

專案的目標

在當今全球化的世界中，雙語教育變得越來越重要。學習英文拼字是英語學習的基石之一，它甚至還可以提升他們的閱讀、書寫和溝通能力。然而，傳統的教學方法可能缺乏趣味性，使學習過程變得沉悶乏味。因此，我們想要開發一個具有目標性且有趣的英文單字拼字遊戲，旨在提供學習者一個互動的學習環境。

我們的專案以SDGs為參考，旨在促進可持續發展。通過提供一個有趣且具有功能性的拼字遊戲，我們致力於支持教育目標(目標4)和全球合作(目標17)。這款遊戲提供了一種互動且多元化的學習方式，旨在提升學生的英文拼字能力。透過遊戲中的反饋和挑戰，我們鼓勵學生在自主學習的環境中不斷進步。同時，我們將拓展這個專案的應用範圍，以配合不同的教學進度和需求，以實現全球教育的目標，促進更廣泛的學習機會和知識傳遞。我們的目標是讓使用者在體驗的過程中同時享受遊戲樂趣和強化他們的拼字能力。透過這種「邊玩邊學」的學習方式，學習者可以在遊戲的互動過程中增強記憶力、注意力和語言技能，同時培養他們的自信心，使他們更能自如地應對英語學習的挑戰。

在遊戲中，使用者將面臨一連串的單字重組拼寫考驗，並在回合結束時獲得相關反饋，包括正向鼓勵訊息和錯誤檢討資訊。我們期望透過這些互動式的遊戲元素提高使用者的參與度，激發他們學習英文拼字的興趣，同時加強學習效果。此外，這個專案還具有廣泛的應用範圍。例如，我們可以擴充遊戲中的單字量，以滿足不同使用者的需求，並配合教學進度調整單字的範圍。這種使用彈性使其可以應用於教學實踐中，更貼合不同使用者的需求。

競品分析

我們以目前市場上知名的拼字遊戲Scrabble和Wordle進行分析，繪製成以下表格：

	Scrabble	Wordle
遊戲性	強(自由度高且有計分對戰)	適中(簡單的1A2B玩法)
規則複雜度	高(遊戲機制和板塊複雜)	低(玩法清楚簡單)
運氣成分	高(隨機分配字母牌)	高(字彙詞庫範圍大)
獲勝/通關難度	高(詞彙量需求大)	低(可用猜題技巧破解)
學習功能 (以一般玩家而言)	弱(單回合內字彙量太多、為了得分會有許多冷門生僻字出現)	高(一天學習一個單字，且遊玩過程中反覆拼寫同一個單字，會加深記憶)

總結來說，Scrabble是一個遊戲性高但複雜、上手難度高且不易用於「邊玩邊學」的拼字遊戲，而Wordle是個玩法簡單但學習功能較強的例子，但兩者皆有字庫單字量大不易控制難易度的缺點。因此我們的目標設立為「在有一定遊戲性的基礎上，降低規則複雜度和運氣成分，並將通關難度設立在適中的範圍。」例如，我們利用字串打散重組的方式來創造遊戲性，無額外添加複雜的規定以降低上手難度、提供錯題回顧功能讓使用者能夠透過練習來逐漸降低通關難度等。最重要的是，我們想要透過可以自訂的字庫來控制字彙範圍以滿足不同使用者的需求，避免字彙冷僻造成難度過高或過低的問題。

實作方式與流程

我們一開始使用Pygame來製作拼字遊戲並確認基礎架構，後來改為使用Flask製作成網頁版、使用Bootstrap排版，並使用自用電腦Linux的Ubuntu作業系統將遊戲架設至網路上。因此接下來將分為四大主軸來說明製作流程：

1. Pygame

我們使用 ChatGPT 來製作了Pygame的基礎雛形(註1)，然而由於指令下達得不夠明確，ChatGPT給出的程式碼並無法順利執行，出現了無法順利換題的Bug，因此依然得人工修正其錯誤，而後續要增加功能時也必須一步一步向ChatGPT下達指令。雖然它可以給出修改程式碼的建議，但卻不一定能順利執行或滿足我們的需求，因此我們後續是透過它所提供的程式碼作為參考，額外再查詢書籍或網路上其他人的做法，以此逐漸修正程式碼的Bug。我們完成主要功能的程式碼後，便發現Pygame所做出來的遊戲有一些缺點。例如要運行就必須額外下載套件(課堂上同學可能無法馬上體驗)，且我們對於它的工作原理不熟悉、上手需要一定的時間。因此我們後續決定以此為基礎，重新製作成網頁版遊戲，方便大家遊玩使用。不過我們在轉向使用Flask製作網頁後，依然保留Pygame的branch，後續也陸續完善其字典和出題功能。

2. Flask

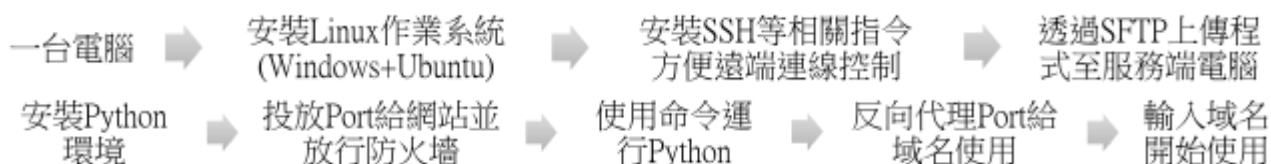
我們使用ChatGPT將原有的Pygame程式碼轉成 Flask(註2)，但由於無法直接轉換兩者，因此僅生成了基本架構跟目錄結構，其餘功能皆得額外添加。後續我們建立了各個頁面各自的功能，包含各個頁面接收POST後的行為、隨機出題、回答正確和錯誤的判定等。並且在確定功能都可正常執行後開始進行優化，例如字典模組化、通關條件設置、單回合內已用過的字詞儲存、條列誤答紀錄功能等，讓遊戲整體更完善，其學習功能也更加強化。

3. Bootstrap

Bootstrap框架有著方便排版的便利性，且其內建多種顏色樣式可以直接套用，因此為較方便的選擇。但由於這次我們對網頁有些較多的客制化的要求，加上網頁排版較為簡潔，因此在顏色上面有稍微使用自定義的顏色，而非使用Bootstrap所提供的內建顏色，且排版方面使用了我們自己設定的佈局。由於Bootstrap有靈活度很高、提供多種的佈局及樣式可選的特性，在我們更深入學習之後，相信可以設計出更美觀且較為複雜的頁面。例如我們這次的設計即選擇使用極簡風格，讓頁面變得清爽簡潔，同時也讓玩家一目瞭然遊戲的規則，以達成友善互動的目標。

4. 網路伺服器搭建

基礎思路是以自用電腦作為伺服器，在申請公網IP的情況下保證外部網絡能夠連接，其餘為環境搭建的工作以及使用一些Linux常用的指令去設定好程式的運作。



使用的套件

我們在運行的程式本體使用了pygame、flask、os、random、Bootstarp等現成的套件，以及我們製作的自訂套件setmode.config，但使用者若直接以網頁版進行遊戲則不需要使用任何套件，大大減輕了此遊戲的操作難度。

遇到的挑戰及解決方式

在撰寫運程式時，我們遇到最大的問題是ChatGPT所提供的Flask基礎架構有些部分有誤，造成後續程式執行時在切換頁面時會產生Bad Request的錯誤。一開始ChatGPT提供的編碼為return redirect('/game')，然而由於我們的程式中存在多個路由函式，而ChatGPT提供的程式碼會直接寫入固定的URL，造成應用程序的路由設置不夠靈活且容易出錯。因此後續我自行上網爬文找解決方法，並且發現使用return redirect(url_for('game'))也能夠達成表單數據的提交和重定向的功能，且使用url_for可以更靈活地根據函式名稱生成正確的URL，無論修改了路由設置中的URL或路由函式的名稱都不需要手動修改重定向的URL。這樣做更具彈性和可維護性，並且能夠適應應用程序結構的變化，也因此解決了被導向到無效URL的問題。

而在網頁運作上遇到問題則是無法順利在html檔案中嵌入css檔及js檔，目前暫時先將css及js內容匯入到html檔案中以保證這些檔案的正常運作，然而在繳交的檔案中則是有獨立分開的，以方便後續能找到問題並解決。

未來發展願景

我們的專案中，目前仍有一些方面可以進行改進，以提供更豐富且有價值的學習體驗。例如可以完善Pygame版本的遊戲，製作成APP功能、增添多樣化的遊戲模式和挑戰，例如時間限制或多人對戰等，讓遊戲性提升以吸引用戶持續使用。甚至可以透過新增附屬的功能來增加其便利性，例如提供一個專屬的教師管理界面，讓教師能夠監控學生的學習進度、設定任務和測驗，並給予個別化的指導和評估、加入學習追蹤系統，讓使用者能夠記錄自己的學習進度、成績趨勢。根據每個學習者的需要和能力，提供個別化的學習建議和挑戰，以促進學習者的成長和進步、增加社群系統，讓使用者能夠與其他學習者分享學習心得、討論單字拼寫的疑問等。目前的專案僅為具有基礎功能與架構的產出，然而後續依然有許多可以改進之處，讓專案變得更加完善。

而我們的專案也具有許多未來應用的潛力。例如搭配教科書或課程使用，以增強學生的英文拼字能力。教師可以將遊戲作為輔助教材，讓學生在課堂外進行互動學習，並將遊戲中學到的單字與教科書中的單字相互對應，也可以將遊戲的得分列入評量方式之一，加強學習效果。此外，我們的專案也可以成為線上英語課程或語言學習平台的一部分。透過整合遊戲元素，學習者可以透過互動的方式進行單字拼字練習，同時受益於自主學習和即時反饋的優勢。這樣的應用形式可以提升學習者的參與度和動機，並加速他們的英語拼字能力的發展。最後，我們的專案也可以持續擴展範圍，參考Scrabble支援不同語言的拼字功能，將其應用於其他語言的拼字學習，例如西班牙文、法文或德文等。這將使我們的專案更具多樣性和全球化的應用價值，為更多的學習者提供多元的學習機會。

備註(給ChatGPT的指令)

註1: 可以幫我做一個python拼字遊戲嗎？我想要用pygame呈現。每次玩會有3回合，畫面上隨機出現一個英文單字的字母(排列順序打散)，使用者要輸入正確的單字並按下enter鍵確認，若正確則得分加一，3回合結束後要顯示分數並出現再玩一次或結束的選項。

註2: 我想要把他放到網頁上，使用Flask，並且重新設計畫面配置，請問我要如何做？