

自主學習 成果發表簡報

桃園市立龍潭高級中等學校



成員：
電機科 徐翊傑 李鎮均
張昀淇 謝閔鴻

介紹

我們製作的遊戲平台是名叫**roblox**，他是個大型多人線上遊戲，屬於沙盒類遊戲，可以自己製作遊戲，裡面有多種工具能夠使用，在**roblox**的世界裡每個人都是一個積木人，可以裝扮自己的人物，使用他進入各種遊戲裡。



動機

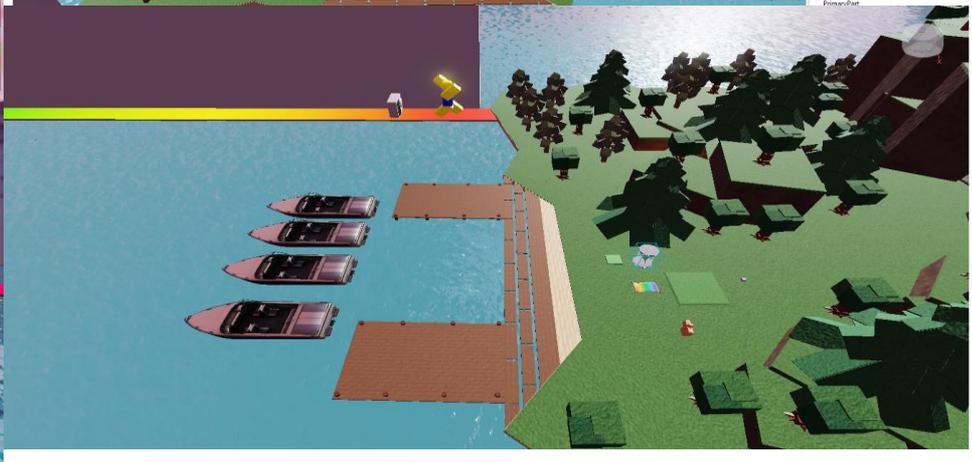
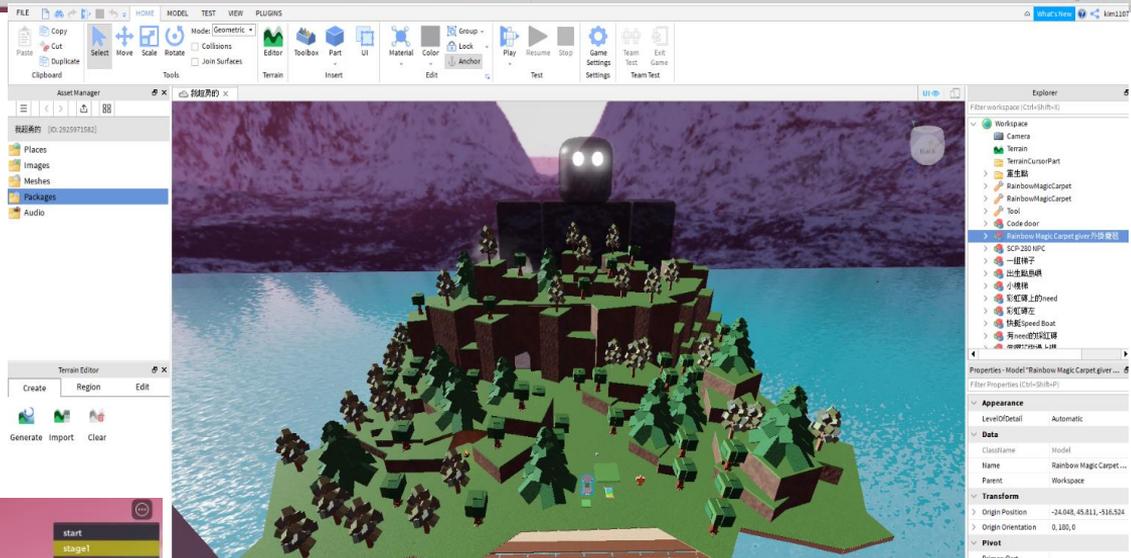
因為喜歡玩roblox上面的遊戲，有跑酷、鬼抓人，槍戰、恐怖類型等等的，想說自己也來製作看看



製作方式

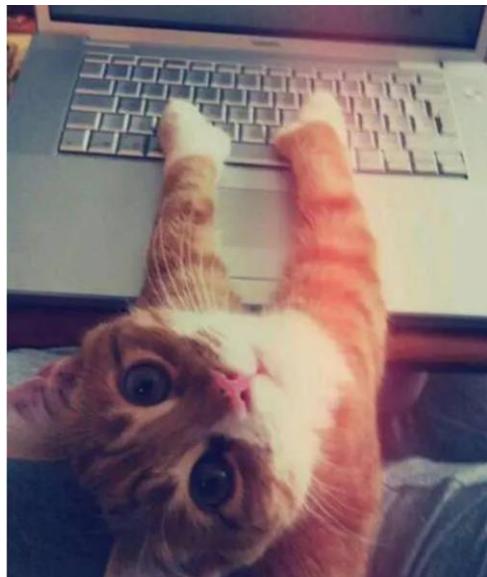
我們先討論一個題目，後開始使用Roblox studio製作，我們把地圖大概的模組做出來，之後再開始製作關卡的部分，關卡完成後決定要加入程式的關卡，再經過不斷的測試後，最終再以遊戲的方式呈現給各位。

製作過程



```
1  --取得玩家按了那些INPUT(鍵盤 滑鼠)的Service
2  local UIS = game:GetService("UserInputService")
3  --定義tool這個項目
4  local tool = script.Parent
5
6  local character
7  local humanoid
8
9  --是否可以二段跳
10 local canDoubleJump = false
11
12 --是否已經二段跳
13 local hasDoubleJumped = false
14 --初始跳躍力道
15 local oldPower
16 --二段跳等待時間
17 local TIME_BETWEEN_JUMPS = 0.2
18 --跳躍力度倍數
19 local DOUBLE_JUMP_POWER_MULTIPLIER = 4
20
21 --必須斷線且清空的連接
22 local stateChangedConn
23 local jumpRequestConn
24
```

二段跳程式P.1



二段跳程式P.2

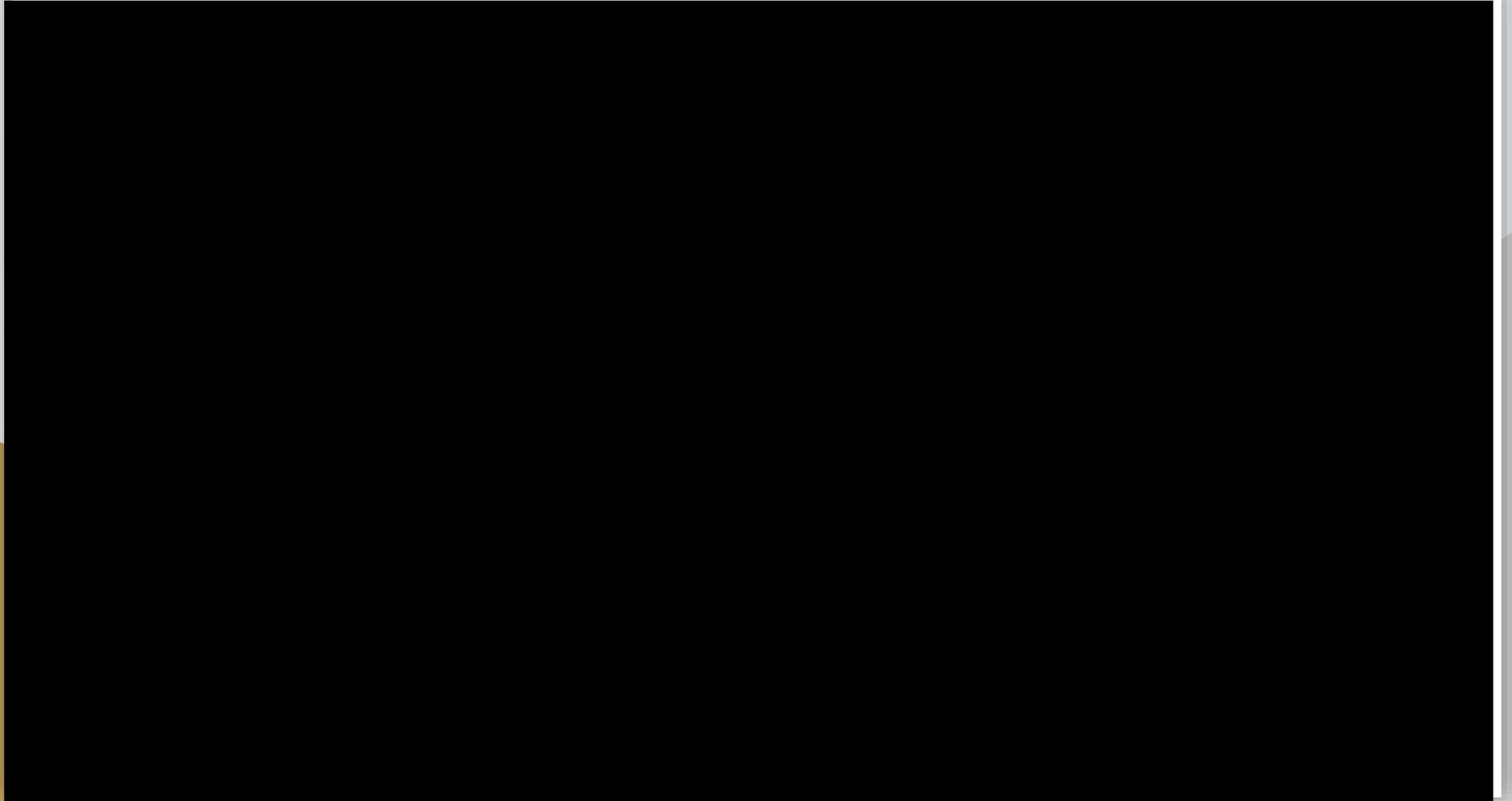
```
25 -->>功能:
26 function onJumpRequest()
27     print("玩家拿者裝備跳起來了")
28     -- 檢查玩家是否處在無法二段跳的狀態
29     if not character or not humanoid or
30         not character:IsDescendantOf(workspace) or
31         humanoid:GetState() == Enum.HumanoidStateType.Dead then
32         return
33     end
34
35     -- 當玩家可以二段跳且還沒二段跳時就讓玩家二段跳
36     if canDoubleJump and not hasDoubleJumped then
37         -- 玩家已經二段跳(hasDoubleJumped) 更變為 true
38         hasDoubleJumped = true
39         humanoid.JumpPower = oldPower * DOUBLE_JUMP_POWER_MULTIPLIER
40         -- 把玩家的狀態改成正在跳躍
41         humanoid:ChangeState(Enum.HumanoidStateType.Jumping)
42     end
43 end
44 -- 當玩家狀態改變時觸發
45 function onStateChanged(oldState, newState)
46     -- 當玩家在掉落(Freefall)的狀態時
47     if newState == Enum.HumanoidStateType.Freefall then
48         -- 讓玩家等待一段時間
49         wait(TIME_BETWEEN_JUMPS)
50         -- 讓玩家可以二段跳
51         canDoubleJump = true
52     elseif newState == Enum.HumanoidStateType.Landed then
```



二段跳程式P.3

```
52  elseif newState == Enum.HumanoidStateType.Landed then
53      --重置玩家二段跳狀態
54      canDoubleJump = false
55      hasDoubleJumped = false
56      humanoid.JumpPower = oldPower
57  end
58  end
59
60  --當玩家裝備 Tool 時觸發
61  local function toolEquipped()
62      print("裝備")
63      --玩家按下跳躍按鍵時的 Event 事件
64      jumpRequestConn = UIS.JumpRequest:Connect(onJumpRequest)
65
66      --為先前建立的變數賦值
67      character = tool.Parent
68      humanoid = character.WaitForChild("Humanoid")
69      --玩家一開始的跳躍力度
70      oldPower = humanoid.JumpPower
71
72      --玩家狀態改變的 Event
73      stateChangedConn = humanoid.StateChanged:Connect(onStateChanged)
74  end
75
76  --當玩家脫下tool時觸發
77  local function toolUnequipped()
78      print("脫下")
79      --主動斷開(Disconnect)和清空 JumpRequest 這一個 Event
80      jumpRequestConn:Disconnect()
81      jumpRequestConn = nil
82
83      --主動斷開(Disconnect)和清空 StateChanged 這一個 Event
84      stateChangedConn:Disconnect()
85      stateChangedConn = nil
86  end
87
88  tool.Equipped:Connect(toolEquipped)
89  tool.Unequipped:Connect(toolUnequipped)
90
```





跑酷-大家都愛跑酷

創作者：@kim11070

成品展示



★
從最愛移除

📶
追蹤

👍
3

👎
0

介紹

商店

說明

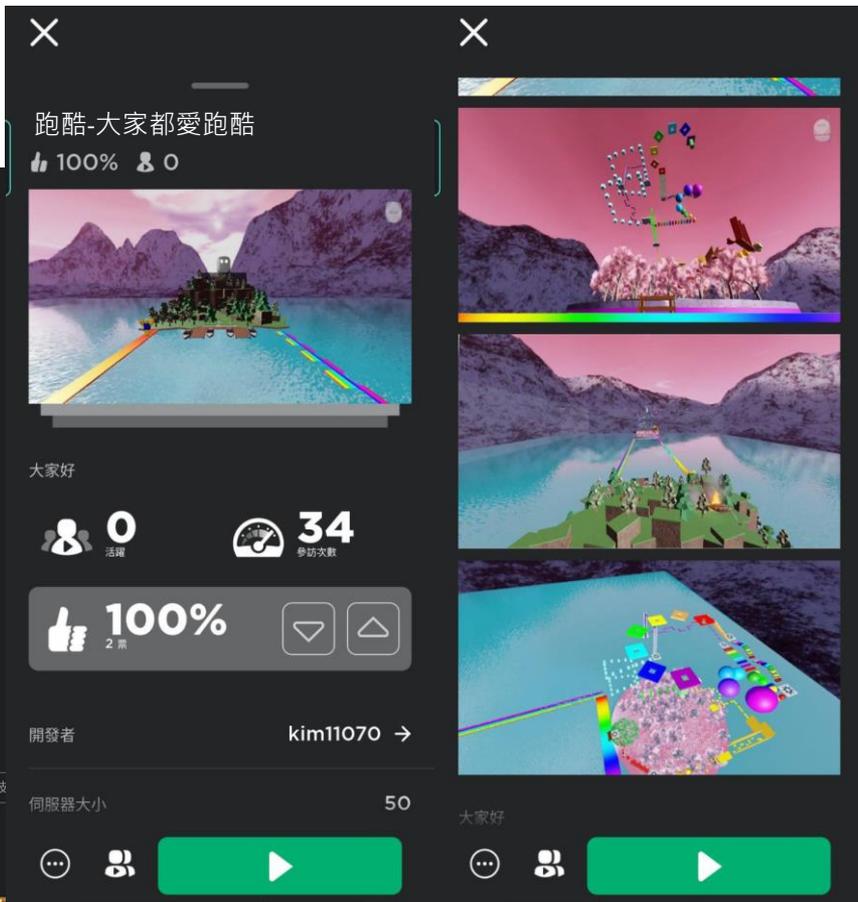
大家好

| 活躍 | 設為最愛人數 | 參訪次數 | 創作時間 | 更新時間 | 伺服器大小 |
|----|--------|------|-----------|-----------|-------|
| 0 | 2 | 225 | 2021/9/20 | 2022/4/28 | 50 |

所有



下載:去商店搜尋
roblox



成品展示

手機也可以遊玩



克服的困難

例如程式打完無法運作，要測試遊戲難度，花了很多時間去找那裡有錯，雖然和團員有溝通上的衝突，但還是透過良性溝通解決了問題。



心得

這次自主學習我受益頗多，讓我學會團結的重要性，有些事不是一個人能夠完成的，從開始想題目到設計地圖到測試到上台發表，過程中我們的解決事情的能力和技術有所進步，我覺得很開心能跟隊員一起完成這次的自主學習，雖然遇到很多困難，但我們大家一起努力終於完成了。



未來展望

因現有的技術和時間所以有些東西還沒做出來，之後我們想嘗試做出計時器和排名表，讓遊玩此遊戲的人能更感受到更多競爭的感覺



>>X1.5