



嶺東科技大學
LING TUNG UNIVERSITY

資訊管理系

謎窟遊戲設計與製作

指導教授： 陳明華 教授

組員名單： 陳韋銘 A08C173

劉彥緯 A08C037

王俊凱 A08C064

中華民國 104 年 5 月

嶺東科技大學

資訊管理系

謎窟遊戲設計與製作

中華民國一〇四年四月



嶺東科技大學
LING TUNG UNIVERSITY

資訊管理系

謎窟遊戲設計與製作

指導教授： 陳明華 教授

組員名單： 陳韋銘 A08C173

劉彥緯 A08C037

王俊凱 A08C064

中華民國104年5月



嶺東科技大學
LING TUNG UNIVERSITY

資訊管理系專題口試委員審定書

謎窟遊戲設計與製作

指導教授： 陳明華 教授

組員名單： 陳韋銘 A08C173

劉彥緯 A08C037

王俊凱 A08C064

指導教授：

口試委員：

中華民國 104年 5月 8日

謝 誌

本專題報告得以順利完成，首先要感謝恩師陳明華老師細心引導我們，耐心的協助我們，克服研究過程中所面臨的困難，給予我們最大的協助，使本專題得以順利完成。

研究報告口試期間，感謝夏則智老師、倪克明老師不辭辛勞細心審閱，不僅給予我們指導，並且提供寶貴的建議，使我們的專題內容以更臻完善，在此由衷的感謝。

最後，感謝系上諸位老師在各學科領域的熱心指導，增進商業管理知識範疇，在此一併致上最高謝意。

陳韋銘、劉彥緯、王俊凱

謹誌

中華民國104年5月於嶺東

摘 要

整個遊戲的製作分成三大方向為「設計」、「製作」與「微調」，「設計」的部分主要是確認製作內容並進行規劃，其中包含角色設定、劇情背景、美術風格、系統規劃、遊戲劇情與物件分析，「製作」的部分是依照設計的內容進行實作，分別有物件建模、撰寫腳本與遊戲實作，最後得「微調」是將完成的遊戲進行最後的調整，主要以除錯與美化為主。

整個製作過程最主要運用的軟體以 Blender 與 Unity 為主，Blender 主要是進行 3D 建模的功能，Unity 主要是將模組與腳本進行結合的遊戲開發引擎，最終整個專題完成的是一款結合解謎與恐怖的 3D 遊戲。

關鍵字：3D 遊戲、遊戲開發、遊戲引擎

目 錄

| | 頁次 |
|------------------------|-----|
| 謝 誌 | I |
| 摘 要 | II |
| 目 錄 | III |
| 表目錄 | V |
| 圖目錄 | VI |
| 第壹章 緒論 | 1 |
| 1.1 開發背景 | 1 |
| 1.2 開發動機 | 1 |
| 1.3 開發目的 | 1 |
| 第貳章 文獻回顧與探討 | 2 |
| 2.1 Unity 市場資料 | 2 |
| 2.2 恐怖遊戲資料 | 2 |
| 2.3 Blender 教學資料 | 2 |
| 第參章 遊戲開發架構與環境 | 3 |
| 3.1 遊戲架構 | 4 |
| 3.2 開發環境 | 4 |
| 3.2.1 作業環境 | 4 |
| 3.2.2 開發引擎 | 4 |
| 3.2.3 遊戲腳本 | 4 |
| 3.2.4 建模軟體 | 5 |
| 第肆章 遊戲設計 | 6 |
| 4.1 角色設定 | 6 |
| 4.2 劇情背景 | 6 |
| 4.3 美術風格 | 7 |
| 4.4 系統規劃 | 7 |
| 4.5 遊戲劇情 | 8 |
| 4.6 分析物件 | 10 |
| 第伍章 遊戲製作 | 20 |

| | |
|----------------------|----|
| 5.1 建模清單 | 20 |
| 5.2 撰寫腳本 | 26 |
| 5.3 遊戲實作 | 32 |
| 第陸章 遊戲製作成果 | 58 |
| 第柒章 結論 | 76 |
| 參考文獻 | 77 |
| 附錄 1 門物件的程式碼 | 78 |
| 附錄 2 檢視道具的程式碼 | 81 |
| 附錄 3 顯示圖片的程式碼 | 84 |
| 附錄 4 調查子彈的程式碼 | 85 |
| 附錄 5 攝影機設定的程式碼 | 86 |
| 附錄 6 玩家移動的程式碼 | 88 |
| 附錄 7 背包顯示的程式碼 | 91 |
| 附錄 8 調查功能的程式碼 | 94 |

表目錄

| | 頁次 |
|---------------------|----|
| 表 4.1 物件分析表 B1..... | 11 |
| 表 4.2 物件分析表 B2..... | 13 |
| 表 4.3 物件分析表 B3..... | 15 |
| 表 4.4 物件分析表 B4..... | 17 |
| 表 4.5 物件分析表 B5..... | 19 |
| 表 5.1 建模清單 | 20 |
| 表 5.2 撰寫腳本 | 26 |

圖目錄

| | 頁次 |
|----------------------------------|----|
| 圖 2.1 遊戲開發使用 Unity 之市場比例[1]..... | 2 |
| 圖 3.1 遊戲開發流程 | 3 |
| 圖 4.1 主角（正面圖） | 6 |
| 圖 4.2 主角（測面圖） | 6 |
| 圖 4.3 地下建築結構圖 | 7 |
| 圖 4.4 樓層介紹圖 | 7 |
| 圖 4.5 心靈旅館截圖（一） | 7 |
| 圖 4.6 心靈旅館截圖（二） | 7 |
| 圖 4.7 平面圖 B1..... | 8 |
| 圖 4.8 平面圖 B2..... | 8 |
| 圖 4.9 平面圖 B3..... | 8 |
| 圖 4.10 平面圖 B4..... | 8 |
| 圖 4.11 平面圖 B5..... | 9 |
| 圖 4.12 物件分析圖 B1..... | 10 |
| 圖 4.13 物件分析圖 B2..... | 12 |
| 圖 4.14 物件分析圖 B3..... | 14 |
| 圖 4.15 物件分析圖 B4..... | 16 |
| 圖 4.16 物件分析圖 B5..... | 18 |
| 圖 5.1 正方體之面編輯 | 32 |
| 圖 5.2 正方體之延伸 | 32 |
| 圖 5.3 正方體之延伸杯柱 | 32 |
| 圖 5.4 正方體之延伸杯底 | 32 |
| 圖 5.5 除去頂部面 | 33 |
| 圖 5.6 模型表面圓弧化 | 33 |
| 圖 5.7 模型拖曳至 Unity | 33 |
| 圖 5.8 地圖拉進場景 | 34 |
| 圖 5.9 模式 Model 切換 | 34 |
| 圖 5.10 模式 Model 內參數設定 | 35 |

| | |
|--------------------------------------|----|
| 圖 5.11 產生光源物件 | 35 |
| 圖 5.12 門物件移動 | 36 |
| 圖 5.13 物件位置微調 | 36 |
| 圖 5.14 物件 Add Component 的選單 | 37 |
| 圖 5.15 選單內的 Box Collider 選項 | 37 |
| 圖 5.16 門物件的碰撞範圍 | 38 |
| 圖 5.17 碰撞設定 | 38 |
| 圖 5.18 套用程式碼到門物件上 | 39 |
| 圖 5.19 門物件程式碼詳細參數 | 39 |
| 圖 5.20 碰撞觸發選項 | 40 |
| 圖 5.21 物件移動至場景 | 40 |
| 圖 5.22 碰撞設定 | 41 |
| 圖 5.23 碰撞範圍 | 41 |
| 圖 5.24 物件套用程式碼 | 42 |
| 圖 5.25 程式碼參數 | 42 |
| 圖 5.26 玩家模型拖曳至場景 | 43 |
| 圖 5.27 模型 Tag 調整 | 43 |
| 圖 5.28 產生 Point Light 光源物件 | 44 |
| 圖 5.29 物件拖曳 | 44 |
| 圖 5.30 調整光源作標 | 45 |
| 圖 5.31 選單 Add Component | 45 |
| 圖 5.32 新增 Character Controller | 46 |
| 圖 5.33 套用程式碼 | 46 |
| 圖 5.34 攝影機參數 | 47 |
| 圖 5.35 製作調查子彈 | 47 |
| 圖 5.36 子彈物件套用顏色 | 48 |
| 圖 5.37 子彈物件套用程式碼 | 48 |
| 圖 5.38 選單 Add Component | 49 |
| 圖 5.39 新增 Rigidbody | 49 |
| 圖 5.40 子彈物件 Rigidbody 參數設定 | 50 |
| 圖 5.41 子彈物件停用材質 | 50 |

| | |
|-------------------------------------|----|
| 圖 5.42 子彈物件改名製程 Prefab | 51 |
| 圖 5.43 子彈物件套用程式碼 | 51 |
| 圖 5.44 新增 Plane 物件 | 52 |
| 圖 5.45 物件 Plane 調整大小與顏色 | 52 |
| 圖 5.46 物件製程 Prefab | 53 |
| 圖 5.47 新增 PatrolPoint 的 Tag | 53 |
| 圖 5.48 路徑編排 | 54 |
| 圖 5.49 新增 Animator Controller | 54 |
| 圖 5.50 拖曳玩家動畫 | 55 |
| 圖 5.51 新增 Make Transition | 55 |
| 圖 5.52 關聯到動畫 | 56 |
| 圖 5.53 命名動畫 | 56 |
| 圖 5.54 關聯之程式碼設定 | 57 |
| 圖 6.1 遊戲開始畫面 | 58 |
| 圖 6.2 遊戲劇情畫面 | 58 |
| 圖 6.3 遊戲操縱畫面 | 59 |
| 圖 6.4 樓層 B1 場景 | 60 |
| 圖 6.5 樓層 B1 地圖畫面 | 60 |
| 圖 6.6 樓層 B1 教室 1 畫面 | 61 |
| 圖 6.7 樓層 B1 教室 1 開門畫面 | 61 |
| 圖 6.8 樓層 B1 教室 1 內部鐵櫃 | 62 |
| 圖 6.9 樓層 B1 教室 1 內部其他擺設 | 62 |
| 圖 6.10 樓層 B1 教室 1 內的紙條 | 63 |
| 圖 6.11 樓層 B1 教室 1 內得到鐵櫃鑰匙 | 63 |
| 圖 6.12 樓層 B1 教室 1 內得到鐵槌 | 64 |
| 圖 6.13 樓層 B1 教室 1 內得到螺絲起子 | 64 |
| 圖 6.14 樓層 B1 教室 2 畫面 | 65 |
| 圖 6.15 樓層 B1 教室 2 內的紙條 | 65 |
| 圖 6.16 樓層 B1 男廁最後一間廁所 | 66 |
| 圖 6.17 樓層 B1 男廁離開時畫面 | 66 |
| 圖 6.18 樓層 B1 教室 4 畫面 | 67 |

| | |
|--------------------------------|----|
| 圖 6.19 樓層 B1 教室 4 內部畫面 | 67 |
| 圖 6.20 樓層 B1 教室 4 內的紙條 | 68 |
| 圖 6.21 樓層 B1 女廁某間廁所 | 68 |
| 圖 6.22 樓層 B1 女廁某間廁所把門關上後 | 69 |
| 圖 6.23 樓層 B1 女廁鏡子 | 69 |
| 圖 6.24 樓層 B1 樓梯口 | 70 |
| 圖 6.25 樓層 B2 教室 8 外走廊 | 71 |
| 圖 6.26 樓層 B2 下樓樓梯口 | 71 |
| 圖 6.27 樓層 B4 樓梯口死亡 | 72 |
| 圖 6.28 樓層 B3 男廁 | 73 |
| 圖 6.29 樓層 B3 表演廳 | 73 |
| 圖 6.30 樓層 B4 教室 2 | 74 |
| 圖 6.31 遊戲結局 | 75 |

第壹章 緒論

1.1 開發背景

現在是二十一世紀的起頭，科技的崛起，從最古老的遊戲裝置陰極射線管娛樂裝置，至街機、掌機、專用機與電腦，而電腦這塊已成普級化的現代，人手一台電腦，最近各個知名遊戲研發場也在鑽研頭戴式虛擬實境裝置；娛樂事業也成了現代重要的角色，其中電腦的普及，使得電腦遊戲的開發也以嶄新的速度，爆炸性的開發，現在處處都是免費遊戲，有的是大型線上遊戲，小的則行動裝置遊戲，而遊戲的開發門檻也越來越低，相對的市場競爭也越來越激烈，能大賣的作品收入上則幾億，有的連聽都沒聽過，而各個企業看準這個商機進攻市場，開發遊戲也成了現代重要的趨勢之一。

1.2 開發動機

隨著時代的推進，遊戲的普級，遊戲的開發門檻降低，遊戲產業也跟著電視、電影、音樂等娛樂產業改革，其中遊戲結合了美術、音樂、劇本、程式等跨領域的綜合體，在製作專題的過程將一邊探討遊戲對於人的吸引力及瞭解開發的整體流程，學習自身對不同事物的觀察、分析，培養多元的視野。

1.3 開發目的

在製作與開發的過程中，可以學習與了解到，完成一個作品，所需要的要素，當然不單單只有遊戲，像是程式開發與專案處理等，都是通用的學習架構，然而遊戲的開發可以學習到劇情編排、遊戲架構、系統分析、邏輯分析、3D 建模、程式撰寫等多元學習，讓本組組員達到多元學習是本專題的最終目的。

第貳章 文獻回顧與探討

2.1 Unity 市場資料

根據 Unity 官方在 2014 年 8 月公佈的報告：Unity 引擎佔據全功能遊戲引擎市場 45% 的比例，如圖 2.1；Unity 全球用戶已經超過 330 萬人，每月活躍用戶數高達 60 萬；全世界有 6 億的玩家在玩使用 Unity 引擎製作的遊戲，用 Unity 創造的應用和遊戲目前的累計下載量達到了 87 億次[1]。

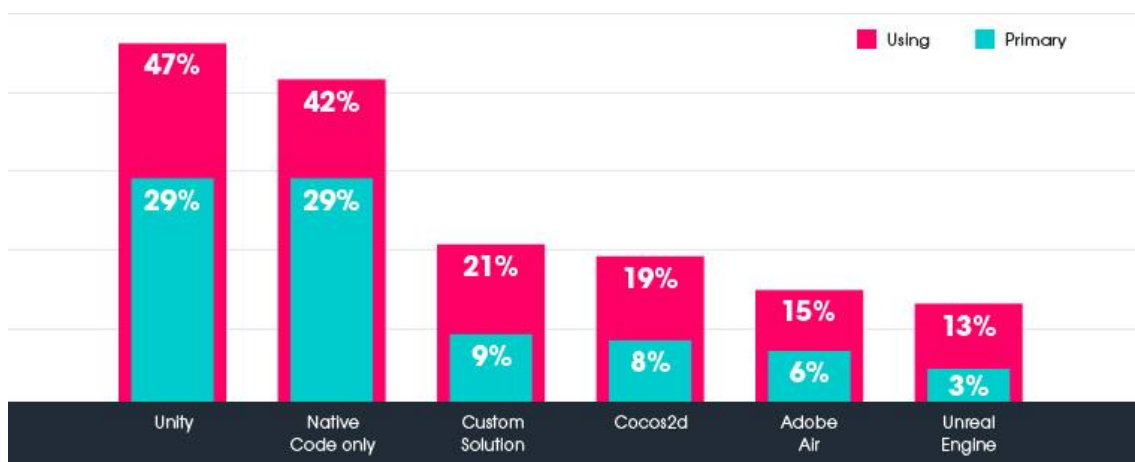


圖 2.1 遊戲開發使用 Unity 之市場比例[1]

2.2 恐怖遊戲資料

恐怖遊戲，又稱驚悚遊戲，英文 survivor horror，是以驚嚇、恐怖為主題的電子遊戲，是大約在二十世紀末開始流行的電子遊戲類型，代表作有惡靈古堡系列、沉默之丘系列、零系列及死魂曲；此類型的遊戲類似恐怖小說與恐怖電影，有著可怕的故事情節、怪異的角色造型，配上懸疑音樂，令讀者或觀眾感到恐懼，以達致娛樂的效果[2]。

2.3 Blender 教學資料

Blender 為本專題建模用的自由軟體，而自學的資料取自於網路上的「Blender 動畫設計 - 陳鍾誠的網站」這個網站，作者為陳鍾誠老師，於金門大學資訊工程系教職，授權方式採用創作共用(Creative Common)3.0 版的「姓名標示-非商業性-相同方式分享」授權條款[3]。

第參章 遊戲開發架構與環境

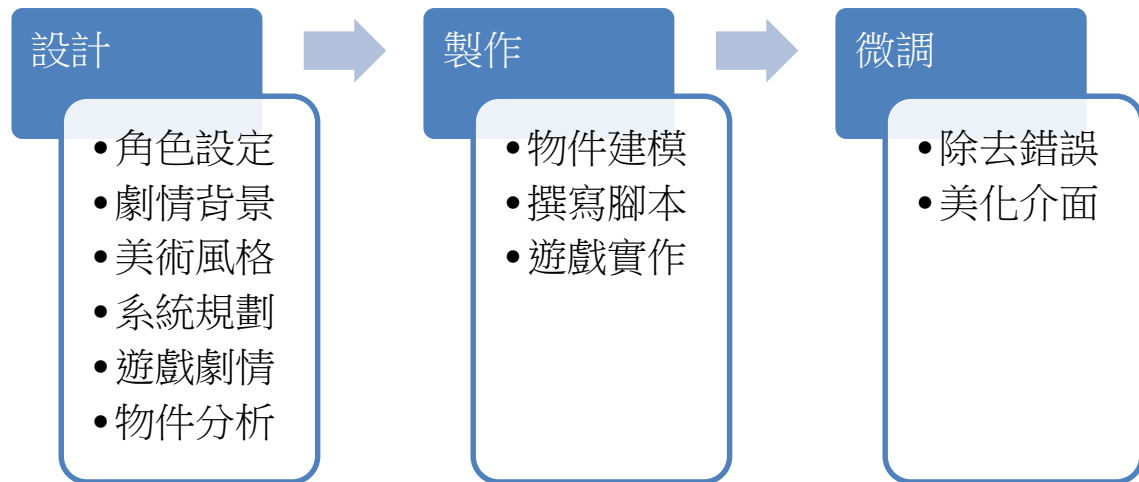


圖 3.1 遊戲開發流程

本專案將遊戲的開發流程粗略分為「設計」、「製作」以及「微調」，必須由左而右執行，如圖 3.1 之流程，其圖中的「設計」也就是遊戲企劃，而製作「製作」的部分，是依照左邊的「設計」所寫好的劇本開始分析，得到需要製作的模組與遊戲處發事件，首先從建模，再來依照事件撰寫腳本，最後再將所有零件整合完成遊戲，「微調」是屬於製作完成後的工作，目前優先於完成遊戲，使的美化介面部分被忽略，將此項目移至最後一項之項目，而「微調」之中有除去錯誤，才可專心作美化介面的部分。

3.1 遊戲架構

透過小組成員開會討論，釐清遊戲的整體架構與走向，其中分成以下幾項。

- 角色設定
- 劇情背景
- 美術風格
- 系統規劃
- 遊戲劇情
- 分析物件

3.2 開發環境

在開發方法之下的遊戲製作為實作時，所使用的工具。

3.2.1 作業環境

Microsoft Windows 7

Windows 7 是微軟公司推出的電腦作業系統，提高了螢幕觸控支援和手寫辨識，支援虛擬硬碟，改善多核心處理器的運作效率，它支援運行 3D 遊戲、視訊播放等多媒體程式，同時 DirectX 10 也可以在遠端桌面環境使用，其中 DirectX 系列支援最高為 11 版[4]。

3.2.2 開發引擎

Unity 3D

Unity 是一套跨平台的遊戲引擎，可開發執行於 PC、Mac OS 單機遊戲等多個跨平台遊戲，Unity 也可開發多人即時線上遊戲，腳本可用 JavaScript、C# 或 Boo 加以編寫[5]。

3.2.3 遊戲腳本

C#

C#，是微軟推出的一種基於 .NET 框架的、物件導向的高階程式語言，C# 由 C 語言和 C++ 衍生而來，繼承了其強大的效能，同時又以 .NET 框架類別庫作為基礎，擁有類似 Visual Basic 的快速開發能力。C# 由安德斯·海爾斯伯格主持開發，微軟在 2000 年發行了這種語言[6]。

3.2.4 建模軟體

Blender

Blender 是一套三維繪圖及渲染軟體。它具有跨平台的特性，支援 FreeBSD，IRIX，GNU/Linux，Microsoft Windows，Mac OS X，Solaris，及 SkyOS，支援不同的幾何圖元，包括多邊形網紋，快速表層塑模，曲線及向量字元，在 Blender 中，物體與資料是分離的，這使其可以快速塑模[7]。

本專題的製作過程採取先「建立各式物件模型」再「程式語言編寫」，讓所有組員全程參與專案的每個細節，透過接觸不同性質的工作，累積不同的經驗，加深對每個流程的瞭解，並透過討論尋找更有效率的專案規劃。

第肆章 遊戲設計

4.1 角色設定

玩家扮演的角色是一位大學生，參考圖 4.1 圖 4.2 為範例。

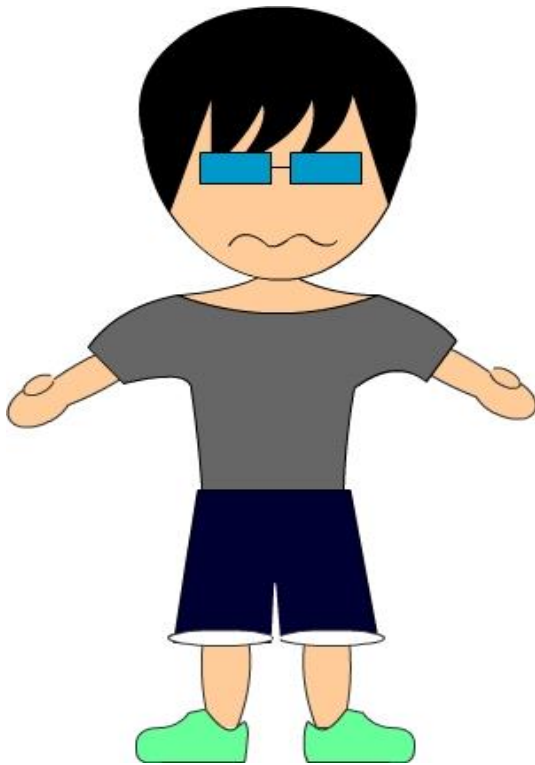


圖 4.1 主角（正面圖）

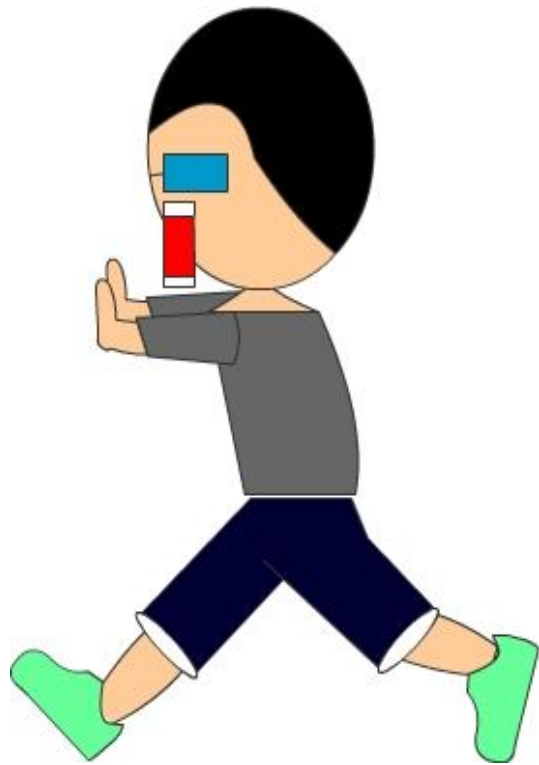


圖 4.2 主角（側面圖）

4.2 劇情背景

玩家扮演的大學生意外受困於校園內的體育館，因而發現一個通往地下建築的樓梯，為了尋找大門鑰匙脫困，大學生冒險往下探索，卻碰上一連串怪異的事件裡，參考圖 4.3 圖 4.4 為範例。

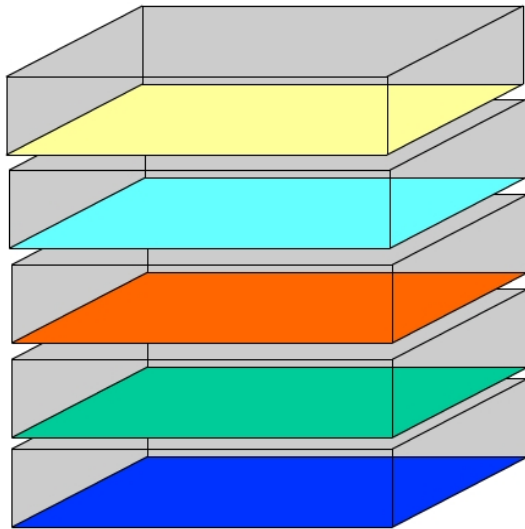


圖 4.3 地下建築結構圖



圖 4.4 樓層介紹圖

4.3 美術風格

小組成員討論過程中，利用影音網站搜尋恐怖遊戲的相關影片，並評估小組的開發能力，初期以完成作品為優先決定以簡單的風格呈現，如圖 4.5 圖 4.6 為採用範例，擷取自網路玩家實況影片[8]。



圖 4.5 心靈旅館截圖（一）

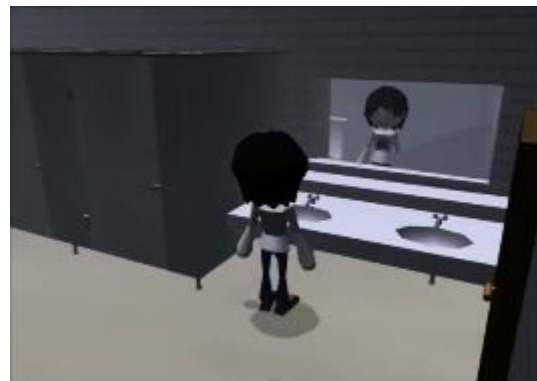


圖 4.6 心靈旅館截圖（二）

4.4 系統規劃

系統規劃包括了操作設定、遊戲道具及遊戲地圖，將條列如下。

- 操作設定：移動鍵（WSAD），蹲下（Ctrl），跑步（Shift），執行（F）
- 遊戲道具：對應各關卡需求，需要蒐集相應的物品滿足過關條件
- 遊戲地圖：一共五層樓，五個關卡，滿足過關條件即可開啟下一關的入

4.5 遊戲劇情

遊戲規劃了短篇的劇情，玩家在探索地圖的過程中會發現一些字條與記事本，記載有關這棟地底建築曾發生過的事，隨著遊戲時間的進行，收集的字條增加，玩家也能逐漸聯想地底建築背後的歷史，但每次蒐集的線索都是片段式的文字，彼此之間並沒有明確的關聯，這樣的設計希望玩家在遊戲途中能運用想像力推測出自己認為合理的結局。

B1 一般教室層，這一層是給玩家熟悉操作的關卡，玩家可以在這一層自由探索，除了封住通往下一層的樓梯以外，將不設置任何阻礙，探索的過程中只會發生一些詭異的事件，讓玩家感覺疑惑，沒有無任何怪物出現，玩家在這一層不會受到生命威脅。B1 樓層參考圖 4.7 為範例。

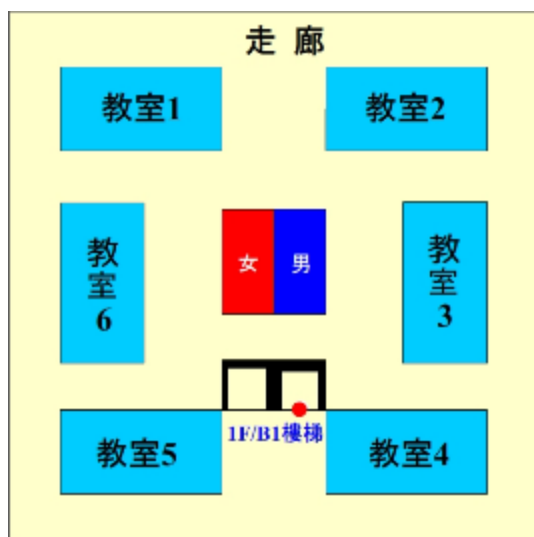


圖 4.7 平面圖 B1



圖 4.8 平面圖 B2

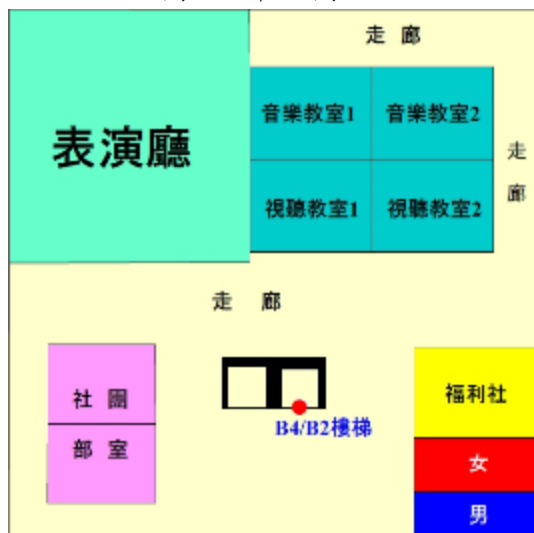


圖 4.9 平面圖 B3

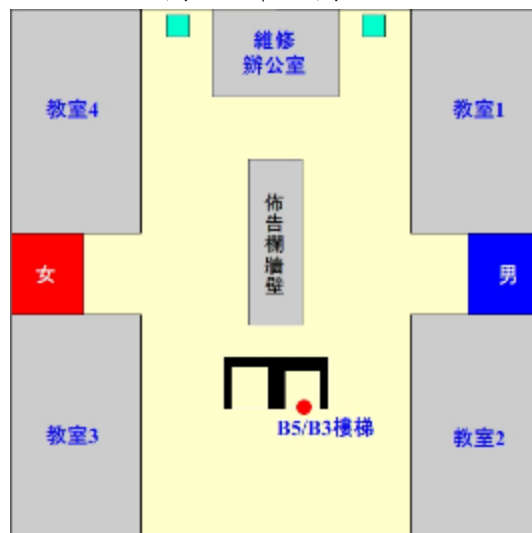


圖 4.10 平面圖 B4

B2 一般教室、化學實驗層，此層樓開始會加入幽靈的攻擊事件，玩家必須一邊閃躲，一邊收尋可用的物品。B2 樓層參考圖 4.8 為範例。

B3 視聽影音層，加入實體怪物，玩家將在此層樓收集到遇難學生留下的訊息瞭解怪物的習性，怪物擁有敏銳的聽力及較差的視力，玩家需要製造聲響來引誘怪物前往某些地方，像是用手錶鬧鐘引開一隻怪物、用音樂引來一群怪物。B3 樓層參考圖 4.9 為範例。

B4 電腦教室層，玩家需要前往維修辦公室找到電源箱，使其運作，樓層恢復電力後，玩家啟動通往五樓樓梯鐵門的時候，恢復供電的消防設備警鈴大作，所有的怪物都醒了！B4 樓層參考圖 4.10 為範例。

B5 老師辦公層，現在開始與時間賽跑，總務處的大門扭曲無法打開；需要破壞，遊戲時間每兩分鐘增加一隻怪物(怪物產生時，玩家聽見吼叫聲)，破壞用的工具在教官室裡，取得 1F 鑰匙後；若玩家成功逃往樓梯口，啟動文字劇情，遊戲結束。B5 樓層參考圖 4.11 為範例。

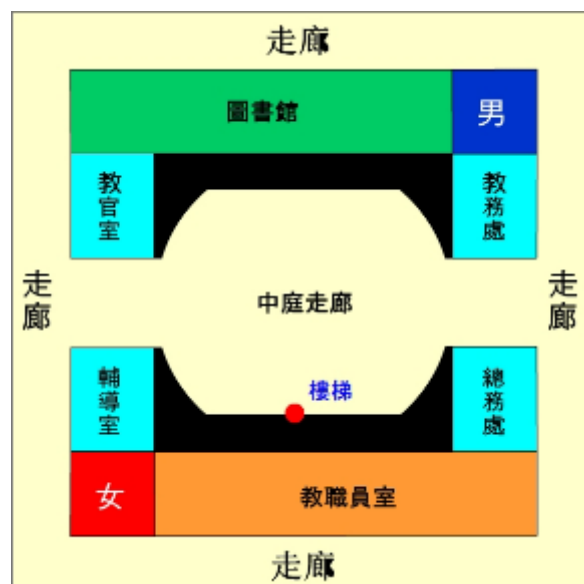


圖 4.11 平面圖 B5

4.6 分析物件

首先從遊戲劇情，分析所有需要的物件，依照一張圖一個表，從 B1 至 B5 依序，如下。

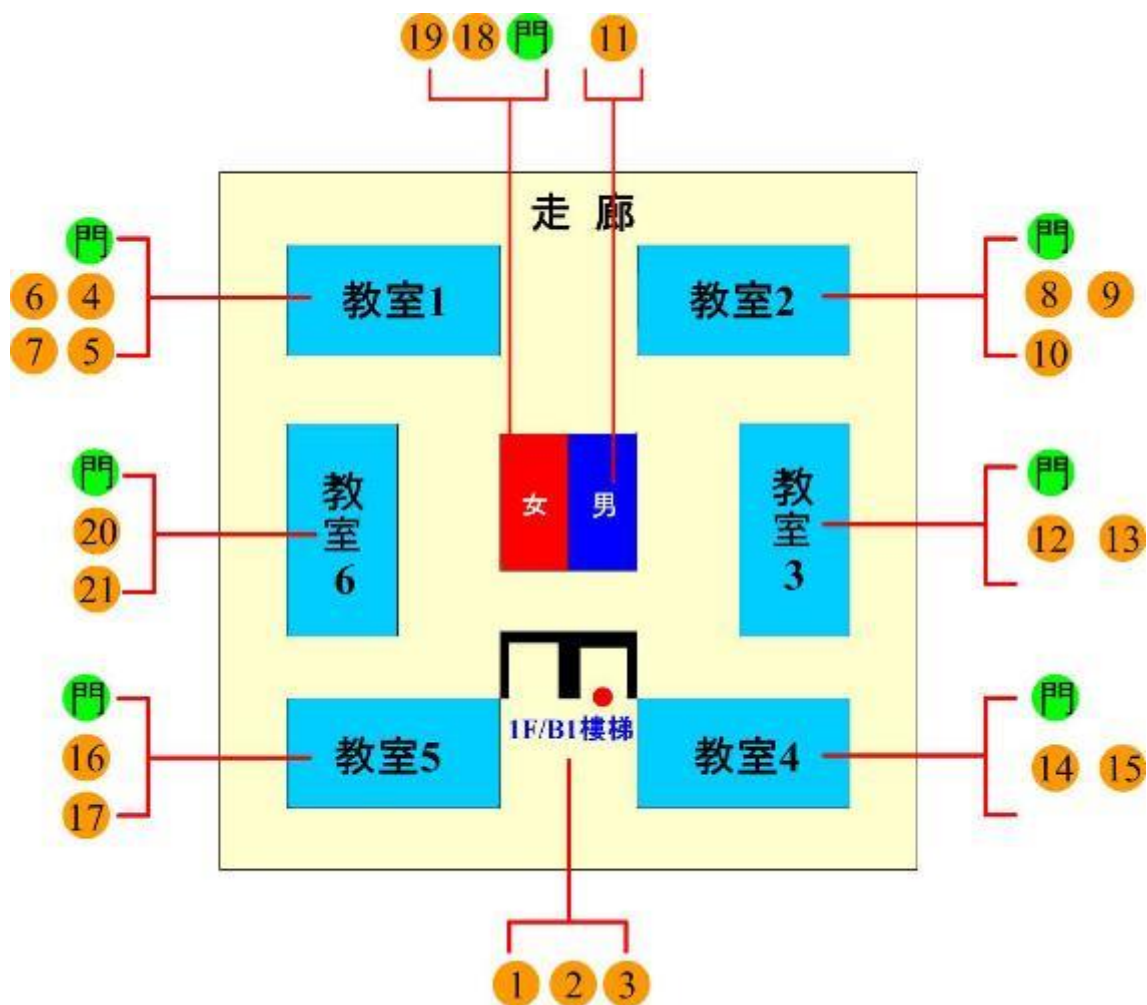


圖 4.12 物件分析圖 B1

表 4.1 物件分析表 B1

| 編號 | 物件 | 事件 |
|----|---------------|-------------------------|
| 1 | 鐵門 | 檢查角色身上是否有開鎖條件 |
| 2 | 說明書 | 角色背包增加該物品、提示訊息 |
| 3 | 教室 1 鑰匙 | 角色背包增加該物品 |
| ※ | 門(教室 1) | 檢查角色身上是否有開鎖條件(鑰匙) |
| 4 | 桌子(教室 1) | 角色背包增加該物品(無意義紙條 0) |
| 5 | 電視櫃(教室 1) | 檢查角色身上是否有開鎖條件(獲得螺絲起子) |
| 6 | 講桌(教室 1) | 角色背包增加該物品(鐵櫃鑰匙) |
| 7 | 鐵櫃(教室 1) | 檢查角色身上是否有開鎖條件(鐵鎚) |
| ※ | 門(教室 2) | 檢查角色身上是否有開鎖條件(鐵鎚) |
| 8 | 講桌(教室 2) | 角色背包增加該物品(老師的筆記 1)、提示訊息 |
| 9 | 無意義紙條 1(教室 2) | 角色背包增加該物品、提示訊息 |
| 10 | 佈告欄(教室 2) | 角色背包增加該物品(螺絲釘) |
| 11 | 教職員屍體(男廁) | 角色背包增加該物品(教室 4 鑰匙) |
| ※ | 門(教室 3) | 提示訊息(上鎖)P.S:只有其中一道 |
| 12 | 桌子(教室 3) | 角色背包增加該物品(無意義紙條 2) |
| 13 | 講桌(教室 3) | 角色背包增加該物品(粉筆) |
| ※ | 門(教室 4) | 檢查角色身上是否有開鎖條件(鑰匙) |
| 14 | 桌子(教室 4) | 角色背包增加該物品(無意義紙條 3) |
| 15 | 講桌(教室 4) | 角色背包增加該物品(老師的筆記 2) |
| ※ | 門(教室 5) | 檢查角色身上是否有開鎖條件(螺絲起子、螺絲釘) |
| 16 | 講桌(教室 5) | 角色背包增加該物品(板擦) |
| 17 | 無意義紙條 4(教室 5) | 角色背包增加該物品、提示訊息 |
| 18 | 牆壁刮痕(女廁) | 檢查角色身上是否有事件道具 |
| 19 | 教室 6 鑰匙(女廁) | 角色背包增加該物品 |
| ※ | 門(教室 6) | 檢查角色身上是否有開鎖條件(鑰匙) |
| 20 | 桌子(教室 6) | 角色背包增加該物品(無意義紙條 5) |
| 21 | 置物櫃(教室 6) | 角色背包增加該物品(鐵撬) |

表 4.1 中第一欄的編號對應到圖 4.12 上的編號，而編號“※”則為門的部分，因為門本身是物件所以不列入編號，則在物件欄位上的內容以物件名稱配合上後面括弧的位置來進行區別，但實際上這些物件都可以重複，而最後一欄的事件為，主角調查物件時所發生事情。

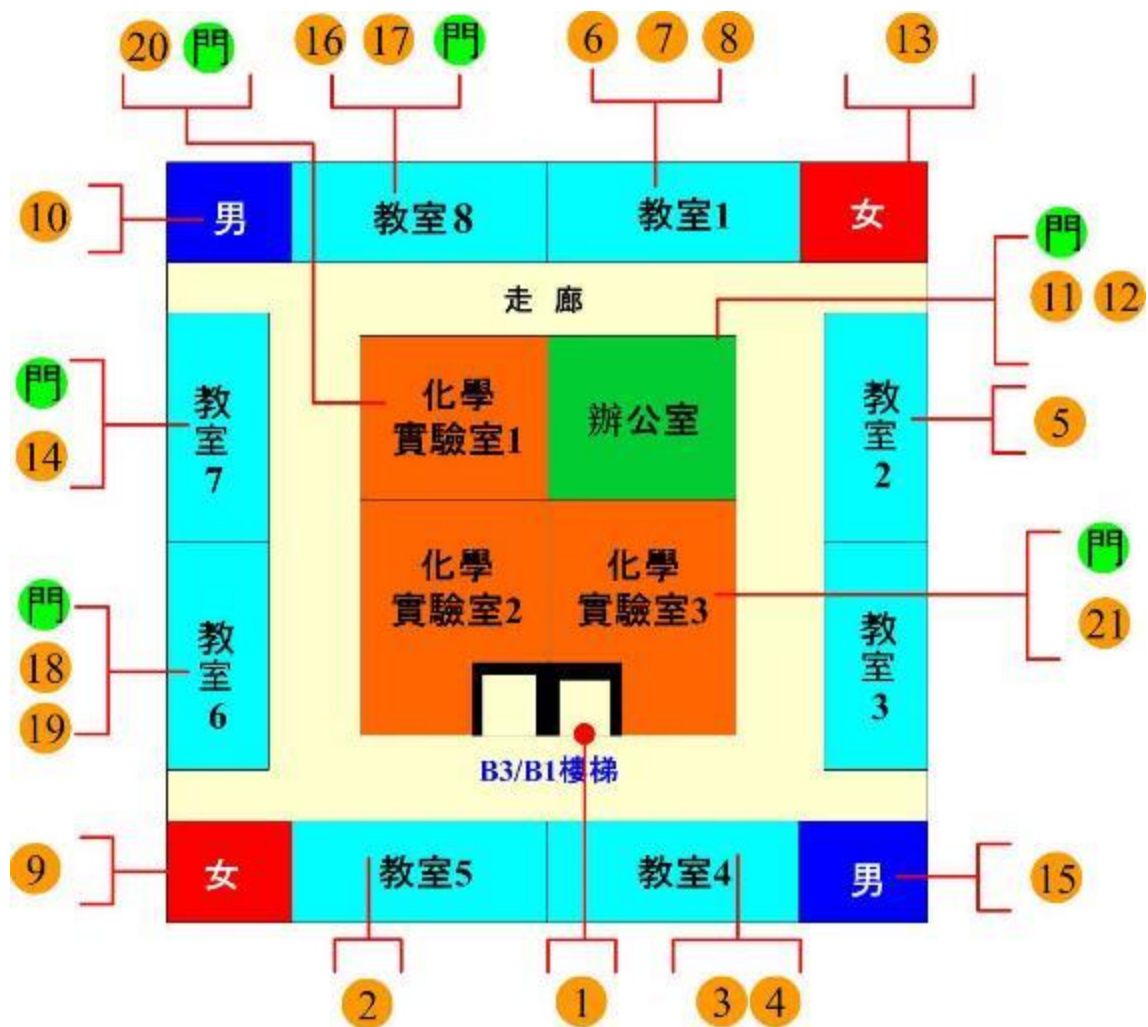


圖 4.13 物件分析圖 B2

表 4.2 物件分析表 B2

| 編號 | 物件 | 事件 |
|----|----------------|-----------------------------|
| 1 | 桌椅堆 | 檢查角色身上是否有開鎖條件(打火機、酒精燈) |
| 2 | 桌子(教室 5) | 角色背包增加該物品(無意義紙條 6) |
| 3 | 講桌(教室 4) | 角色背包增加該物品(老師的筆記 3) |
| 4 | 置物櫃(教室 4) | 角色背包增加該物品(斷掉的木竿) |
| 5 | 講桌(教室 2) | 角色背包增加該物品(老師的筆記 4) |
| 6 | 桌子(教室 1) | 角色背包增加該物品(無意義紙條 7) |
| 7 | 置物櫃(教室 1) | 角色背包增加該物品(沒有竿子的掃把) |
| 8 | 相框(教室 1) | 角色背包增加該物品(剪刀) |
| 9 | 女屍體(左下女廁) | 檢查角色身上是否有事件道具(獲得鑰匙 8)、移動、攻擊 |
| 10 | 小便斗(左上男廁) | 角色背包增加該物品(辦公室鑰匙) |
| ※ | 門(辦公室) | 檢查角色身上是否有開鎖條件(鑰匙) |
| 11 | 辦公桌(辦公室) | 角色背包增加該物品(老師的筆記 5、打火機) |
| 12 | 飲水機(辦公室) | 角色背包增加該物品(紅色鋼筆) |
| 13 | 屍體(右上女廁) | 角色背包增加該物品(教室 7 鑰匙) |
| ※ | 門(教室 7) | 檢查角色身上是否有開鎖條件(鑰匙) |
| 14 | 桌子(教室 7) | 角色背包增加該物品(無意義紙條 8、手機電池) |
| 15 | 手機(右下男廁) | 檢查角色身上是否有事件道具(手機電池) |
| ※ | 門(教室 8) | 檢查角色身上是否有開鎖條件(鑰匙) |
| 16 | 桌子(教室 8) | 角色背包增加該物品(無意義紙條 9) |
| 17 | 不知名的鑰匙(教室 8) | 角色背包增加該物品 |
| ※ | 門(教室 6) | 檢查角色身上是否有開鎖條件(不知名的鑰匙) |
| 18 | 化學鑰匙 1~3(教室 6) | 角色背包增加該物品 |
| 19 | 桌子(教室 6) | 提示訊息、角色背包增加該物品(無意義紙條 11) |
| ※ | 門(化學教室 1) | 檢查角色身上是否有開鎖條件(鑰匙) |
| 20 | 置物櫃(化學教室 1) | 角色背包增加該物品(酒精燈) |
| ※ | 門(化學教室 3) | 檢查角色身上是否有開鎖條件(鑰匙) |
| 21 | 置物櫃(化學教室 3) | 角色背包增加該物品(酒精燈) |

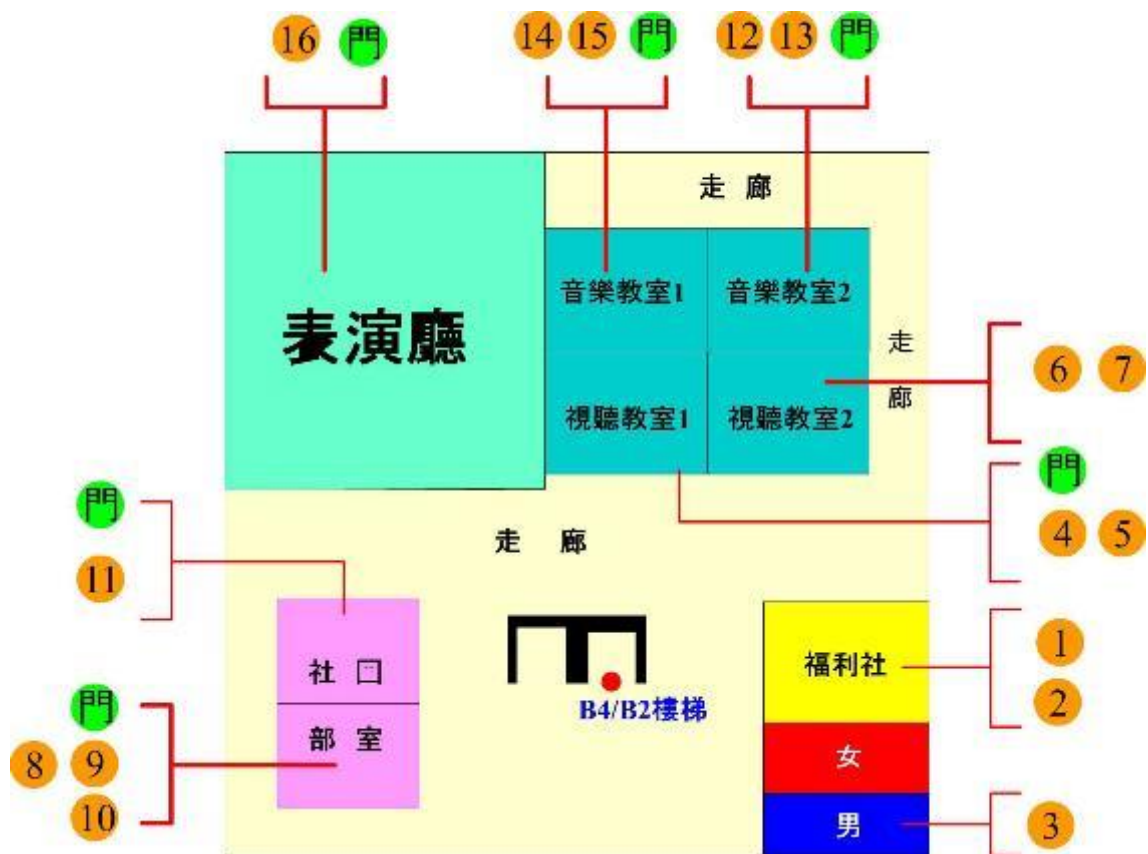


圖 4.14 物件分析圖 B3

表 4.3 物件分析表 B3

| 編號 | 物件 | 事件 |
|----|-------------------|-------------------|
| 1 | 視聽教室 1 鑰匙(福利社) | 角色背包增加該物品 |
| 2 | 電池(福利社) | 角色背包增加該物品 |
| 3 | 鬧鐘(男廁) | 檢查角色身上是否有事件道具 |
| ※ | 門(視聽教室 1) | 檢查角色身上是否有開鎖條件(鑰匙) |
| 4 | 探險學生的遺書 1(視聽教室 1) | 角色背包增加該物品 |
| 5 | 電腦(視聽教室 1) | 角色背包增加該物品(開機磁片) |
| 6 | 社團部室鑰匙(視聽教室 2) | 角色背包增加該物品 |
| 7 | 無意義紙條 11(視聽教室 2) | 角色背包增加該物品 |
| ※ | 門(社團部室) | 檢查角色身上是否有開鎖條件(鑰匙) |
| 8 | 探險學生的遺書 2(社團部室) | 角色背包增加該物品 |
| 9 | 社團倉庫鑰匙(社團部室) | 角色背包增加該物品 |
| 10 | 音樂教室 2 鑰匙(社團部室) | 角色背包增加該物品 |
| ※ | 門(社團倉庫) | 檢查角色身上是否有開鎖條件(鑰匙) |
| 11 | 音響連接線(社團倉庫) | 角色背包增加該物品 |
| ※ | 門(音樂教室 2) | 檢查角色身上是否有開鎖條件(鑰匙) |
| 12 | 音樂教室 1 鑰匙(音樂教室 2) | 角色背包增加該物品 |
| 13 | 探險學生的遺書 3(音樂教室 2) | 角色背包增加該物品 |
| ※ | 門(音樂教室 1) | 檢查角色身上是否有開鎖條件(鑰匙) |
| 14 | 音樂光碟(音樂教室 1) | 角色背包增加該物品 |
| 15 | 表演廳鑰匙(音樂教室 1) | 角色背包增加該物品 |
| ※ | 門(表演廳) | 檢查角色身上是否有開鎖條件(鑰匙) |
| 16 | 電腦(表演廳) | 檢查角色身上是否有事件道具 |

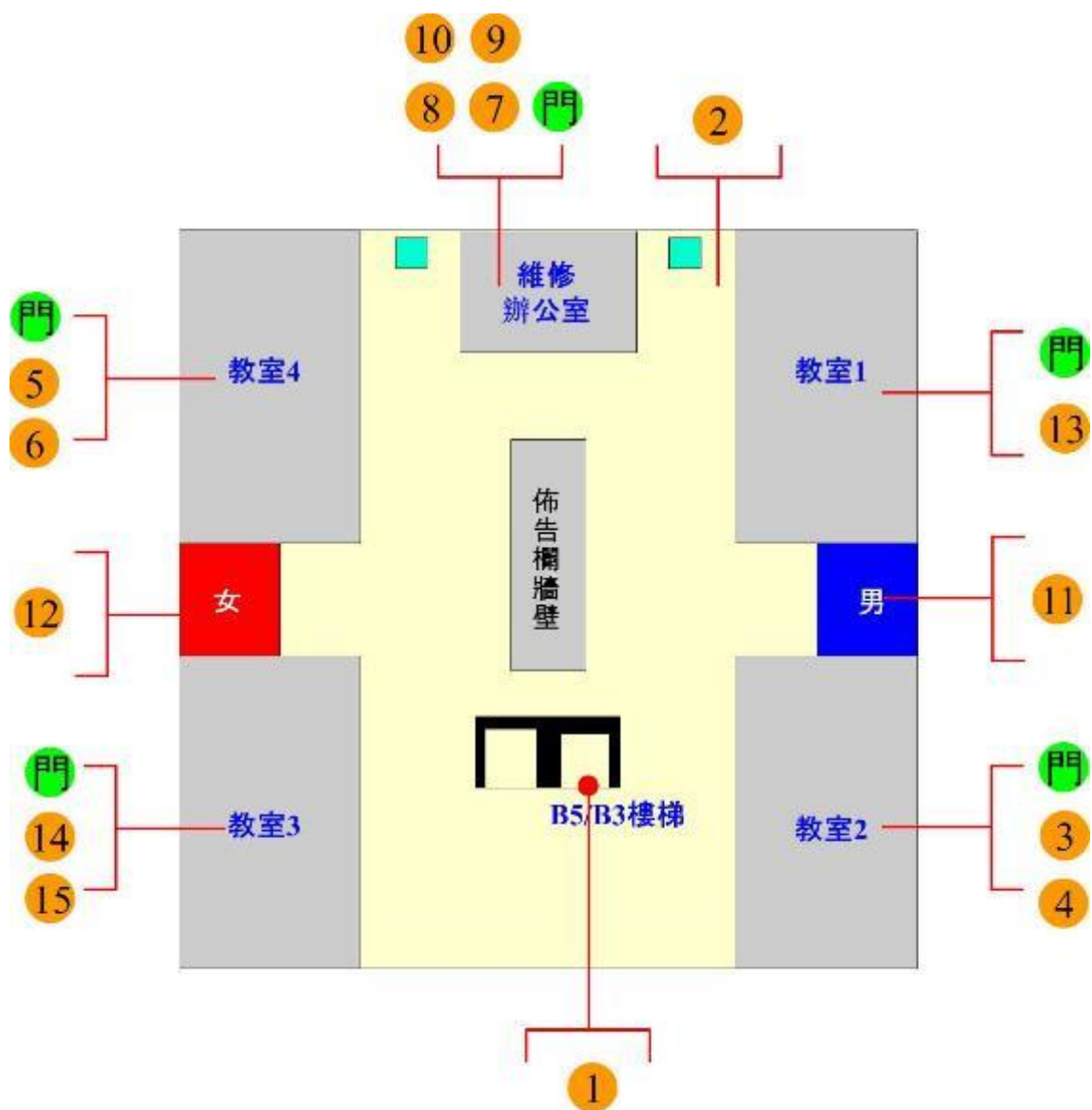


圖 4.15 物件分析圖 B4

表 4.4 物件分析表 B4

| 編號 | 物件 | 事件 |
|----|----------------|----------------------------|
| 1 | 鐵門 | 檢查是否滿足過關條件(電源箱啟動) |
| 2 | 磁卡 2(飲水機旁) | 角色背包增加該物品 |
| ※ | 門(教室 2) | 檢查角色身上是否有開鎖條件(磁卡) |
| 3 | 磁卡 4(教室 2) | 角色背包增加該物品 |
| 4 | 緊急供電系統 2(教室 2) | 提示訊息(滿足過關條件 1/4) |
| ※ | 門(教室 4) | 檢查角色身上是否有開鎖條件(磁卡) |
| 5 | 維修辦公室磁卡(教室 4) | 角色背包增加該物品 |
| 6 | 緊急供電系統 4(教室 4) | 提示訊息(滿足過關條件 2/4) |
| ※ | 門(維修辦公室) | 檢查角色身上是否有開鎖條件(磁卡) |
| 7 | 磁卡 3(維修辦公室) | 角色背包增加該物品 |
| 8 | 磁卡 1(維修辦公室) | 角色背包增加該物品 |
| 9 | 老師的筆記 6(維修辦公室) | 角色背包增加該物品 |
| 10 | 總電源箱(維修辦公室) | 檢查角色身上是否有開鎖條件(鑰匙)、提示訊息(過關) |
| 11 | 鬧鐘(男廁) | 角色背包增加該物品、定時設定、血量 |
| 12 | 鬧鐘(女廁) | 角色背包增加該物品、定時設定、血量 |
| ※ | 門(教室 1) | 檢查角色身上是否有開鎖條件(磁卡) |
| 13 | 緊急供電系統 1(教室 1) | 提示訊息(滿足過關條件 3/4) |
| ※ | 門(教室 3) | 檢查角色身上是否有開鎖條件(磁卡) |
| 14 | 緊急供電系統 3(教室 3) | 提示訊息(滿足過關條件 4/4) |
| 15 | 電源箱鑰匙(教室 3) | 角色背包增加該物品 |

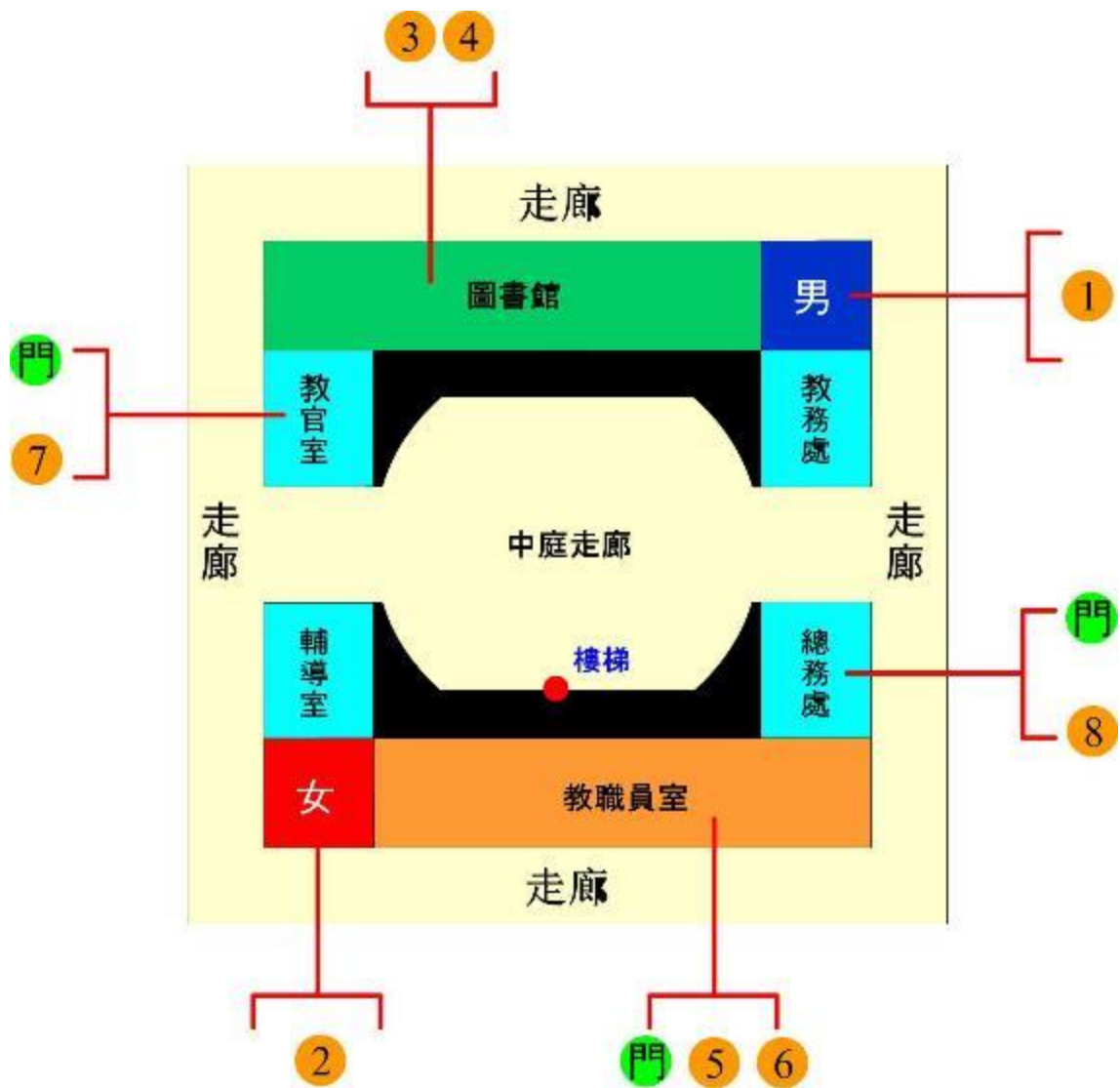


圖 4.16 物件分析圖 B5

表 4.5 物件分析表 B5

| 編號 | 物件 | 事件 |
|----|---------------|--------------------------|
| 1 | 輔導室鑰匙(男廁) | 角色背包增加該物品 |
| 2 | 教務處鑰匙(女廁) | 角色背包增加該物品 |
| 3 | 教職員室鑰匙(圖書館) | 角色背包增加該物品 |
| 4 | 老師的筆記 7(圖書館) | 角色背包增加該物品 |
| ※ | 門(教職員室) | 檢查角色身上是否有開鎖條件(鑰匙) |
| 5 | 教官室鑰匙(教職員室) | 角色背包增加該物品 |
| 6 | 老師的筆記 8(教職員室) | 角色背包增加該物品 |
| ※ | 門(教官室) | 檢查角色身上是否有開鎖條件(鑰匙) |
| 7 | 鐵槌(教官室) | 角色背包增加該物品 |
| ※ | 門(總務處) | 檢查角色身上是否有開鎖條件(鐵槌) |
| 8 | 頂樓鑰匙(總務處) | 角色背包增加該物品、提示訊息(引導玩家回到樓梯) |


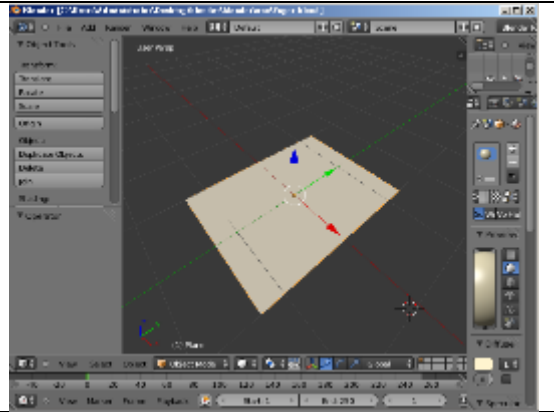
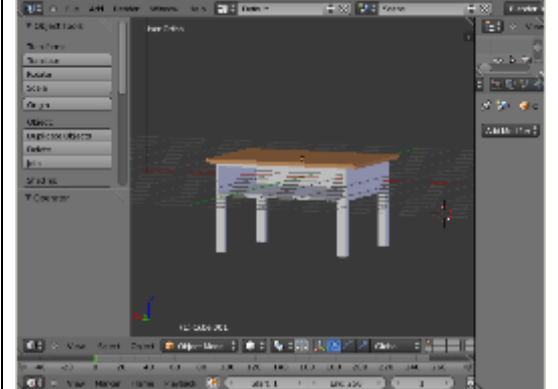
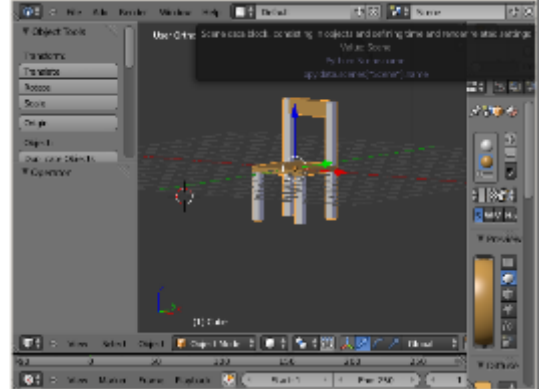
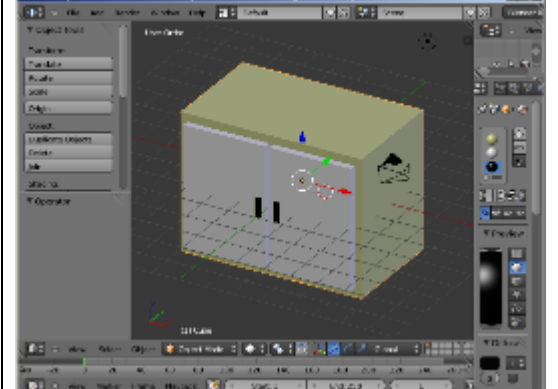
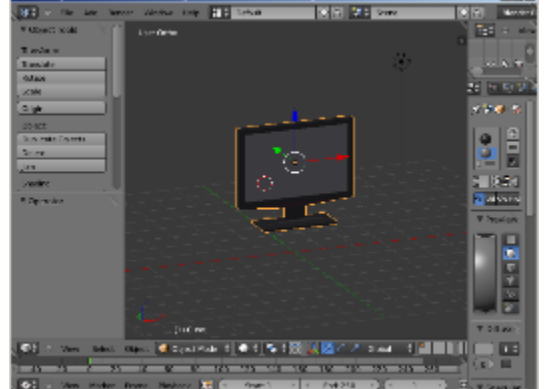
從劇情裡開始物件分析，主要可以拆成物件與事件兩大類，其中以這資料當中，整理出建模清單。

第五章 遊戲製作

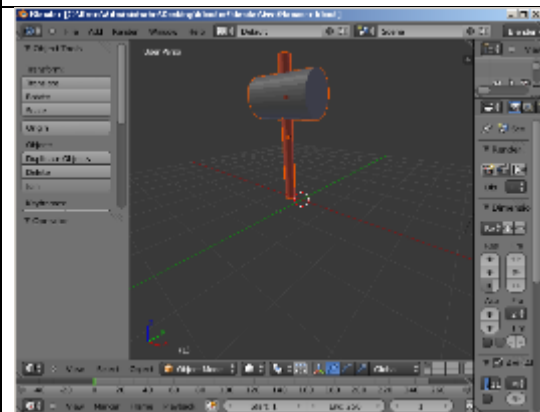
5.1 建模清單

將物件分析中可以建模的物品、角色與場景等，運用 Blender 建模，以一格名稱一格圖片呈現，如下表 5.1。

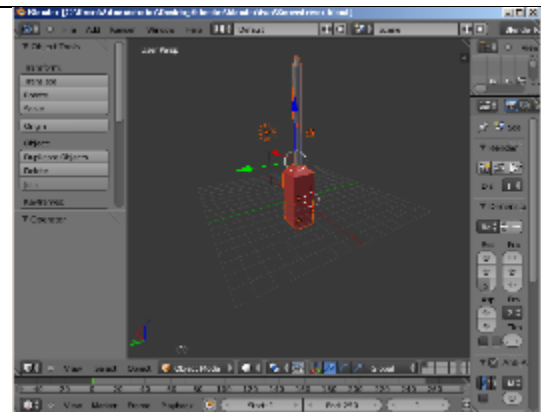
表 5.1 建模清單

| | |
|--|--|
| <p>鑰匙</p>  | <p>紙</p>  |
| <p>桌</p>  | <p>椅</p>  |
| <p>電視櫃</p>  | <p>電視</p>  |

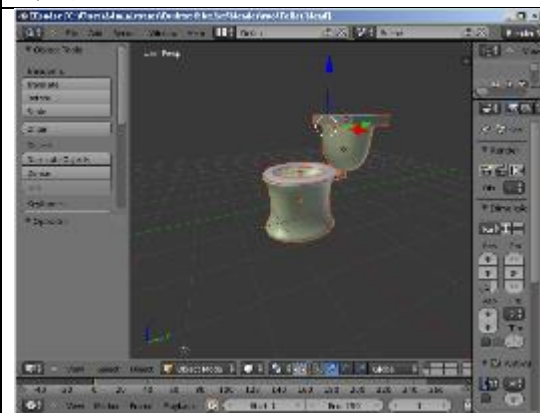
鐵鎚



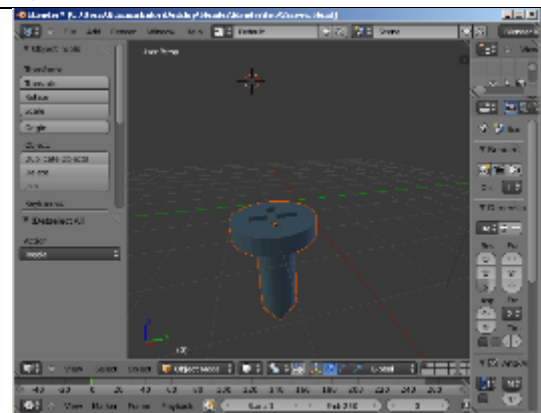
螺絲起子



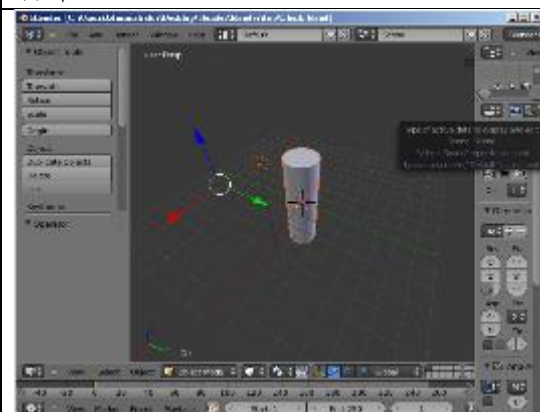
馬桶



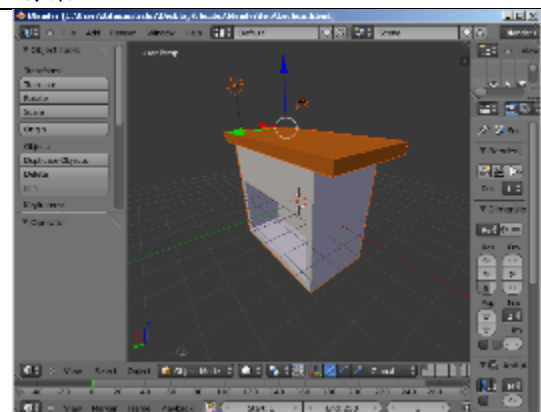
螺絲釘



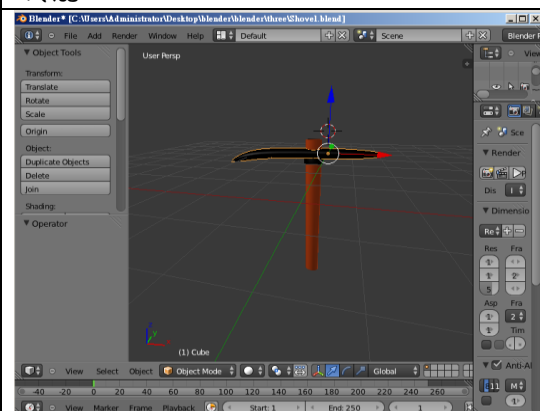
粉筆



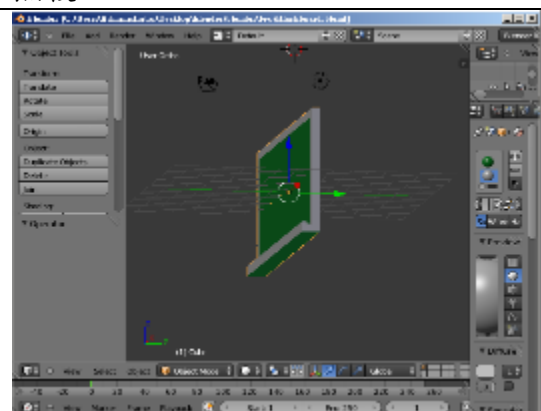
講桌



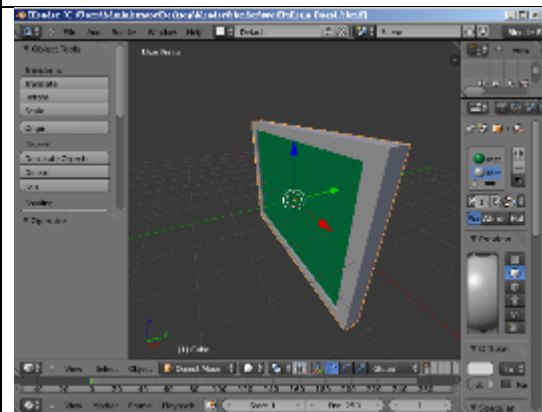
鐵撬



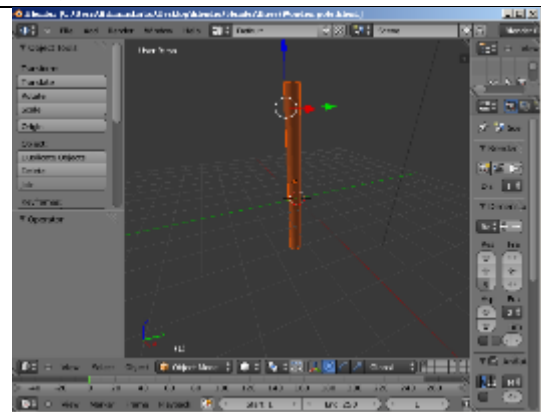
黑板



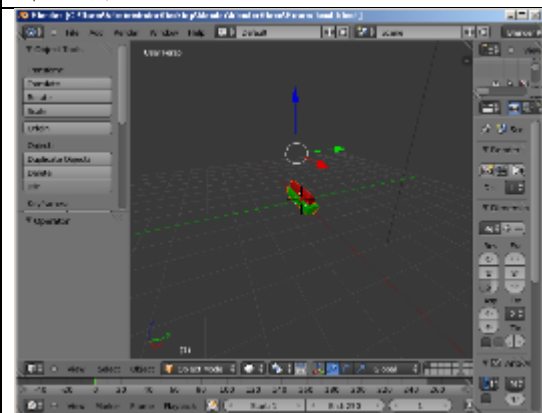
佈告欄



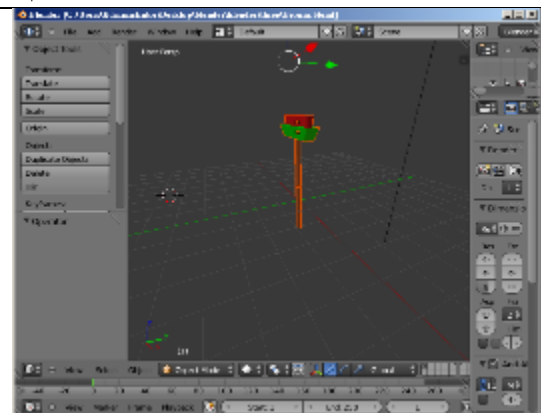
木竿



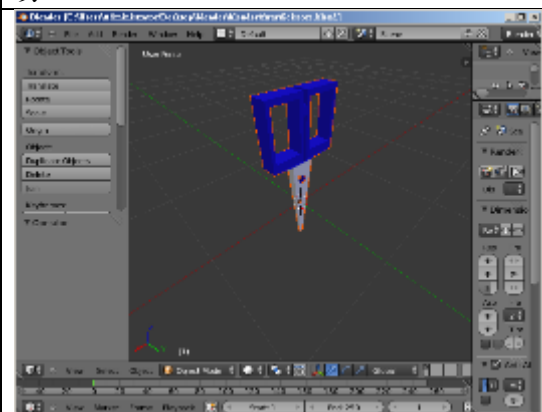
掃把頭



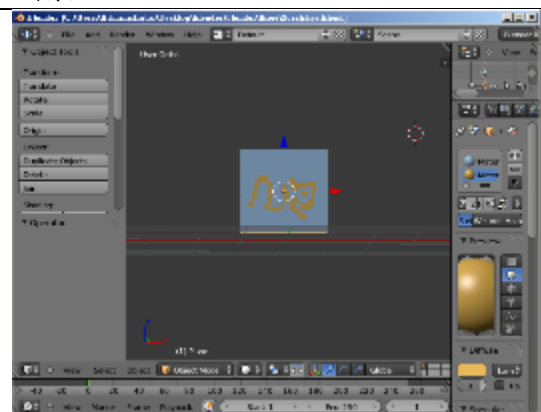
掃把



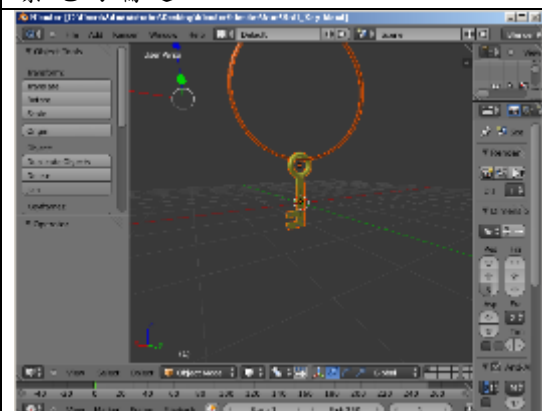
剪刀



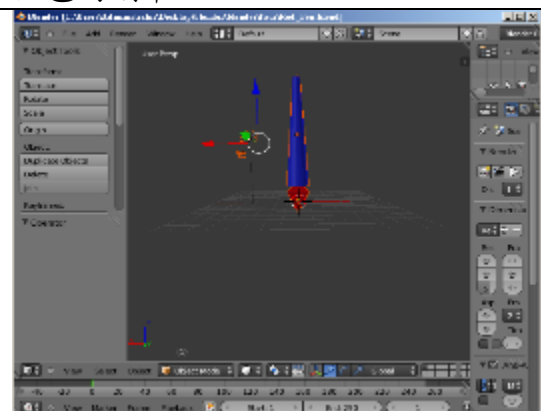
划痕



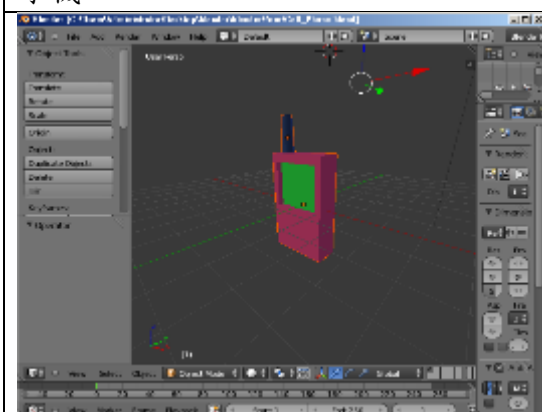
繫繩的鑰匙



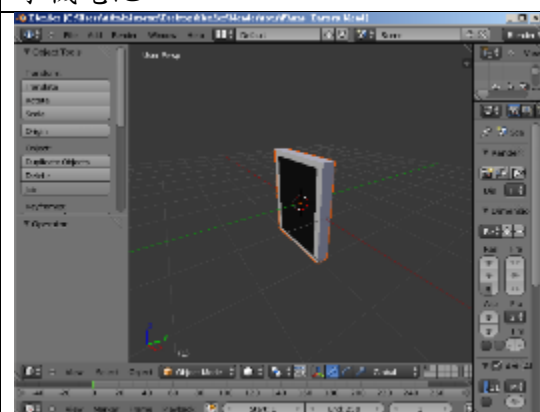
紅色的鋼筆



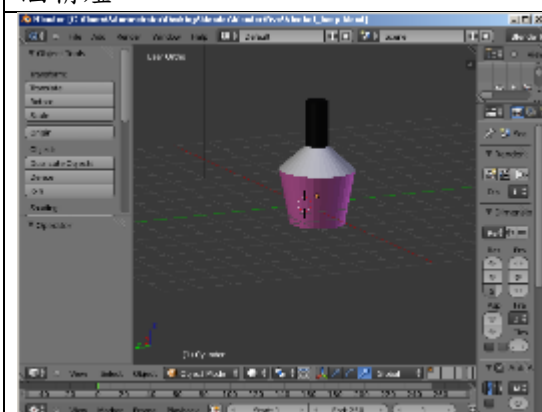
手機



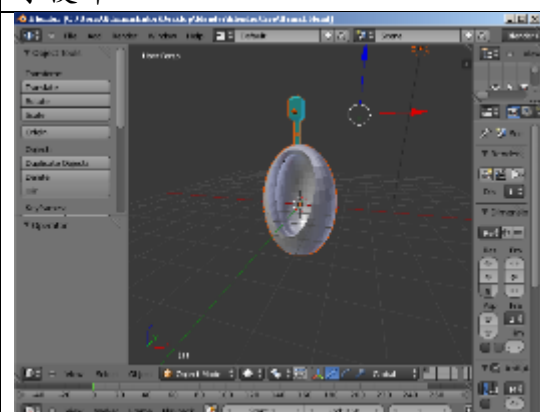
手機電池



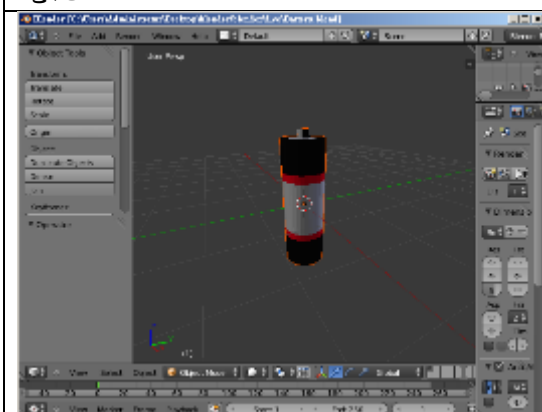
酒精燈



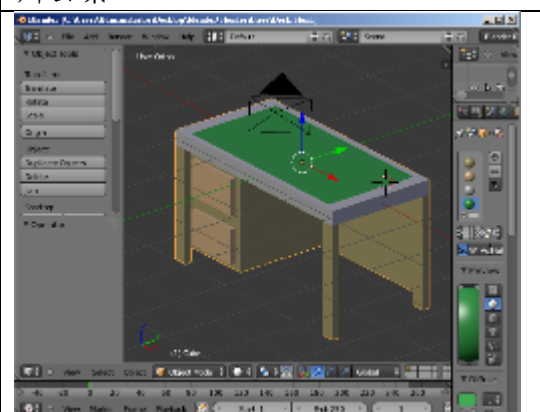
小便斗



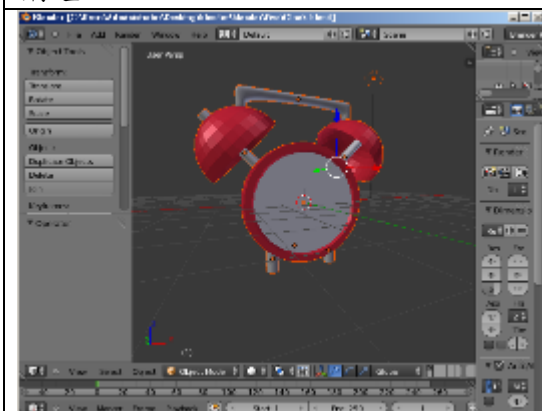
電池



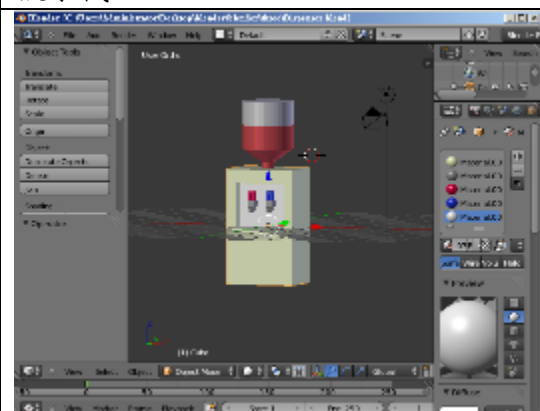
辦公桌



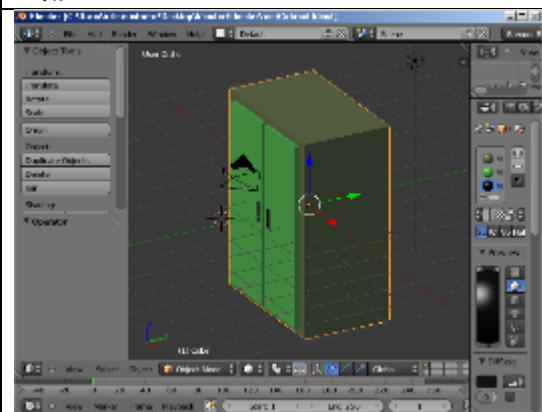
鬧鐘



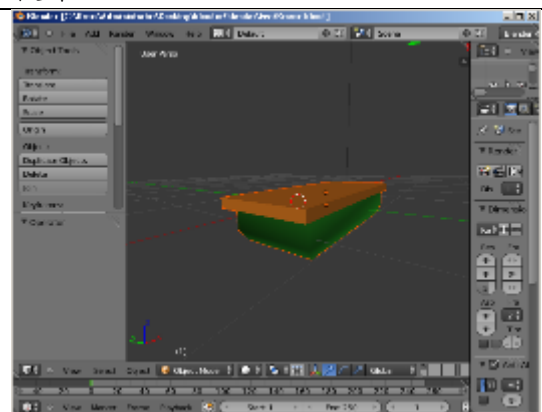
飲水機



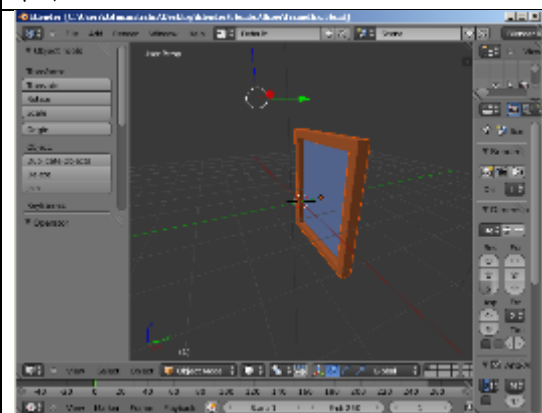
鐵櫃



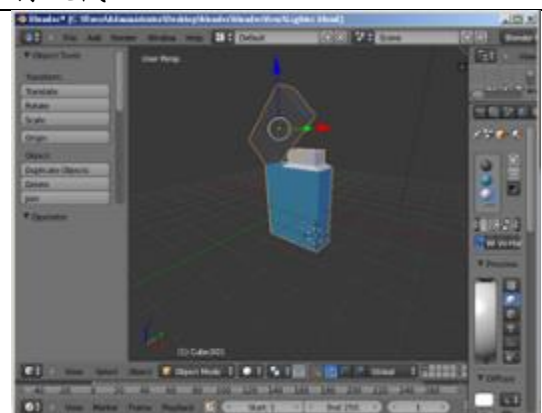
板擦



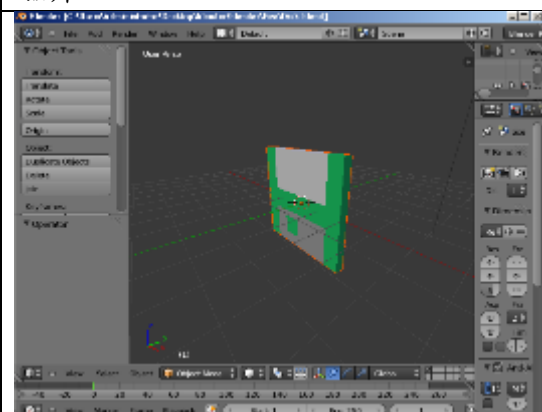
相框



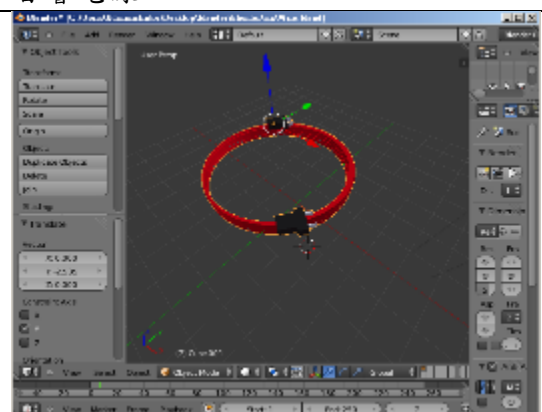
打火機



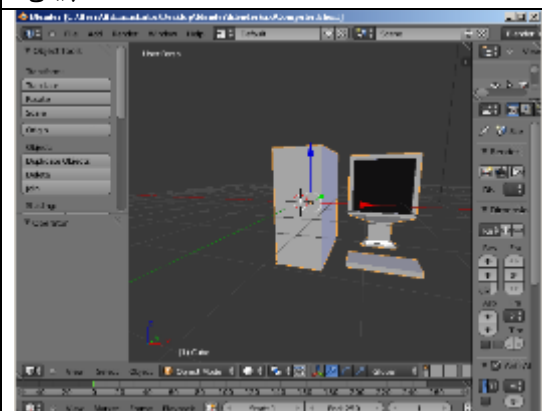
磁片



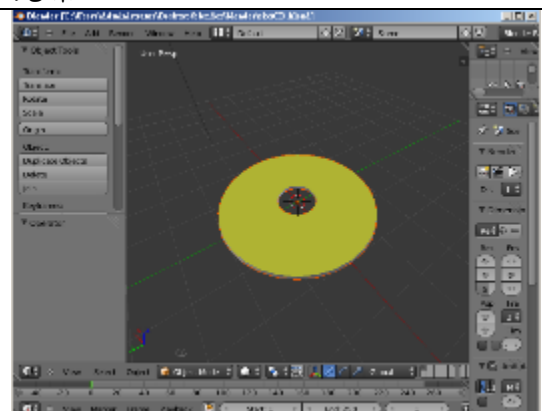
音響電線



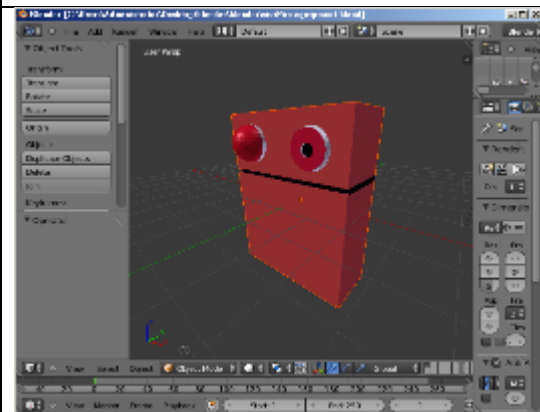
電腦



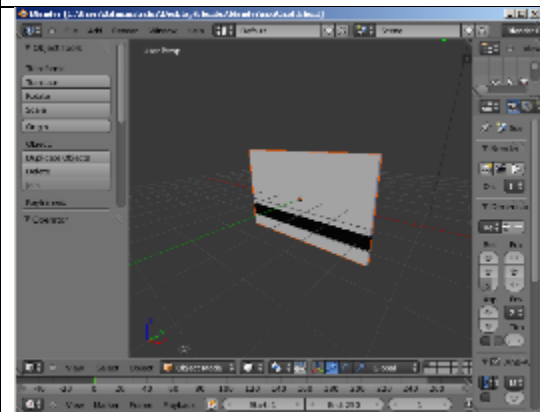
光碟



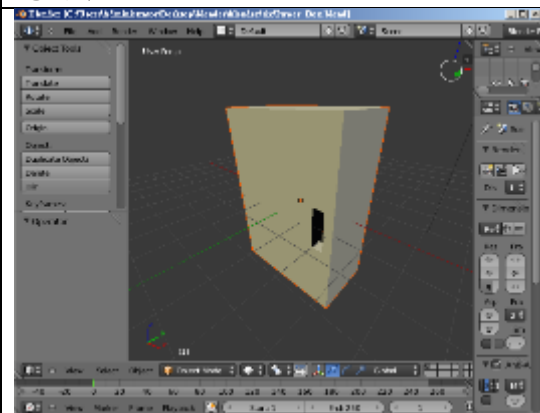
消防設備



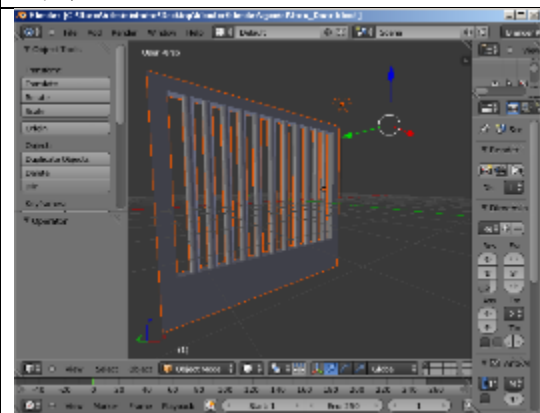
磁卡



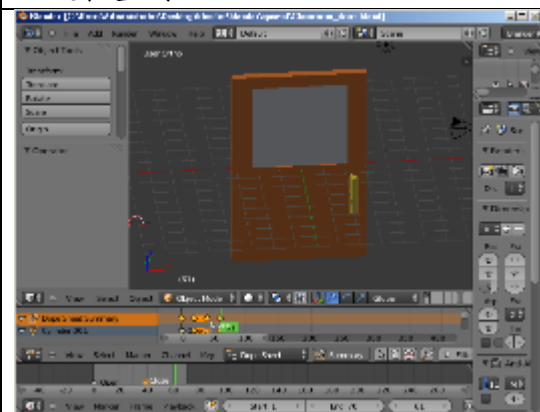
電源箱



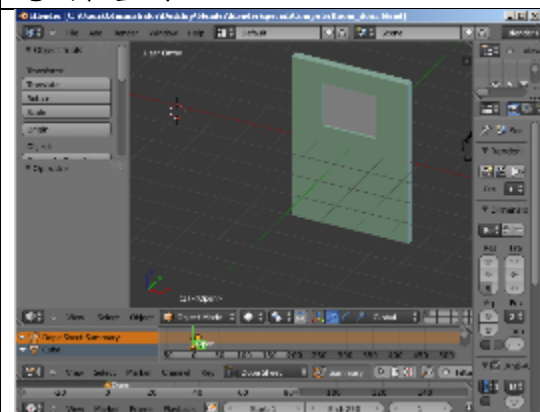
鐵門



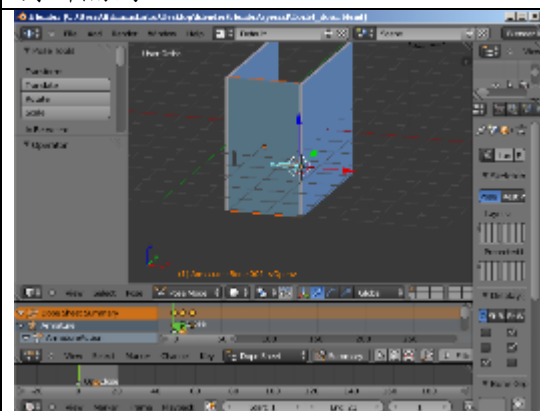
一般教室門



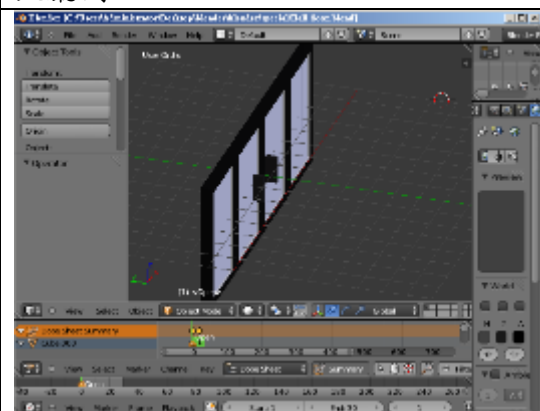
電腦教室門

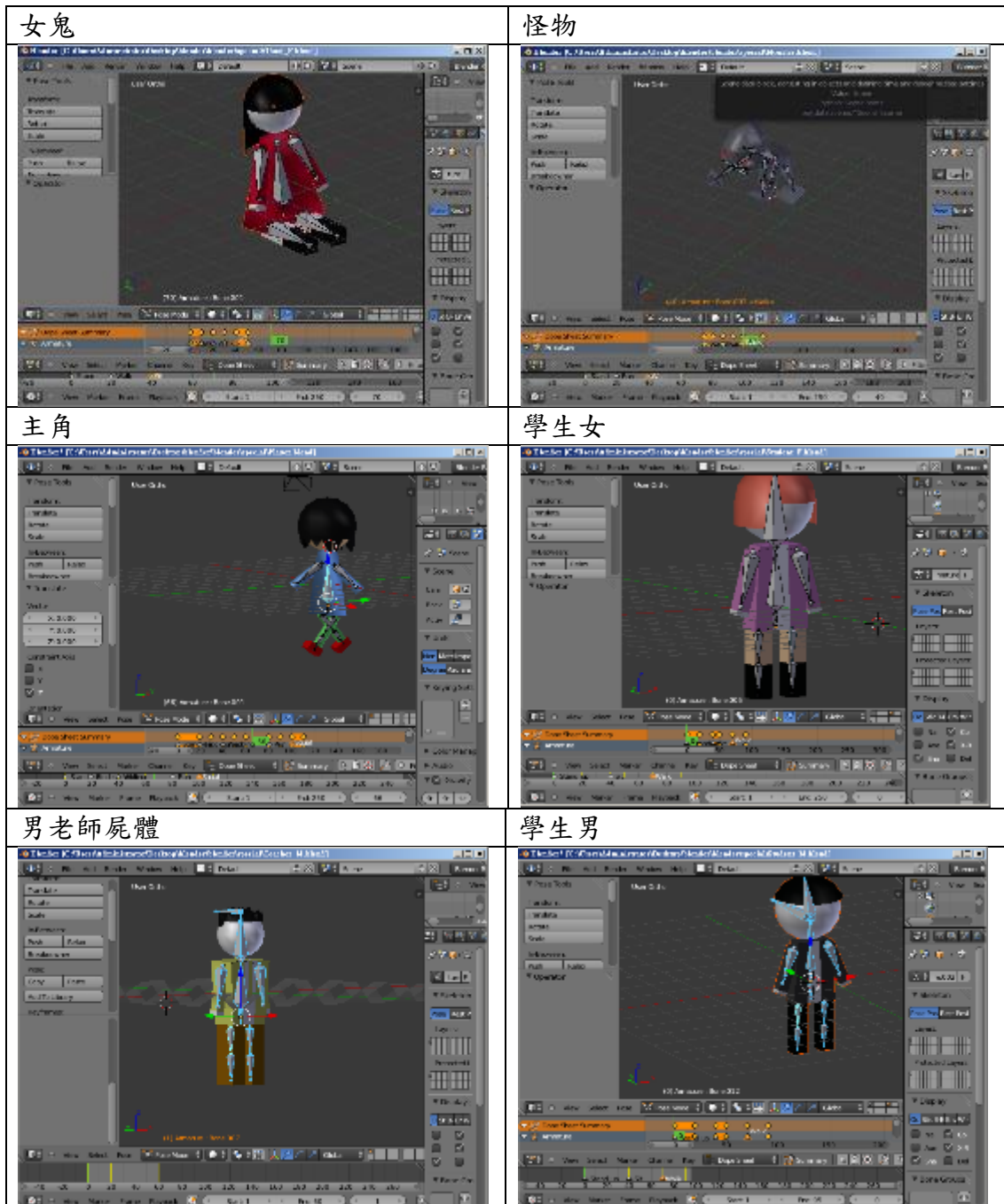


廁所隔間



大廳門口





5.2 撰寫腳本

所謂的撰寫腳本就是將程式語言寫進遊戲引擎，藉此操控物件，將功能列表分工，表 5.2 中”X”為非樓層物件，”B1”為地下一樓，以此類推，第二欄為物件名稱，第三格為功能。

表 5.2 撰寫腳本

| 樓層 | 名稱 | 功能 |
|----|----|--------------------------|
| X | 玩家 | 移動、跑步、蹲下、調查、使用者介面(背包、存檔) |

| | | |
|----|--------------------------------|--|
| X | 鬼魂 怪物 | 移動、攻擊 |
| X | 門 | 開、關 |
| X | 道具 | 背包新增該物品、提示訊息 |
| B1 | 1 鐵門 Iron_Door | 檢查角色身上是否有開鎖條件 |
| B1 | 2 說明書 Paper | 角色背包增加該物品 提示訊息 |
| B1 | 3 教室 1 鑰匙 Key | 角色背包增加該物品 |
| B1 | ※ 門(教室 1) Classroom_door | 檢查角色身上是否有開鎖條件(鑰匙) Key |
| B1 | 4 桌子(教室 1) Table | 角色背包增加該物品(無意義紙條 0) Paper |
| B1 | 5 電視櫃(教室 1) TV_cabinet | 檢查角色身上是否有開鎖條件(獲得 螺絲起子) Screwdriver |
| B1 | 6 講桌(教室 1) Lectern | 角色背包增加該物品(電視櫃鑰匙) Key |
| B1 | 7 置物櫃(教室 1) Cabinet | 角色背包增加該物品(鐵鎚) Hammer |
| B1 | ※ 門(教室 2) Classroom_door | 檢查角色身上是否有開鎖條件(鐵鎚) Hammer |
| B1 | 8 講桌(教室 2) Lectern | 角色背包增加該物品(老師的筆記 1) 提示訊息 Paper |
| B1 | 9 無意義紙條 1(教室 2) Paper | 角色背包增加該物品 提示訊息 |
| B1 | 10 佈告欄(教室 2) Bulletin Board | 角色背包增加該物品(螺絲釘) Screws |
| B1 | 11 教職員屍體(男廁) Teacher_M | 角色背包增加該物品(教室 4 鑰匙) Key |
| B1 | ※ 門(教室 3) Classroom_door | 提示訊息(上鎖) P.S:只有其中一道 |
| B1 | 12 桌子(教室 3) Table | 角色背包增加該物品(無意義紙條 2) Paper |
| B1 | 13 講桌(教室 3) Lectern | 角色背包增加該物品(粉筆) Chalk |
| B1 | ※ 門(教室 4) Classroom_door | 檢查角色身上是否有開鎖條件(鑰匙) Key |
| B1 | 14 桌子(教室 4) Table | 角色背包增加該物品(無意義紙條 3) Paper |
| B1 | 15 講桌(教室 4) Lectern | 角色背包增加該物品(老師的筆記 2) Paper |
| B1 | ※ 門(教室 5) Classroom_door | 檢查角色身上是否有開鎖條件(螺絲 起子、螺絲) Screwdriver、Screws |

| | | |
|----|-----------------------------|--|
| B1 | 16 講桌(教室 5) Lectern | 角色背包增加該物品(板擦) |
| B1 | 17 無意義紙條 4(教室 5) Paper | 角色背包增加該物品 提示訊息 |
| B1 | 18 牆壁刮痕(女廁) Scratches | 檢查角色身上是否有事件道具 |
| B1 | 19 教室 6 鑰匙(女廁) Key | 角色背包增加該物品 |
| B1 | ※ 門(教室 6) Classroom_door | 檢查角色身上是否有開鎖條件(鑰匙) Key |
| B1 | 20 桌子(教室 6) Table | 角色背包增加該物品(無意義紙條 5) Paper |
| B1 | 21 置物櫃(教室 6) Cabinet | 角色背包增加該物品(鐵撬) Shovel |
| B2 | 1 桌椅堆 Table、Chair | 檢查角色身上是否有開鎖條件(打火機、酒精燈) Lighter、Alcohol_lamp |
| B2 | 2 桌子(教室 5) Table | 角色背包增加該物品(無意義紙條 6) Paper |
| B2 | 3 講桌(教室 4) Lectern | 角色背包增加該物品(老師的筆記 3) Paper |
| B2 | 4 置物櫃(教室 4) Cabinet | 角色背包增加該物品(斷掉的木竿) |
| B2 | 5 講桌(教室 2) Lectern | 角色背包增加該物品(老師的筆記 4) Paper |
| B2 | 6 桌子(教室 1) Table | 角色背包增加該物品(無意義紙條 7) Paper |
| B2 | 7 置物櫃(教室 1) Cabinet | 角色背包增加該物品(沒有竿子的掃把) Wooden pole |
| B2 | 8 相框(教室 1) framefbx | 角色背包增加該物品(剪刀) Scissors |
| B2 | 9 女屍體(左下女廁) Ghost_F | 檢查角色身上是否有事件道具(獲得鑰匙 8) Roll_Key 移動 攻擊 |
| B2 | 10 小便斗(左上男廁) Urinal | 角色背包增加該物品(辦公室鑰匙) Key |
| B2 | ※ 門(辦公室) Classroom_door | 檢查角色身上是否有開鎖條件(鑰匙) Key |
| B2 | 11 辦公桌(辦公室) Desk | 角色背包增加該物品(老師的筆記 5、打火機) Paper、Lighter |
| B2 | 12 飲水機(辦公室) Dispenser | 角色背包增加該物品(紅色鋼筆) Red_Pen |
| B2 | 13 屍體(右上女廁) Student_F | 角色背包增加該物品(教室 7 鑰匙) Key |

| | | |
|----|-------------------------------|--|
| B2 | ※ 門(教室 7) Classroom_door | 檢查角色身上是否有開鎖條件(鑰匙) Key |
| B2 | 14 桌子(教室 7) Table | 角色背包增加該物品(無意義紙條 8、 手機電池) Paper、Phone_Battery |
| B2 | 15 手機(右下男廁) Cell_Phone | 檢查角色身上是否有事件道具(手機 電池) Phone_Battery |
| B2 | ※ 門(教室 8) Classroom_door | 檢查角色身上是否有開鎖條件(鑰匙) Key |
| B2 | 16 桌子(教室 8) Table | 角色背包增加該物品(無意義紙條 9) Paper |
| B2 | 17 不知名的鑰匙(教室 8) Key | 角色背包增加該物品 |
| B2 | ※ 門(教室 6) Classroom_door | 檢查角色身上是否有開鎖條件(不知 名的鑰匙) Key |
| B2 | 18 化學鑰匙 1~3(教室 6) Key | 角色背包增加該物品 |
| B2 | 19 桌子(教室 6) Table | 提示訊息 角色背包增加該物品(無意義紙條 11) Paper |
| B2 | ※ 門(化學教室 1) Classroom_door | 檢查角色身上是否有開鎖條件(鑰匙) Key |
| B2 | 20 置物櫃(化學教室 1) Cabinet | 角色背包增加該物品(酒精燈) Alcohol_lamp |
| B2 | ※ 門(化學教室 3) Classroom_door | 檢查角色身上是否有開鎖條件(鑰匙) Key |
| B2 | 21 置物櫃(化學教室 3) Cabinet | 角色背包增加該物品(酒精燈) Alcohol_lamp |
| B3 | 1 視聽教室 1 鑰匙(福利社) Key | 角色背包增加該物品 |
| B3 | 2 電池(福利社) Battery | 角色背包增加該物品 |
| B3 | 3 鬧鐘(男廁) Clock | 檢查角色身上是否有事件道具 |
| B3 | ※ 門(視聽教室 1) Classroom_door | 檢查角色身上是否有開鎖條件(鑰匙) Key |
| B3 | 4 探險學生的遺書 1(視聽教室 1) Paper | 角色背包增加該物品 |
| B3 | 5 電腦(視聽教室 1) Computer | 角色背包增加該物品(開機磁片) Disk |
| B3 | 6 社團部室鑰匙(視聽教室 2) Key | 角色背包增加該物品 |
| B3 | 7 無意義紙條 11(視聽教室 2) Paper | 角色背包增加該物品 |
| B3 | ※ 門(社團部室) Classroom_door | 檢查角色身上是否有開鎖條件(鑰匙) Key |

| | | |
|----|---------------------------------|--|
| B3 | 8 探險學生的遺書 2(社團部室) Paper | 角色背包增加該物品 |
| B3 | 9 社團倉庫鑰匙(社團部室) Key | 角色背包增加該物品 |
| B3 | 10 音樂教室 2 鑰匙(社團部室) Key | 角色背包增加該物品 |
| B3 | ※ 門(社團倉庫) Classroom_door | 檢查角色身上是否有開鎖條件(鑰匙) Key |
| B3 | 11 音響連接線(社團倉庫) Wire | 角色背包增加該物品 |
| B3 | ※ 門(音樂教室 2) Classroom_door | 檢查角色身上是否有開鎖條件(鑰匙) Key |
| B3 | 12 音樂教室 1 鑰匙(音樂教室 2) Key | 角色背包增加該物品 |
| B3 | 13 探險學生的遺書 3(音樂教室 2) Paper | 角色背包增加該物品 |
| B3 | ※ 門(音樂教室 1) Classroom_door | 檢查角色身上是否有開鎖條件(鑰匙) Key |
| B3 | 14 音樂光碟(音樂教室 1) CD | 角色背包增加該物品 |
| B3 | 15 表演廳鑰匙(音樂教室 1) Key | 角色背包增加該物品 |
| B3 | ※ 門(表演廳) Hall_door | 檢查角色身上是否有開鎖條件(鑰匙) Key |
| B3 | 16 電腦(表演廳) Computer | 檢查角色身上是否有事件道具(開機 磁片、音樂光碟、音響連接線) Disk、CD、Wire |
| B4 | 1 鐵門 Iron_Door | 檢查是否滿足過關條件(電源箱啟動) Power_Box |
| B4 | 2 磁卡 2(飲水機旁) Card | 角色背包增加該物品 |
| B4 | ※ 門(教室 2) ComputerRoom_door | 檢查角色身上是否有開鎖條件(磁卡) Card |
| B4 | 3 磁卡 4(教室 2) Card | 角色背包增加該物品 |
| B4 | 4 緊急供電系統 2(教室 2) Power_Box | 提示訊息(滿足過關條件 1/4) |
| B4 | ※ 門(教室 4) ComputerRoom_door | 檢查角色身上是否有開鎖條件(磁卡) Card |
| B4 | 5 維修辦公室磁卡(教室 4) Card | 角色背包增加該物品 |
| B4 | 6 緊急供電系統 4(教室 4) Power_Box | 提示訊息(滿足過關條件 2/4) |
| B4 | ※ 門(維修辦公室) ComputerRoom_door | 檢查角色身上是否有開鎖條件(磁卡) Card |
| B4 | 7 磁卡 3(維修辦公室) Card | 角色背包增加該物品 |
| B4 | 8 磁卡 1(維修辦公室) | 角色背包增加該物品 |

| | | |
|----|--------------------------------|--------------------------------------|
| | Card | |
| B4 | 9 老師的筆記 6(維修辦公室) Paper | 角色背包增加該物品 |
| B4 | 10 總電源箱(維修辦公室) Power_Box | 檢查角色身上是否有開鎖條件(鑰匙) 提示訊息(過關) Key |
| B4 | 11 鬧鐘(男廁) Clock | 角色背包增加該物品 定時設定 血量 |
| B4 | 12 鬧鐘(女廁) Clock | 角色背包增加該物品 定時設定 血量 |
| B4 | ※ 門(教室 1) ComputerRoom_door | 檢查角色身上是否有開鎖條件(磁卡) Card |
| B4 | 13 緊急供電系統 1(教室 1) Power_Box | 提示訊息(滿足過關條件 3/4) |
| B4 | ※ 門(教室 3) ComputerRoom_door | 檢查角色身上是否有開鎖條件(磁卡) Card |
| B4 | 14 緊急供電系統 3(教室 3) Power_Box | 提示訊息(滿足過關條件 4/4) |
| B4 | 15 電源箱鑰匙(教室 3) Key | 角色背包增加該物品 |
| B5 | 1 輔導室鑰匙(男廁) Key | 角色背包增加該物品 |
| B5 | 2 教務處鑰匙(女廁) Key | 角色背包增加該物品 |
| B5 | 3 教職員室鑰匙(圖書館) Key | 角色背包增加該物品 |
| B5 | 4 老師的筆記 7(圖書館) Paper | 角色背包增加該物品 |
| B5 | ※ 門(教職員室) Classroom_door | 檢查角色身上是否有開鎖條件(鑰匙) Key |
| B5 | 5 教官室鑰匙(教職員室) Key | 角色背包增加該物品 |
| B5 | 6 老師的筆記 8(教職員室) Paper | 角色背包增加該物品 |
| B5 | ※ 門(教官室) Classroom_door | 檢查角色身上是否有開鎖條件(鑰匙) Key |
| B5 | 7 鐵槌(教官室) Hammer | 角色背包增加該物品 |
| B5 | ※ 門(總務處) Classroom_door | 檢查角色身上是否有開鎖條件(鐵槌) Hammer |
| B5 | 8 頂樓鑰匙(總務處) Key | 角色背包增加該物品、提示訊息(引導玩家回到樓梯) |

5.3 遊戲實作

在這裡做一個簡單的建模教學，製作杯子，首先一個正方體，使用面編輯模式，如圖 5.1，然後正下方的面向下延伸一點點，為杯頭，案 S 縮小，如圖 5.2，然後向下延伸長方體，為杯柱，如圖 5.3，再向下延伸一點點，案 S 拉大，再向下延伸一點點，為杯底，如圖 5.4，然後對著最上面的面，案 X 選刪除面，如圖 5.5。

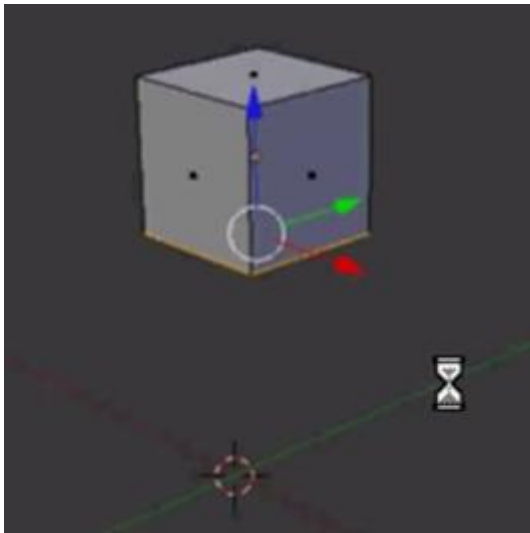


圖 5.1 正方體之面編輯

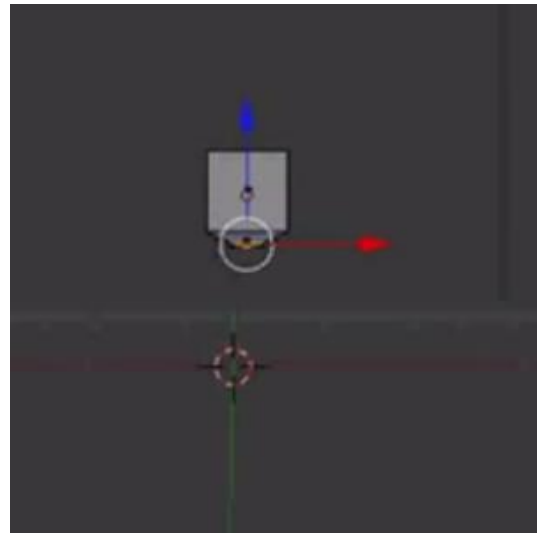


圖 5.2 正方體之延伸

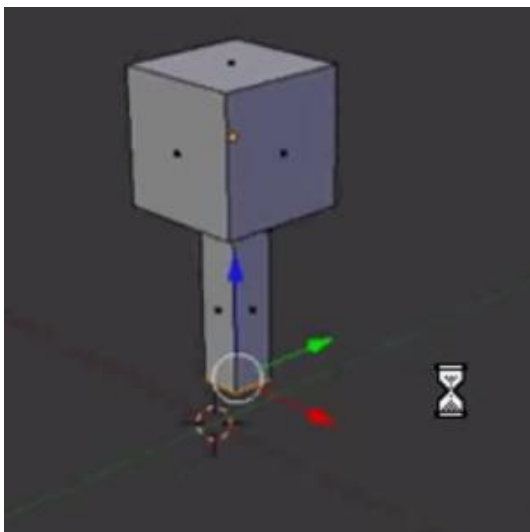


圖 5.3 正方體之延伸杯柱

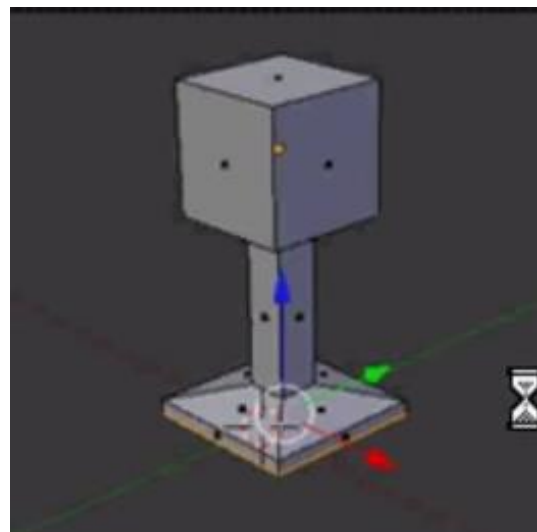


圖 5.4 正方體之延伸杯底

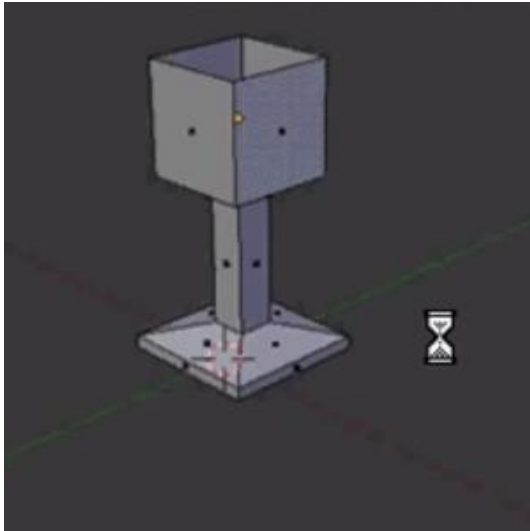


圖 5.5 除去頂部面

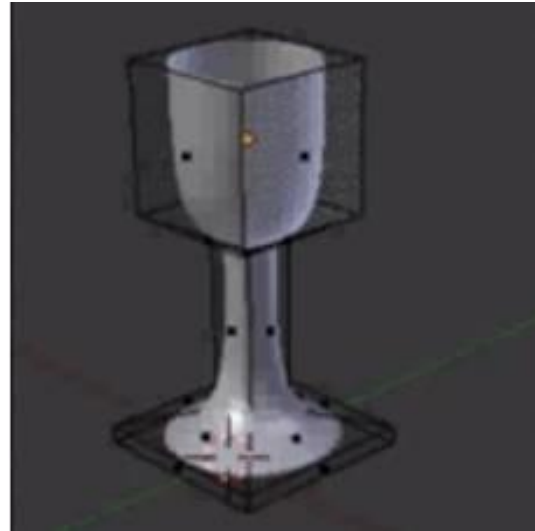


圖 5.6 模型表面圓弧化

最後選取工具箱內的 Subdivision Surface，功能是讓物件表面更多點更圓弧化，參數條成 3 以後，如圖 5.6 一個杯子 3D 模型就完成了。

依照前面製作的模型將檔案拉進 Unity 的專案夾內，將物件依照樓層歸類，放入各個樓層的場景內，如圖 5.7，再將場景內的模型拉進上方的 Scene 內進行擺設。

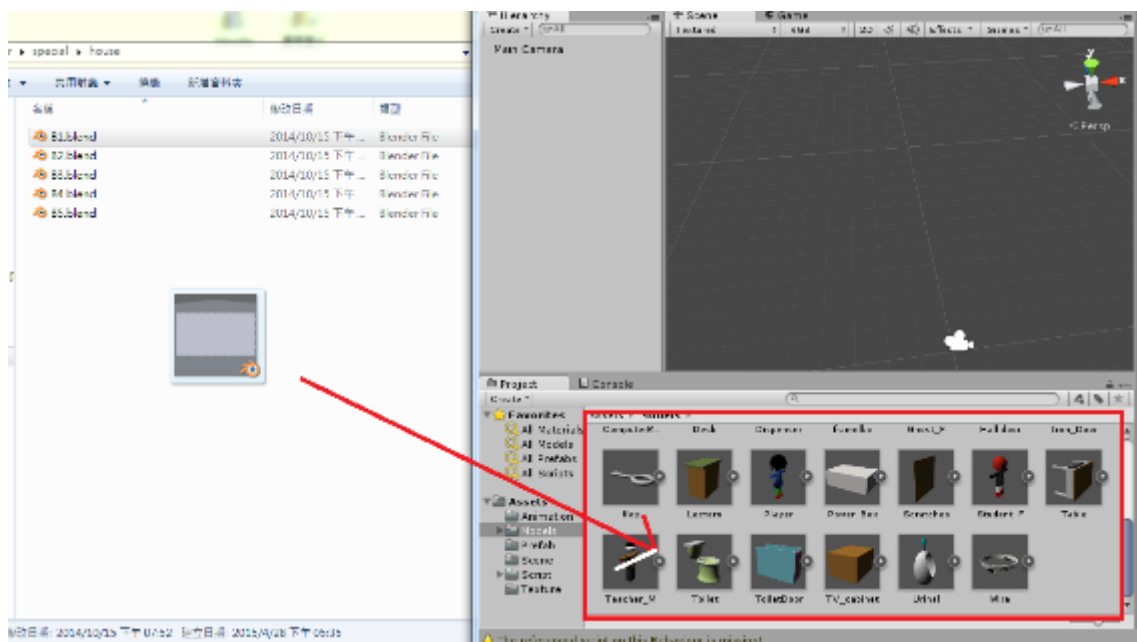


圖 5.7 模型拖曳至 Unity

將地圖模型拉進 Scene 內擺設，如圖 5.8。

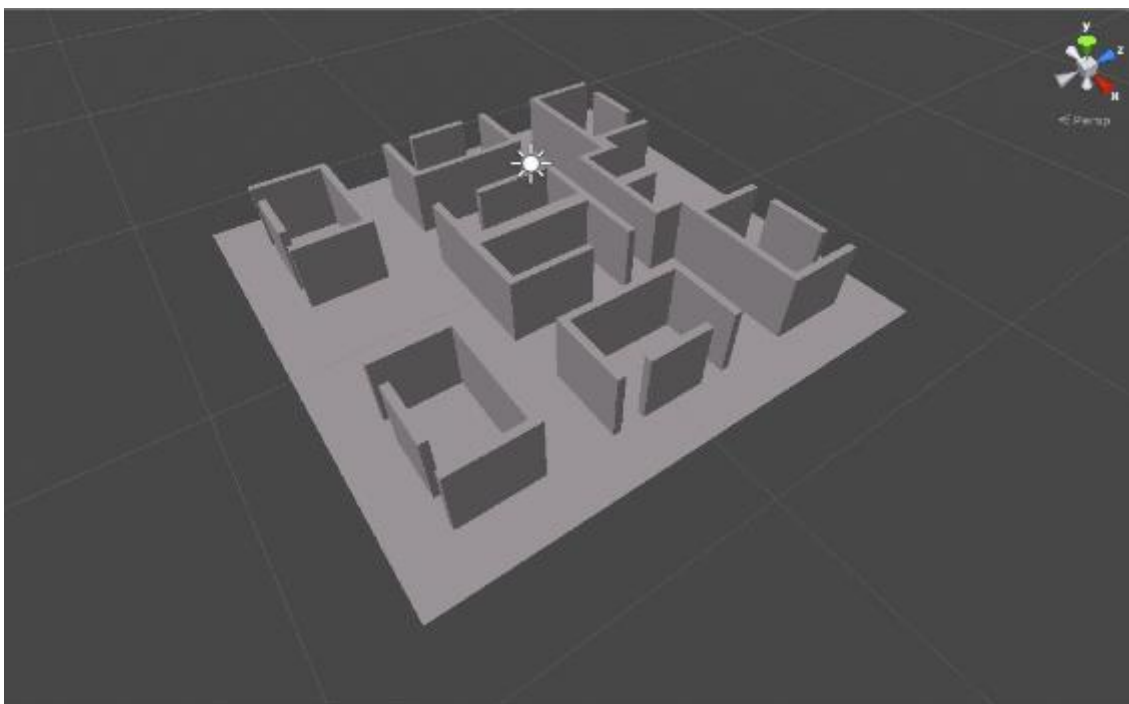


圖 5.8 地圖拉進場景

點選 Model，切換到 Model 的頁面，如圖 5.9。

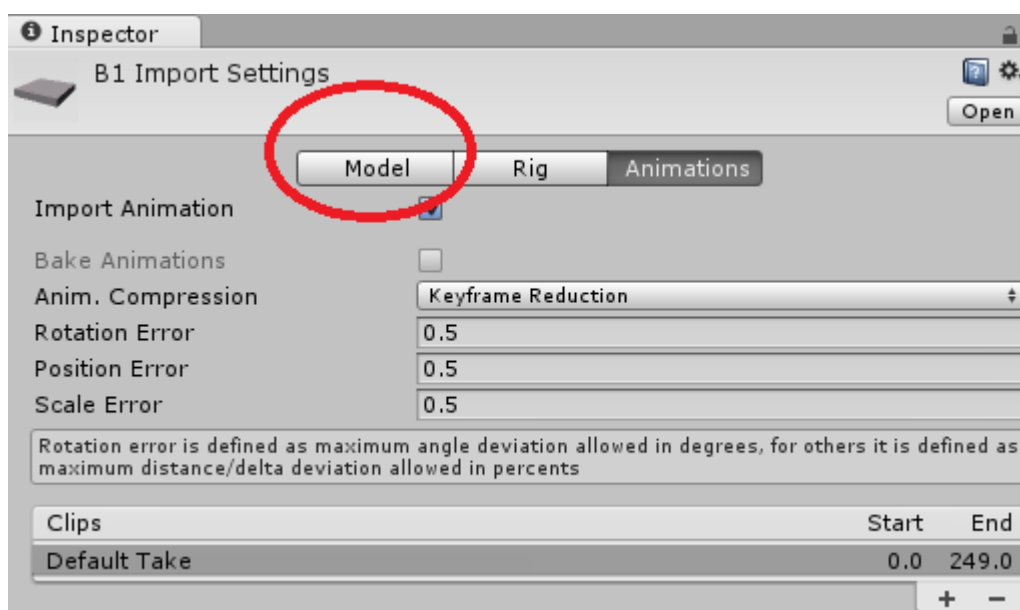


圖 5.9 模式 Model 切換

更改 Scale Factor 的參數，改變模型大小，如圖 5.10。

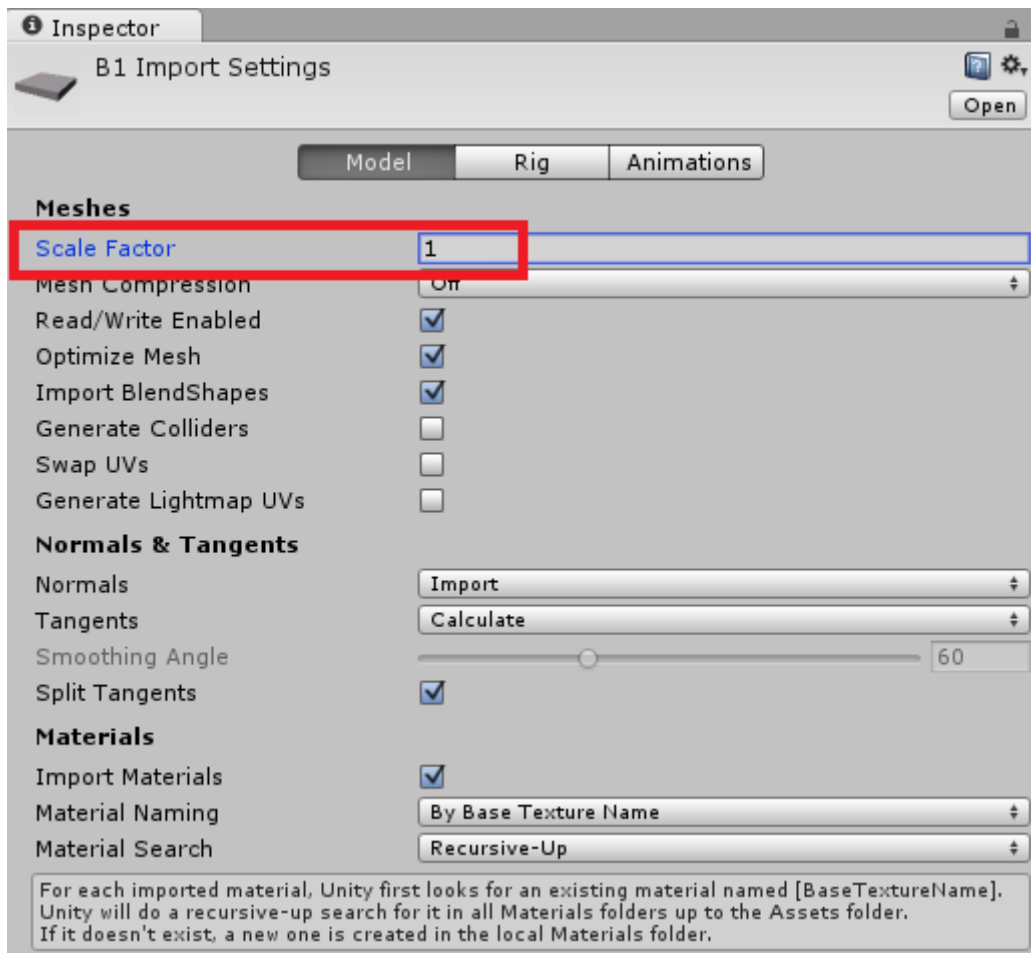


圖 5.10 模式 Model 內參數設定

在 GameObject 選單，Light 選 Directional Light 產生光源物件，如圖 5.11。

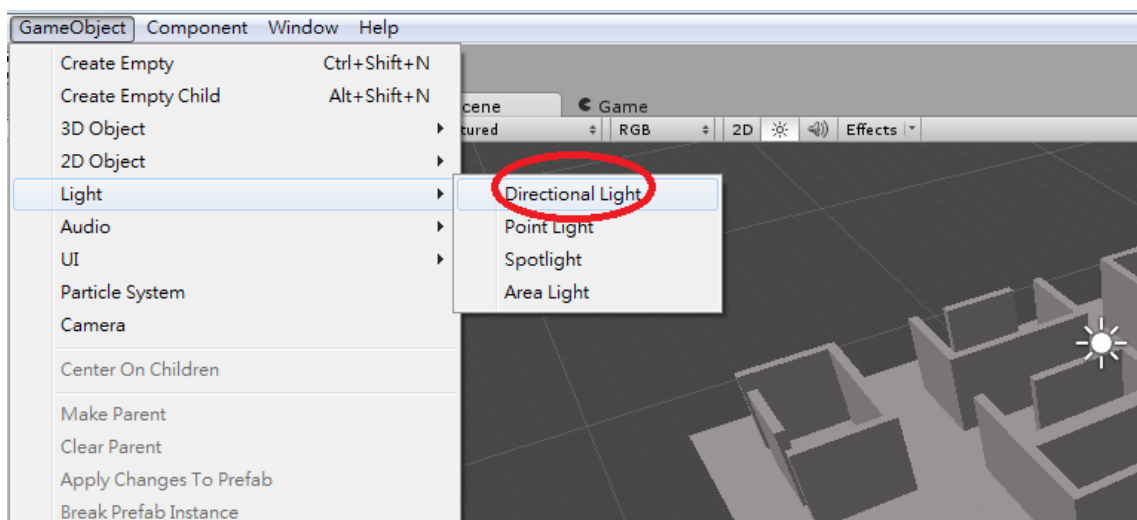


圖 5.11 產生光源物件

把門的物件拖到地圖場景中的大概位置上，如圖 5.12。

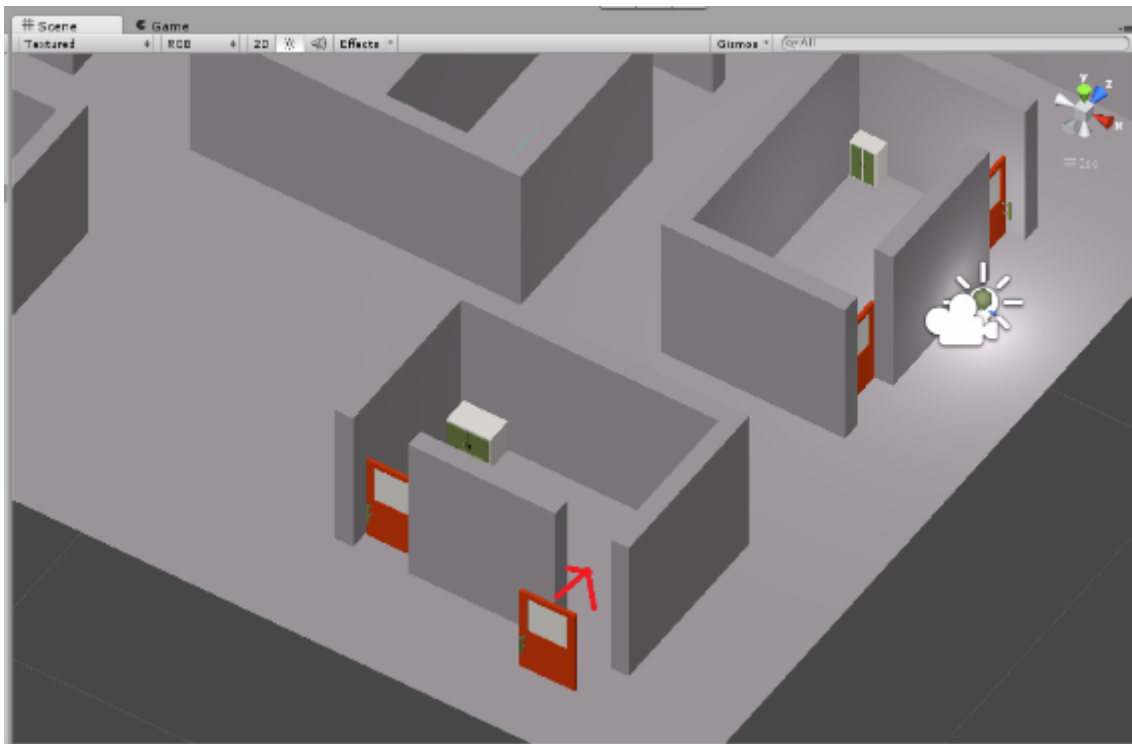


圖 5.12 門物件移動

再將門物件的 Position 和 Rotation 來作精準微調門的座標位置，如圖 5.13。

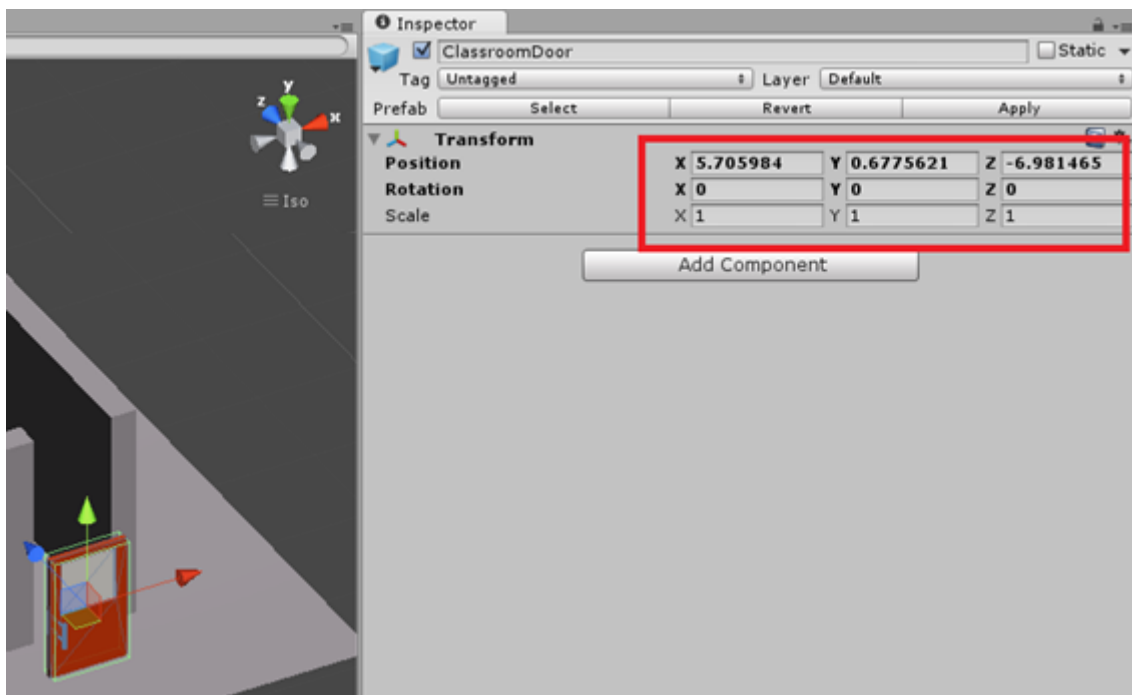


圖 5.13 物件位置微調

在物件的 Inspector 的 Add Component 的 Physics 選項內，如圖 5.14。

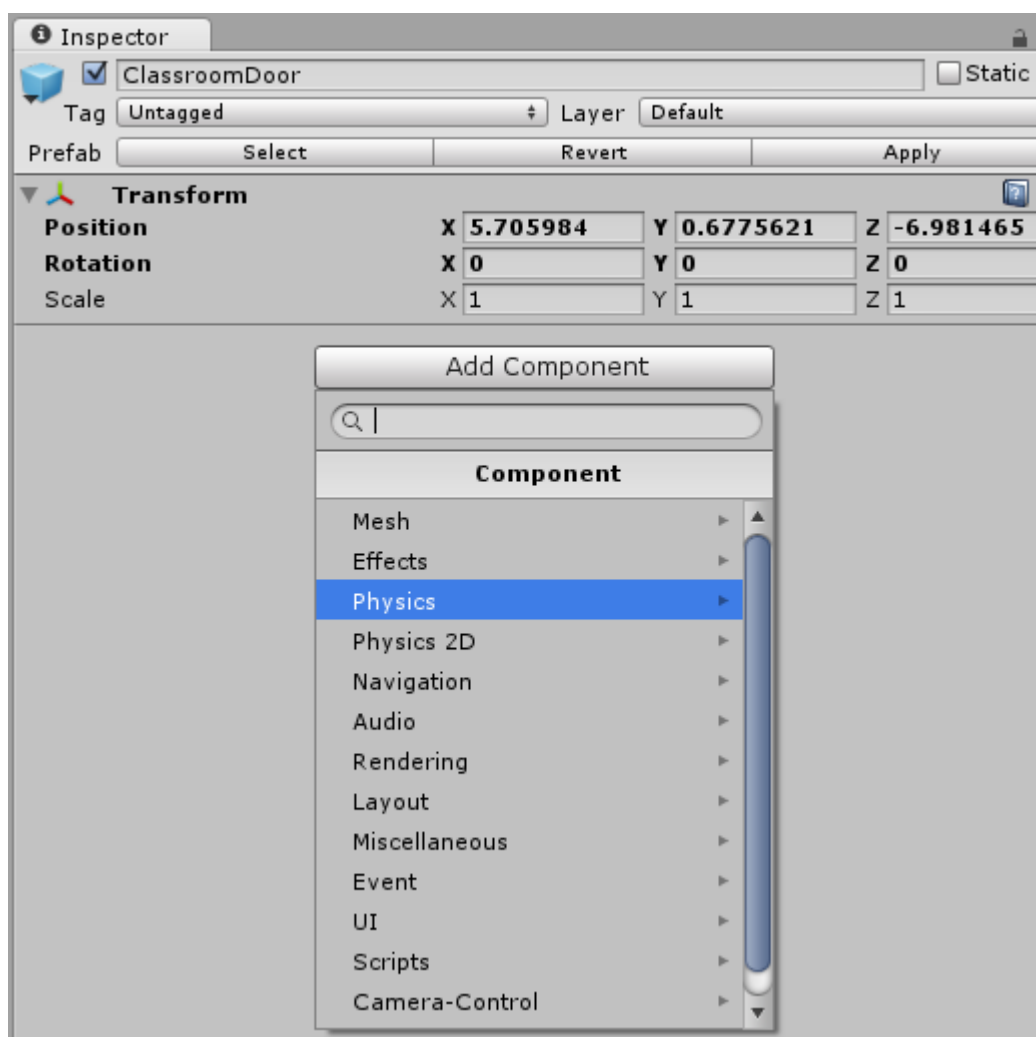


圖 5.14 物件 Add Component 的選單

點選 Box Collider，幫門上面新增碰撞功能，如圖 5.15。

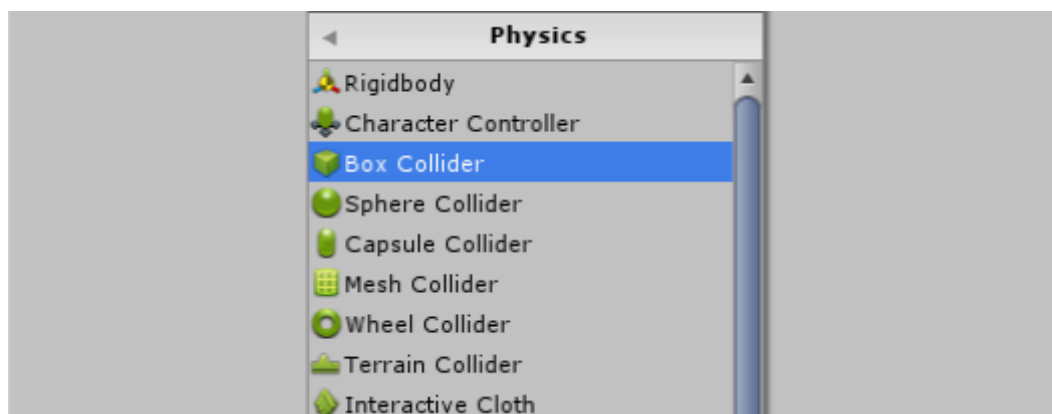


圖 5.15 選單內的 Box Collider 選項

在門的物件上加上碰撞，如圖，的綠色範圍，如圖 5.16。

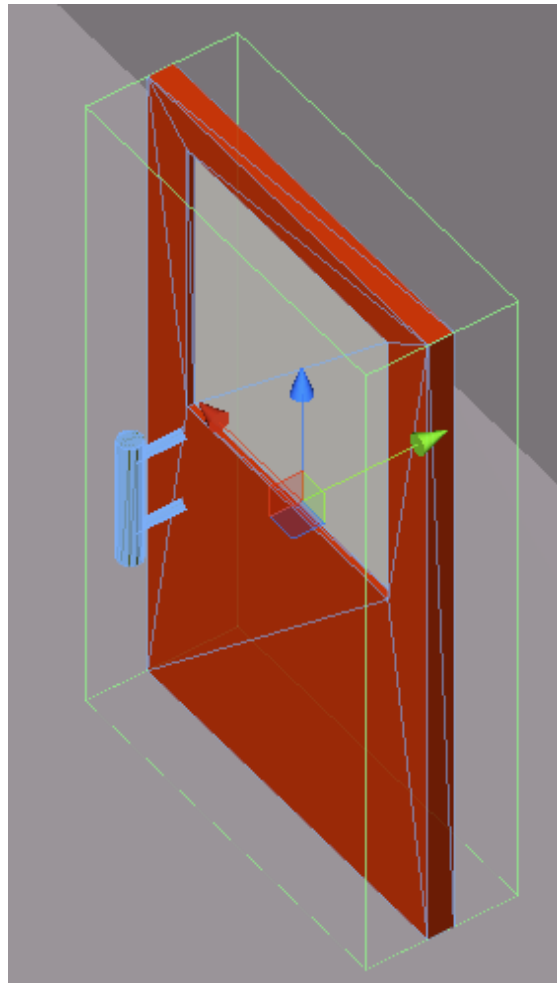


圖 5.16 門物件的碰撞範圍

這是碰撞的設定，Center 是修改位置，Size 是改變大小，如圖 5.17。

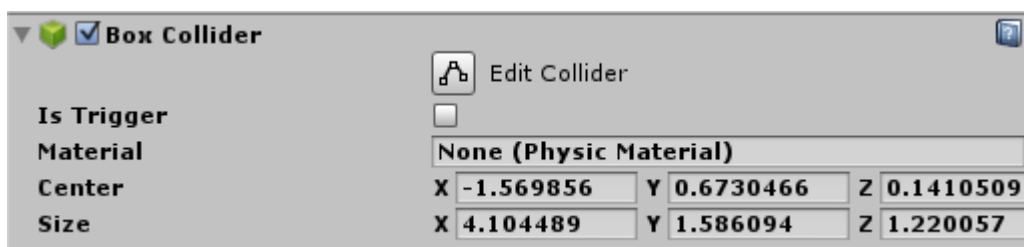


圖 5.17 碰撞設定

加上碰撞後玩家不會直接穿越物件，而會檔在物件外。

將 C# 程式碼拖曳到門上，如圖 5.18。

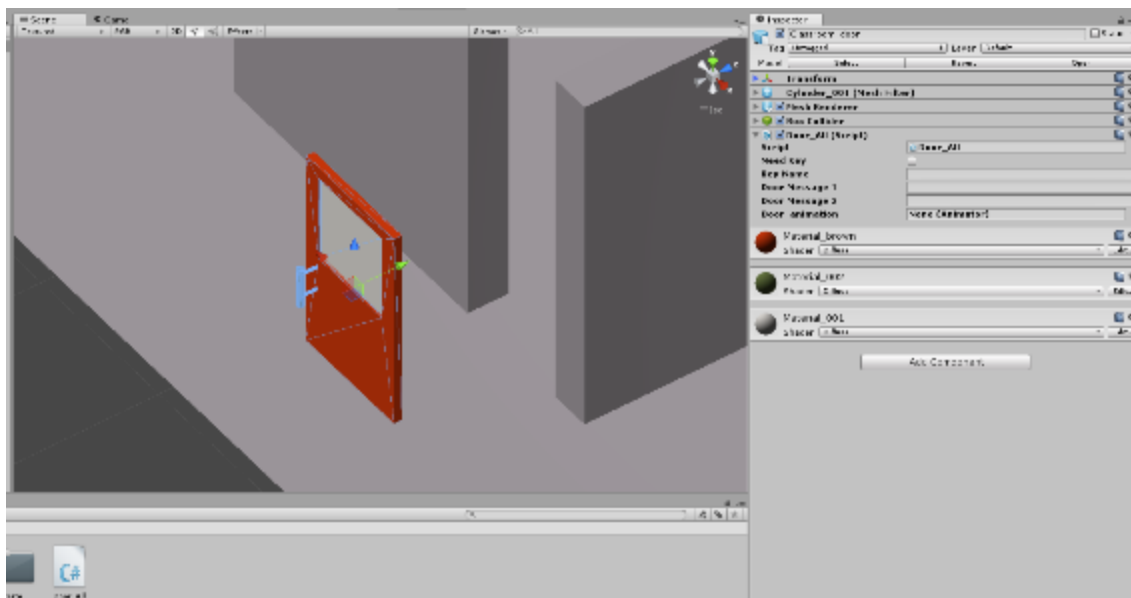


圖 5.18 套用程式碼到門物件上

對程式碼參數進行編輯，針對各個物件調整不同數值，如圖 5.19。

- Need Key (判斷門是否上鎖)
- Key Name (開鎖鑰匙的名子)
- Door Message 1 (門上鎖的訊息)
- Door Message 2 (門解鎖後的訊息)

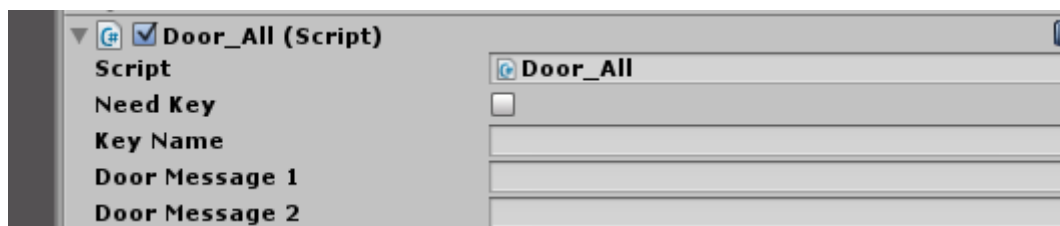


圖 5.19 門物件程式碼詳細參數

在新增一個碰撞，將 Is Trigger 打勾，讓程式碼可以控制門的觸發，如圖 5.20。

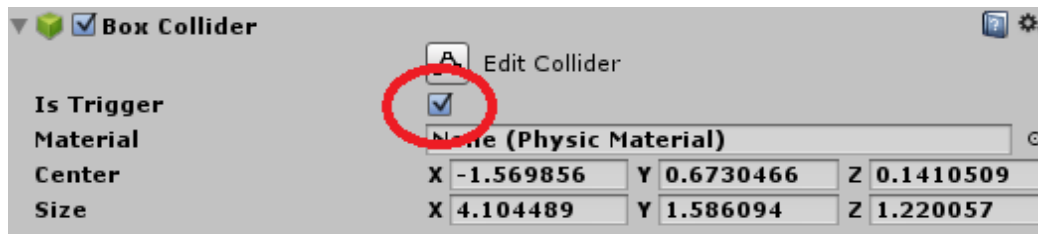


圖 5.20 碰撞觸發選項

將物件拉進場景，並調整位置，如圖 5.21。

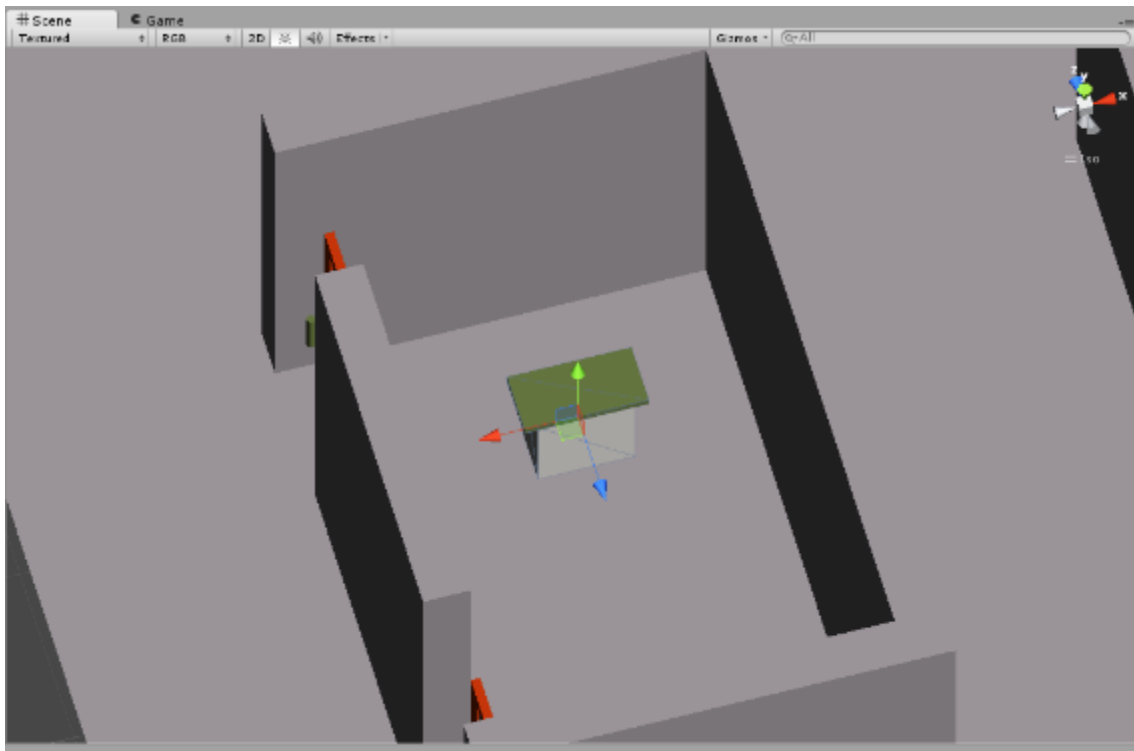


圖 5.21 物件移動至場景

新增兩個碰撞，其中一個碰撞的 Is Trigger 打勾，如圖 5.22。

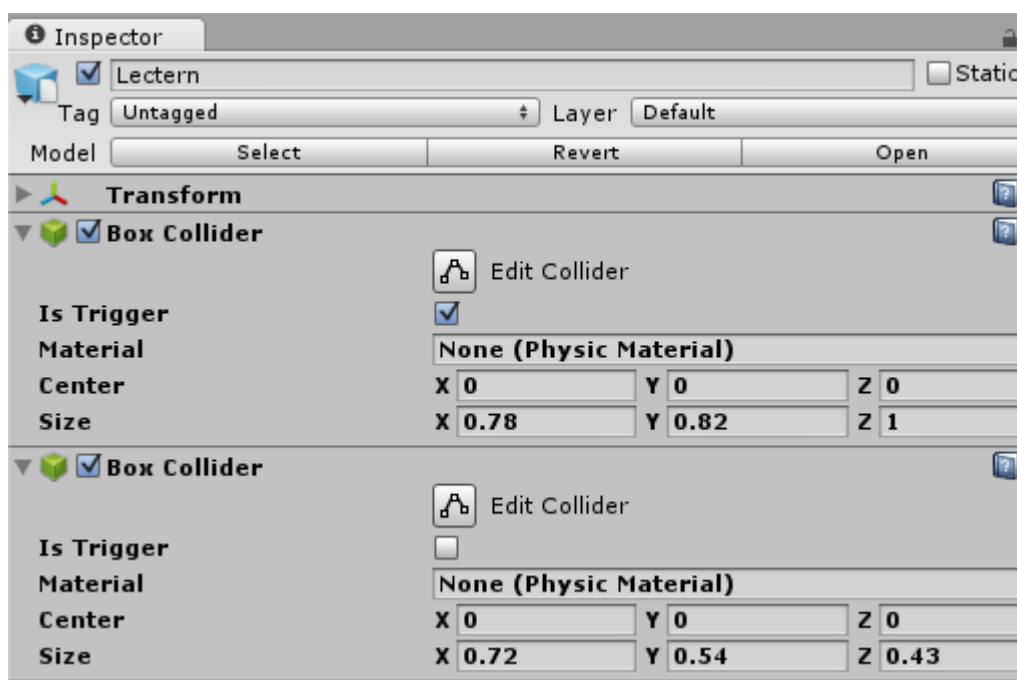


圖 5.22 碰撞設定

將兩個不同的碰撞分別依照物體的大小以及程式碼的範圍分別調整大小，如圖 5.23。

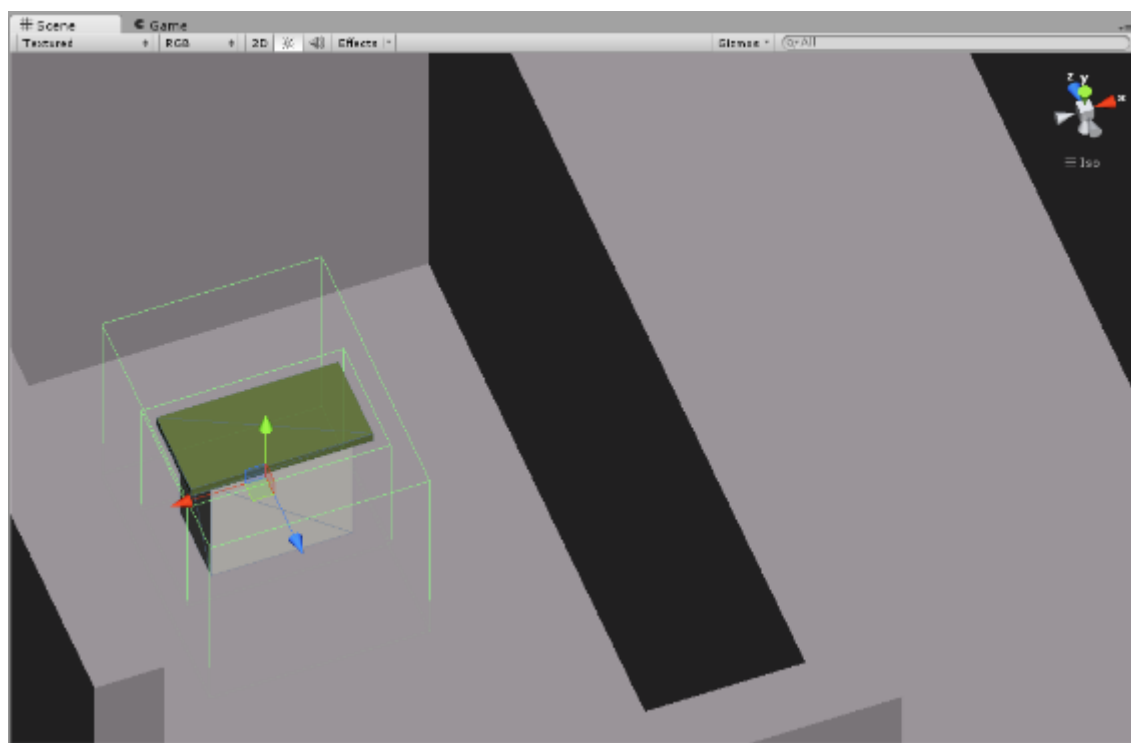


圖 5.23 碰撞範圍

將下方檔案內的 C#程式碼拉進物件內，如圖 5.24。

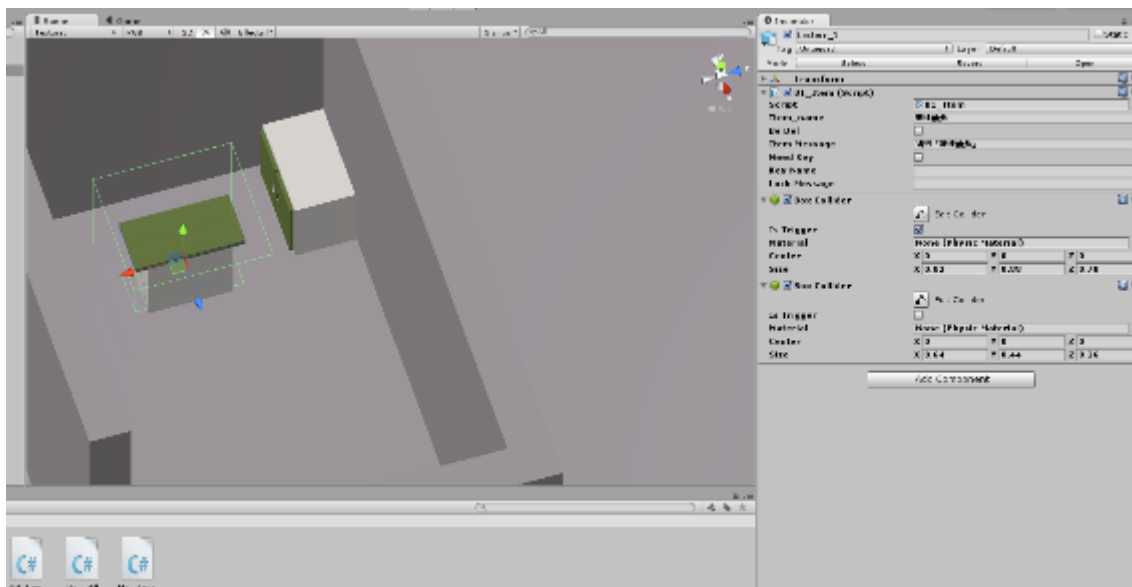


圖 5.24 物件套用程式碼

根據不同的道具，進程式碼參數的設定，如圖 5.25。

- Item_name（道具名稱）
- Be Del（是否可被刪除）
- Item Message（調查道具得到的訊息）
- Need Key（需不需要解鎖道具）
- Key Name（解鎖道具名稱）
- Lock Message（提示訊息）

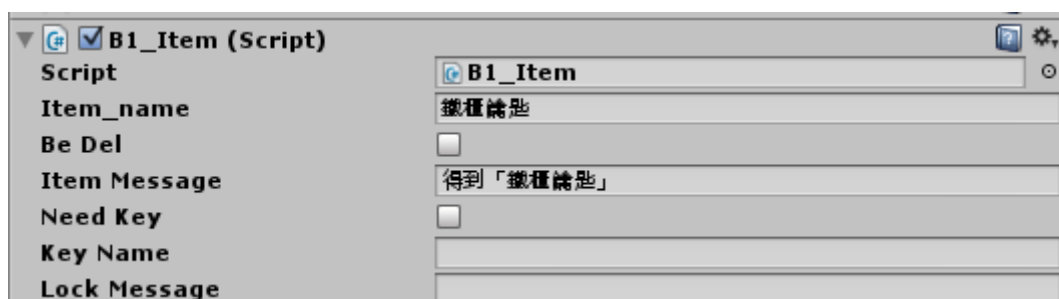


圖 5.25 程式碼參數

以上為地圖與模型的相關設定。

玩家的部分，將玩家的模型放進場景後，如圖 5.26。

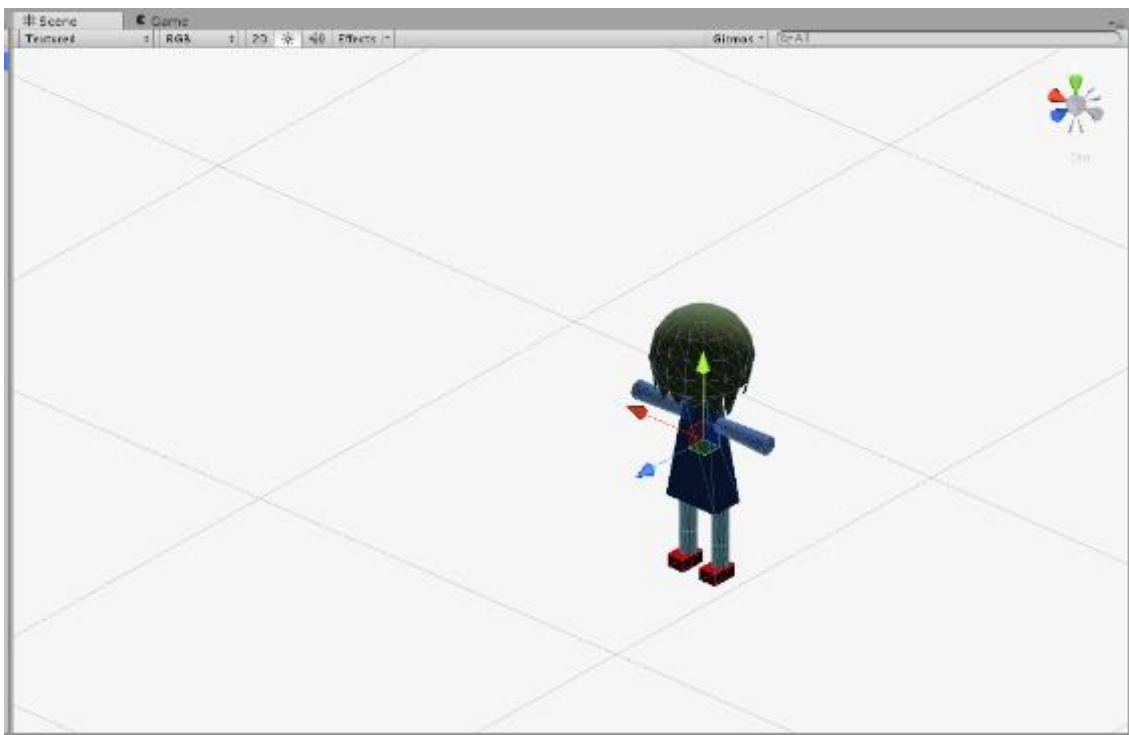


圖 5.26 玩家模型拖曳至場景

將玩家的 Tag 調成 Player，Tag 類似像標籤，讓遊戲引擎知道該物件是甚麼東西，如圖 5.27。

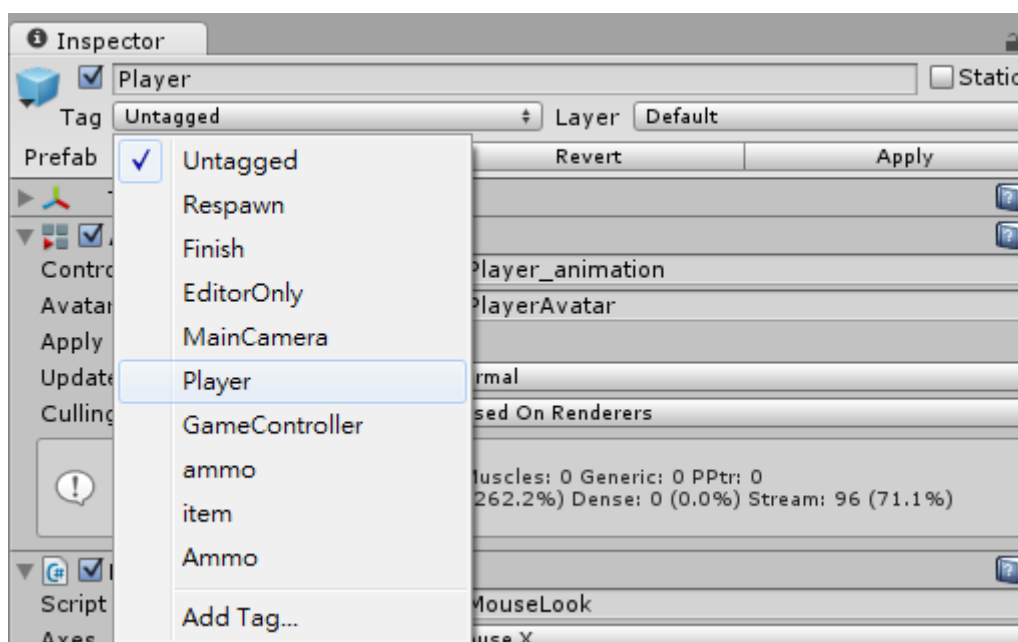


圖 5.27 模型 Tag 調整

在玩家物件上新增一個光源物件 Point Light，如圖 5.28。

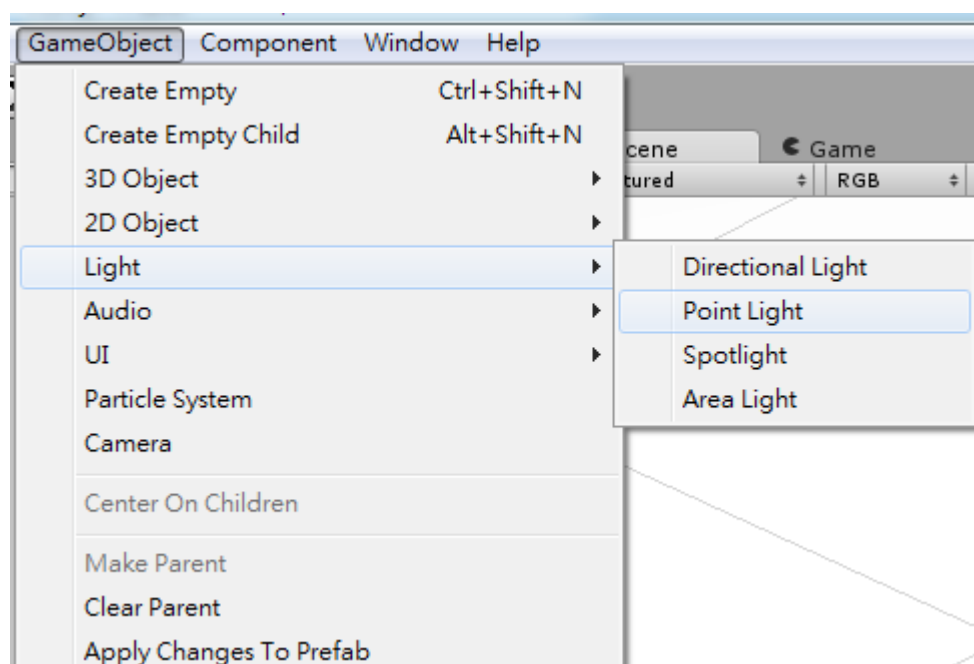


圖 5.28 產生 Point Light 光源物件

把光源物件 Point Light 拖曳到 Player 物件底下，成為 Player 的子物件，如圖 5.29。

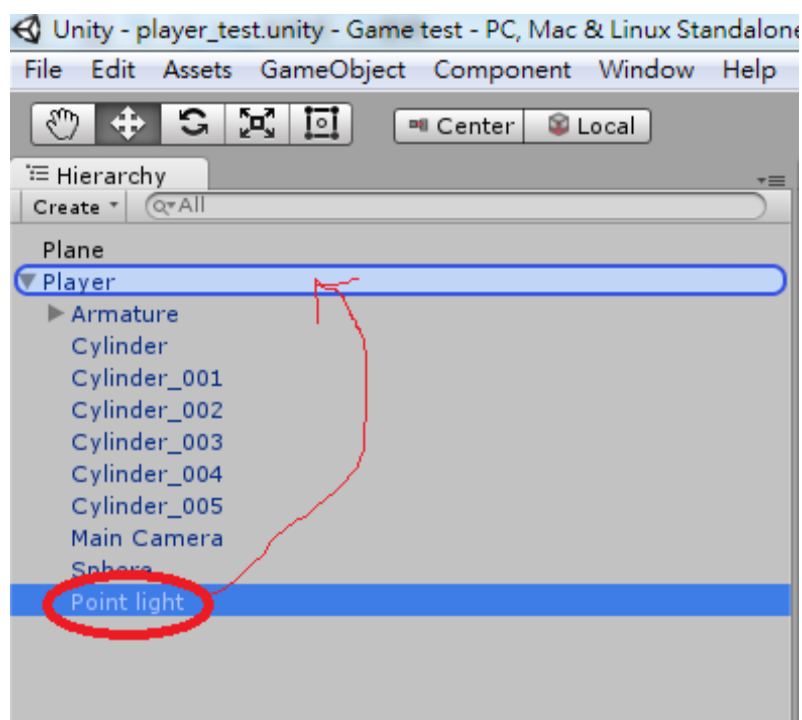


圖 5.29 物件拖曳

調整光源 Point Light 的座標，對齊與 Player 一致，當作遊戲中玩家的照明光源，如圖 5.30。

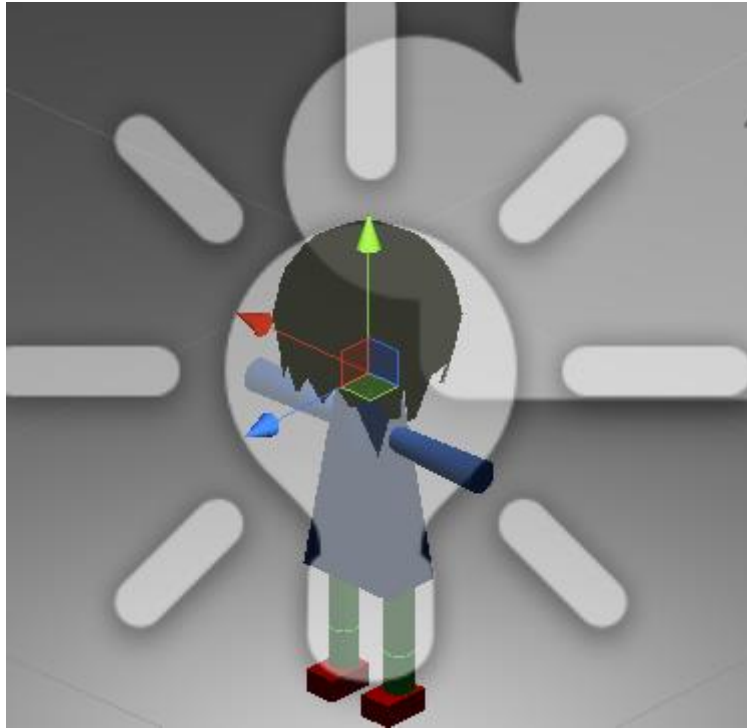


圖 5.30 調整光源作標

在玩家的 Inspector 欄位點選 Add Component，選擇 Physics，如圖 5.31。

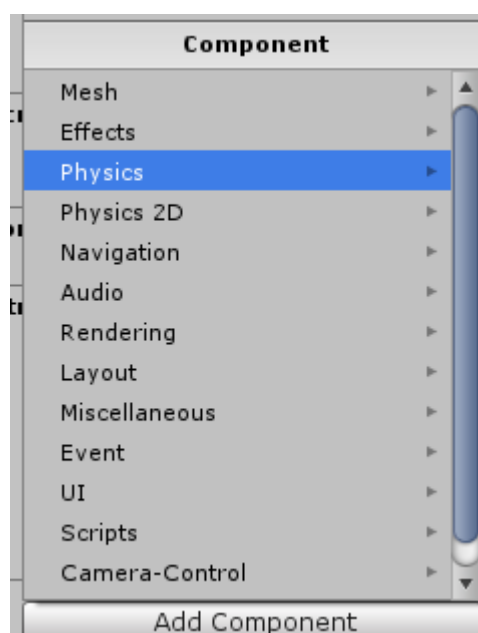


圖 5.31 選單 Add Component

幫 Player 新增 Character Controller，控制角色物件座標移動，和重力的調整，如圖 5.32。

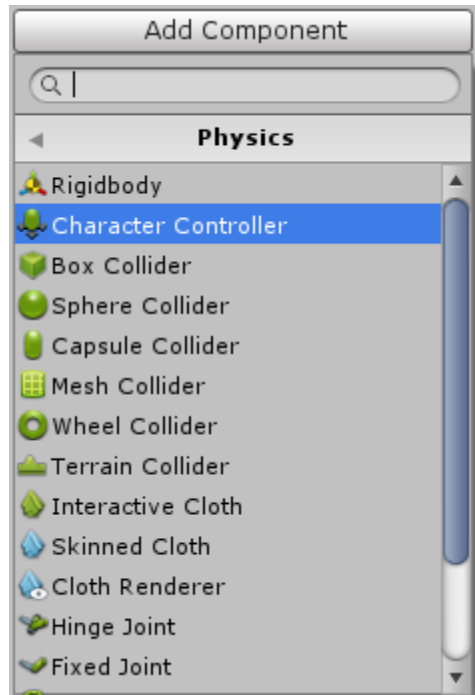


圖 5.32 新增 Character Controller

將程式碼拉到 Player 物件裡面，如圖 5.33。

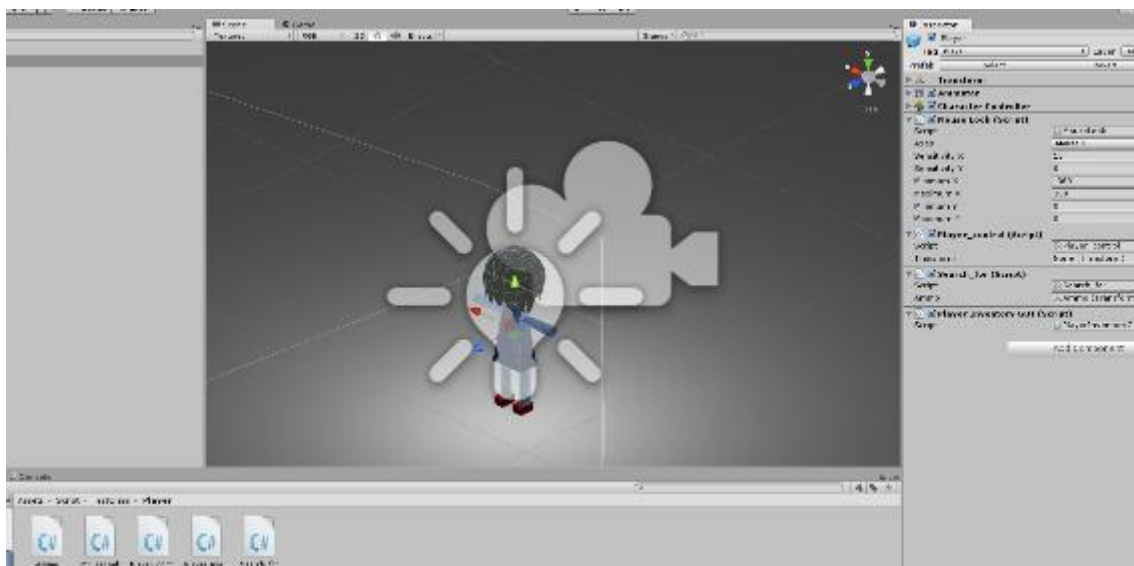


圖 5.33 套用程式碼

這是 Unity 內建控制 Camera 的程式碼，參數設定皆用預設數值，如圖 5.34。

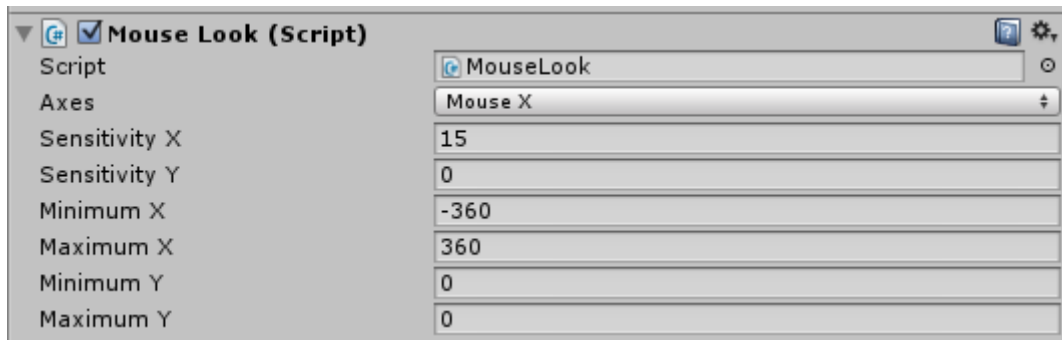


圖 5.34 攝影機參數

以上為基本人物的光源與攝影機設定。

接著製作玩家調查時，觸發程式碼的調查子彈，從選單新增一個 Sphere 的 3D 圖形，如圖 5.35。

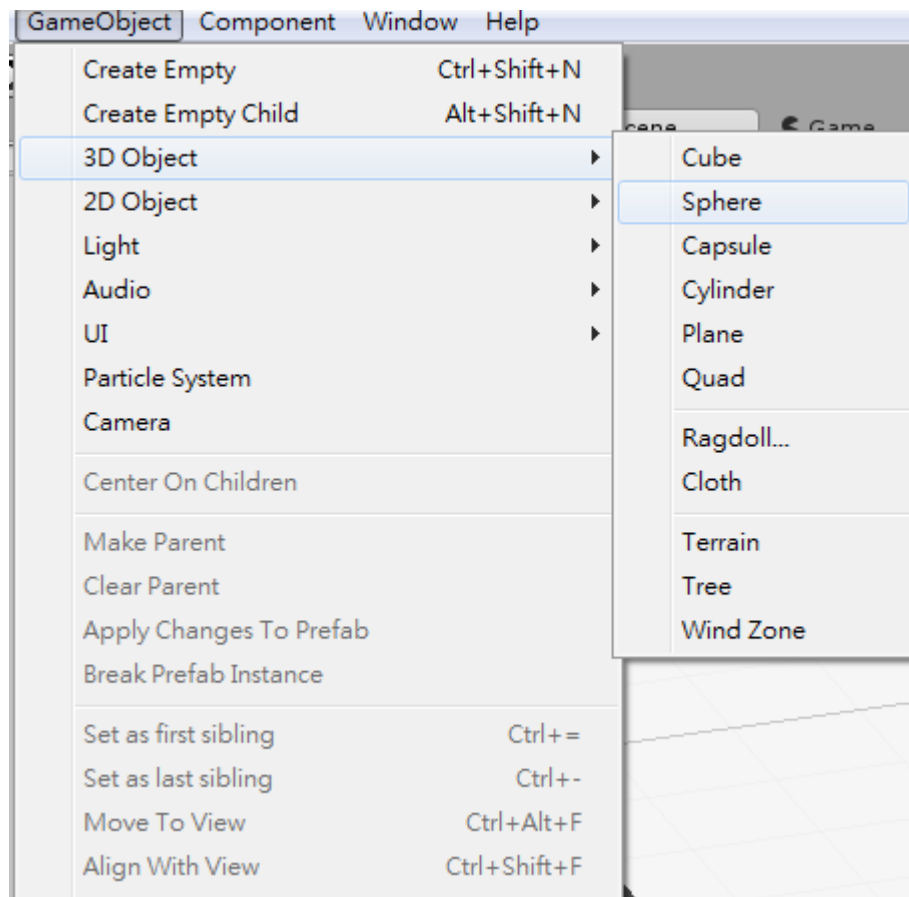


圖 5.35 製作調查子彈

為了操作時辨識的方便，暫時替物件加上顏色，如圖 5.36。

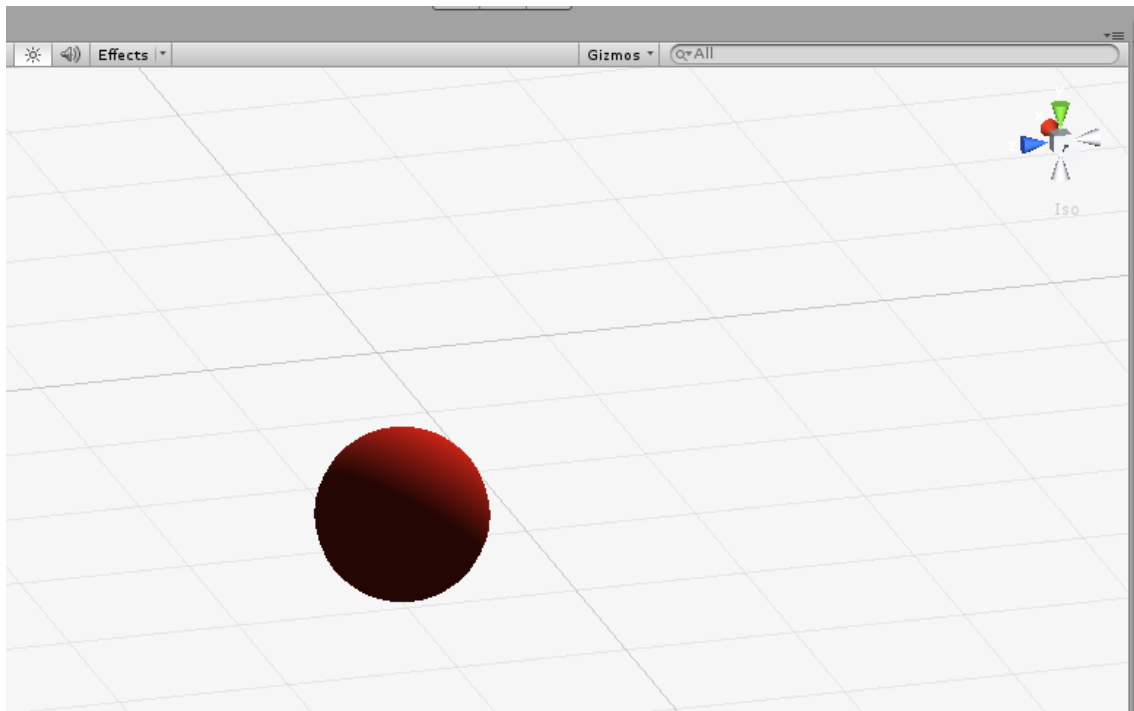


圖 5.36 子彈物件套用顏色

所謂的子彈是利用球型物件製作成透明，掛在玩家模型的手上，利用碰撞原理，間接觸發程式，進行判斷或執行程式碼，進而產生所需要的現象或事件。

幫子彈掛上程式碼，如圖 5.37。

- Live Time（子彈存活時間）

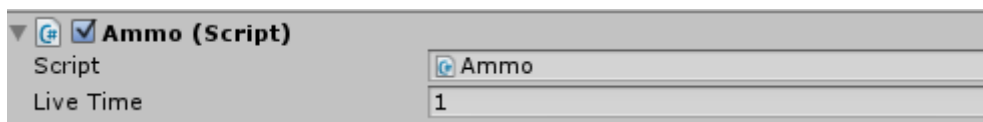


圖 5.37 子彈物件套用程式碼

在子彈物件下點選 Add Component 的 Physics 選項內，如圖 5.38。

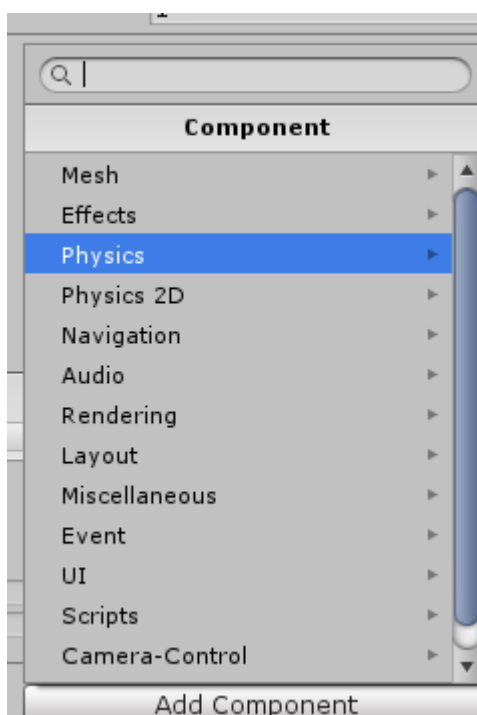


圖 5.38 選單 Add Component

點選 Rigidbody，控制物件碰撞相關的原件，如圖 5.39。

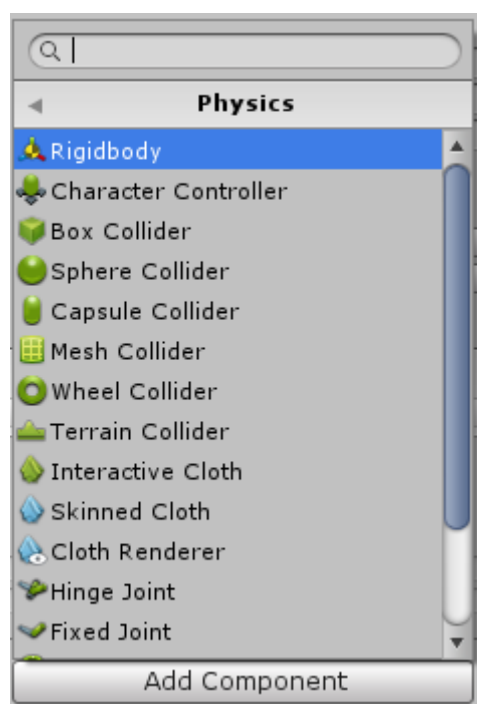


圖 5.39 新增 Rigidbody

將 Use Gravity 的選項取消，把 Is Kinematic 的選項勾起來，如圖 5.40。

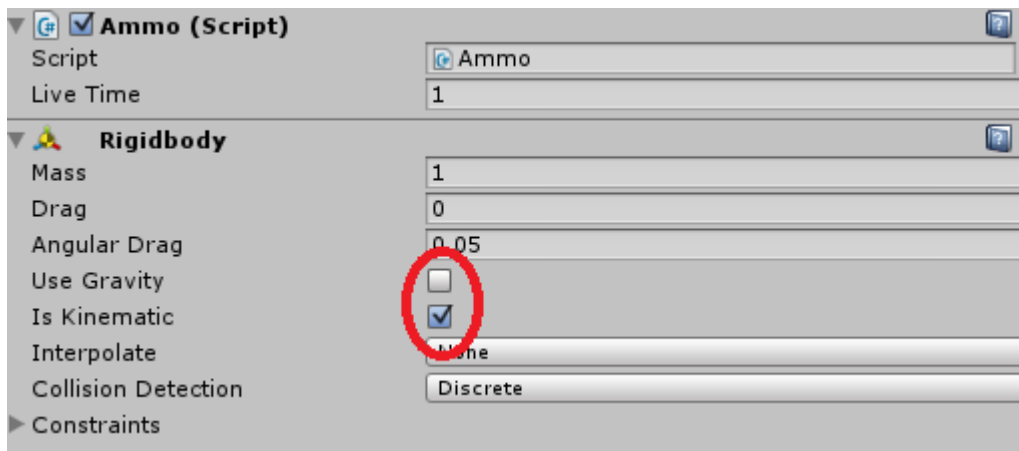


圖 5.40 子彈物件 Rigidbody 參數設定

Mesh Renderer 的勾取消掉，停用子彈的材質，並將 Sphere Collider 的 Is Trigger 打勾，使其讓程式碼控制，如圖 5.41。

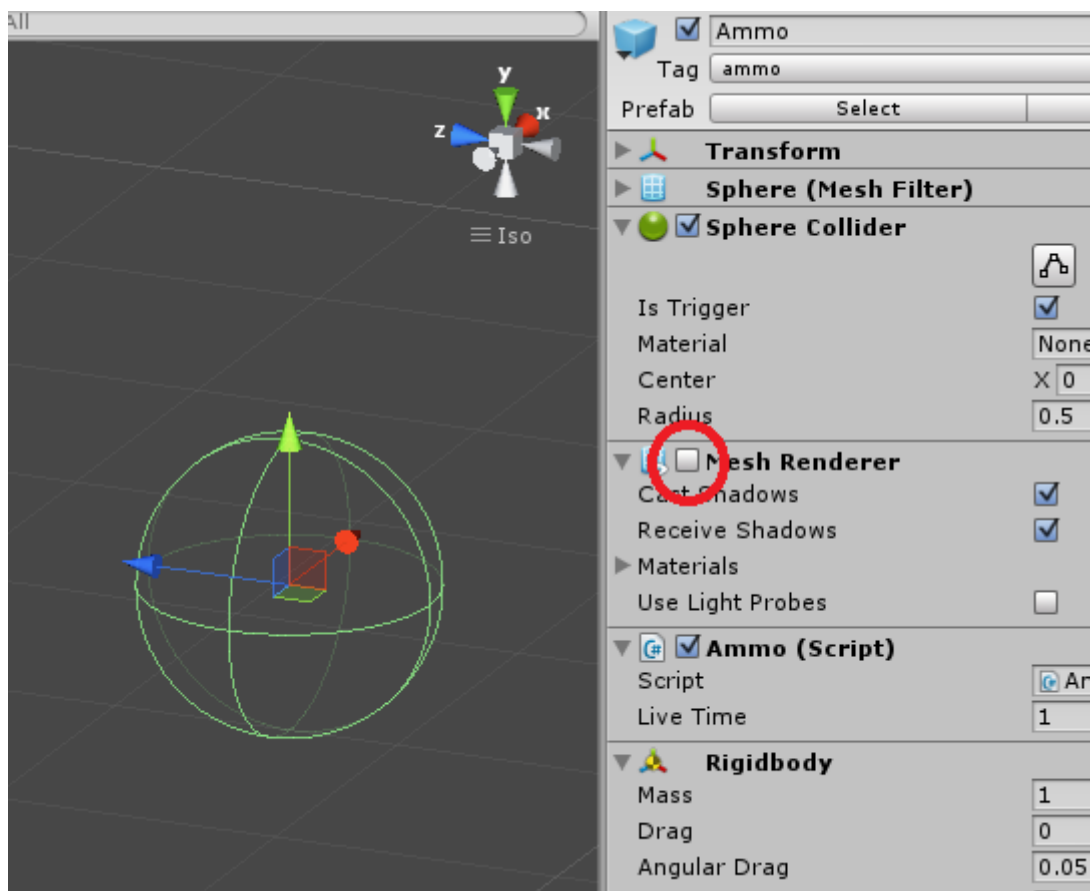


圖 5.41 子彈物件停用材質

將子彈改名成 Ammo，拖到下方欄位製作成 Prefab，如圖 5.42。

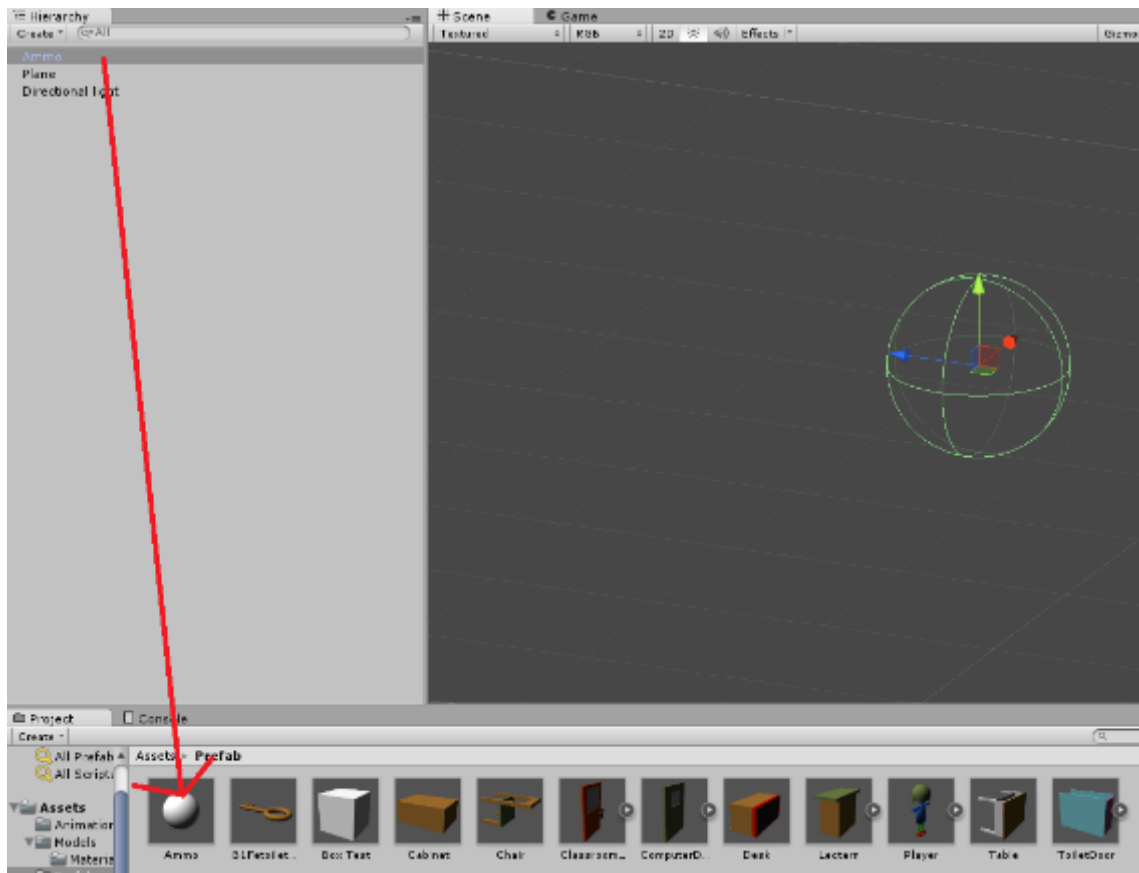


圖 5.42 子彈物件改名製程 Prefab

回到 Player 的物件，將 Ammo 的 Prefab 拖曳到，程式碼 Ammo 參數的欄位，如圖 5.43。

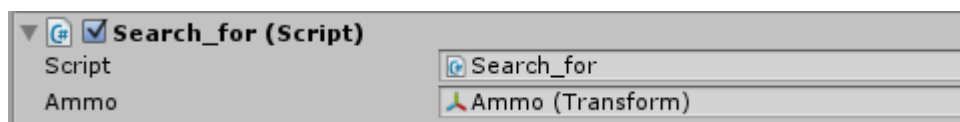


圖 5.43 子彈物件套用程式碼

以上為基本人物的調查子彈製作與設定。

從選單新增一個 Plane 的圖形，如圖 5.44。

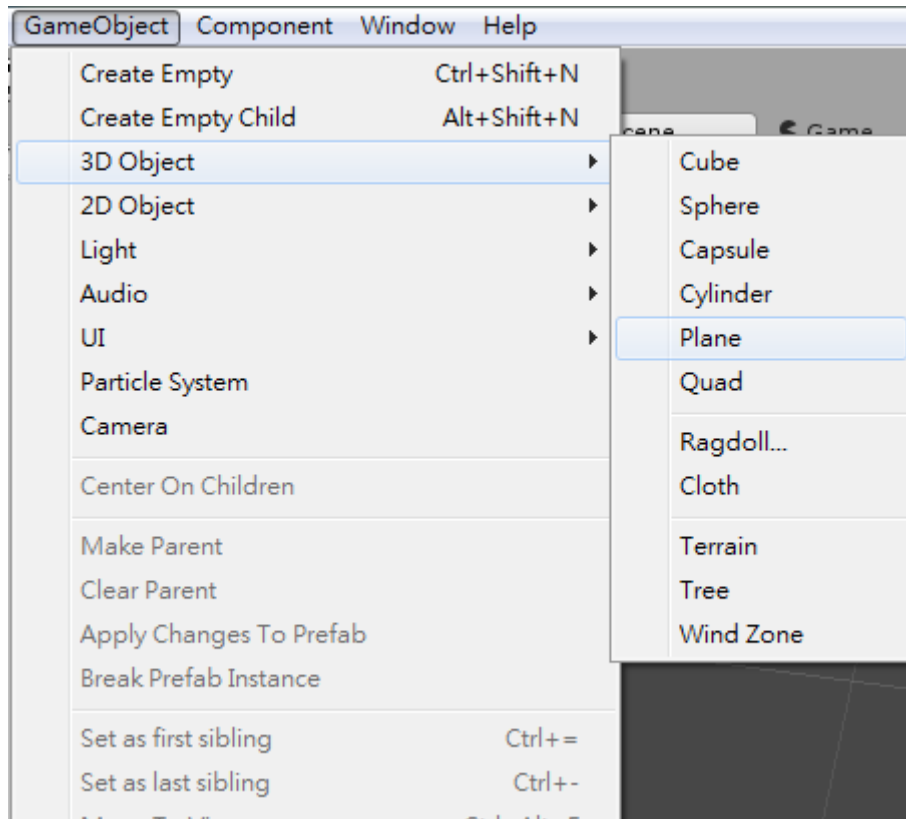


圖 5.44 新增 Plane 物件

將 Plane 調整適當大小，並加上顏色使其在地圖上方便辨識，如圖 5.45。

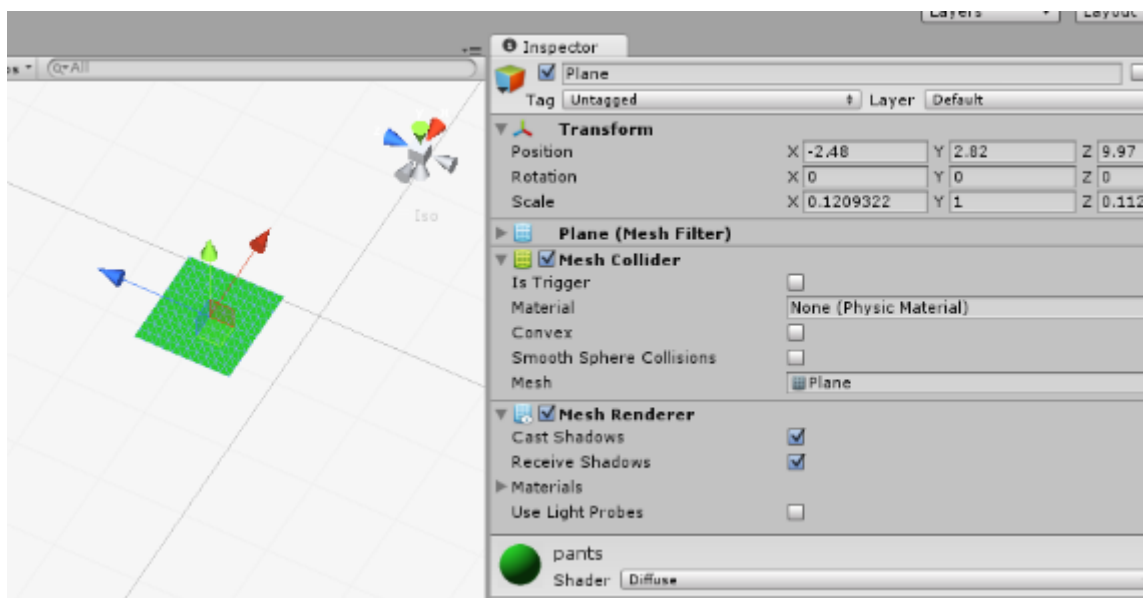


圖 5.45 物件 Plane 調整大小與顏色

將物件命名為 PatrolPoint，把物件拖曳到底下製成 Prefab，如圖 5.46。

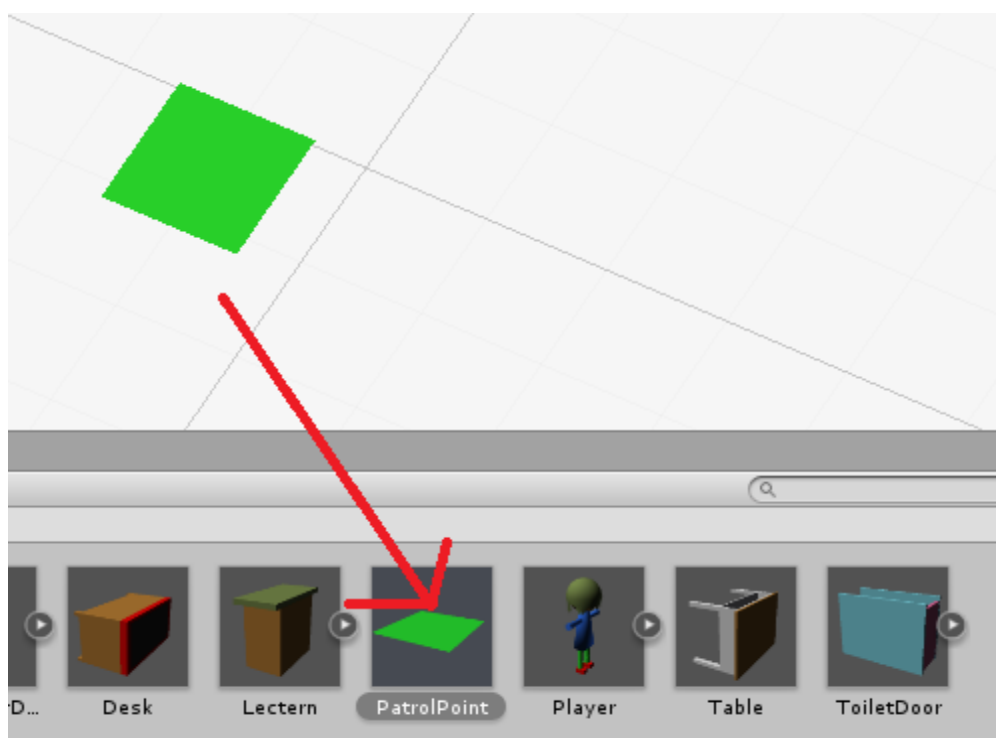


圖 5.46 物件製程 Prefab

新增 Tag 標籤名叫 PatrolPoint，如圖 5.47。

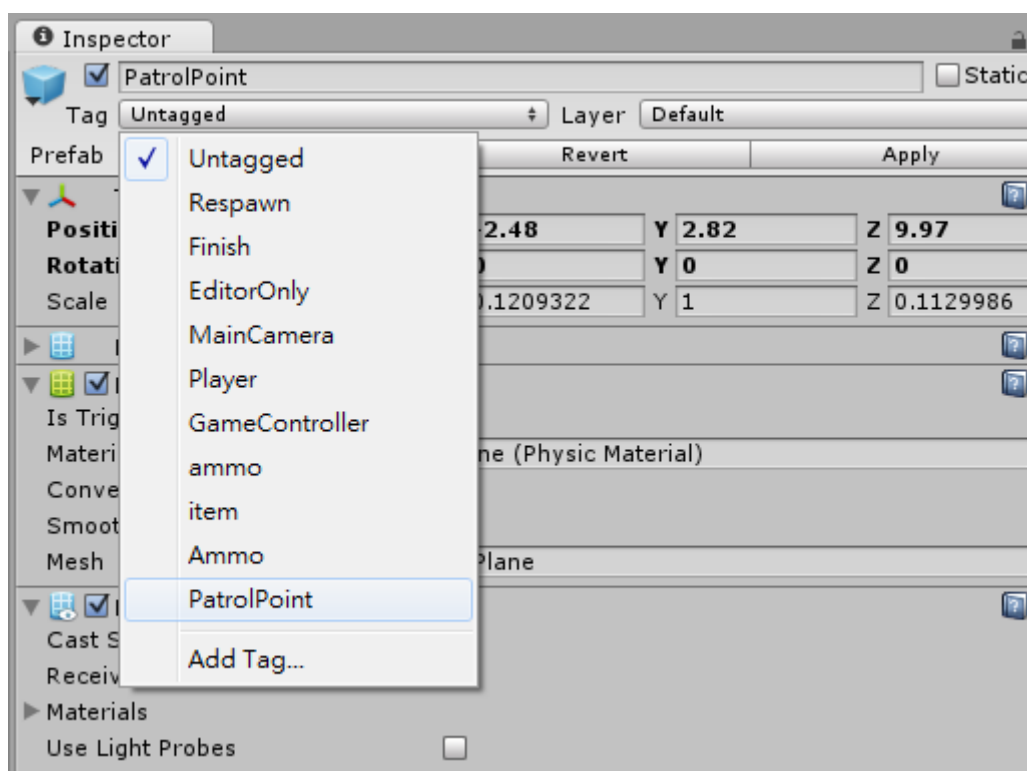


圖 5.47 新增 PatrolPoint 的 Tag

以上為怪物巡邏之製作與設定。

A screenshot of the Unity Inspector window. The 'Create' dropdown menu is open, showing a list of asset types. The 'Animator Controller' option is highlighted with a blue selection bar. Other visible options include Folder, Javascript, C# Script, Boo Script, Shader, Compute Shader, Prefab, Material, Cubemap, Lens Flare, Animation, and Animator Override Controller. The background shows a blurred view of the Unity Hierarchy and Hierarchy View panels.

–54–

將玩家的動畫拖曳到畫面上，如圖 5.50。

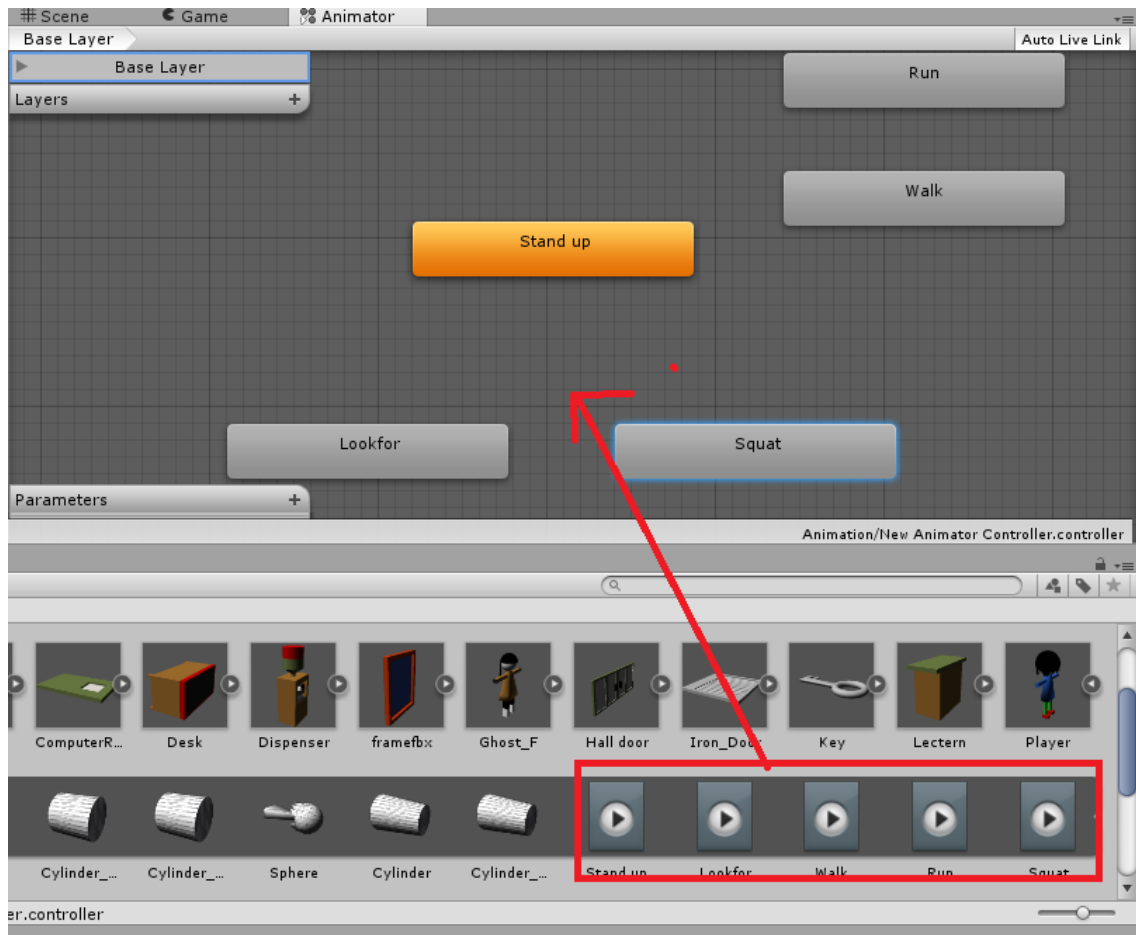


圖 5.50 拖曳玩家動畫

對著動畫按右鍵，點選 Make Transition，如圖 5.50。

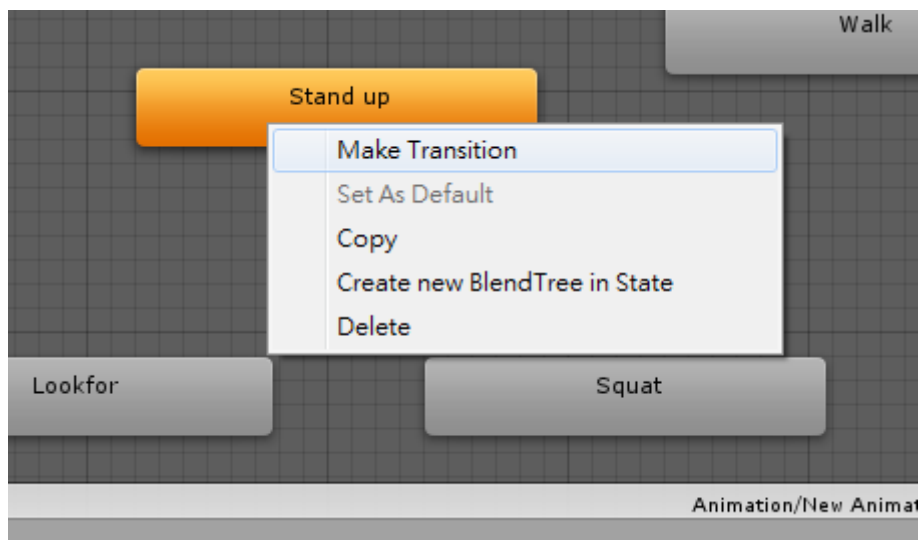


圖 5.51 新增 Make Transition

將白線連到另一個動畫作為關聯，如圖 5.52。

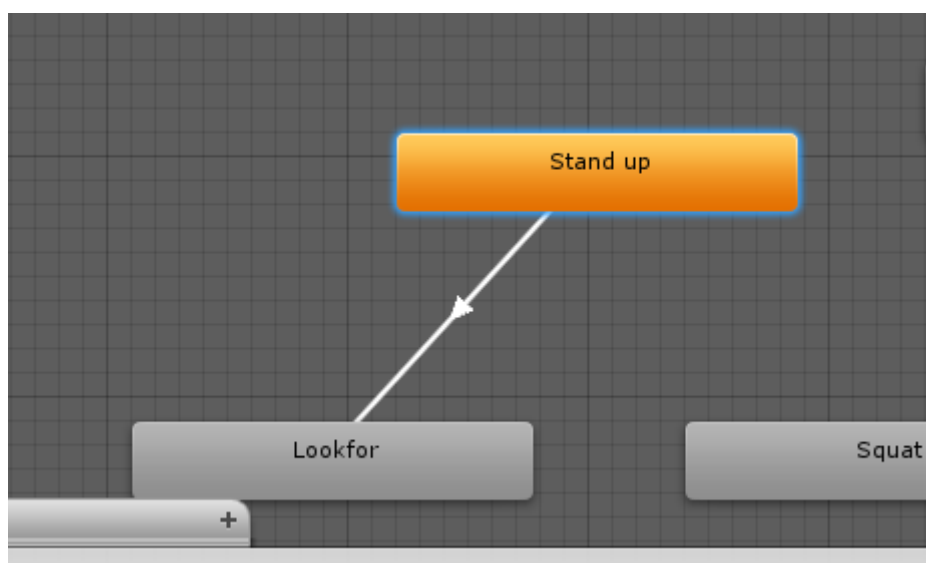


圖 5.52 關聯到動畫

新增 Int 屬性的欄位，並以動畫名稱命名，如圖 5.53。

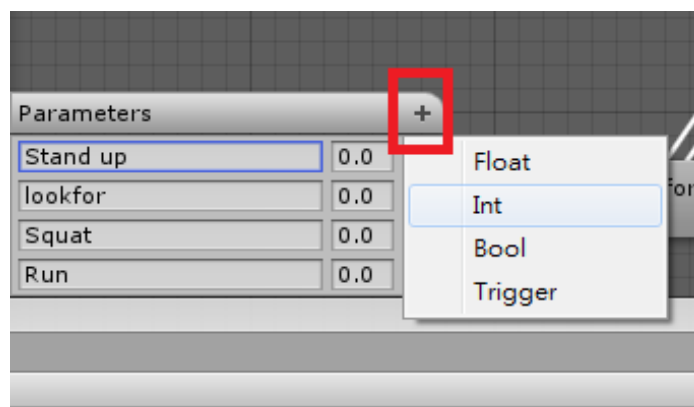


圖 5.53 命名動畫

點選箭頭白線，設定其變數名稱與數值，使其受程式碼控制，如圖 5.54。

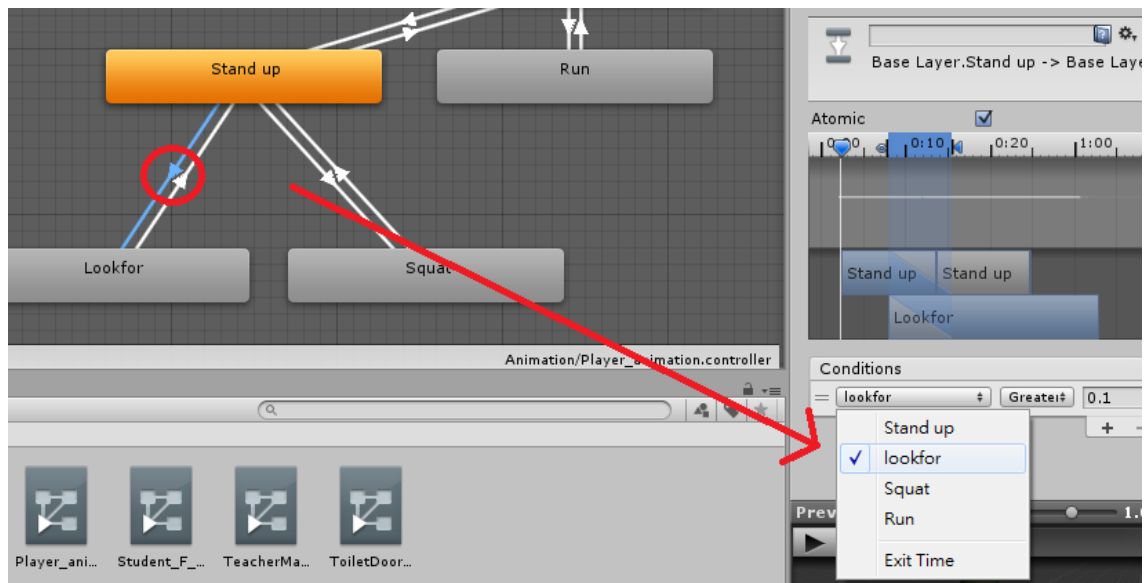


圖 5.54 關聯之程式碼設定

這樣就可以完成玩家在場景上的動畫，可以隨著程式碼的條件，而改變物件的動畫，以此仿造，整個樓層的物件設定好後，就可以完成整個場景，以此類推其他樓層也是這樣製作，在此以一樓層為主的製作介紹，到此告一段落。

第陸章 遊戲製作成果

開起遊戲後首先會出現有遊戲名稱的畫面，如圖 6.1。



圖 6.1 遊戲開始畫面

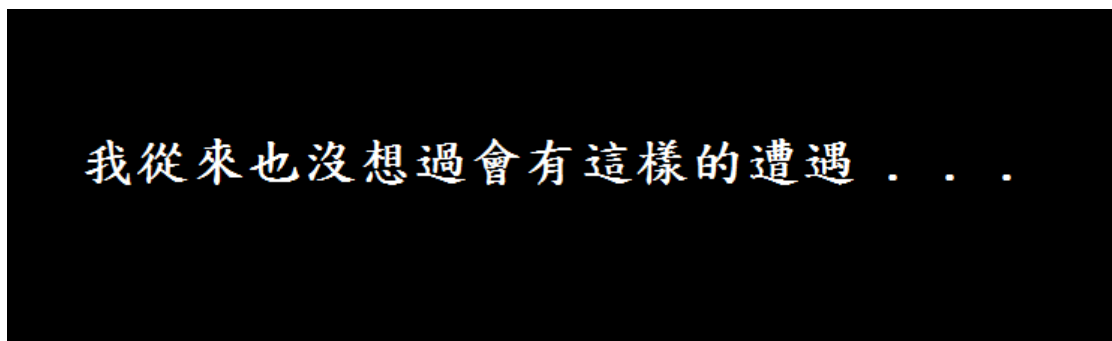


圖 6.2 遊戲劇情畫面

案 Enter 開始劇情文字，如圖 6.2，內容順序如下。

- 我從來也沒想過有這樣的遭遇...
- 大鎖生鏽的門、破舊的空教室、還有那身不見底的樓梯...
- 我們學校怎麼會有這種鳥地方
- 該死的!
- 一大早就這麼倒楣~
- 左邊牆上掛著一面樓層圖
- 嗯~~~ 如果這些教室曾經使用過的話
- 地下五樓的總務處應該是租借鑰匙的地方吧!
- 我凝視著那漆黑的樓梯.. 全身毛細孔猛顫抖...
- 「ㄟ! 都二十幾歲了，怕什麼」我故作鎮定的乾笑著
- 邁開腳步，我摸黑慢慢的往下走 當時的我殊不知...
- 這是我後悔的開始

玩家進入遊戲後會先有操作介紹，如圖 6.3。

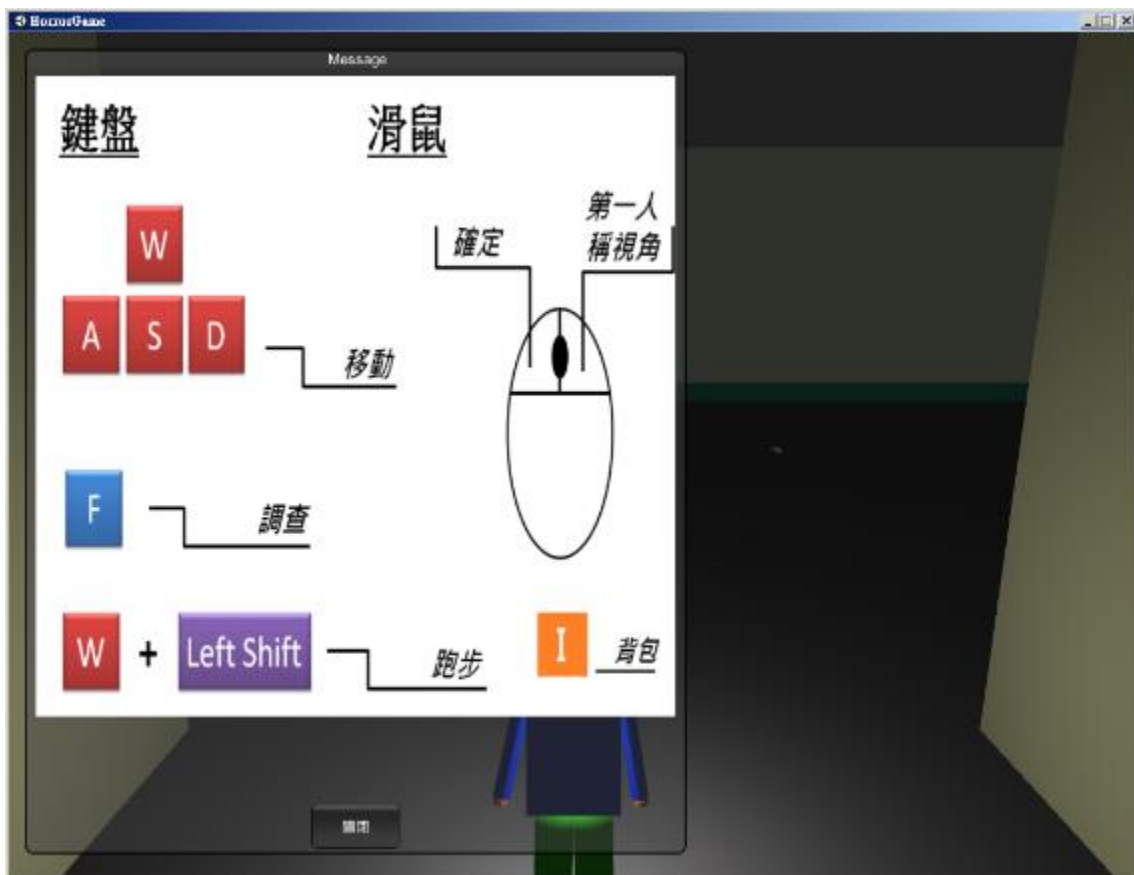


圖 6.3 遊戲操縱畫面

操作介紹關掉後，如圖 6.4，會先出現 A 跟 B 道具，A 是操作介紹，可重複閱讀，B 是教室 1 的鑰匙。

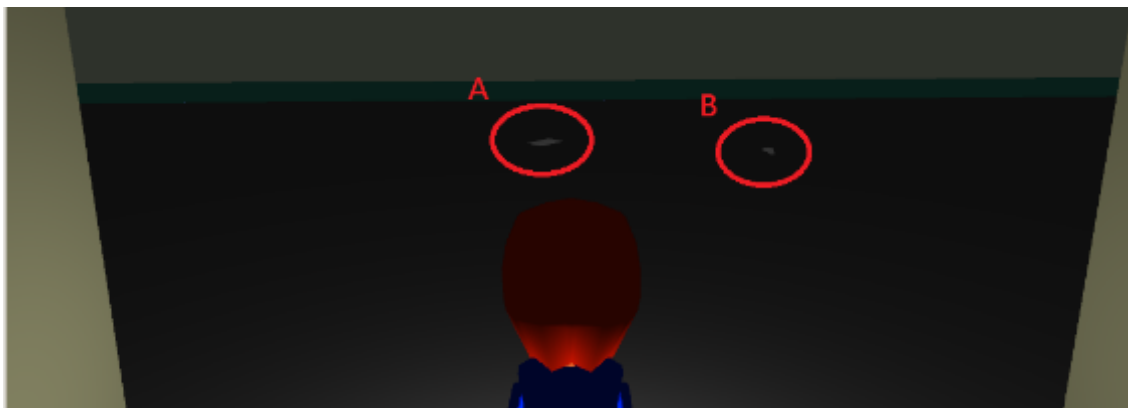


圖 6.4 樓層 B1 場景

依照圖 6.4 的左手邊會有地圖可以觀看，調查後會如圖 6.5 的畫面。



圖 6.5 樓層 B1 地圖畫面

依照劇情引導，拿到教室 1 的鑰匙後到教室 1，如圖 6.6。



圖 6.6 樓層 B1 教室 1 畫面

然後打開教室 1 的門，如圖 6.7。



圖 6.7 樓層 B1 教室 1 開門畫面

如圖 6.8，這是教室 1 內部的景觀，A 的部分是鐵櫃。

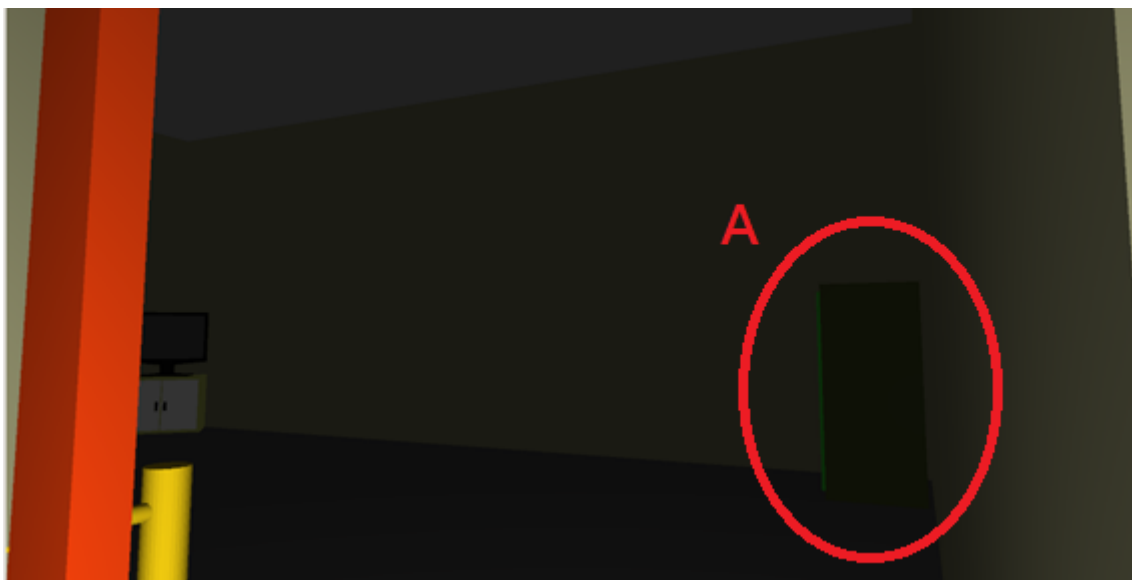


圖 6.8 樓層 B1 教室 1 內部鐵櫃

如圖 6.9，這是教室 1 內部的景觀，A 的部分是無意義紙條，被放在桌子上，B 的部分是講桌，C 的部分是電視櫃，後面牆壁上的是黑板。

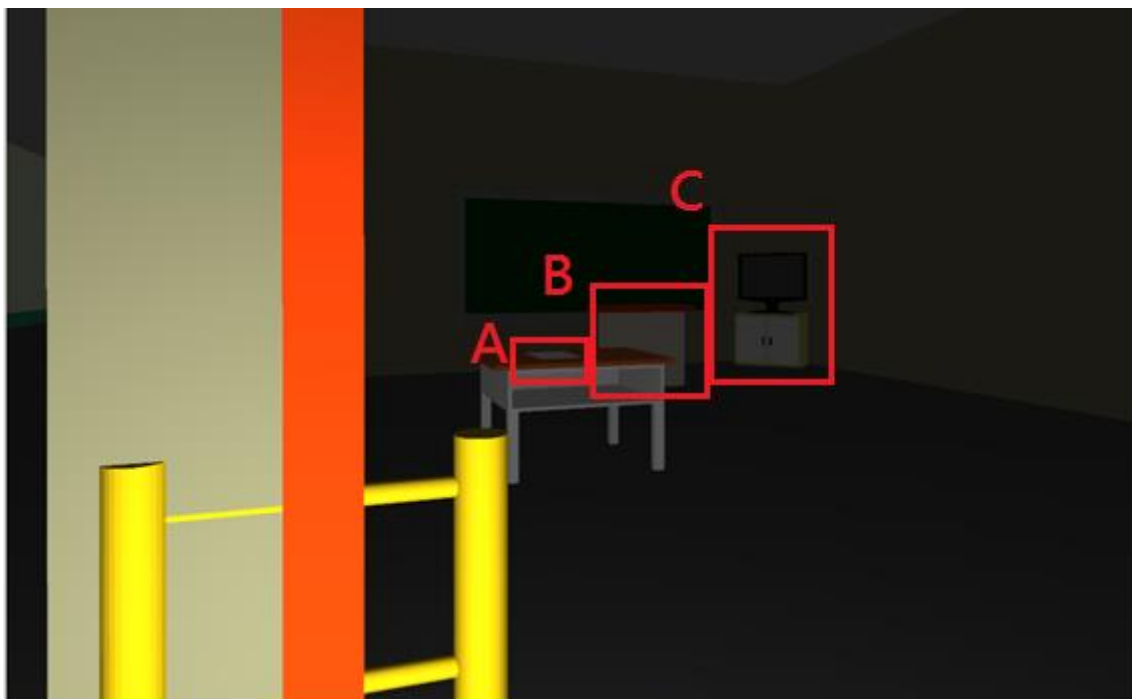


圖 6.9 樓層 B1 教室 1 內部其他擺設

如圖 6.10，調查剛剛看到的紙條，會出現遊戲內的提示訊息。

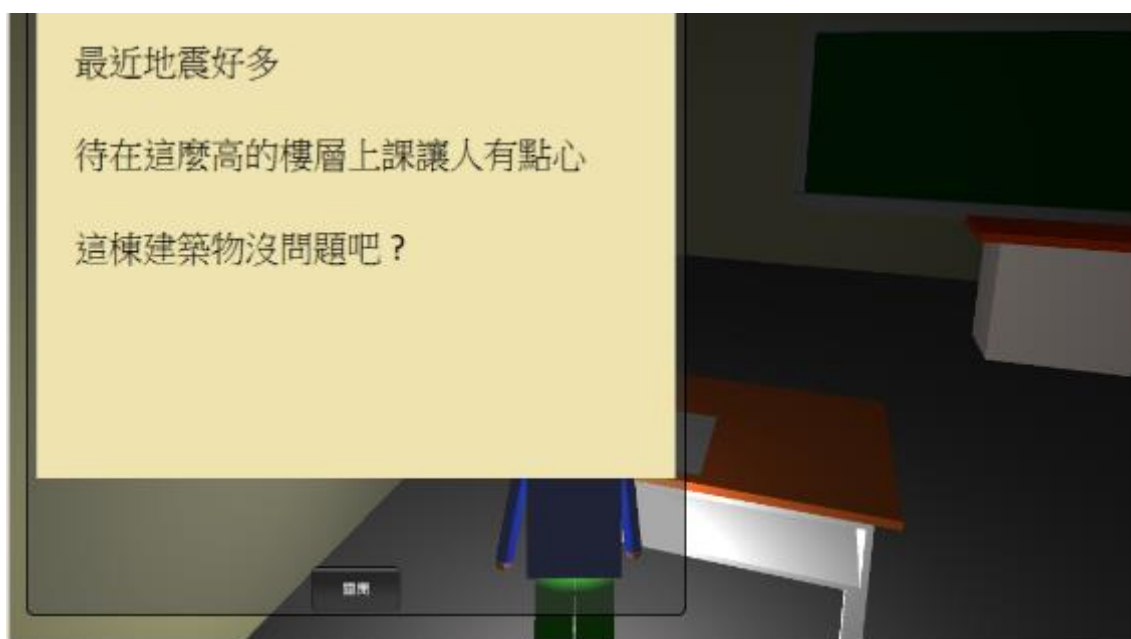


圖 6.10 樓層 B1 教室 1 內的紙條

如圖 6.11，調查剛剛看到的講桌，會得到「鐵櫃鑰匙」。

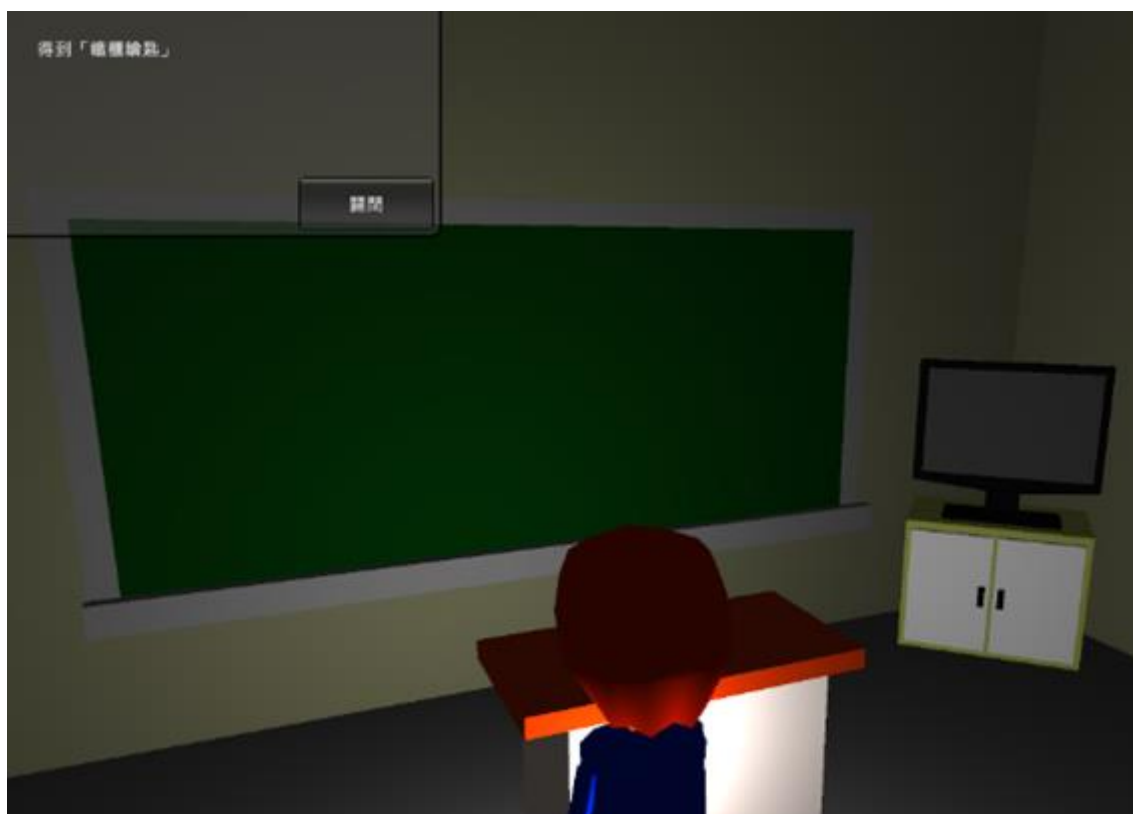


圖 6.11 樓層 B1 教室 1 內得到鐵櫃鑰匙

如圖 6.12，拿著剛剛拿到的鐵櫃鑰匙，調查鐵櫃，得到「鐵槌」。



圖 6.12 樓層 B1 教室 1 內得到鐵槌

如圖 6.13，調查電視櫃，得到「螺絲起子」。

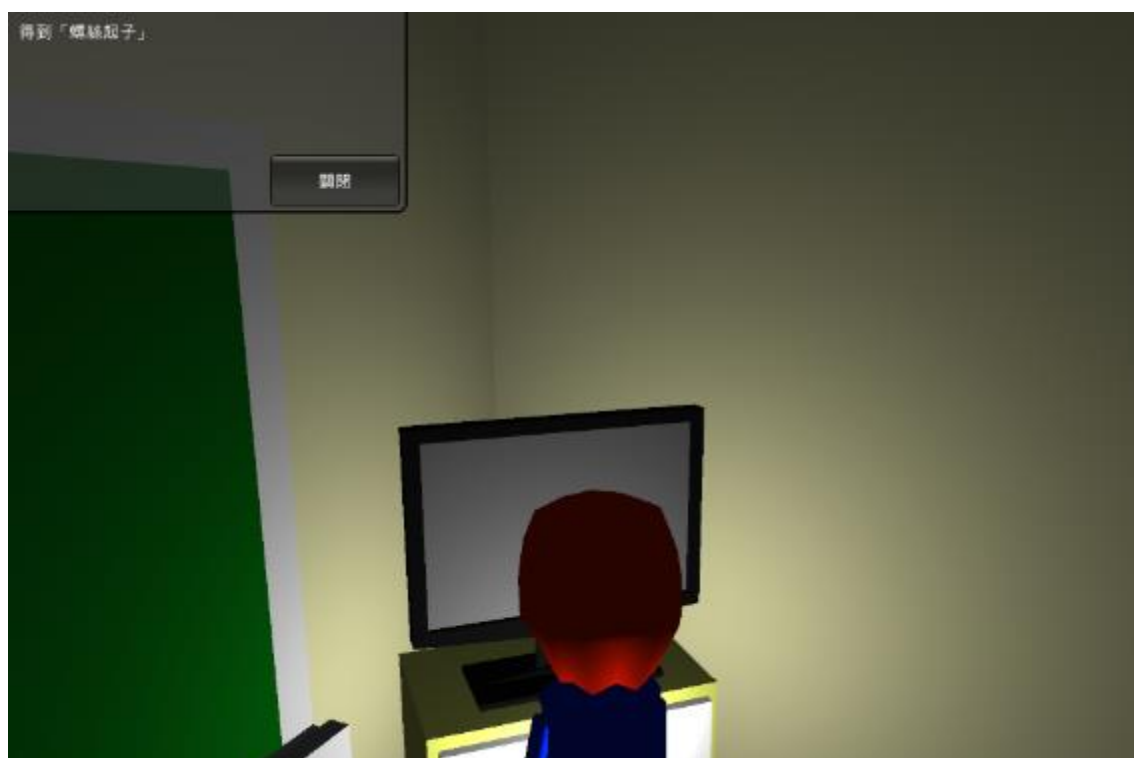


圖 6.13 樓層 B1 教室 1 內得到螺絲起子

基本上第一次玩的玩家，會對所有的物件進行調查，而剛剛講解的已經是教室 1 內所有的道具，其他教室也是如上解說的方式進行，下列將以文字的解說整個遊戲的流程順序。

在教室 1 拿完道具後離開，會收到「旁邊教室 2 傳來聲響」的訊息，將玩家引導到教室 2。



圖 6.14 樓層 B1 教室 2 畫面

如圖 6.14，玩家來到教室 2，開門時發現教室門無法打開，調查後出現「門鎖變形扭曲，只能強行破壞了」，玩家持有「鐵槌」才可以打開，這時教室 1 在鐵櫃拿到的「鐵槌」就可以打開門。

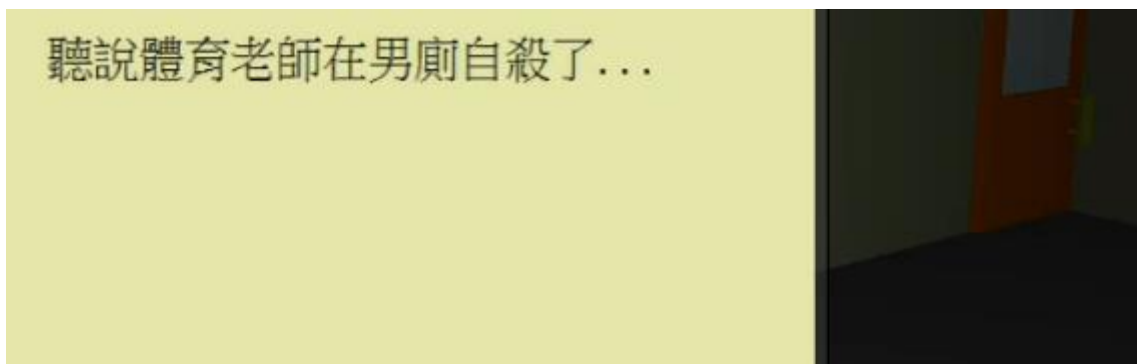


圖 6.15 樓層 B1 教室 2 內的紙條

在教室 2 內調查後方的公布欄，可獲得「螺絲釘」，然後調查地上的紙條後，會收到「聽說體育老師在男廁自殺了...」的訊息，將玩家引導到男廁，如圖 6.15。



圖 6.16 樓層 B1 男廁最後一間廁所

到了男廁後開始調查，調查到最後一間廁所，會發現屍體，調查屍體後會得到「教室 4 鑰匙」，將玩家引導到教室 4，如圖 6.16。



圖 6.17 樓層 B1 男廁離開時畫面

當玩家想離開男廁時會突然出現的幽靈，然後過一陣子，會消失，如圖 6.17。

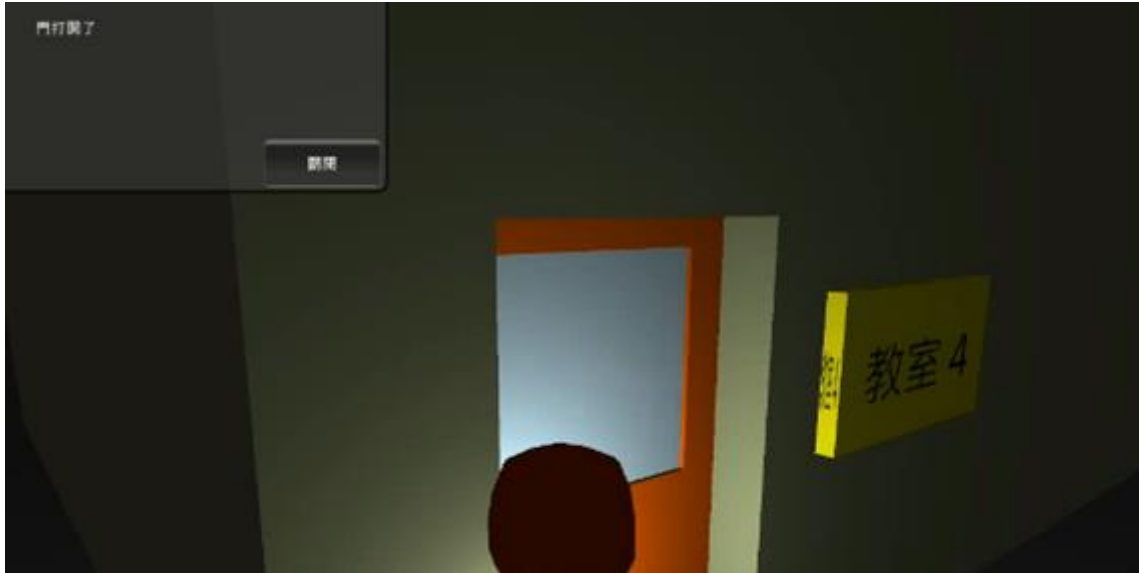


圖 6.18 樓層 B1 教室 4 畫面



圖 6.19 樓層 B1 教室 4 內部畫面

如圖 6.18 到達教室 4 後，打開門會看到一位男同學坐在中間，如圖 6.19，靠近後馬上消失，接著調查桌子上的紙條會出現，「大家都在傳說學校要倒塌了，聽說隔壁班有被老師以造謠為由抓去寫悔過書，真是可憐阿，哈哈」的訊息，如圖 6.20，將玩家引導到隔壁教室，也就是教室 3。

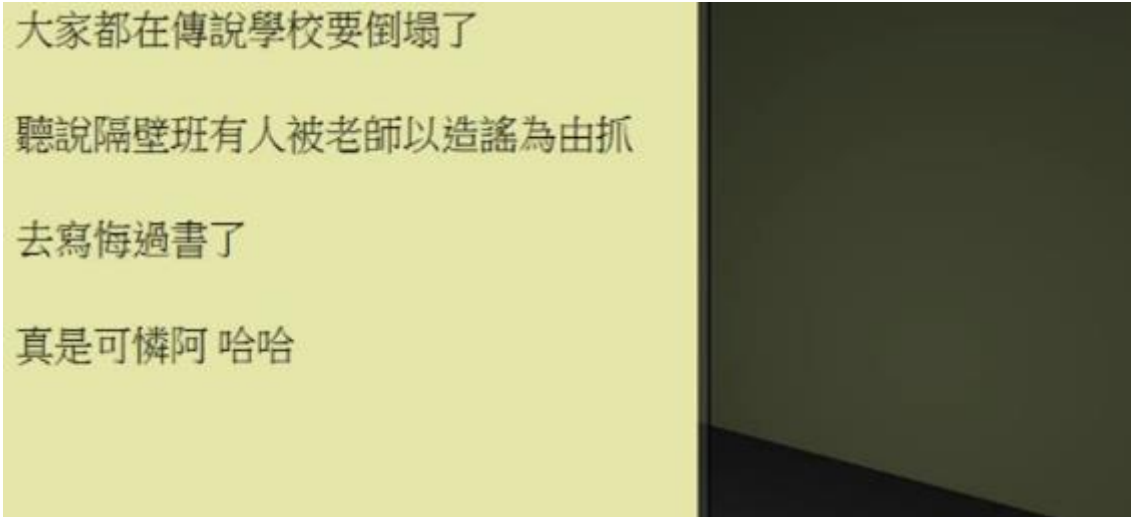


圖 6.20 樓層 B1 教室 4 內的紙條

到了教室 3 後調查門，發現其中有一扇門可以開，進去後可以調查講桌得到「粉筆」，這時只剩女廁、教室 5 跟教室 6 沒調查。

現在調查教室 6 也不會開門，所以只剩女廁跟教室 5，要先調查哪個都可以，這裡就先調查女廁，會發現某間廁所內的牆壁上有奇怪的塗鴉，如圖 6.21，調查後「粉筆」會消耗，出現「紅紅的一片，把多餘的地方擦掉吧」，接著繼續調查教室 5。



圖 6.21 樓層 B1 女廁某間廁所

調查教室 5 的門時，會出現「門把鬆脫，需要鎖緊」的訊息，如果背包內有「螺絲起子」和「螺絲釘」，則可消耗掉，並把門打開，進入教室 5 調查講桌，會獲得「板擦」，這時可以去女廁把剛剛用粉筆畫上去的部分擦掉，可以得到「把門關上」的訊息，然後玩家把門關上後，會發現門附近掛上了甚麼東西，如圖 6.22，調查後得到「教室 6 鑰匙」。



圖 6.22 樓層 B1 女廁某間廁所把門關上後

當要離開女廁時，在鏡子上會發現自己沒被映上去，且出現女鬼的影像，但是回過頭甚麼都沒看到，再回過頭看鏡子上的女鬼又不見蹤影。

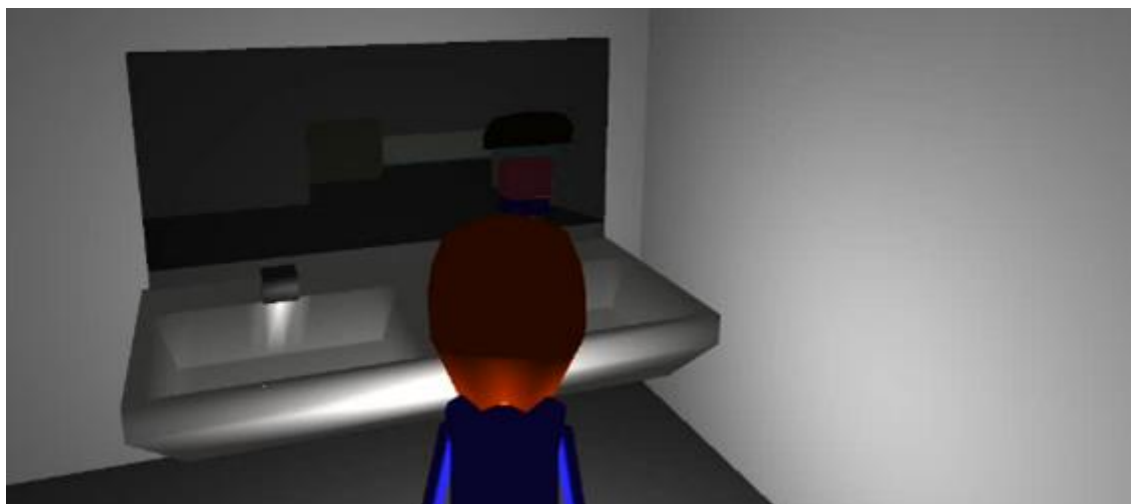


圖 6.23 樓層 B1 女廁鏡子

進到教室 6 後，可調查鐵櫃得到「鐵撬」，這時可以把 B2 樓梯口的鐵門撬開，使的鐵門消失，如圖 6.24，這樣就可以前進到樓層 B2，這樓層 B1 主要是引導玩家熟悉基本操作，不會死亡的關卡。



圖 6.24 樓層 B1 樓梯口

進入樓層 B2 後會發現，下樓梯的樓梯口被桌椅堵住，這關開始就是玩家自由探險，這邊開始只講解重要的要點。

樓層 B2 開始會加入幽靈的攻擊事件，玩家必須一邊閃躲，一邊搜尋可用的道具；首先調查地圖左上男廁，在門口數過來第二個小便斗，調查後得到「辦公室鑰匙」；調查教室 4，門沒鎖，後方置物櫃獲得「斷掉的木竿」；調查教室 1，門沒鎖，調查置物櫃獲得「沒有竿子的掃把」，這「斷掉的木竿」和「沒有竿子的掃把」兩個道具合成「掃把」，然後調查黑板上方的照相框，會得到「剪刀」。

得到「剪刀」就可以到地圖左下女廁，把屍體掛在屍體上的鑰匙剪下來，得到「教室 8 的鑰匙」，這時屍體消失，女鬼會出現開始追殺玩家，當玩家擺脫女鬼一段距離，女鬼會開始徘徊在教室 8 前面，如圖 6.24。

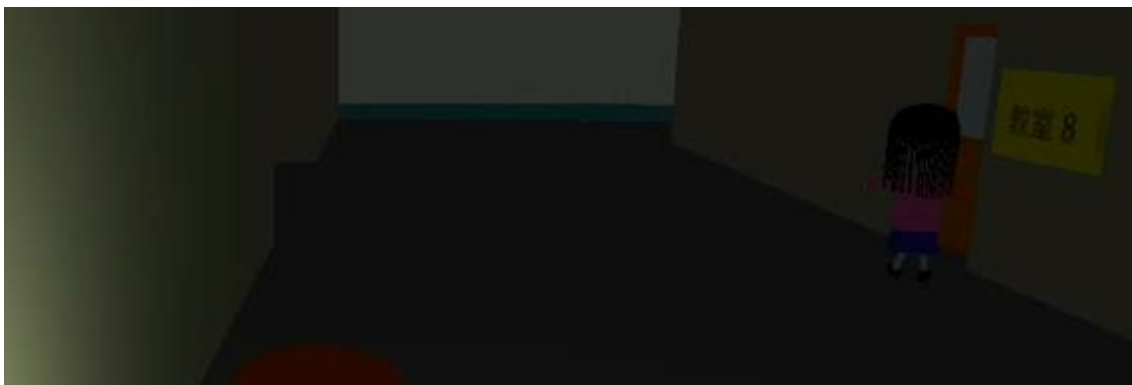


圖 6.25 樓層 B2 教室 8 外走廊

進入辦公室，在最裡面的桌子可以得到「打火機」，飲水機調查後得到「紅色的鋼筆」；進入地圖右上女廁最後一間，需要「紅色的鋼筆」才可開門，發現屍體，拿走「教室 7 鑰匙」，離開時，每間廁所衝出鬼魂追殺玩家，離開廁所時則消失；靠近教室 7 時要小心徘徊的女鬼，打開教室 7 後，調查桌子得到「手機電池」；進入地圖右下男廁，發現地上有隻手機，運用身上的「手機電池」，讓手機響起來，把徘徊在教室 8 外的女鬼引過來，過一段時間女鬼就會開始在整個走廊巡邏，要注意；進入教室 8，得到「不知名的鑰匙」。

調查後發現「不知名的鑰匙」是開啟教室 6 的鑰匙，進入教室 6，可以獲得「化學教室 1 鑰匙」、「化學教室 2 鑰匙」和「化學教室 3 鑰匙」，運用這三把鑰匙在每間化學教室最裡面的鐵櫃調查後，最後會獲得共兩個「酒精燈」。



圖 6.26 樓層 B2 下樓樓梯口

有了兩個「酒精燈」跟「打火機」就可以把堵在樓梯口的桌椅燒毀，使得消失，如圖 6.26，可以進入樓層 B3。

進入樓層 B3 會發現下樓梯時沒有阻礙，但是下去時，如圖 6.27，馬上被 B4 門口的 4 之怪物包圍殺死，出現死亡畫面，按下按鈕後回到樓層 B3。



圖 6.27 樓層 B4 樓梯口死亡

到了樓層 B3 先到福利社得到「視聽教室 1 鑰匙」和「電池」；打開視聽教室 1，調查電腦得到「開機磁片」；視聽教室 2 門沒鎖，掉查桌子上的鑰匙，可獲得「社團部室鑰匙」；社團部室打開後，調查可獲得「音樂教室 2 鑰匙」和「社團倉庫鑰匙」；打開社團倉庫後，內藏一隻怪物，需要迅速逃出，運用身上的「電池」把在男廁地上的鬧鐘啟動，把怪物吸引過去，如圖 6.28，然後繞回去把社團倉庫內的「音響連接線」拿走。

前往開啟音樂教室 2 的同時，鬧鐘的聲響就會停止，怪物會開始巡邏，要多加注意，音樂教室 2 內可以獲得「音樂教室 1 鑰匙」。



圖 6.28 樓層 B3 男廁

開啟音樂教室 1，調查可獲得「音樂光碟」和「表演廳鑰匙」；打開表演廳後，會發現有台電腦需要「開機磁片」啟動電腦，放入「音樂光碟」，連接「音響連接線」至喇叭即可播放音樂，把 B4 門口堵住的怪物們引到表演廳內，如圖 6.29，這時從後方的逃生門逃出，前往 B4。



圖 6.29 樓層 B3 表演廳

樓層 B3 加入怪物，玩家將在此層樓收集到遇難學生留下的訊息瞭解怪物的習性，怪物擁有敏銳的聽力及較差的視力，玩家需要製造聲響來引誘怪物前往某些地方，像是用手錶鬧鐘引開一隻怪物、用音樂引來一群怪物。

進入樓層 B4 時，旁邊通往樓層 B5 的樓梯被鐵門擋住，需要啟動電源才能打開，樓梯前方的牆壁裝有消防設備，調查後出現「沒有電力」的訊息，過了一陣子會出現「樓梯傳來怪異的吼叫聲」的訊息，這時樓梯口會出現怪物在中間走廊巡邏，共會發生此事件兩次，共兩隻怪物巡邏，然後還有多出來的一隻怪物守在鐵門前，要多加注意。

樓層 B4 教室 1 旁飲水機附近，撿到「磁卡 2」，前往開啟教室 2 後怪物會因為開啟時發現聲響，而前來此教室查看，如圖 6.30，每間教室都是這樣，因此要多加注意，調查教室 2 後會得到「磁卡 4」，可以啟動「緊急供電系統 2」，以此類推其他四間教室也是一樣。



圖 6.30 樓層 B4 教室 2

當四間教室的緊急供電系統都啟動後，可以到辦公室將總電源開啟，鐵門就可以打開前往下一個樓層，樓層 B4 其中有兩間廁所，各一個鬧鐘，有機會把三隻怪物都吸引十幾秒鐘的機會，去開起鐵門。

開起鐵門後，進入樓層 B5 後，因為恢復供電的消防設備警鈴大作，所有的怪物都醒了，這時樓梯口就會開始會開始每兩分鐘產生一隻怪物，而且不會停止，這時開始跟時間賽跑，一邊躲避大量怪物的巡邏，一邊尋找頂樓的鑰匙，逃出這鬼怪的謎窟。

首先前往男廁，獲得「輔導室鑰匙」後，會出現不會動的男性鬼魂，女廁可獲得「教務處鑰匙」，輔導室與教務處主要是給玩家躲避怪物用的房間，沒有其他道具，下一個是圖書館，獲得「教職員室鑰匙」，教職員室可以獲得「教官室鑰匙」；到了教官室，可在最裡面的鐵櫃獲得「鐵鎚」，有了「鐵鎚」可以把總務處已經變形的門給破壞，進入總務處內調查後，獲得「頂樓鑰匙」，這時會有一隻怪物衝進總務處，玩家必須擺脫，衝出總務處後，一路衝到樓梯口。

我衝上樓梯間 . . .

圖 6.31 遊戲結局

到達樓梯口後，如圖 6.31，遊戲進入結束之前，還有劇情如下。

- 我衝上樓梯間...
- 頭也不回的拼命往上跑
- 怪物的聲音迴盪在我周圍
- 我沒有回頭的勇氣
- 『我要活下去』
- 腦袋裡剩下的唯一念頭
- 連滾帶爬的我回到了地表，顫抖的雙手扭開了上鎖的大門
- 用盡力氣往建築外面跑
- 插在門把上的鑰匙，我已經沒辦法管它了...
- 我將這件事埋藏在心底，沒有向任何人提起
- 在我即將畢業的這段時間裡，學校又傳出了幾起學生神祕失蹤的案件
- 校方介入調查後仍毫無所獲...
- 離開校園的我知道
- 這個祕密將纏著我一輩子，直到老死...

第柒章 結論

本專題製作一款結合解謎與恐怖的 3D 遊戲，整個劇情背景是以玩家困在校園內的體育館，意外間發現通往地下建築的樓梯，而玩家為了脫困，必須尋找能夠通過地下樓梯的道具。

整個地下室有五層，在玩家解謎的過程中引導，了解遊戲劇情，並發現這個地下室，其實是以前的學校建築，因某日的大地震，加上校方的建築工程灌水，而地層下陷活埋了當時的在校學員，形成了現在體育館的地下室，在此校方為了隱蔽這件事情造成的負面影響，才會在上方蓋上體育館。

整個遊戲的開發分成「設計」、「製作」與「微調」三大主軸，其中的「製作」以 Blender 與 Unity 軟體為主，Blender 進行 3D 模組的建製，這當中的 3D 模型可以運用當前最新的 3D 列印技術印出，而 Unity 是將模組放入場景，進行腳本撰寫，完成遊戲的主要開發平台。

在製作與開發的過程中，學習與了解到完成一個作品，所需要的要素，當然不單單只有遊戲，像是程式開發與專案處理等，都是通用的學習架構，然而遊戲的開發可以學習到劇情編排、遊戲架構、系統分析、邏輯分析、3D 建模、程式撰寫等多元學習。

本專題所完成的遊戲，雖然系統架構等內容完整，但對於建模的美術方面沒有專業的知識，所以對於模型的美術還有很大的修改空間，若未來能以專業的美術知識結合到此遊戲的話，想必能夠讓此恐怖遊戲更加的恐怖更加的貼近玩家。

參考文獻

- [1] Unity - Fast Facts (民 104 年 04 月 01 日)。Unity - Game Engine。民 104 年 04 月 01 日，取自：<https://unity3d.com/cn/public-relations>
- [2] 恐怖遊戲 - 維基百科，自由的百科全書 (民 104 年 03 月 21 日)。維基百科，自由的百科全書。民 104 年 04 月 01 日，取自：
<http://zh.wikipedia.org/wiki/恐怖遊戲>
- [3] Blender 動畫設計- 陳鍾誠的網站 (民 101 年 04 月 27 日)。Wikidot - Free and Pro Wiki Hosting。民 104 年 04 月 01 日，取自：
<http://ccckmit.wikidot.com/3d:main>
- [4] Windows 7 - 維基百科，自由的百科全書 (民 103 年 04 月 13 日)。維基百科，自由的百科全書。民 103 年 05 月 25 日，取自：
http://zh.wikipedia.org/wiki/Windows_7
- [5] Unity 遊戲引擎教學 - CG 數位學習網 (民 103 年 05 月 25 日)。CG 數位學習網。民 103 年 05 月 25 日，取自：<http://www.cg.com.tw/Unity>
- [6] C# - 維基百科，自由的百科全書 (民 103 年 11 月 15 日)。維基百科，自由的百科全書。民 103 年 11 月 15 日，取自：<http://zh.wikipedia.org/wiki/C#>
- [7] Blender - 維基百科，自由的百科全書 (民 103 年 11 月 15 日)。維基百科，自由的百科全書。民 103 年 11 月 15 日，取自：
<http://zh.wikipedia.org/wiki/Blender>
- [8] Moonikai 的遊戲時間：229.心靈旅館 (上) - YouTube (民 102 年 11 月 17 日)。YouTube。民 103 年 05 月 25 日，取自：<http://youtu.be/qERKIPZBE10>

附錄 1 門物件的程式碼

```
public class Door_All : MonoBehaviour {
    public bool needKey ;
    public string keyName;
    public string doorMessage1;
    public string doorMessage2;
    private bool ckeckKey ;
    private bool DoorOn;
    private bool DisplayDialog = false;
    public Animator m_door_animation;
    // Use this for initialization
    void Start () {
    }
    // Update is called once per frame
    void Update () {
    }
    void OnGUI()
    {
        if(DisplayDialog == true)
        {
            int door_msg=0;
            Player_control.mouseIsLock=false;
            GUI.BeginGroup(new Rect((Screen.width -400)/2,
(Screen.height -200)/2, 400, 200));
            GUI.Box(new Rect(0,0,400,200),"");
            if(ckeckKey == true)
            {
                GUI.Label(new Rect((400-230)/2, (200-30)/2, 230, 30),
doorMessage2);
            }
            else
            {
                GUI.Label(new Rect((400-230)/2, (200-30)/2, 230, 30),
doorMessage1);
            }
        }
    }
}
```

```

    }
    if(GUI.Button(new Rect(300, 170, 100, 30), "關閉"))
    {
        DisplayDialog = false;
        Player_control.mouseIsLock= true;
    }
    GUI.EndGroup();
}

void OnTriggerEnter(Collider other)
{
    if(other.tag.CompareTo("ammo") == 0)
    {
        if(needKey == false)
        {
            DoorMoveOn();
        }
        else
        {
            SearchKey();
        }
    }
}

void DoorMoveOn()
{
    m_door_animation = this.GetComponent<Animator>();
    if(DoorOn==false)
    {
        DoorOn = true;
        //Debug.Log ("Open");
        m_door_animation.Play("Open");
    }
    else if(DoorOn == true)
    {
        DoorOn = false;
    }
}

```

```

        //Debug.Log ("Close");
        m_door_animation.Play("Close");
    }
}
void SearchKey()
{
    int getKey=0;
    for(int i=0; i<PlayerInventoryGUI.itemArray.Length; i++)
    {
        getKey = PlayerInventoryGUI.itemArray[i].IndexOf(keyName);
        if(getKey >= 0)
        {
            PlayerInventoryGUI.itemArray[i]="";
            needKey = false;
            ckeckKey = true;
        }
        else
        {
            DisplayDialog = true;
        }
    }
}
}

```

附錄 2 檢視道具的程式碼

```
public class Item_All : MonoBehaviour {
    public string item_name;
    public bool BeDel ;
    public string itemMessage;
    public bool needKey ;
    public string keyName;
    public string lockMessage;
    private bool BePickup = false;
    bool DisplayDialog = false;
    bool betalk = false;
    // Use this for initialization
    void Start () {
    }
    // Update is called once per frame
    void Update () {
    }
    void OnGUI()
    {
        if(DisplayDialog == true && betalk == false)
        {
            Player_control.mouseIsLock=false;
            GUI.BeginGroup(new Rect((Screen.width -400)/2,
(Screen.height -200)/2, 400, 200));
            GUI.Box(new Rect(0,0,400,200),"");
            if(needKey == true)
            {
                GUI.Label(new Rect((400-230)/2, (200-30)/2, 230, 30),
lockMessage);
            }
            else
            {
                GUI.Label(new Rect((400-230)/2, (200-30)/2, 230, 30),
itemMessage);
            }
        }
    }
}
```

```

    }
    if(GUI.Button(new Rect(300, 170, 100, 30), "關閉"))
    {
        DisplayDialog = false;
        Player_control.mouseIsLock= true;
        BeTouch();
    }
    GUI.EndGroup();
}

void OnTriggerEnter(Collider other)
{
    if(other.tag.CompareTo("ammo") == 0)
    {
        if(needKey == true)
        {
            SearchKey();
        }
        else
        {
            BeTalk();
        }
    }
}

void BeTouch()
{
    if(BePickup==false && needKey ==false)
    {
        BePickup = true;
        betalk = true;
        PlayerInventoryGUI.ItemSet(item_name);
        PlayerInventoryGUI.BagClean();
        if(BeDel == true)
        {
            Destroy(gameObject);
        }
    }
}

```

```

        }
    }
}

void BeTalk()
{
    DisplayDialog = true;
}

void SearchKey()
{
    int getKey=0;
    for(int i=0; i<PlayerInventoryGUI.itemArray.Length; i++)
    {
        getKey = PlayerInventoryGUI.itemArray[i].IndexOf(keyName);
        if(getKey >= 0)
        {
            PlayerInventoryGUI.itemArray[i]="";
            needKey = false;
        }
        else
        {
            DisplayDialog = true;
        }
    }
}
}

```


附錄 3 顯示圖片的程式碼

```
public class Map_Level : MonoBehaviour {  
    private bool guiShow = false;  
    public Texture mapPic = null;  
    private Rect textRect = new Rect(Screen.width/3.4f, Screen.height/5, 300, 300);  
    // Use this for initialization  
    void Start () {  
    }  
    // Update is called once per frame  
    void Update () {  
    }  
    void OnGUI()  
    {  
        if(guiShow == true)  
        {  
            GUI.DrawTexture(textRect, mapPic);  
        }  
    }  
    void OnTriggerEnter(Collider other)  
    {  
        if(other.tag == "ammo")  
        {  
            guiShow = true;  
        }  
    }  
    void OnTriggerExit(Collider other)  
    {  
        if(other.tag == "Player")  
        {  
            guiShow = false;  
        }  
    }  
}
```

附錄 4 調查子彈的程式碼

```
public class Ammo : MonoBehaviour {  
    public float m_liveTime = 1;  
    protected Transform m_transform;  
    // Use this for initialization  
    void Start () {  
        m_transform = this.transform;  
    }  
    // Update is called once per frame  
    void Update () {  
        m_liveTime -= Time.deltaTime;  
        if(m_liveTime <= 0)  
            Destroy(this.gameObject);  
    }  
    void OnTriggerEnter(Collider other)  
    {  
        if(other.tag.CompareTo("item") == 0)  
            Destroy(this.gameObject);  
    }  
}
```

附錄 5 攝影機設定的程式碼

```
[AddComponentMenu("Camera-Control/Mouse Look")]
public class MouseLook : MonoBehaviour {
    public enum RotationAxes { MouseXAndY = 0, MouseX = 1, MouseY = 2 }
    public RotationAxes axes = RotationAxes.MouseXAndY;
    public float sensitivityX = 15F;
    public float sensitivityY = 15F;
    public float minimumX = -360F;
    public float maximumX = 360F;
    public float minimumY = -60F;
    public float maximumY = 60F;
    float rotationY = 0F;
    void Update ()
    {
        if (axes == RotationAxes.MouseXAndY)
        {
            float rotationX = transform.localEulerAngles.y +
Input.GetAxis("Mouse X") * sensitivityX;
            rotationY += Input.GetAxis("Mouse Y") * sensitivityY;
            rotationY = Mathf.Clamp (rotationY, minimumY, maximumY);
            transform.localEulerAngles = new Vector3(-rotationY, rotationX,
0);
        }
        else if (axes == RotationAxes.MouseX)
        {
            transform.Rotate(0, Input.GetAxis("Mouse X") * sensitivityX, 0);
        }
        else
        {
            rotationY += Input.GetAxis("Mouse Y") * sensitivityY;
            rotationY = Mathf.Clamp (rotationY, minimumY, maximumY);

            transform.localEulerAngles = new Vector3(-rotationY,
transform.localEulerAngles.y, 0);
```

```
        }  
    }  
    void Start ()  
    {  
        // Make the rigid body not change rotation  
        if (rigidbody)  
            rigidbody.freezeRotation = true;  
    }  
}
```

附錄 6 玩家移動的程式碼

```
using UnityEngine;
using System.Collections;

public class Player_control : MonoBehaviour {
    public Transform m_transform;
    CharacterController m_ch;
    float m_movSpeed = 3.0f;
    float m_runSpeed = 10.0f;
    float m_gravity = 2.0f;
    private Animator m_animator;
    public static bool mouseIsLock = true;
    // Use this for initialization
    void Start () {
        m_transform = this.transform;
        m_ch = this.GetComponent<CharacterController>();
        m_animator = this.GetComponent<Animator>();
        Screen.lockCursor = true;
    }
    // Update is called once per frame
    void Update () {
        Control();
        MouseLockon();
    }
    void Control()
    {
        float xm=0, ym=0, zm=0;
        ym -= m_gravity * Time.deltaTime;
        if(Input.GetKey(KeyCode.W))
        {
            if(Input.GetKey(KeyCode.LeftShift))
            {
                m_animator.Play("Run");
                zm += m_runSpeed * Time.deltaTime;
            }else
```

```

        {
            m_animator.Play("Walk");
            zm += m_movSpeed * Time.deltaTime;
        }
    }
    if(Input.GetKey(KeyCode.S))
    {
        zm -= m_movSpeed * Time.deltaTime;
        m_animator.Play("Walk");
    }
    if(Input.GetKey(KeyCode.A))
    {
        xm -= m_movSpeed * Time.deltaTime;
        m_animator.Play("Walk");
    }
    if(Input.GetKey(KeyCode.D))
    {
        xm += m_movSpeed * Time.deltaTime;
        m_animator.Play("Walk");
    }
    m_ch.Move(m_transform.TransformDirection(new
Vector3(xm,ym,zm)));
    if(Input.GetKeyDown(KeyCode.F))
    {
        m_animator.Play("Lookfor");
    }
    if(Input.GetKey(KeyCode.LeftControl))
    {
        m_animator.Play("Squat");
    }
}
void MouseLockon()
{
    if(mouseIsLock==true)
        Screen.lockCursor = true;
}

```

```
        if(mouseIsLock==false)
            Screen.lockCursor = false;
    }
}
```

附錄 7 背包顯示的程式碼

```
public class PlayerInventoryGUI : MonoBehaviour {
    private bool inventoryWindowOn = false;
    private Rect inventoryWindowRect = new Rect(Screen.width/3, 100,400, 400);
    public static string[] itemArray = new string[6];
    static int itemArrayCount = 0;
    // Use this for initialization
    void Start () {
        for(int i=0; i < itemArray.Length; i++)
        {
            itemArray[i] = "";
        }
    }
    // Update is called once per frame
    void Update () {
        if(Input.GetKeyDown(KeyCode.I) && inventoryWindowOn == false)
        {
            inventoryWindowOn = true;
            BagClean();
        }
        else if(Input.GetKeyDown(KeyCode.I) && inventoryWindowOn ==
true)
        {
            inventoryWindowOn = false;
        }
    }
    void OnGUI()
    {
        if(inventoryWindowOn)
        {
            GUI.Window(0, inventoryWindowRect,
InventoryWindowMethod, "背包");
        }
    }
}
```



```

void InventoryWindowMethod(int windowId)
{
    GUILayout.BeginArea(new Rect(10,50, Screen.width/2,
Screen.height/2));
    GUILayout.BeginHorizontal();
    GUILayout.Button (itemArray[0], GUILayout.Height(50));
    GUILayout.Button (itemArray[1], GUILayout.Height(50));
    GUILayout.Button (itemArray[2], GUILayout.Height(50));
    GUILayout.EndHorizontal();
    GUILayout.BeginHorizontal();
    GUILayout.Button (itemArray[3], GUILayout.Height(50));
    GUILayout.Button (itemArray[4], GUILayout.Height(50));
    GUILayout.Button (itemArray[5], GUILayout.Height(50));
    GUILayout.EndHorizontal();
    GUILayout.EndArea();
}

public static void ItemSet(string a)
{
    itemArray[itemArrayCount] = a;
    itemArrayCount++;
}

public static void BagClean()
{
    string temp="";
    for(int i=0; i < itemArray.Length; i++)
    {
        for(int j=i; j < itemArray.Length; j++)
        {
            if(itemArray[i] == "")
            {
                if(itemArray[j] != "")
                {
                    temp = itemArray[j];
                    itemArray[j] = itemArray[i];
                    itemArray[i] = temp;
                }
            }
        }
    }
}

```

```

        continue;
    }
}
}
}
for(int i=0; i< itemArray.Length; i++)
{
    if(itemArray[i] == "")
    {
        itemArrayCount = i;
        Debug.Log ("目前可以存的陣列數" + i);
        break;
    }
}
}
/*public static string ItemCheak()
{
    return itemArray[0] + " "+itemArray[1] + " "+itemArray[2] + "
"+itemArray[3] + " "+itemArray[4] + " "+itemArray[5];
}*/
}

```

附錄 8 調查功能的程式碼

```
public class Search_for : MonoBehaviour {

    public Transform m_ammo;
    Transform m_transform;
    float m_rocketRate = 0;

    // Use this for initialization
    void Start () {
        m_transform = this.transform;
    }
    // Update is called once per frame
    void Update () {
        m_rocketRate -= Time.deltaTime;
        if(m_rocketRate <=0)
        {
            m_rocketRate = 0.1f;

            if(Input.GetKey(KeyCode.F))
            {
                Instantiate(m_ammo, m_transform.position,
m_transform.rotation);
            }
        }
    }
}
```