# รายงานความก้าวหน้าวิชา CE Project

**ครั้งที่ 2**

**ระหว่างวันที่ 28 ส.ค. 65 ถึงวันที่ 09 ก.ย. 65**

1. ชื่อโครงงาน (อังกฤษ) The Rouge‐Like Action RPG Game Development

## การดำเนินงานมีความก้าวหน้า % (ใช้ค่า **% Complete** จาก MS Project)

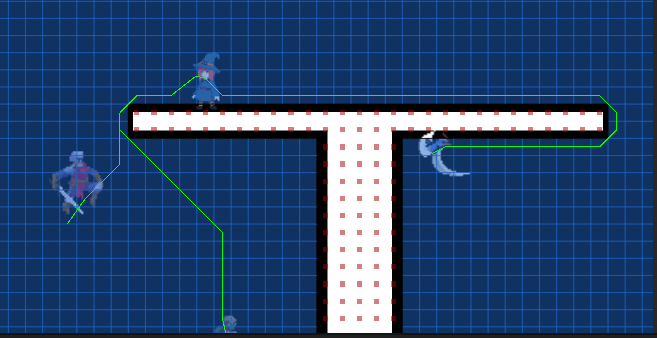
มีความก้าวหน้าเพิ่มขึ้นจากรายงานความก้าวหน้า ครั้งก่อน %

□ เร็วกว่าแผน วัน □ ช้ากว่าแผน วัน

## รายละเอียดความก้าวหน้า

**ศัตรู ( Enemy )**

• ทำ Template ของศัตรูและ Ai ของศัตรู : ส่วนนี้เราสามารถพัฒนาได้อย่างราบลื่นแต่ยังไม่เสร็จสมบูรณ์นักโดยในปัจจุบันระบบ Ai ของศัตรูสามารถพัฒนาไปได้ถึงการหาเส้นทางที่จะวิ่งไปหาผู้เล่นได้แล้ว แต่ปัจจุบันยังเป็นการเคลื่อนที่ไปหาผู้เล่นเพียงอย่างเดียว ( ยังไม่สามารถโจมตี , ออก Action ต่างๆที่ไม่ซ้ำซาก ) ซึ่งในส่วนนี้จะพัฒนาต่อไป



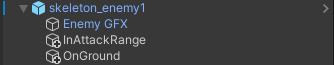
ภาพแสดงเส้นทางต่างๆที่ศัตรูจะใช้เดินทางไปยังผู้เล่น

• การทำ Animation ต่างๆของศัตรู : ในส่วนนี้พบปัญหาเล็กน้อยแต่ได้แก้ไขไปเรียบร้อยแล้ว สามารถผูกและโยง Animation ต่างๆจนนำไปแสดงผลในเกมได้แล้ว

## ปัญหาที่เกิดขึ้นและแนวทางการแก้ไข

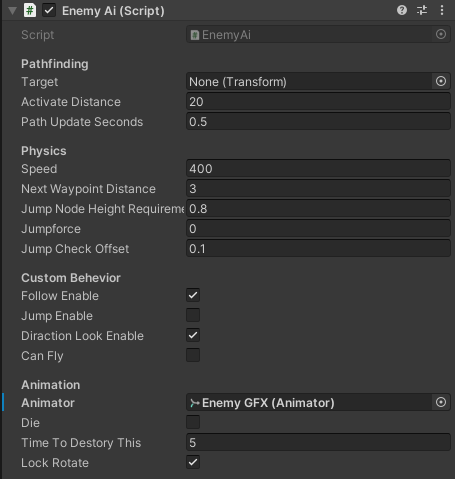
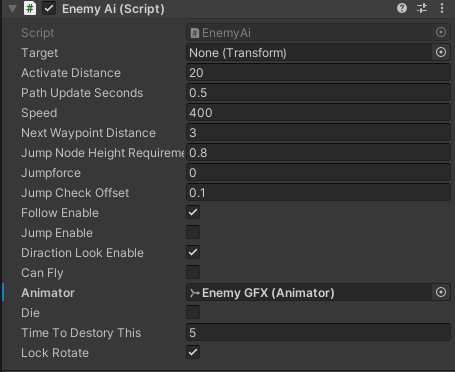
**ศัตรู ( Enemy )**

ปัญหาที่พบ : ทำ Template ของศัตรูจากที่ออกแบบไว้ตอนแรกคิดว่าจะใช้วิธีแบ่งเป็น Object แต่ละประเภท แต่หลังจากทำตามที่ออกแบบจนเสร็จแล้วพบว่า Objectของแต่ศัตรูแต่ละตัวจะประกอบไปด้วย Object ย่อยและ Object ย่อยก็จะมี Script ย่อยลงไปอีก



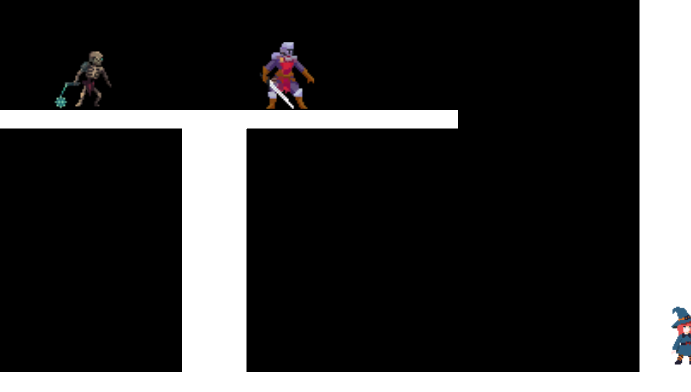
ซึ่งผู้พัฒนาได้ออกความเห็นว่าถ้า Script ต่างๆถูกแบบย่อยออกจากกันเกินความจำเป็นจะทำให้ความยุ่งยากในช่วง Parameter tuning ( การปรับบาลานซ์ของเกม ) ในภายหลัง

วิธีแก้ปัญหา : ใช้วิธีรวม Parameter ต่างๆไว้ใน Script เดียวโดยนำ Tag [Header] เข้ามาช่วยวิธีใช้จะทำให้สามารถแก้ไข Parameter ต่างๆได้โดยง่ายและจะสามารถแยกประเภทของ Parameter ได้อีกด้วย



ภาพเปรียบเทียบระหว่างการใช้ Tag Header และไม่ใช้

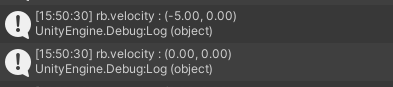
ปัญหาที่พบ : บางครั้งตัวผู้เล่นจะเคลื่อนที่ทะลุฉากหรือสิ่งที่ไม่ควรทะลุไปได้เช่นกำแพง



รูปที่ตัวผู้เล่นทะลุไปยังหลังกำแพง

วิธีแก้ปัญหา : ในช่วงที่เกิดปัญหาเราได้ใช้วิธีการเคลื่อนที่แบบ Transform.Translate วิธีนี้จะเป็นการเคลื่อนที่แบบไม่สนฟิสิกส์ของ Unity คล้ายๆกับการบังคับให้ Object นั้นไปยังพิกัดนั้นๆ ( อ้างอิง [Link](https://docs.unity3d.com/ScriptReference/Transform.Translate.html) )

หลังจากที่เราศึกษาวิธีการเคลื่อนที่ด้วยเทคนิคอื่นแล้วพบว่ามีเทคนิคอีก 2 อย่างคือ rb.velocity และ rb.addforce โดยครั้งนี้เราเลือก rb.velocity เนื่องเทคนิคนี้เป็นการใส่แรงและทิศทางเมื่อเรากดปุ่มและจะปรับให้แรงนั้นเป็น 0 เมื่อเราปล่อยปุ่ม



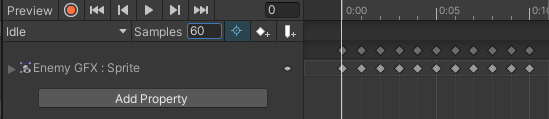
ภาพที่แสดงค่า velocity หลังจากปล่อยปุ่ม

โดย rb.addforce จะเป็นการใส่แรงและทิศทางไปเหมือนกันแต่ว่าหากปล่อยปุ่มแรงนั้นจะไม่เหลือ 0 แต่จะเป็นการค่อยๆผ่อนแรงนั้นๆลงแทน

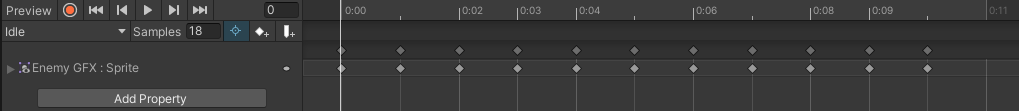
**Animation**

ปัญหาที่พบ : Frame ของตัวละครต่างๆมีค่อนข้างน้อยและเมื่อนำมาทำ Animation จะทำให้ Animation นั้นๆ เร็วเกินไปจนดูไม่เป็นธรรมชาติ

วิธีแก้ปัญหา : นำระบบ Sample rate ของ Animation เข้ามาช่วยโดยระบบนี้จะช่วยหน่วงความเร็วของ Animation ตามความเร็วที่เรากำหนดจึงทำให้ Animation render ได้อย่างสบายตา



ภาพเมื่อ Samples rate เป็น 60 ( ค่าเริ่มต้น )



ภาพเมื่อ Samples rate เป็น 18 ( ค่าที่ปรับแล้วคิดว่าลงตัว )

ทั้ง 2 ภาพจะแสดงการเล่นของ Animation ใน Time scale เท่า ๆ กัน

## สิ่งที่จะดำเนินการต่อไป

**ศัตรู ( Enemy )**

ในส่วน AI ของศัตรูจะทำการพัฒนาให้สมบูรณ์เท่าที่จะทำได้โดยในส่วนนี้จะรวมไปถึงบอสต่างๆในเกมด้วย