. За пійману рибу отримує гроші для покращення зброї. Після ловлі певної кількості риби гравець отримує рівень і відкриваються нові пушки у магазині. Гра є нескінченною та після виходу зберігається ігровий процес.
Історія	Відсутня	Відсутня	Відсутня	Відсутня
Естетика	Упор на реалістичну графіку.	Проста та приємна мультяшна графіка.	Графіка у грі більш реалістична, ніж у першого аналогу, але досі примітивна	Графіка у грі більш реалістична, ніж у першого аналогу, але досі примітивна
Технологія	Зроблена для HTC Vive VR Headset	Зроблена для HTC Vive VR Headset	Зроблена для HTC Vive VR Headset	Зроблена для HTC Vive VR Headset, Vavle index та Oculus rift

Розробив	Орлов О.К.			Методи та технології розробки навчальних програм із застосуванням віртуальної реальності	Аркуш	Аркушів
Перевірила	Калита Н.І.				1	1
Н. контр.	Калита Н.І.				ХНУРЕ Кафедра СТ	
Затвердив	Гребеннік І.В.					

## СЕРЕДНІ ЗНАЧЕННЯ ОТРИМАНИХ РЕЗУЛЬТАТІВ

Рівень	Кількість балів	Кількість балів
1	385	–
2	355	–
3	350	–
4	325	270

Розробив	Орлов О.К.			Методи та технології розробки навчальних програм із застосуванням віртуальної реальності	Аркуш	Аркушів
Перевірила	Калита Н.І.				1	1
Н. контр.	Калита Н.І.				ХНУРЕ Кафедра СТ	
Затвердив	Гребеннік І.В.					

ДОДАТОК Б

ТЕКСТ ПРОГРАМИ

ГЮИК.502640.012 – 01 12 01  
(позначення документу)

ЗАТВЕРДЖЕНИЙ

ГЮИК.504200.004 – 01 12 01 – ЛУ(ПЗ)

Методи та технології розробки навчальних програм із застосуванням віртуальної  
реальності

Текст програми

ГЮИК.502640.012 – 01 12 01 – ЛУ(ПЗ)

АРКУШІВ 4

2020 р.

Харківський національний університет радіоелектроніки

«ЗАТВЕРДЖУЮ»  
керівник атестаційної роботи  
проф. Калита Н.І.

Методи та технології розробки навчальних програм із застосуванням віртуальної  
реальності

Текст програми

ЛИСТ ЗАТВЕРДЖЕННЯ

ГЮИК.502640.012 – 01 12 01 – ЛУ(ПЗ)

РОЗРОБИВ:  
ст.гр. СПРМ-18-2  
Орлов О.К.

2020 р

## Приклад коду управління персонажем

```

void
ABP_FishingPlayerController_C_pf3910431431::bpf_ExecuteUbergraph_BP_Fishi
ngPlayerController__pf_0(int32 bpp_EntryPoint__pf)
{
    APlayerController*
bpfv__CallFunc_GetPlayerController_ReturnValue2__pf{};
    check(bpp_EntryPoint__pf == 17);
    b01__Temp_struct_Variable__pf = b01__K2Node_InputKeyEvent_Key__pf;
    if(IsValid(bpv__W_Bucket__pf))
    {
        bpv__W_Bucket__pf->SetVisibility(ESlateVisibility::Visible);
    }
    bShowMouseCursor = true;
    bpfv__CallFunc_GetPlayerController_ReturnValue2__pf =
UGameplayStatics::GetPlayerController(this, 0);
    UWidgetBlueprintLibrary::SetInputMode_GameAndUIEx(bpfv__CallFunc_GetP
layerController_ReturnValue2__pf, bpv__W_Bucket__pf,
EMouseLockMode::DoNotLock, true);
    return; //KCST_EndOfThread
}
void
ABP_FishingPlayerController_C_pf3910431431::bpf_ExecuteUbergraph_BP_Fishi
ngPlayerController__pf_1(int32 bpp_EntryPoint__pf)
{
    APlayerController*
bpfv__CallFunc_GetPlayerController_ReturnValue2__pf{};
    check(bpp_EntryPoint__pf == 12);
    b01__Temp_struct_Variable__pf = b01__K2Node_InputKeyEvent_Key2__pf;
    if(IsValid(bpv__W_Bucket__pf))
    {
        bpv__W_Bucket__pf->SetVisibility(ESlateVisibility::Hidden);
    }
    bShowMouseCursor = false;
    bpfv__CallFunc_GetPlayerController_ReturnValue2__pf =
UGameplayStatics::GetPlayerController(this, 0);
    UWidgetBlueprintLibrary::SetInputMode_GameOnly(bpfv__CallFunc_GetPlay
erController_ReturnValue2__pf);
    return; //KCST_EndOfThread
}
void
ABP_FishingPlayerController_C_pf3910431431::bpf_ExecuteUbergraph_BP_Fishi
ngPlayerController__pf_2(int32 bpp_EntryPoint__pf)
{
    check(bpp_EntryPoint__pf == 69);
    b01__Temp_struct_Variable3__pf = b01__K2Node_InputKeyEvent_Key7__pf;
    // optimized KCST_UnconditionalGoto
    if(IsValid(bpv__CharacterRef__pf))
    {
        bpv__CharacterRef__pf->bpv__ReelUp__pf = false;
    }
    AActor* __Local__11 = ((AActor*) nullptr);
    b01__K2Node_DynamicCast_AsBP_Fishing_Rod__pf =
Cast<ABP_FishingRod_C_pf3910431431>(((IsValid(bpv__CharacterRef__pf) &&
IsValid(bpv__CharacterRef__pf->bpv__FishingRod__pf)) ?
((*(AccessPrivateProperty<AActor* >(bpv__CharacterRef__pf-

```

```

>bpv__FishingRod__pf), UChildActorComponent::__PPO__ChildActor() )))) :
(__Local__11));
    b01__K2Node_DynamicCast_bSuccess2__pf =
(b01__K2Node_DynamicCast_AsBP_Fishing_Rod__pf != nullptr);;
    if (!b01__K2Node_DynamicCast_bSuccess2__pf)
    {
        return; //KCST_EndOfThreadIfNot
    }
    if(IsValid(b01__K2Node_DynamicCast_AsBP_Fishing_Rod__pf))
    {
        b01__K2Node_DynamicCast_AsBP_Fishing_Rod__pf-
>bpf__ReelUpStop__pf();
    }
    return; //KCST_EndOfThread
}
void
ABP_FishingPlayerController_C__pf3910431431::bpf__ExecuteUbergraph_BP_Fishi
ngPlayerController__pf_3(int32 bpp__EntryPoint__pf)
{
    check(bpp__EntryPoint__pf == 67);
    b01__Temp_struct_Variable3__pf = b01__K2Node_InputKeyEvent_Key6__pf;
    // optimized KCST_UnconditionalGoto
    if(IsValid(bpv__CharacterRef__pf))
    {
        bpv__CharacterRef__pf->bpv__ReelUp__pf = true;
    }
    AActor* __Local__12 = ((AActor*)nullptr);
    b01__K2Node_DynamicCast_AsBP_Fishing_Rod4__pf =
Cast<ABP_FishingRod_C__pf3910431431>(((IsValid(bpv__CharacterRef__pf) &&
IsValid(bpv__CharacterRef__pf->bpv__FishingRod__pf)) ?
((* (AccessPrivateProperty<AActor* >(bpv__CharacterRef__pf-
>bpv__FishingRod__pf), UChildActorComponent::__PPO__ChildActor() )))) :
(__Local__12));
    b01__K2Node_DynamicCast_bSuccess5__pf =
(b01__K2Node_DynamicCast_AsBP_Fishing_Rod4__pf != nullptr);;
    if (!b01__K2Node_DynamicCast_bSuccess5__pf)
    {
        return; //KCST_EndOfThreadIfNot
    }
    if(IsValid(b01__K2Node_DynamicCast_AsBP_Fishing_Rod4__pf))
    {
        b01__K2Node_DynamicCast_AsBP_Fishing_Rod4__pf-
>bpf__ReelUp__pf();
    }
    return; //KCST_EndOfThread
}

```

## Приклад коду реалізації AI риби

```

void
ABP_FishBase_C__pf3910431431::__CustomDynamicClassInitialization(UDynamicCl
ass* InDynamicClass)
{
    ensure(0 == InDynamicClass->ReferencedConvertedFields.Num());
    ensure(0 == InDynamicClass->MiscConvertedSubobjects.Num());
    ensure(0 == InDynamicClass->DynamicBindingObjects.Num());
    ensure(0 == InDynamicClass->ComponentTemplates.Num());
}

```

```

ensure(0 == InDynamicClass->Timelines.Num());
ensure(nullptr == InDynamicClass->AnimClassImplementation);
InDynamicClass->AssembleReferenceTokenStream();
// List of all referenced converted classes
InDynamicClass-
>ReferencedConvertedFields.Add(ABP_FishingPlayerCharacter_C__pf3910431431::
StaticClass());
    InDynamicClass-
>ReferencedConvertedFields.Add(ABP_FishingRod_C__pf3910431431::StaticClass(
));
    // List of all referenced converted structures
extern UScriptStruct*
Z_Construct_UScriptStruct_FFishDetails__pf3910431431();
    InDynamicClass-
>ReferencedConvertedFields.Add(Z_Construct_UScriptStruct_FFishDetails__pf39
10431431());
    FConvertedBlueprintsDependencies::FillUsedAssetsInDynamicClass(InDyna
micClass, &__StaticDependencies_DirectlyUsedAssets);
    auto __Local__11 = NewObject<USceneComponent>(InDynamicClass,
USceneComponent::StaticClass(), TEXT("DefaultSceneRoot_GEN_VARIABLE"));
    InDynamicClass->ComponentTemplates.Add(__Local__11);
    auto __Local__12 =
NewObject<UComponentDelegateBinding>(InDynamicClass,
UComponentDelegateBinding::StaticClass(),
TEXT("ComponentDelegateBinding_7"));
    InDynamicClass->DynamicBindingObjects.Add(__Local__12);
    __Local__12->ComponentDelegateBindings =
TArray<FBlueprintComponentDelegateBinding>();
    __Local__12->ComponentDelegateBindings.AddUninitialized(1);
    FBlueprintComponentDelegateBinding::StaticStruct()-
>InitializeStruct(__Local__12->ComponentDelegateBindings.GetData(), 1);
    auto& __Local__13 = __Local__12->ComponentDelegateBindings[0];
    __Local__13.ComponentPropertyName = FName(TEXT("PawnSensing"));
    __Local__13.DelegatePropertyName = FName(TEXT("OnSeePawn"));
    __Local__13.FunctionNameToBind =
FName(TEXT("BndEvt__PawnSensing_K2Node_ComponentBoundEvent_0_SeePawnDelegat
e_DelegateSignature"));
}
void
ABP_FishBase_C__pf3910431431::bpf__ExecuteUbergraph_BP_FishBase__pf_0(int32
bpp__EntryPoint__pf)
{
    int32 __CurrentState = bpp__EntryPoint__pf;
    do
    {
        switch( __CurrentState )
        {
            case 1:
            {
                bpv__AlreadyOutOfWater__pf = false;
                __CurrentState = -1;
                break;
            }
            case 34:
            {
            }
            case 35:
            {

```

```
        if (!bpv__InWater__pf)
        {
            __CurrentState = 37;
            break;
        }
    }
    case 36:
    {
        bpv__FishCaught__pf = false;
        __CurrentState = 1;
        break;
    }
    case 37:
    {
        if (!bpv__AlreadyOutOfWater__pf)
        {
            __CurrentState = 38;
            break;
        }
        __CurrentState = -1;
        break;
    }
    case 38:
    {
        bpv__FishCaught__pf = true;
    }
    case 39:
    {
        bpv__AlreadyOutOfWater__pf = true;
        __CurrentState = -1;
        break;
    }
    default:
        break;
    }
} while( __CurrentState != -1 );
}
}
```

