

# 我在班上開了一間合作社—以聊天機器人記帳

313-11 林孟寬



完整程式碼：[https://github.com/mklin95/cooperative\\_store/tree/main](https://github.com/mklin95/cooperative_store/tree/main)

## 摘要

由於我的消費習慣，從高二上學期開始衍伸出在班上轉賣零食、開合作社的生意，從一開始的現金交易，高一下學期改成在我的記事本記錄餘額。因為帳目不公開、資訊不足，高二暑假自學 Discord Python 寫了聊天機器人，改成我自己在 Discord 線上記帳，雖然整學期總共有 7297 元的消費額，但還是有許多不便之處。所以高三寒假再新增會員透過聊天機器人記錄消費的功能，並在高一下學期提升了 34.3% 的消費額。過程中除了在開發聊天機器人時磨練我的程式能力，第一次有使用者直接參與我的實作成品，也讓我需要練習從使用者的角度出發，不斷整合大家的意見並改善功能、提升 UX，並從中得到以下收穫：

- 善用程式解決問題的方便性
- 寫程式前事先釐清問題事半功倍、更清楚程式架構
- 從使用者角度設計產品的重要性，除了 UI，UX 也很重要。
- 累積經營客戶關係上的經驗
- 會員和我形成的團隊持續讓我的記帳機器人進步

### 1. 高二上 (2022.08)：小合作社生意的形成 (現金交易)

升上高二重新編班後，我的教室從二樓搬到了四樓，還離合作社變遠了。一開始每次肚子餓的時候我還是會乖乖走到一樓合作社買，但每次去合作社都要爬樓梯太麻煩了，所以我後來都一次買一個禮拜的份，大概 5、6 包零食放在抽屜，這樣就不用每天多爬一趟樓梯了。不過，同學們也懶得走去合作社，他們發現了我的習慣後，就老是跟我說：「你還有那麼多包零食，可以賣我一包嗎？」我也不好意思拒絕不賣，漸漸的，我常常一回教室東西就都被買光了，為了應付同學們的消費需求，我去合作社時只好越買越多，一次都要買個十幾包零食，買到後來合作社阿姨都認識我了，還會送我紙箱裝東西、關心我在班上的「生意」如何。就這樣過了幾個月，我在班上轉賣合作社零食、「小福利社」的生意也大致成形了。

### 2. 高一下學期 (2023.02)：現金交易 → 儲值 → 人工記帳

不過，隨著生意越來越大，我買賣時收到的銅板也越來越多，而且有時候還會錢不夠找不開，太麻煩了。所以我高一下學期改成記帳的方式，同學們預先儲值在我這裡，每次買東西就用扣錢的，而大家的餘額都記錄在我手機的記事本裡。當時我只有簡單記錄同學座號和相對應的餘額 (如圖 1)，每次交易只用前一次的餘額去做加減運算，都沒有消費記錄。當時全班有 7 個人在我這裡記帳，其他人還是現金交易。

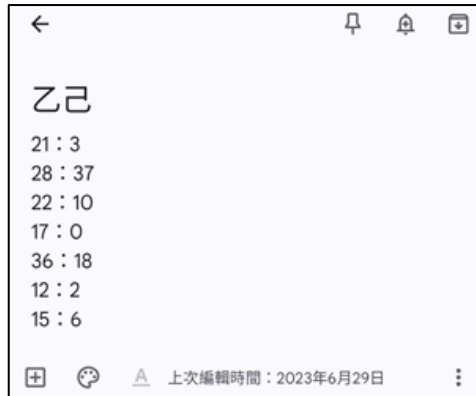


圖 1：高二下的記帳系統，只有在手機記事本簡單記錄同學的座號跟餘額

雖然解決了現金交易的問題，但新的問題也出現了：

- 因為我都是心算，餘額可能會算錯。
- 如果想知道餘額只能來問我，比較不方便。
- 沒有消費記錄難以對帳，同學總是臭我有沒有偷黑他們的錢。
- 學校合作社記錄每個人的消費額，每年年底依據營收狀況，回饋大概 5~10%的現金。但我只有記錄餘額，沒辦法得知消費額去分配回饋金給會員們。
- 由於只有我有大家的餘額，資訊不公開會使人懷疑、不安，這也是比較少人(7 人)在我這裡記帳的主因，一開始只有跟我最熟的幾個朋友願意相信我、預付現金給我。

### 3. 高二暑假 (2023.07)：人工記帳 → 開發 Discord 機器人來記帳

由於社群聊天軟體的普遍性及社群聊天機器人可提供的服務，我認為聊天機器人可以解決上述的問題。評估要開發哪個聊天軟體的機器人時，因為我對 Discord (以下簡稱 dc) 機器人有點概念，而且基本指令和 python 幾乎一樣，對我而言入手門檻最低，最後決定趁著暑假開發一個 dc 機器人來記帳。

#### 3.1. 自學創建 dc 機器人

由於我習慣自學，我決定先上網收集資料，學習如何創建 dc 機器人與特定的基本語法。我參考了這篇 [Discord.py 機器人從 0 到 1 超詳細教學](#)，完成前置作業並使機器人可以上線。至於機器人運作的想法是開 2 個 array 分別記錄餘額跟消費額，並根據我輸入的不同文字加減餘額跟消費額並輸出給同學看。

因為我一開始也是因為我的消費習慣，順便給同學方便和樂趣，所以我沒有打算用合作社回饋金賺錢，因此記錄大家的消費額是為了年底按照比例回饋同學。

### 3.2. 遇到問題：執行線上服務 → 選擇 Replit

不過，我程式寫下去才想起來 dc 機器人須要 web 線上運作，所以就尋找線上的免費 web 服務來跑程式，而不是原本在自己的電腦上運作。之前做自主學習遇到程式的問題我都會問 ChatGPT，且都能提供我實用的答案，我這次也我問了 ChatGPT 【Any recommend online web service to host a discord bot 24/7 ?】，得到建議 Replit、Glitch 跟 Heroku(圖 2)。

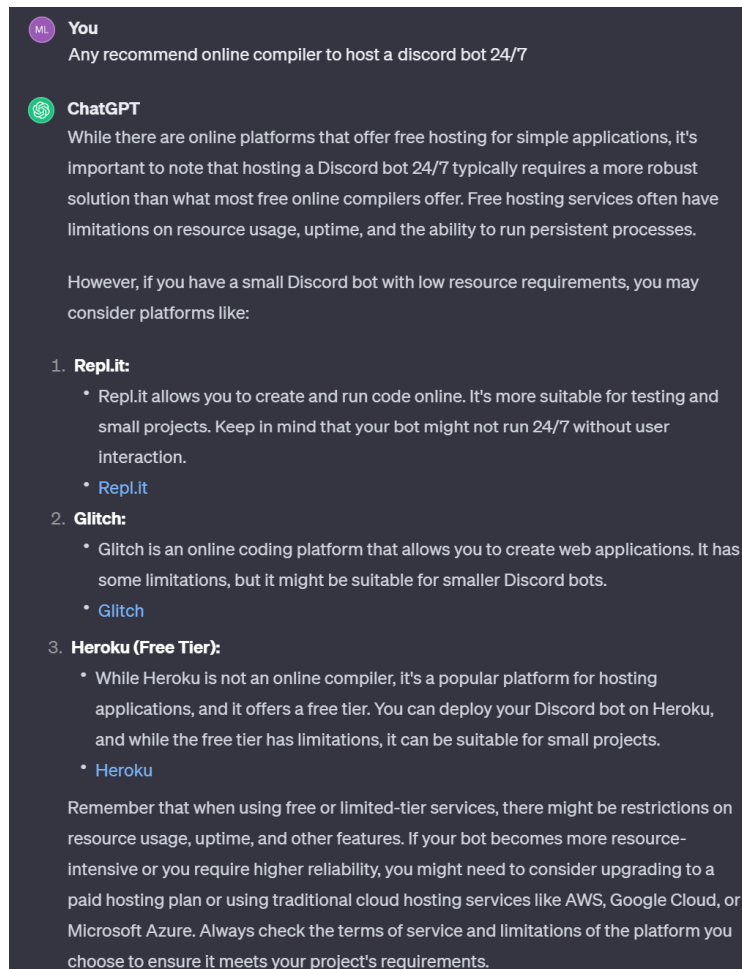


圖 2：尋問 ChatGPT 推薦線上服務平台

考量 dc 機器人的使用時機主要都在學校，有手機板比較方便，因為三個 compiler 中只有 Replit 有手機 app，最後決定改用 Replit 編譯 (圖 3)。

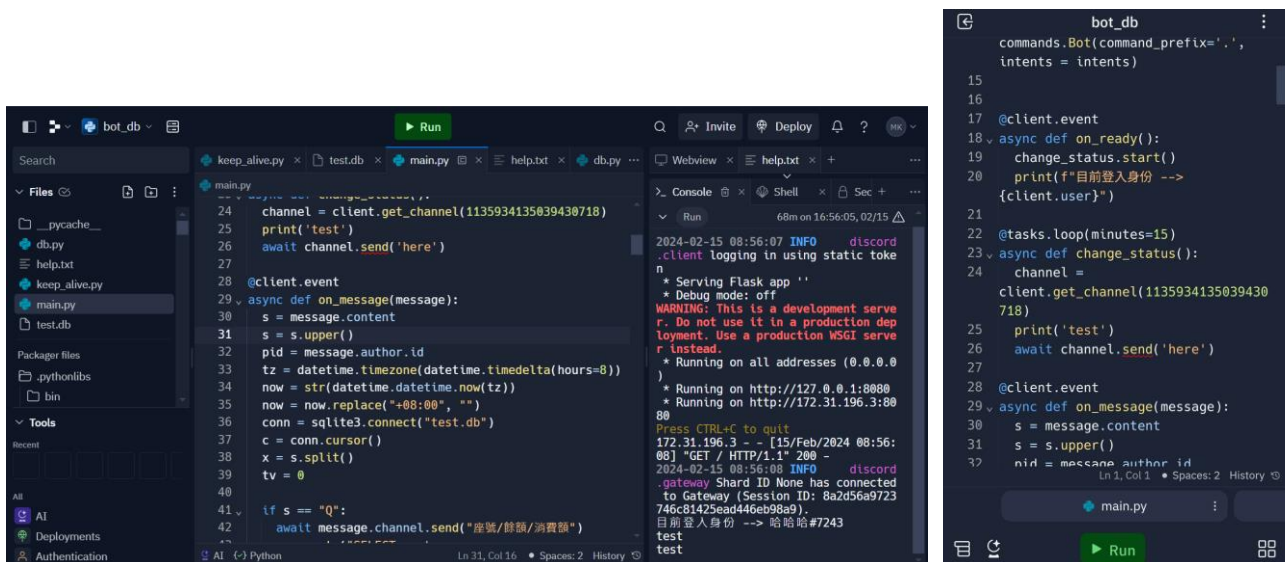


圖 3：(左)replit 網頁版執行程式畫面，(右) 手機板

### 3.3. 遇到問題：Replit 程式會自動斷開 → 改成讀取、改寫檔案

程式大致寫完要測試時，發現 Replit 執行程式後一段時間就會自動斷開，本來以 2 個 array 記錄餘額跟消費額，程式斷開後紀錄也會跟著消失。想起以前寫 C++ 學到的 i/o redirect 可以由讀取檔案文字取代預設的鍵盤輸入，如果把 array 的資料改到文字檔內，改成讀取檔案的資料，計算後再改寫文字檔，除了當作備份，也可以避免程式斷開後資料消失。再學習 python 的檔案讀取、改寫語法後，也修正程式改成從檔案讀取、改寫，測試後需要的功能都完成了。

## 4. 高三上學期 (2023.08)：機器人上線開始記帳

把機器人跟同學邀請進新開的 dc 伺服器，開始用 dc 機器人記帳。記帳頻道內我輸入的文字為交易記錄，機器人會自動回復餘額跟消費額在頻道內，解決了計算、消費記錄跟資訊公開的問題。因為怕同學自己亂記帳，伺服器的記帳頻道僅限我輸入文字，其他人只有唯讀權限。當時包含以下功能：

- 當我輸入同學座號跟消費金額 (前面加負號)後，機器人會自動算出餘額跟總消費額，如圖 4。



圖 4：我輸入【4 號花 10 元】，機器人自動算出餘額 79 元，在小福利社消費共 321 元

- 當我輸入同學座號跟儲值金額後，機器人會自動加在餘額上，如圖 5。



圖 5：我輸入【17 號儲值 100 元】，機器人自動算出餘額為 69 元

- 若交易後同學餘額為負數，機器人會另外標註同學提醒他還錢，如圖 6。



圖 6：機器人會自動標註欠錢的同學

- 一開始先輸入 A 代表提款功能，如圖 7，雖然此筆交易金額為負，但不列入消費額計算。



圖 7：我輸入【38 號提款 20 元】，機器人算出餘額

- 因應同學跟我提出有轉帳的需求，後來新增了轉帳功能(一開始先輸入 T)，如圖 8。

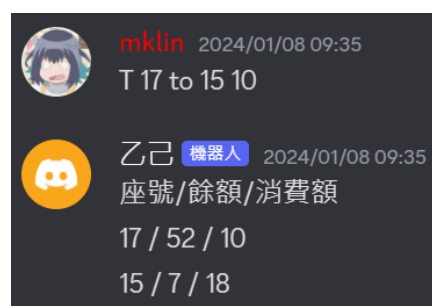


圖 8：我輸入【17 號轉給 15 號 10 元】，機器人分別算出兩人的餘額



機器人上線、資訊公開後，同學們也逐漸願意在我這裡記帳，截至高三上期末(2024.01)我已經有 16 個會員了。

除了記帳，我也在伺服器開了個訂貨區，讓同學可以預訂需要的東西，我去合作社時可以順便買，為大家帶來便利 (圖 9 左)。

而且開始記消費額後才發現我們班消費力驚人，光是跟我買零食泡麵就花了 7000 多(圖 9 右)。



圖 9：(左) 訂貨區讓同學可以預訂東西，(右) 學期結束時同學們的消費狀況

- 2023 年底結算我整年消費了 20591 元，回饋 1978 元，大約 9.6% (圖 10 左圖)。我也有以大家的消費額按比例分配回饋金給大家 (圖 10 右圖)。

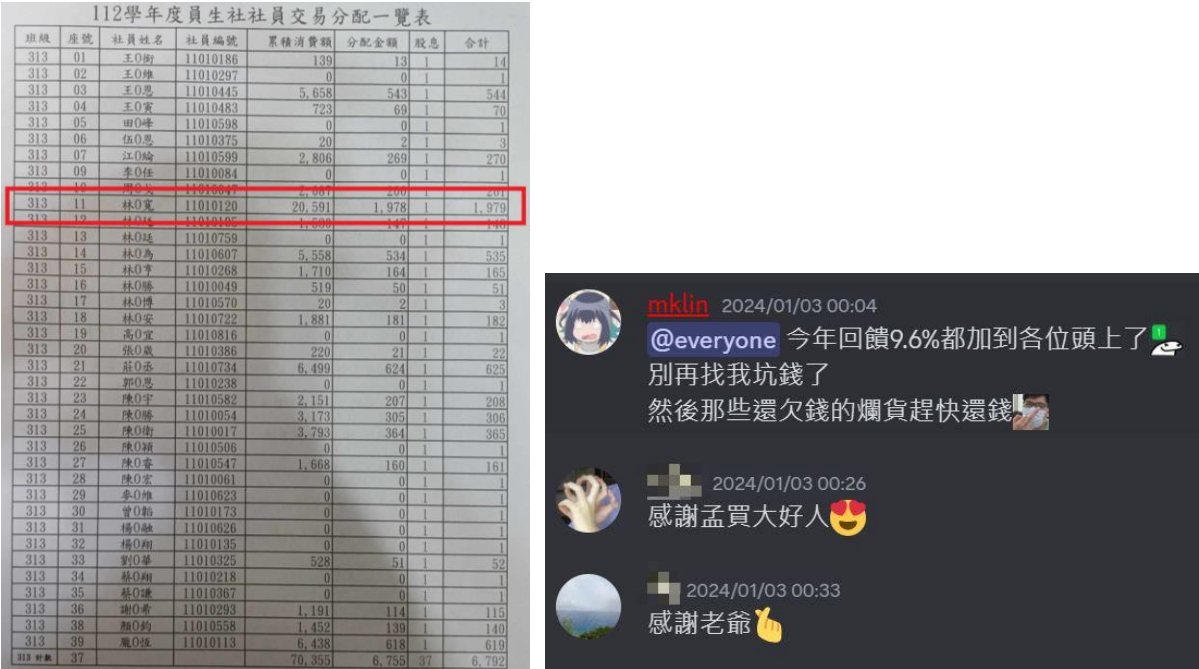


圖 10：(左圖) 2023 年累計消費額跟回饋分配明細，(右圖) 我是真的有把錢分給大家的

## 5. 高三寒假 (2024.01)：在資料庫新增消費與品項紀錄、同學可以自己操縱機器人記帳

在使用機器人一個學期後，雖然比起人工記帳方便許多，但我跟同學使用、討論後認為還是有些不便之處可以改進：

- 單純記錄何時消費多少，雖然有消費記錄，但只有消費額難以對帳，有記跟沒記一樣。
- 只有我能記帳，同學有時候找不到我就忘記記帳了，或是還要通知我比較麻煩。

剛好學測完有空寫程式了，我的目標是結合自主學習時自學的資料庫，希望能在資料庫記錄較完整的消費記錄，包含消費時間、金額和品項。且讓同學可以自己輸入訊息，操控機器人消費扣錢跟轉帳給同學，不用每次都透過我。

首先，我想在 python 中結合資料庫，我問 ChatGPT【How can I integrate database into python?】，得到建議是 python 內建的 SQLite。我建立兩個資料表【Member】跟【Trade】，分別記錄【每個人的座號、Discord ID 消費額跟餘額】跟【交易記錄】，並在每次輸入訊息後，分別在 Member 資料表更新 (update) 餘額和消費額、在 Trade 資料表插入 (insert) 新的交易記錄，包含消費時間、金額和交易種類 (提款、存錢、轉帳、買甚麼東西)。由於這部分的語法之前自主學習都實作過，對我來說問題不大。

接著目標讓同學能自己輸入操縱機器人，但同學輸入不能有提款跟存錢權限，所以輸入訊息時先取得輸入者的 Discord ID 判斷輸入者是誰，除了我的輸入者，訊息的第一個字母只能輸入 C (消費)、T (轉帳)、Q (查詢餘額)、All (查詢所有個人消費紀錄)。為了輸入方便，我把可能會帶回教室賣的商品 11 種分別給一個代號，買東西只要在 C 後面空一格輸入商品代號就能完成交易 (圖 11 左圖)，至於其他不在列表上的交易，則把代號改成消費金額 (圖 11 圖 11 右圖)。轉帳部分第一個字母輸入 T，再輸入轉帳給誰跟轉帳金額 (圖 12)。輸入 All 可查詢所有消費記錄 (圖 13 圖 13)。

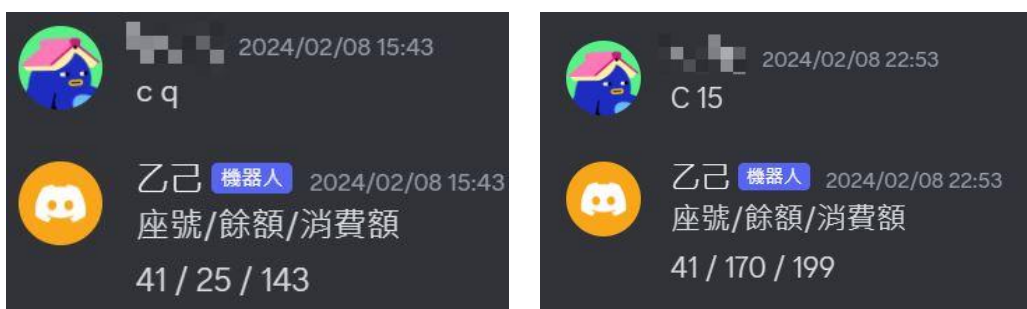


圖 11：(左) 由 discord ID 判斷輸入者為 41 號同學(測試帳號)，輸入【C Q】代表 41 號買一碗阿 Q 桶麵，(右) 輸入【C 15】代表消費 15 元





圖 12：41 號同學輸入【T 40 32】代表轉帳 32 元給 40 號同學(測試帳號)



圖 13：輸入 ALL 查詢個人所有消費記錄

因為輸入格式都是我自訂的，同學輸入問號時機器人會完整列出所有輸入教學。且輸入者不再只有我，防呆機制也是必要的，目前只有輸入不存在的商品代號時會提醒 (圖 14)。

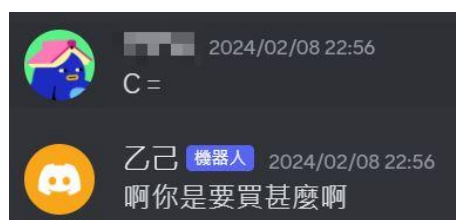


圖 14：輸入不存在的商品代號時會提醒，交易不成立

## 6. 高三下學期 (2024.02)：讓機器人持續在線上服務、同學開始自己記帳

本學期的目標是給同學控制機器人的權限，但程式跑在 Replit 上最多一小時就會斷開，無法因應同學隨時可能都會有消費、轉帳或查帳的需求，所以要想辦法讓 Replit 一直運作，由於之前自主學習有實作過 python selenium 抓取網頁資料，想法是用 selenium 開啟我的 Replit 頁面，並每隔一段時間檢查程式有沒有還在跑，如果斷開的話就點一下 Run 讓程式繼續跑。

### 6.1. 遇到問題：Selenium 語法新舊版本差異

一開始我都用以前學的語法寫程式。不過，像是以【driver.find\_element\_by\_css\_selector】通

過 CSS 選擇器來查找網頁上的元素，執行後卻是錯誤訊息 (圖 15)，上網搜尋後才知道舊的語法在更新為 selenium4.3.0 版本後便不再支援，於是我繼續搜尋後改成新的語法【driver.find\_element(By.CLASS\_NAME, name)】便可以執行了 (圖 16)。

```
In [3]: ► Q = driver.find_element_by_class_name('class="css-16mw829"')

-----
AttributeError                                Traceback (most recent call last)
~\AppData\Local\Temp\ipykernel_10600\1527629198.py in <module>
----> 1 Q = driver.find_element_by_class_name('class="css-16mw829"')

AttributeError: 'WebDriver' object has no attribute 'find_element_by_class_name'
```

圖 15：舊版 (3.141.0) selenium 語法無法執行

```
In [4]: ► from selenium.webdriver.common.by import By

Q = driver.find_element(By.CLASS_NAME, 'css-11iypm2')
```

圖 16：改成新版 (4.3.0) selenium 語法後便可執行

解決問題後，我也發現程式領域的變動真的很快，除了各種更新與新功能，才半年語法就不一樣了。

## 6.2. 遇到問題：Replit 在 selenium 視窗無法登入 → 改成 GitHub 登入

解決語法問題後，selenium 開啟的視窗顯示【captcha token invalid】，無法登入 Replit，如圖 17，猜測是 selenium 控制時 captcha 認證過不了，改成從 GitHub 手動登入就成功了。

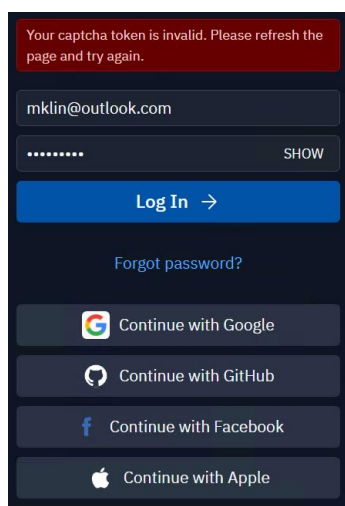


圖 17：captcha token invalid 無法登入

## 6.3. 成果

程式的流程為 selenium 開啟網站→手動從 GitHub 登入 Replit→亂數 50~70 秒檢查一次執行程式的 Run 鍵是否存在，如果存在代表程式已經斷開，再按一次 Run 讓程式執行，若找不到 Run

鍵代表程式還在執行。自己測試了一個週末 Replit 都還在執行，沒有因長時間而斷開，認為已經可以確保 dc 機器人會一直在線上服務，我也開始讓同學們可以自己記帳了，除了存錢、提款都不再用透過我進行交易了。



圖 18：同學們開始自己記帳

## 7. 總結

### 7.1. 會員人數

從高一下學期結束使用人工記帳時只有 7 個會員。dc bot 上線後 (2023.08)，會員數在高三上學期陸續成長至 16 人。2024.02 上線新版本的 dc bot 後，截至 2024.03 一個月又新增了 3 個會員達到 19 人。我們班目前 39 人(扣掉我算 38 人)，其中 19 人是我的會員，佔比一半，其實我們班跟我買過東西的有 30 人，這些人是我的潛在客戶，但他們主要都因為消費量不高、不想過於依賴記帳系統而沒加入我的會員。



圖表 1：會員人數長長趨勢

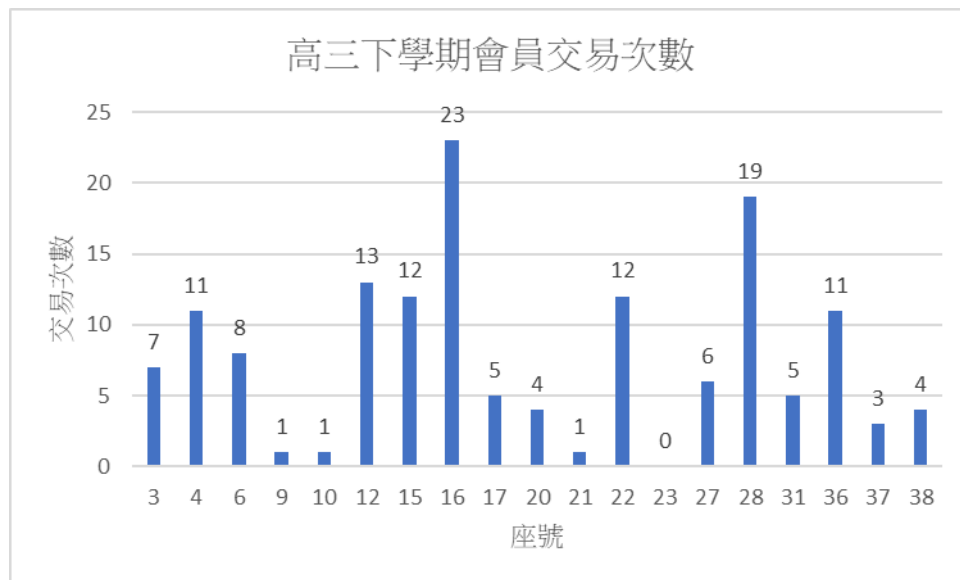
## 7.2. 營運狀況 (2023)

2023 年在合作社的總消費額為 20591 元，高一下學期沒有記錄消費額，只記得我跟同學當時大約花了 9000 元，而高三上學期我這邊陸續加入的 16 個會員總共消費了 7297 元 (圖 10 左圖)，剩下的約 4000 元則是我跟其他現金交易的非會員的消費額。

## 7.3. 營運狀況 (高三下學期開學後 5 週)

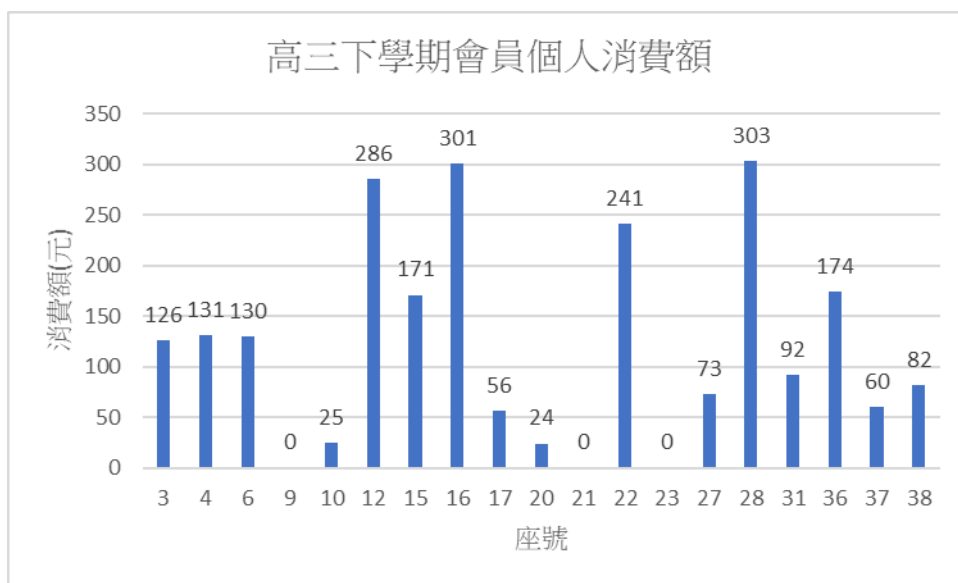
由於這學期除了記錄消費額，還以 SQLite 實作，新增交易類別、購買品項紀錄，透過 SQL 語法中的 group by 就可以輕鬆從資料表中根據特定項目篩選、整理出需要的資料，以分析顧客交易次數、偏好的品項等，並根據銷量去調整進貨量。以下統計資料範圍都只有 2024.02.16 至 2024.03.22 (高三下學期開學到開學後 5 週)，共 26 天。

- 交易次數：交易包含存款、提款、消費與轉帳，只要有錢的流動都記一次交易，目前有 146 筆交易紀錄，平均每天有 5.61 筆交易，個人交易次數範圍為 0 次到 23 次，多數同學的交易次數都落在 4~8 之間。



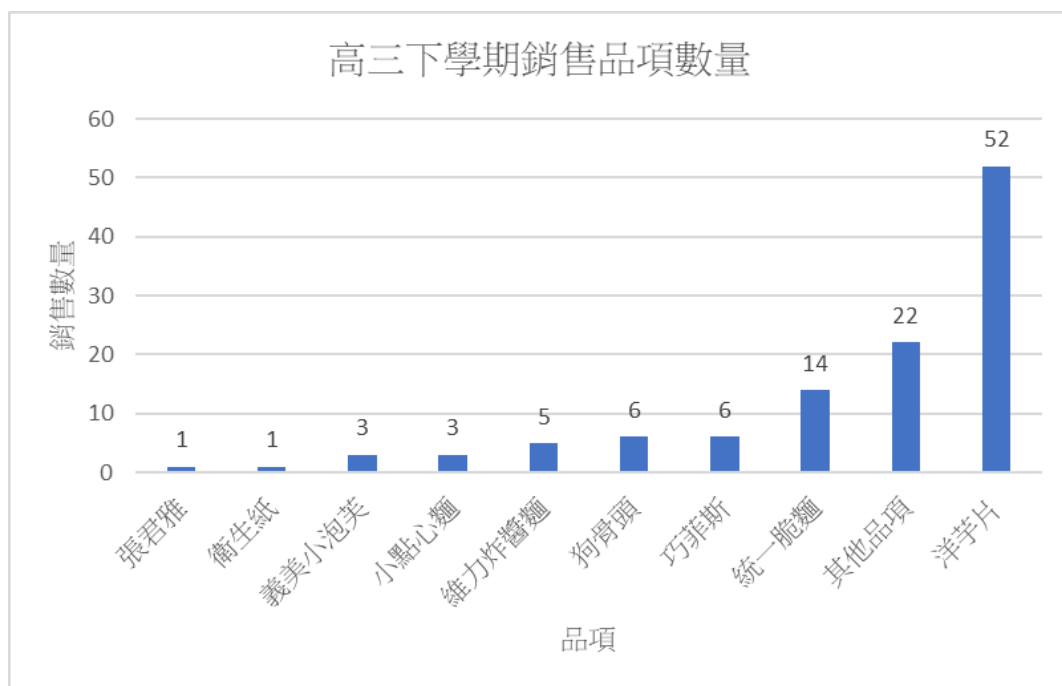
圖表 2：高三下學期交易次數分析

- 消費額：目前所有會員消費額共 2275 元，平均每天消費額 87.5 元，個人消費額範圍為 0 到 303 元，多數同學的消費額都落在 100~200 之間。



圖表 3：高三下學期消費額分析

- 各品項銷售狀況：銷售狀況最好的是洋芋片，成交 52 包，而張君雅這學期只賣了 1 包，和上學期印象中的銷售狀況差很多，猜測是因為單價在我的合作社偏高(25 元)又吃不飽導致銷售量低。以前進貨都只能憑銷量的印象採購，現在有詳細的銷況紀錄就可以更精確的調整進貨比例了。



圖表 4：高三下學期各品項銷售狀況分析

#### 7.4. 消費額比較

由於高三才開始記錄消費額，因此以高三上、下學期的消費額做比較。高三上學期 112 天(含暑輔)累計消費額 7297 元，平均每日消費額 65.15 元，而高三下學期 26 天累計 2275 元，平均

每日消費額 87.5 元，平均每日消費額成長了 34.3%，我認為新版的 dc bot 提供使用者消費機器人記帳的功能，除了吸引新的會員，舊會員不用再透過我記帳也比較方便，促進大家的消費意願。此外，我有一天突然想到既然有交易紀錄有流水號，就公告達成的 500、1000 筆交易的人再加碼 100 元(圖 19)，當天就有同學來跟我說他會好好努力拚第 500 筆交易的，刺激消費的活動可能也是讓消費額大幅成長的誘因之一。

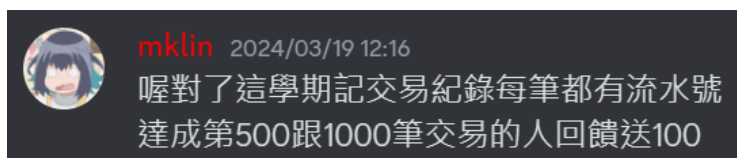


圖 19：突發奇想公告達成交易筆數再送獎金

## 8. 心得與反思

在整個營運小合作社的過程中，我的收穫如下：

### 8.1. 程式方面

- 善用程式解決問題：從一開始人工記帳，到後來寫程式讓機器人幫我記帳，為我省下很多時間、避免人為失誤，再次讓我體會到在生活中善用程式輔助可以解決很多問題。
- 事先定義問題：在每個階段開始寫程式之前，我都有先整理我的問題、訂出明確目標，寫程式時就很順利。跟之前用 C++ 解題有很大的不同，在有限的時間內解題時一看到題目就要直接開始寫，常常是邊寫邊思考下一步，而事先釐清需求、確立目標可以在腦中有比較清楚的程式架構，實作時才會更順、減少問題。
- 搜尋資訊解決問題的能力：由於之前自主學習結合了 ChatGPT 開發英文單字學習平台，程式的問題我習慣問 ChatGPT，通常也能得到想要的答案，但像是 6.1 提到的語法更新問題，因為 ChatGPT 的資料還沒更新到這麼新，無法得到正確答案。尤其程式領域的變動很快，才不到一年語法就變了，所以即使 ChatGPT 很好用，還是不能太過依賴，要有自己搜尋與辨識資訊的能力。
- 經驗：經驗的累積很重要，以前學的 SQL、Selenium 還有一些程式概念，沒想到在實作 dc bot 時派上用場了，由於之前都有稍微接觸過，讓我能快速的聯想到並實際運用，有效率的提升機器人的功能性。

### 8.2. 傾聽反饋

這是第一次有使用者直接參與我的實作成果，不像平常自己寫程式，從一開始人工記帳改機器人記帳，到機器人運作後陸續新增的功能，我都需要整合、觀察與考量大家的需求與問題並改進功能。因為程式和機器人的輸入規則都是我寫的我很熟悉，但使用者不一定覺得方便，同學也會跟我反映想要的功能或不習慣的使用方式，讓我在改善程式的同時意識到從使用者角度設計產品的重要性，除了 UI，能帶給使用者良好 UX 才能做出讓人想用的產品並持續經



營。

雖然現在以輸入的文字字串處理去簡單的控制機器人，在使用者 20 人左右的情況下營運問題不大，通常都是同學輸入格式錯誤或打錯字而已，我可以個別跟同學解釋我的輸入規則，或是後端直接從資料庫改掉錯誤的資料就好。但是如果系統的規模變大，我現在這些簡單粗暴的解決辦法就沒用了，根據目前遇過的問題，應該針對存貨狀態、取消交易或輸入格式防呆機制等訂出更嚴謹的規則並優化程式與功能，才能真正做出一套完整的系統開放給更多使用者。

### 8.3. 營運方面

雖然我做這個沒有營利目的，但還是會有興趣去想一些營運活動刺激消費，像是回饋金、抽獎或是消費筆數達標，事後看來這些活動成效都不錯，除了提高原本的會員的消費意願，也常常透過這些有趣的活動吸引到其他同學加入會員。這些在經營客戶關係上的經驗讓我更了解消費者的喜好，並感受到好的應用服務，除了系統技術，還須要針對客戶需求不斷改善 UI 和 UX。

## 9. 後記

- 乙己的出處：出自高一下學期國文課本《孔乙己》—「孔乙己還欠十九個錢呢！」

有一次我想起陳同學還欠我 30 元，我就很順的說了：「陳乙己你還欠我三十個錢哩！」，同學們聽到後都大笑，教室內充滿了快活的空氣。從此以後，我們班對欠錢的代稱就變成「乙己」了。又因為同學來我這裡買零食是記帳，常常買一買就不小心負債了，所以 dc 機器人就取名乙己，代表同學們常常負債的情況。

- 我的合作社就長這樣，一個紙箱裝著零食泡麵，但也裝著我跟同學的快樂與回憶，尤其高三學測衝刺時，這個箱子為大家帶來了快活的空氣。

