國立臺北商業大學

資 訊 管 理 系

113’資訊系統專案設計

**系統手冊**

一張含有 美工圖案, 圓形, 圖形, 卡通 的圖片

自動產生的描述

**組 別：第113201組**

**題 目：補習班學生資訊系統**

**指導老師：林俊杰老師**

**組 長：11236001 林奕辰**

**中華民國114年12月04日**

# 目錄

[第一章 前言 6](#_Toc180024314)

[1-1 背景介紹 6](#_Toc180024315)

[1-2 動機 6](#_Toc180024316)

[1-3 系統目的與目標 6](#_Toc180024317)

[1-4 預期成果 7](#_Toc180024318)

[第二章 營運計畫 8](#_Toc180024319)

[2-1 可行性分析 8](#_Toc180024320)

[2-2 商業模式－Business model 8](#_Toc180024321)

[2-3 市場分析－STP 9](#_Toc180024322)

[2-4 競爭力分析SWOT-TOWS 10](#_Toc180024323)

[第三章 系統規格 11](#_Toc180024324)

[3-1 系統架構 11](#_Toc180024325)

[3-2 系統軟、硬體需求與技術平台 11](#_Toc180024326)

[3-3 使用標準與使用工具 12](#_Toc180024327)

[第四章 專案時程與組織分工 13](#_Toc180024328)

[4-1 專案時程 13](#_Toc180024329)

[4-2 專案組織與分工 13](#_Toc180024330)

[第五章 需求模型 16](#_Toc180024331)

[5-1 使用者需求 16](#_Toc180024332)

[5-2 使用個案圖 17](#_Toc180024333)

[5-3 使用個案描述 17](#_Toc180024334)

[5-4 分析類別圖 17](#_Toc180024335)

[第六章 設計模型 18](#_Toc180024336)

[6-1 循序圖 18](#_Toc180024337)

[6-2 設計類別圖 18](#_Toc180024338)

[第七章 實作模型 19](#_Toc180024339)

[7-1 佈署圖 19](#_Toc180024340)

[7-2 套件圖 19](#_Toc180024341)

[7-3 元件圖 19](#_Toc180024342)

[7-4 狀態機 19](#_Toc180024343)

[第八章 資料庫設計 20](#_Toc180024344)

[8-1 資料庫關聯表 20](#_Toc180024345)

[8-2 表格及其Meta data 20](#_Toc180024346)

[第九章 程式 21](#_Toc180024347)

[9-1 元件清單及其規格描述 21](#_Toc180024348)

[9-2 其他附屬之各種元件 21](#_Toc180024349)

[第十章 測試模型 22](#_Toc180024350)

[10-1 測試計畫 22](#_Toc180024351)

[10-2 測試個案與測試結果資料 22](#_Toc180024352)

[第十一章 操作手冊 23](#_Toc180024353)

[系統之元件及其安裝及系統管理 23](#_Toc180024354)

[第十二章 使用手冊 24](#_Toc180024355)

[各畫面、操作之移轉 24](#_Toc180024356)

[第十三章 感想 25](#_Toc180024357)

[感想與建議 25](#_Toc180024358)

[第十四章 參考資料 26](#_Toc180024359)

[附錄 28](#_Toc180024360)

[審查評審意見之修正情形 28](#_Toc180024361)

**圖目錄**

[圖 2-2‑1 8](#_Toc180620860)

[圖 3-1‑1 11](#_Toc180620861)

[圖 6-1‑1 18](#_Toc180620862)

**表目錄**

[表 2-3‑1 9](#_Toc180620875)

[表 2-4‑1 10](#_Toc180620876)

[表 3-3‑1 12](#_Toc180620877)

[表 4-2‑1 專案組織與分工 13](#_Toc180620878)

[表 4-2‑2 專題成果工作內容與貢獻度表 15](#_Toc180620879)

# 前言

## 背景介紹

隨著教育需求的增加，補習班市場的競爭越發激烈，傳統的小班制補習班將面臨著運營管理與教學品質的挑戰，尤其是當管理模式過於依賴人工時，容易出現資源分配不當、學生進度無法及時追蹤等問題，為了解決這些挑戰，亟需一個資訊管理系統來提升運作效率，並能即時追蹤學生進度，從而實現適時的調整與改進。

## 動機

繁瑣的行政流程需要大量人工處理，導致效率低下與失誤率上升，且管理者與教師難以及時掌握學生的學習進度與表現，影響後續學生的教學規劃與課程設計。

為了解決這些問題，使用資訊管理系統成為必要，能夠顯著提升行政流程的處理速度，降低人工操作帶來的錯誤風險，系統還能即時記錄學生的學習進度與狀況，方便教師與管理員為後續課程進行規畫與調整。此外整合LINE Bot功能，進行通知提醒與訊息回覆，不僅提升溝通效率，還能提升小班制補習班的服務與教學品質。

## 系統目的與目標

本系統旨在透過資訊系統管理，解決傳統小班制補習班在運營與管理上的困難，具體目標如下：

1. 改善行政流程效率：透過系統處理日常行政工作，例如學生出缺勤紀錄、課程安排與管理，提高行政處理速度，減少作業時間與人工操作錯誤的機會。
2. 學生學習進步與狀況追蹤：系統記錄學生的學習進度與表現，協助管理者和教師掌握學生狀況，便於後續教學規劃與調整，設計出符合個人需求的教學方案，提升教學品質與學生學習成效。
3. 整合LINE Bot提升服務品質：利用LINE Bot進行自動化課程相關通知與基本資料回覆，如上課通知與繳費通知提醒、課程學期期間查詢、學生出席狀況查詢、學期進度查詢，減少等待訊息回覆的時間，改善溝通效率，進一步提升補習班的服務品質。

## 預期成果

1. 提升管理效率：透過系統處理行政作業，減少人工操作的時間與錯誤。

2. 提升學生學習效果：系統能記錄學生的學習進度與表現，協助教師制定個人化的

教學計畫。

3. 提升服務品質與溝通效率：整合LINE Bot功能，提供重要通知提供與資料查詢，

使學生能更快速接收資訊。

# 營運計畫

## 可行性分析

1. 技術可行性：系統將基於Python Flask框架進行開發，結合MySQL資料庫進行資料存取，並使用Chart.js、FullCalendar等前端工具實現視覺化效果，團隊成員具備相關的開發經驗，能夠順利推進系統的實施與維護。
2. 操作可行性：具備簡潔直觀的介面，讓不同角色的使用者（如學生、教師、管理者）均能輕鬆操作，降低使用障礙。
3. 經濟可行性：系統開發使用的技術工具多為開源或免費資源，這大幅降低了開發與後續維護成本，隨著系統使用，降低行政管理成本，提升學生學習品質、溝通效率，為補習班帶來經濟效益。

## 商業模式－Business model

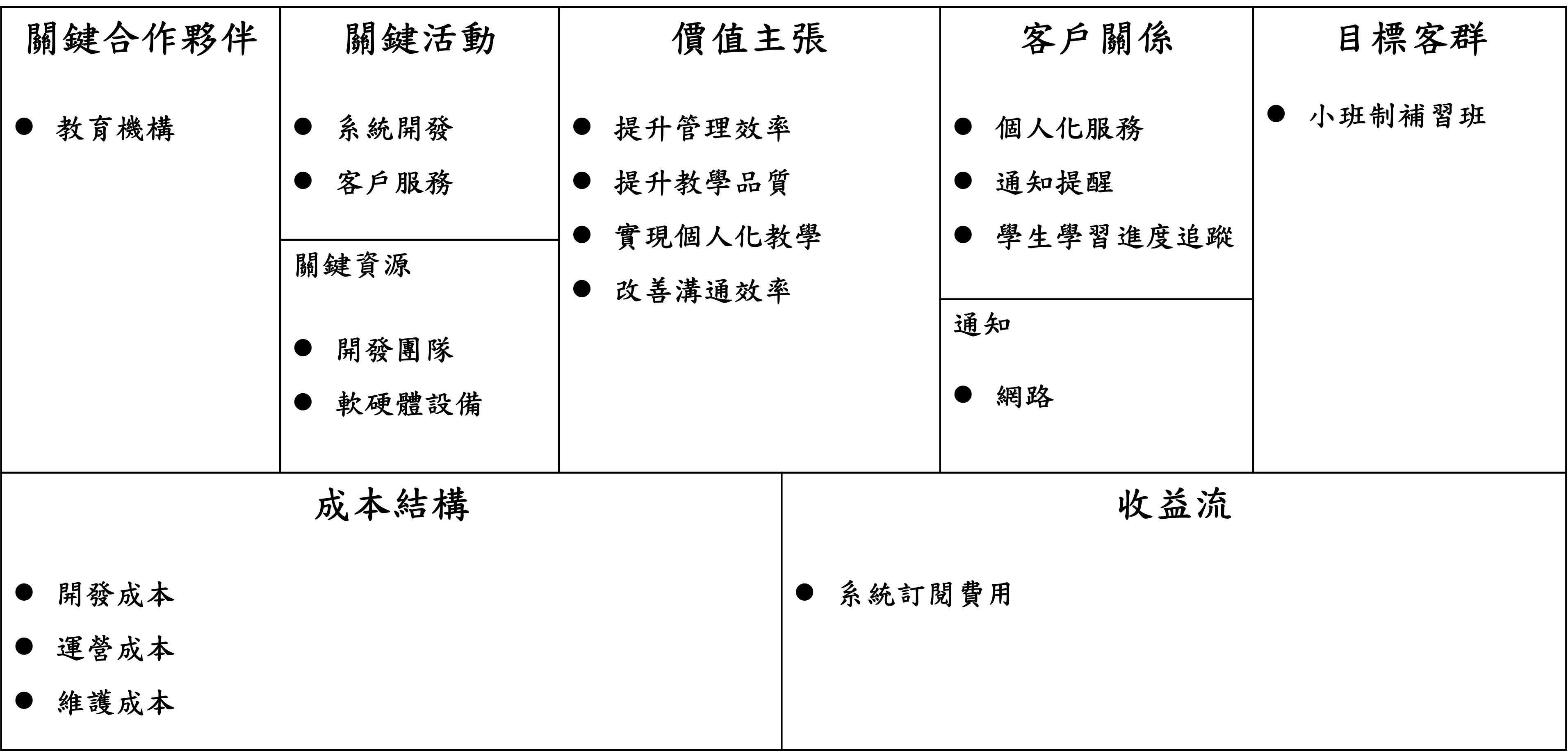


圖 2-2‑1

## 市場分析－STP

表 2-3‑1

|  |  |
| --- | --- |
| 市場區隔（Segmentation） | * 教學模式：   個人化教學模式針對每位學生的特定需求，提供量身訂做的學習計劃和內容。   * 教學規模：   小班制教學每班人數少，教師能進行個別指導增進教學互動。 |
| 市場目標（Targeting） | 小班制補習班：提供個人化教學的補習班，需要完善的管理系統以提升教學品質、管理效率。 |
| 市場定位（Positioning） | * 個人化學生管理系統：   提供一個針對小班制補習班設計的學生資訊系統，特過學習進度追蹤，讓老師和管理者能更方便地掌握學習情況，以便及時調整教學內容。   * 提升管理效率、教學與服務品質：   透過系統管理，提升補習班管理效率與教學品質，學生也可以自行透過系統進行調、補課時段，LINE Bot上課、繳費通知提醒與學生基本資料查詢。 |

## 競爭力分析SWOT-TOWS

表 2-4‑1

|  |  |  |
| --- | --- | --- |
| **內部因素**  **外部因素** | **優勢 (Strengths)**   1. 個性化教學 2. 提升管理效率 3. 學生進度追蹤 | **劣勢 (Weaknesses)**   1. 市場規模小 |
| **機會 (Opportunities)**   1. 市場需求成長 2. 個人化學習需求上升 3. 擴展服務功能 | **SO**   1. 整合更多個人化功能應對個人化學習需求 2. 利用學生進度追蹤來滿足個人化學習需求 | **WO**   1. 強調個人化的服務來提升市場吸引力 2. 利用線上課程拓展市場規模 |
| **威脅 (Threats)**   1. 資料外洩 2. 維護與運行成本增加 | **ST**   1. 加強資安措施，避免資料外洩 2. 利用個人化教學自動推薦適合的課程資訊，提升營運效率、降低人力成本 | **WT**   1. 加強行銷，增加曝光率，從而帶來資金支持 |

# 系統規格

## 系統架構

一張含有 文字, 螢幕擷取畫面, 字型, 圓形 的圖片

自動產生的描述

圖 3-1‑1

## 系統軟、硬體需求與技術平台

1. 硬體需求：

* 伺服器（建議配置）：

CPU：四核處理器或以上

RAM：8GB或以上

儲存空間：SSD 256GB以上

網路連接：至少100Mbps

* 用戶端設備：可使用任何支援瀏覽器的設備

1. 軟體需求：

* 作業系統：Windows 10

## 使用標準與使用工具

表 3-3‑1

|  |  |
| --- | --- |
| 系統開發環境 | |
| 作業系統 | Windows 10 |
| 程式撰寫工具 | Visual Studio Code |
| 程式開發工具 | |
| 前端 | HTML、CSS、JS |
| 後端 | Python |
| 框架 | Flask |
| 資料庫 | MySQL |
| 開發套件 | Xampp |
| 文件 | |
| 紙本文件 | Microsoft Word |
| 圖表、UML | Draw.io |
| 簡報 | Canva |
| 專案管理及版本控制工具 | |
| 版本控制 | GitHub |

# 專案時程與組織分工

## 專案時程

## 專案組織與分工

表 4-2‑1 專案組織與分工

●主要負責人 〇次要負責人

| 項目/組員 | | 11236001  林奕辰 |
| --- | --- | --- |
| 後端開發 | 資料庫建置 | ● |
| 伺服器架設 | ● |
| 程式撰寫 | ● |
| 功能測試 | ● |
| 前端開發 | 功能構想 | ● |
| 程式撰寫 | ● |
| 功能測試 | ● |
| 美術設計 | UI/ UX | ● |
| Web/APP介面設計 | ● |
| 色彩設計 | ● |
| Logo設計 | ● |
| 素材設計 | ● |
| 文件撰寫 | 統整 | ● |
| 第1章 前言 | ● |
| 第2章 營運計畫 | ● |
| 第3章 系統規格 | ● |
| 第4章 專題時程與組織分工 | ● |
| 第5章 需求模型 | ● |
| 第6章 設計模型 | ● |
| 第7章 實作模型 | ● |
| 第8章 資料庫設計 | ● |
| 第9章 程式 | ● |
| 第10章 測試模型 | ● |
| 第11章 操作手冊 | ● |
| 第12章 使用手冊 | ● |
| 報告 | 簡報製作 | ● |

表 4-2‑2 專題成果工作內容與貢獻度表

|  |  |  |  |
| --- | --- | --- | --- |
| 序號 | 姓名 | 工作內容 | 貢獻度 |
| 1 | 組長  11236001  林奕辰 | 前後端程式撰寫與測試、資料庫建置、文件製作、美術設計 | 100% |
|  | | | 總計:100% |

# 需求模型

## 使用者需求

1. 學生：

* 查詢課程資訊：能夠方便地查看課程安排，包括上課時間、地點。
* 請假功能：學生可以進行單堂課程請假。
* 調課功能：學生選擇的上課時間、地點，系統會判斷是否有空位及合適教師，符合條件則調課成功，反則失敗。
* 個人檔案：學生能夠查看、修改自己個人檔案的資料。
* 出席紀錄：紀錄學生課程的出缺勤狀況。
* 上課筆記紀錄：學生上完課要填寫本次上課的進度內容。
* Line Bot上課通知提醒：學生加入Line Bot後，Line Bot會根據學生的上課時段前一天發出上課提醒通知，請學生記得上課，否則要請假。
* Line Bot繳費通知提醒：學生加入Line Bot後，Line Bot會根據學生本期剩餘堂數為5時，發出繳費提醒通知，請學生記得繳交下期學費。

1. 教師：

* 點名功能：老師進行課程點名，記錄該堂學生的出席狀況。
* 查看學生進度：老師能夠查詢負責學生的學習進度。

1. 管理者：

* 師生資訊管理：管理者能夠查看、編輯和管理所有學生和教師的基本信息。
* 教室管理：管理者查看、新增各教室的上課時段。
* 學生出席紀錄管理：管理者可以查看、修改學生的出缺勤狀況及當日的授課老師。
* 學生與授課教師安排：管理者可以新增或刪除指定學期學生上課時段、地點。
* 財務管理：管理者可以記錄學生的繳費狀況(繳費日期、繳費方式、繳費細項)。
* 教室出席學生人數查詢：管理者可以查看各教室時段之出席學生人數及授課老師。

## 使用個案圖

## 使用個案描述

## 分析類別圖

# 設計模型

## 循序圖

一張含有 美工圖案, 圓形, 圖形, 卡通 的圖片

自動產生的描述

圖 6-1‑1

## 設計類別圖

# 實作模型

## 佈署圖

## 套件圖

## 元件圖

## 狀態機

# 資料庫設計

## 資料庫關聯表

## 表格及其Meta data

# 程式

## 元件清單及其規格描述

## 其他附屬之各種元件

# 測試模型

## 測試計畫

## 測試個案與測試結果資料

# 操作手冊

系統之元件及其安裝及系統管理

# 使用手冊

各畫面、操作之移轉

# 感想

感想與建議

# 參考資料

**前端技術**

* **HTML 模板**

Themewagon. Free Bootstrap 4 HTML 5 Admin Dashboard Template Stisla.

<https://themewagon.com/themes/free-bootstrap-4-html-5-admin-dashboard-template-stisla/>

* **FullCalendar官網**

<https://fullcalendar.io/>

* **Chart.js Documentation**

<https://www.chartjs.org/docs/2.7.3/>

* **JavaScript**

W3Schools. JavaScript Tutorial.  
<https://www.w3schools.com/js/>

* **Font Awesome官網**

<https://fontawesome.com/>

* **Bootstrap 4 - Modal**

<https://getbootstrap.com/docs/4.6/components/modal/>

**後端技術**

* **Flask**

STEAM教育學習網 網頁服務與應用- Flask 函式庫

<https://steam.oxxostudio.tw/category/python/example/flask.html#a8>

* **Python-MySQL**

W3Schools. Python MySQL  
<https://www.w3schools.com/python/python_mysql_getstarted.asp>

* **Python-APScheduler**

HackMD**定時任務、自動化執行**  
<https://hackmd.io/@aaronlife/python-topic-scheduler>

* **Ngrok**

Ngrok官網  
<https://ngrok.com/>

STEAM教育學習網 網頁服務與應用-使用ngrok服務

<https://steam.oxxostudio.tw/category/python/example/ngrok.html>

* **Line Bot**

LINE Biz. LINE 官網  
<https://tw.linebiz.com/>

LINE Developers. LINE Bot 官網  
<https://developers.line.biz/en/>

STEAM教育學習網 網頁服務與應用-LINE BOT教學

<https://steam.oxxostudio.tw/category/python/example/line-bot.html>

* **DBDiagram官網**

<https://docs.dbdiagram.io/>

**附錄**

審查評審意見之修正情形

審查評審意見之修正情形

|  |  |
| --- | --- |
| 評審建議事項 | 修正情形及回覆 |
| 無 | 本專題為初評尚未有評審建議事項 |