

## CHAPTER 01

# 手機遊戲開發概論

HORAZON

手遊程式設計

# 課程目標

本學期目標：

製作一款完整的 **2D 橫向捲軸平台遊戲**。

我們將學習如何從零開始，直到發布到手機上。

# 核心策略

為了讓大家能順利完成作品，我們採用以下策略：

## 1. 先設計，後程式

- 前期不寫程式，使用現成工具。
- 讓大家先體驗「做遊戲」的樂趣。

## 2. 循序漸進

- 從擺放積木開始，到調整參數。
- 最後才學習如何寫程式控制它們。

# 手機平台介紹：IOS

Apple (蘋果) 的作業系統。

- **封閉系統**：只能裝而在 App Store 下載的軟體。
- **硬體統一**：螢幕尺寸與規格固定，好測試。
- **上架嚴格**：每年需繳交約 \$3000 台幣年費，且審核嚴謹。

# 手機平台介紹：ANDROID

Google 主導的開放系統。

- **開放系統**：各家手機廠牌 (Samsung, Pixel, 小米...) 都在用。
- **硬體破碎**：螢幕尺寸千奇百怪，測試較麻煩。
- **上架容易**：只需繳一次性費用 (\$25 美金)，審核較寬鬆。

# 軟體商店比較：APP STORE

屬於 Apple 的封閉花園，強調品質與安全。

## 1. 審核嚴謹

- 人工審核為主，重視隱私權與 App 完成度。
- 被退件 (Reject) 是家常便飯，需要來回修改。

## 2. 高含金量

- iOS 用戶平均消費力較高，是遊戲營收的主力。

## 3. 年費制

- 每年需繳交 **99 美金** (約 \$3,100 台幣)。

# 軟體商店比較：GOOGLE PLAY

屬於 Google 的開放市場，強調多元與快速。

## 1. AI 審核輔助

- 審核速度較快，但近年來對政策規範 (如兒童隱私) 變嚴格。

## 2. 海量用戶

- Android 設備市佔率全球第一，適合衝高下載量。

## 3. 一次性費用

- 只需繳一次 **25 美金** (約 \$800 台幣)，帳號終身有效。

# APP 開發方式：原生開發 (NATIVE)

使用該平台原本的語言來寫。

- iOS：使用 Swift 或 Objective-C。
- Android：使用 Kotlin 或 Java。
- **優點**：效能最好，功能支援最完整。
- **缺點**：寫一次只能在一個地方跑，要學兩套語言。



# APP 開發方式：跨平台 (CROSS-PLATFORM)

使用第三方工具，寫一次程式碼，轉換成雙平台版本。

- **工具**：Unity, Flutter, React Native。
- **優點**：開發快速，維護容易。
- **Unity 優勢**：專門為遊戲設計，圖形與物理效能極強。

# 為什麼需要「遊戲引擎」？

如果沒有引擎，要讓一張圖片在螢幕上動起來，你需要：

1. 寫程式跟顯示卡溝通 (OpenGL/DirectX)。
2. 自己算物理公式 (重力加速度  $F=ma$ )。
3. 自己處理音效解碼。

**這太難了！**

# 遊戲引擎的功能

引擎幫我們做好了底層的工作：

- **渲染 (Rendering)**：幫你畫圖。
- **物理 (Physics)**：幫你算碰撞和重力。
- **音效 (Audio)**：幫你播音樂。
- **輸入 (Input)**：幫你偵測觸控或按鍵。

我們只要專注在**遊戲好不好玩**。

# 為什麼選擇 UNITY？

1. **市場主流**：超過 50% 的手機遊戲是用 Unity 做的。
2. **資源豐富**：Asset Store 有買不完的素材跟外掛。
3. **直觀好用**：所見即所得，像排積木一樣。
4. **免費使用**：對學生與小團隊完全免費。

# 環境安裝：UNITY HUB

我們不直接安裝 Unity，而是先裝 Unity Hub。

它是一個管理中心，負責：

- 管理你的帳號 (Unity ID)。
- 管理你的專案 (Projects)。
- **安裝不同版本的 Unity Editor。**

# 環境安裝：UNITY EDITOR

在 Hub 裡面，我們要安裝 Editor (編輯器)。

- LTS (Long Term Support)：長期支援版，最穩定，不容易有 Bug。
- **建議版本**：Unity 2022.3 LTS。
- 安裝時記得勾選 **Android Build Support** (為了之後輸出手機版)。

# 介面簡介：PROJECT (專案)

這裡是你的**素材倉庫**。

- 所有圖片、音樂、腳本都在這。
- 如果這裡刪除了，檔案就真的不見了。
- 建議依照類型建立資料夾分類。

# 介面簡介：HIERARCHY (階層)

這裡是**目前場景的清單**。

- 列出現在關卡裡有哪些東西。
- 如果是父子關係 (Parent-Child)，會像資料夾一樣收合。
- 雙擊這裡的物件，鏡頭會自動飛過去看它。



# 介面簡介：SCENE (場景)

這裡是編輯畫面。

- 你是上帝，可以自由移動視角。
- 選取物件、移動、旋轉、縮放。
- 這裡看到的畫面不等於玩家看到的 (會有輔助線)。

# 介面簡介：INSPECTOR (屬性)

這裡是**詳細設定**。

- 當你選取一個物件，這裡會顯示它的所有資訊。
- Transform：位置、旋轉、大小。
- Components：掛在它身上的功能 (如圖片顯示器、物理元件)。

# 介面簡介：GAME (遊戲)

這裡是**玩家視角**。

- 顯示攝影機 (Camera) 拍到的畫面。
- 按下 Play 按鈕後，主要在這裡試玩。
- 可以在這裡切換不同的手機解析度預覽。

# 總結

我們建立了正確的觀念：

1. 選擇 **Unity** 做為開發工具，因為它強大且跨平台。
2. 了解 **iOS** 與 **Android** 的基本差異。
3. 認識了 **Unity 五大介面** (Project, Hierarchy, Scene, Inspector, Game)。

下一章開始，我們將實際動手建立第一個場景！