

中華民國第 56 屆中小學科學展覽會 作品說明書

高級中等學校組 電腦與資訊學科

佳作

052508

物聯網 (IoT) 教室投影機網路控制 App 程式

學校名稱：新北市立淡水高級商工職業學校

作者： 職三 謝浩哲 職三 楊濠維 職三 林羿瑄	指導老師： 羅漢全 林御臻
---	-----------------------------

關鍵詞：網路控制、物聯網、投影機

摘要

現在的教學模式漸漸走向數位化，老師在課堂上時常需要使用投影機輔助教學，使用投影機時，常常要使用遙控器來開關，但萬一沒有遙控器或是遙控器故障，會相當麻煩，所以我們開發了一個不需遙控器就能輕鬆開關投影機的程式——iProject。

現在很注重環保議題，使用塑膠外殼的遙控器，是間接傷害地球的行為，因為製造塑膠會排放二氧化碳，造成全球暖化加速。此程式具有環保、操作簡單及可攜性高等優點，只要輸入投影機的 IP 位址，就能控制想要開關的投影機，不需要遙控器與電池，除了達到節能減碳愛地球的功效，也不會有想要開啟投影機時，卻不知道遙控器放在哪裡的問題。

壹、研究動機

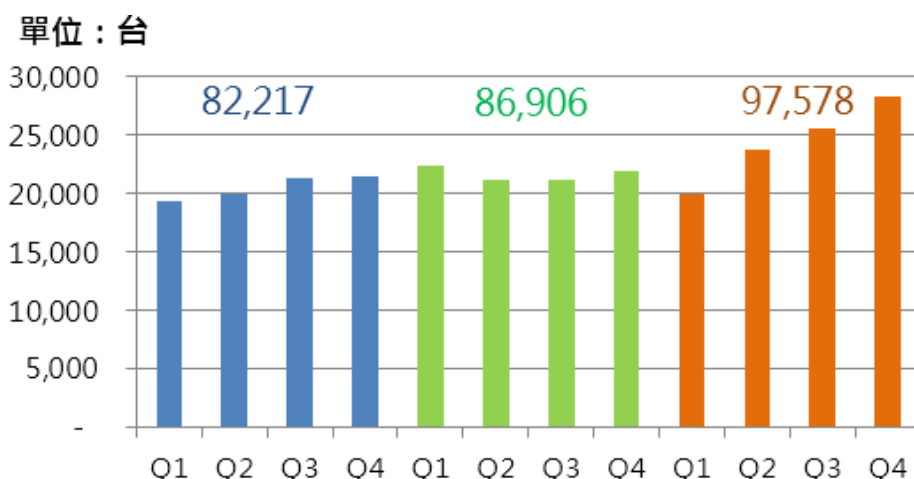
有很多課程需要透過投影機教學，各會議場所也經常使用投影機開會。現在投影機大多會搭配遙控器來使用，不需要像過去一樣，踩在桌椅上墊高來按機身上的電源鈕，以開關投影機，讓投影機的使用更加便利，遙控器既安全又方便，可說是必要配備。但現在環保意識抬頭，製造塑膠外殼的遙控器，不但增加二氧化碳的排放，造成全球暖化加速，不易分解的塑膠產物更增加環境的負擔，遙控器可說是環境殺手之一。

另外，使用遙控器操作投影機，有時會因為電池沒電，或找不到遙控器等狀況無法開啟投影機，耽誤大家的時間，甚至遇到遙控器故障或沒有遙控器的問題。這時靈機一動，想要在沒有遙控器的情況下，也能夠開關投影機，再加上計概課老師教電腦網路與應用時，多次提到物聯網的應用，讓我們非常感興趣，於是找出了原廠網路控制投影機程式，但操作介面不夠簡易、只有英文並且只有 Windows 作業系統能使用，於是我們希望製作一個操作容易、介面乾淨且在任何作業系統下皆能使用的網路控制投影機程式。我們向老師請教有關這方面的問題，最後藉由物聯網的概念，寫了一個能夠開關投影機的程式。

貳、研究目的

為了讓公司與教育單位在使用投影機時，不因為遙控器出問題而耽誤流程，所以我們設計出一項既環保、便利又簡易的程式——iProject，省去攜帶遙控器的麻煩。這幾年環保意識抬頭，當使用遙控器控制投影機時，需要使用電池，下表為 2008 年市場投影機銷貨前十大廠商，總銷售量達到六百萬台，如此龐大的銷貨總量，累積起來的電池量會很驚人。

2013-2015 年各季台灣投影機出貨量¹



市面上遙控器大多為塑膠外殼，而製造塑膠製品，難免會有二氧化碳排放的問題，若投影機需求量持續增加，一台投影機配置一支遙控器，這樣二氧化碳的排放會日益增加，加速全球暖化。若使用 iProject，就能減少遙控器的生產，降低二氧化碳的排放，還能夠減少電池的使用量，降低重金屬汙染。遙控器的功能雖然很多，但有很多功能很少使用，常造成使用者操作上的不便，原因不外乎就是介面過於複雜，而我們開發的 iProject 介面乾淨、操作容易且不需要額外購買遙控器與電池，不僅能達到環保效果，也適合所有年齡層的人使用。

評估自身技術與時間後，我們將 iProject 的開發分成三個階段，希望每個階段的 iProject 能夠為一般使用者、程式開發人員、設備及程式管理者帶來不同的貢獻。

角度	原廠程式的特色	iProject 1.0 / 2.0 的效益 ²	iProject 3.0 的期望
對使用者	要安裝.NET Framework，並且在英文介面中找到投影機圖示，在方框中輸入要開啟的投影機 IP，按下開啟投影機圖示即可執行。	只需要安裝 Google Chrome 瀏覽器，以及一個 Google 帳號，就能夠使用，不需要額外使用遙控器與電池。	只需要一個瀏覽器就能夠使用，不需要額外使用遙控器與電池。
對開發人員	此檔案僅支援 Windows 版本，若想使用其他作業系統來執行，則需另外開發。	只需要開發 Chrome App 版本，若要使用手機控制，則要轉換成 Android、iOS 能夠支援的版本，不需要各種作業系統都製作一種版本。	只需要開發網頁版本，只要瀏覽器能夠執行，完全不需要在意作業系統的需求。
對程式管理者	需要將各版本的檔案，上傳至雲端儲存空間，或燒錄成光碟保存。	只需要將程式發佈到 Chrome Web Store 即可。	只需要將網頁上傳到託管網站即可。

¹ 表格資料從 IDC（國際數據資訊）台灣投影機市場出貨量研究報告整理而來

² 目前已完成 iProject 2.0，iProject 3.0 開發中

參、研究設備及器材



電腦³



遙控器⁴



投影機-NEC NP500⁵



Android 智慧型手機⁶

³圖片取自 <http://open999.blogspot.tw/2013/08/acer-aspire-e1-571g-1562gi5-3230.html>

⁴圖片取自 http://tw.nec.com/zh_TW/products/NP500/

⁵圖片取自 http://tw.nec.com/zh_TW/products/NP500/

⁶圖片取自 http://www.sogi.com.tw/products/sony_xperia_c5_ultra/11710

肆、研究過程及方法

一、研究步驟

研究步驟大綱如下：

步驟一：研究 HTML5、CSS3 和 JavaScript。

步驟二：研究如何使用 Wireshark 監聽與解析封包資料。

步驟三：研究 Chrome App 的開發。

步驟四：程式介面設計與功能追加。

步驟五：軟體開發完成及維護。

二、基礎理論及使用技術

(一) TCP/IP 協定傳輸方式

TCP/IP 協定組中，負責傳送的協定分別是 TCP 與 UDP。TCP 提供連線導向的可靠性傳輸，當傳送資料給接收端時，若未收到接收端確認已收到的訊息，傳送端必須重新傳送一次資料；而 UDP 雖然傳輸速度較快，但只負責傳送資料，並不會確認接收端是否真正收到資料。

IP 位址就像網路上的身分證一樣，每個 IP 位址都是唯一的，而 IP 位址又再細分公有 IP (Public IP) 與私有 IP (Private IP)，差別在於公有 IP 能夠直接上網，私有 IP 能夠使用於內部網路，但無法直接上網。IP 共分為五個類別，每個類別的用途皆不同，下表為 IP 的五大類別與私有 IP 範圍的介紹：

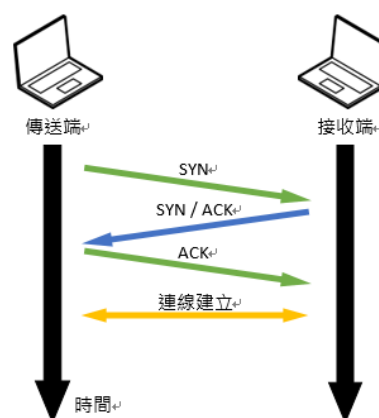
IP 類別	IP 第一個數值	私有 IP 範圍	用途
Class A	1 ~ 127	10.0.0.0 ~ 10.255.255.255	政府機構及軍事單位
Class B	128 ~ 191	172.16.0.0 ~ 172.31.255.255	大型企業及大專院校
Class C	192 ~ 223	192.168.0.0 ~ 192.168.255.255	一般家庭
Class D	224 ~ 239	-	保留作為特殊用途
Class E	240 ~ 255	-	

(二) 分封交換介紹

將要傳送的資料切割成數個封包，並在封包上標示發送及接收的 MAC 位址與 IP 位址，接著將封包個別傳送，到達目的地後，再將封包組回原來的資料型態，通訊時線路不會被獨占，是現在網路常用的形式。

(三) TCP 連線建立過程

TCP 封包內含了六種控制旗標，若要建立連線，則必須利用這六個控制旗標來進行三方交握的連線機制，由此可知，TCP 是連線導向的傳輸協定。右圖為三方交握示意圖，圖中 SYN 與 ACK 為其中兩個控制旗標，ACK 是用來回應上一個封包，只有第一個發出的封包沒有設定，而 SYN 是要求進行同步處理，也就是建立連線。傳送端發出第一個封包要求建立連線，接收端會回應此連線要求，同意建立連線，而傳送端則再一次做出回應，就成功建立連線了。



(四) HTML (超文件標示語言)

是一種網頁設計語言。HTML 屬於製作網站的基礎語言，通常須配合 CSS 與 JavaScript 來製作一個具有設計感的網頁及網頁應用程式。而 HTML 的形式為尖括弧所包圍的 HTML 元素，例如：`<html>`，而每個元素的表示方法又可再分兩種，一種是成對標籤，例如：`<title></title>`，另一種是獨立標籤，例如：`
`。HTML 是網頁製作的基石，能夠利用標題`<title>`、段落`<p>`、按鈕`<button>`等元素，製作出網頁基本框架與基本元素，甚至支援嵌入圖像與物件，讓網頁整體更加美觀。

(五) CSS (層疊式樣式表)

是一種用來增加網頁樣式的語言。在 HTML 中，若要使用 CSS 改變字型、排版、顏色等，有兩種使用方法，一種是寫在 HTML 尖括弧裡面，例如：

```
1 <div style="background-color: #FF0000;"></div>
```

上述舉例是將 div 區塊的背景顏色改為紅色（#FF0000 是十六進位碼，根據 RGB 規則，

顯示出來是純紅色)，另一種是獨立寫在 HTML 尖括弧外面，並用<style></style>來表示，例如：

```
1 <style>
2   div{ background-color: #FF0000; }
3 </style>
```

效果與上一個例子相同，差別在於前者是只變動一個 div 元素，後者是變動所有 div 元素。

(六) JavaScript

是一種用來製作網頁動態功能的語言。最早是使用在 HTML 網頁上，現在也廣泛應用於網頁伺服器，例如：Node.js。在 HTML 中，要使用 JavaScript 來製作動態效果，則需要使用<script></script>來表示，例如：

```
1 <script>
2   alert("Hello");
3 </script>
```

上述舉例是跳出警告視窗並在警告視窗中顯示 Hello。而我們這支程式所有的功能都是使用 JavaScript 來製作，可見 JavaScript 不只是用來製作網頁，還能夠製作網頁應用程式的功能。

(七) MongoDB 簡介



MongoDB 屬於 NoSQL 的資料庫，是以 C++ 撰寫而成，主要用來處理大數據等級的資料。以下是 MongoDB 儲存資料的格式：



```
{
  _id : 0001,
  Name : "iProjectUser",
  Age : 18,
  Password : "ilikeiProject",
  Email : [
    "iProject@xxx.com",
    "iProjectUser@xxx.com",
    "ilikeiProject@xxx.com"
  ]
}
```

資料中的_id 為該筆資料的 id，若無特別設定，系統會自動設置。資料的格式其實就是 JSON (JavaScript Object Notation) 格式，只要看得懂 JSON 格式，在解讀資料時不會有什麼困難。從上方資料可以看出，能夠記錄字串型態、數值型態的資料，更能夠儲存陣列型態的資料。

三、研究過程

(一) 研究 HTML、CSS 與 JavaScript

由於本作品是以 HTML、CSS 與 JavaScript 做為開發語言，故需要先學習其使用方法與語法。我們透過 Google 的 Blogger 來當作實驗媒介，並使用網路教材學習，最一開始必須了解如何利用 HTML 與 JavaScript 製作一個有功能的按鈕，如下表所示，按下「開啟」按鈕能夠顯示開啟，按下「關閉」按鈕能夠顯示關閉。

	
開啟按鈕的效果	關閉按鈕的效果

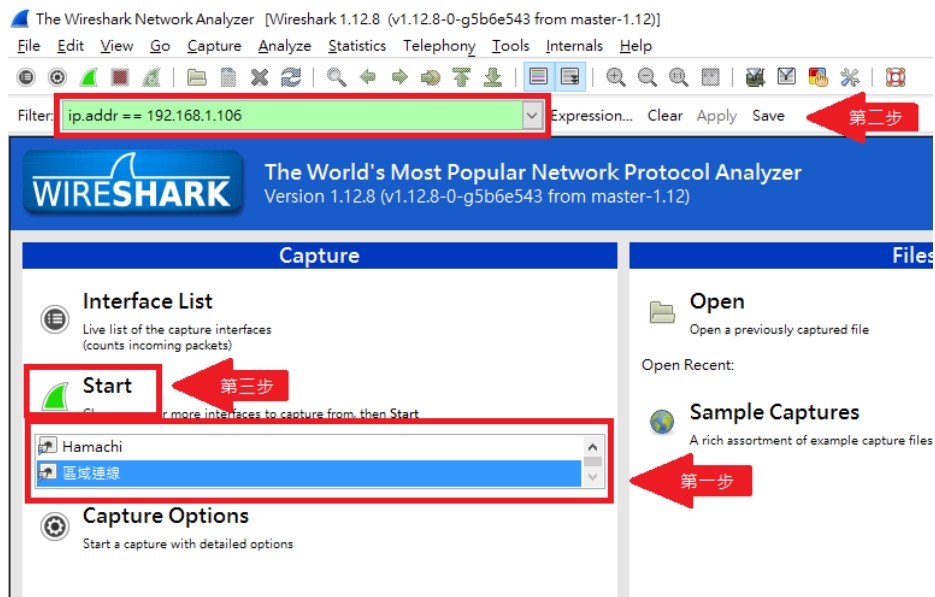
(二) 研究如何用 Wireshark 監聽與解析封包

了解這三種語言的使用方法與語法後，接著要研究如何使用 Wireshark 監聽封包。由於控制投影機需要確定有沒有完整建立連線，故須採用 TCP 傳輸協定，所以必須了解 TCP 協定下，封包的資料項位於何處，而 Wireshark 使用方法的詳細說明如下：

第一步：先選擇一個要監聽的網路區域。

第二步：輸入 `ip.addr ==` + 指定的 IP 位址（選擇性）。

第三步：按下按鈕 **Start**，開始監聽封包。



第四步：監聽活動開始後，選擇 TCP 傳輸協定的封包，可以從 **Source** 下方發現此封包的來源端 IP 位址（下圖實作的發送端 IP 位址為 203.117.158.174），亦可從 **Destination** 下方發現此封包的接收端 IP 位址（下圖實作的接收端 IP 位址為 192.168.1.106）。

第五步：在下方找到 **Data** 項，裡面就是此封包的資料，下圖實作的資料共有 9Bytes，資料為 05 00 00 00 c4 80 f5 1d 00，為什麼資料都是英文與數字呢？因為 Wireshark 監聽封包資料顯示為十六進位，故只會顯示英文及數字。

Capturing from 區域連線 [Wireshark 1.12.8 (v1.12.8-0-g5b6e543 from master-1.12)]

File Edit View Go Capture Analyze Statistics Telephony Tools Internals Help

Filter: ip.addr == 192.168.1.106 Expression... Clear Apply Save

No.	Time	Source	Destination	Protocol	Length	Info
1	0.00000000	192.168.1.106	64.233.189.188	TCP	55	62024-5228 [ACK] Seq=1 Ack=1 win=63259 Len=1
2	0.01028600	64.233.189.188	192.168.1.106	TCP	66	5228-62024 [ACK] Seq=1 Ack=2 win=47168 Len=0 SLE=1 SRE=2
3	0.09245500	192.168.1.106	203.117.158.174	TCP	59	61996-9100 [PSH, ACK] Seq=1 Ack=1 win=64854 Len=5
4	0.15292800	203.117.158.174	192.168.1.106	TCP	63	9100-61996 [PSH, ACK] Seq=1 Ack=6 win=64509 Len=9
5	0.20343600	192.168.1.106	203.117.158.174	TCP	54	61996-9100 [ACK] Seq=6 Ack=10 win=64845 Len=0
6	0.26392400	203.117.158.174	192.168.1.106	TCP	85	9100-61996 [PSH, ACK] Seq=10 Ack=6 win=64509 Len=31
7	0.31395800	192.168.1.106	203.117.158.174	TCP	54	61996-9100 [ACK] Seq=6 Ack=41 win=64814 Len=0
8	0.63151700	192.168.1.106	111.250.203.48	UDP	112	Source port: 56803 Destination port: 1045
9	0.63231400	111.250.203.48	192.168.1.106	UDP	112	Source port: 1045 Destination port: 56803
10	0.63246100	192.168.1.106	111.250.203.48	UDP	112	Source port: 56803 Destination port: 1045
11	0.66365400	111.250.203.48	192.168.1.106	UDP	112	Source port: 1045 Destination port: 56803
12	2.52456800	192.168.1.106	74.125.71.120	SSL	55	Continuation Data
13	2.82192200	74.125.71.120	192.168.1.106	TCP	66	443-62491 [ACK] Seq=1 Ack=2 win=64975 Len=0 SLE=1 SRE=2
15	4.56223400	173.194.72.189	192.168.1.106	TLSv1.2	106	Application Data
16	4.61246900	192.168.1.106	173.194.72.189	TCP	54	62499-443 [ACK] Seq=1 Ack=53 win=63062 Len=0
17	4.79013100	fe80::ffff:ffff:ffff02::2		ICMPv6	103	Router Solicitation
18	5.00625900	fe80::8000:f227:62cfe80::ffff:ffff:fff		ICMPv6	151	Router Advertisement
21	15.62476000	111.250.203.48	192.168.1.106	UDP	112	Source port: 1045 Destination port: 56803
22	15.62507000	192.168.1.106	111.250.203.48	UDP	112	Source port: 56803 Destination port: 1045
23	15.65621800	111.250.203.48	192.168.1.106	UDP	112	Source port: 1045 Destination port: 56803

第四步

Frame 4: 63 bytes on wire (504 bits), 63 bytes captured (504 bits) on interface 0

Ethernet II, Src: D-LinkIn_f7:e9:52 (90:94:e4:f7:e9:52), Dst: AsustekC_f2:c6:89 (54:04:a6:f2:c6:89)

Internet Protocol Version 4, Src: 203.117.158.174 (203.117.158.174), Dst: 192.168.1.106 (192.168.1.106)

Transmission Control Protocol, Src Port: 9100 (9100), Dst Port: 61996 (61996), Seq: 1, Ack: 6, Len: 9

Data (9 bytes)
 Data: 05000000c480f51d00
 [Length: 9]

```

0000 54 04 a6 f2 c6 89 90 94 e4 f7 e9 52 08 00 45 00  T.....R..E.
0010 00 31 47 f4 40 00 71 06 95 9c cb 75 9e ae c0 a8  .iG.@.q....u...
0020 01 6a 23 8c f2 2c 85 53 42 68 d9 e2 1e a2 50 18  .]#...S Bh....P.
0030 fb fd f2 f6 00 00 05 00 00 00 c4 80 f5 1d 00  .....
  
```

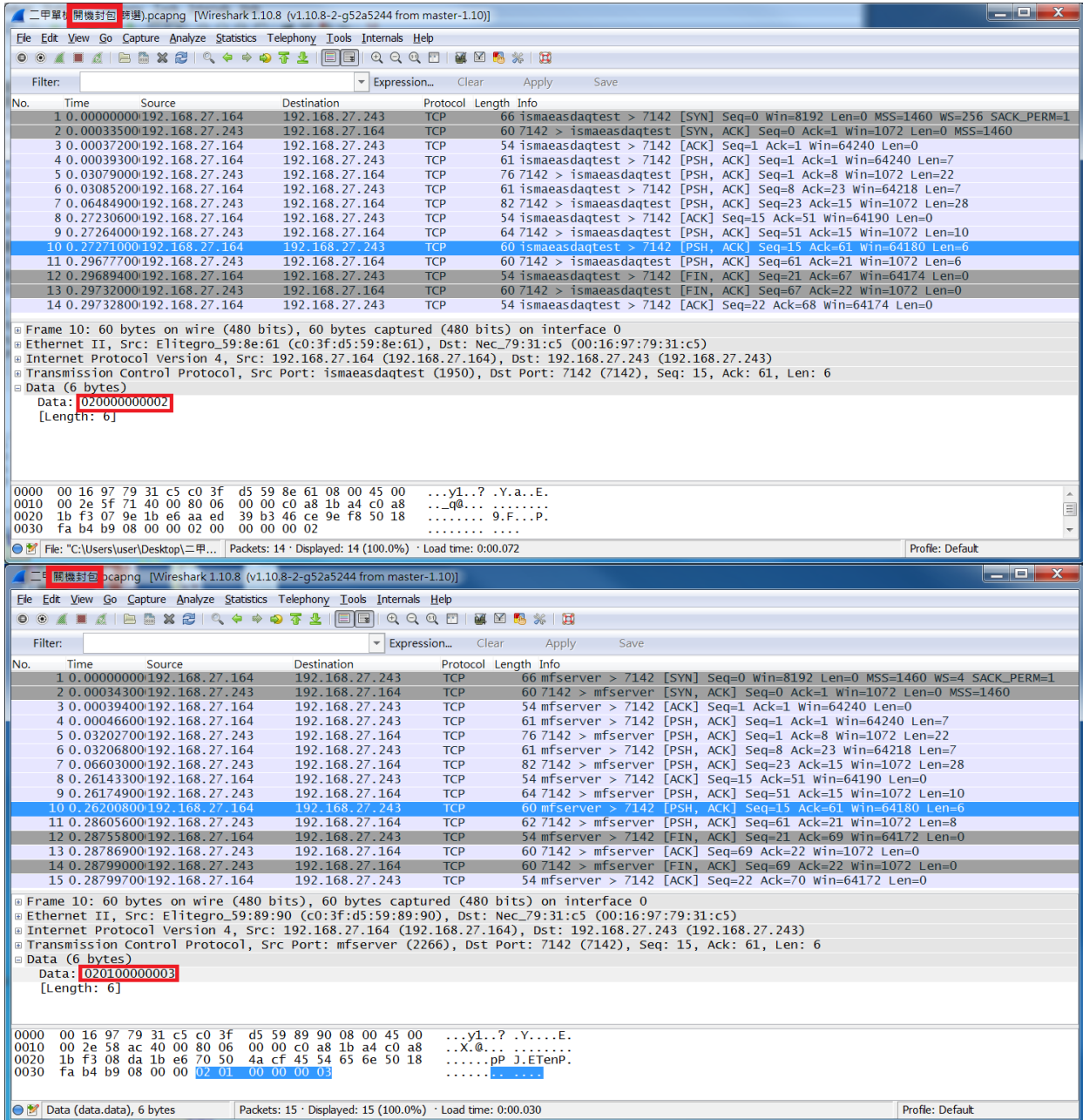
Data (data), 9 bytes | Packets: 23 - Displayed: 20 (87.0%)

第五步

知道如何使用 Wireshark 後，就來使用原廠程式控制投影機，並監聽其開機、關機及切換訊號源的控制碼，以下為監聽到的開機與關機的控制碼：

- 開機控制碼有 6Bytes，資料為 02 00 00 00 00 02
- 關機控制碼有 6Bytes，資料為 02 01 00 00 00 03

下圖為利用 Wireshark 監聽到開關機資料封包的實據：



(三) 研究 Chrome App 的開發

接著，要開始研究 Chrome App 的使用方式與程式格式，在網路上搜尋資源並了解其規定格式後，首先要做的就是製作一個 **manifest.json** 檔，讓瀏覽器讀取後，知道這個應用程式的名稱、版本、使用哪些特定功能等。以下為實際操作的 json 檔程式碼：

第1點：須設定 manifest 檔案版本為 **2**，為目前最新的版本，此屬性為必要屬性。

第2點：在此輸入程式的名稱，屬於必要屬性。

第3點：**"sockets"**為網路通訊用，屬於非必要屬性，但此程式需要使用，固有輸入此屬性，而 sockets 又有兩種傳輸協定，一個是 UDP，一個是 TCP，在此

保留 UDP 的原因為將來可以利用 UDP 傳輸協定來偵測區域網路內的投影機，可以讓使用者更方便於控制投影機，而使用 TCP 的原因不再多說明。

第4點：為此程式的版本數，為必要屬性。

第5點：**"permissions"**設定某些特殊權限用，在此使用了**"storage"**，是用來記錄使用者儲存的資料。

第6點：讓程式啟動時觸發背景作業 **background.js**，讓整體程式順利運行，屬於必要屬性。

第7點：設定程式的 icon，為非必要屬性。

```
1 {
2   "manifest_version": 2, 1
3   "name": "iProject", 2
4   "sockets": { 3
5     "udp": {
6       "send": "*"
7     },
8     "tcp": {
9       "connect": ""
10    }
11  },
12  "version": "2.0", 4
13  "permissions": [ 5
14    "storage"
15  ],
16  "app": {
17    "background": { 6
18      "scripts": ["background.js"]
19    }
20  },
21  "icons": { "128": "iPjicon.png" } 7
22 }
```

接著要製作一個背景作業，也就是上面提到的 **background.js** 檔，以下為背景作業的功能說明與程式碼：

當被觸發背景作業時，則創造一個 Chrome App 視窗，使用檔案為 **window.html**，並將其寬度設為 400px，高度設為 550px，並利用**"resizable"**來取消視窗最大化選項。

```
1 chrome.app.runtime.onLaunched.addListener(function() {
2   chrome.app.window.create('window.html', {
3     "bounds": {
4       "width": 400,
5       "height": 550,
6     },
7     "resizable": false,
8   });
9 });
10
```

有了基本的啟動方式後，就來研究 Chrome Socket API，並且實際操作，以下為最基本的 Chrome TCP Socket 傳訊方法與解說：

先宣告兩個變數為空值，一個作為存取 socketId 的值，另一個作為存取輸入的 IP 位址的值，當點擊 id 為 `openPr` 的按鈕時，則觸發 function `openProjector`，觸發後，會宣告一個變數為 7Bytes 的緩衝區 (ArrayBuffer) 與宣告一個 8 位元不帶符號的陣列 (Uint8Array)，接著要判斷 socketId 是否為空值 (主要是為了避免已經建立連線卻又發出連線要求的防呆措施)，如果是空值，則創建一個 TCP Socket，將創造後的創建資訊 (createInfo) 寫入變數 socketId 中，再來要將輸入的 IP 位址寫入變數 IP 中，接著要建立一個 TCP 連線，會根據 socketId 與 IP 透過 7142 通訊埠 (port) 建立連線，建立完成後，則根據 socketId 發送存在 buf 裡面的陣列資料，但若一開始 socketId 不是空值的話，代表先前已經建立連線了，不需要再創建 socket 與連線，直接傳輸資料即可。

```
1  var socketId = null;
2  var IP = null;
3
4  function openProjector(){
5
6      var buf = new ArrayBuffer(7);
7      var ints = new Uint8Array(buf);
8          ints[0]=0x00;
9          ints[1]=0xbf;
10         ints[2]=0x00;
11         ints[3]=0x00;
12         ints[4]=0x01;
13         ints[5]=0x00;
14         ints[6]=0xc0;
15
16         if (socketId == null) {
17             chrome.sockets.tcp.create({},function(createInfo) {
18                 socketId = createInfo.socketId;
19                 Ip = document.getElementById("IpAdd").value ;
20                 console.log(Ip.toString());
21                 chrome.sockets.tcp.connect(socketId,Ip.toString(),7142, function (result) {
22                     chrome.sockets.tcp.send(socketId,buf, function (abc){console.log(buf)});
23                 });
24             });
25         }
26         else {
27             chrome.sockets.tcp.send(socketId,buf, function (abc){console.log(buf)});
28         }
29     }
30
31     document.getElementById('openPr').addEventListener('click', openProjector);
```

(四) 介面設計與功能追加

當主要的功能完成以後，則要進行介面的設計，讓程式更有人性化與美感，而所有的按鈕、文字方塊、選項按鈕等，皆是以 HTML 與 CSS 製作。為了讓程式更加便利，我們額外追加了記錄功能 (使用 storage)，方便使用者儲存各投影機 IP 位址 (Name-Address Book)，以下為此程式的介面：



(五) 軟體開發完成

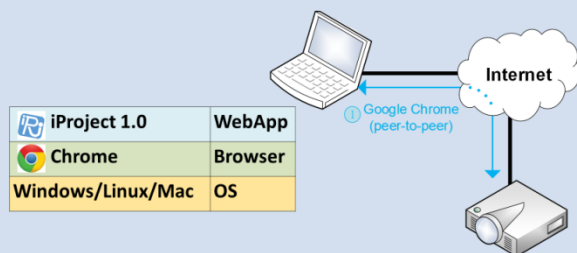
以下為各版本的 iProject 規格表：

iProject 各版本規格表			
版本	iProject 1.0	iProject 2.0	iProject 3.0
介面或圖示			
檔案容量	113KB (用戶端)	122KB (用戶端)	0KB (用戶端)
設備需求	電腦	電腦	任何能上網的設備

iProject 各版本規格表			
版本	iProject 1.0	iProject 2.0	iProject 3.0
軟體需求	Google Chrome	Google Chrome	支援的瀏覽器
帳號需求	無	Google 帳號	可無
網路需求	是	是	是
軟體屬性	Chrome App	Chrome App	Web 網頁
開發語言	HTML5 CSS3 JavaScript	HTML5 CSS3 JavaScript	HTML5 CSS3 JavaScript
封包協定	TCP	TCP	TCP
軟體功能	開關投影機及選擇訊號源	開關投影機及選擇訊號源 並且提供 IP-Name Address Book	開關投影機及選擇訊號源 提供 Device Name Address Book
多元發展	可運用在任何能連上網的家電做控制	可運用在任何能連上網的家電做控制	可運用在任何能連上網的家電做控制
開發狀況	開發完成	開發完成	持續開發 (最終理想： 物聯網入口網站)

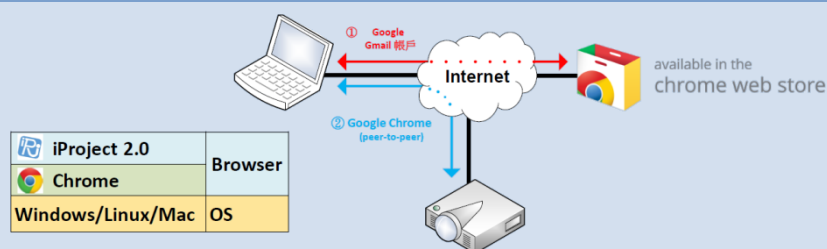
(六) 各版本架構圖

iProject 1.0 架構介紹



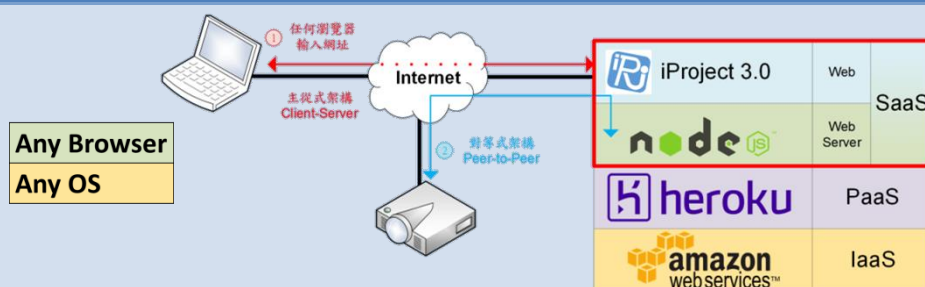
此版本傳輸方法為點對點傳輸，採用 Chrome 非封裝擴充功能的形式，使用者必須下載該檔案，再將檔案安裝至 Chrome 中。

iProject 2.0 架構介紹



此版本傳輸方法為點對點傳輸，採用 Chrome App 的形式，使用者須使用 Google 帳號，至 Chrome Web Store 將 iProject 2.0 加入到自己的 Google 帳號中，當使用者登入 Google Chrome 時，會自動將程式安裝至設備中。

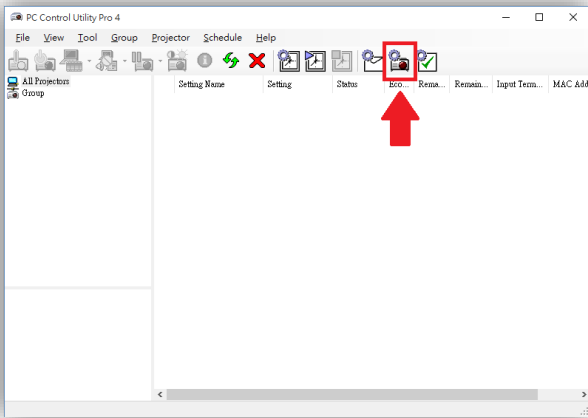

iProject 3.0 架構介紹

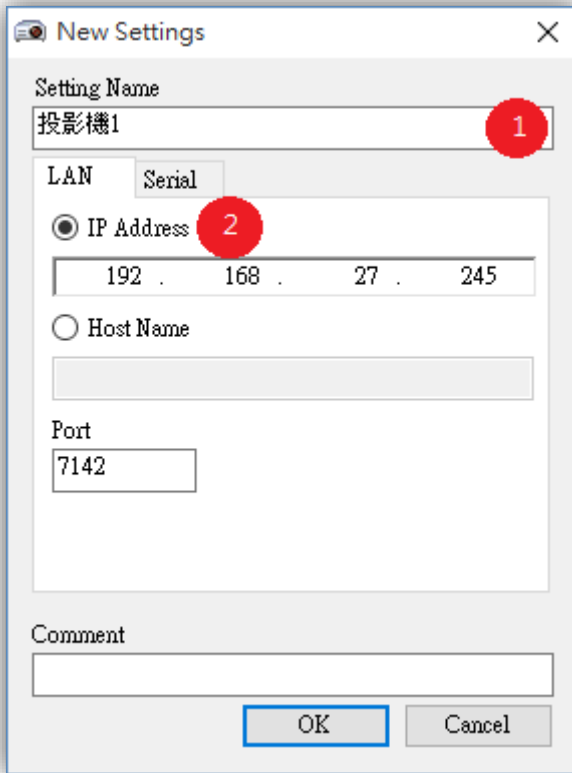


此版本傳輸方式較為特別，從使用者端發出控制命令，再經由伺服器端轉送給目的裝置，使用者與伺服器屬於主從式架構，伺服器與目的裝置屬於點對點傳輸，此版本的 iProject 以 Node.js 做為後台程式，我們將程式部署至提供 PaaS 的廠商，例如：Heroku，或是使用彈性較大的 IaaS，例如：Amazon Web Services，進而減少架設伺服器的支出。

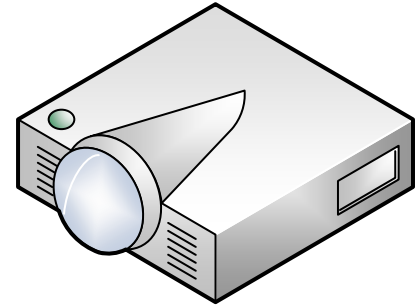
(七) 原廠程式與 iProject 操作流程比較

以下我們用已開發出來的 iProject 2.0 與原廠程式在開關投影機的操作比較，發現確實能夠簡化使用流程：

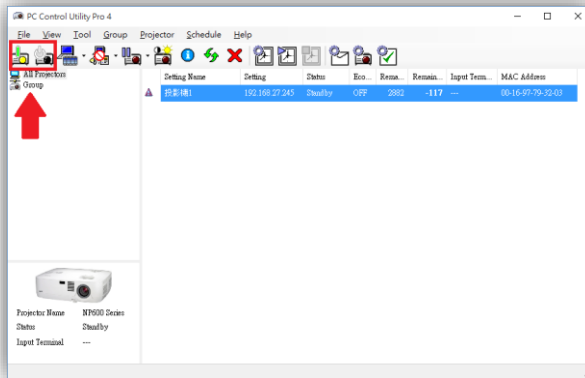
操作	原廠程式	iProject 2.0
第一步	 <p>按下紅色框框內的按鈕，設定要控制的投影機。</p>	 <p>在下方 IP Address 下方文字方塊內，輸入投影機的 IP 位址及 Port 號。</p>
第二步	 <p>按下按鈕 New Settings，進行設定的動作。</p>	 <p>按下 ON，即可開啟投影機； 按下 OFF，即可關閉投影機。</p>



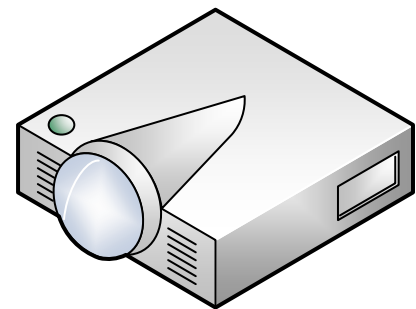
第1點：先設定投影機名稱
第2點：輸入投影機 IP 位址



投影機已開啟



按下紅色框框內左邊的按鈕，即可開啟投影機；要關閉投影機，則按下右邊的按鈕。



投影機已開啟

伍、研究結果

一、iProject 2.0 版

(一) 投影機控制

根據我們的研究結果，成功設計出 iProject 1.0 與 2.0 版，以下為 iProject 2.0 的實際操作方式：

第一步：在 IP Address 下方文字方塊內，輸入投影機的 IP 位址及 Port 號（預設 7142）。

第二步：按下按鈕 ON，即可開啟投影機；按下按鈕 OFF，即可關閉投影機。

第三步：按下選項按鈕 **Computer**，可以將訊號源調整為電腦；按下選項按鈕 **Video**，可以將訊號源調整為電視機。

第四步：按下按鈕 **Bye!**，即可關閉 iProject 程式。

(二) IP 記錄簿

另外，iProject 2.0 也能夠記錄投影機 IP 位址，方便下一次使用的 Name-Address Book 功能操作如下：

第一步：按下按鈕 **Name-Address Book** 即可開啟。

第二步：在 Projector Name、IP Address、Port 下方各文字方塊內，依序輸入投影機名稱、投影機 IP 位址、投影機 Port 號，最下方的文字方塊屬於備註區，當有更多投影機時，可以用備註區記下它們的名稱（IP Address）。

第三步：當資料輸入完畢後，按下按鈕 **Save**，即可儲存。

第四步：按下按鈕 **EXIT**，即可關閉 Name-Address Book。



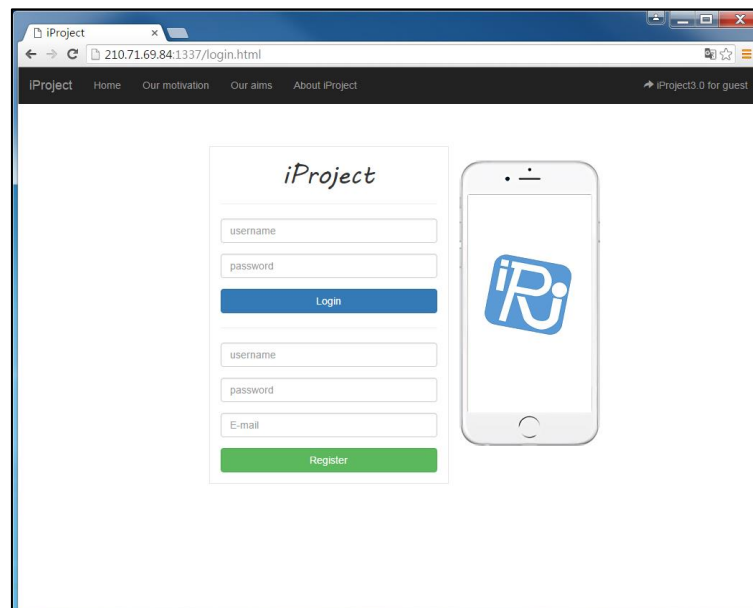
二、iProject 3.0 版

(一) 簡介

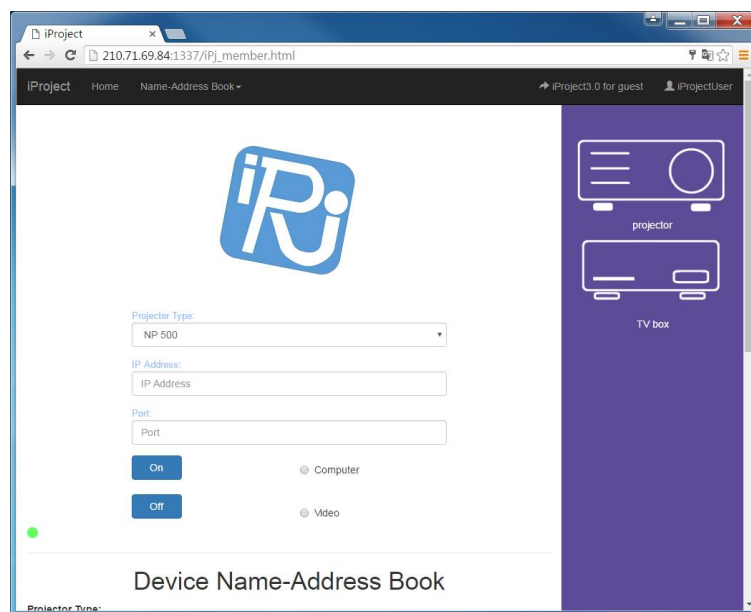
3.0 版目前雖為測試階段，但已經有一定的完成度，主要是以 Node.js 做為後台程式，使用 MongoDB 做為資料庫，以下是 iProject 3.0 的實際操作方式與簡介：



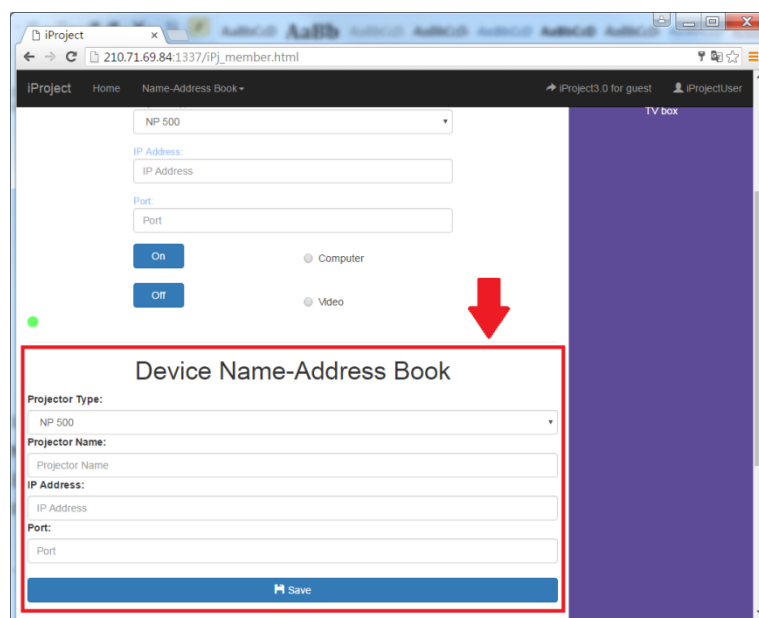
上圖為 iProject 3.0 版的首頁，只要將 3.0 版部署至託管網站，輸入網址即可進入該網頁，上方的導覽列能夠讓使用者更快的找到我們的動機、我們的目標、關於 iProject 等。iProject 3.0 有分會員版與非會員版，主要的差異在於有無提供記錄簿、是否能同步開啟設備等功能，若要登入或申請會員可以點選藍色會員鈕或點選導覽列中的 Login、Register。



上圖為登入與註冊頁，由圖中可知，申請會員步驟簡單不繁雜，僅需輸入使用者名稱、密碼、電子信箱，讓使用者在申請會員時更加簡便，登入時輸入使用者名稱與密碼即可。



上圖為會員版的控制頁面，在沒有使用記錄簿的情況下，需手動輸入投影機型號、IP 位址、Port，以控制要控制的設備，若已經記錄要控制的投影機資料，只需在導覽列中點選 **Name-Address Book** 來做選擇，並會將選擇的設備型號、IP 位址、Port 號輸入至下方的控制頁面中。左邊紫色選單區是讓使用者選擇要控制的設備為投影機或是機上盒，只要點選該設備的圖示，就會切換控制介面，未來會新增更多的設備種類，來達到網頁控制家電的目標。



上圖為設備記錄簿，提供會員記錄常使用的設備型號、IP 位址、Port，更可以將該設備定義名稱，讓使用者更加清楚該筆資料為何處的設備。

陸、討論

Q：為什麼有原廠程式還要製作 iProject？

A：原廠程式介面不夠人性化、難以摸索，若要上手需要花很多時間；但 iProject 就不一樣了，介面乾淨、操作簡單，不管大人、老人、小孩通通輕易上手！

Q：已經有廠商推出類似的 App 或產品，請問為何還要做 iProject 呢？

A：iProject 的功能不侷限於同一個區域網路，就算人在美國要控制台灣的投影機也不是問題；而類似的 App 通常只侷限於同一個區域網路內。

Q：iProject 有什麼特色？

A：iProject 的特色在於不需要使用不環保的遙控器與電池，能夠用手機控制投影機，減少遙控器與電池的產量，來達到降低 CO₂ 排放的效果，將帶給世界很大的轉變。

Q：請問 iProject 只能做到開關投影機、切換訊號源嗎？

A：這個說法不完全正確，若要新增別的功能也是可以，但我們強調簡單操作、介面乾淨，所以目前只先將最重要的功能做出來。

Q：那我可以使用智慧型手機來操控投影機嗎？

A：可以，iProject 3.0 可以讓任何能開啟瀏覽器的裝置用來操控投影機。

Q：iProject 還是需要以電腦或手機作為媒介，這樣真的有環保到嗎？

A：絕對是有的。雖然以電腦或手機作為媒介，但在這廣大的資訊時代，3C 產品人人皆有，是生活中的必需品；而遙控器就會變成劣等財，當大家都不使用遙控器的時候，遙控器的生產就會停止，製造遙控器的汙染也會減少，如此一來，就能減少地球汙染。

Q：無線網路能夠使用 iProject 嗎？

A：可以的。只要是投影機與電腦都有連上網路就可以使用。

Q：那在沒有網路的情況下該怎麼辦？

A：iProject 是針對有網路環境下所開發的應用程式，適合在網路通暢的情況下使用，現在大多數家庭、學校、公共場所等，都設有網路，沒有網路的情形會越來越少，縱使沒有對外的網路，也可以使用內部網路，在同一個區域網路內操作。

Q：iProject 安全嗎？如果其他人也使用 iProject 是否能夠任意操作我的投影機呢？

A：iProject 3.0 已經製作帳號登入的機制，可用來防止有心人士任意操作投影機。

Q：我發現我家裡是私有 IP，但我想在其他場所開啟怎麼辦？

A：若是沒有連上網際網路的內部網路（即封閉式網路），則無法開啟；若是有連上網際網路的內部網路，則需進行 NAT 轉址的動作。

Q：當你無法知道你所要控制的投影機 IP 該怎麼辦？

A：若無特別設定，投影機預設 IP 位址為 192.168.0.10，若要更改其 IP 位址，可在網址列輸入 192.168.0.10，就可以做更改的動作，要是怕忘記自己設定的 IP 位址，可以多利用 Device Name-Address Book 來記錄 IP 位址。

Q：是否能同時開關兩個以上投影機，要怎麼做？

A：可以，iProject 3.0 可以新增此功能。

Q：使用 iProject，是否需要特定的投影機？

A：這個說法不算正確，也可以客製化，另外幫別種型號的投影機製來開發，或是將來投影機控制碼統一，就能夠控制任何一種型號。

柒、結論

未來可延伸至各種家電，如：電視機、冷氣機、電扇、暖氣等，將它們的按鈕匯集於同一軟體介面上，就能方便開關。在家中省去尋找的時間，也能讓工廠人員更為方便。

也希望經過慢慢改良，取代資訊講桌。資訊講桌適合放於教學教室，但價格昂貴，一台通常要價新台幣三萬元，擁有開關投影機、利用數位手寫螢幕教學等功能，還能放置電腦以讀取 CD、DVD 與隨身碟。只要擁有一台能觸控螢幕的筆記型電腦配合著程式，這樣一來，除了麥克風之外，就包含了所有資訊講桌的功能，更名為學校省下龐大的資金支出。

iProject 3.0 版本只需要在網頁中開啟即可使用，且支援各大瀏覽器，相當方便。我們的網站設有會員管理的功能，擁有了自己的帳號密碼，便能使用 Device Name-Address Book 記錄簿，輸入個人平時較常使用的投影機 IP 位址，並做同步開啟設備的動作。運用專屬自己的帳密，可依個人喜好控制多種設備的設定。

iProject 3.0 版本因為是採用 Web 型態，不再需要安裝一堆應用程式在電腦上，3.0 版本的 SaaS 實現，可以拓展到其他能夠連網且用網路控制的設備，不再需要下載及安裝程式，將控制功能彙整到 3.0 的 SaaS 上，就能減少更多軟體的安裝，硬碟容量需求減少了，便能促使固態硬碟取代磁性硬碟，減少終端使用者在硬體上的負擔。更進一步，iProject 3.0 的持續開發，有可能促進物聯網入口網站的誕生。

捌、參考資料及其他

- 何旅良。實戰封包分析 | 使用 Wireshark (2012)。臺北市：碁峯。
- 施威銘、吳文立、李亮生 (2014)。計算機概論 II。臺北市：旗立。
- 德瑞工作室 (2012)。打造 HTML5+CSS3 網頁設計法則：jQuery 行動加碼。臺北市：松崗。
- Chrome DevTools (無日期)。chrome.socket.tcp。檢索日期：2015 年 5 月 20 日，取自 https://developer.chrome.com/apps/sockets_tcp
- Chrome DevTools (無日期)。chrome.storage。檢索日期：2015 年 5 月 20 日，取自 <https://developer.chrome.com/extensions/storage>
- Chrome DevTools (無日期)。Network Communications。檢索日期：2015 年 5 月 20 日，取自 https://developer.chrome.com/apps/app_network
- Chrome platform API (無日期)。禁用網頁特性。檢索日期：2015 年 5 月 20 日，取自 https://crxdoc-zh.appspot.com/apps/app_deprecated
- IDC 國際數據資訊 (2013)。2013 年第三季台灣投影機市場出貨較去年同期衰退。檢索日期：2016 年 4 月 29 日，取自 <http://www.idc.com.tw/about/430.html>
- IDC 國際數據資訊 (2014)。受教育專案聯合採購案刺激，2014 第二季台灣投影機市場出貨較去年同期微幅上升。檢索日期：2016 年 4 月 29 日，取自 <http://www.idc.com.tw/about/442.html>
- IDC 國際數據資訊 (2015)。年終預算消化強勁，2014 第四季台灣投影機市場銷售暢旺。檢索日期：2016 年 4 月 29 日，取自 <http://www.idc.com.tw/about/451.html>
- IDC 國際數據資訊 (2016)。年底專案預算消化，2015 年第四季台灣投影市場出貨量達近年高峰。檢索日期：2016 年 4 月 29 日，取自 <http://www.idc.com.tw/about/476.html>
- Netman 網中人 (2002)。TCP 與 UDP。檢索日期：2014 年 12 月 28 日，取自 http://www.study-area.org/network/network_ip_tcp.htm
- W3Schools (無日期)。Online Web Tutorials。檢索日期：2014 年 3 月 15 日，取自 <http://www.w3schools.com>
- 維基百科 (2015)。HTML。檢索日期：2015 年 3 月 15 日，取自 <https://zh.wikipedia.org/wiki/HTML>
- 維基百科 (2016)。Node.js。檢索日期：2016 年 6 月 12 日，取自 <https://zh.wikipedia.org/wiki/Node.js>

【評語】 052508

1. 本作品開發一個透過有線網路來控制投影機開關功能的程式。
2. 本作品成功達成目標並可同時控制多台投影機。
3. 具有類似功能的系統已經存在，建議多開發創新有用的功能，讓作品更能有自己的特色。