

中華民國第 60 屆中小學科學展覽會

作品說明書

國中組 生活與應用科學(一)科

第三名

032803

機不可失－建構教學用平板電腦推車之借還平台

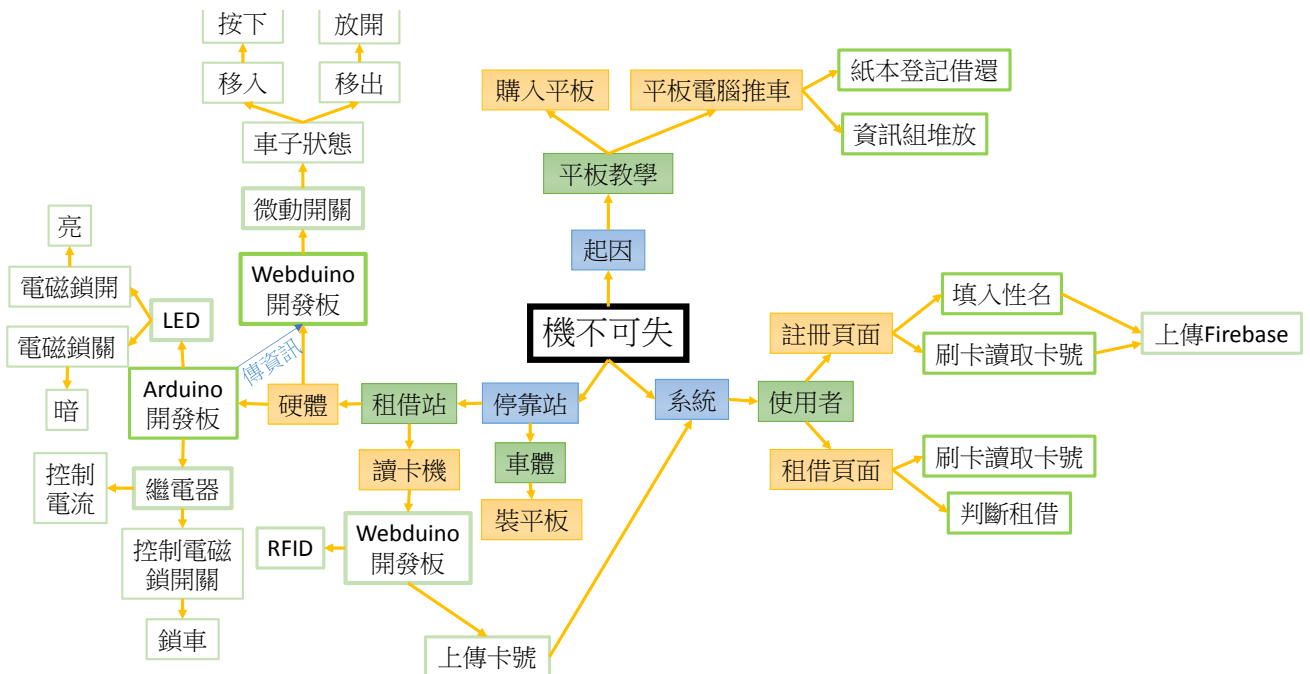
學校名稱：臺北市立敦化國民中學

作者： 國二 李綦瑀	指導老師： 任建安
---------------	--------------

關鍵詞：Arduino、Webduino、Firebase

摘要

看到校園內的平板車租借，需要到資訊組用紙本登記，而因為學校電梯只有一台，等電梯耽誤了很多時間，而且紙本消耗了很多資源，因此想做一個可以利用悠遊卡註冊、借車、歸還的系統，將停靠站放置各個樓層，有需要時可隨時去借還。此次成品製作是利用 Webduino、Arduino 開發板，藉由網路連動 Firebase 資料庫，逐步完成租借系統。經過了多次修正與測試，已完成可註冊、租借的系統平台，但因時間和成本有限，目前只能以模型測試，期待日後能製作出停靠站並實踐於在平板車的租借。



壹、研究動機

以前學生上課不管是抄筆記、寫功課、考試、看書等等都使用紙本，造成了很多環境資源的浪費。近年來台北市政府開始推動平板教學，學校為此購入許多平板車，這些平板車大部分都停放在狹小的資訊組內，每到借車的高峰時間，資訊組總是人滿為患，大家搶著要登記，這樣使租借車子要花很多時間，也讓許多老師望而卻步。

這些缺點包括：第一，租借平板車有許多時間是花在填寫租借記錄簿，只要多人同時租借，就得依序等待填寫。第二，雖然使用平板教學減少了紙張用量，但用紙本登記還是浪費

了很多的紙，且紙本登記容易遺失，不便於登記、查詢、追蹤等動作。第三，如果任課班級與資訊組在不同樓層，租借與歸還都必須搭乘學校裡那一台又舊又慢的老電梯，這樣也花了不少時間。以上這些原因使租借平板車變得又麻煩又浪費時間，讓許多老師不願意嘗試，因此，我想利用 Webduino 物聯網開發一個使用感應卡就能隨時借還平板車的系統，它能縮短借還時間、不用紙張，而且資料都能上傳到網路儲存，且管理員隨時隨地都能查看修改，也因為資料存在雲端，相較紙本更容易尋找，最後只要各個樓層都能設立停靠站，就能減少等待電梯而浪費的時間了，再加上我們使用電腦程式進行判斷，時間若長久人力成本就會大幅下降了。

在製作作品的過程中我使用了在七年級生活科技課程中學習到有關問題解決的方法，在資訊科技課程中學習到有關積木程式的撰寫與應用。另外，在八年級生活科技課程與九年級的理化課程中，學習到有關電與控制的知識與製作。

<p>平板推車目前借還方式</p> <p>紙本登記租借</p> <p>教師使用者</p> <p>租借</p> <p>歸還</p> <p>紙本登記歸還</p> <p>平板電腦推車放置在資訊組辦公室</p> <p>推至各班教室上課</p>	<p>目前校園內平板電腦推車的借用方式是先到資訊組利用紙本登記後，再將車推至各班教學。教學完畢後，再推回資訊組歸還，並在紙本登記。容易造成紙本浪費，且不容易尋找借用者的紀錄。</p>
<p>利用物聯網修改平板電腦推車借還方式</p> <p>雲端物聯網</p> <p>軟體：網頁、Firebase資料庫</p> <p>資料檢核更新</p> <p>車輛解鎖、上鎖</p> <p>使用者</p> <p>刷卡 RFID</p> <p>租借</p> <p>歸還</p> <p>推至各班教室上課</p> <p>平板電腦推車放置停靠站</p> <p>停靠站硬體：RFID感測器、電磁鎖、Arduino、Weduino開發板</p>	<p>將 Webduino、Arduino 開發板、RFID 感測器、繼電器製作停靠站，利用物聯網與雲端資料庫 Firebase 連結。使用者利用刷卡方式註冊、登入資訊來借用平板電腦推車。租借狀態與時間可直接紀錄在雲端資料庫中，方便查詢且節省紙張。</p>

貳、研究目的

- 一、用卡註冊基本資料及登入
- 二、知道卡片是否註冊過
- 三、租借或歸還平板車的停靠站
- 四、得知車是否放置至正確位置、是否移出
- 五、管理者可得知使用者姓名、登入及借還的日期時間、租借狀態

參、研究器材與設備


- 一、工具類：熱熔膠槍、手工線鋸、美工刀、手搖鑽
- 二、材料類：電池、變壓器、杜邦線、密集板、紙箱
- 三、開發板與感測器元件：Arduino 開發板、Webduino 開發板、RFID 感測器、微動開關、電磁鎖
- 四、資訊設備：Firebase 資料庫、電腦、照相機

肆、研究過程與方法



- 一、確認問題
 - (一) 如何設計製作出一個可借還平板車的裝置及系統
 - (二) 如何不利用紙本登記，就可租借平板車
 - (三) 如何判斷是否有此使用者
 - (四) 如何得知使用者目前是否有租借平板車
- 二、設定目標
 - (一) 製作可藉由 RFID 註冊人員的系統平台
 - (二) 製作可藉由 RFID 租借平板車的系統平台
 - (三) 製作出平板車租借停靠站模型

三、蒐集資料

(一) Ubike 租借方法

<p>租車時直接在停車柱上的感應區感應悠遊卡，確認綠色警示燈開始閃爍之後雙手用力把車向後拉出即可使用。還車時將車擺正推入到底。鎖上後藍燈會閃爍並發出短鳴聲，在感應區感應悠遊卡會自動扣款，待燈號跳回第一個綠燈，即完成還車手續。</p>	
<p>此次研究以 Ubike 借還平台的方式為基礎進行發想。利用自行設計的刷卡機、雲端資料庫、停靠站，配合程式設計及物聯網方式，製作出平板電腦推車的借還平台。</p>	

(二) 開發板介紹與比較

	Arduino Uno	Weduino Smart
開發板圖示		
開發語言	C/C++	HTML/JavaScript
開發環境	Arduino IDE	瀏覽器、網頁編輯器
連接方式	USB、藍芽	Wifi、Websocket、USB、藍芽
更新程式	連接燒錄	立即更新、連接燒錄
運作方式	通電後即可一直運作	網頁驅動，若關閉網頁則停止運作(需將程式寫入後端伺服器中)
<p>此次研究中，利用 WebduinoSmart 開發板與 RFID 感測器連接，傳送資料至雲端資料庫比對分析，由於程式碼可由 Blockly 圖形化介面完成，再修改 JavaScript 程式的方式較易編寫。但 Webduino 開發板需要網頁驅動才可進行</p>		

運作，而電磁鎖需要一直運作狀況下，藉由 Tx、Rx 腳位連接，讓 Smart 可以驅動 Arduino 開發板進行運作。即使網頁關閉，Arduino 依然可以持續運作使得電磁鎖依然是上鎖狀態。

(三) 電磁鎖

電磁鎖介紹又稱磁力鎖，用電生磁的原理，當電流通過矽鋼片的時候，電磁鎖會產生強大的吸力緊緊的吸住鐵板達到鎖門的效果，斷電後就會立即失去磁力。



此次研究中，選用電磁鎖是可利用開發板控制電流通路與斷路，進而達到停靠站對平板電腦推車開鎖與解鎖的目的。

(四) 感測器

1. RFID



RFID 是 Radio Frequency IDentification 的縮寫，中文翻譯為「無線射頻辨識」，是一種常見的無線通訊技術，透過 RFID 識別裝置（讀卡器）所產生的「電磁場」，能讓附著在物品上的「電子標籤」獲得能量進而發送無線電頻率的訊號。

此次研究中，RFID 是用來辨識使用者身分的物件，藉由刷卡機得知卡片編號，與使用者作為連結。利用 RFID 刷卡登入比一般利用網頁點選輸入資訊較為快速便利。

2. 繼電器




繼電器是一種電子控制器件，它具有控制系統和被控制系統，通常應用於自動控制電路中，是用較小的電流去控制較大電流的一種「自動開關」，在電路中起著自動調節、安全保護、轉換電路等作用

此次研究中，繼電器連接在 Arduino 與電磁鎖之間，控制繼電器的開啟與關

閉，使電磁鎖開鎖與解鎖。另外，由於電磁鎖是用 12V 電壓驅動，為避免大電流回衝至開發板中，繼電器是相當重要的存在。

(五) Firebase 資料庫

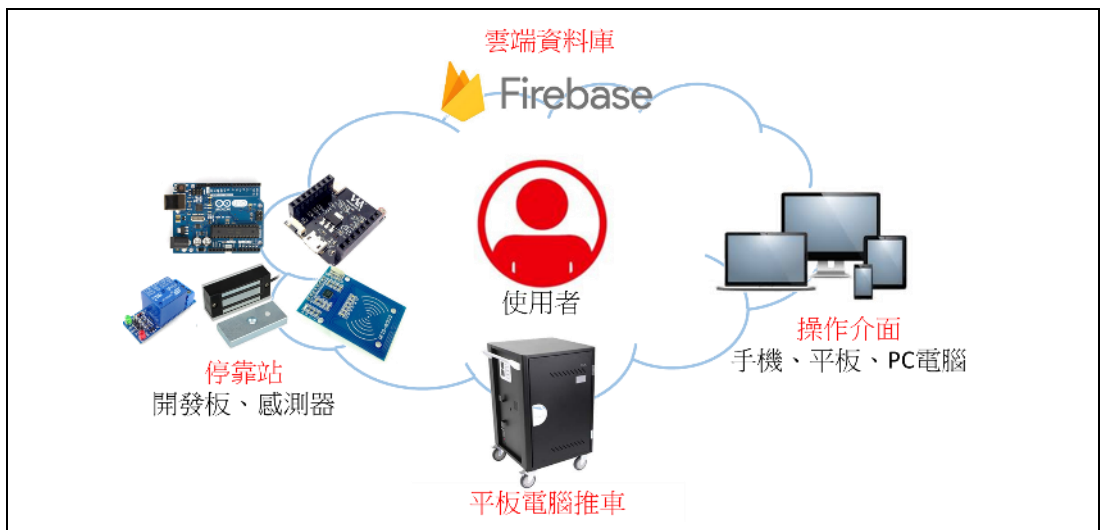
	Firebase 是一款雲端資料庫，它提供了身分驗證 (Authentication)、即時資料庫 (Cloud Firestore)、儲存 (Storage)、主機 (Hosting)、儲存 (Storage)、函數編寫代碼 (Function) 及機器學習功能 (ML Kit) 等後端服務。
此次研究中，使用 Firebase 的即時資料庫進行資料更新(update)與讀取(on)的功能進行資料傳輸與分析判斷。	

四、構想解決方案

(一) 借還系統概念：

類似圖書館租借圖書或是 Ubike 租借公用腳踏車的平台，想利用學校課程與社團所學習的資訊科技概念進行設計製作。首先，要有註冊系統，方便使用者可以即時註冊帳號並租借平板車(參考 ubike 的做法)。RFID 感測器可以支援提供人們掃描悠遊卡進行登入，也節省輸入帳密的時間。登入系統後，可在短時間內將平板車借還完成，並將借還資訊傳送至雲端資料庫登入，以節省現階段紙本登記的消耗浪費。

(二) 平板電腦推車借還平台概念圖



由使用者角度開始發想，需要將停靠站、操作介面、雲端資料庫、行動平板推車彼此連接在一起，利用物聯網的方式，進行設計。

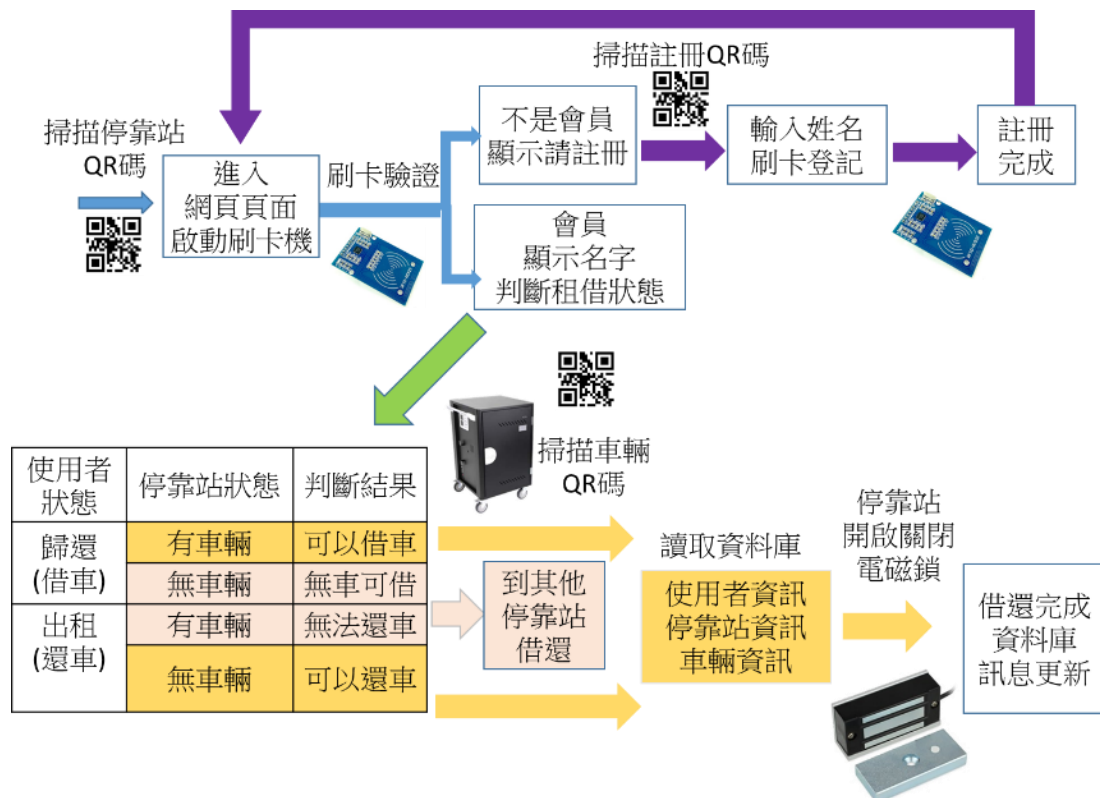
1.停靠站：由 Arduino、Webduino 開發板、RFID 感測器、繼電器、電磁鎖、微動開關組成，利用微動開關判斷推車是否進入或移出停靠站，再利用繼電器控制電磁鎖以開鎖或解鎖平板推車。藉由掃描停靠站上的 QR 碼進入 RFID 刷卡畫面，是讓使用者藉由卡片註冊資訊、驗證狀態，將使用者資訊傳送至雲端資料庫進行比對後，判斷是否能租借車輛。

2.平板電腦推車：推車上有 QR 碼，掃描連線網頁後，將車輛資訊傳送到雲端資料庫，將使用者、停靠站進行比對分析後，讓停靠站解鎖或上鎖車輛。

3.操作介面：使用者可用手機、平板或電腦進行掃描進入登入畫面。在操作介面上得知資料庫訊息。

4.Firebase 雲端資料庫：將使用者、停靠站、平板推車狀態紀錄在資料庫中，並利用網頁程式碼進行資料比對分析。另外，可將車輛借還紀錄狀態登錄在雲端資料庫中，以節省紙筆登記的浪費，提高搜尋效率。

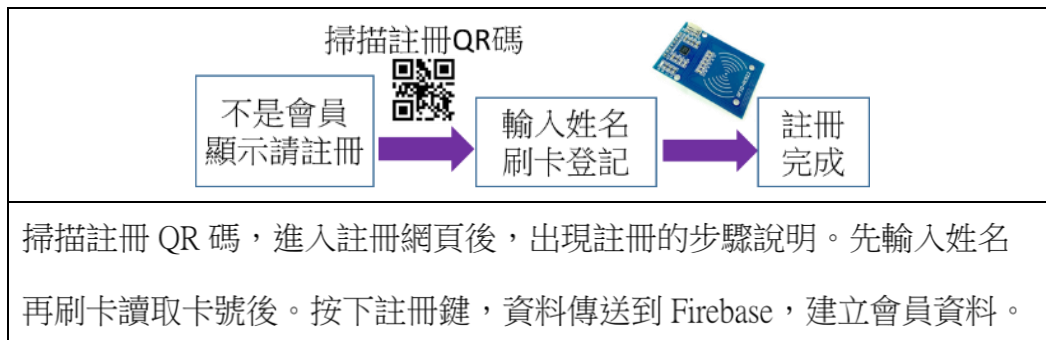
(三) 平板電腦推車借還平台運作流程圖



五、執行製作

(一) 註冊網頁

1. 流程說明：



2. 問題解決過程：

問題 1：如何輸入姓名欄位？

製作方式：利用 HTML 網頁製作出欄位，並讀取欄位的文字。

輸入完成後請按刷卡註冊

卡片號碼

姓名：

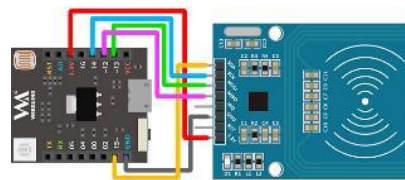
```
<p>姓名：<input id="book" type="text"></p>
```

註冊

請先輸入姓名再刷卡片
按下註冊按鈕就可完成註冊。

問題 2：如何讀取卡片序號？

製作方式：將 Webduino Smart 開發板與 RFID 感測器連接後，利用 Blockly 撰寫程式碼讀取卡號。



RFID 接線圖



Blockly 圖形化程式

```
rfid.read();  
rfid.on('enter', function(_uid){  
  rfid._uid = _uid; //讀取卡片數值  
  R = rfid._uid; //設定變數R為卡片數值  
  document.getElementById('show2').innerHTML = R; //顯示R數值  
  rfid.stopRead(); //RFID停止偵測
```

JavaScript 語法

問題 3：按下註冊鍵後，判斷是否輸入姓名與取得卡號？

製作方式：利用 if-else 邏輯判斷，按下註冊鍵後，先判斷姓名空格是否有被填入資料，如果沒有則顯示未填入姓名；在接著判斷是否有刷卡卡號，若是空白則顯示尚未刷卡。若是都不是空白時，則顯示完成註冊，將資料上傳雲端。

尚未輸入姓名

卡片號碼

姓名：

請先輸入姓名再刷卡片
按下註冊按鈕就可完成註冊。

```
document.getElementById('btn1').addEventListener('click',function () {  
  A_NAME= document.getElementById('book'); //讀取空格內資料  
  if (A_NAME.value === "") { //若是值為空白
```

```
document.getElementById('show1').innerHTML = '尚未輸入姓名'; //顯示尚未輸入姓名
```

尚未刷卡

卡片號碼

姓名：

請先輸入姓名再刷卡片
按下註冊按鈕就可完成註冊。

```
if (R === '') { //若是卡號值為空白  
  document.getElementById('show1').innerHTML = '尚未刷卡'; //顯示尚未刷卡
```

完成註冊

1EA2A459

RFID 卡號

姓名：

請先輸入姓名再刷卡片
按下註冊按鈕就可完成註冊。

完成註冊，離開網頁後，就可從停靠站上的 QR 碼進入租借網頁進行租借平板電腦推車的流程。

問題 4：如何將資料上傳至 Firebase？

製作方式：若是利用 blocky 積木程式上傳，Firebase 會以亂碼方式寫入資料，日後就不易讀取資訊。所以要修改 JavaScript 程式碼，將指令 push(新增)改為 update(更新)，並利用 child()製作 RFID 代碼的根目錄。



Blockly 圖形化程式

myFirebase = new Firebase('https://'); //設定Firebase存放位置
 myFirebase.push({ //添加節點位置
 姓名: A_NAME,
 編號: R
 });

會以亂碼產生節點位置，以防止資料重複但之後不易找尋資料，因此要改變寫法。

M9pln-IV0M4A0J0PLvV

姓名: "美國隊長"
 編號: "1EA2A45"

Firestore資料庫中呈現內容

修正後寫法

myFirebase = new Firebase('https://'); //設定Firebase存放位置
 myFirebase.child(R).update({ //以卡號(R)為節點名稱，更新內容
 姓名: A_NAME,
 編號: R
 });

以卡號為節點名稱，以便之後搜尋資料使用

1EA2A459

姓名: "美國隊長"
 編號: "1EA2A45"

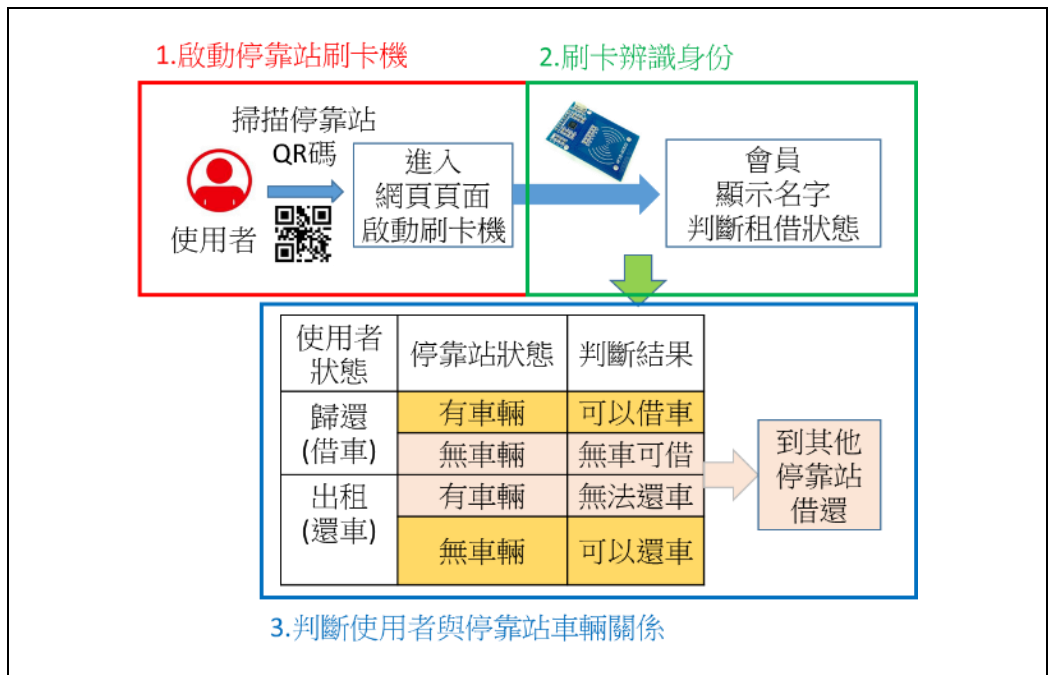
Firestore資料庫中呈現內容

3. Firebase 資料庫內容

項目	圖示	文字說明
人員		以卡號為節點，以方便搜尋資訊。 可判斷停靠站位置、租借狀態與租借車輛編號。
停靠站		停靠站有租借者編號、停靠車輛以及開發板編號，以方便啟動停靠站的開發板。
車輛		車輛若在停靠站內，則有停靠站資訊，若被借走，則有租借者資訊。
紀錄表		車輛紀錄表，以日期為第一節點，時間為第二節點，借用者與租借者為最後資料，以方便日後查詢。

(二) 停靠站模型與網頁

1. 流程說明：



步驟 1：進入停靠站網頁，從 Firebase 資料庫停靠站節點中尋找連接 RFID 感測器開發板的 ID，啟動刷卡機準備讀取卡號。

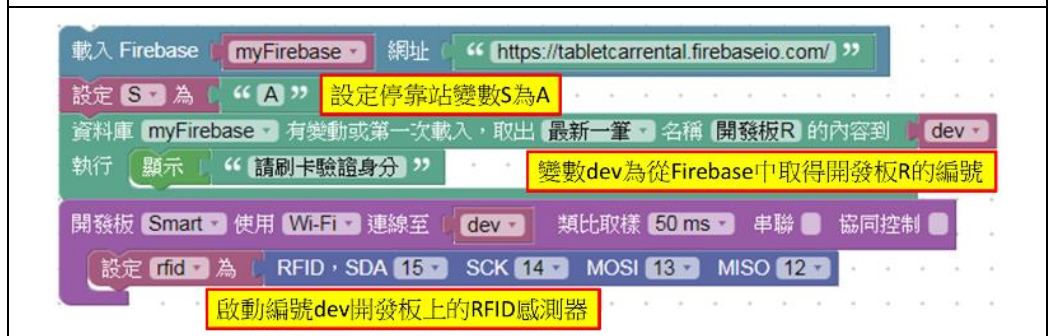
步驟 2：讀取卡號後，從所有會員編號中去比對資料是否符合，若非會員出現請註冊，若是會員則讀取租借狀態進行判斷。

步驟 3：從 Firebase 資料庫取得此停靠站內是否有車輛，並與使用者狀態進行判讀。

2. 問題解決過程

問題 1：如何啟動停靠站上的 RFID 感測器進行刷卡？

製作方式：每個停靠站給予固定的站名(如：A、B...)，先自資料庫中，搜尋站名的節點下的(開發板 R)的編號名稱，帶入開發板編號內再啟動。



修正 JavaScript 程式碼

```
myFirebase.limitToLast(1).on('child_added', function (snapshot) {  
  dev = snapshot.val().開發板R;
```

只讀取最新一筆更新的開發板R的值



修改

```
myFirebase.child(S).on('value', function (snapshot) {  
  dev = snapshot.val().開發板R;
```

讀取變數S下開發板R的值

依照不同停靠站站名的網頁，變更變數 S 值即可啟動該停靠站的刷卡機

問題 2：如何判斷是否為會員？

製作方式：先從資料庫讀取所有的編號值，再依序比對，每比對完一次後，再累加次數，持續比對。若有編號與讀取卡號相符時，即可進行後續動作。若無相符，則判別為非會員，顯示請註冊字樣。

變數num為從Firebase中取得所有人員的卡片編號內容
設定變數n為1

如果n的值比編號筆數(會員人數)值小

判斷第n筆的編號是否為RFID讀取到的卡號(R)

不是就讓n+1後，等0.2秒重複此流程VER2再判讀

都沒有編號相符，就是尚未註冊，顯示請註冊

問題 3：如何判斷使用者能否使用此停靠站借還車輛？

製作方式：先讀取使用者的狀態，再讀取停靠站狀態進行比對分析結果，若是使用者可以租借或歸還，則讓使用者資料內更新停靠站站名，停靠站內更新使用者編號。若是不能使用此停靠站，則顯示請到其他停靠站字樣。

讀取使用者名稱與租借狀態

```
myFirebase.child(R).on('value', function (snapshot) { //讀取編號R的使用者資訊
  na = snapshot.val().姓名; //變數na為使用者姓名
  st = snapshot.val().租借狀態; //變數st為使用者租借狀態
```

判斷停靠站與使用者關係

```
if (car == '0000') { //如果停靠站沒有車輛
  if (st == '歸還') { //使用者租借狀態為歸還(可借車)
    document.getElementById('show1').innerHTML = '停靠站無車輛請至其他停靠站租借'; //無車可借
  } else if (st == '出租') { //使用者租借狀態為出租(要還車)
    document.getElementById('show1').innerHTML = (String(na) + String('請掃描車輛QR碼進行歸還'));
    //顯示姓名，提醒刷車輛QR碼進入網頁進行歸還
    VER4();
  }
} else { //如果停靠站有車輛
  if (st == '歸還') { //使用者租借狀態為歸還(可借車)
    document.getElementById('show1').innerHTML = (String(na) + String('請掃描車輛QR碼進行租借'));
    //顯示姓名，提醒刷車輛QR碼進入網頁進行租借
    VER4();
  } else if (st == '出租') { //使用者租借狀態為出租(要還車)
    document.getElementById('show1').innerHTML = '停靠站有車輛請至其他停靠站歸還'; //有車不能還
  }
}
```

更新停靠站與使用者資訊

```
function VER4() {
  myFirebase = new Firebase('https://tabletcarrental.firebaseio.com/');
  myFirebase.child(S).update({
    租借者: R
  }); //在資料庫中停靠站S的節點下，更新使租借者編號為R
  myFirebase.child(R).update({
    停靠站: S
  }); //在資料庫中使用者R的節點下，更新停靠站為S
}
```

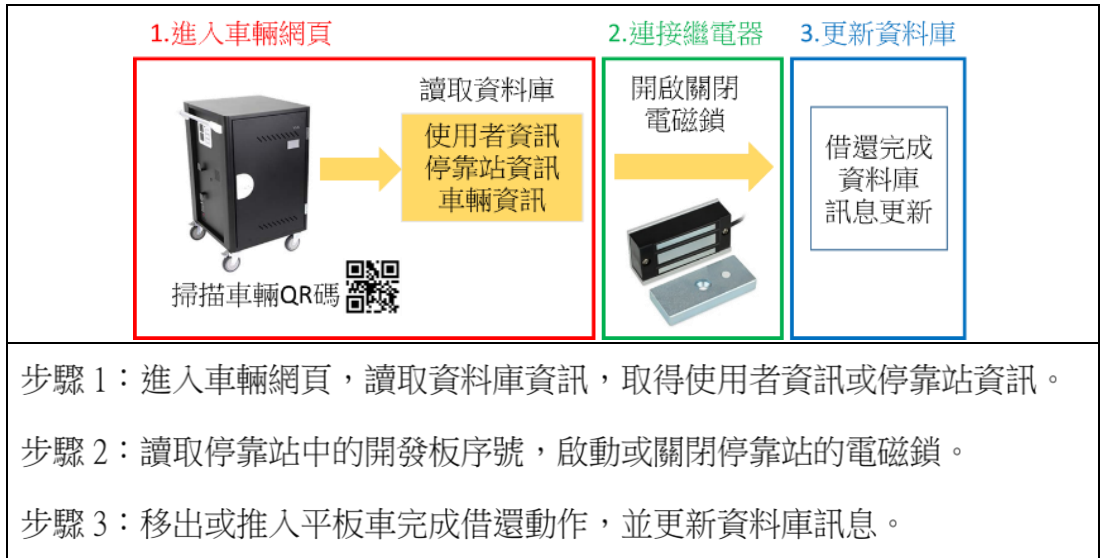
問題 4：如何製作停靠站模型？

製作方式：利用白木條、密集板搭建停靠站模型，並將停靠站上所需的開發板(Arduino、Smart)、感測元件(繼電器、RFID)、電磁鎖、微動開關進行組裝。



(三) 車輛模型與網頁

1. 流程說明：



2. 問題解決流程：

問題 1：如何製作平板電腦推車模型？

製作方式：利用密集板，依照比例大小，切割方形與圓形零件進行組裝接合，將鐵塊放置前端，模擬上鎖解鎖情境。

問題 1：進入車輛網頁，如何得知停靠站與使用者資訊？

製作方式：

使用者狀態	車輛資訊
歸還(要借車)	有停靠站資訊
出租(要還車)	有租借者資訊

→ 1. 由停靠站名內讀取租借者卡號
 2. 判斷停靠站車號與此車輛編號是否相同

→ 1. 由車輛資料內讀取租借者卡號
 2. 判斷租借者車號與此車輛編號是否相同

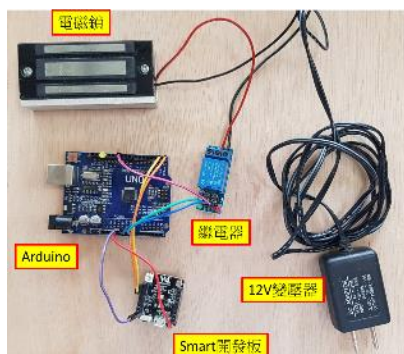
使用者狀態出租(要還車)

```
myFirebase = new Firebase('https://tablecarrental.firebaseio.com/');
CAR = '0001'; //車輛編號為0001
myFirebase.child(CAR).on('value', function (snapshot) {
  S = snapshot.val().停靠站; //由車輛編號搜尋停靠站
  if (S == "0") { //若是車輛不在停靠站內，代表是被租借中
    myFirebase.child(CAR).on('value', function (snapshot) {
      RC = snapshot.val().租借者; //從車輛編號取得租借者編號(被借走所以有租借者)
      myFirebase.child(RC).on('value', function (snapshot) {
        S1 = snapshot.val().停靠站; //從租借者編號得知停靠站的位置
        na = snapshot.val().姓名; //從租借者編號得知姓名
        if (S1 == "0") { //如果沒有停靠站位置，代表使用者沒有先從停靠站登入
          document.getElementById('show1').innerHTML = '請先掃描停靠站QR碼登入會員';
        } else {
          myFirebase.child(S1).on('value', function (snapshot) {
            RS = snapshot.val().租借者; //從停靠站得知租借者編號
            dev2 = snapshot.val().開發板; //從停靠站得知控制繼電器的開發板編號
            if (RC == RS) { //若是車輛租借者跟停靠站租借者是同一個，則可以還車
              E();
            });
          } else { //編號不同，則可能是誤刷已經被其他人租借的車輛上的QR碼
            document.getElementById('show1').innerHTML = '人員與車輛不符合';
          }
        }
      }
    }
  }
});
```

使用者狀態歸還(要借車)

```
else { //若是車輛在停靠站內
  myFirebase.child(S).on('value', function (snapshot) {
    RS = snapshot.val().租借者; //可從停靠站得知使用者編號
    if (RS == "0000") { //如果沒有租借者編號，代表使用者沒有先在停靠站登入
      document.getElementById('show1').innerHTML = '請先掃描停靠站QR碼登入會員';
    } else {
      myFirebase.child(S).on('value', function (snapshot) {
        dev2 = snapshot.val().開發板; //可從停靠站讀取開發板序號
        myFirebase.child(RS).on('value', function (snapshot) {
          S2 = snapshot.val().停靠站; //從租借者編號取得停靠站名
          na = snapshot.val().姓名; //從租借者編號取得姓名
          if (S == S2) { //車輛停靠站與租借者登入的停靠站相同
            F(); //進入出租流程
          });
        } else {
          document.getElementById('show1').innerHTML = '停靠站與車輛不符合';
          //若是不相同，代表刷錯停靠站內的車輛
        }
      }
    }
  }
});
```

問題 2：如何開啟與關閉繼電器，控制電磁鎖的開鎖與解鎖？



製作方式：繼電器能持續運作，不因網頁關閉而斷線，使用 Arduino 與繼電器連接，再將 Smart 開發板的 TX、RX 與 Arduino RX、Tx 腳位對接，可將訊息傳送給 Arduino 執行。


```

sketch_mar13a $
int relayPin = 10;

void setup()
{
  Serial.begin(9600);
  pinMode( 13 , OUTPUT);
  pinMode(relayPin, OUTPUT);
}

void loop()
{
  if( ((Serial.read())=='1'))
  {
    digitalWrite(relayPin, HIGH);
    digitalWrite( 13 , HIGH );
  }

  if( ((Serial.read())=='2'))
  {
    digitalWrite(relayPin, LOW);
    digitalWrite( 13 , LOW );
  }
}

```

設定繼電器腳位為10

假如Arduino讀取到訊號為"1"則繼電器開啟，LED亮

假如Arduino讀取到訊號為"2"則繼電器關閉，LED暗

Arduino 程式碼，繼電器腳位 10，若是接收訊息為" 1" 則啟動繼電器，讓電磁鎖通電鎖上，並讓 LED 亮起。若是接收訊息為" 2" ，則關閉繼電器，讓電磁鎖斷電解鎖，並讓 LED 關閉。

問題 3：如何判斷是否有車輛移出或推入？

製作方式：

使用者狀態	微動開關	
歸還(要借車)	放開	借還失敗，資料庫不更新
	未放開	
出租(要還車)	未按下	1. 歸還成功 2. 更新歸還紀錄表、租借狀態 3. 租借者清空，車輛與停靠站綁定
	按下	

1. 租借成功
2. 更新出租紀錄表、租借狀態
3. 停靠站清空，車輛與租借者綁定

歸還情況

設定 T 為 10，變數 b 為 1

執行：uart 傳送資料 "1"，顯示 "請推入平板車"，(非同步) 等待 1 秒之後，執行：設定 T 為 T-1，每隔 1 秒減 1，持續直到 T=0

如果 T=0 且 b=1：執行：重複 3 次，uart 傳送資料 "2"，顯示 "歸還失敗"，傳送訊息 "2"，繼電器關閉，電磁鎖解鎖，歸還失敗

當 button2 進行 放開 時：設定 b 為 0，設定 T 為 0，顯示 "歸還成功"，歸還成功

微動開關被按下，變數 T=0, b=0，避免回到上方程式區

租借情況

設定 T 為 10，變數 b 為 1

執行：uart 傳送資料 "2"，顯示 "請移出平板車"，(非同步) 等待 1 秒之後，執行：設定 T 為 T-1，每隔 1 秒減 1，持續直到 T=0

如果 T=0 且 b=1：執行：重複 3 次，uart 傳送資料 "1"，顯示 "租借失敗"，傳送訊息 "1"，繼電器開啟，電磁鎖上鎖，租借失敗

當 button 進行 放開 時：設定 b 為 0，設定 T 為 0，顯示 "租借成功"，租借成功

微動開關被放開，變數 T=0, b=0，避免回到上方程式區

問題 4：如何更新資料庫訊息？

製作方式：

利用 child()設定節點與 update(更新)指令，更新節點下方的資訊。

- 1.租借成功
- 2.更新出租紀錄表、租借狀態
- 3.停靠站清空，車輛與租借者綁定

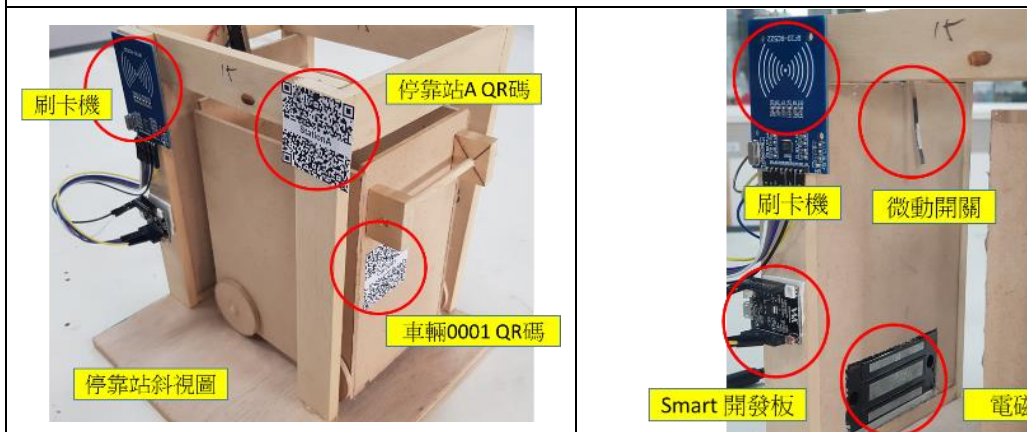
- 1.歸還成功
- 2.更新歸還紀錄表、租借狀態
- 3.租借者清空，車輛與停靠站綁定

```
myFirebase.child(RC).update({ //更新租借者資訊  
  車輛: '0000', //已歸還車輛  
  停靠站: '0', //已完成此次租借  
  租借狀態: '歸還' //狀態改為歸還(可借車)  
});
```

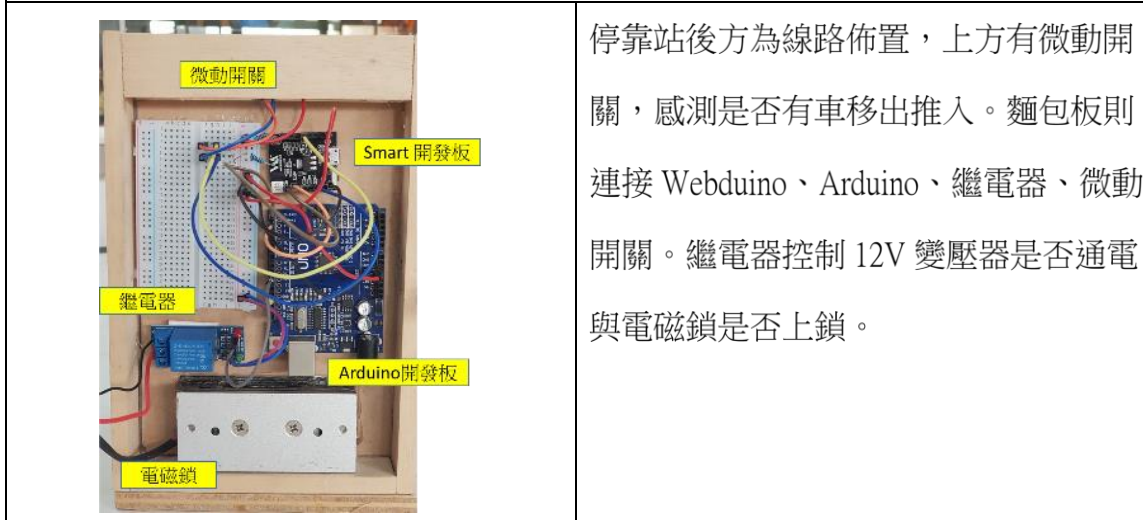
伍、研究結果

一、作品外觀


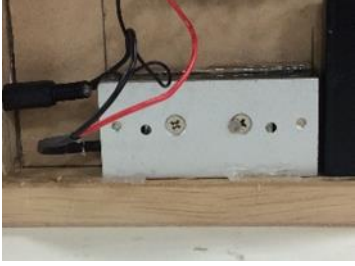
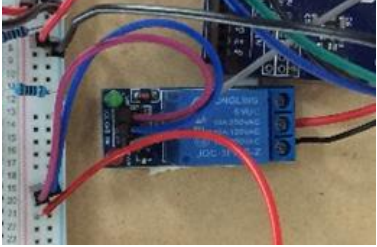
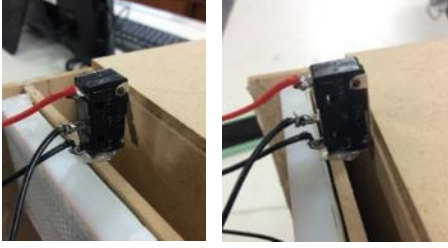
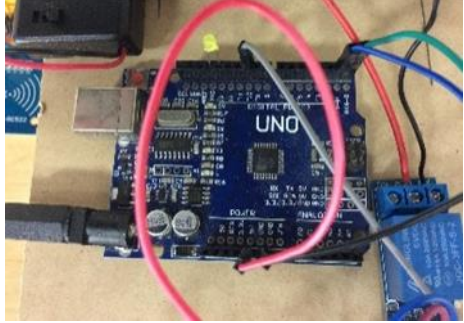
停靠站與車輛斜視圖



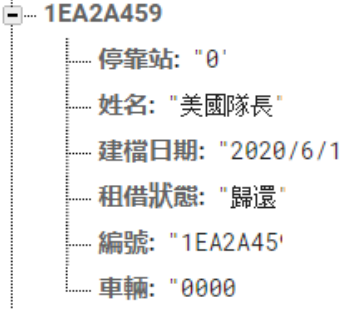


停靠站後方電路系統



二、作品特點

項目	圖片	文字
RFID 感測器		<p>通過無線射頻訊號識別卡號，是租借系統中很重要的感應元件，用來讀取卡號。</p>
電磁鎖		<p>通電後會產生強大的吸力吸住鐵板，一般情況下無法拔開，用來上鎖平板車。電磁鎖黏在停靠站下方，鐵塊黏在車輛上。</p>
繼電器		<p>控制電磁鎖的開關。系統傳訊息給 Arduino，Arduino 再傳給繼電器，繼電器再控制電磁鎖。</p>
微動 開關		<p>有 COM(正電)、NO(2)、NC(16)三個腳位，未壓下時腳位 16 通電，壓下時腳位 2 通電。用來判斷平板車是否有推入或移出。</p>
開發板		<p>系統用了兩種開發板，一種是 Arduino，一個種是 Webduino，Arduino 因為可以一直運作，所以連接了電磁鎖和繼電器，Webduino 連接了微動開關和 RFID</p>

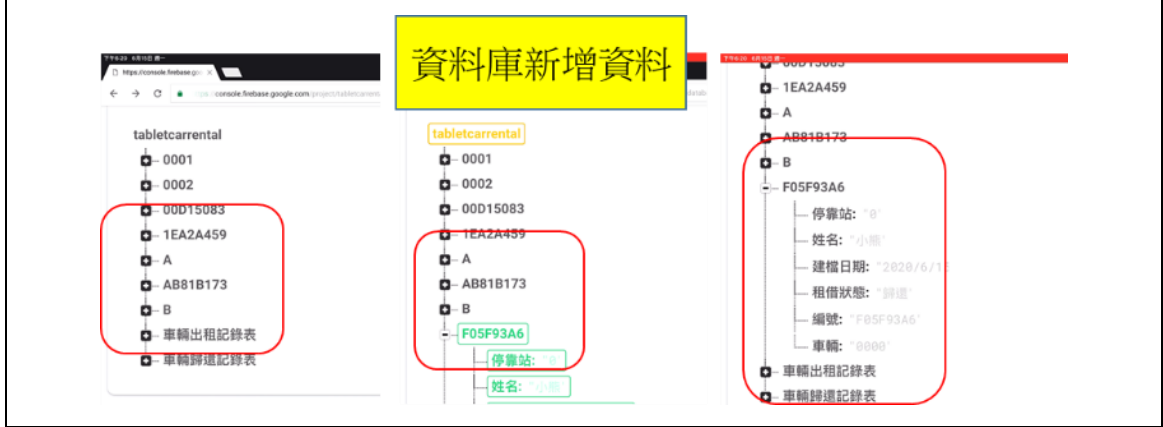
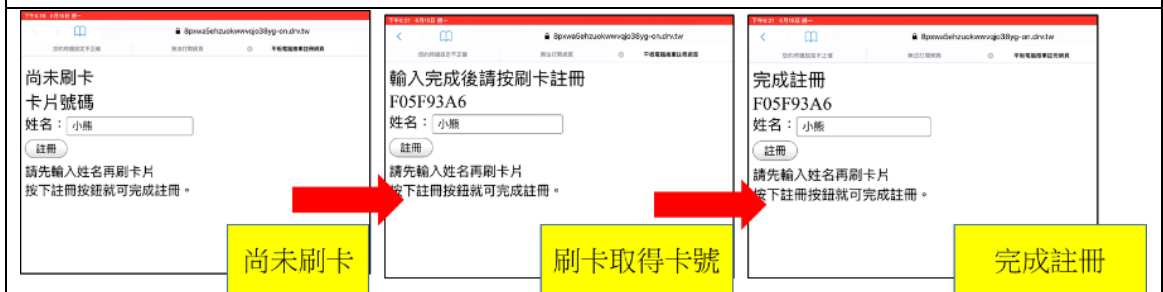
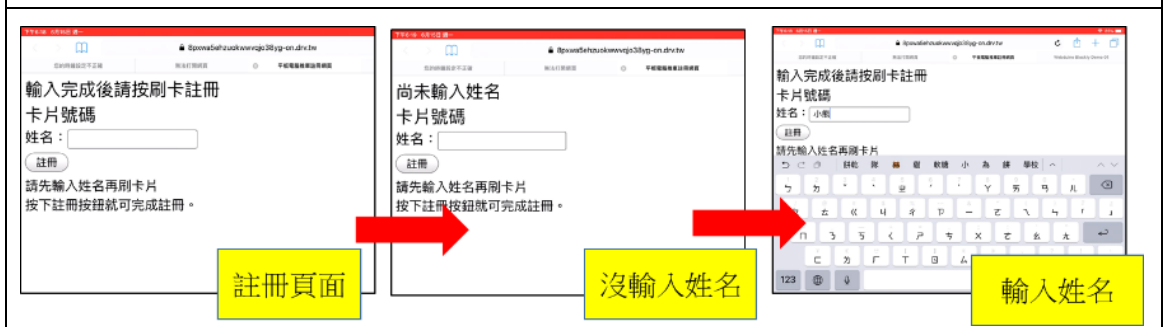
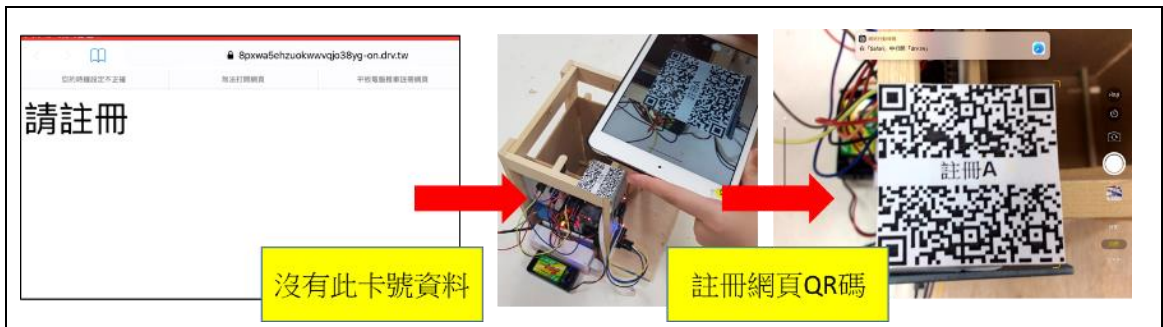
Webduino		<p>是製作系統中最重要的一個，系統的程式碼除了控制電磁鎖開關，其他的都是用 Webduino 寫的，Webduino 可以把積木塊轉成程式碼，非常方便</p>
QR 碼		<p>將網頁製作成 Qrcode 形式，讓使用者可用自己的行動載具掃描後進入頁面，就不用設置介面在停靠站旁邊。</p>
Firebase 資料庫		<p>用來紀錄使用者資料，可用 Webduino 控制，資料用樹狀圖呈現，很容易看懂。</p>

三、實際操作

依照所設計的流程步驟，進行實際操作。利用平板電腦的螢幕畫面錄製功能，錄製擷取網頁及 Firebase 資料庫呈現的畫面。

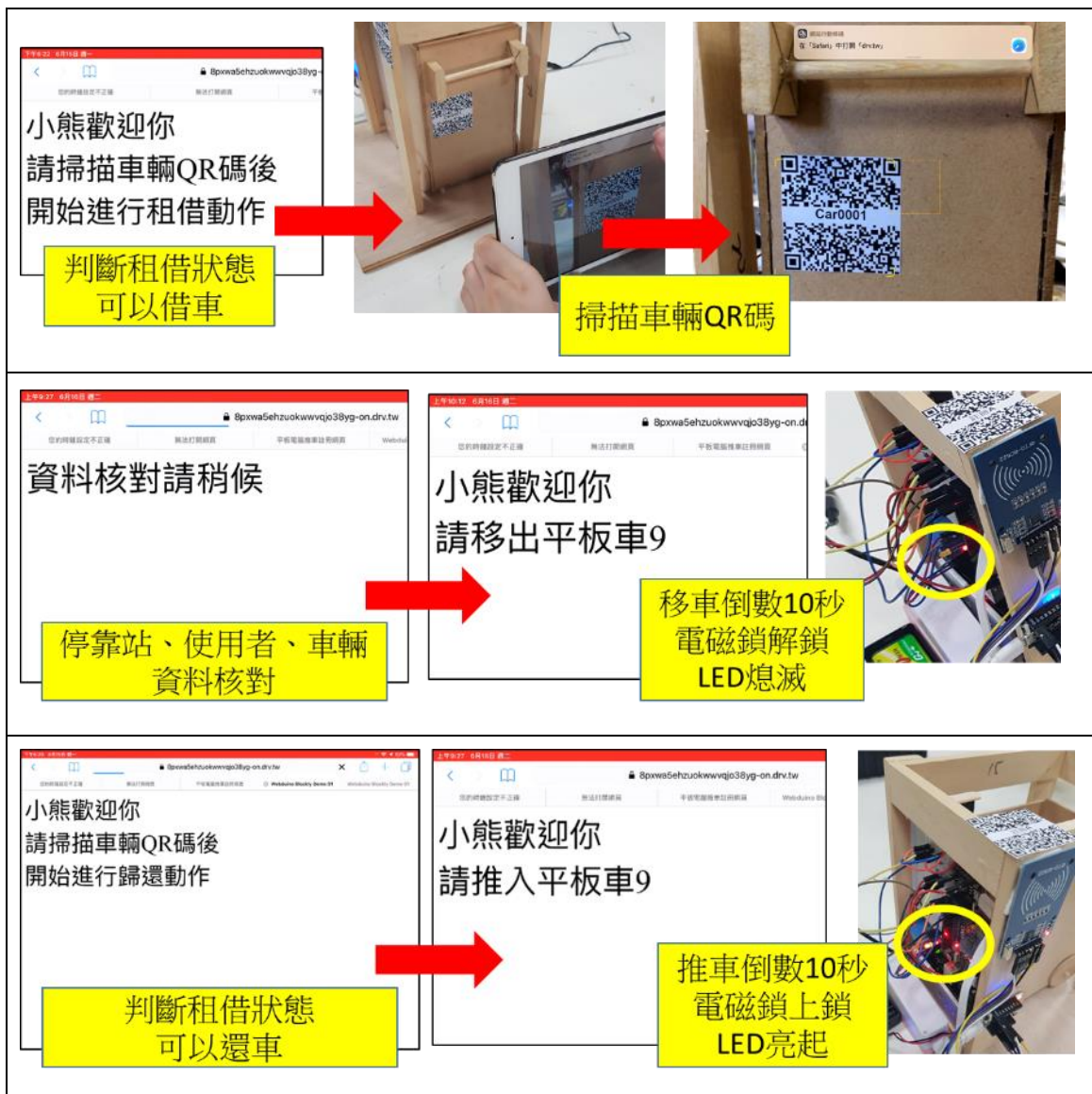
(一) 註冊





(二) 租借與歸還





陸、討論

根據研究目標進行檢視是否完成此次研究內容。

一、製作可藉由 RFID 註冊人員的系統平台

(一) 用卡註冊基本資料及登入

已完成，第一次使用系統時，要先到註冊頁刷卡及填寫基本資料，系統將其登入資料庫後才可使用租借系統，如要登入租借系統只須刷卡，等系統判定是否有註冊後即可使用。

(二) 知道卡片是否註冊過

已完成，刷卡後，系統去資料庫抓取所有使用者的卡片編號並製程列表，若讀取到的卡號等於列表中的其中一項，就表示此卡號已註冊，可以使用借還系

統，若不等於任何一項，就表示未註冊，無法使用借還系統。

(三) 管理者可得知使用者姓名、登入及借還的日期時間、租借狀態

已完成，註冊時會將使用者填的姓名、職業以及卡片編號寫入 firebase，而每次租借或歸還會更新借車時間、歸還時間及狀態，這些資料都會變成樹狀圖存在網路上，只有知道網址的管理員可以隨時查看、修改資料。

二、用簡單的零件達成完整的效果

我們只使用了 Webduino 和 Arduino 兩種開發板以及一些基本的原件，就能像市面上完整的平台一樣得知並且控制這些零件。系統可以透過 Wi-Fi 連接到 Webduino 開發板，開發板會將個原件的狀況傳送回系統，而連接在 Arduino 開發板上的電磁鎖及 LED 燈則是用 Arduino 和 Webduino 開發板的 TX、RX 兩個腳位互接，將鎖及燈泡的狀態透過 Webduino 傳給系統。

三、將資料儲存到 Firebase 資料庫裡以方便讀取與查閱

在程式碼內加入 update、once 來更新及讀取 Firebase 裡的資料，並在兩個指令之前加入 child 來指定更新及讀取的項目。程式碼 myFirebase.child(R).update，意思是將資料的第一層寫入 RFID 代碼(變數 R)，myfirebase.child(rfid._uid).on，意思則是指定要讀取第一層 RFID 卡號底下的項目。

四、製作可藉由 RFID 租借平板車的系統平台

已完成，一開始註冊完成後就會寫入租借狀態為歸還，表示使用者目前並未借走平板車，登入系統後，系統會抓取該使用者的狀態以判斷可租借還是可歸還，完成借還後會將租借狀態改成出租或歸還，並更新租車時間或歸還時間。

五、製作出平板車租借停靠站模型

得知車是否放置至正確位置、是否移出；已完成，原本是用磁簧開關的 pin 腳位偵測是否歸位，但只要使用到讀取 pin 腳，程式就會當掉，後來我們想到用微動開關來偵測，但是微動開關無法得知最開始的狀態，只有改變狀態才能得知，雖然不知道此停靠站是否有車，但是將車移出或推入(按下或放開開關)時，系統可以偵測的到，所以最後我用微動開關來偵測車是否有推到正確的位子(按下開關)；偵測是

否有移出平板車(放開開關)。

六、其他

在製作時遇到了一些問題，但礙於時間和成本，無法達成，例如：無法得知是否在同一個停靠站歸還，若兩層樓各有一個停靠站，一樓借的車推到二樓還，則二樓借的就需要到一樓還，這樣需要多花費更多人力，因為時間的關係，目前還無法做出、想出解決方案。此外，如果有人強行拉出平板車，系統並不會知道，又或者有人刷卡後沒有確實歸還，而是拿鐵塊去吸並壓下開關，系統也不知道，不只這些問題，還有很多地方需要修改，這些都使未來要改良、解決的。

柒、結論

此次研究製作已建構平板車租借系統，利用 Webduino 結合 RFID 感測器執行刷卡註冊與登入租借、藉由 Arduino 控制繼電器與電磁鎖開關、微動開關判別是否推入與移出，以及資料建構在雲端資料庫方便存取與減少紙本登記的資源浪費，能更即時的掌握每臺平板車的租借狀況，而且系統因為在 Firebase 內新增了停靠站這個新的集合已經能透過資料庫來得知停靠站是否有車可借或有地可還來補足微動開關無法得知系統開始運作時是被壓下還是放開的缺陷，並且可以甲地租乙地還，還能通過掃描 QRcode 來進入網頁，可直接在自己的手機或平板上操作。此次在過程中，遇到最多挑戰的是在構想程式碼與修改程式碼的部分，雖然 Webduino 的程式積木可以提供基礎程式編寫，但若需要其他功能時，則需利用 JavaScript 進行修改，雖然複雜但也有趣，失敗時會想要繼續嘗試並找尋資料答案，最後當程式可以運作時的成就感，也是這次製作過程中的最大收穫。此系統其實還有很大的發展空間，像：因為可以從 Firebase 裡知道哪個停靠站有車哪個沒車，所以能依此作出一個車站預覽圖，圖中會顯示各個停靠站是否有車，這樣使用者就可以知道現在該去哪個站借車、去哪個站還車，並且還能延伸出預約系統，讓使用者可以在遠端先預約完再到停靠站進行租借，保證自己能借到車，不會白跑一趟，除了這些以外系統還有很多可以增加的功能，但由於時間與成本關係，本次製作只能以模型呈現，無法實際架設停靠站，但軟體功能性已經初步實際應用，期望在未來能將裝置製作得更加完善、方便。

捌、參考資料

- 一、Webduino 學習手冊 <https://tutorials.webduino.io/zh-tw/docs/basic/index.html>
- 二、Firebase 教學 <https://www.oxxostudio.tw/>
- 三、Arduino 筆記 <http://coopermaa2nd.blogspot.com/2011/03/lab21-12v.html>
- 四、網頁設計教學-HTML、CSS、Javascript(js)語法入門學習筆記
https://www.gameislearning.url.tw/article_content.php?getb=7&foog=9998
- 五、康軒版 七下資訊科技
- 六、康軒版 七上生活科技

【評語】 032803

作品實做成一個推車之借還車系統，整體概念類似 YouBike 之借、還車數位化系統，整合 RFID、資料庫、電磁鎖等設備，使應用情境具有實用性。在競爭替代技術方面仍可多加探討，比較作品於應用情境有實用性、在競爭替代技術方面仍可多加探討比較。整體而言作品相當完整具實用性，惟創意較為不足。

摘要

看到校園內的平板車租借，需要到資訊組用紙本登記，而因為學校電梯只有一台，等電梯耽誤了很多時間，而且紙本消耗了很多資源，因此想做一個可以利用悠遊卡註冊、借車、歸還的系統，將停靠站放置各個樓層，有需要時可隨時去借還。此次成品製作是利用Webduino、Arduino開發板，借由網路連動Firebase資料庫，逐步完成租借系統。經過了多次修正與測試，已完成可註冊、租借的系統平台，但因時間和成本有限，目前只能以模型測試，期待日後能製作出停靠站並實踐於在平板車的租借。

研究動機

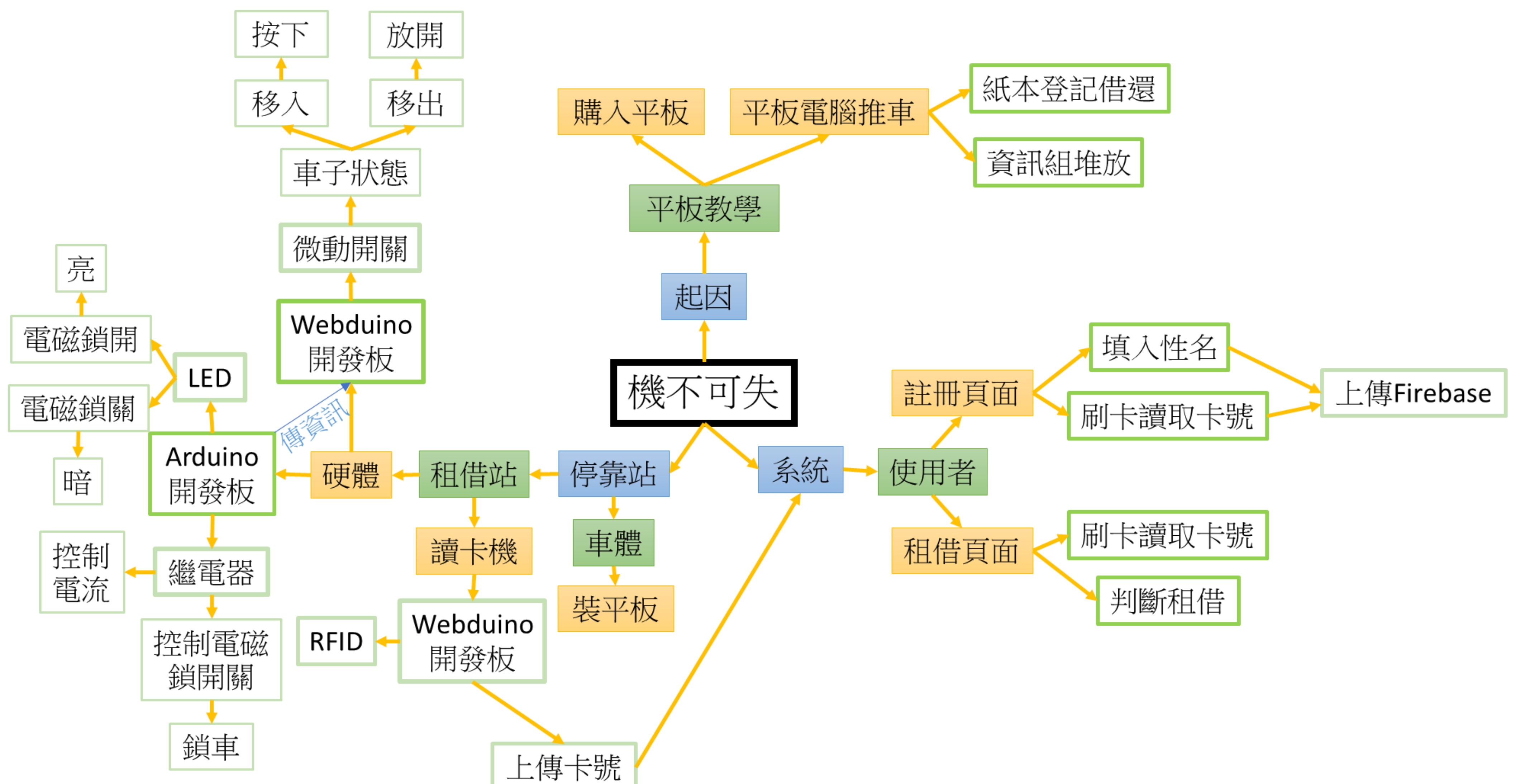


先到資訊組利用紙本登記後，再將車推至各班教學。教學完畢後，再推回資訊組歸還，並在紙本登記。
造成紙本浪費且不易尋找租借紀錄

將Webduino、Arduino開發板、RFID、繼電器製作停靠站，利用物聯網與Firebase連結。使用者利用刷卡方式註冊、登入資訊。租借狀態與時間可直接記錄在雲端資料庫中，
方便查詢且節省紙張。

研究目的

- 一. 用卡註冊基本資料及登入
- 二. 知道卡片是否註冊過
- 三. 租借或歸還平板車
- 四. 得知車是否放置至正確位置、是否移出
- 五. 管理者可得知使用者姓名職業、登入及借還的日期時間、租借狀態



研究過程

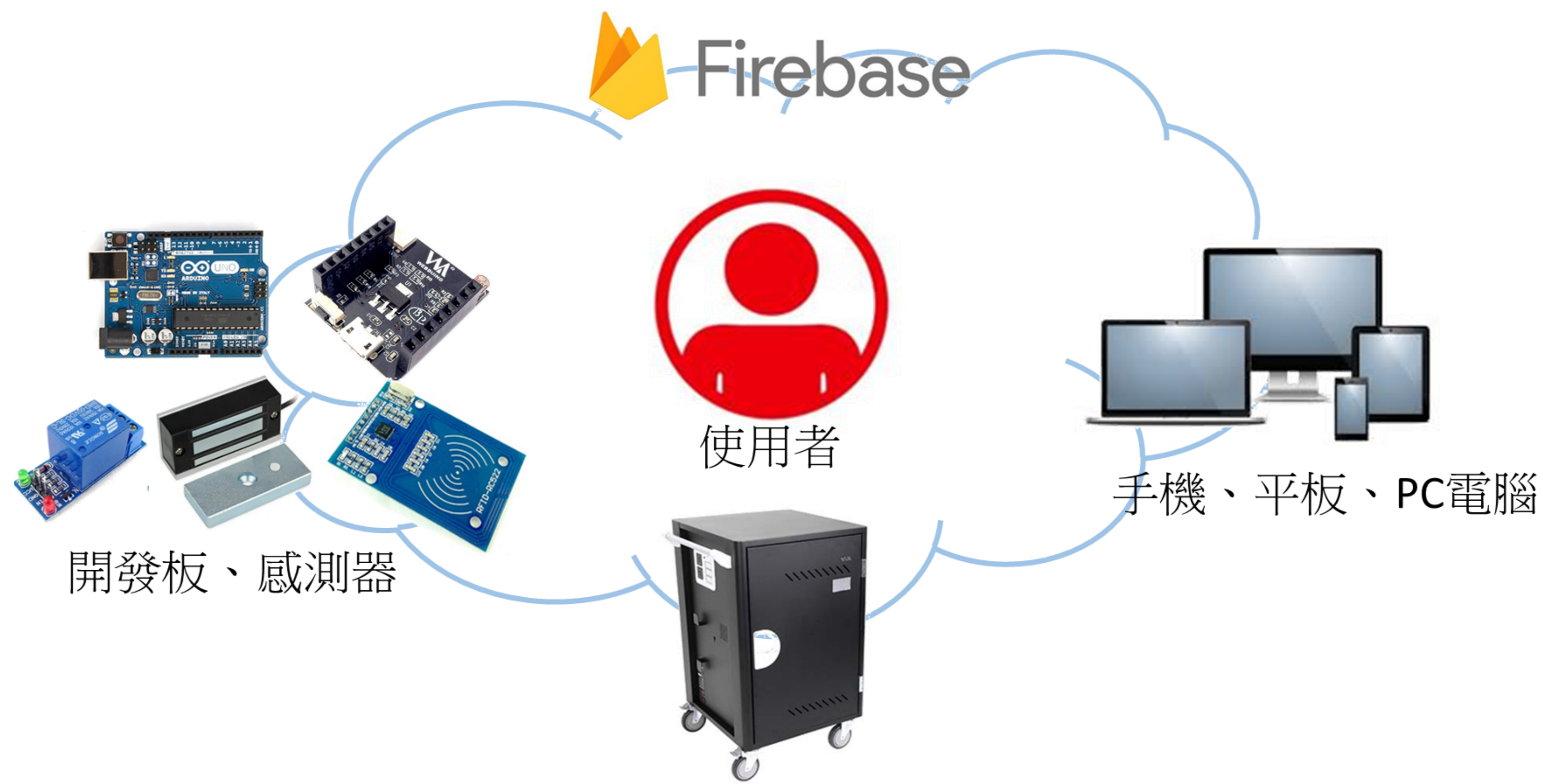
概念圖

由使用者角度開始發想，需要將停靠站、操作介面、雲端資料庫、行動平板推車彼此連接在一起，利用物聯網的方式，進行設計。

雲端資料庫：將使用者、停靠站、平板推車狀態紀錄在資料庫，進行資料比對分析。

停靠站：

Arduino、RFID、Webduino開發板、繼電器、電磁鎖、微動開關組成。

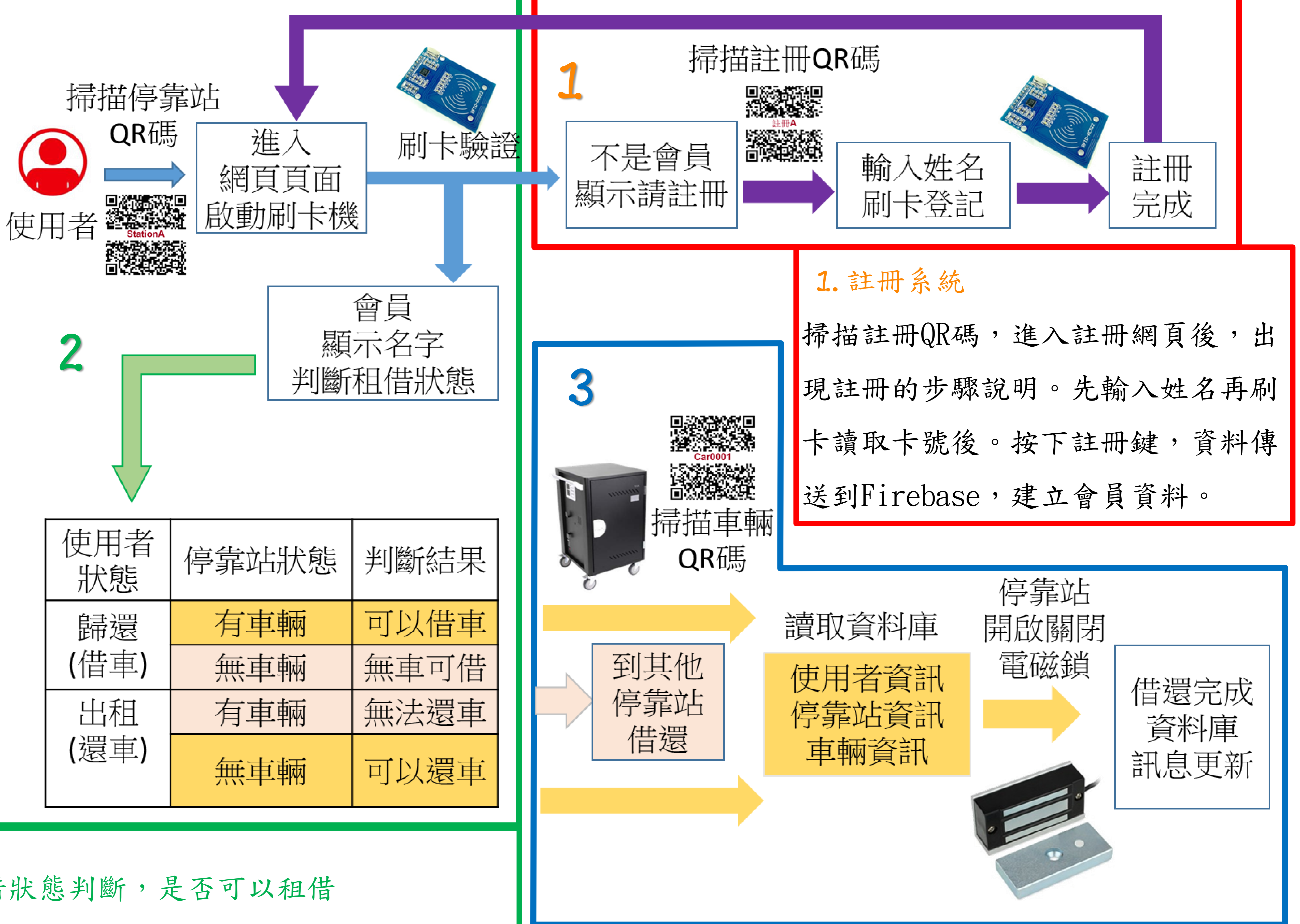


操作介面：

使用者可用手機、平板或電腦進行掃描進入網頁畫面。

平板電腦推車：推車上有QR碼，掃描連線網頁後，將車輛資訊傳送到雲端資料庫。

運作流程圖



使用者狀態	停靠站狀態	判斷結果
歸還 (借車)	有車輛	可以借車
	無車輛	無車可借
出租 (還車)	有車輛	無法還車
	無車輛	可以還車

2. 租借狀態判斷，是否可以租借

步驟1：進入停靠站網頁，從Firebase資料庫停靠站節點中尋找連接RFID感測器開發板的ID，啟動刷卡機準備讀取卡號。

步驟2：讀取卡號後，從所有會員編號中去比對資料是否符合，非會員出現請註冊，若是會員則讀取租借狀態進行判斷。

步驟3：從Firebase資料庫取得此停靠站內是否有車輛，並與使用者狀態進行判讀。

3. 車輛資料判斷，控制停靠站電磁鎖

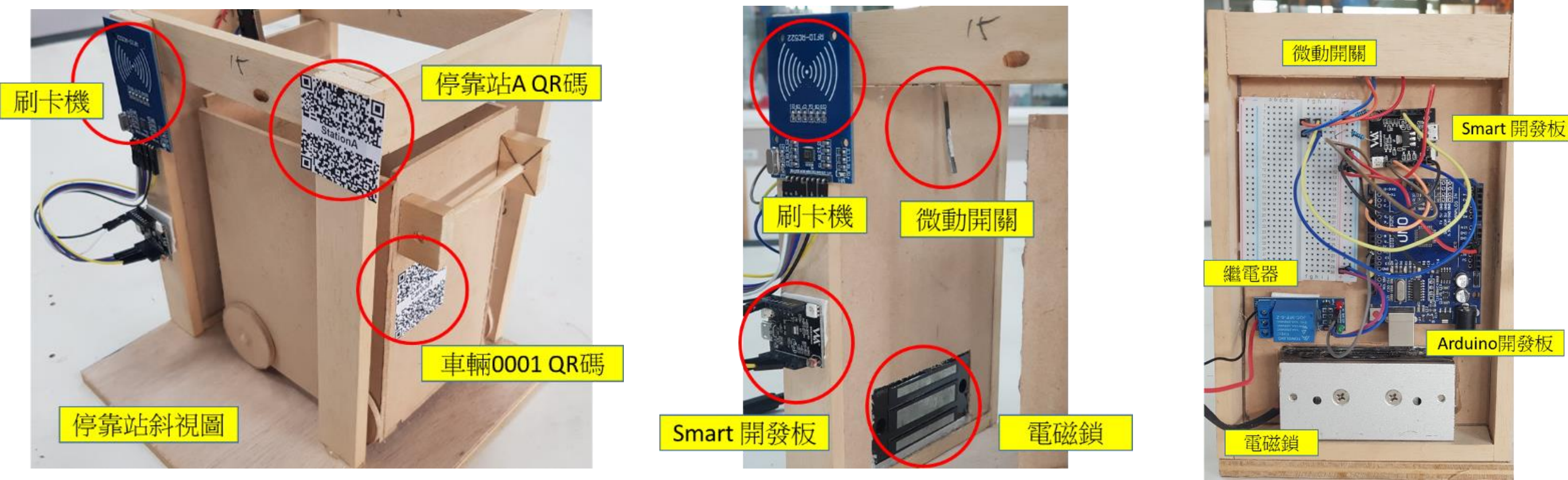
步驟1：進入車輛網頁，讀取資料庫資訊，取得使用者資訊或停靠站資訊。

步驟2：讀取停靠站中的開發板序號，啟動或關閉停靠站的電磁鎖。

步驟3：移出或推入平板車完成借還動作，並更新資料庫訊息。

研究結果

停靠站與車輛斜視圖



管理者監控頁面

A 停靠站狀態	B 停靠站狀態		
無車	0002		
車0001	車0002		
1EA2A459	可借		
編號	姓名	狀態	車輛
00D15083	奇異博士	歸還	
1EA2A459	美國隊長	出租	0001
AB81B173	蜘蛛人	歸還	
F05F93A6	小熊	歸還	

停靠站後方為線路佈置，上方有微動開關，感測是否有車移出推入。麵包板則連接Webduino、Arduino、繼電器、微動開關。繼電器控制12V變壓器是否通電與電磁鎖是否上鎖。

從Firebase中抓資料製成表格，只要有人借還，網頁馬上就會更新，方便管理者掌握即時資訊。

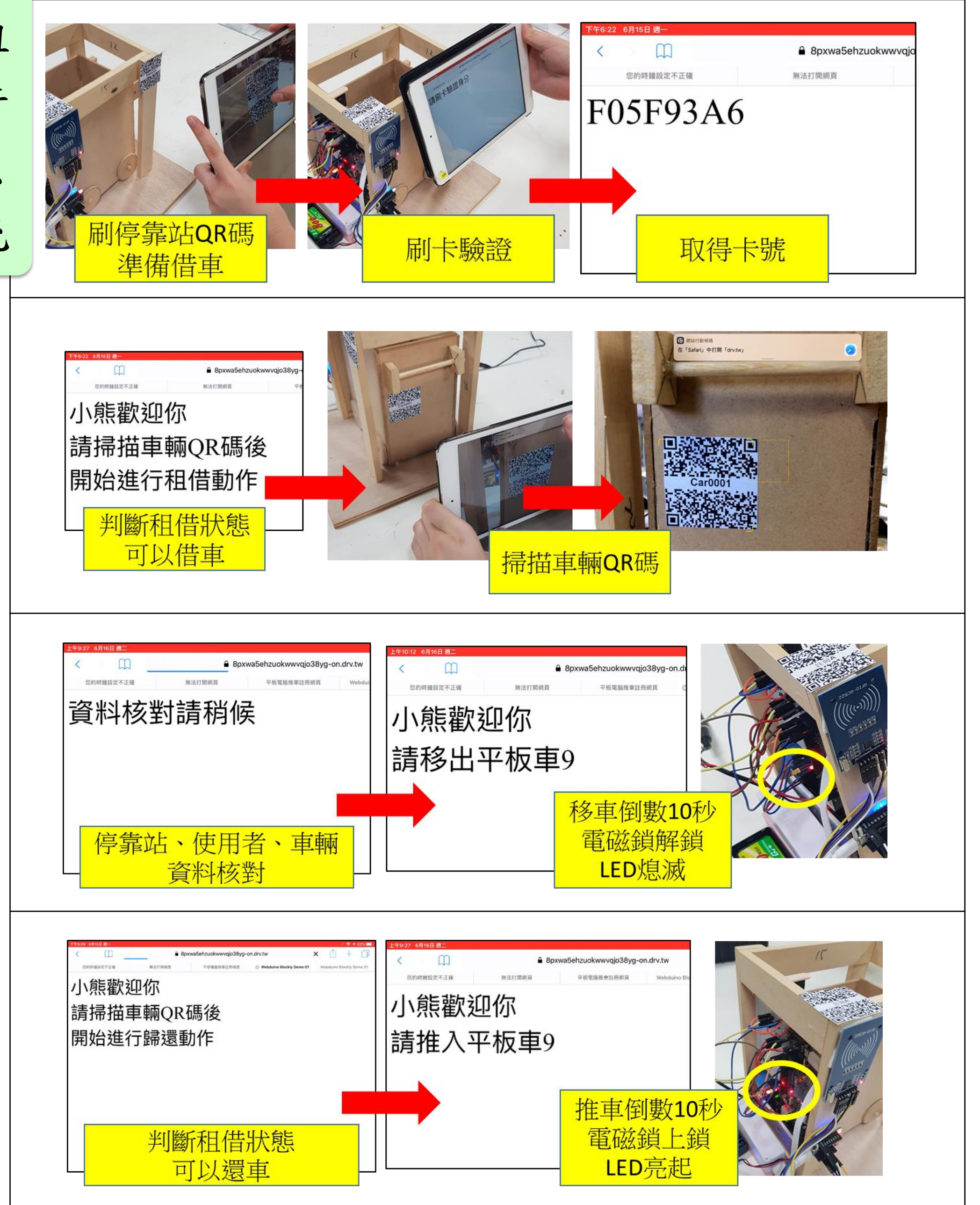
實際操作

依照所設計的流程步驟，進行實際操作。利用平板電腦的螢幕畫面錄製功能，錄製擷取網頁及Firebase資料庫呈現的畫面。

註冊系統



租借系統



結論

● 遇到的挑戰：

構想程式碼與修改程式碼，雖然Webduino的程式積木可以提供基礎程式編寫，但需要其他功能時，需利用JavaScript進行修改。

● 未來展望：

目前平台主要以使用者的角度進行設計，因此此系統在後端管理者方面還有很大的發展空間，例如：

1. 預約系統，可從Firebase中知道哪個停靠站有車哪個沒車，所以能依此作出一個車站預約系統，網頁中會顯示各個停靠站是否有車，讓使用者能事先預約車輛，保證自己能借到車，也方便管理者管理。
2. 主動通知，當使用者進行借還時，系統主動傳訊息通知管理者，讓管理者能即時掌握資訊。

除了這些以外系統還有很多可以增加的功能，但由於時間與成本關係，本次製作只能以模型呈現，無法實際架設停靠站，但軟體功能性已經初步實際應用，期望在未來能將裝置製作得更加完善、方便。

討論

製作目標	已完成內容
利用RFID進行註冊與租借動作	註冊系統：可用RFID卡片註冊基本資料及登入且可得知卡片是否註冊過。 租借系統：管理者可得知使用者姓名、登入及借還的日期時間、租借狀態
製作停靠站模型	使用Webduino和Arduino兩種開發板和一些基本的原件，就能像市面上完整的平台一樣運行，得知並且控制這些零件的狀態。 利用微動開關、電磁鎖偵測與控制，停靠站與車輛的開鎖與解鎖。
Firestore資料庫更新與讀取	程式碼內加入update、once來更新及讀取Firestore裡的資料，並在兩個指令之前增加child來指定更新及讀取的項目。