

# PhoneGap功能介紹

☐ Matis Hsiao

# 學習目標

- ☐ 了解PhoneGap API使用流程
- ☐ 了解PhoneGap 1.4 API功能

# 課程大綱

- ☐ PhoneGap API操作流程
- ☐ PhoneGap API初始化
- ☐ PhoneGap API功能介紹

# PhoneGap 函式庫初始化

## □ 引用 PhoneGap JavaScript Library

```
<script type="text/javascript"  
  charset="utf-8" src="phonegap.js"></script>
```

## □ 呼叫 PhoneGap API 初始化

```
document.addEventListener("deviceready",onDeviceReady,false);
```

## □ 指令說明：

在網頁中加入監聽事件,監聽deviceready事件,確認PhoneGap Js 函式庫是否讀取完畢,此API為必需的API

# PhoneGap 函式庫引用測試

```
<html>
<head>
<title>Phonegap Init</title>
<script type="text/javascript" charset="utf-8" src="phonegap.js"></script>
<script type="text/javascript" charset="utf-8">
// 監聽PhoneGap JS初始化
document.addEventListener("deviceready",onDeviceReady,false);

// PhoneGap 已經可以使用
function onDeviceReady() {
    alert("裝置初始化完成");
    //初始化完成後接著執行自己所要跑的函式庫
    //appInit();
}
</script>
</head>
<body>
<h1>PhoneGap 初始化測試</h1>
</body>
</html>
```

# PhoneGap API功能介紹

- ▣ 1.相機使用實作(Camera)
- ▣ 2.網路狀態監控實作(Connection)
- ▣ 3.加速度計實作(Accelerometer)
- ▣ 4.指南針實作(Compass)
- ▣ 5.通訊錄實作(Contacts)
- ▣ 6.取得裝置資訊(Device)
- ▣ 7.事件說明(Events)
- ▣ 8.音效撥放(Media)
- ▣ 9.通知實作(Notification)
- ▣ 10.GPS實作(Geolocation)
- ▣ 11.檔案操作實作(File)
- ▣ 12.資料庫操作實作(Storage)

# 相機實作

## □ camera.getPicture

- ❖ 使用Android拍照功能
- ❖ 取得Android全部相片
- ❖ 取得Android全部相簿資源

# 網路狀態監控實作

## □ connection.type

❖ 取得目前網路狀態

- 4G
- 3G
- 2G
- wifi
- 有線網路



# 加速度計實作

## □ accelerometer.getCurrentAcceleration

- ❖ 取得目前的x,y,z軸的加速度

## □ accelerometer.watchAcceleration

- ❖ 定時監聽x,y,z軸加速度並取得加速度資料

## □ accelerometer.clearWatch

- ❖ 清除監聽事件

# 指南針實作

## □ compass.getCurrentHeading

❖ 取得目前的指南針角度0~359.99

## □ compass.watchHeading

❖ 定時監聽指南針角度資訊

## □ compass.clearWatch

❖ 清除監聽事件

0度=北,90度=東,180度=南,270度=西

# 通訊錄實作

## □ contacts.create

❖ 建立新的聯絡人資訊

## □ contacts.find

❖ 搜尋聯絡人,取得符合條件的聯絡人陣列

## □ contacts.save

❖ 儲存聯絡人

# 取得裝置資訊

▣ device.name

❖ 取得裝置名稱

▣ device.phonegap

❖ 取得Phonegap版本號

▣ device.platform

❖ 取得裝置平台

▣ device.uuid

❖ 取得裝置的唯一UID

▣ device.version

❖ 取得裝置平台的版本資訊

# 事件說明

- ▣ deviceready (裝置準備完成)
- ▣ pause(當App進入背景作業時)
- ▣ resume(當App從背景進入目前操作介面時)
- ▣ online(可連線到網際網路時)
- ▣ offline(無法連線到網際網路時)
- ▣ backbutton(當使用者按下返回鍵)
- ▣ menubutton(當使用者按下選單鍵)
- ▣ searchbutton(當使用者按下搜尋鍵)

# 音效撥放

- `media.getCurrentPosition` : 取得當前秒數
- `media.getDuration` : 取得總長度秒數
- `media.play` : 撥放音樂
- `media.pause` : 暫停音樂
- `media.release` : 釋放音樂資源
- `media.seekTo` : 移動到特定秒數後繼續撥放
- `media.startRecord` : 開始錄音
- `media.stopRecord` : 停止錄音
- `media.stop` : 停止撥放

# 通知實作

## □ notification.alert

❖ 顯示警告通知

## □ notification.confirm

❖ 顯示確認通知

## □ notification.beep

❖ 使用系統音效通知

## □ notification.vibrate

❖ 使用震動通知

# GPS實作

□ geolocation.getCurrentPosition

❖ 取得目前GPS座標資訊

□ geolocation.watchPosition

❖ 定時監聽GPS座標資訊

□ geolocation.clearWatch

❖ 清除監聽事件



# 檔案操作實作

- DirectoryEntry
  - ❖ 建立目錄結構物件,供目錄處理使用
- DirectoryReader
  - ❖ 讀取目錄供處理
- File
  - ❖ 檔案資源物件
- FileEntry
  - ❖ 建立檔案結構物件,供檔案處理使用
- FileReader
  - ❖ 讀取檔案供處理
- FileSystem
  - ❖ 檔案系統資源物件
- FileTransfer
  - ❖ 檔案上傳物件
- FileWriter
  - ❖ 檔案寫入物件
- LocalFileSystem
  - ❖ 本地端檔案系統資源物件

# 資料庫操作實作

## □ openDatabase

- ❖ 建立SQL Lite資料庫物件

## □ SQLTransaction

- ❖ 執行SQL指令

## □ SQLResultSet

- ❖ 取得SQL單筆查詢結果

## □ SQLResultSetList

- ❖ 取得多筆查詢結果

## □ localStorage

- ❖ 使用精簡型本地儲存DB物件, key : value模式