

居家電量分析系統 期末報告

第三組 00957050張銀軒 00957009鄧暉宣

一、製作動機

因應今年(2023)4/1 電價上漲，一般家庭的電費成本提高，在萬物齊漲的情況下，我們想用成本較低的軟、硬體達到掌控居家用電的目標。我們希望使用者在透過我們開發的系統能隨時監控、分析並預測用電量是否過高或異常。除此之外我們的系統也可以應用在無獨立電表的租屋中，用以計算每位房客的電費。

二、相關作品

1. 一般電力檢測儀

如一般的三用電表或是帶螢幕的插座電力檢測儀，只能透過螢幕查看目前的電壓與電流，無法將資料上傳並統計，也只能檢測單一插座。

2. 大葉資工系智慧物聯網電力監測系統

能夠監測各個插座的電力消耗，但是將工廠做為主要客群，較無考慮成本問題，一個插座約需 2000 元，過高的成本讓一般民用無法負擔。

3. 小米智慧插座

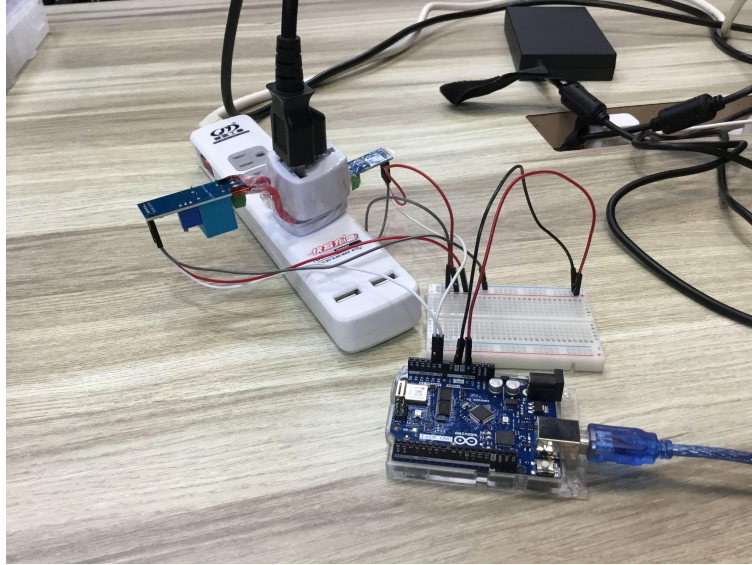
能夠監測並用手機 APP 查看使用電量，但台灣版的插頭已停產，且APP的功能沒有完整的電量監測、分析與預測功能。

三、系統架構/展示

Arduino & 感測器 & 插座

我們改造了一個萬國轉接頭(圖一)，使用電壓感測器 ZMPT101B 與電流感測器 ACS712 來取得插座的電壓及電流。

Arduino 程式碼會先連接到 Wifi。然後，它會分別取得並計算電壓及電流，從而獲得真實的電量。這些數據將每 5 秒透過 Wifi 傳送到後端伺服器來儲存。



(圖一)

後端

我們使用 express.js 作為後端，它的好處是開發 API 相對簡單且快速。

資料庫的部分我們使用 MongoDB Atlas，其特別之處在於降低開發的難度，不需要架設伺服器，只需要在Atlas的網站上進行操作即可。此外它也能簡化資料的儲存，可以以物件的形式儲存的資料庫上，NOSQL 的方式也讓開發者不必考慮資料庫的正規化，專注於資料的儲存。

而後端提供的 API 功能如下，並且所有 API 都能分辨用戶：

- 註冊/登入。
- 插座、類別的 CURD。
- 查詢 單一插座/單一類別/全部 在給定時間區間的 電費/電量。
- 若查詢的時間包含超過現在時間，則會給出預測未來的電費/電量。

AI預測

我們使用 REFIT Electrical Load Measurements 資料集，他蒐集了幾個家庭在幾年之間的用電量，並使用 LSTM 作為預測未來用電量的 AI 模型，因為其在預測有迴歸特性的數據比 Transformer 更良好。我們架設了簡單的多層 LSTM 模型，並讀入前一段時間的用電量數據，預測下一個時間點的用電量，再以此做為新的輸入，持續預測接下來每個時間點的用電量，直到使用者要求的時間。

前端

我們使用 Flutter 作為前端的開發框架，因為他能快速開發並展示後端的功能，未來也能在 Android、iOS、Windows 等平台上執行。

使用者登入介面(圖二)

可以讓使用者登入自己的帳號來查看、管理插座及電費。

插座、類別總覽(圖三)

列表式的呈現所有插座，使用者可以透過點擊類別來展開、收合各類別的插座，無類別的插座會放在 Uncategorized 中。而插座/類別右側會有目前消耗的電量、電費，可以透過上方的第一個插座按鈕來切換檢視。將插座/類別向左滑，會出現刪除與修改相關資訊的選項。

插座、類別增刪操作(圖五~圖八)

點擊右上角的第四個更多選項的按鈕可以執行增加插座/類別，或刪除全部的操作，當點擊時會出現如圖中的對話框，要求輸入名稱等資訊，或是確認是否刪除。

總電量/電費 時間區間查詢(圖四)

點擊下方導覽列右邊即可查看該頁面，使用者可以在此查詢一段時間內使用的總電量即電費，輸入起始與結束時間後，點擊下方的查詢按鈕即可看到上方出現統計資訊。當輸入的結束時間超過目前時間時，系統就會用AI根據之前的數據來預測接下來的用電量與電費，方便使用者管理未來的電力消耗。另外，我們也提供一個現在時間按鈕能查詢從過去到目前的總電量。

The image displays three screenshots of a mobile application interface for managing power sockets and electricity usage.

Figure 2 (Left): Login Screen
The login screen features a blue header with a back arrow and the text "Login". Below the header, there is a text input field labeled "Username" containing the text "jason". A blue "Login" button is positioned below the input field.

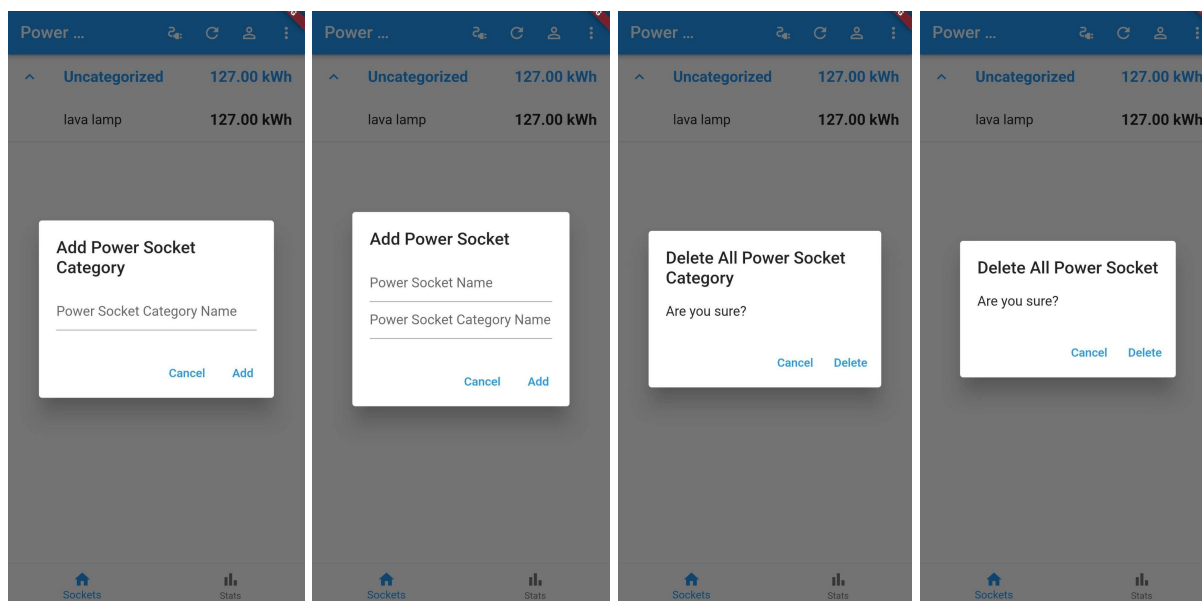
Figure 3 (Middle): Power Sockets Overview
The overview screen shows a list of power sockets categorized by room. Each category is expandable, indicated by a chevron icon. The categories and their associated sockets are:

- bedroom** (32.07 kWh): lava lamp (32.07 kWh)
- kitchen** (0.00 kWh): oven (0.00 kWh), microwave (0.00 kWh), dishwasher (0.00 kWh)
- living room** (0.00 kWh): tv (0.00 kWh)
- study** (0.00 kWh): acer laptop (0.00 kWh)
- Uncategorized** (0.00 kWh)

At the bottom, there are two navigation icons: a house icon labeled "Sockets" and a bar chart icon labeled "Stats".

Figure 4 (Right): Electricity Usage Statistics
The statistics screen displays the total electricity usage and cost. It shows "總電量" (Total Electricity) as 0.00 kWh and "總電費" (Total Electricity Cost) as 0.00 NT. Below this, there are two input fields for the time interval: "起始時間" (Start Time) set to 2010-01-01 00:00:00 and "結束時間" (End Time) set to 2030-01-01 00:00:00. At the bottom, there are three buttons: "現在時間" (Current Time), "test", and "查詢" (Query). The bottom navigation bar shows the "Stats" icon selected.

(圖二、圖三、圖四)



(圖五、圖六、圖七、圖八)

四、結論

使用我們的系統可以達到：

1. 能測量電量，並把數據上傳伺服器。
2. 達到降低成本的目標(電壓感測器:85元、電流感測器76元，2.0mm電線約4元，不考慮晶片的情況下，總共160元即可，比起市面上大多數的產品都還要便宜)。
3. 提供監測和預測，讓使用者能夠找出家中的吃電怪獸，也能在電器因電壓等非人為因素產生損害時，提供數據證明來保障自己的權益。

五、未來展望

硬體

1. 需要了解感測器的用線規格(粗細等等)
在連接感測器與插座的過程中，遇到了許多困難

- a. 電線的選擇

考量到我們連接的是110V的電，我們最終選擇一個2.0mm的線，然而這條線與電流感測器的規格不符，最終只能用斜口鉗把電線削細，也因為尺寸不合，無法使用感測器提供螺絲鎖緊，導致無法穩定固定，為後續可能出現的接觸不良問題造成了隱憂。

- b. 硬體結構不穩定

一開始我們打算在插座上面挖洞，透過插座上的洞連接感測器與插頭的鐵片，但是2.0mm的電線又粗又硬，很難達到預想中的結構，所以最終只能把插頭拆開，讓待測電器以不穩定的方式插到插頭上。

2. 把感測器、Arduino與插座整合成一個完整的成品，不會有線外露
 - a. 在開發的過程中，因為兩個感測器與Arduino 連結的接頭露在外面，一直很擔心會壓壞接頭。
 - b. 圖片上感測器的位置並不影響其他電器的使用，但是若遇到方向不同的插座，有可能會擋住其他電器使用插座。

Arduino

1. 連接 Wifi & 註冊插座

目前 Arduino 的程式碼中，WiFi 的SSID和密碼是寫死的。在現實生活中，每一個家庭的WiFi 帳號皆不同，最理想的方式應該是使用者透過手機偵測到附近智能插座，並且透過app輸入帳號密碼使插座能夠連上網際網路。連線上網之後，便能夠透過app編輯插座詳細資訊，接著註冊插座，最終開始上傳統計數據到伺服器。
2. 電壓、電流計算的精確度

在 Arduino 的程式碼中，對感測器收集到的數值額外做處理，使數據能符合瓦特/安培，也就是說感測器並不是直接偵測電流和電壓，如果能深入瞭解處理的過程，我們便能知道那些數據是不合理的，進而捨棄這些數據並通知使用者家中可能出現電力異常的情況。
3. 斷網時暫存資料

目前插座在斷網的狀況下，資料遺失了。更好的做法是在上傳失敗/斷網的情況下，把收集到的數據儲存，等連線恢復之後，再把這些資料上傳到伺服器。

後端

1. 電費計算方式

目前是以開始時間決定是否以夏季來計算，實際上若牽涉到橫跨夏季&非夏季的查詢，應該要分成2組計算之後再相加。
2. API 設計

目前取得所有類別、插座、插座資訊的 API 都是分開的，如果提供一個能獲得所有使用者資料的 API 能讓系統的處理速度獲得提升。
3. AI 預測改良

目前的模型設計精確度不足，需研究 LSTM 模型該如何調整，並且應該以歷史幾年的用電量作為輸入，加上目前使用的電器種類，才會考慮到不同季節不同用電量的情況。且當預測的時間過長時會執行太久，可以改用以不同的時間單位來預測，加速模型的速度。

前端

1. 資料顯示方式

可以用折線圖來顯示歷史的電量、電費，讓使用者能更快速解讀過去的用電情況。在插座的顯示上，除了類別條列之外，也可以用2D/3D的位置圖顯示每個插座在家中的位置，在管理插座上能更方便。