

# Unreal Engine 遊戲開發(Kingsglave-Origin)

專題編號：111-CSIE-S013

執行期限：110 年第 1 學期至 111 年第 1 學期

指導教授：白敦文

專題參與人員：108590010 張景辰  
108590020 林郁凱

## 一、摘要

《Kingsglave-Origin》為一款 ARPG 遊戲，由 Unreal Engine 4.27(UE4)遊戲引擎開發而成，使用引擎提供的可視化腳本系統「藍圖(Blueprint)」代替純 C++的開發，大幅減少時間成本，藉由關卡設計工具配合藍圖通訊建構出交互式場景，使用「行為樹(BehaviorTree)」設計 AI 任務執行序並為人物創建動畫狀態機與「動畫蒙太奇(AnimMontage)」來表現流暢的 3D 動作與邏輯行為。

本專題由始至終都希望能讓玩家有沉浸式的體驗，從一開始的引導到之後學會自行操作來體驗戰鬥，每次的推進都將帶來一些成就感，當玩家更好奇下一次會學到和發現什麼時，探索、互動和操作的目標已經達成，這款 ARPG 想帶給大家的，就是從中自由探索與學習的樂趣。

**關鍵詞：**動畫蒙太奇(AnimMontage) 藍圖(Blueprint)、行為樹(BehaviorTree)、ARPG、Unreal Engine

## 二、緣由與目的

從幾十年前的二維世界到現在因為硬體提升而達成的實時 3D 演算和物理引擎，更多的材質跟面數，讓復刻現實場景與採集人的行為數據變得更容易，還有基於深度學習算法的 AI，我們能看到越來越多的遊戲技術應用，遊戲在科技背後帶來的價值和技術是習習相關的。

製作遊戲的過程中會遇到動畫、物理、編成和美術等領域，當遇到困難時並

不會讓人有倦怠感，反而是覺得新鮮，會為了實現功能而產生學習的欲望，這是從前沒有過的學習體驗，也讓我們有機會去接觸一些新事物，接近學無止境的理想，藉此機會專題想試著做一款簡單且包括大部分元素的遊戲，ARPG 類型遊戲剛好符合以上條件，因此選擇了這個的題材。

## 三、開發工具

使用由 Epic Games 開發的 UE 遊戲引擎，底層用 C++編程的「藍圖(Blueprint)」腳本系統加速開發，美術方面的視覺、渲染、圖形和遊戲場景使用引擎內提供的設計工具和 Epic Games 提供的模組製作。

## 四、遊戲系統與設計

### (一) 地形編輯系統

使用地形工具從零開始新建地形，創建巨大且優化的開放式地塊，讓我們能在多種設備上維持正常遊戲的幀率。

### (二) 植被系統

在其他幾何體表面上渲染靜態網格體植被或 Actor 植被的工具，我們選擇使用靜態網格體植被，如此僅需調用一次繪製即可渲染大量實例。

### (三) 藍圖可視化腳本系統

在 UE 編輯器中，使用基於節點的介面創建遊戲的可玩性元素，通過定義在引擎中的面向對象的類或者對象來建構邏輯並創建實例。

### (四) 骨架網格體動畫系統

由多個動畫工具和編輯器組成，基於骨架的頂點變形來構建動畫，可播放和混

合動畫序列或自行定義動作，能創建基於邏輯的狀態機指定角色在不同情境下使用不同的動畫。

### (五) AI 系統

提供多個子系統使用，AI 行為樹系統處理決策進程，AI 感知系統將環境刺激傳輸到行為樹，EQS 系統處理環境數據的查詢，傳回行為樹並用於後續決策的過程。

### (六) UMG 系統

UMG 是一個可視化的 UI 創作工具，用來創建 UI 元素，UMG 使用控件在專門的控件藍圖中編輯，該藍圖分兩部分，設計器(Designer)部分允許界面的布局，圖表(Graph)部分則提供控件背後的功能。

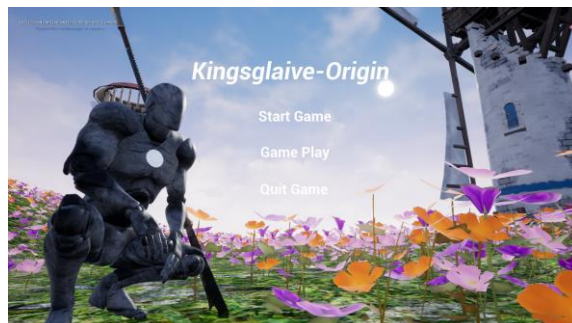
## 五、問題及解決方法

製作過程中最常發生藍圖通信時回報 None 的問題，原因是當類之間的互相調用發生時，該實例或變數不存在或已經消失，而隨著遊戲製作過程的往後推進，遊戲內包含的藍圖類通信與變數調用的數量已經開始膨脹，這時候會在調用函式 TickEvent(每幀自動調用的事件)時發生問題。我們的電腦每幀需要偵測的事件與變數太多，可能會有來不及在毫秒之間處理完畢的情況，這時候變數設置跟藍圖通訊的先後順序就變得更重要，可以試著在變數值快速的調用之間加上延遲(Delay)，讓電腦有足夠的時間反應。

在藍圖類裡有一個能與場景中 Actor 互動的關卡藍圖(Level Blueprint)，在其他藍圖裡想要和關卡藍圖達成通信的方法並不直觀，因為我們無法直接獲取他的類，這是 UE 當初思考過的設計，他們不希望你在一個關卡藍圖裡做太多事情，因為這是浪費資源的一件事情，正常一個遊戲是有好幾個關卡組成的，而當關卡切換時這些原本的設置就等於放空沒有再使用了，大家熟悉的遊戲過圖其實就是關卡間的切換，應該要設置的是我們頻繁使用的 Actor 的藍圖類，而我們在與關卡互動的設計上使用 GameInstance 類，他是一個從遊戲開始到結束都不會消失的類，我們

用它記錄其他藍圖類的轉變，再從關卡藍圖裡實時檢測來達到關卡與玩家的互動。

## 六、成果展示



▲圖 1. 開始介面



▲圖 2. 角色狀態介面



▲圖 3. 戰鬥畫面

## 七、結論

在製作遊戲的過程中常常會因為想把新想法加入而導致遊戲整體變得更大更複雜，而導致「範圍蔓延」的錯誤，我們在製作中期也遇到了這個問題，多虧有及時發現才如期把遊戲完善，我們會帶著這次學到的經驗與熱情繼續向前邁進。

## 參考文獻

[1] 虛幻官網

<https://www.unrealengine.com/zh-CN/>