

0) การเตรียม Project

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

<https://github.com/Game-Design-Dev-IT-KMITL/2020-GameDevWithUnreal-5.git>

0.2. สร้าง Branch

- สร้าง Branch (ใน GitHub Desktop) โดยใช้ชื่อ กลุ่มและรหัสนักศึกษา เช่น _Sec1_61xxxxxx
 - สร้างโฟลเดอร์ (ใน Unreal Engine) ใน Content สำหรับทำงาน โดยใช้ชื่อดีกว่ากับ Branch และ
สร้าง sub-folders จำนวน 4 folders: Blueprints, Materials, Maps และ UI
 - งานทั้งหมดจะต้องทำอยู่ในโฟลเดอร์ที่สร้างขึ้นใหม่เท่านั้น

0.3. การส่งงาน

- Commit และ Push ไปที่ Branch ที่นักศึกษาสร้างขึ้นเพ่านั้น

<<< ห้ามส่งเข้า Main Branch >>>

ชิ้นปลั๊กปั้งน้ำ

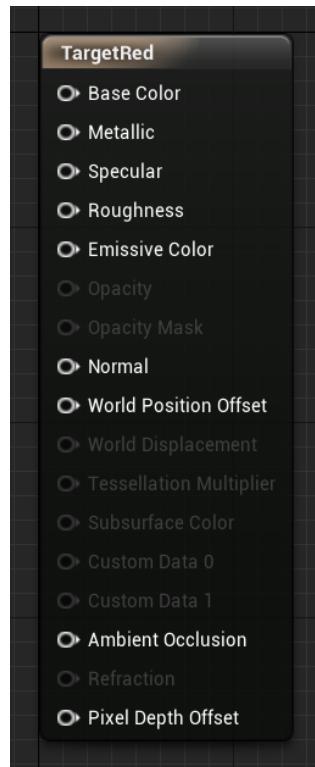
1) Object Interaction

1.1. Copy Level “FirstPersonExampleMap” จาก “FirstPersonBP/Maps” มาใน Maps folder

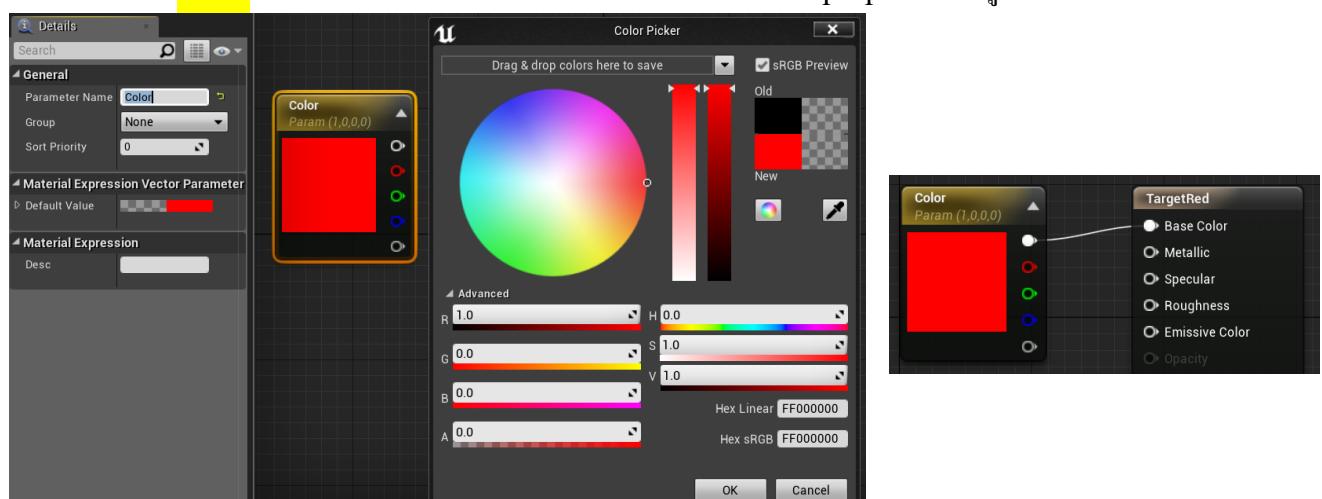
ของตัวเอง แล้วตั้งชื่อว่า “Map01_Mat”

1.2. สร้าง Material ใหม่ “TargetRed” ใน Materials folder และทำการเปิด Material Editor

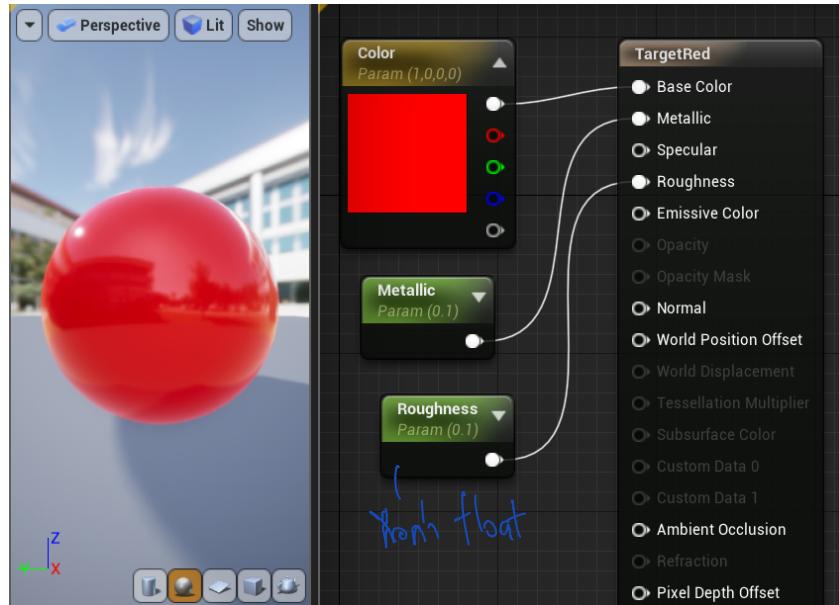
โดยการ double-click ที่ material ที่ได้สร้างขึ้น จะได้ดังรูป



- กำหนดค่า โดยการสร้าง (click ขวา) Vector Parameter node เพื่อกำหนดค่าสี ทำการตั้งชื่อว่า “Color” และกำหนดค่าสีเป็นสีแดง แล้วเชื่อมต่อ กับ material properties ดังรูป

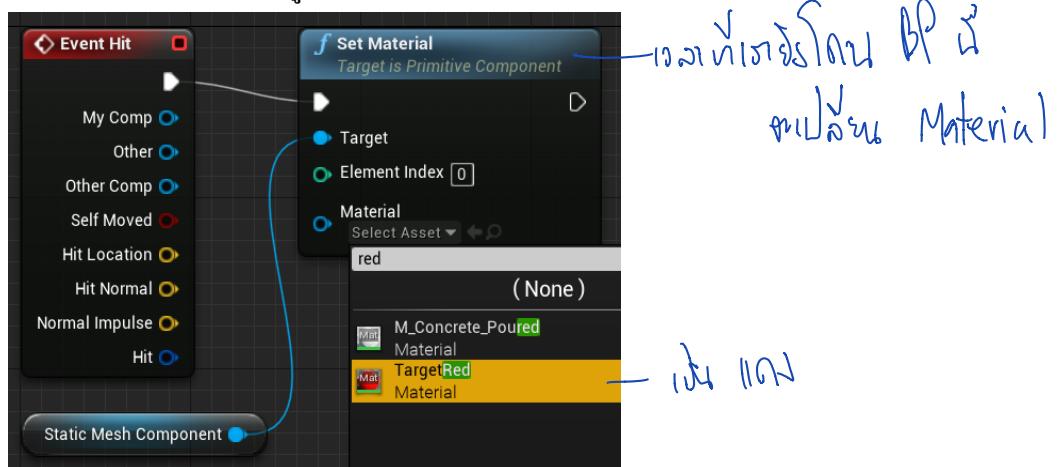


- สร้าง Substance ให้กับ Material โดยการสร้างเพิ่ม Scalar Parameter node จำนวน 2 ตัว ดังนี้
 - Metallic กำหนดค่าเป็น **0.1** แล้วเชื่อมต่อกับ Metallic node
 - Roughness กำหนดค่าเป็น 0.1 แล้วเชื่อมต่อกับ Roughness node
 - ผลลัพธ์จะเป็นดังรูป



1.3. สร้าง Blueprint ใหม่ สำหรับการตรวจสอบการยิง

- เพิ่ม Cylinder object ลงใน Level ตั้งชื่อว่า “**CylinderTarget**” และเลือกค่าเป็น **Movable** ในช่อง Transform (หน้าต่าง Detail)
- สร้าง Blueprint โดยการ Click เลือก Cylinder ใน World Outliner แล้วกดปุ่ม **Blueprint/Add Script** ในหน้า Detail แล้วทำการเลือก Blueprint folder ในการ save (สร้าง Blueprint อัตโนมัติให้กับ object ที่เลือก) ซึ่ง Engine จะตั้งชื่อให้ยัตติโภน์มีตัวว่า “**CylinderTarget_Blueprint**”
- เปิด Blueprint เพื่อตรวจสอบการยิง ดังรูป



- Play เมื่อทำการยิงโดน CylinderTarget ระบบจะทำการเปลี่ยนสี (Material) ให้เป็นสีแดงตามที่กำหนด



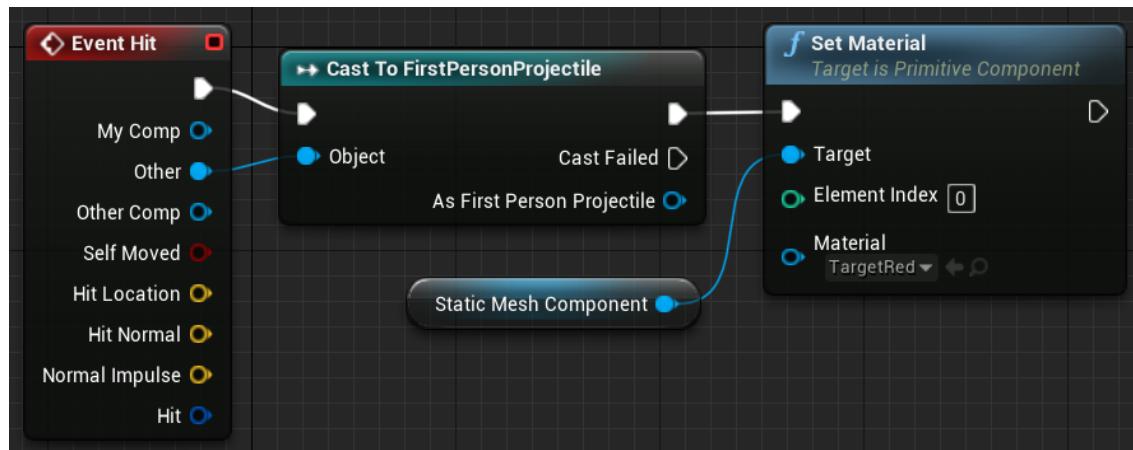
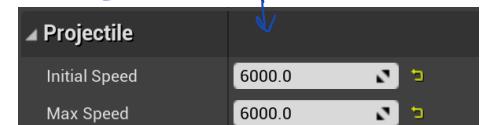
→ ก็ Projectfile

1.4. Modify Blueprint เพื่อให้สามารถตรวจสอบประเภท Actor ได้

ในตอนนี้ CylinderTarget จะเปลี่ยนสี ไม่ว่าจะเกิดจากการโคนชน หรือ โคนยิ่ง ดังนั้นควบคุมให้เกิดการเปลี่ยนแปลงเมื่อโคนยิงเท่านั้น สามารถทำได้โดยการแก้ไขเงื่อนไขใน Event Hit ดังรูป

- FirstPersonProjectile คือ Blueprint ที่ควบคุมลูกกระสุน หากต้องการเพิ่มความเร็วกระสุน ทำได้โดยการเปิด Blueprint และแก้ไขค่า Projectile object เช่น

→ Projectile BP

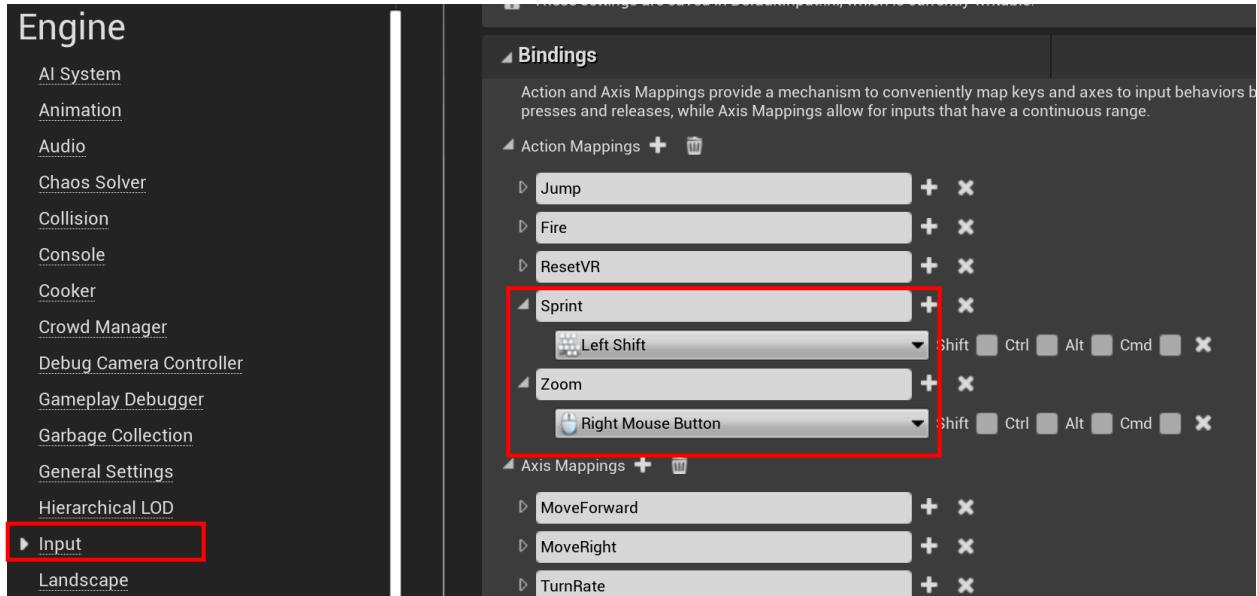


- Play โดย CylinderTarget จะเปลี่ยนสีเมื่อโคนยิงเท่านั้น

1.5. การเพิ่มความสามารถให้ผู้เล่น

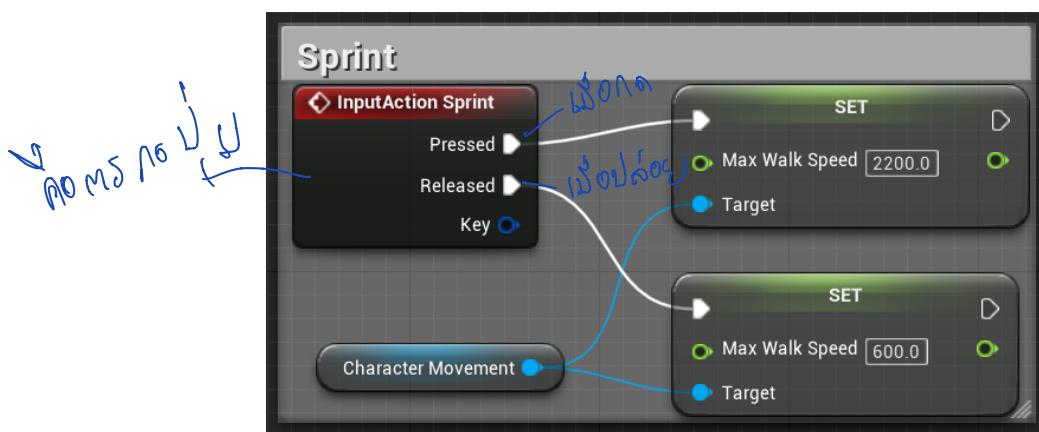
1.5.1. การเพิ่ม Input ในการควบคุม: Sprint และ Zoom

- เพิ่มการควบคุมผ่าน Keyboard โดยการเปิด Project Setting (ในเมนู Edit)
 - เลือก Engine/Input แล้วทำการเพิ่ม Sprint (เมื่อกด Left-Shift) และ Zoom (เมื่อกด right-click)
- ดังรูป



- ทำการเพิ่มการควบคุมให้กับ Player เมื่อทำการกด Left-Shift โดยการแก้ไข

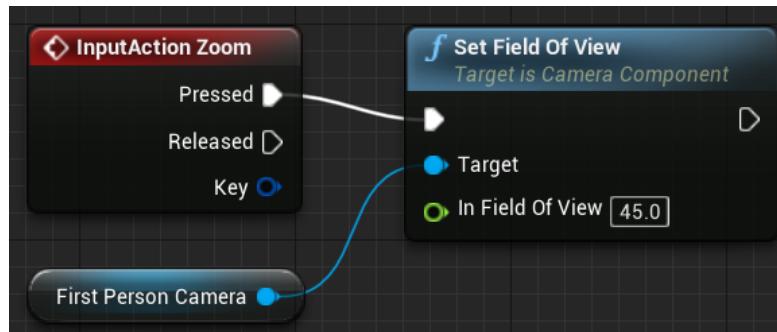
FirstPersonCharacter Blueprint (ใน FirstPersonBP/Blueprints folder) ดังรูป



- Play เมื่อเคลื่อนที่ และกด Shift-Left ตัว Player จะเคลื่อนที่ได้เร็วขึ้นตามค่าที่กำหนดไว้

1.5.2. การเปลี่ยนมุมมอง Field of View (FOV)

- ทำการเพิ่มการควบคุมให้กับ Player เมื่อทำการกด right-click โดยการแก้ไข **FirstPersonCharacter Blueprint** (ใน FirstPersonBP/Blueprints folder) ดังรูป

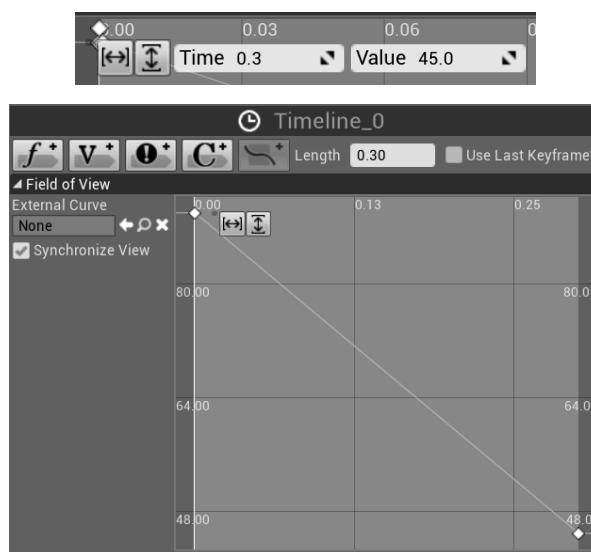


- Play เมื่อกด right-click ก็จะทำการแสดงผลในมุมมองที่เก็บลง (เหลือ 45 องศา)

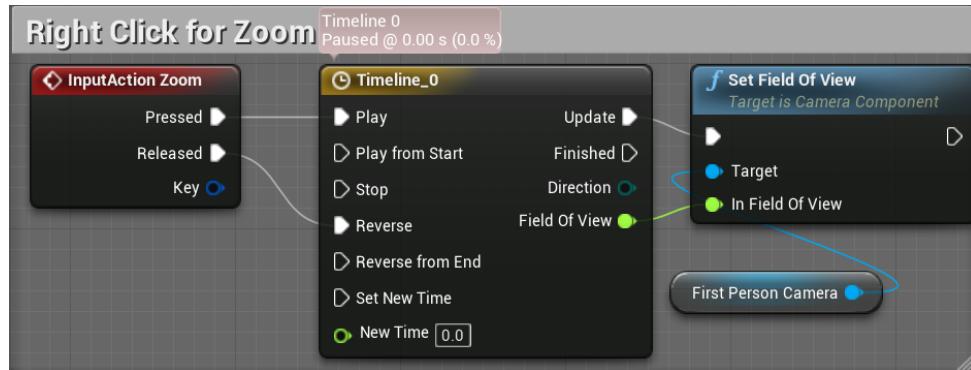
1.5.3. การใช้ Timeline สร้าง Animation อ่าย่างง่าย

จากข้อที่ผ่านมา เมื่อทำการกด right-click จะเกิดการ Zoom (เปลี่ยนค่า FOV) แต่ยังไม่ Smooth เพื่อเพิ่มความสวยงามในการเปลี่ยน FOV สามารถใช้เครื่องมือชื่อ Timeline ในการสร้าง Animation อ่าย่างง่ายได้

- ทำการแก้ไข **FirstPersonCharacter Blueprint** จากข้อที่ผ่านมา
- Break การเข้ามต่อจาก InputAction Zoom ไปยัง Set Field Of View และทำการเข้ามต่อใหม่โดยการลากจากปุ่ม Pressed ไปยังที่ว่าง แล้ว search หา Timeline และทำการ Add Timeline
- เปิด Timeline Editor โดยการ Double-click ที่ Timeline และทำการกำหนดค่า ดังรูป
 - กด Shift ค้างไว้แล้วกดลงตรงตำแหน่ง ใกล้ 0.0 และกำหนดค่า time =0, value = 90
 - ทำซ้ำอีกครั้ง แต่เปลี่ยนค่า time =0.3, value = 45 จะได้ 2 ตำแหน่ง ดังรูป



- ทำการแก้ไข Blueprint ดังรูป

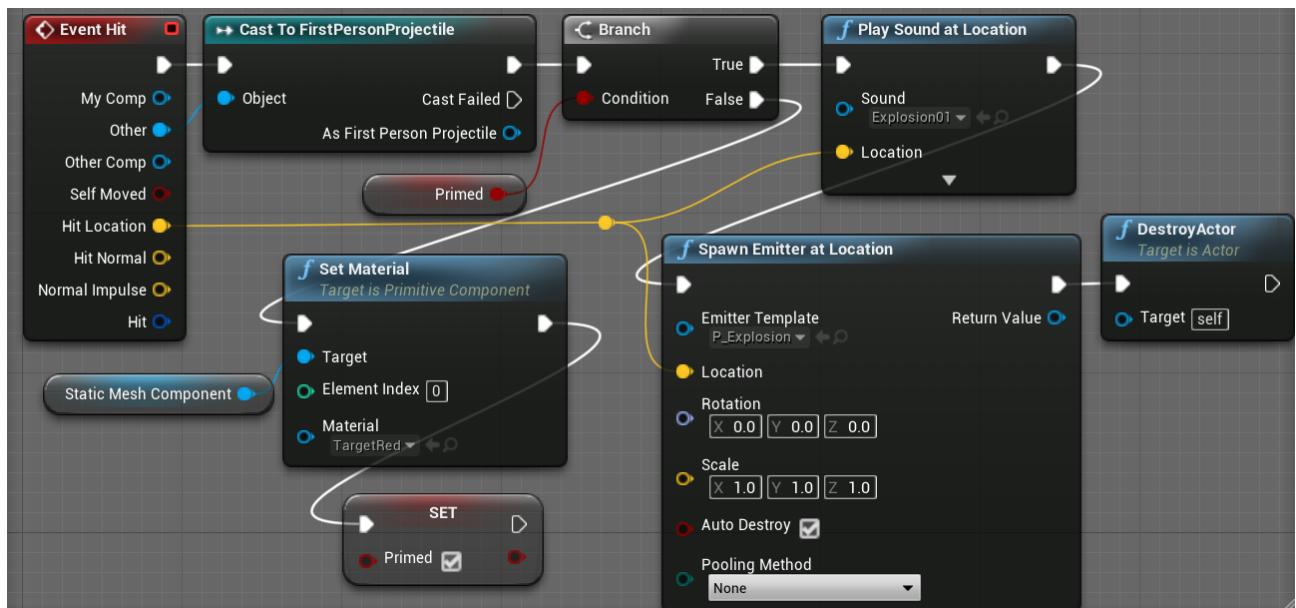


- Play แล้วทำการ Zoom จะเห็นว่ามีการเคลื่อนที่ของการ Zoom ที่ Smooth ขึ้น

1.5.4. การเพิ่มเสียง และ Particle Effect ในการ Interact

ในกรณีที่ยิงโดน CylinderTarget จะเกิดการเปลี่ยนสี หลังจากนั้น หากยิงโดนอีกรอบให้เกิดการทำลาย ตัวเองพร้อมเสียงและ Special Effect ทำได้โดยการเก็บค่า State ของ CylinderTarget ในตัวแปร

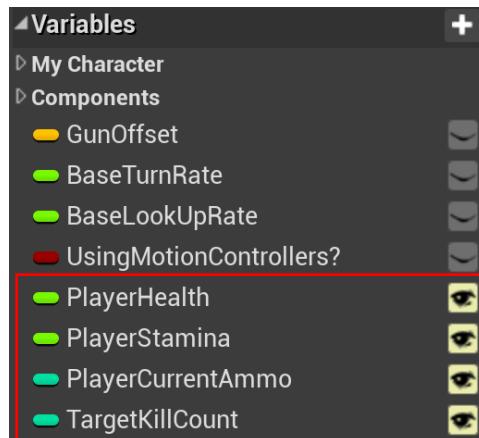
- เปิด **CylinderTarget_Blueprint** แล้วสร้าง Variable ชื่อ Primed ประเภท Boolean ให้ค่าเริ่มต้นเป็น False
- ทำการแก้ไข Graph ดังรูป
 - เมื่อถูกยิงครั้งแรก Primed จะเปลี่ยนจาก F → T
 - เมื่อถูกยิงครั้งที่ 2 (ในขณะที่ Primed เป็น True) จะทำการ Destroy ตัวเองพร้อมแสดง Effect ที่เลือก และเสียงเกิดขึ้นพร้อมกัน



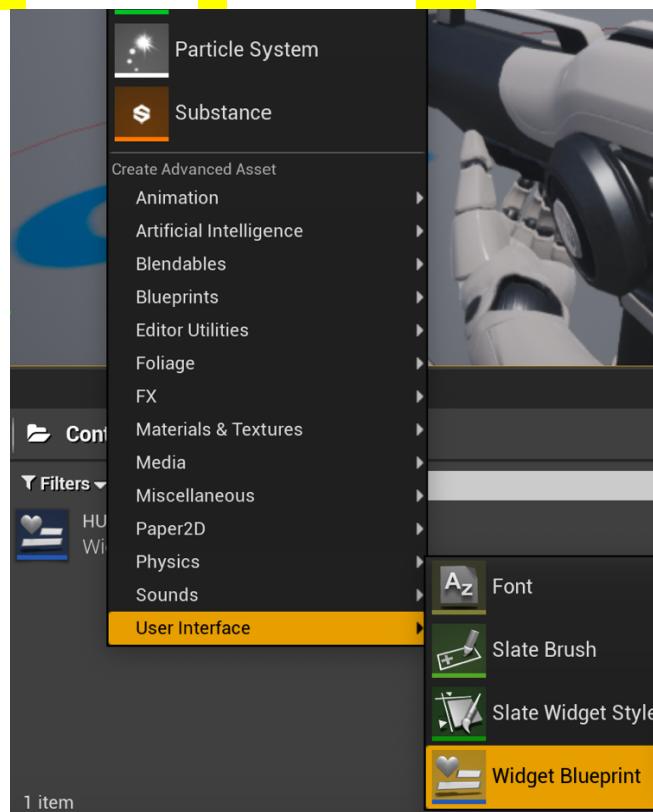
2) UI Elements ด้วย UMG (ใช้ “Map01_Mat”)

2.1. สร้าง UI ด้วย UMG

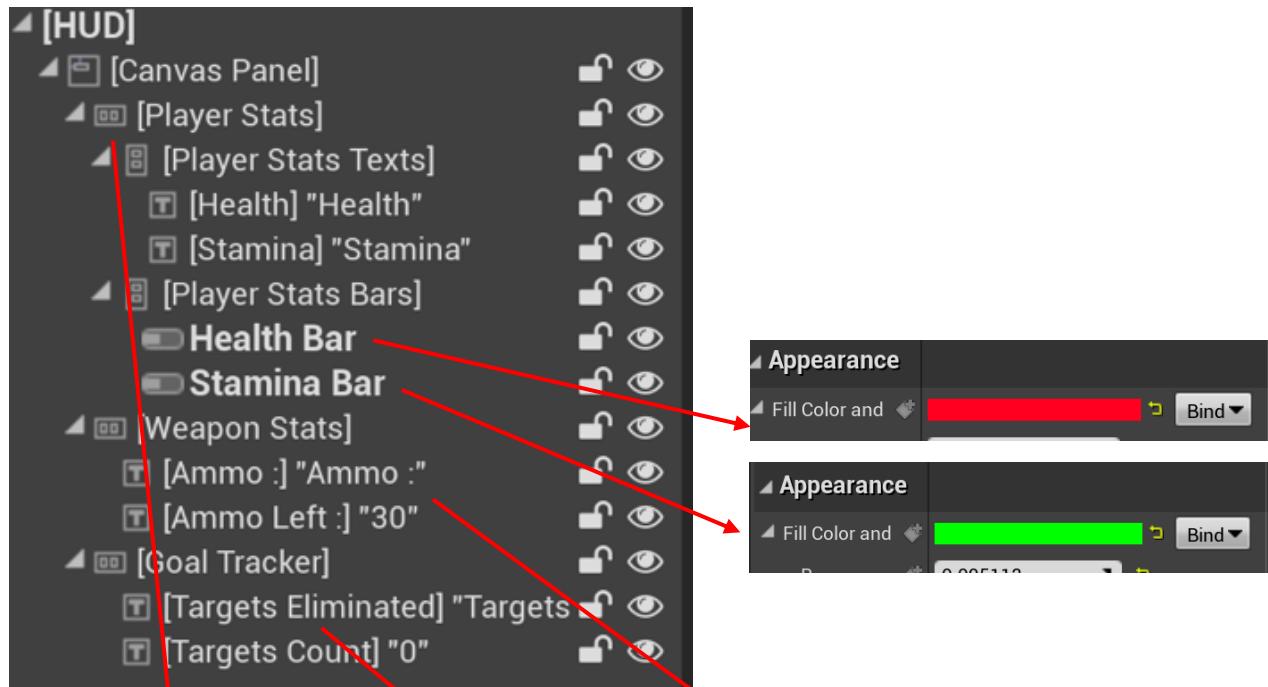
- เปิด FirstPersonCharacter Blueprint (FirstPersonBP/Blueprints)
- เพิ่มสร้างตัวแปร แล้ว set ให้เป็น Instance Editable ดังรูป
 - ประเภท Float: PlayerHealth และ PlayerStamina
 - ประเภท Integer: PlayerCurrentAmmo และ TargetKillCount



- สร้าง Widget Blueprint ใน UI folder ตั้งชื่อว่า HUD



- เปิด UMG Editor โดยการ double click ที่ HUD blueprint
- ทำการสร้าง UI ดังรูป
 - เลือก element ที่ต้องการจาก Palette แล้วลากมาวางที่ Hierarchy ให้ Canvas Panel
 - ทำการ Search เพื่อหา Horizontal box หรือ Vertical box ตามรูป (ตามความเหมาะสม)



- ทำการ Align UI elements ดังรูป (ใช้เครื่องมือในการ Alignment)



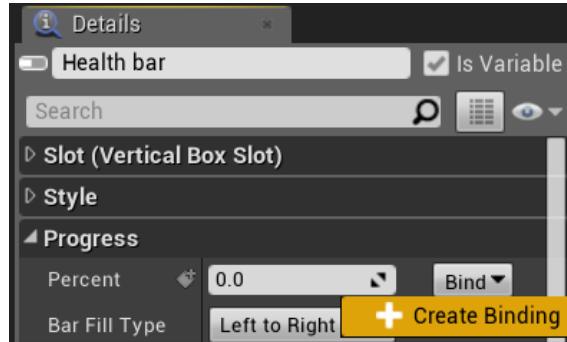
- เปิด FirstPersonCharacter Blueprint และทำการเชื่อมต่อ HUD Widget (Create Widget แล้วเลือก HUD) กับ Event Being Play ดังรูป



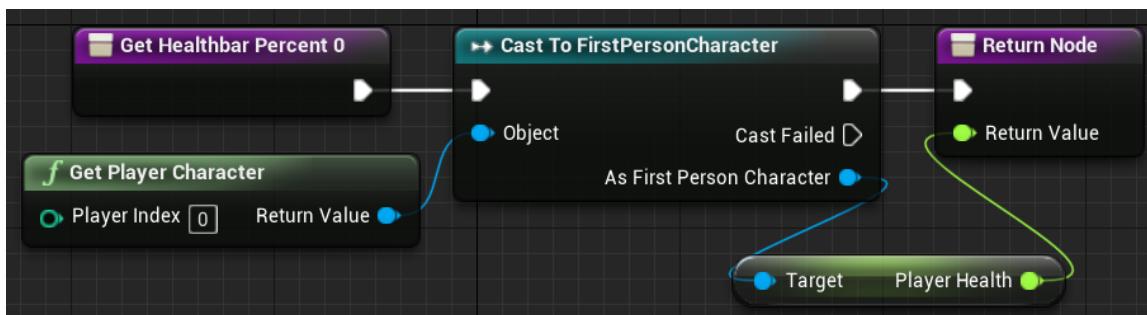
- Play จะมีการแสดง HUD ที่สร้างขึ้นในหน้าจอขณะทำการเล่น

2.2. Connecting and Tracking UI values

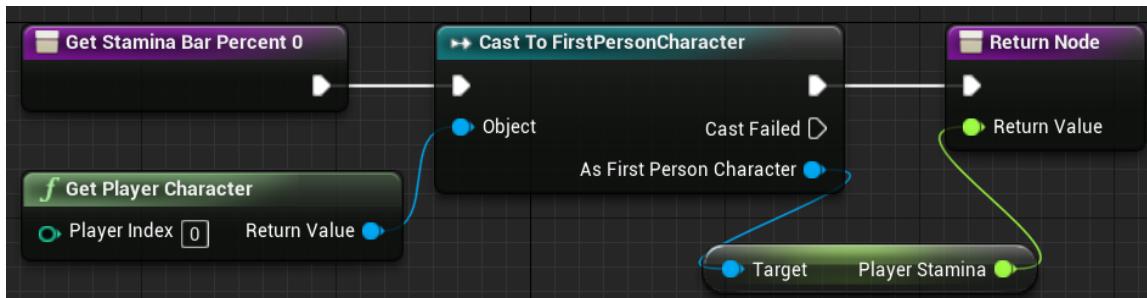
- เปิด HUD Blueprint (UMG Editor)
- เลือก Health Bar (ใน Hierarchy) และดูในเมนู Percent ในส่วน Detail ทำการสร้าง Create Binding จาก Bind Drop Down List ดังรูป (หน้าต่าง UGM Editor จะเปลี่ยนจาก Designer เป็น Graph)



- ทำการเชื่อมต่อ Graph ดังรูป เพื่อทำการเชื่อมต่อ Health Bar กับ FirstPersonCharacter Blueprint ผ่านวิธีการ Casting เพื่อเอาข้อมูลจาก Player มาแสดงผลผ่านหน้าจอ HUD



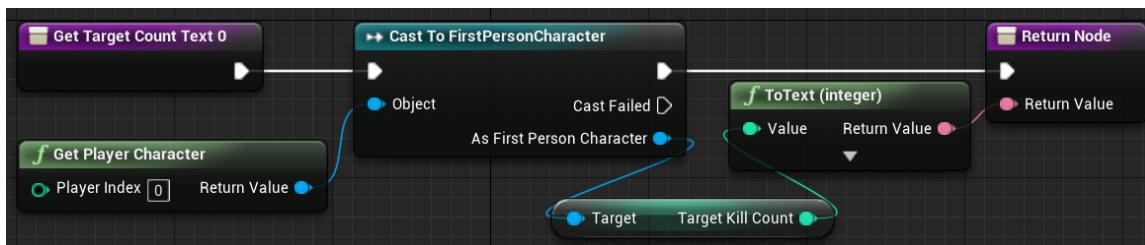
- ทำซ้ำกระบวนการจากข้อที่แล้ว แต่เลือกเป็น Stamina Bar และทำการต่อ Graph ดังรูป



- ทำชี้กรอบนการจากข้อที่แล้ว แต่เลือกเป็น **Ammo Left** แล้วทำการต่อ Graph ดังรูป



- ทำชี้กรอบนการจากข้อที่แล้ว แต่เลือกเป็น **Target Count** แล้วทำการต่อ Graph ดังรูป



- Play จะมีการแสดงผล HUD ขณะเดิน ดังรูป



Exercise

- แก้ไข Game โดยให้จำนวน Ammo ลดลงทุกครั้งที่ทำการยิง และเมื่อเหลือค่า Ammo เป็น 0 จะไม่สามารถยิงได้

Hint: FirstPersonCharacter Blueprint ส่วน Spawn Projectile

- เมื่อทำการยิงและสามารถทำลาย CylinderTarget ให้มีการเพิ่มค่า Targets Eliminated ในหน้าจอ HUD (ตรงกลาง) เพิ่มขึ้นครั้งละ 1

Hint: CylinderTarget_Blueprint