



新北市 104 年度國中職業試探暑假育樂營

新北市 104 年度國中職業試探暑假育樂營

新北市鶯歌工商資料處理科 小盧老師



開發環境

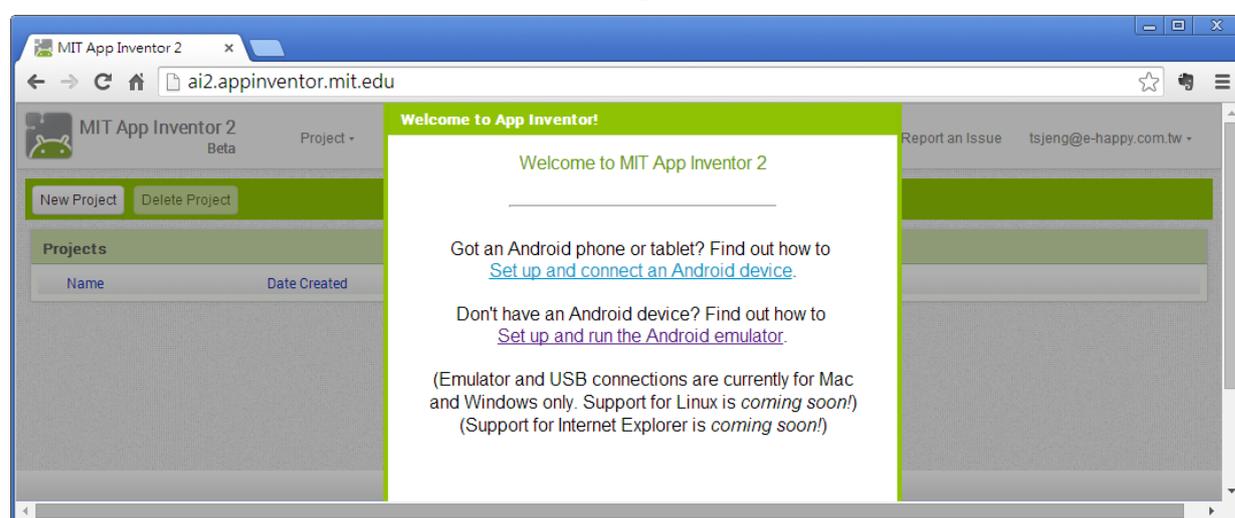
建議使用 Chrome 瀏覽器

安裝 App Inventor 套件(模擬器)

在 <MIT_App_Inventor_Tools_2.3.0_win_setup.exe> 上按滑鼠左鍵兩下。

登入 App Inventor 管理頁面

在瀏覽器中輸入「[http:// ai2.appinventor.mit.edu](http://ai2.appinventor.mit.edu)」→ 以 Google 帳號登入。



建立專案

按 **New Project** 鈕 → 輸入專案名稱後按 **OK** 鈕 (名稱只能使用大小寫字母、數字及「_」符號，而且名稱的第一個字元必須是大小寫字母)。

下載原始檔

按 **Project** 鈕，於下拉式選單中點選 **Export selected project(.aia) to my computer** → 下載檔案置於 <C:\使用者\電腦名稱\Downloads>，檔名為 <專案名稱.aia>。

移除專案

勾選要移除專案左方的核選方塊 → 按 **Delete Project** 鈕 → 於確認對話方塊中按 **確定** 鈕

上傳原始檔

按 **Project** 鈕，於下拉式選單中點選 **Import project (.aia) from my computer** → 按 **選擇檔案** 鈕選取要上傳的專案壓縮檔，然後按 **OK** 鈕。

複製專案

按 **Project** 鈕，下拉式選單中點選 **Save project as** → 輸入專案名稱後按 **OK** 鈕。

加入 Label 及 Button 元件

拖曳 **Label** 元件到介面設計區 → 修改名稱:LabelShow → 修改 **Text**、**FontBold**、**FontSize**、**TextColor** 等屬性 → 拖曳 **Button** 元件到介面設計區 → 修改名稱: **ButtonShow** → 修改各種屬性。

開啟拼塊設計師

按 **Blocks** 鈕。

建立拼塊

建立 **Button** 元件 **Click** 事件拼塊 → 建立 **Label** 元件的設定 **Text** 屬性拼塊。

建立模擬器

啟動 **aiStarter** → 按 **Connect** 鈕後在下拉式選單中點選 **Emulator** 項目 → 出現鎖頭圖示時，向右拖曳鎖頭以解鎖 → 由系統自動安裝 **MIT AI2 Companion**。

在模擬器中執行應用程式

按 **Connect** 鈕後在下拉式選單中點選 **Emulator** 項目。

實機中執行前準備工作

1. 如果是要在實機中安裝下載的 **apk** 檔，改變手機應用程式設定：執行 **設定 / 安全性**，勾選 **未知的來源** 選項。
2. 如果是要在實機上進程式模擬，除了上述動作外，請再執行 **設定 / 開發人員** 選項，勾選 **USB 除錯** 中。

WIFI 連線實機執行應用程式

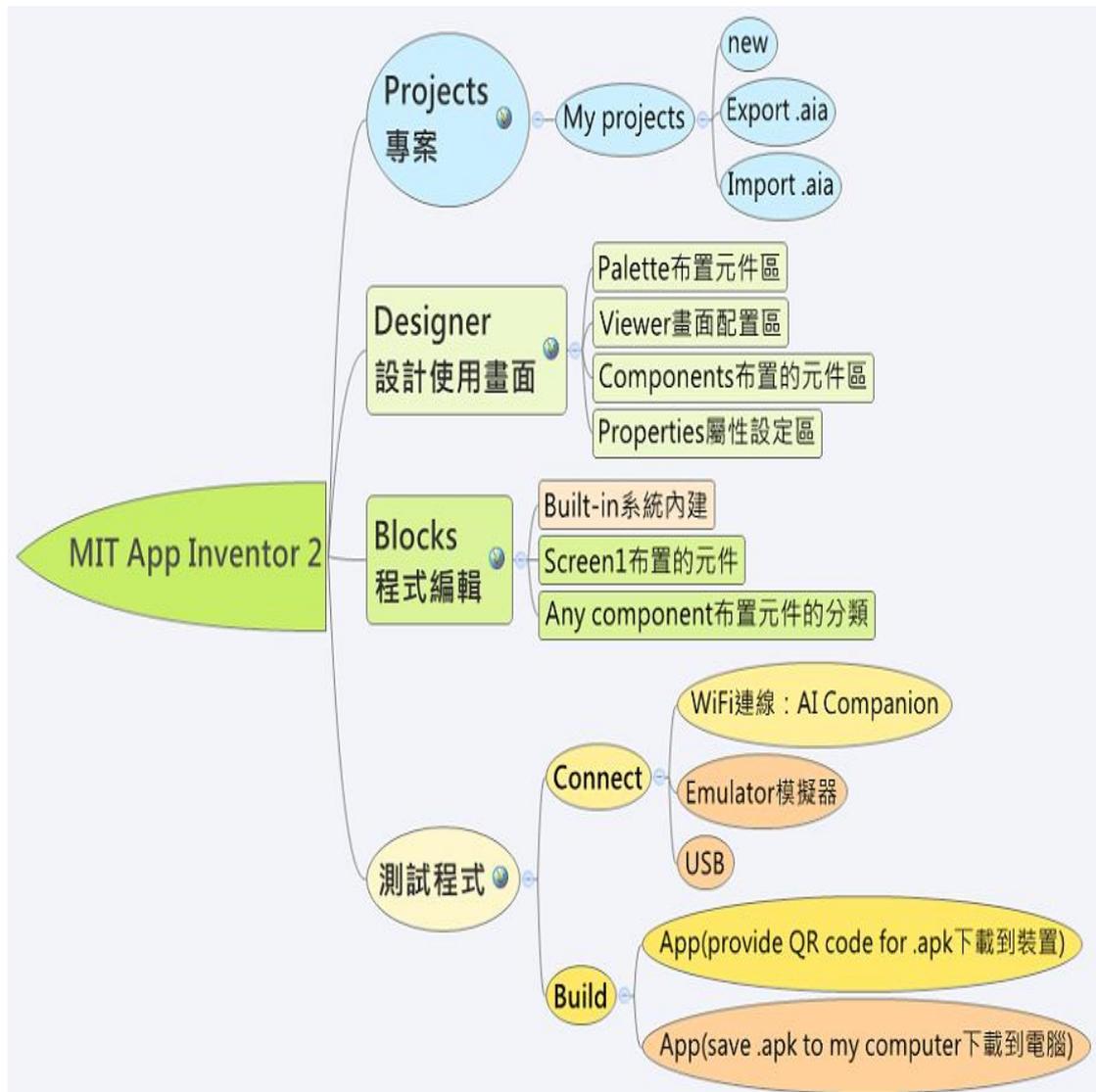
1. 條件：電腦與手機網路在同一個網段。
2. 手機開啟「**Play 商店**」→ 搜尋「**MIT AI2 Companion**」→ 安裝 **MIT AI2 Companion**。
3. 手機執行 **MIT AI2 Companion** → 按 **Connect** 鈕，在下拉式選單中點選 **AI Companion**，將產生的六個字元輸入實機，再於實機按 **connect with code** 鈕；或在實機按 **scan QR code** 鈕，再以實機掃描對話方塊中的 **QR Code** 圖形。

在實機中安裝應用程式

1. 手機開啟「Play 商店」→ 搜尋「QuickMark」→ 安裝 QuickMark。
2. 按 **Build** 鈕，於下拉式選單中點選 **App (provide QR code for .apk)** → 使用實機中的 QuickMark 掃描 QRCode。

USB 連線實機執行應用程式

1. 安裝 91 助手：在 <91assistant_v3.3.15.1023.exe> 上按滑鼠左鍵兩下。
2. 手機以 USB 傳輸線與電腦連接 → 按 **Connect** 鈕後在下拉式選單中點選 **USB** 項目。



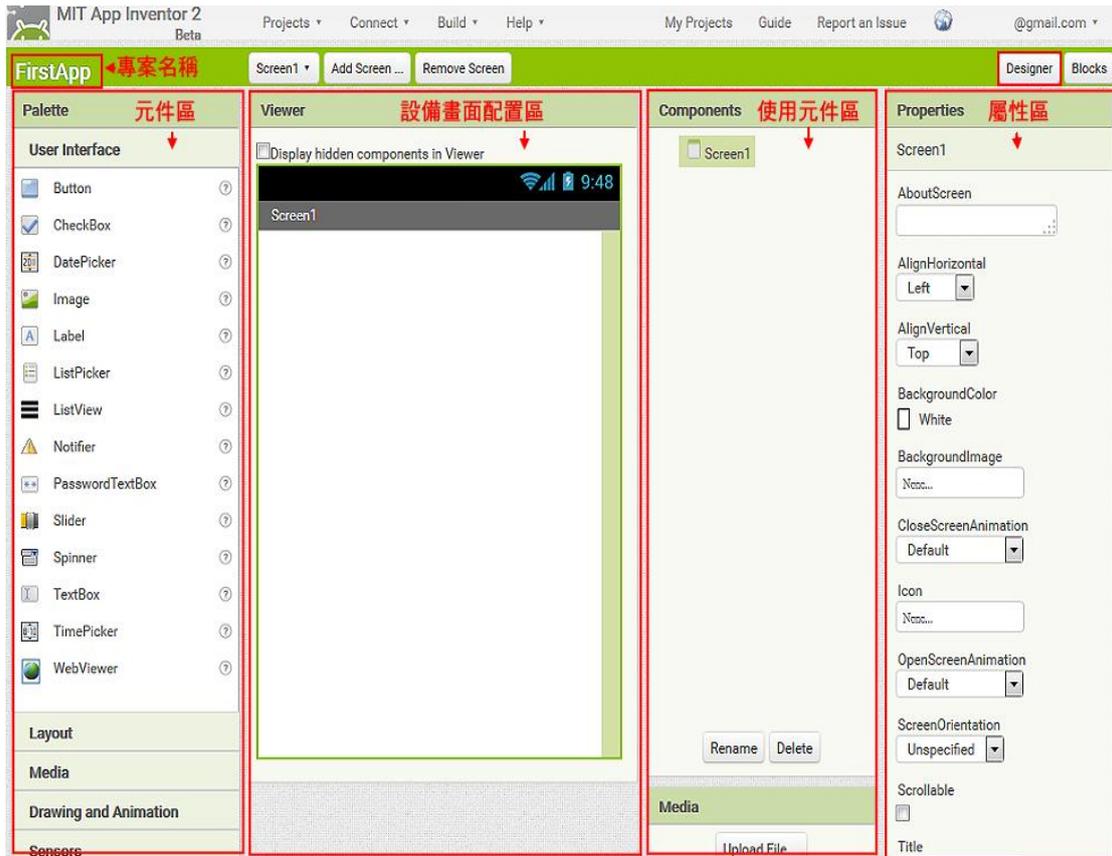
<https://sites.google.com/site/jingprogram/app>

第 1 節-開發環境介紹

準備工作:

- 1、先登入 app inventor
- 2、手機是否有條碼掃描 APP
- 3、試掃描一下是否可以安裝軟體(用第一支程式當範例)

介紹 App Inventor2



第 2 節-語音元件介紹

語音元件介紹

- 語音辨識元件及文字轉語音元件介紹
- SpeechRecognizer 元件
- 文字轉語音(TextToSpeech)

TextToSpeech 語音文字

使您的裝置能念出文字資料。

方法

Speak(Text message)

TextToSpeech.Speak：發送指定文字資料。

```
call TextToSpeech1 .Speak  
message
```

Speech-Recognizer

使用語音辨識器元件聽取使用者口語，並以Android的語音辨識功能，將語音資料轉換成文字。

屬性

Result

SpeechRecognizer.Result：辨識器產生的最後一段文字資料。

```
SpeechRecognizer1 . Result
```

方法

GetText()

SpeechRecognizer.GetText：請使用者發言，並將語音資料轉換為文字資料。

當有結果可用時，會發出 AfterGettingText 事件訊號。

```
call SpeechRecognizer1 .GetText
```

事件

AfterGetting(Text result)

SpeechRecognizer.AfterGetting：在辨識器產生文字資料之後發出訊號，這個參數代表產生的文字資料。

```
when SpeechRecognizer1 . AfterGettingText  
result  
do
```

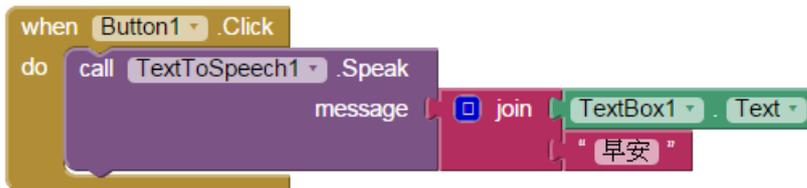
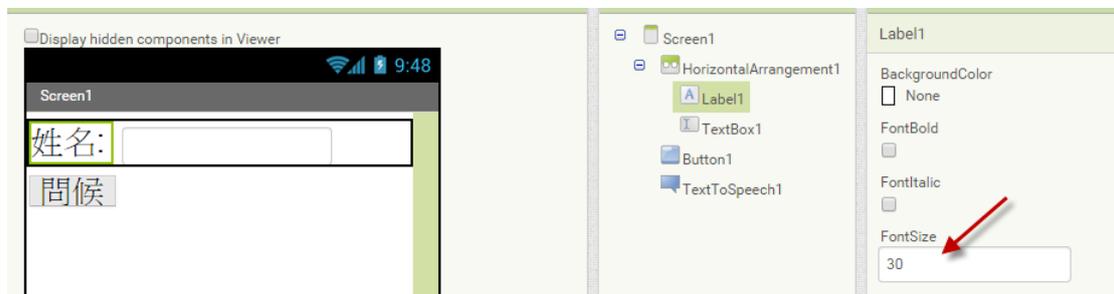
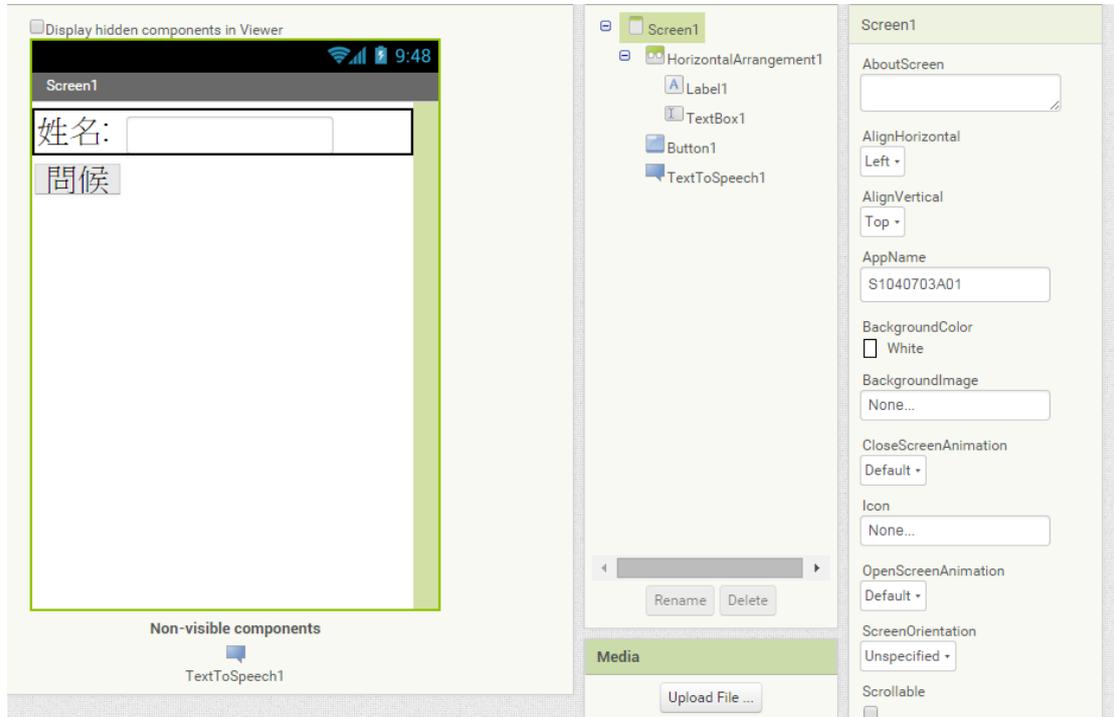
BeforeGetting Text()

SpeechRecognizer.BeforeGettingText：在辨識器呼叫之前發出訊號。

```
when SpeechRecognizer1 . BeforeGettingText  
do
```

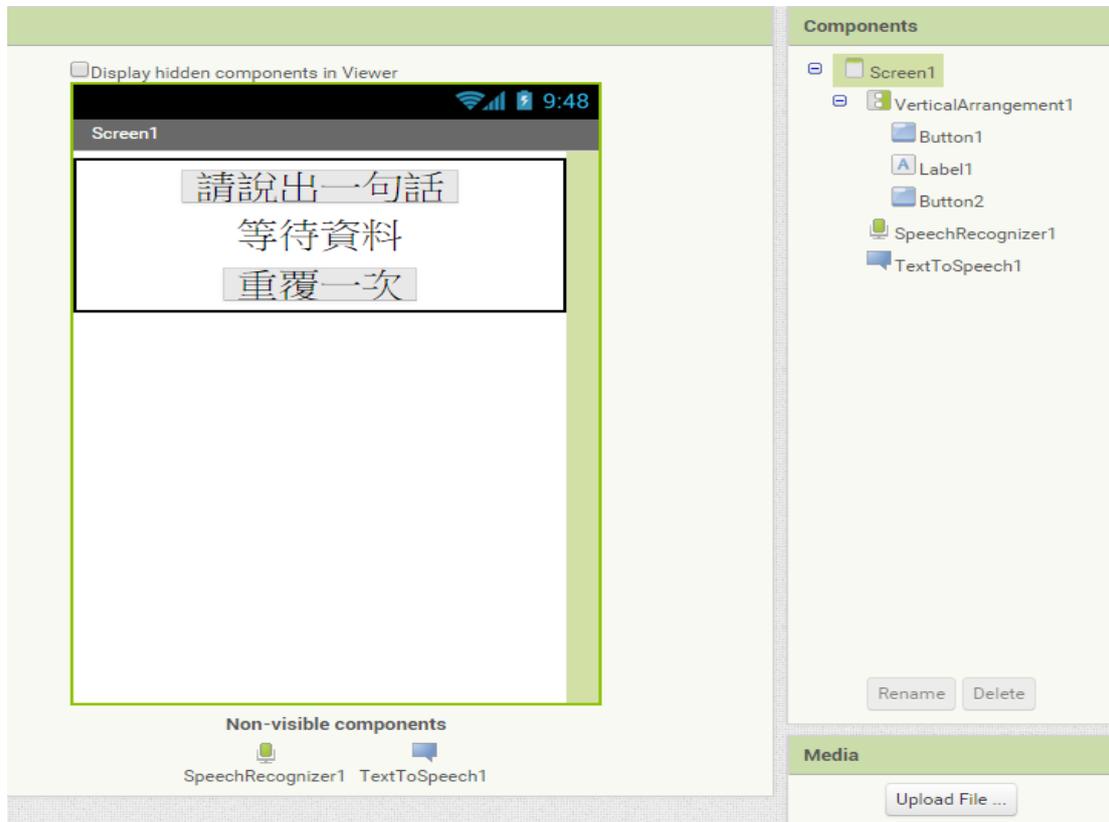
第 3 節-聲控練習(一)

製作第一支 APP-打招呼 APP



第 4 節-聲控練習(二)

製作第二支 APP-聲控點歌機辨識 APP



說一句話，語音轉文字

```
? when Button1 .Click  
do call SpeechRecognizer1 .GetText
```

將語音文字放至label物件

```
? when SpeechRecognizer1 .AfterGettingText  
result  
do set Label1 . Text to get result
```

重覆說一次，將文字轉語音

```
? when Button2 .Click  
do call TextToSpeech1 .Speak  
message Label1 . Text
```

第 5 節課-拍照功能元件

拍照功能元件:製作第三支 APP-我的照像機

1. 建立專案：camera。

Title 屬性：照相機。

2. 介面設計：

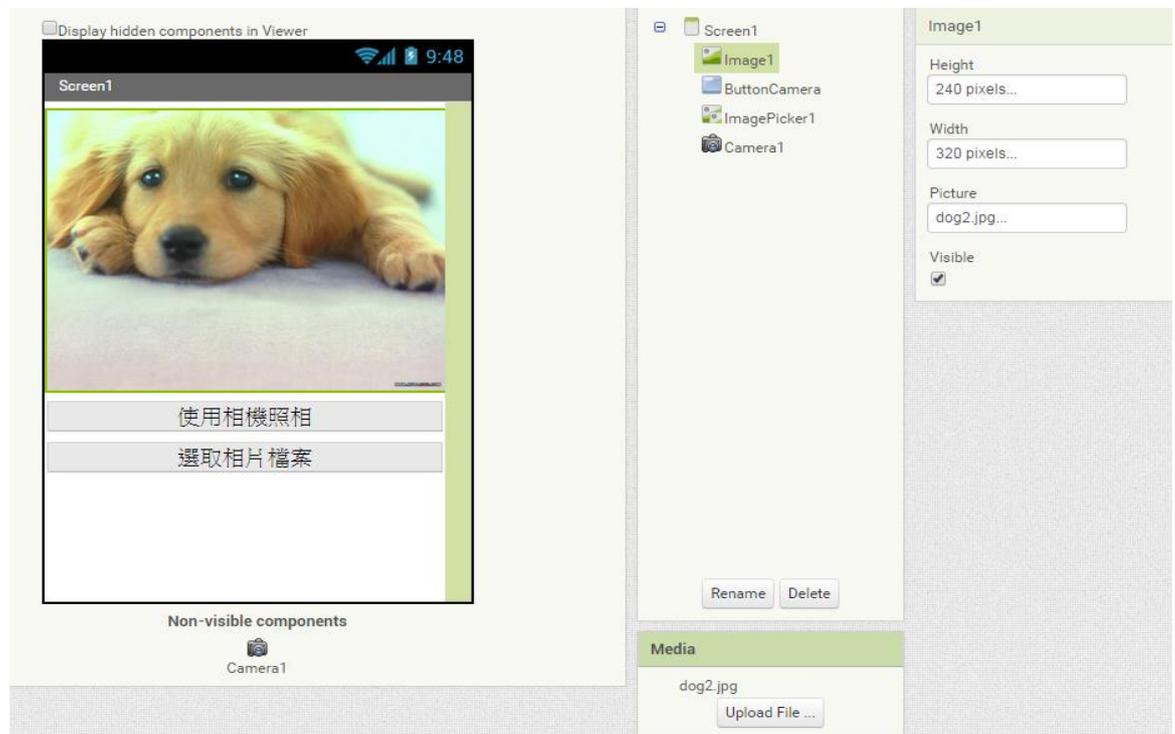
上傳 <img01.jpg>。

建立 Image 元件：Picture: img01.jpg、Width:320 pixels、Height:240 pixels。

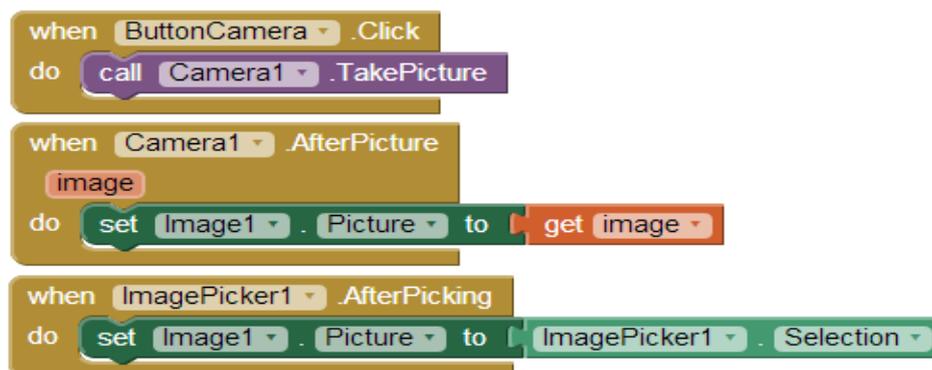
建立 Button 元件：Rename:ButtonCamera、FontSize:20、Text:使用相機照相、Width:Fill parent。

建立 ImagePicker 元件：FontSize:20、Text: 選取相片檔案、Width:Fill parent。

建立 Camera 元件。



3. 拼塊設計：



第 6 節課-聲音元件介紹

SOUND 元件及 Player 元件介紹

Sound 聲音

Sound 元件可用來播放較短的音效檔，或使裝置震動。

Sound 為一非可視元件，它可用來播放音效檔和讓手機震動（單位為毫秒）。要播放的音效檔檔名可在 Designer 或 Block Editor 中設定。

請參考 <http://developer.android.com/guide/appendix/media-formats.html> 來參考有關檔案類型的詳細資訊。

Sound 元件適用於播放較短的聲音檔，如果要播放較長的聲音檔例如一首歌，這時請使用 Player 元件。

屬性

Source

指定要播放的音效檔。

Sound.Source : 取得 Sound 播放的聲音檔

Set Sound.Source : 設定 Sound 播放的聲音檔

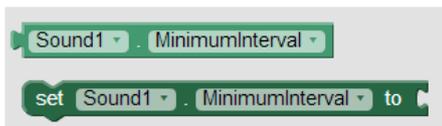


MinimumInterval

播放時間長度，單位為毫秒。

Sound.Source : 取得 Sound 播放的時間長度

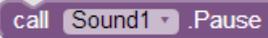
Set Sound.Source : 設定 Sound 播放的時間長度



方法

Pause

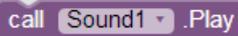
Sound.Pause：暫停播放音效檔。



```
call Sound1 .Pause
```

Play

Sound.Play：開始播放音效檔。

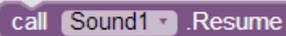


```
call Sound1 .Play
```

Resume

回復先前狀態。

Sound.Resume：重新播放已暫停之音效檔。



```
call Sound1 .Resume
```

Stop

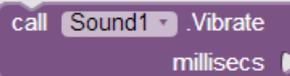
Sound.Stop：停止播放音效檔。



```
call Sound1 .Stop
```

Vibrate(number millisecs)

Sound.Vibrate：使手機震動，需指定震動時間，單位為毫秒。



```
call Sound1 .Vibrate  
                  millisecs
```

第 7 節課-聲音練習(一)

製作第四支 APP-動物叫叫 APP

第 8 節課-聲音練習(二)

製作第五支 APP-我的小鋼琴

1. 建立專案：piano。

Title 屬性：小鋼琴家、ScreenOrientation:Landscape。

2. 介面設計：

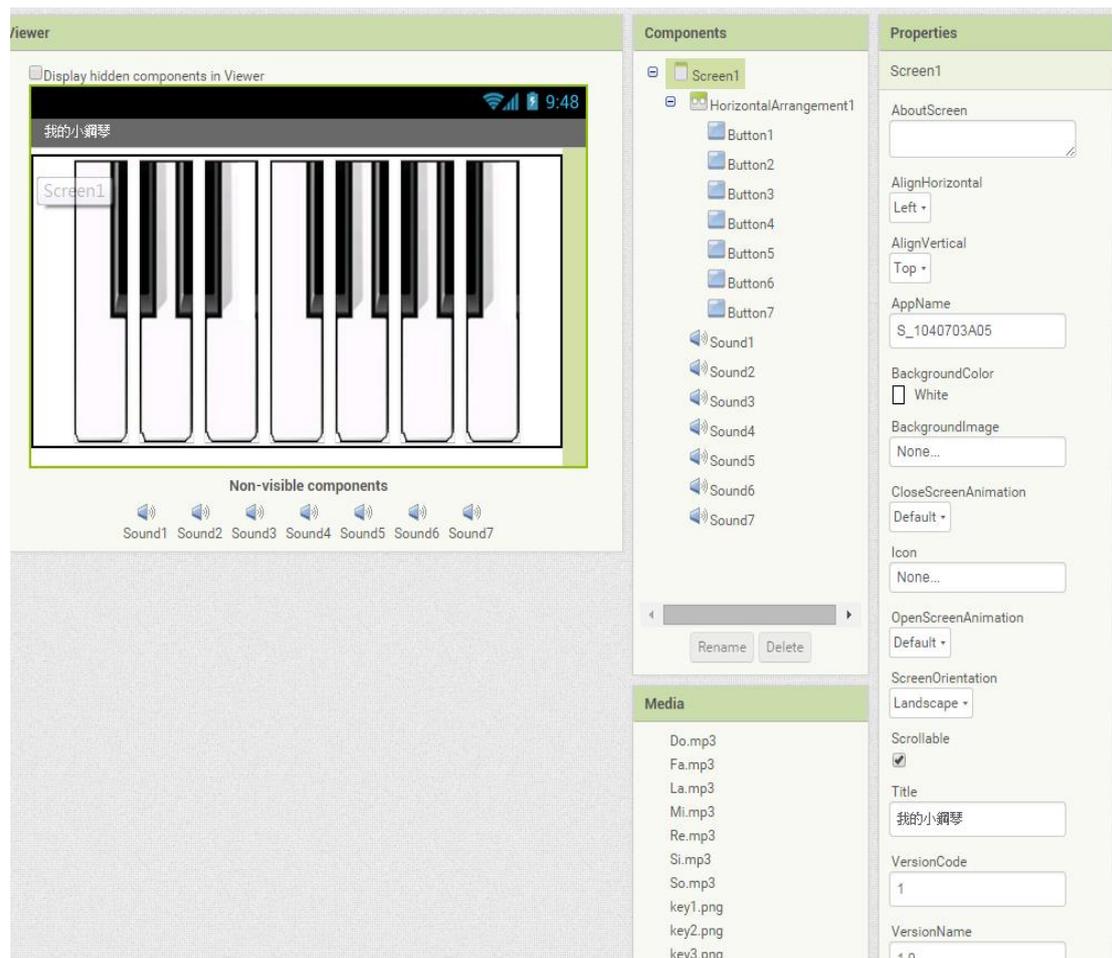
上傳 <key1.png> 到 <key3.png>、<Do.mp3> 到 <Si.mp3>。

建立 HorizontalArrangement 版面，AlignHorizontal:Center、Width:Fill parent、Height:Automatic。

建立 Button 元件：Image: key1.png、Text:無、Width:50 pixels、Height:240pixels。

再建立六個 Button 元件，名稱為 Button2 到 Button7：Image 屬性依次為 key2.png、key3.png、key1.png、key2.png、key2.png、key3.png，其餘屬性與 Button1 相同。

建立七個 Sound 元件。



3. 拼塊設計：程式開始時設定七個按鍵的音效。

```
when Screen1.Initialize
do
  set Sound1.Source to "Do.mp3"
  set Sound2.Source to "Re.mp3"
  set Sound3.Source to "Mi.mp3"
  set Sound4.Source to "Fa.mp3"
  set Sound5.Source to "So.mp3"
  set Sound6.Source to "La.mp3"
  set Sound7.Source to "Si.mp3"
```

4. 按下按鍵時播放音效。

```
when Button1.Click
do call Sound1.Play

when Button2.Click
do call Sound2.Play

when Button3.Click
do call Sound3.Play

when Button4.Click
do call Sound4.Play

when Button5.Click
do call Sound5.Play

when Button6.Click
do call Sound6.Play

when Button7.Click
do call Sound7.Play
```