

Embedded Systems Labs

Final Project - Intelligent Vending Machine

第15組

b03901016 陳昊

b03901022 卓伯鴻

b03901152 陳景由

分工：陳昊 - rpi軟體實作以及硬體架構

卓伯鴻 - 硬體線路及外形

陳景由 - 網頁server client實作

1. Introduction

本次實驗的成品是自動販賣機，支援三種不同的商品，藉由rpi的gpio控制輸入以及輸出的訊號，精確的使指定的物品掉落，並且同時在傳送消費記錄到聯網的伺服器上，記錄販售的內容以及品項，時間等等，此外更支援多種不同型式的圖表，以不同時間間隔週期顯示販售的統計資料，清楚呈現所有的銷售記錄。

2. 平臺使用軟硬體

- ☐ Raspberry pi 3
- ☐ LCD顯示器
- ☐ servo伺服馬達
- ☐ python3(操控馬達以及顯示器和接收rpi之GPIO訊號，為主要功能的實作)
- ☐ javascript(販賣機用以傳送消費記錄至伺服器之client實作以及架設server網站)

3. 功能簡介

使用者可以在觸發欲購買的商品項目Trigger後，於LCD屏幕上得知自己鎖購買的商品資訊，並或得商品。此外，另有雲端同時記錄使用者的消費資訊，以便使用者自己得知購買記錄，並以一目瞭然的圖表清楚比較購買物品的銷售情形及購買數量資訊等等，並有線上充值的機制避免使用者在忘記攜帶隨身錢包的情況下無法購買販賣機內商品的情形。

4. 詳細功能介紹

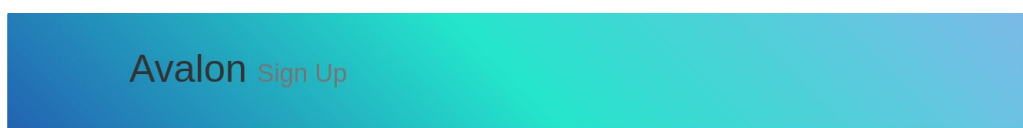
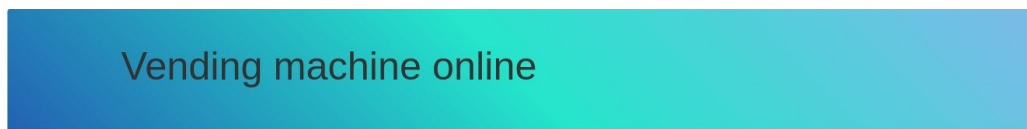
網頁部分：

整體架構是用react寫成，配合chart.js做出圖表，而圖表的數據有連結到socketio，也就是說若我們有新的交易產生，對於所有目前在線上觀看銷售情況的使用者們都可以馬上得到最新的結果。

登入與註冊：

在使用此智慧販賣機之前，使用者可以先行上往註冊一個帳號，方便以後多種實用功能的支援，除了可以看見購買商品的記錄之外，還可以線上充值等等，非常實用。而使用者的資訊用是以mysql的方式儲存在資料庫中方便管理。

(網頁登入圖示)



銷售紀錄：

當販賣機有物品販售出去時，都會自動傳送一筆交易記錄給連線的server。每一筆銷售紀錄都會儲存品項名稱，物品金額等有關商品的基本資訊，以及其銷售時間，選取不同的時間單位將會即時排序，加總單次消費後顯示結果，消費者可以清楚的檢視自己的購買紀錄，甚至可以藉此作為未來消費的參考因素。

Log outaaaaaa profile

#	Product Name	Purchase Time	Amount	Sum money
0	coke	Thu Jun 11 2017 20:50:15 GMT+0800 (CST)	1	25\$
1	apple juice	Thu Jun 13 2017 20:50:15 GMT+0800 (CST)	1	25\$
2	coke	Thu Jun 15 2017 20:50:15 GMT+0800 (CST)	1	25\$
3	7up	Thu Jun 15 2017 20:50:15 GMT+0800 (CST)	1	25\$
4	apple juice	Thu Jun 15 2017 20:50:15 GMT+0800 (CST)	1	25\$

Back

銷售圖表：

將在一個時段內的所有品項數量加總，畫出數量-時間的圖表，目標客群主要是經銷商，經銷商可以分析最近哪些商品比較熱門。還有熱銷的時段，以擬定未來的銷售方案。

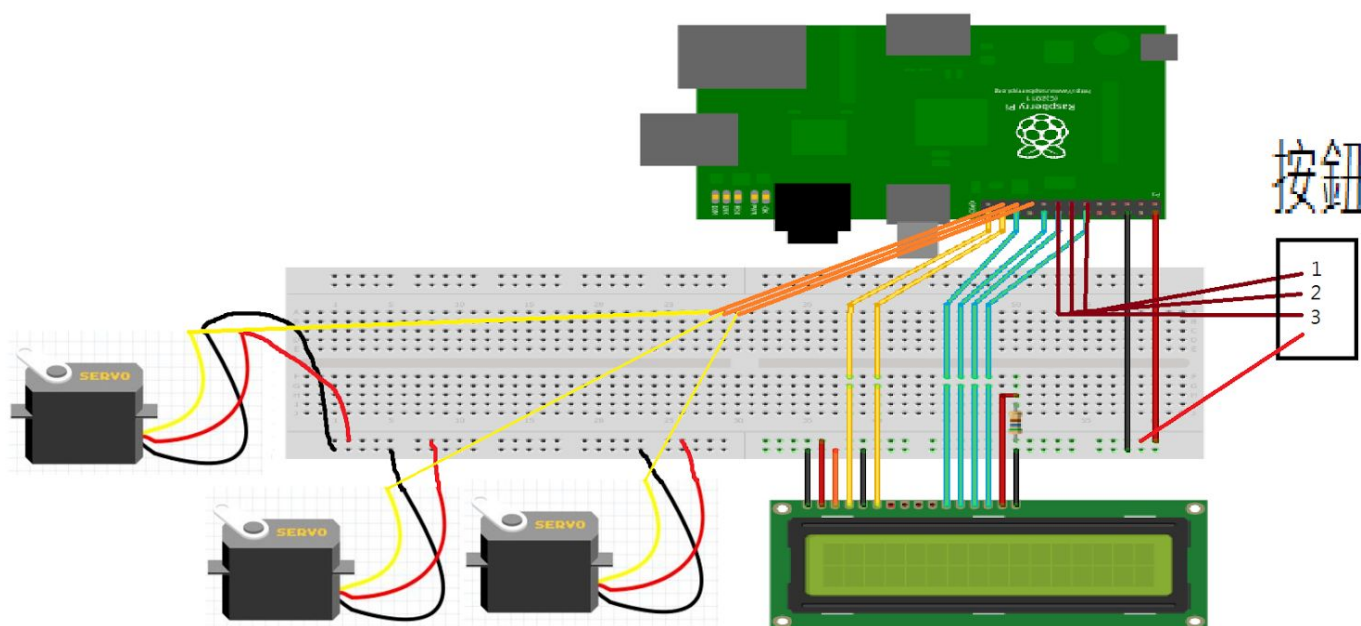


RPi3 實作：

1. 連結RPi3 client端和其主要的python物理運作程式：

要使python端和javascript的client端能夠互相溝通，我們欲建立使client端即時監控python的output訊號，起初這邊我們卡住了許久，是沒有考慮到標準輸出的flush和buffer問題，找了很久才解決。使用nodejs中利用child_process開啟python來運行RPi3，購買商品一時RPi3將會用stdout的方式輸出“1”，購買商品二時RPi3將會用stdout的方式輸出“2”，並且驅動servo使商品掉出，依此類推。nodejs接受到該購買商品的請求後(stdout)，就會用fetch發給server。server更改當前儲存資訊後會在用io socket傳送給目前所有的連線者資訊已經被更新。

2. 電路架構：



主要設計：

從這張圖可以看出我們主要的線路分成三個部分，第一，是按鈕的部分；第二，是servo的部分；第三，是雙排LCD顯示器的部分。第一，按鈕的部分，我們是使用rpi3之GPIO的high跟low的切換來判定，是否該按鈕有接通（有接通的話是1，沒接通的話是0，若input為高電位，則將驅使與此訊號線連結的伺服馬達轉動，推進物品向前移動並掉落，始得使用者可以拿到。而servo是如何來推進物品向前進？我們是使用servo馬達的旋轉的方式，來讓下圖中地鐵絲開始旋轉，來帶動存放在上排空間中的物品。第三，雙排LCD顯示器的部分，就是直接從rpi3 GPIO傳出訊號並接至放在箱子前版的顯示器。

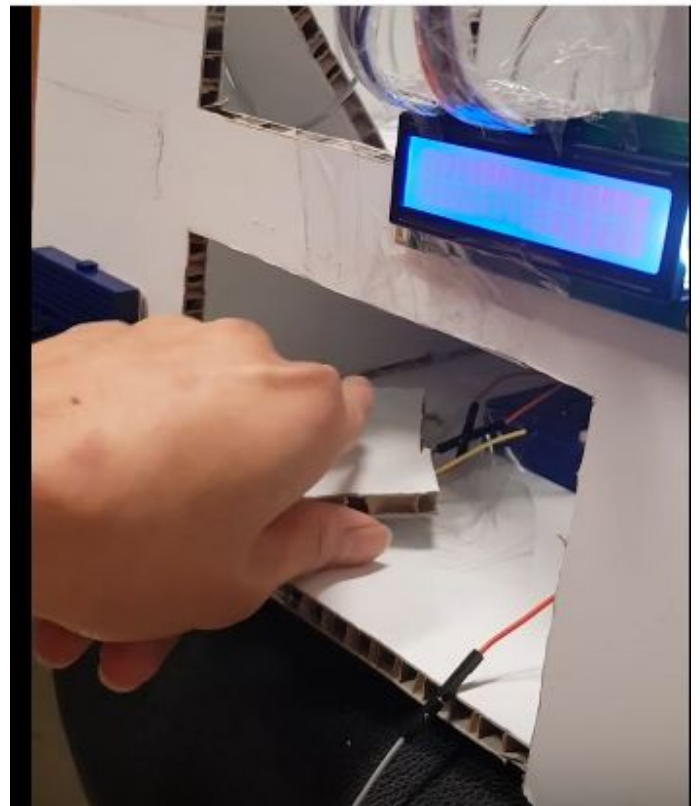
遇到的困難：

我們在處理rpi接module上遇到了一些困難，主要是發生在找到合適可用的module，然後還有範例的documentation上。還有就是結構上的，我們在處理麻煩的線路接觸的問題，特別是雙排LCD顯示器的部分，因為這個module需要特別多的GPIO連接，並擁有一排的in/out的port，所以我們用了相當多的膠帶在控制電線的方面，在之後的實驗中，可能要針對這種狀況，找尋更多的束帶，或是能輔助杜邦線轉接的用具。

(圖一，裝置上拍圖)



(圖二、三，裝置前拍圖)



3. 影片連結:

小故事 : <https://youtu.be/Ni4I3oT3vaw>

demo vedio : <https://youtu.be/C0HWIbc0fo0>

demo: <http://52.179.13.42:3000/> 工作站會再幾天後關閉

code: <https://github.com/b03901152/2017ESL/tree/master/final>