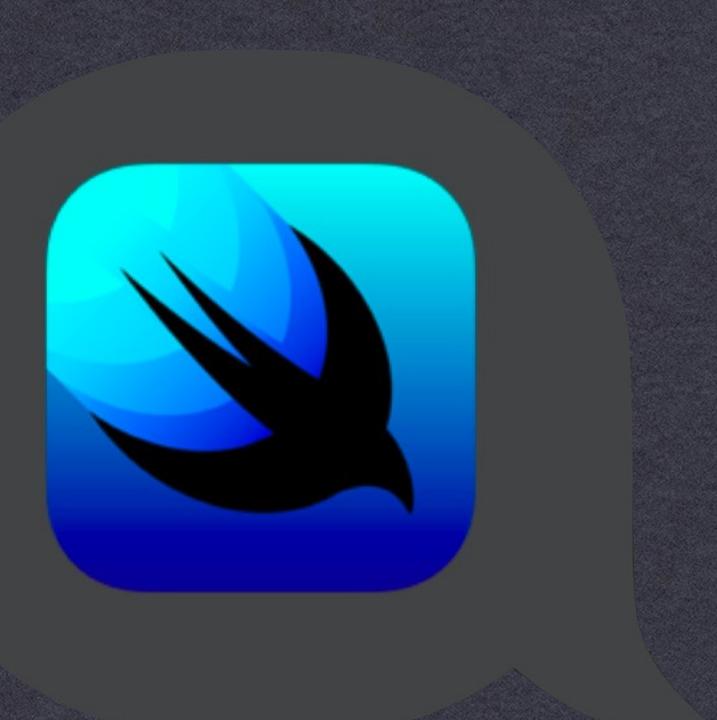


IOS 應用開發期中作業

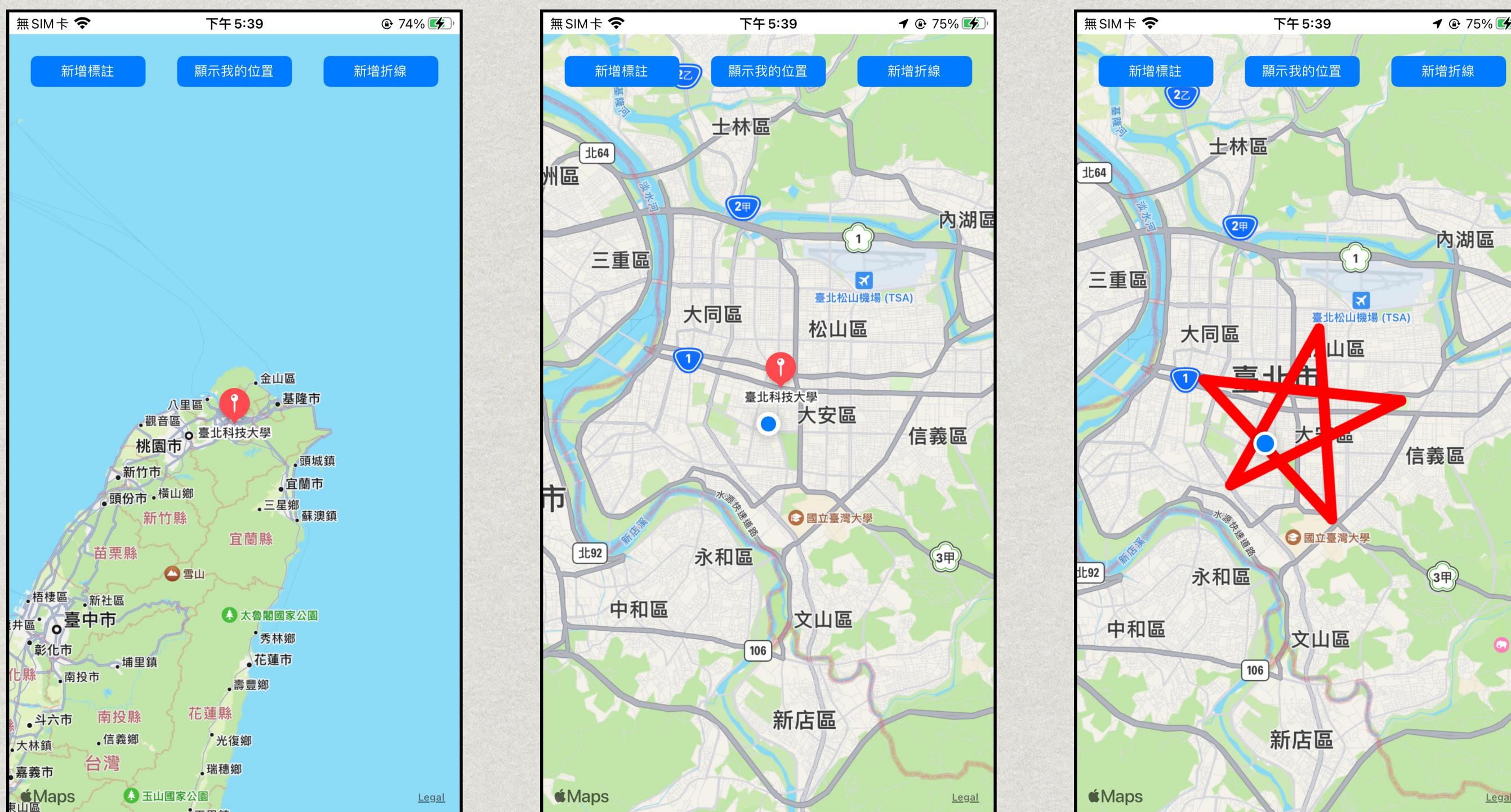
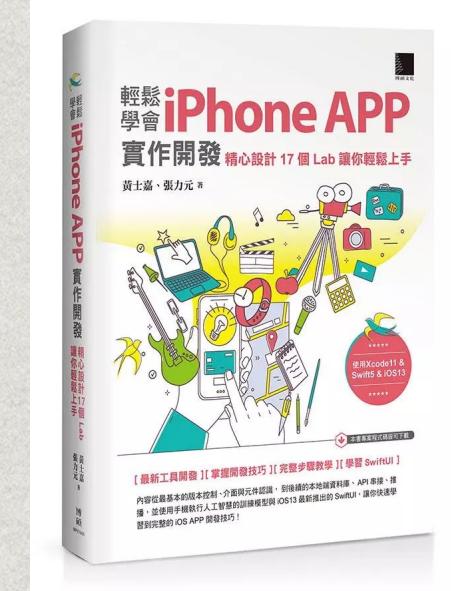
姓名：楊皓麟 112368017

班級：電子碩一



IOS 地圖與相關物件

Ch10 地圖實作練習

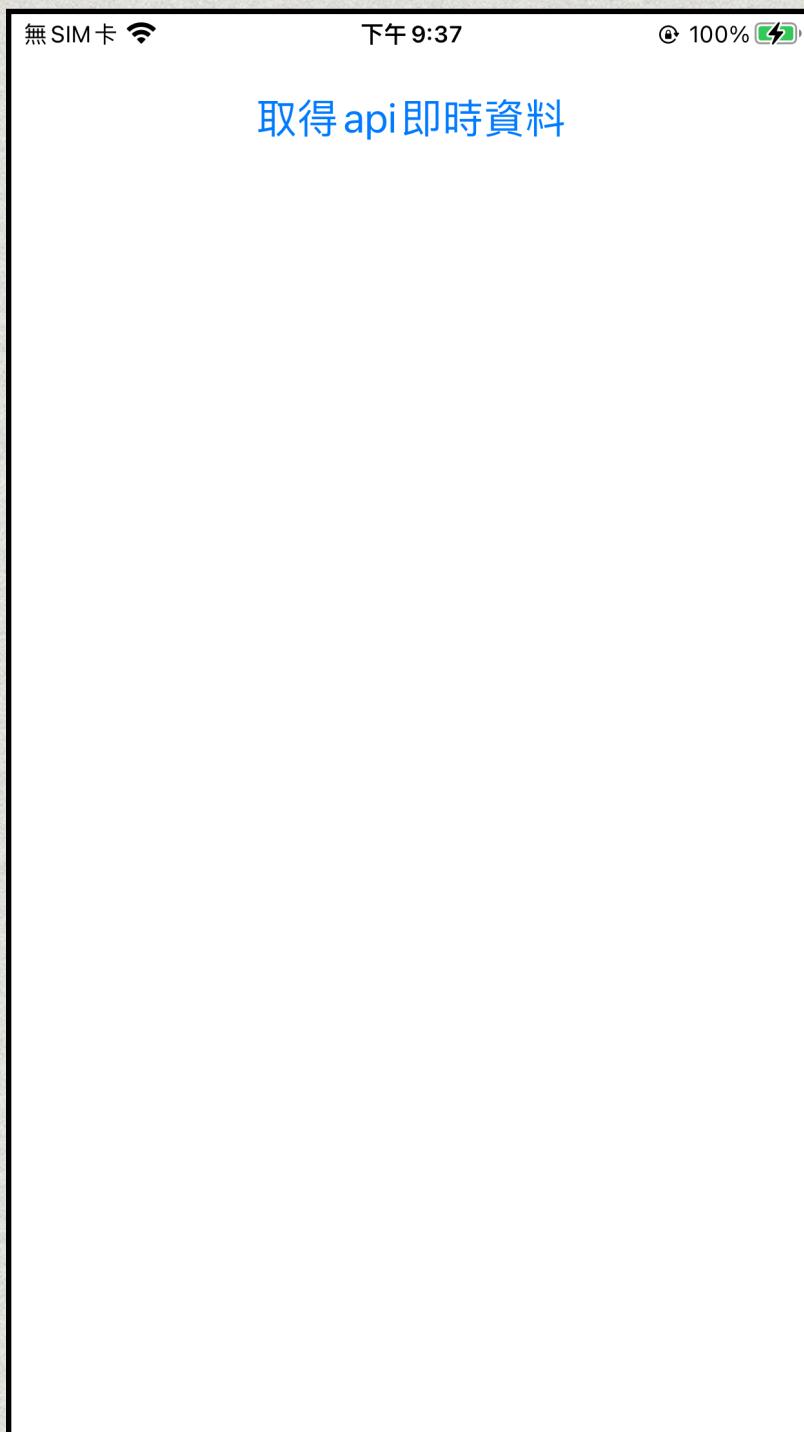


功能：

1. 在畫面上方設定三個按鈕，分別為**新增標註**、**顯示我的位置**、以及**新增折線**
2. 點擊**新增標註**後會在北科大的座標放置大頭針
3. 點擊**顯示我的位置**後會跳轉至當前座標
4. 點擊**新增折線**後會在地圖上根據預先設定好的座標位置放置折線

* IOS API

Ch13 開放API資料實戰

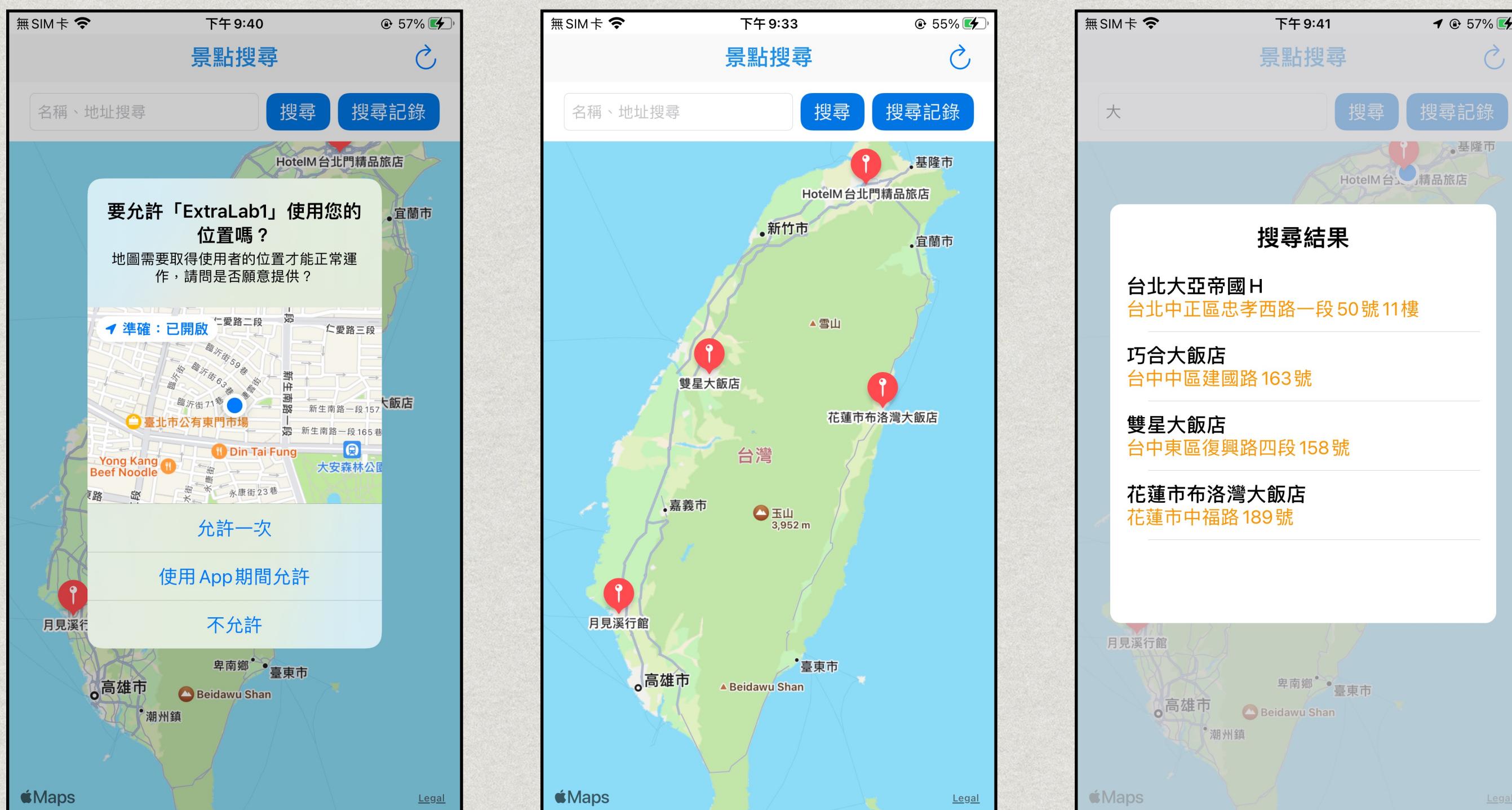


功能：

1. 按下按鈕後，送出Http Get的Request
2. 使用 Codable 將 Server Response 的 JSON 字串轉換為物件
3. 將得到的資料以表格方式呈現

* IOS 飯店搜索

景點搜尋

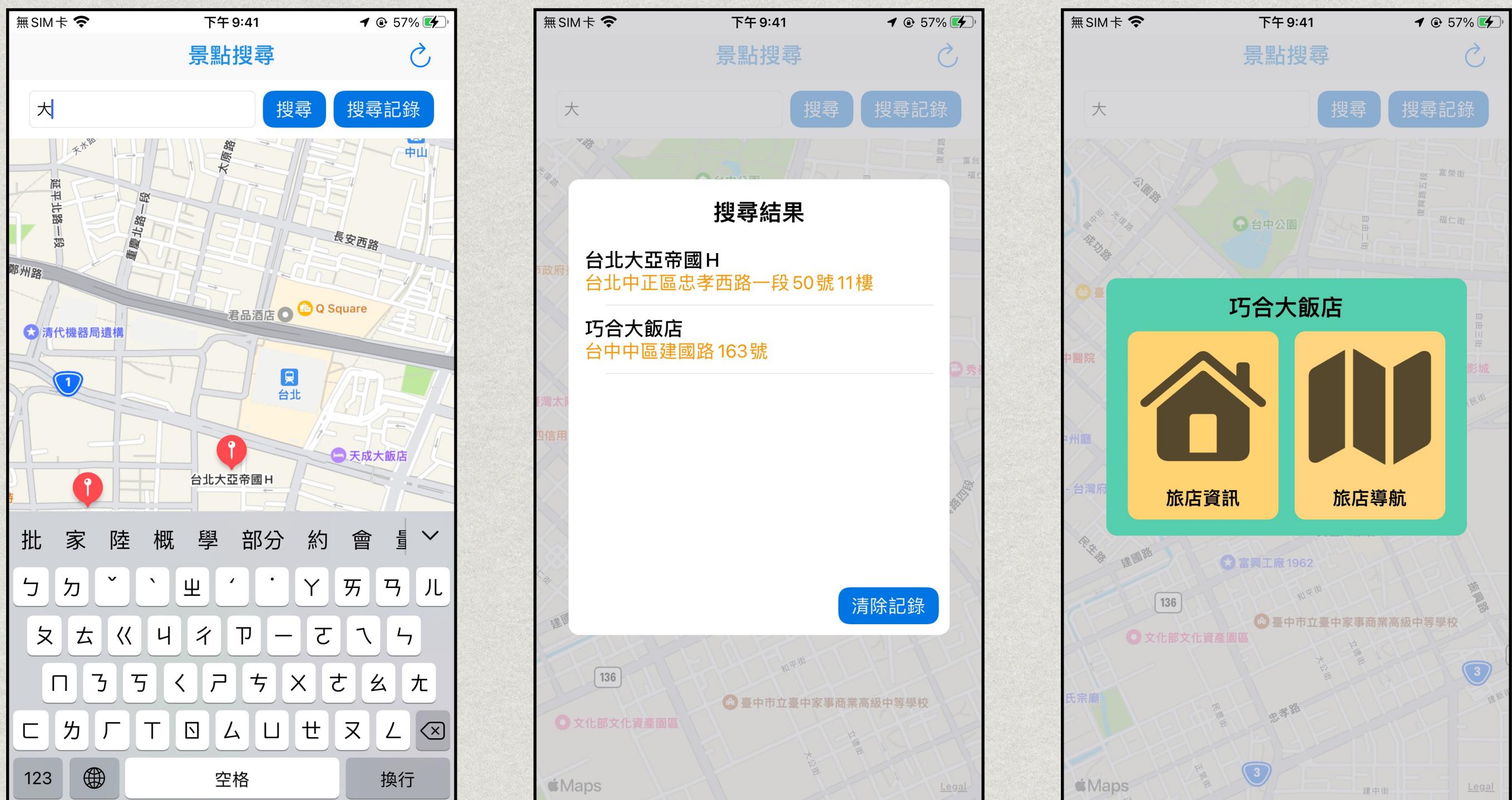


功能：

1. 一開啟 app 就會跳出定位權限請求，並且連接API
2. 接到 API 資料後在地圖上依照座標插上大頭針
3. 點擊右上角的刷新按鈕後會將地圖視角移動到用戶當前位置
4. 在上方文字輸入框可以搜尋地點或地址，點擊搜尋會跳出視窗顯示搜尋相符的結果

* IOS 飯店搜索

景點搜尋

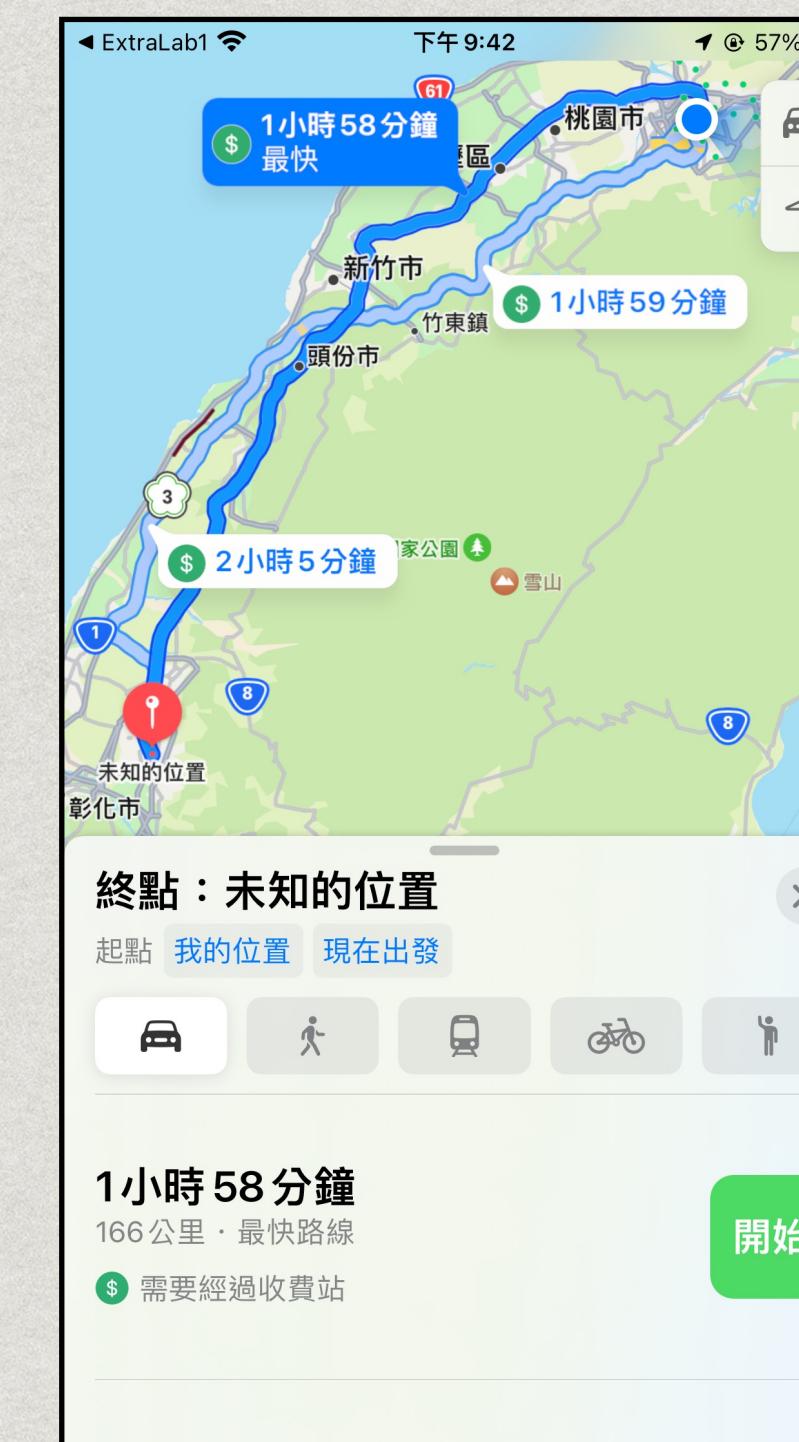


功能：

1. 點擊符合的搜尋結果會將地圖視角跳至該地點
2. 點擊搜尋記錄按鈕可以查看搜尋過的地點
3. 在地圖上點擊大頭針會跳出視窗，可以顯示旅店資訊或導航至該地方

* IOS 飯店搜索

景點搜尋



功能：

1. 選擇旅店資訊後會跳出該地點的地址、星級評分、以及景觀圖
2. 點擊旅店導航會跳轉至IOS內建地圖並導航至該地點

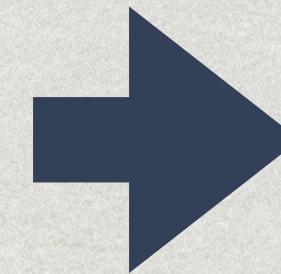
* IOS 飯店搜索

景點搜尋（影片展示）



* 程式改善

```
switch myData?.results.content[myDataNumber].star{
    //根據API資料設定星星為黃色
    case 1:
        star1.tintColor = color
    case 2:
        star1.tintColor = color
        star2.tintColor = color
    case 3:
        star1.tintColor = color
        star2.tintColor = color
        star3.tintColor = color
    case 4:
        star1.tintColor = color
        star2.tintColor = color
        star3.tintColor = color
        star4.tintColor = color
    case 5:
        star1.tintColor = color
        star2.tintColor = color
        star3.tintColor = color
        star4.tintColor = color
        star5.tintColor = color
    default:
        print("error")
}
```

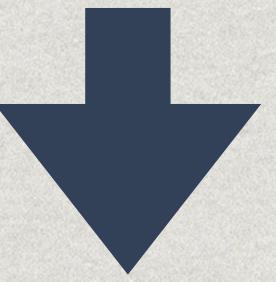


```
func setStar(myDataNumber:Int){
    let basicColor = UIColor(rgb: 0xC6C6C8)
    let color = UIColor(rgb: 0xECC733)
    //設定星星顏色為灰色
    star1.tintColor = basicColor
    star2.tintColor = basicColor
    star3.tintColor = basicColor
    star4.tintColor = basicColor
    star5.tintColor = basicColor
    let starIndex = myData?.results.content[myDataNumber].star ?? 0
    //根據得到的API資料設定星星為黃色
    star1.tintColor = starIndex >= 1 ? color : basicColor
    star2.tintColor = starIndex >= 2 ? color : basicColor
    star3.tintColor = starIndex >= 3 ? color : basicColor
    star4.tintColor = starIndex >= 4 ? color : basicColor
    star5.tintColor = starIndex >= 5 ? color : basicColor
}
```

此功能可以利用最簡單的 if else 或者 select case 達成，不過這樣做較沒有效率，可以利用三元運算式，程式的效率以及可讀性也會增加

* 程式改善

```
//設定當點擊到normalView以外的地方時，關閉此view
@IBAction func tapGesture(_ sender: Any) {
    let touchLocation = (sender as AnyObject).location(in: normalView)
    if normalView.bounds.contains(touchLocation) {
        return // 如果點擊位置在normalView內部，則不執行後續程式碼
    }
    print("觸發TGR")
    dismissViewController()
}
```



```
//設定當點擊到normalView以外的地方時，關閉此view
@IBAction func tapGesture(_ sender: Any) {
    let touchLocation = (sender as AnyObject).location(in: normalView)
    guard !normalView.bounds.contains(touchLocation) else {
        return // 如果點擊位置在normalView內部，則不執行後續程式碼
    }
    print("觸發TGR")
    dismissViewController()
}
```

要習慣將If更改为guard else語法，可以使程式的可讀性增加，也可以實現提早離開的效果，不會浪費時間執行用不到的程式，提高程式效率！

除此之外也需要考慮API資料異常的情況，應該要設定例外情況，以免程式崩潰，造成不好的使用者體驗，



THE END