



[03] 翻譯麥克風

- 學習下拉式選單元件
- 學習對話框元件顯示警告訊息
- 學習語音辨識元件
- 學習文字語音轉換器元件
- 學習 Yandex 語言翻譯器元件





3.1 認識 App 專題：翻譯麥克風

3.1.1 專題介紹

最近售價高達新台幣 8000 元的日本翻譯神器 ILI 在台灣賣翻了，下訂單後常需等半個月才能拿到貨品，其實利用 **語音辨識**、**文字語音轉換器** 及 **Yandex 語言翻譯器** 元件，只要少許程式拼塊，就能製作出比 ILI 功能更強大的即時語音翻譯機了！

本專題設計原理是先以 **語音辨識** 元件將語音轉換為文字，再以 **Yandex 語言翻譯器** 元件翻譯為輸出語言，最後用 **文字語音轉換器** 元件以輸出語言唸出翻譯文稿。

3.1.2 專題作品預覽

「翻譯麥克風」App 預設翻譯語言是英文，使用者可以點選 **翻譯語言** 下拉式選單設定要翻譯的語言，為簡化程式，本專題只能使用英文、日文、韓文三種語言。

使用者點按下方麥克風按鈕就可用中文語音輸入文句，程式會顯示說出的中文，然後以翻譯語言進行翻譯，翻譯完成後會顯示翻譯後的文字，並且用翻譯語言讀出翻譯後的文字。



學習小叮嚀

本專題會使用到麥克風以及網路語言套件，測試時建議要確定網路使用狀態並使用實機測試才能正確操作其中的功能。



[04] 電子書

- 學習多畫面螢幕切換
- 加入畫布、設定畫布背景圖片
- 學習當畫布被滑過使用拼塊控制畫面切換
- 學習複製不同畫面的程式拼塊
- 學習音樂播放器播放聲音





4.1 認識 App 專題：電子書

4.1.1 專題介紹

近年來，教育界盛行將閱讀融入教學活動中，除了紙本形式之外，更大量採用電子書來呈現。這不只是環保的考量，電子書可以加入互動遊戲的特色以及多媒體的聲光效果，都是傳統紙本書所無法比擬的。

本專題將設計一款具有影音效果的 App，並且可用左右滑動上、下的翻頁，讓整個 App 更具張力和特色。

4.1.2 專題作品預覽

「電子書」App 共有 5 個螢幕 (Screen)，執行之後會由首頁 (Screen1) 開始播放，並循環播放背景音樂。使用者在 Screen1 向左滑動即可翻到第二頁 (Page1)，在 Page1 向左滑動可翻到下一頁 (Page2)，向右滑動則可回到上一頁 (Screen1)。Page2~Page3 和 Page1 相同，都可以上、下翻頁，在最後一頁 Page4 向右滑動可回到上一頁 (Page3)。



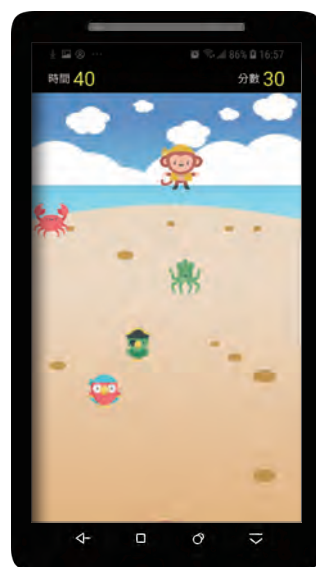
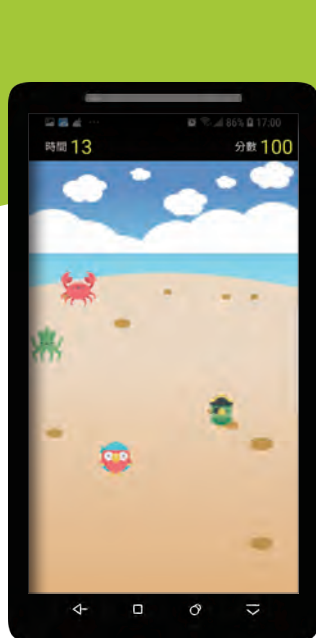
學習小叮嚀

本專題如果要檢視到多個頁面之間換頁的效果，以及避免背景音樂重疊的問題，測試時建議可以將專題發佈為 apk 檔，安裝到實機上進行測試。



[08] 猴子奪寶記

- 學習水平配置高度填滿時元件操作
- 學習畫布、圖像精靈元件
- 學習計時器和音效元件操作
- 學習加速度感測器元件





8.1 認識 App 專題：猴子奪寶記

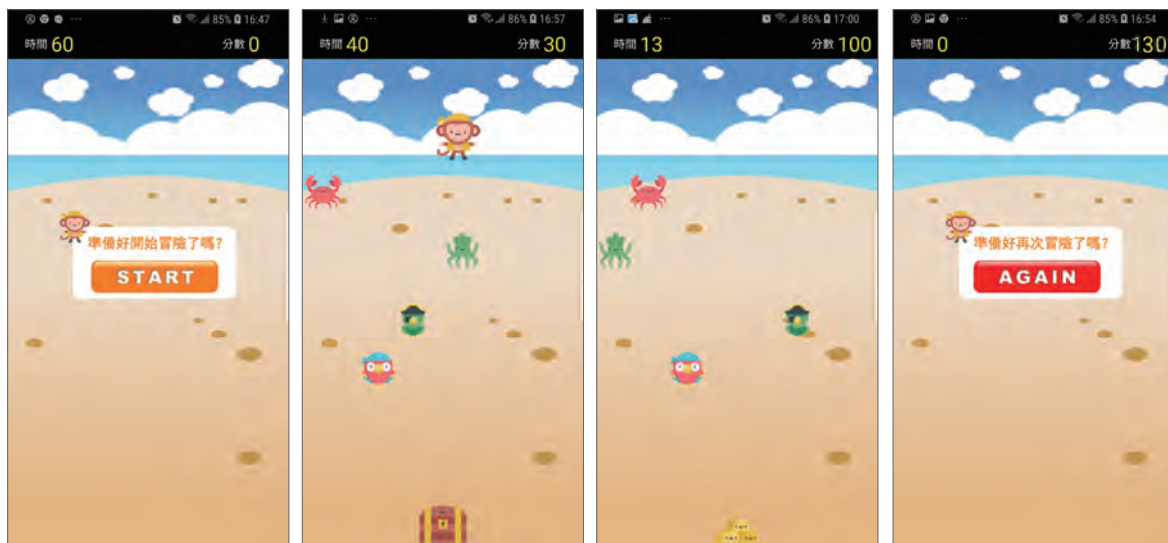
8.1.1 專題介紹

在智慧型手機中大部分都有 **加速度感測器**，可以偵測 Android 行動裝置傾斜狀況，也可以偵測 X、Y、Z 三個軸分量的狀態，單位是 m/s^2 。當搖動行動裝置會觸發 **加速度感測器** 的 **被晃動** 事件。

本專題結合 **加速度感測器**，利用 **加速度感測器** 控制 **主角** 上、下、左、右 的移動，以避開 **防守員** 的防守，到達下方的 **寶藏**。

8.1.2 專題作品預覽

在「猴子奪寶記」App 中，按下 **START** 鈕，開始計時 60 秒，遊戲者必須利用 **加速度感測器** 控制 **主角** 上、下、左、右 的移動，通過 **防守員** 的防守，取得下方的 **寶藏**。當 **主角** 碰到 **寶藏** 得分加 10 分並播放成功的音效，同時顯示 **寶藏** 中的金塊。如果 **主角** 移動過程中碰撞到 **防守員**，則遊戲失敗並播放失敗的音效。



學習小叮嚀

本專題會使用到加速度感測器，建議使用實機測試才能正確操作其中的功能。

8.2 App 畫面編排

使用 App Inventor 2 製作「猴子奪寶記」App，在規劃程式功能及流程後，再依架構設計版面並收集素材，最後即可開始進行畫面編排。

8.2.1 App 畫面編排完成圖



8.2.2 新增專案及素材上傳

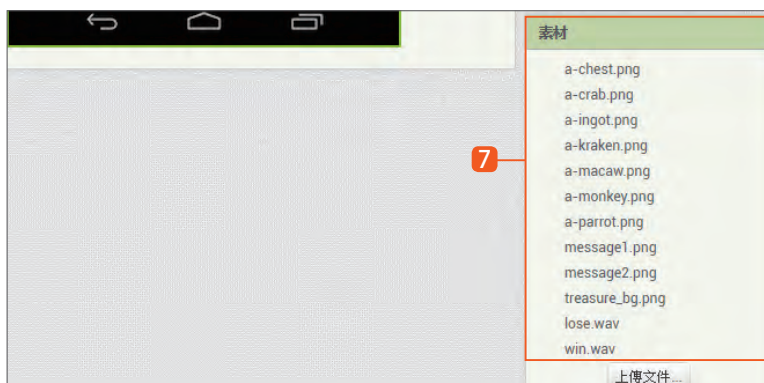
在「猴子奪寶記」App 的範例畫面編排中，除了顯示時間和得分的 **標籤** 之外，最重要的是要加入 **畫布**、**圖像精靈** 及 **音效**，並利用 **計時器** 計時，同時以 **加速度感測器** 控制猴子的移動。



- 1 登入開發頁面按 **新增專案** 鈕。
- 2 在對話方塊的 **專案名稱** 欄位中輸入「ex_treasurehunt」。
- 3 按下 **確定** 鈕完成專案的新增並進入開發畫面。



- 4 按 **素材** 的 **上傳文件** 鈕。
- 5 在對話方塊按 **選擇檔案** 鈕。
- 6 在對話視窗中選取本章原始檔資料夾，選取圖片 <a-chest.png> 檔後按 **開啟** 鈕，然後在對話方塊中按 **確定** 鈕。

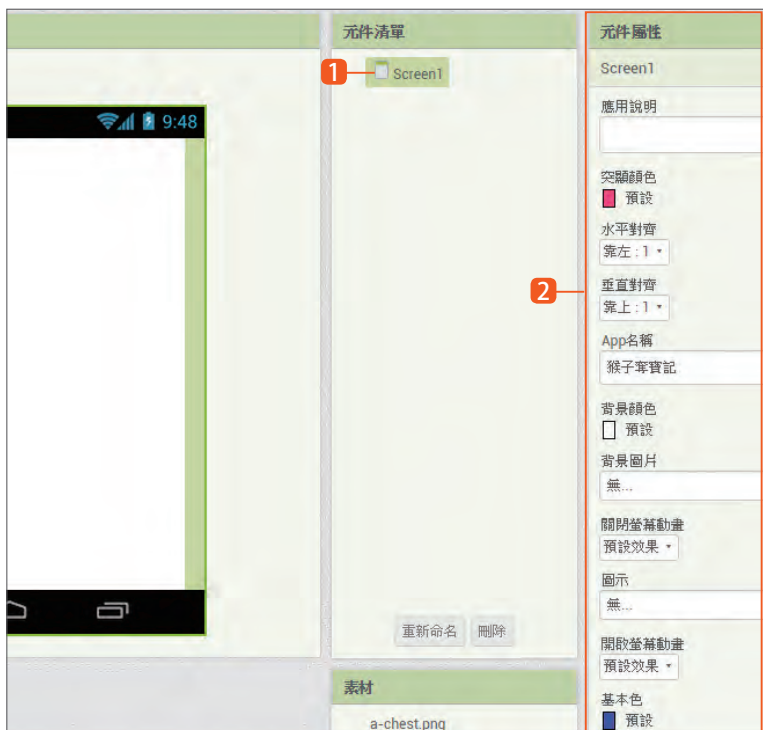


- 7 完成後即可在 **素材** 中看到上傳的圖片檔名 <a-chest.png>。請利用相同的方式將其他圖片和音效上傳到 **素材** 中。

8.2.3 設定畫面

首先進行外觀編排：在 **Screen1** 元件屬性依下列表格進行設定。

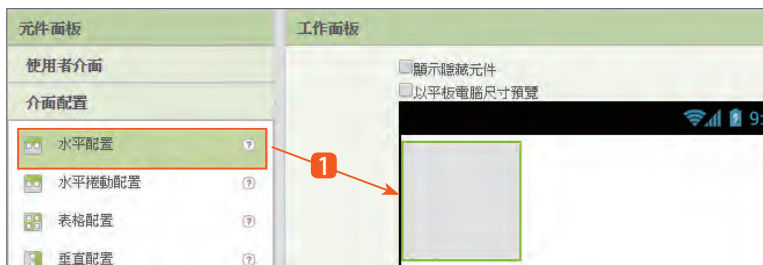
欄位	值	欄位	值
水平對齊	置左：1	允許捲動	取消核選
垂直對齊	靠上：1	視窗大小	自動調整
App 名稱	猴子奪寶記	Theme	Device Default
背景顏色	預設	標題	猴子奪寶記
螢幕方向	鎖定直式畫面	標題顯示	取消核選



- 1 首先進行外觀編排：在 **元件清單** 選按 **Screen1** 準備進行設定，這是預設的畫面元件。
- 2 請在 **元件屬性** 依前面表格資料進行欄位設定。

8.2.4 加入時間和得分標籤

接著在面板中加入水平配置，並在水平配置中加入顯示時間和得分的標籤，請依下述步驟操作：



- 1 在 **元件面板** 拖曳 **介面配置 / 水平配置** 到工作面板中。



- 2 在 **元件屬性** 設定 **水平對齊**：置中、**垂直對齊**：置中，**寬度**：填滿。



3 在 元件面板 拖曳 使用者介面 / 標籤 到剛才的配置區域中。



4 請在 元件屬性 設定 背景顏色：透明、寬度：50 像素、文字：時間、文字顏色：白色。

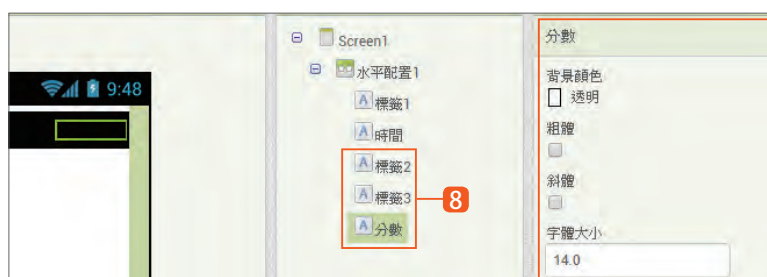


5 在 元件面板 拖曳 使用者介面 / 標籤 到剛才的標籤 1 的右方。



6 按 重新命名 鈕，在對話方塊的 新名稱 欄輸入「時間」後按 確定 鈕。

7 請在 元件屬性 設定 字體大小：24、寬度：50 像素、文字：0、文字對齊：靠左、文字顏色：黃色。

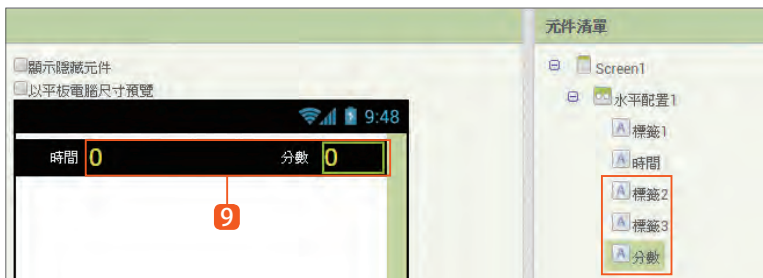


8 重複的操作，再拖曳 3 個標籤到剛才 時間 標籤的右方，預設的名稱為 標籤 2、標籤 3、標籤 4，並將 標籤 4 名稱更改為「分數」。

請依下列表格設定 **標籤 2**、**標籤 3** 和 **分數** 的屬性。

元件名稱	欄位	值
標籤 2	寬度	填滿
	文字	" "
標籤 3	文字	分數
	文字顏色	白色

元件名稱	欄位	值
分數	字體大小	24
	寬度	50 像素
	文字	0
	文字顏色	黃色



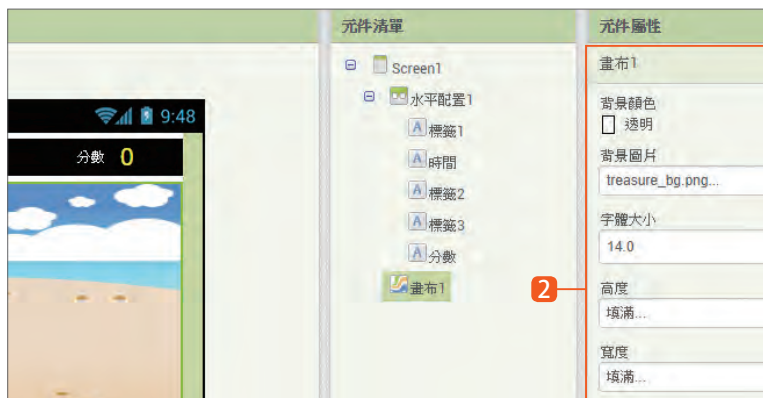
9 完成後的介面如左圖。

8.2.5 加入畫布和圖像精靈

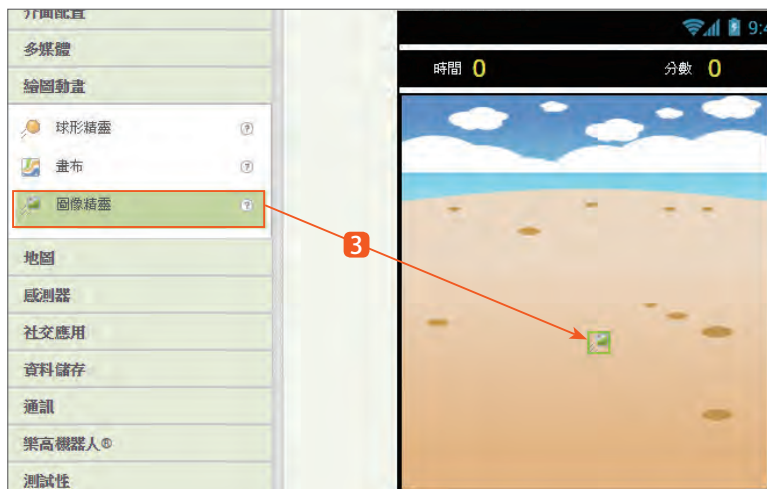
接著要在 **水平配置 1** 下方加入 **畫布**，並在畫布中加入包括 **主角**、**防守員**、**寶藏** 和 **啟動按鈕** 等 **圖像精靈**，請依下述步驟操作：



1 在 **元件面板** 拖曳 **繪圖動畫 / 畫布** 到 **水平配置 1** 的下方。



2 在 **元件屬性** 設定 **背景顏色**：透明、**背景圖片**：**treasure_bg.png**、**高度**：填滿、**寬度**：填滿。



- 3 在 元件面板 拖曳 繪圖動畫 / 圖像精靈 到 畫布1 中，並重新命名為「防守員 1」。



- 4 在 元件屬性 設定 高度：46 像素、寬度：46 像素，間隔：200、圖片：a-crab.png，旋轉：取消核選、速度：0、X 座標：83、Y 座標：148。

重複的操作，再拖曳 6 個圖像精靈到 防守員 1 的下方，預設的名稱為 圖像精靈 1~ 圖像精靈 6，請依序將名稱更改為「防守員 2」、「防守員 3」、「防守員 4」、「寶藏」、「主角」和「啟動按鈕」。



- 5 拖曳 6 個圖像精靈到 防守員 1 的下方，並更改名稱 為 防守員 2、防守員 3、防守員 4、寶藏、主角 和 啟動按鈕。

8.3 App 程式設計

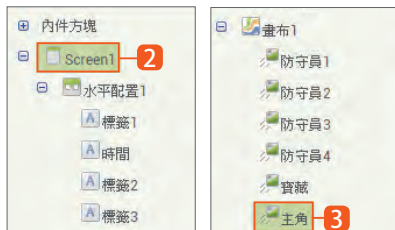
完成了「猴子奪寶記」App 的畫面編排後，接下來就要加入程式設計的拼塊。

8.3.1 應用程式初始化設定

應用程式執行後，隱藏畫布中的所有圖像精靈，只顯示 **啟動按鈕**，並設定時間為 60、分數為 0，同時停止 **加速度感測器 1**。



1 按功能表右方的 **程式設計** 進入程式設計模式。

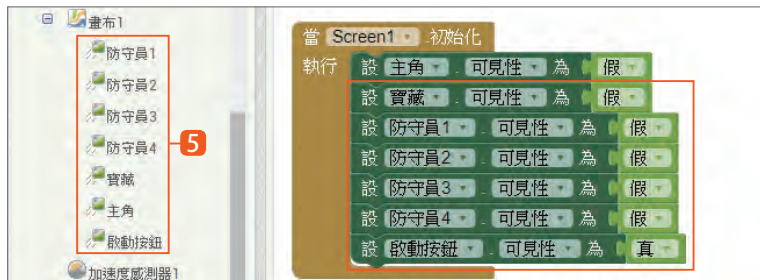


2 取 **Screen1** / 當 **Screen1**. 初始化 到 工作面板 中。

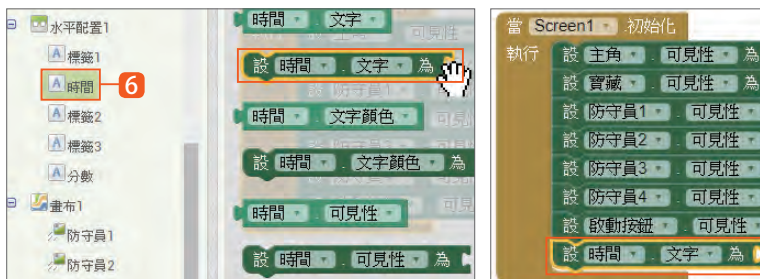
3 取 **主角** / 設 **主角**. 可見性 為 到剛才的方塊中。



4 取 **內件方塊** / **邏輯** / **假** 到剛才的方塊後方。



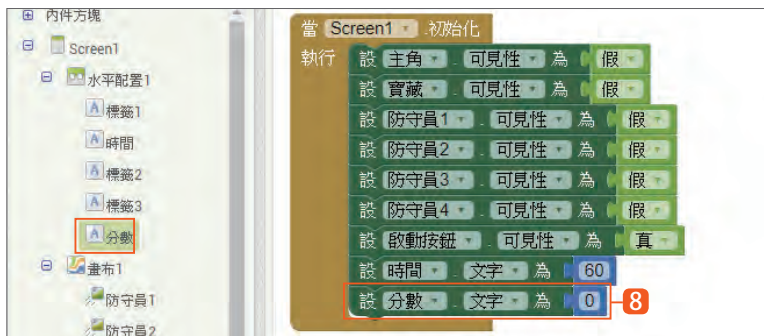
5 請參考左圖再加入畫布中其他圖像精靈的 **可見性** 設定。請注意：除了 **啟動按鈕** 的 **可見性** 為 **真** 外，其它的 **可見性** 均為 **假**。



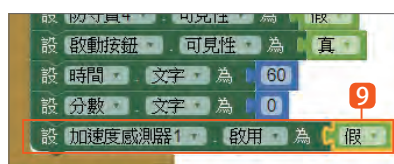
6 取 **時間** / 設 **時間**. 文字 為 到 **啟動按鈕**. 可見性 為 **真** 方塊下方。



- 7 取 內件方塊 / 數學 / 0 到前方塊後方並更改數字為 60。



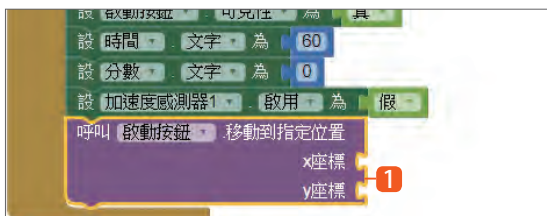
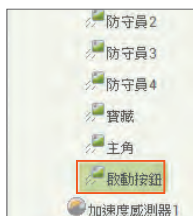
- 8 參考 6 7 的操作，再加入設分數. 文字 為 0 到前方塊後方。



- 9 取 加速度感測器 1 / 設加速度感測器 1. 啟用 為 到剛才方塊下方。取 內件方塊 / 邏輯 / 假 到前方塊後方。

8.3.2 將啟動按鈕移動到指定位置

應用程式執行後，將 啟動按鈕 移動到螢幕中央位置。



- 1 取 啟動按鈕 / 呼叫 啟動按鈕. 移動到指定位置 到剛才方塊下方。



- 2 取 內件方塊 / 數學 / ...-... 到 x 座標 欄位中。



- 3 取 內件方塊 / 數學 / .../... 到 ...-... 的前方。



[12] 自行車道景點地圖

- 學習水平配置指定高度時元件操作
- 學習下拉式選單元件
- 學習地圖和標記元件
- 學習檔案管理元件





12.1 認識 App 專題：台北自行車道景點地圖

12.1.1 專題介紹

地圖類別元件是 App Inventor 新添加的功能，它包括 **地圖**、**標記**、**線條字串**、**多邊形**、**長方形**、**圓形工具** 和 **特徵集** 等七個元件，可以利用這些元件建立地理定位應用程式。

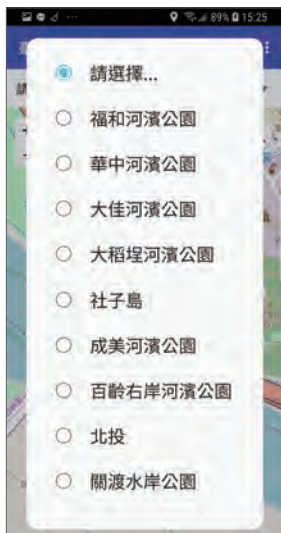
地圖 中可以包含專用於 **地圖** 的特殊元件（稱為「特徵」），其行為與「畫布」上的圖像精靈類似。地圖類型有三種：道路、空中和地形，可以在地圖上觸發許多事件，包括點擊、拖曳和縮放。

標記：表示地圖上的點。可以自訂標記圖示，設定寬度、高度，標記可以設定為可拖曳，也可以資訊框顯示詳細的訊息。

本專題結合 **檔案管理**、**下拉式選單**、**地圖** 和 **標記**，以 **地圖**、**標記** 顯示指定景點的地圖和標記，並以資訊框顯示該景點的詳細資訊。

12.1.2 專題作品預覽

「台北自行車道景點地圖」App 執行後會顯示地圖，並將地圖的中心點移動到「臺北市河濱自行車道景點」附近，同時載入 csv 檔中的景點資訊到 **下拉式選單** 中，選取 **下拉式選單** 中指定的景點，即會將地圖的中心點移動到該景點，並顯示該景點的標記，同時以資訊框顯示該景點的詳細資訊。



學習小叮嚀

本專題因為要使用到地圖元件，在測試時必須要有網路才能載入，這點要請讀者特別留意。



[13] 發票速掃快手

- 學習使用圖像按鈕
- 學習在標籤元件中顯示 HTML 格式的內容
- 學習以水平配置元件分隔元件
- 學習以條碼掃描器元件取得 QRCode 內容
- 學習使用網路元件取得網路資料





13.1 認識 App 專題：發票速掃快手

13.1.1 專題介紹

QRCode 的應用在 App 中一直是許多人喜歡的功能，行動載具就是那麼讓人著迷，不僅擁有多種感測器，還有相機鏡頭能夠捕捉相片，拍攝影片，甚至是掃描條碼。

「發票速掃快手」專題中，我們先透過網路取得發票的得獎號碼，分析網路資料後取出六獎號碼存於清單中，再利用條碼掃描器元件掃描發票 QRCode，取得發票號碼與清單中得獎號碼資料進行比對，即可完成對獎的動作。

為了簡化程式，本專題僅進行「六獎」核對。當然這個專題還能再深入，讓整個 App 更完整、更專業，不過關鍵的技巧都已經在這個專題中學習到了。

13.1.2 專題作品預覽

應用程式執行後會提示需掃描發票「左方」的 QRCode。按 **掃描** 鈕就開啟行動裝置的相機準備掃描發票的 QRCode，掃描後會分析發票資料以取得發票號碼的最後三碼做為比對依據（本專題只對六獎，所以僅需最後三碼）。

接著讀取網路上各期發票中獎號碼資料，擷取三個頭獎的後三碼及增開六獎號碼即為當期的中獎號碼清單，以掃描得到的發票最後三碼和中獎號碼清單比對，若相符就表示該發票中獎，顯示「恭喜你中獎了」訊息；若沒有相符的資料表示未中獎，顯示「好可惜，再對一張！」訊息。如果是最近尚未開獎的發票，系統會顯示「抱歉，資料庫沒有相關資料。」訊息。



學習小叮嚀

本專題因為使用相機鏡頭及網路，在測試時建議要使用實機才能檢視專題的功能。