開發工具與環境（技術說明＆引用API來源）

Github專案連結：<https://github.com/lawrence0510/Grduation_Repository>

1. 遊戲引擎

我們使用 Godot Engine 作為遊戲開發引擎。Godot 是一個功能強大的開源遊戲引擎，支援 2D 和 3D 遊戲的跨平台開發。

* Godot Engine 版本：v3.5.3.stable.official [6c814135b]

1. API 開發

我們的 API 是使用 Python 開發，結合 Flask 框架與相關工具進行構建與管理，Swagger 用於生成 API 文件，方便後續測試與擴展。

* Python 版本：3.12.3
* Flask：一個輕量級的 Python Web 框架，簡單靈活，適合快速構建網路應用。
* Flask-RESTx：基於 Flask 的擴展，提供了一套完整的工具來開發 RESTful API，並自動生成文件。
* Swagger：用於 API 文件自動生成與測試。

API Table（看最底下）

1. 人工智慧使用

我們使用OpenAI API以及Midjourney這兩個外部資源來產生問題、作為使用者追問之回答、以及產生主視覺、背景和敵人之圖片。

* OpenAI API：GPT-3.5-turbo-0125，此模型回答迅速、價格低廉，在我們的調校和prompt的設定下能順利發揮我們預期所需要的效果。
* Midjourney：

1. 資料庫

我們使用 MySQL 作為資料庫管理系統，並使用 phpMyAdmin 作為資料庫管理工具和操作介面，方便進行數據操作和管理。

* phpMyAdmin：5.1.1deb5ubuntu1。
* MySQL 版本：8.0.39-0ubuntu0.22.04.1

1. 伺服器環境

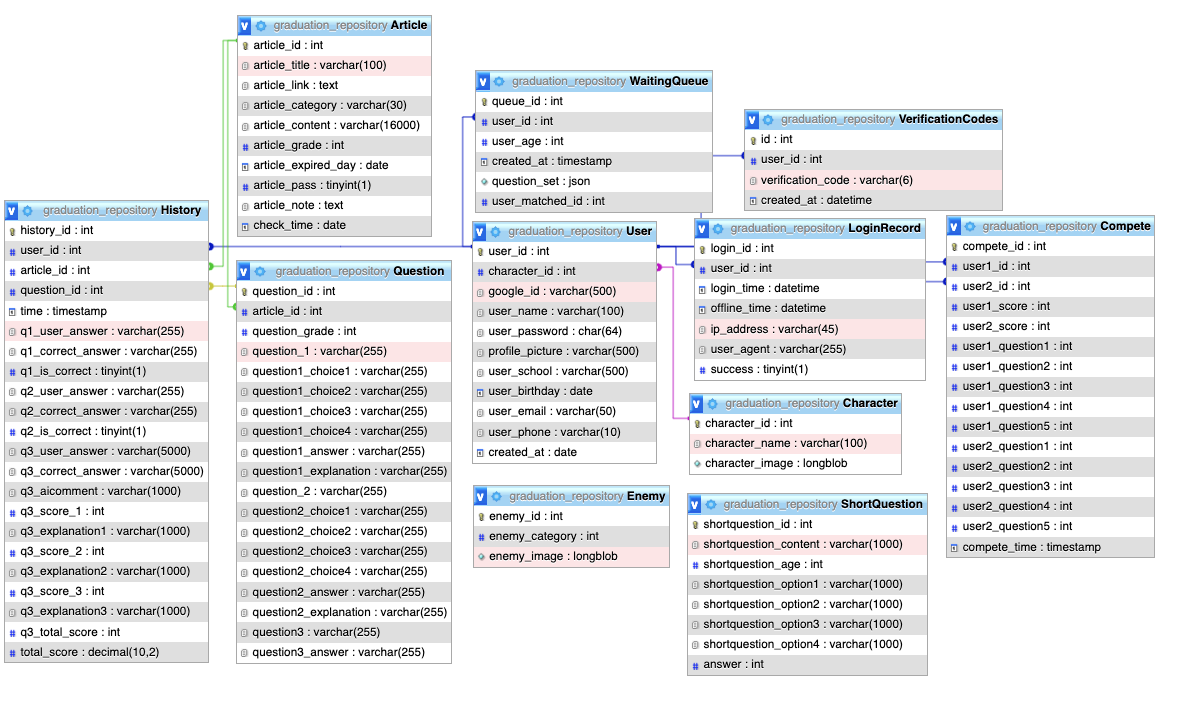
我們的伺服器運行在 Linux Ubuntu 環境中，使用 Nginx 來處理反向代理，並結合 Apache 來支援 PHP 應用，確保伺服器的高效能與安全性。

* 作業系統：Linux Ubuntu 22.04.4 LTS (GNU/Linux 6.8.0-40-generic x86\_64)
* 伺服器 IP：140.119.19.XXX（學校提供的 IP）
* Nginx：反向代理伺服器，用於將用戶請求分發至後端服務。
* PHP 版本：8.1.2-1ubuntu2.18

API Table

| **Endpoint** | **HTTP Method** | **Description** |
| --- | --- | --- |
| /User/normal\_register | POST | 註冊新用戶 |
| /User/image\_register | POST | 加入角色ID |
| /User/normal\_login | POST | 用戶登入 |
| /User/update\_offline | POST | 更新下線時間 |
| /User/get\_all\_history\_from\_user\_id | GET | 抓取所有上下線歷史、上線時長、作答紀錄 |
| /User/google\_login | GET | 用戶Google登入 |
| /User/authorize | GET | Google登入授權 |
| /User/send\_verification\_code | POST | 寄送驗證碼到使用者的 email |
| /User/check\_verification\_code | GET | 檢查使用者的驗證碼是否有效 |
| /User/logout | POST | 用戶登出 |
| /User/get\_user\_from\_id | GET | 根據 user\_id 獲取用戶資料 |
| /User/reset\_password | POST | 重設密碼 |
| /User/reset\_phone | POST | 重設手機號碼 |
| /User/reset\_school | POST | 重設學校資料 |
| /User/reset\_birthday | POST | 重設生日 |
| /User/get\_login\_record | GET | 獲取登入紀錄 |
| /Article/get\_random\_article | GET | 隨機獲取文章 |
| /Article/get\_random\_unseen\_article | GET | 根據使用者id找出此使用者還沒看過且年齡適合的文章 |
| /Article/get\_random\_uncheck\_article | GET | 隨機選取文章進行檢查 |
| /Article/submit\_article\_fixed | POST | 提交文章檢查結果 |
| /Article/upload\_articles | POST | 上傳文章資料 (Excel檔案) |
| /Article/upload\_pansci | POST | 從泛科學爬取文章資料並插入到數據庫 |
| /Article/upload\_reader\_digest | POST | 從讀者文摘爬取文章資料並插入到數據庫 |
| /OpenAI/get\_questions\_from\_article | POST | 傳送多個文章內容至OpenAI API，獲得三個問題及標準答案 |
| /OpenAI/get\_rate\_from\_answers | POST | 傳送回答獲得評分與評語 |
| /OpenAI/follow\_up\_question | POST | 追問問題並解答，根據最近歷史紀錄找出文章和問題 |
| /History/get\_all\_history | GET | 取得所有History資料 |
| /History/record\_new\_history | POST | 記錄新的歷史答題數據 |
| /History/get\_all\_data\_with\_history\_id | GET | 根據 history\_id 獲取歷史資料並包括對應的文章資料 |
| /History/get\_history\_from\_user | GET | 根據 user\_id 獲取歷史資料並篩選 article\_category |
| /Character/get\_character\_from\_id | GET | 根據 character\_id 獲取角色資料 |
| /Character/upload\_character\_image | POST | 上傳角色圖片並儲存到資料庫 |
| /Enemy/get\_enemy\_from\_id | GET | 根據 enemy\_category 獲取敵人資料 |
| /Enemy/upload\_enemy\_image | POST | 上傳敵人圖片並儲存到資料庫 |
| /Compete/match\_user | POST | 配對用戶，匹配對戰 |
| /Compete/cancel\_queue | DELETE | 取消匹配隊列中的用戶 |
| /Compete/get\_compete\_from\_id | GET | 根據 compete\_id 獲取比賽資料 |
| /Compete/update\_answer | POST | 根據 user\_id 和 compete\_id 更新對應的問題答案與分數 |

ERD



系統架構圖