Cloud Computing Final Report

MLB Discord Bot: Real-Time Sports Data & Fan Engagement

Group 18

111502516

113525016

113526008

113527010

**1. Motivation and Objectives**

**動機**

對於喜愛美國職棒大聯盟（MLB）的球迷來說，想要隨時隨地查詢比賽的即時資訊或歷史數據，其實不是件容易的事。目前的解決方案，例如 ESPN 或 Yahoo Sports，雖然提供了全面的數據，但缺少直接在社群平台（如 Discord）上互動的功能。

很多球迷需要來回切換不同的平台來找資料，既麻煩又浪費時間，還可能因手動整理數據出錯，影響體驗。因此，我們希望透過開發一款專為 MLB 球迷設計的 Discord 機器人，整合數據來源，提供更方便、快速、準確的查詢方式，同時增加粉絲之間的互動樂趣。

**目標**

1. 開發一款實用的 Discord 機器人  
   設計一個能即時查詢 MLB 比賽數據和球員統計的機器人，讓球迷只需簡單輸入指令，就能輕鬆獲取想要的資訊。
2. 實現即時互動功能  
   提供簡單明瞭的聊天指令，讓使用者可以直接在 Discord 上與機器人互動，快速獲得最新的比賽資訊。
3. 運用 AWS 雲端技術  
   利用 AWS 提供的服務來打造穩定又能應對高需求的系統，包括：

- EC2：託管機器人，負責處理用戶請求和數據查詢邏輯。

- Lambda：實現數據處理的自動化，按需運行更有效率。

- DynamoDB：儲存歷史數據和查詢記錄，確保查詢快速且穩定。

- API Gateway：負責 API 請求的管理和安全性，讓機器人與外部服務 高效溝通。

1. **Related Work/Market Survey**

**現有解決方案**

目前市面有一些與體育數據相關的解決方案，但每個方案都有不足之處：

1. **ESPN APIs**

* 提供全面的體育數據，涵蓋多種賽事與聯賽。
* **不足**：沒有與 Discord 平台整合，無法讓用戶直接互動獲取數據。

1. **Yahoo Sports**

* 提供即時比分和比賽更新，適合想快速掌握比賽進度的用戶。
* **不足**：缺乏自動化和互動功能，無法滿足用戶的深度需求。

1. **其他 Discord 機器人**

* 支援基本的運動比分查詢，但大多聚焦於多聯賽支援，功能較為一般。
* **不足**：缺少針對 MLB 的專屬功能和即時互動能力，無法提供深入的體驗。

**比較與特色**

我們的 MLB 專屬機器人針對這些不足，具備以下優勢：

1. **專屬 MLB 的功能**

* 提供精準的球員數據和比賽資訊查詢，滿足棒球迷的需求，讓用戶獲得專業且全面的 MLB 資訊。

1. **高度自動化**

* 整合 API 與網頁爬蟲技術，實現數據的自動更新與即時查詢，避免手動操作的誤差。

1. **與 Discord 深度整合**

* 用戶可以直接在 Discord 平台輸入指令，快速查詢比賽結果和球員數據，提升互動體驗。

**現有解決方案與我們的機器人比較**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **功能特點** | **ESPN APIs** | **Yahoo Sports** | **其他 Discord 機器人** | **我們的 MLB 機器人** |
| **數據覆蓋範圍** | 全面 | 全面 | 有限 | 全面且專注於 MLB |
| **Discord 整合** | 無 | 無 | 有 | 深度整合 |
| **自動化程度** | 部分支持 | 有限 | 有限 | 完全支持（API 整合 + 爬蟲） |
| **穩定性與可擴展性** | 不適用 | 不適用 | 有限 | 高 |

**3. Solution Overview**

我們的專案設計並實作了一個 MLB Discord Bot，主要功能是為美國職棒大聯盟（MLB）的球迷提供即時資訊查詢服務，並提升粉絲之間的互動體驗。這個機器人結合了 API 整合與爬蟲技術，能夠透過簡單的指令，讓使用者在 Discord 平台上快速查詢以下資訊：

(1)查詢投手資訊：輸入!pitcher可以顯示先發投手的資訊

(2)今日賽程：有時候我們會想知道今天有哪些比賽舉行，可以輸入!schedule查詢當天的比賽賽程，包括主客隊和比賽時間。

(3)球隊代號清單：由於其他查詢功能需要用到球隊代號，但不是每個使用者都能把三十隊的代號記的清清楚楚，因此設一個功能方便使用者查詢。這個指令會顯示 MLB 球隊的完整名單及其縮寫。

(4)特定日期的比賽紀錄：輸入!history LAD 2024-10-30就可以查詢指定球隊特定比賽的結果。

(5)近期比賽成績：提供指定球隊最近數場比賽的結果與詳細資訊。

(6)球員數據查詢：討論特定球員的數據在棒球迷間是非常普遍的現象，可以輸入!hstat Freddie Freeman來查詢Freddie Freeman的數據。數據經過挑選，只會顯示在現代棒球仍存在實際意義的資訊，舊有的打點、勝投等過時的數據都不會顯示。

(7)隨機棒球名言：透過 API Gateway 和 Lambda ，從DB中提供隨機的棒球名言，增添趣味性。

(8) 指令列表：輸入!help，機器人會顯示所有可用的指令，方便使用者快速了解機器人的功能

這些功能讓球迷能夠省去多平台查找的麻煩，提升了查詢效率和互動體驗。

一張含有 文字, 圖表, 行, 螢幕擷取畫面 的圖片

自動產生的描述

在這個專案中，我們充分利用了 AWS 的雲端技術來確保系統的穩定性和效能，主要使用了以下服務：

AWS EC2

應用方式：用於託管 Discord Bot 的主程序，負責接收用戶的指令並處理邏輯。

優勢：提供高可用性和彈性，可根據需求調整計算資源。

AWS Lambda

應用方式：實現隨機棒球名言的功能，負責按需執行並返回結果。

優勢：以事件為驅動，節省資源，適合短時間執行的小型任務。

AWS DynamoDB

應用方式：存儲用戶查詢記錄和行為數據，並作為機器人後端數據的主要存儲解決方案。

優勢：提供高效能的 NoSQL 資料庫，能處理大量即時查詢。

API Gateway

應用方式：作為 Discord Bot 與 Lambda 的中介層，負責管理 API 請求，確保數據交互安全與高效。

優勢：簡化了 API 請求的設置，同時提升安全性和性能。

這些雲端技術的整合不僅大幅提升了系統的穩定性和可擴展性，也讓我們能夠專注於核心功能的開發，避免陷入繁瑣的基礎設施管理。

**4. Project Outcomes**

1. 指令列表：

輸入!help，機器人會顯示所有可用的指令

一張含有 文字, 螢幕擷取畫面, 字型, 軟體 的圖片

自動產生的描述

1. 明日比賽先發投手查詢：

一張含有 文字, 字型, 螢幕擷取畫面, 圖形 的圖片

自動產生的描述

由於現在是休賽季沒有比賽，因此顯示以上資訊

1. 今日賽程查詢：

一張含有 文字, 螢幕擷取畫面, 字型 的圖片

自動產生的描述

若是使用當下時間則是：  
一張含有 文字, 螢幕擷取畫面, 字型 的圖片

自動產生的描述

1. 顯示球隊代號：

輸入!teams可以得到以下回答：

一張含有 文字, 螢幕擷取畫面, 設計 的圖片

自動產生的描述

可以看到完整的球隊清單，包含城市名稱以及縮寫。

1. 查詢特定日期的比賽紀錄

輸入!history LAD 2024-10-30就可以查詢指定球隊特定比賽的結果

一張含有 文字, 螢幕擷取畫面, 字型 的圖片

自動產生的描述

1. 查詢特定球隊近期比賽

一張含有 文字, 螢幕擷取畫面, 字型 的圖片

自動產生的描述

1. 查詢特定球員的統計資訊：

指令開頭的h代表hitter，若要查詢只需要換成p(pitcher)即可。

一張含有 文字, 螢幕擷取畫面, 設計 的圖片

自動產生的描述

1. 輸出隨機的棒球名言：

一張含有 文字, 螢幕擷取畫面, 字型, 標誌 的圖片

自動產生的描述

**5. Mind Map**

一張含有 文字, 圖表, 字型, 行 的圖片

自動產生的描述**6. The most time-consuming part of the development process**

1. 挑戰：

整合 API 並處理不同格式的數據

由於功能的複雜度不一，有些功能是使用簡單的爬蟲完成，有些則使用API的方式處理。而為了配合Discord的輸出function，有些抓下來的數據類型與樣式必須要調整過後才能適配且正常輸出。

1. Future work
   * 1. 嘗試獲得savant上面的數據，以圖表的方式檢視球員成績。
     2. 加入特定投打歷史對決成績功能
     3. 加入伺服器管理功能(如禁言、身分組分配等)，使機器人具備更全面的功能

**7. Team Interaction Photos**

**8. Reflections on Project Development**

* Each team member must write their own reflections and append them to the end of the report.

**9. Team Members and Task Allocation (Indicate Contribution Proportions)**

* Each team member should write their own task description, including the proportion of their contribution.

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Student ID | Job Description | Contribution Ratio |
|  | 100000 | Ex: Collect data and do the preprocessing…. | 30% |
|  | 100001 | Ex: Build the cloud network…. | 20% |