資訊檢索期末報告–競賽時程APP

組別 : 第七組

組員 : A1045516 蔡湘俊

A1045505 施彥廷

指導老師 : 黃健峰

1. 動機

在整個資訊工程學系的就學路途上，程式競賽可以說和我們是形影不離，又加上其被設定為畢業門檻的條件，所以這個程式競賽對於我們而言就更加的重要了。

那一提到程式競賽，可以說競賽的形式與種類是包羅萬象，每個競賽都有屬於它自己的特點之處，那對於我們學生而言參加最為頻繁的競賽莫過於就是CPE、ITSA、PTC、NCPC、ACM-ICPC這五種程式競賽了，但這時問題就來了，每當我們想報名其中任何一項競賽的時候，我們就必須時時刻刻的前去主辦單位的網站來去密切注意競賽的舉辦日期，在一開始的幾次報名可能不會有什麼感覺，可是等到比了有一陣子的程式競賽之後，其實就會漸漸的發覺到每次都必須去注意競賽的報名日期是一件十分麻煩的事情，所以說我們為了解決這麻煩的查詢過程，我們就想到了網路爬蟲這項技術的存在。



1. 電腦 V.S. 手機

今天不論是在電腦端也好或者是手機端也好，其實這兩個都可以拿來去實作網路爬蟲，可是雖然兩者都有包含網路爬蟲的實作技術，但電腦端的網路爬蟲實作技術不論是在實用性上或者是操作的便利性上都要遠大於手機端，可是為何我們最後還是選擇要去使用手機端來去實作網路爬蟲的原因主要有兩個，第一個，雖然說今天手機端的網路爬蟲技術還不像電腦端的來的那麼的成熟，可是它還是有提供一定基本的網路爬蟲技術，又加上我們此次所做的期末專題並不需要用到多麼進階的網路爬蟲技術，所以說手機端所提供的網路爬蟲技術就足以滿足我們此次期末專題的製作，但這還不是我們最主要選用手機端平台來去實作期末專題的主要原因，更主要的原因其實是第二個，正如上頭動機所提到的今天正是因為每次都必須去到競賽的官網進行查詢的動作，所以我們才想說可以透過網路爬蟲來去幫我們解決這項麻煩，可是今天如果是實作在電腦端上的話，那我們還必須電腦是在開機的狀態下我們才有辦法開啟軟體來去進行查詢的動作，而且雖然說有筆電的存在，可是不管怎麼說在行動上就不是那麼的便利，可是反觀今天實作在手機端的話，那麼其便利性以及行動力可以說都是不在話下，又加上現今手機硬體的強大以及網速的提高，拿來實作網路爬蟲技術可以說是綽綽有餘，這也促使了我們組會想選用手機端來去實作這次期末專題的主要原因。

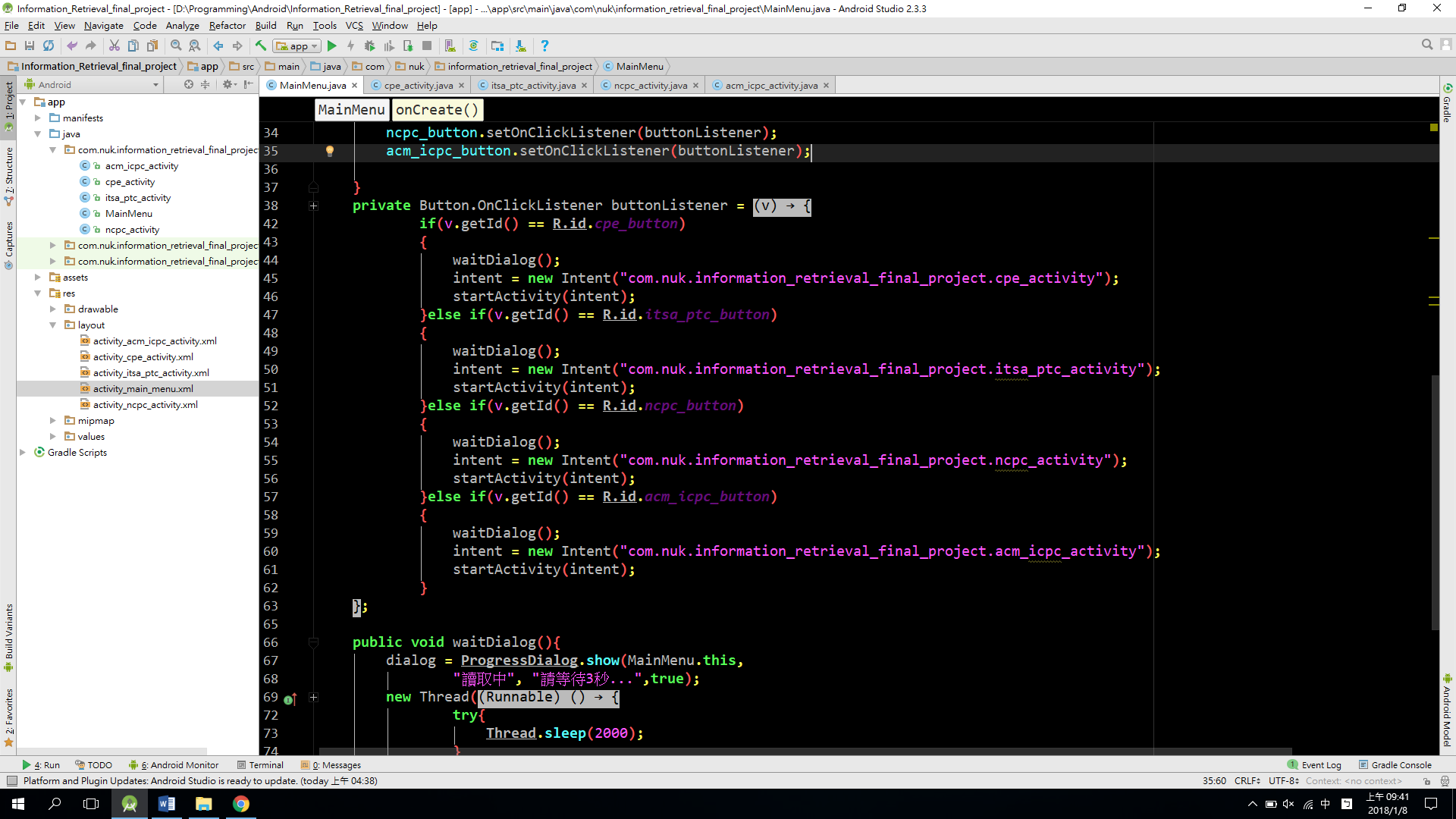


1. 設計與實作

那本次期末專題所實作的APP就是可以用來去即時查詢競賽的報名日期、比賽日期，我們最主要做了4大項競賽分別是 : CPE、ITSA和PTC、NCPC、ACM-ICPC，那下圖這一個部份就是我們的首頁



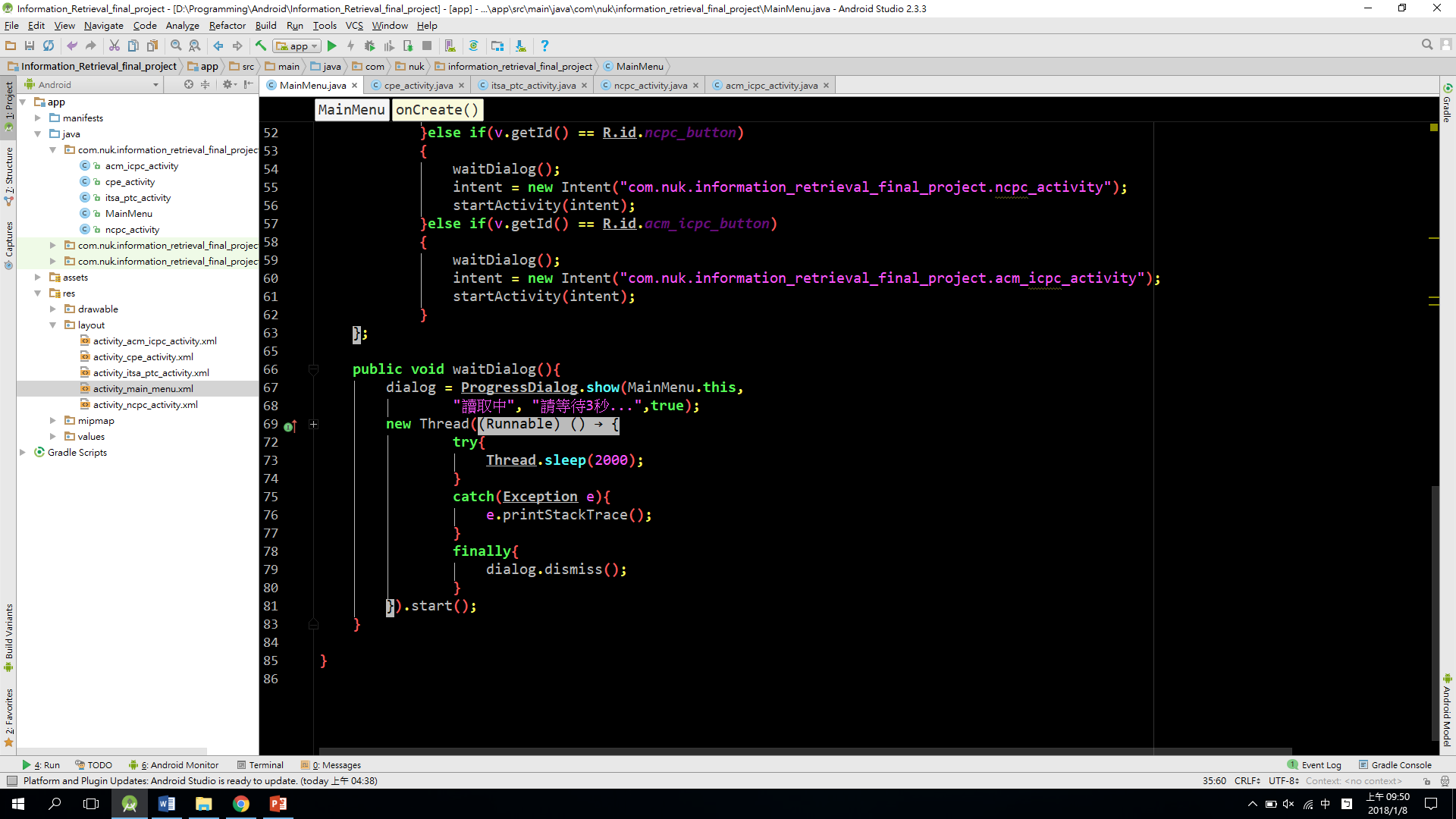
而下方的四個圖示分別是各個競賽的點入按鈕，在這邊就可以去選取自己想要查詢的競賽的報名日期、比賽日期。

那下面這張圖就是上面每一個按鈕的實作內容，每一個按鈕的設計內容是大同小異的，在我按下按鈕當下，會先跳出加載的視窗，之後才導向到該競賽所對應的頁面。 

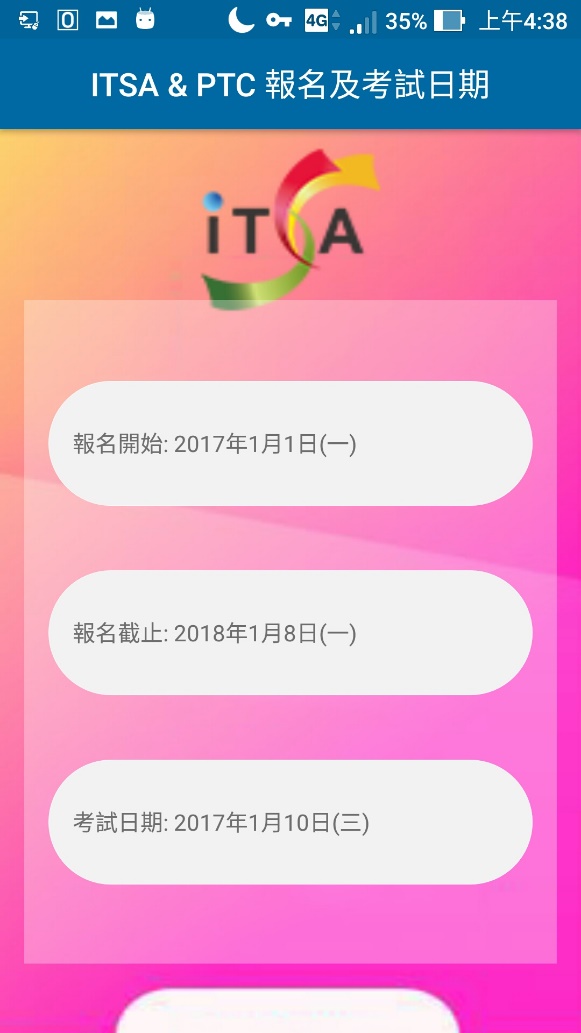
那下面這張圖就是為了讓整個APP的運作更加的寫實，讓其使用者能夠去感覺到說這個APP真的有透過網路來去抓去對應競賽的官網的報名資料，而不是說直接就已經預設好要輸出的結果，所以我們刻意的加入了加載的視窗，但這當然不是我們加入這加載視窗最主要的原因，最主要的原因是在於有些競賽的官網的網頁原始碼是十分的複雜且龐大，所以會造成說有些競賽在按鈕點下去之後要等到幾秒之後才會抓取完成，那在還在抓取的這段時間如果使用者再去點按競賽的按鈕的話就會導致整個APP馬上當掉，更甚至會使整個APP直接CRASH掉，那多加入了這個加載的畫面就能夠有效的避免這樣的問題的發生。



那下面這張圖就是加載畫面的實作內容，那我們之所以會把它設為等待3秒的緣故就在於說經過我們反覆的測試之後，我們可以去確保說在經過這三秒之後其我們所要抓取的內容一定能夠抓取完畢，這樣子就能使我們的運行結果不會發生錯誤。

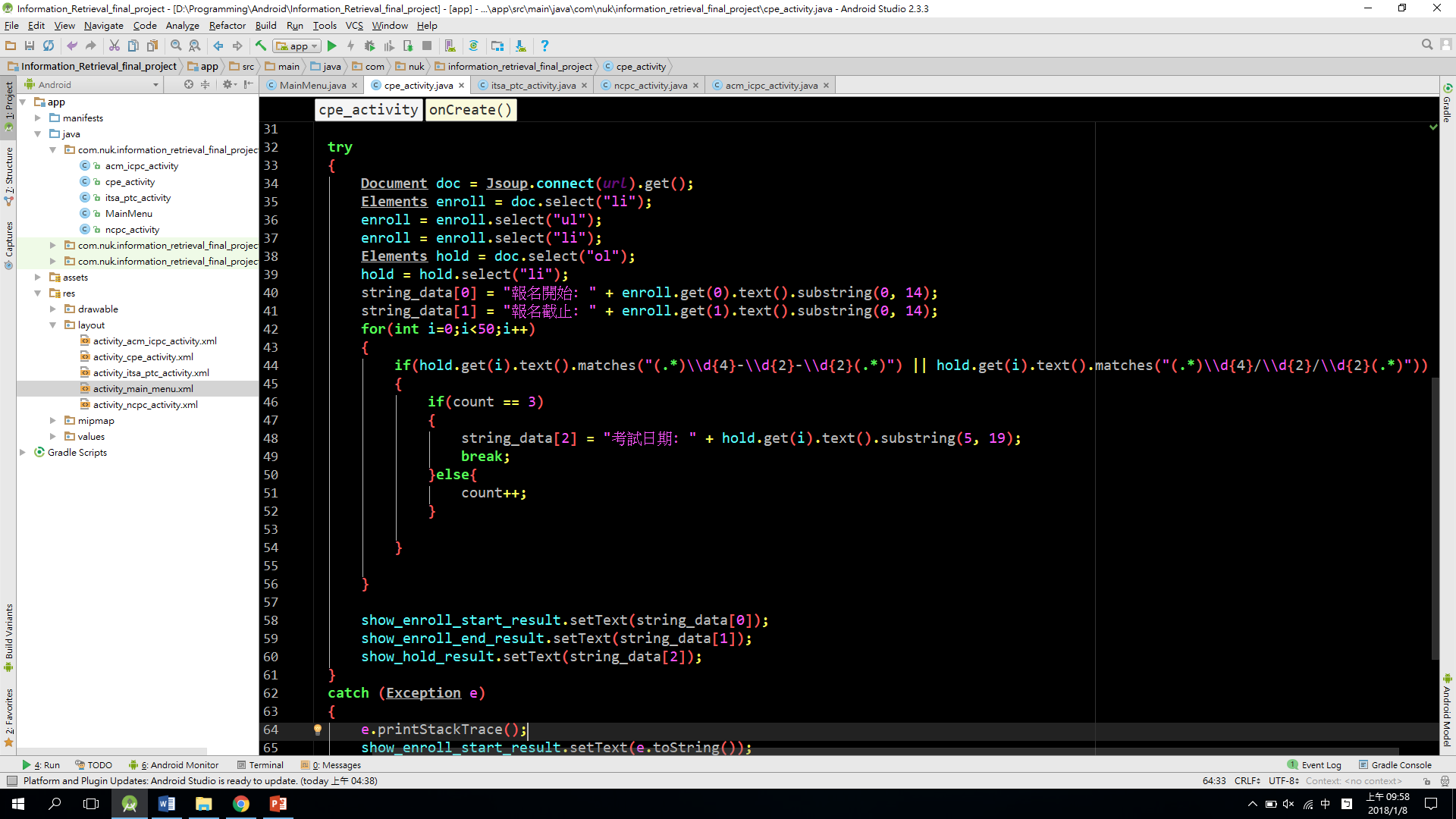


那下面這四張圖分別就是顯示CPE、ITSA和PTC、NCPC、ACM-ICPC的報名日期與比賽日期的時間點出來

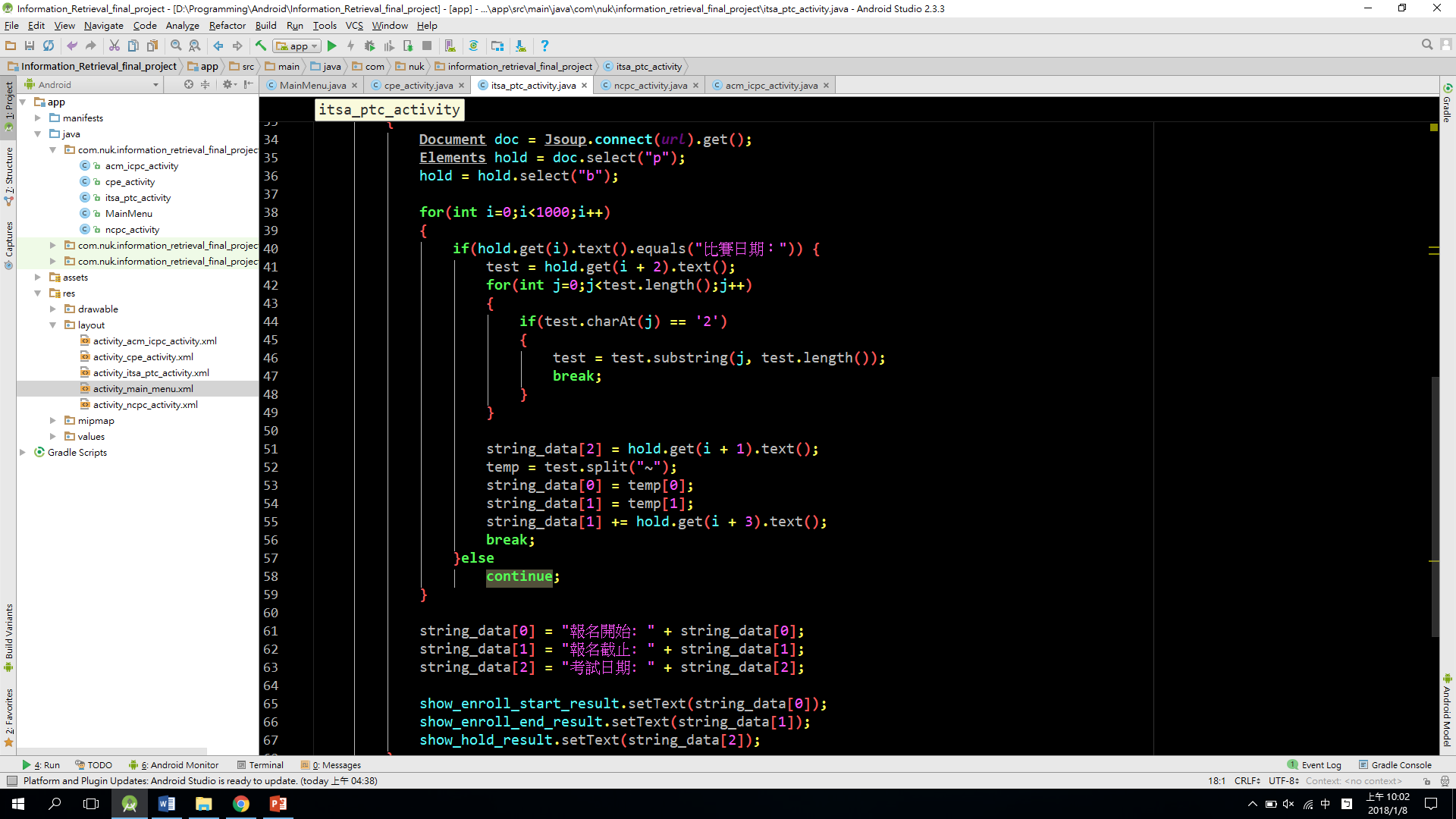
  

那這四個頁面的實作方式其實都大同小異，它們都分別使用到了Java裡頭的Jsoup、Document和Element這三個module，而Jsoup這一個module可以說是Python的Requests和BeautifulSoup這兩個module的結合體，它既能和目標網頁進行連結的動作來去將該網頁的原始碼給抓取下來，也會利用網頁原始碼裡頭的HTML的Tag來去進行建樹的動作，也就是將原本雜亂且無規則的網頁原始碼來去建成一個有組織的樹狀結構，而這些Tag下所存放的內容也會一併的放在這個Tag的底下，來去當作這一個Tag的子類別，然後再將剛剛建出來的樹狀結構給放到Document這個類別來去進行管理與內容提取的動作，也就是我們可以來去取出特定Tag裡頭所存放的內容，之後再利用Element這個類別將提取出來的內容給儲存起來以便作為之後資料的存取與使用，那藉由這三者的互相搭配就能夠將每個競賽網頁的報名日期以及比賽日期給抓取出來。

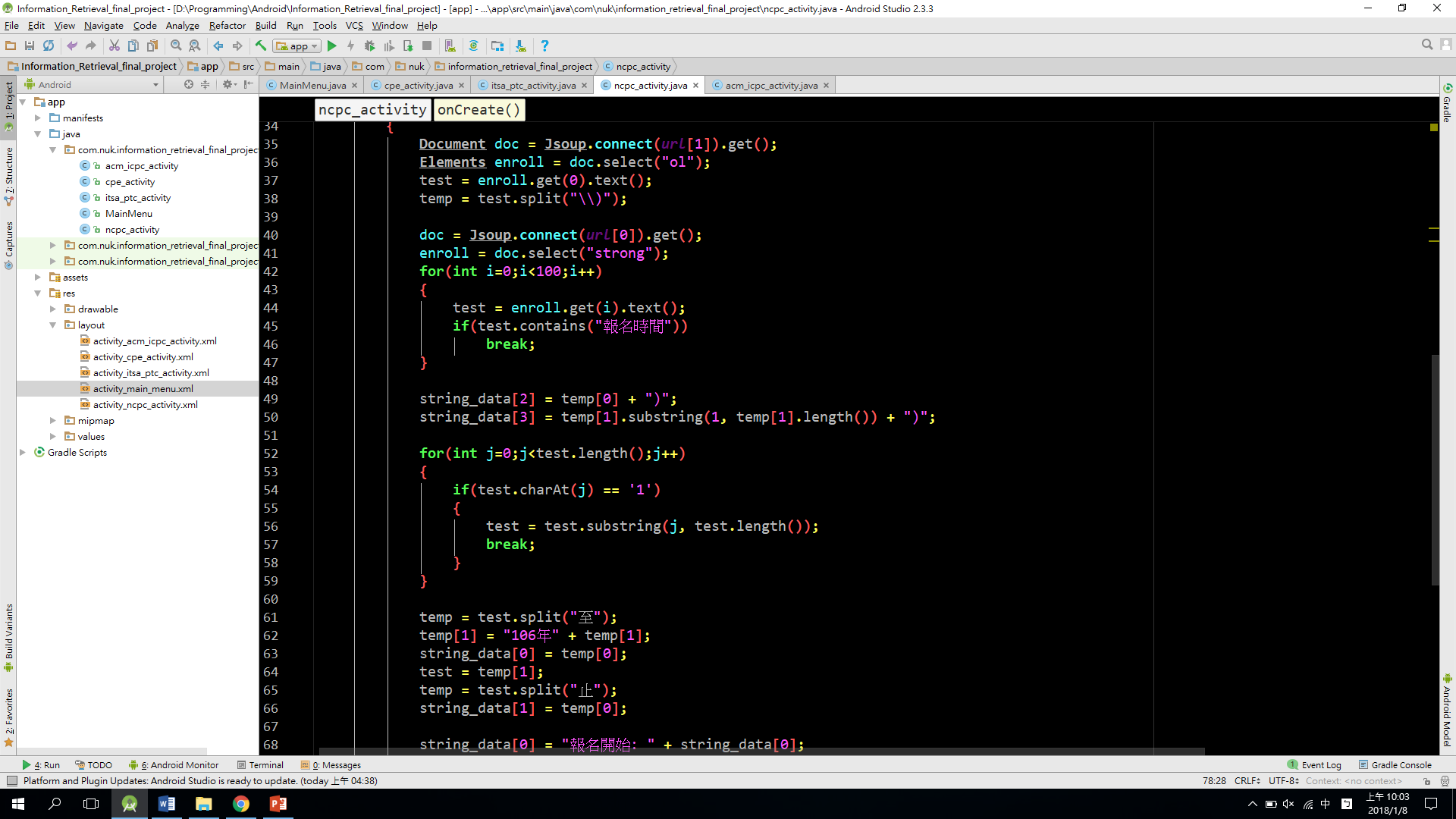
下面這四張圖分別就是CPE、ITSA和PTC、NCPC、ACM-ICPC的實作內容，那從下面這四張圖我們可以發覺到說其實它們的實作內容都是大同小異的，除了因為每個網頁的原始碼不相同，並且架構也都不相同，所以在字串的處理上是有著蠻大的差異，但這四個內容的實作絕對和Java裡頭的Jsoup、Document和Element這三個module脫離不了任何的關係，因為他們就是在APP上實作網路爬蟲的核心技術，那其中ITSA和PTC之所以會放在一起的原因就在於說因為他們是同一個主辦單位，所以他們的報名日期以及比賽日期都是一模一樣的，也因為如此，這兩個程式競賽才能夠被寫在一起。

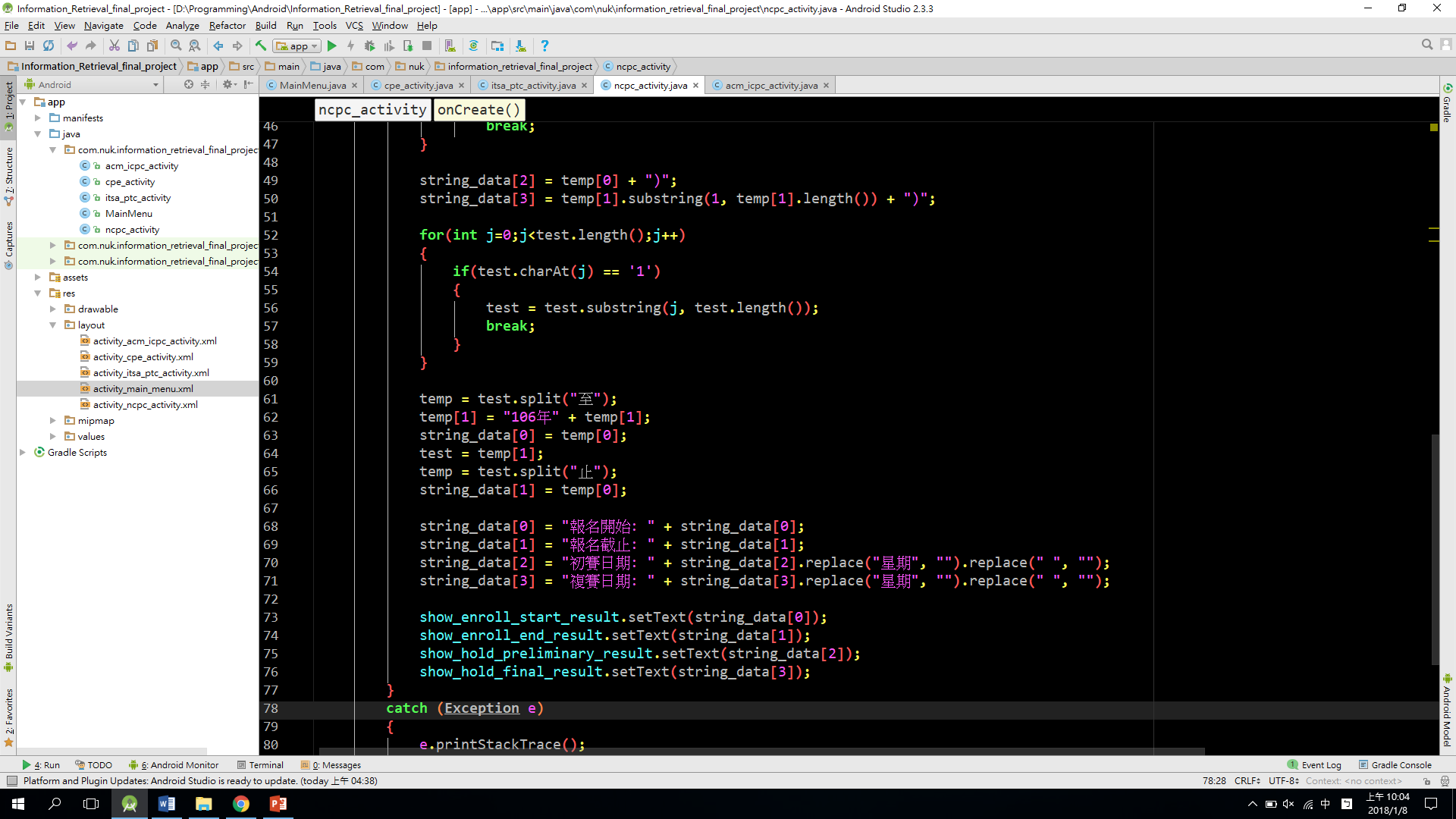
CPE

ITSA 和 PTC

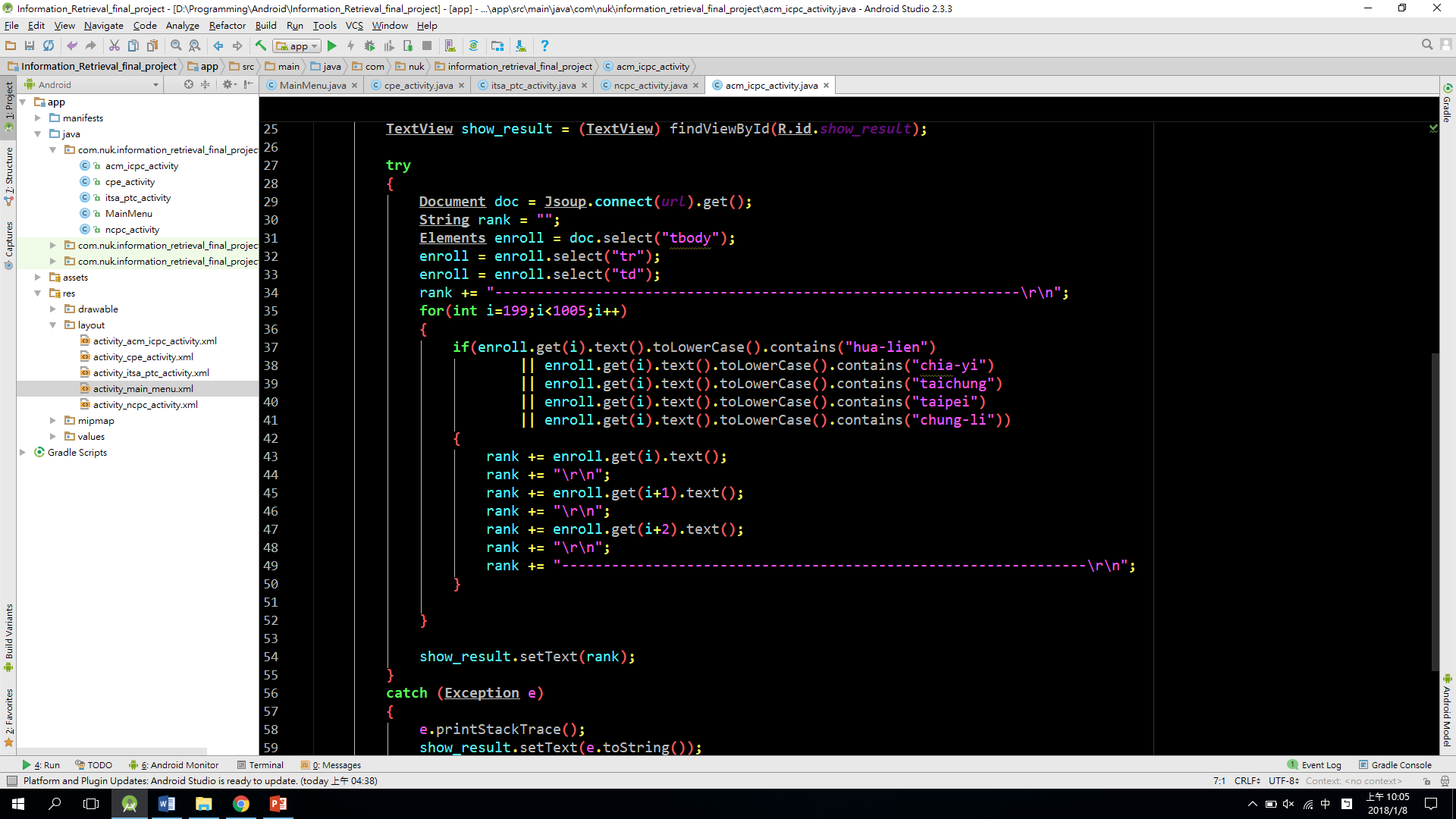


NCPC





ACM-ICPC



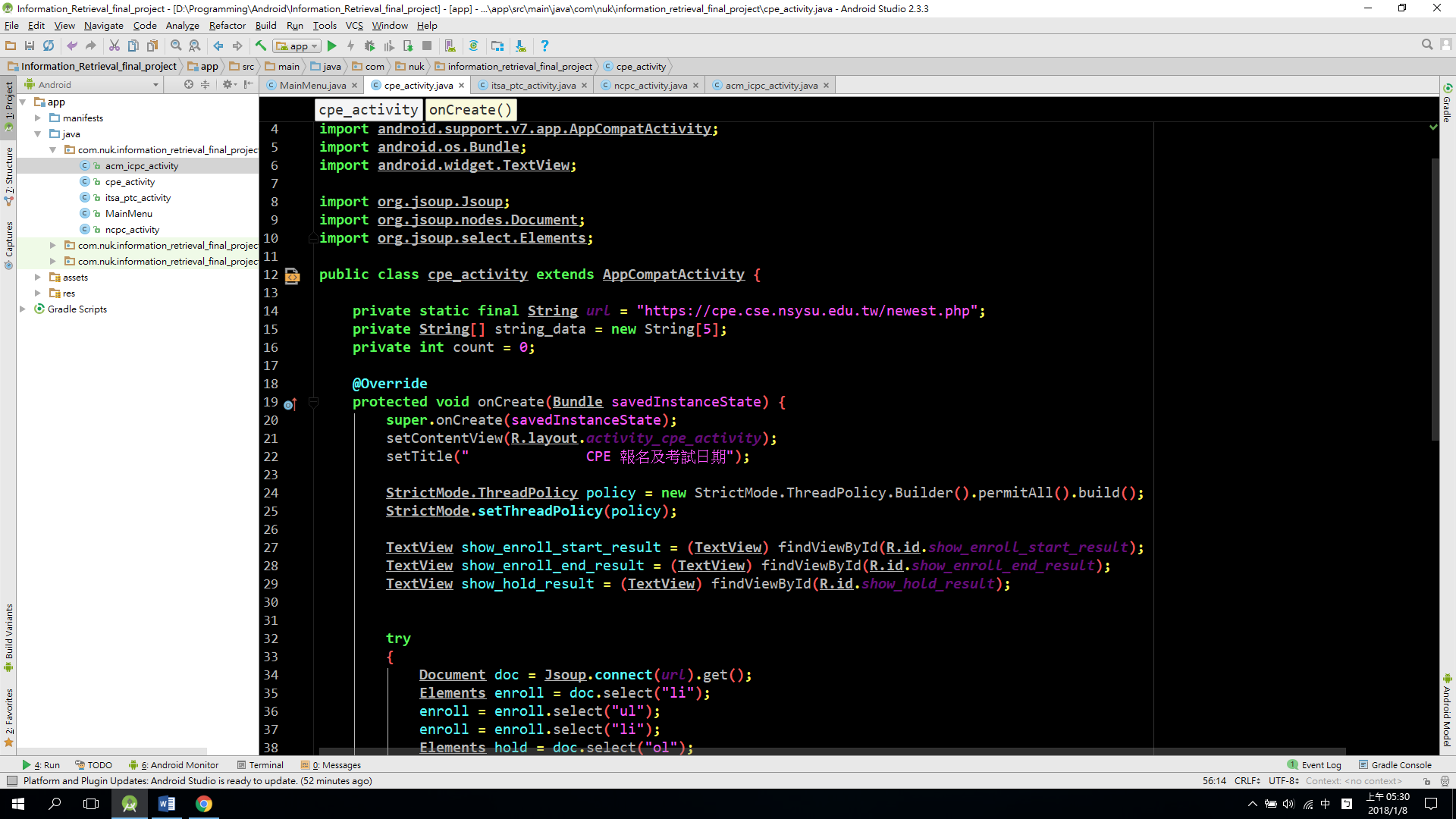
還有就是ACM-ICPC的部分，這個競賽因為它是國際性的比賽，所以它是以地區的方式下去做區分，並且在時間點上每一區都稍微不太一樣，那這時如果只要去抓取到台灣這個地方所舉辦的ACM-ICPC的比賽，那就必須以舉辦的地方作為資料抓取的單位，那因為它在台灣每年舉辦的地方都不大一樣，可是在經過我們詳細的分析之後，我們發覺到舉辦的地方是有趨於循環的情況發生，那我們分析出來的最主要的主辦地方分別有花蓮、嘉義、台中、台北、中壢這五個地方，所以我們就以這幾個地方作為篩選的條件，這樣子就能夠將ACM-ICPC在台灣所舉辦的比賽的地點以及時間全部都給抓取出來，所以相對其它競賽網頁而言，ACM-ICPC的競賽網頁是比較難去提取出競賽的訊息，所以在要實作這一個部分的時候比其它三個部分所花的時間還要來的多。

1. 實作過程中所遭遇的困難

其實在做這一份專案的時候也不是完全都是順順利利的，在過程當中我們也是遇上了許多的困難，分別如下 :

* 1. 網路流的錯誤

在一開始要去實作爬蟲的時候，當我們嘗試的去把網頁的內容給他抓取下來的時候，其都一直顯示出網路流的錯誤，那在當下我們原本以為是我們沒有給APP開啟存取網路的權限，可是在經過一番的檢查之後，我們發覺到說所有應該開啟的權限全部都已經開啟了，可是同樣的網路流的錯誤依然是不斷的發生，而且一開始在網路查找這項錯誤的時候，所有網友所提供的方法我們都試過了，可是同樣的網路流錯誤依然是不斷的發生，這促使當下的我們也開始變得不知所措了，更甚至是萌生了想要換題目的念頭，但好加在我們並沒有去放棄這一項專題的實作，在某一天我們心想再去試試看找不找的到解決的辦法，結果總算是皇天不負苦心人，我們總算找到了解決的方法，就是在程式碼中額外加入下面這兩行



那之所以會產生網路流錯誤的原因就是因為Google它們基於安全性的考量，為了避免使用者一下子對網頁傳輸過多的request，造成了網路壅塞的情況發生，所以一旦在APP裡頭有任何大量的對網頁產生request的情況發生就會被阻擋下來，那加入了上面這兩行之後就可以避免說在對網頁傳送大量的request的時候被阻擋下來的情況發生。

* 1. 報名日期及考試日期的解析

關於這一部份我們原本想說會是最沒有問題的，可是卻花了我們一堆的時間在處理這一部份的解析，那之所以會產生像這樣的狀況的原因就在於說有些網頁在呈現報名日期與考試日期的時候，它們不是放在同一個HTML的Tag底下，而是把這些報名日期以及考試日期的資訊給它拆開來，更甚至是把它拆的極度複雜，導致我們要去抓取的時候都一直抓取不到我們想要的內容，可是幸好在經過一系列的嘗試與修改之後，總算是把我們想提取出來的內容給提取出來了，那這是第一個問題點。

那第二個問題點是全形與半形，我們在抓取內容的過程當中有幾次我們都發覺到說明明所編寫的程式碼並沒有任何的問題，可是抓取出來的內容居然會有遺漏，那這讓當下的我們是更加不知問題出在哪裡了，原本想說會不會總字元數我們是有數錯的，可是在經過多次的驗算之後都沒有發現任何算錯字元總數的現象，可是就是會漏掉一些內容，直到我們再去觀察該競賽官網的網頁原始碼的時候我們才發覺到說是全形和半形在那搞鬼，有些競賽網站的字型是十分的統一，所以在抓取的過程當中十分的容易，都沒有發生任何奇怪的問題，但有些競賽網站是十分的奇怪，它整個字的組成是由全形以及半形兩個所組合而成的，但也因為這個緣故，才會導致說我們每次抓取出來的內容都是有遺漏的，那到後我們解決的方法就是將所有我們要的內容裡頭它是全形字的部分先給統計出來，之後再利用判斷條件來去將這幾個比較特殊的字給它做一些額外的處理，經過這樣一系列的處理之後，才使得我們最終抓取出來的結果是完整的且沒有任何的遺漏。

1. 結論

網路爬蟲的技術在現如今可以說已經成為了一個不可或缺的技術，我們處處都可以看到網路爬蟲這項技術的應用與身影，並且網路爬蟲這項技術也是變得越來越成熟，而這項技術的成熟也促使我們所能爬到的內容是越來越多，而且又加上手機的蓬勃發展，這促使了網路爬蟲與手機端產生了新的應用面向，而且假如能夠將網路爬蟲完全活用於手機APP的話，那麼所帶來的效益可以說是十分的可觀，雖然目前手機端的網路爬蟲技術不像電腦端一樣有著那麼成熟的技術，可是我們相信隨著時間的推演，手機端的網路爬蟲技術將會越來越成熟，或許在未來的某一天，網路爬蟲這項技術變成了手機APP不可或缺的功能也說不定。

但當一項技術十分的強大的時候，那麼勢必所帶來的就是更加可怕的危害，這項技術被用在好的地方勢必可以為我們帶來極大的效益，可是如果是被用在壞的地方的話，那麼對網頁經營者所造成的傷害將會十分的巨大，一項技術的進步其實都不為了別的，只為了能夠為使用者帶來更多的便利性，而不是為了拿來當成做壞事的工具，作為技術的使用者，我們應該要將技術導向更光明的道路，並且讓這項技術可以造福更多的人、事、物，這才是技術一直不斷的進步真正的意義。

那在這份專題的製作過程當中我們也深深的感受到我們技術上的不足，其實我們在過程當中遇到的問題其實都是十分簡單的問題，可是就只因為我們經驗的不足，又加上是第一次遇到像這樣的問題，導致我們是一度的感到徬徨並且不知所措，但雖然說在製作的過程當中遭遇了許多的困難，可是這也讓我們在過程當中獲得了許多的新知，這些新知都是課堂上所不會提到的，假如今天沒有去做過這個專題，或許我們至今還是不知道有這樣的問題以及錯誤存在著，這也讓我們深深的感受到理論固然是重要的，可是如果空有理論卻實作不出任何一項相關事物出來的話，那麼就會使得這些理論淪落成無用武之地的情況發生，而且這對於我們而言也是一個很大的警惕，假如今天沒有這項期末專題的存在，我們或許都還一直沉溺於自己什麼都做得出來的幻覺當中，所以多虧了這一次的專題，才能夠讓我們有這麼大的進步與成長。

1. 參考資料
   1. CPE

網址 : <https://cpe.cse.nsysu.edu.tw/newest.php>

* 1. ITSA和PTC

網址 : <https://sites.google.com/site/itsancku/home>

* 1. NCPC

網址 : <http://ncpc.nsysu.edu.tw/files/11-1351-99.php?Lang=zh-tw>

* 1. ACM-ICPC

網址 : <https://icpc.baylor.edu/regionals/upcoming>

* 1. 《Android》『Jsoup』- 如何使用 Jsoup 取得 HTML 網頁上的資訊

網址 : <https://xnfood.com.tw/jsoup/>