

0) การเตรียม Project

0.1. ดาวน์โหลด GitHub Repository (ใช้ GitHub Desktop ได้) ซึ่งจะมีโครงสร้างดังรูป

<https://github.com/Game-Design-Dev-IT-KMITL/2020-GameDevWithUnreal-6.git>

0.2. สร้าง Branch

- สร้าง Branch (ใน GitHub Desktop) โดยใช้ชื่อ ก្នុំនិងរាជសារកីឡា ! เช่น _Sec1_61xxxxxxxx
 - สร้างโฟลเดอร์ (ใน Unreal Engine) ใน Content สำหรับทำงาน โดยใช้ชื่อดีบกัน Branch และ
สร้าง sub-folders จำนวน 2 folders: Enemies และ Maps
 - งานทั้งหมดจะต้องทำอยู่ในโฟลเดอร์ที่สร้างขึ้นใหม่เท่านั้น

0.3. การส่งงาน

- Commit และ Push ไปที่ Branch ที่นักศึกษาสร้างขึ้นเพ่านั้น

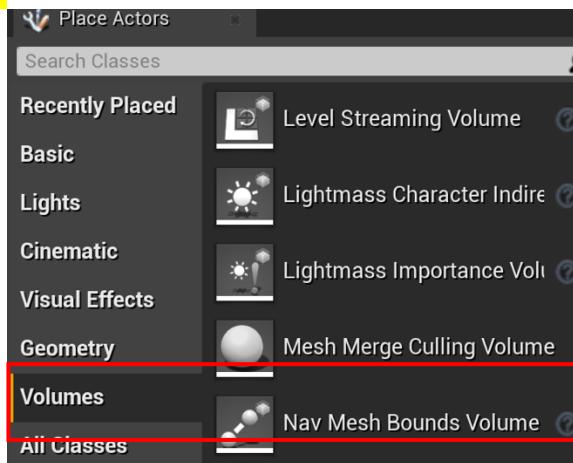
<<< ห้ามส่งเข้า Main Branch >>>

1) Smart Enemies with AI

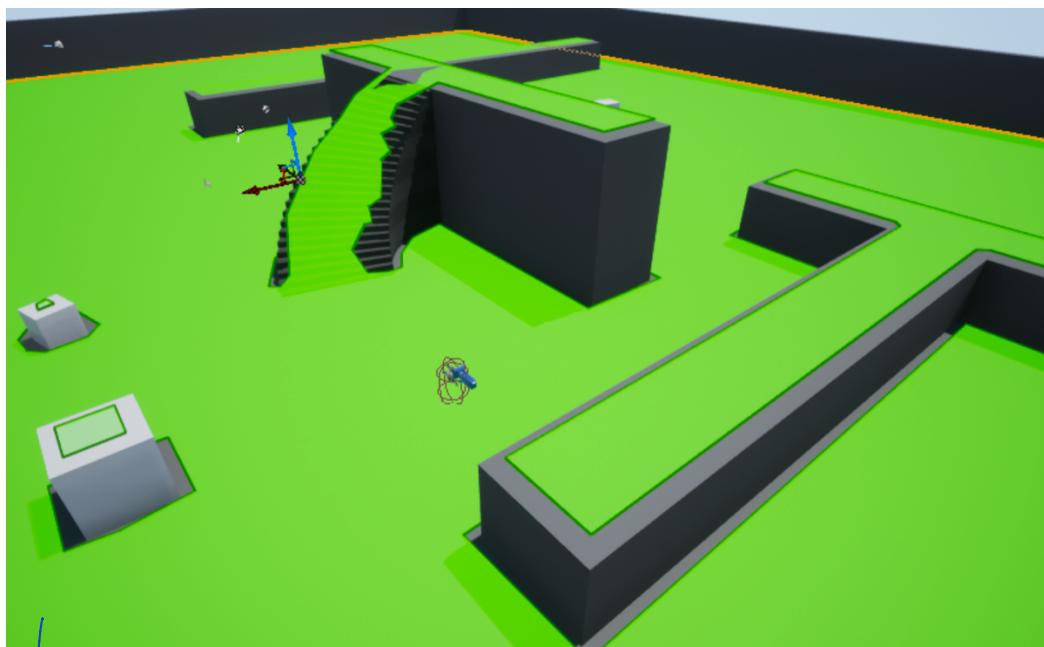
- 1.1. Copy `FirstPersonExampleMap` ใน `FirstPersonBP/Maps` แล้วแก้ไขชื่อเป็น `Map01_Navigation` ใน `Maps` folder

- 1.2. เพิ่ม NavMes ลงใน Level เพื่อกำหนดพื้นที่ให้กับ AI

- เลือก `NavMesh` และวางใน Level (จาก `Volume\Nav Mesh Bounds Volume`)



- ทำการขยายขนาดของ NavMesh จนกว่าจะครอบคลุมพื้นที่ ๆ สามารถเดินได้ทั้งหมด
หมายเหตุ เมื่อขนาดของ Volume ใหญ่ขึ้น จะเห็นพื้นที่มีเสียง ซึ่งแสดงถึงพื้นที่ ๆ Agent สามารถเดิน
ได้ หากต้องการซ่อน/แสดง พื้นที่เสียงทำได้โดยการกดปุ่ม P



เห็นไหมว่าส่วนไหนบ้างที่ทางสร้างได้ไม่ถูก ใจ Nav Mesh ก็จะมีปัญหา

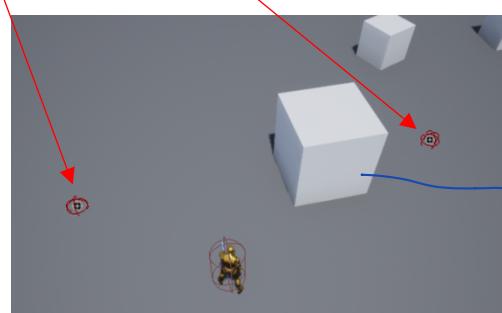
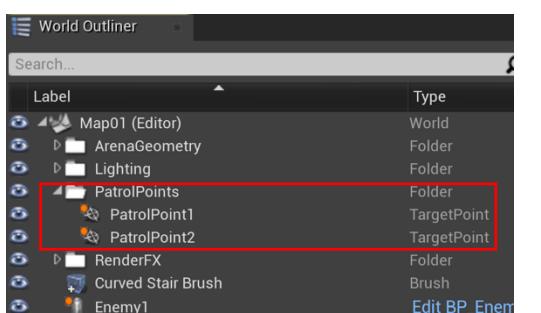
1.3. สร้าง Enemies AI

- สร้าง Blueprint จาก Parent Class เป็น Ue4ASP_Character และ save ใน Enemy folder โดยตั้งชื่อว่า BP_EnemyCharacter
 - เปิด Blueprint ทำการเปลี่ยนสี Character (เลือก Mesh (inherited) ใน Components) เป็นสีอื่น (เปลี่ยนใน Element_0)
- สร้าง Blueprint จาก Parent Class เป็น AI_Controller และ save ใน Enemy folder โดยตั้งชื่อว่า BP_EnemyController
- สร้าง AI file เลือก category เป็น Behavior Tree และ save ใน Enemy folder โดยตั้งชื่อว่า AI_EnemyBehavior
- สร้าง AI file เลือก category เป็น Blackboard และ save ใน Enemy folder โดยตั้งชื่อว่า AI_EnemyBlackboard
- เปิด BP_EnemyCharacter และทำการเปลี่ยน Property ดังนี้
 - เลือก BP_EnemyCharacter (self) จากหน้าต่าง Components
 - กำหนดค่า Pawn ในหน้า Detail เป็น BP_EnemyController
 - เลือก CapsuleComponent (Inherited) ในหน้าต่าง Components
 - กำหนดค่า Collision ในหน้า Detail เป็น BlockAllDynamic
 - ใน Rendering กด check ที่ Hidden in Game
 - เลือก Mesh (Inherited) ในหน้าต่าง Components
 - เปลี่ยน Material 0 เป็น M_Metal_Gold
- เพิ่ม BP_EnemyCharacter ลงใน Level และตั้งชื่อใหม่ว่า Enemy1 (ใน WorldOutliner)
- Play ตัว Enemy1 ซึ่งไม่มีความสามารถอะไร ณ ปัจจุบัน

1.4. สร้าง Enemies Navigation Behavior

1.4.1. กำหนดจุดลาดตระเวนของ Enemies AI

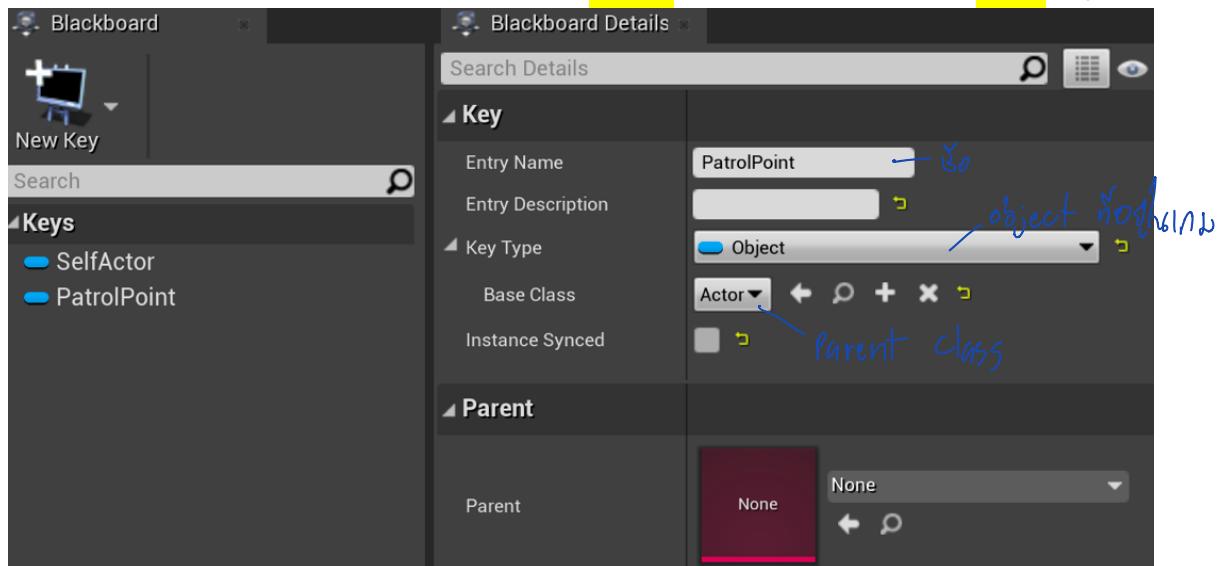
- เพิ่ม Target Point object จำนวน 2 ตัว (จาก All Classes) ลงใน Level
 - เปลี่ยนชื่อใหม่เป็น PatrolPoint1 และ PatrolPoint2
 - กำหนดตำแหน่งให้ห่างกันพอสมควร



เก็บป้อม
ลุยกองทราย

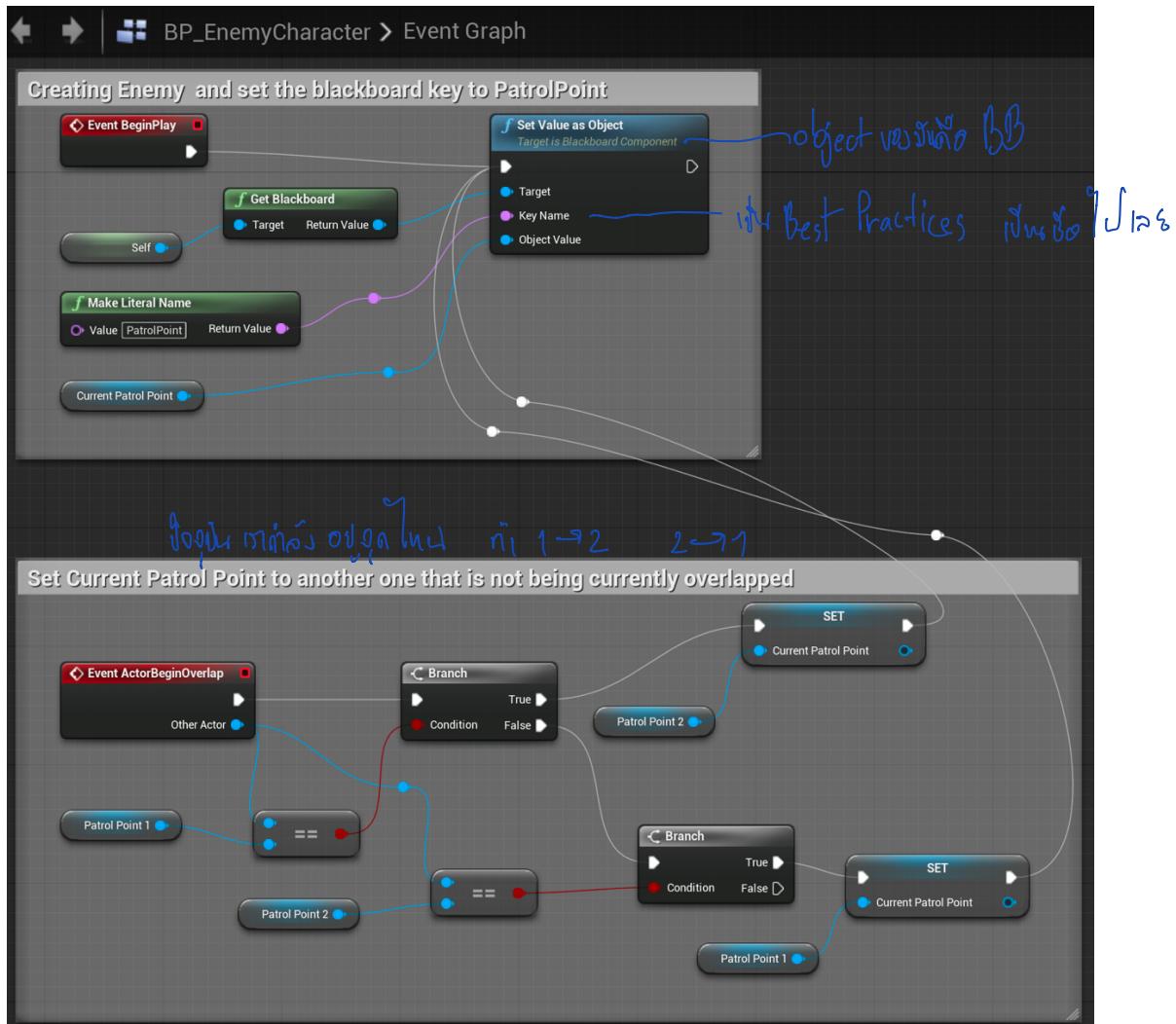
1.4.2. สร้างการสื่อสารระหว่าง Assets (PatrolPoint)

- สร้างการสื่อสารระหว่าง Enemy และ Patrol Points ที่ได้สร้างขึ้น
 - เปิด AI_EnemyBlackboard และทำการสร้าง New Key ชื่อ PatrolPoint และกำหนดค่า Key Type เป็น Object และเลือก Base Class เป็น Actor ดังรูป



- สร้าง Script ในการจัดการ ๆ กำหนดค่าของจุดการลาดตระเวน
 - เปิด BP_EnemyCharacter สร้างตัวแปร ดังนี้
 - PatrolPoint1 ประเภท Actor
 - PatrolPoint2 ประเภท Actor
 - CurrentPatrolPoint ประเภท Actor

- สร้าง Script ดังนี้



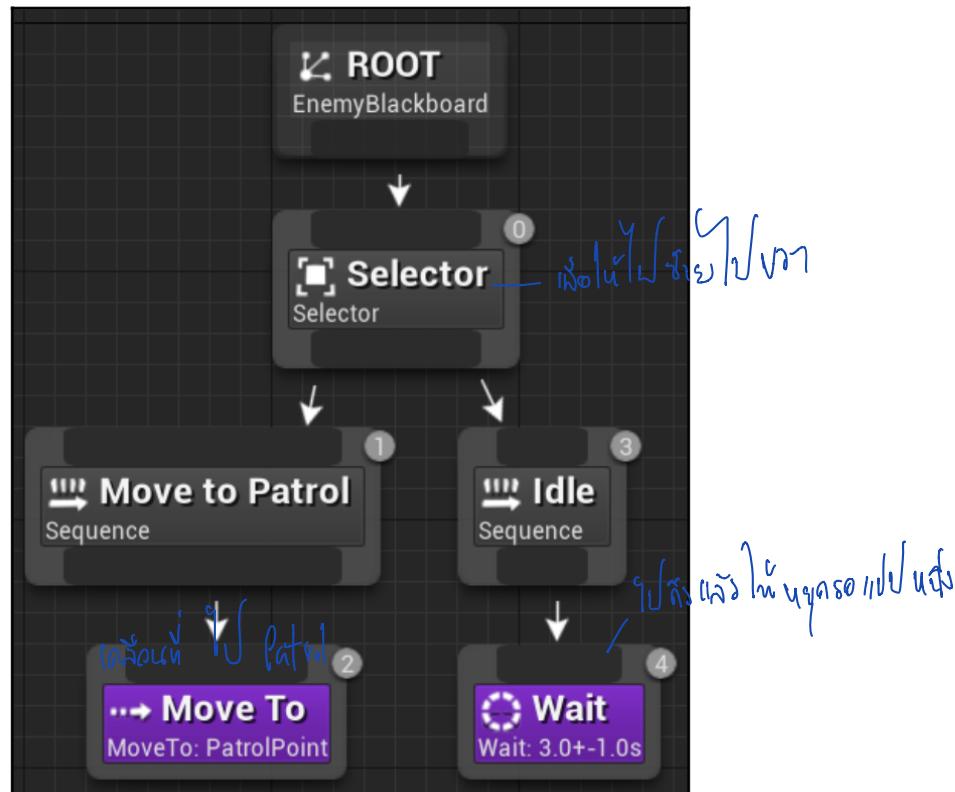
- สร้าง AI ให้สั่งงาน Behavior Tree เพื่อกำหนด Behavior ของ Enemy (ลาดตระเวน)

- เปิด BP_EnemyController และสร้าง Script ดังนี้



1.4.3. การสอน Enemy AI ให้มีความสามารถในการเดินด้วย Behavior Tree

- เปิด AI_EnemyBehavior และสร้างเงื่อนไข ดังรูป
 - Move to ใช้ค่า Default และเลือก Blackboard Key เป็น PatrolPoint
 - Idle ตั้งค่า Wait Time = 3.0 และ Random Deviation = 1.0



1.4.4. กำหนดค่าตัวแปร Patrol Points ทั้ง 3 ตัวของ Enemy1 ในหน้า Level Editor ดังรูป

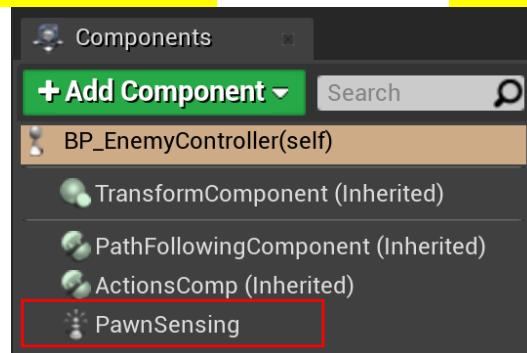


- Play ตัว Enemy จะทำการคาดคะเนระหว่างจุดที่สร้างขึ้น

1.4.5. Enemy AI chase the player

- เพิ่มความสามารถให้ Enemy มองเห็น Player ผ่าน Pawn Sensing feature

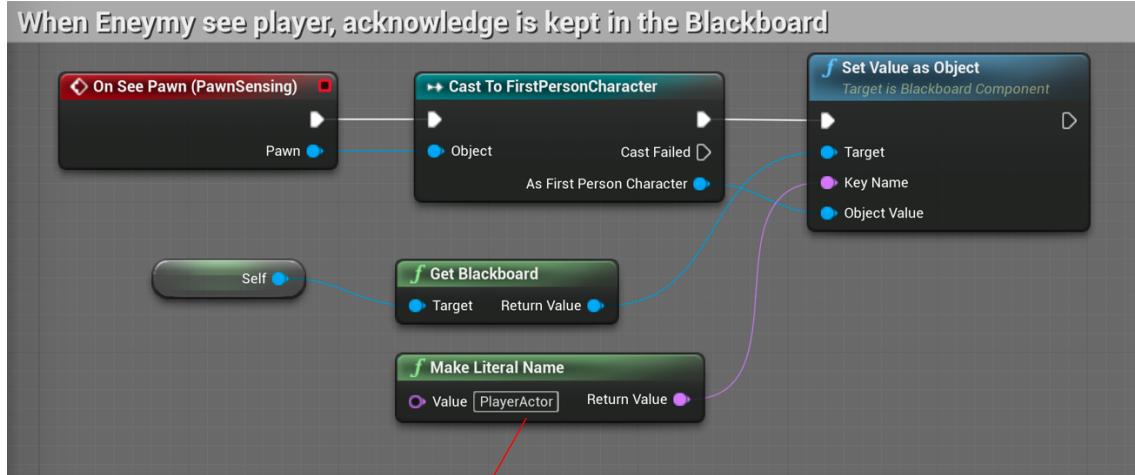
- เปิด BP_EnemyController Blueprint และเพิ่ม PawnSensing ใน Components



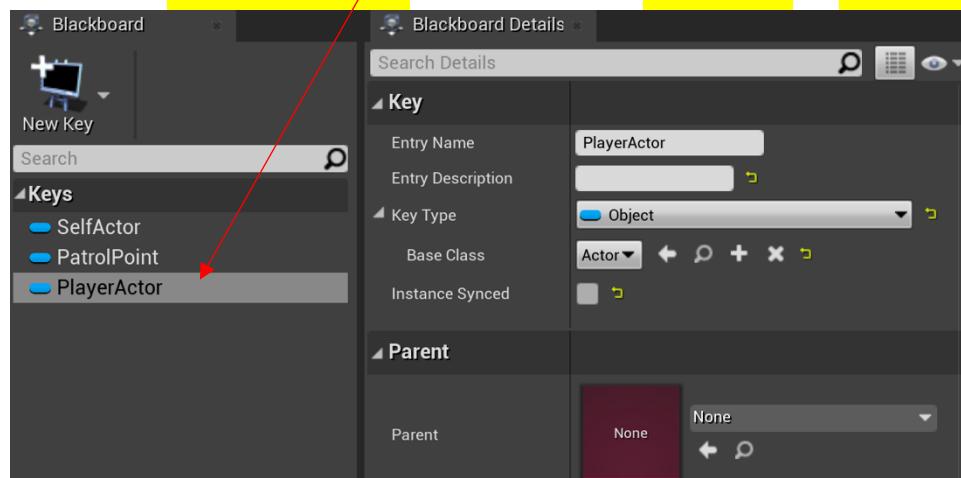
- Click เลือก PawnSensing และเพิ่ม On See Pawn ใน Graph แล้วสร้าง Script ดังรูป

On See Pawn จะเพิ่มความสามารถให้ Enemy มองเห็น Player ในระดับสายตา (Line of sight)

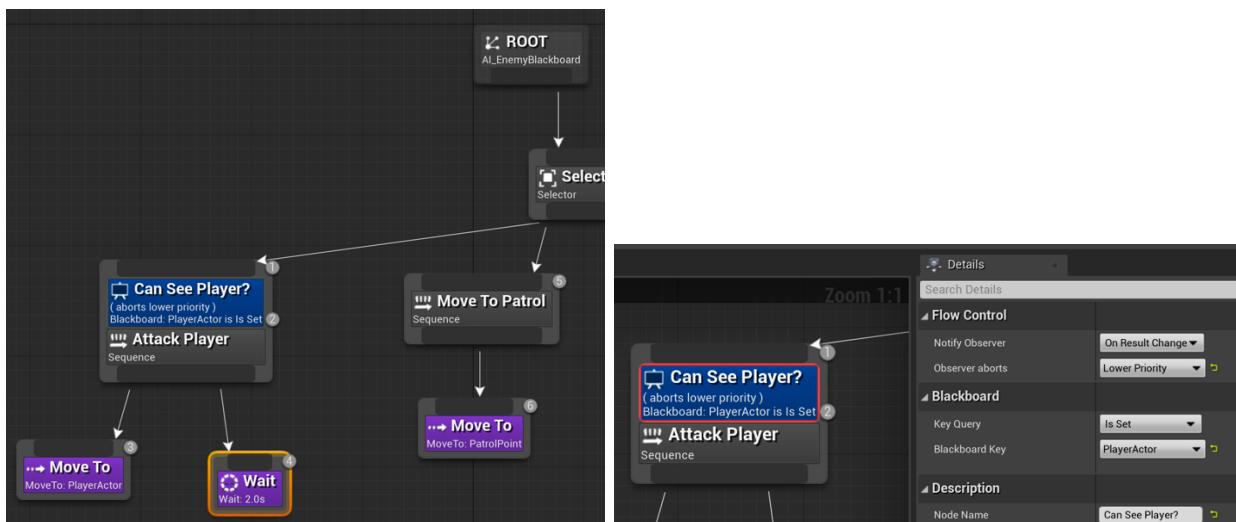
When Enemy see player, acknowledge is kept in the Blackboard



- เปิด AI_EnemyBehavior Blueprint และเพิ่ม PlayerActor ใน Blackboard ดังรูป

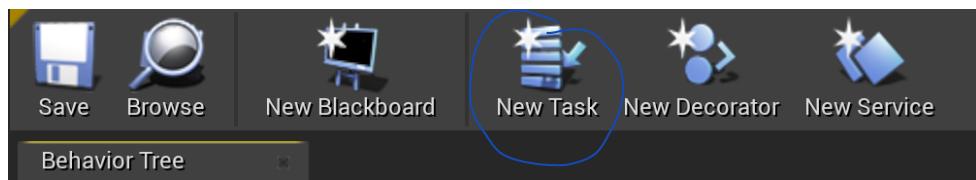


- เพิ่ม Sequence ใน Behavior Tree เพื่อรับ PlayerActor ดังนี้
 - เปลี่ยนชื่อ Sequence เป็น **Attack Player**
 - เพิ่ม Decorator (right-click บน Attack Player) เลือกเป็น Blackboard ตั้งชื่อว่า Can See Player? และกำหนดค่าดังรูป
 - เพิ่ม 2 nodes จาก Attack Player
 - Move To เลือก PlayerActor
 - Wait กำหนดค่าเป็น 2s
- Play ตัว Enemy จะเคลื่อนไปลาดตระเวนระหว่าง 2 จุดที่กำหนดไว้ เมื่อเจอ Player (ในระดับสายตา) จะทำการติดตาม Player จนเจอ แล้วหยุดรอ 2 วินาที แล้วตาม Player ต่อไปเรื่อยๆ จนกว่าจะเจอ แล้วหยุดรอ 2 วินาที สลับกันไป



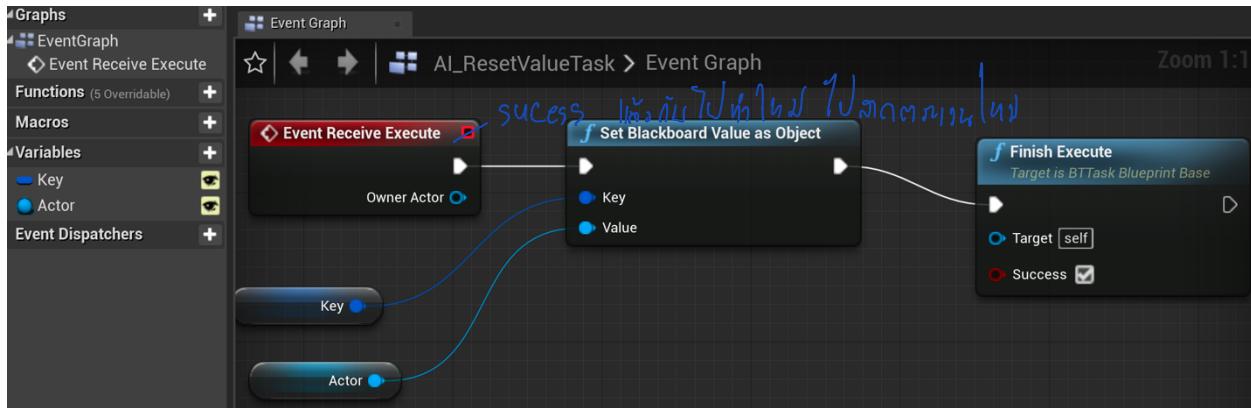
- ใส่เพิ่มความสามารถให้ Enemy หยุดติดตาม Player กลับไปลาดตระเวน

- ใน AI_Enemybehavior กด New Task

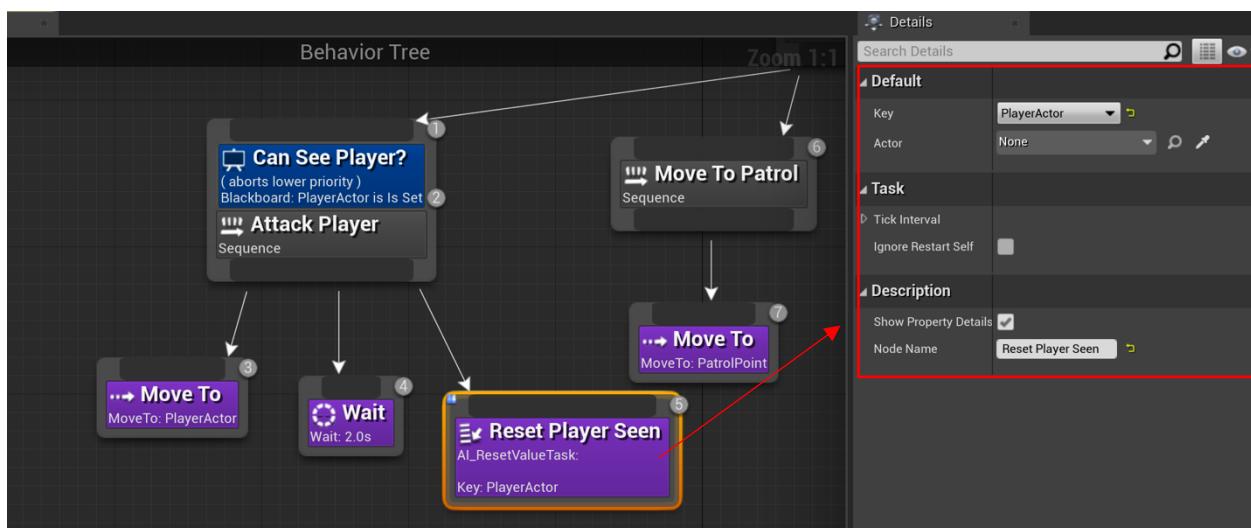


- จะมีไฟล์ใหม่ชื่อ BTTask_BlueprintBase_New ใน Enemy folder
 - เปลี่ยนชื่อ File เป็น **AI_ResetValueTask**

- เปิด AI_ResetValueTask และสร้างตัวแปรใหม่ ดังนี้
 - Key ประเภท BlackboardKeySelector ทำการ check Instance Editable
 - Actor ประเภท Object ทำการ check Instance Editable
 - สร้าง Script ดังรูป



- เปิด AI_EnemyBehavior และแก้ไข Script ดังนี้
 - เพิ่ม AI_ResetValueTask ที่เพิ่งสร้างขึ้นมา เพื่อให้ Enemy หยุดการตาม Player เมื่อเจอแล้วกลับไปทำการลาดตระเวนใหม่



- Play เมื่อ Enemy ลาดตระเวนแล้วเจอ Player จะทำการตาม Player จนเจอแล้วหยุด 2 วินาที แล้วกลับไปลาดตระเวนใหม่ และจะเริ่มตาม Player เมื่อเจออีกรอบหนึ่ง วนซ้ำไปเรื่อยๆ