ITIS/ITCS 4180/5180 Mobile Application Development
Homework 6

Basic Instructions

1. In every file submitted you MUST place the following comments:
   - Assignment #.
   - File Name.
   - Full name of all students in your group.
2. Each group should submit only one assignment. Only the group leader is supposed to submit the assignment on behalf of all the other group members.
3. Your assignment will be graded for functional requirements and efficiency of your submitted solution. You will lose points if your code is not efficient or does unnecessary processing or blocks the UI threads.
4. Please download the support files provided with this assignment and use them when implementing your project
5. Export your Android project and create zip file which includes all the project folder and any required libraries.
6. Submission details:
   - The file name should follow the following format: **Group#_HW06.zip**
7. **Failure to follow the above instructions will result in point deductions.**

Homework 6 (100 points)

In this assignment you will get familiar with SQLite operations, RecyclerView and PreferenceActivity. You will build a weather app with a different look and features. You will learn how to use Preference Activity to add settings or preferences. You will also learn how to build RecyclerView and display items in a list using it.

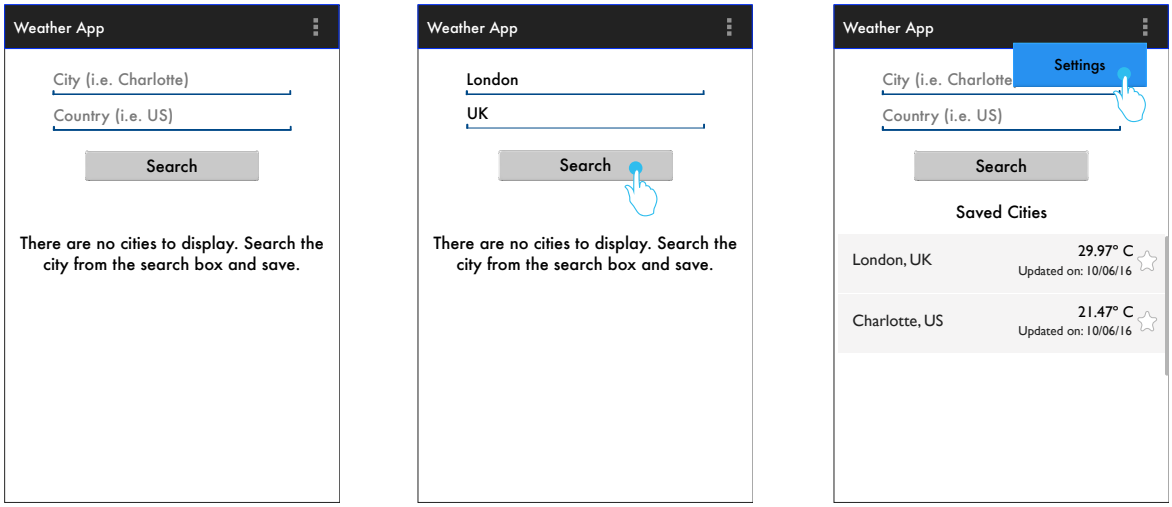
Initial Setup and API Description

You should use Open Weather Map API (https://openweathermap.org/api) for loading the weather information. The API of interest is the 5day/3 hour forecast API (https://openweathermap.org/forecast5) which is based on the city name and country initials. You need to create an account in order to create an API Key. Follow the steps given below:

1. Go to https://openweathermap.org/appid and Sign Up.
2. Fill up Username, Email and Password to create your account
3. You will receive an email confirming your account's been created.
4. Sign in with your credentials.
5. Generate a Key for your future use, or use the default key they provide.

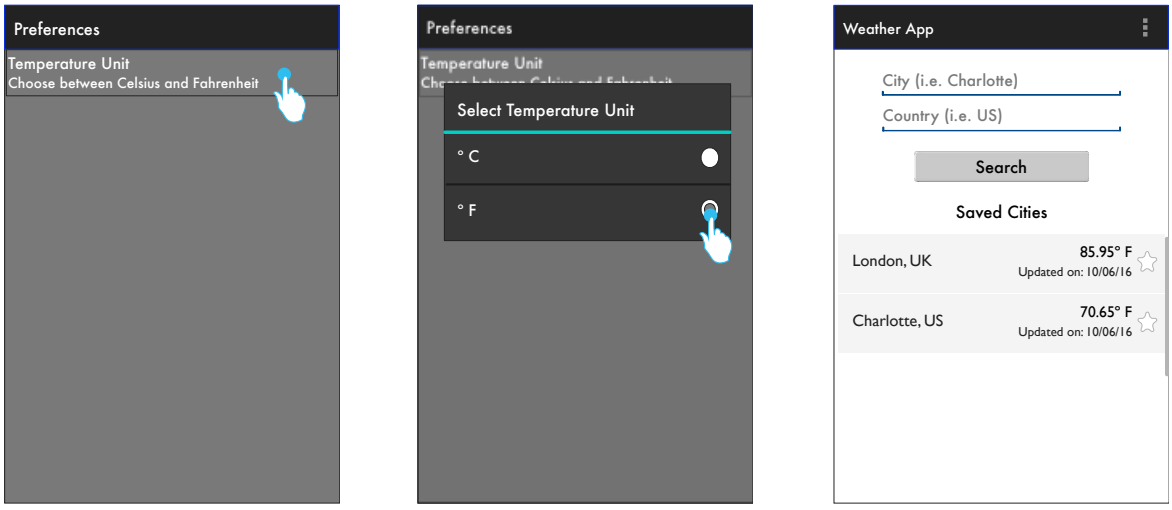
Now that you have your key, go ahead and Subscribe the Free Current weather subscription (http://openweathermap.org/price). Again, we will be using 5days/3hours forecast API.

API Call: The API call format URL will be: api.openweathermap.org/data/2.5/forecast?q={city name},{country code}. For example, the call for London, UK will be **api.openweathermap.org/data/2.5/forecast?q=London,UK** . For San Francisco it will be: **api.openweathermap.org/data/2.5/forecast?q=San_Francisco,US**



(a) Main Activity without any added cities (b) Search city for weather (c) Menu button named "Settings"

Figure 1, Main Activity



(a) Preference activity for changing Temperature Unit

(b) Option for Celsius and Fahrenheit

(c) Changed Units from ºC to ºF

Figure 2, Main Activity

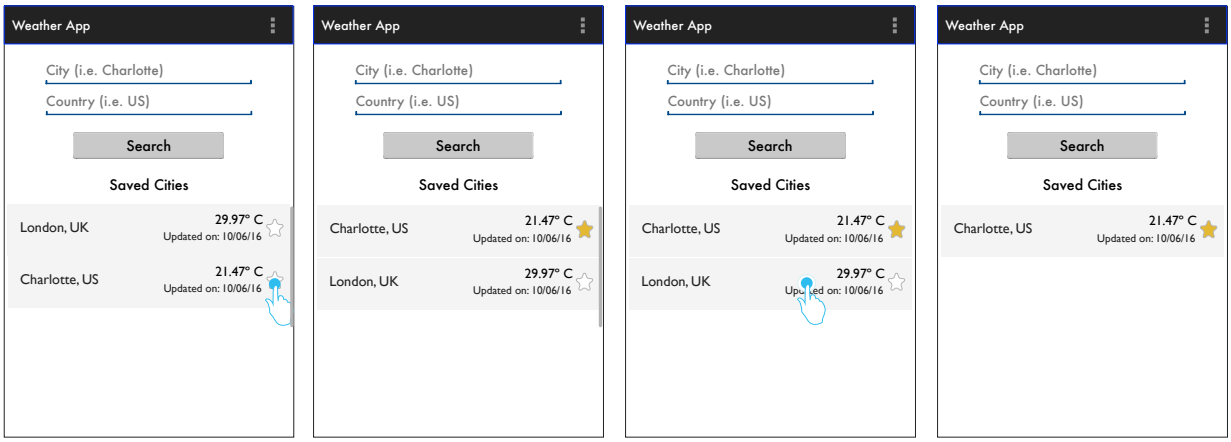
Part 1: Main Activity (40 points)

The activity UI should match the UI presented in Figures 1 and 2. The requirements are given below.

1. The activity should initially display two Edit Texts to enable users to give inputs of City name and Country name (Figure 1) along with a Search button.
2. User should provide the City name and Country as inputs and press Search button. Pressing Search button should start the CityWeather Activity.
3. Create the SQLite database with table **Cities** with columns: cityname, country, temperature, favorite. This table should be used to store the perviously saved cities.
4. This activity should also include a list of previously saved cities in SQLite database. Each item must display the City name, Country name, Temperature when saved, update date and a star button. The star button will act as the defining sign for the favorites. Initially the start should be grey, when the user taps the grey star it should mark the stored city as a favorite and should change the star to gold. Use “star_gray.png” and “star_gold.png” from the Resources.
5. If there are no stored cities in the database then show the TextView displaying, “There are no cities to display. Search the city from the search box and save.” as shown in Figure 1.
6. The List of saved cities should be displayed using **RecyclerView** (https://developer.android.com/reference/android/support/v7/widget/RecyclerView.html). You should use LinearLayoutManager to display the list as a vertical scrolling list. You can find how to use RecyclerView here: https://guides.codepath.com/android/using-the-recyclerview
7. There should be one Menu button in the Main Activity. The menu item is “Settings.” Clicking on the Settings menu item should start a PreferenceActivity (Figure 2). In the preference activity you should design Preference for temperature units. User can change the temperature from Celsius to Fahrenheit or vice versa. Figure 2(c) shows the changed temperature from Celsius to Fahrenheit. **You must maintain the preferences in shared preferences and use them in the rest of the activities.**

The details of PreferenceActivity will be found at : https://developer.android.com/guide/topics/ui/settings.html

8. A Toast message should confirm that the "Temperature Unit has been changed to ºC to ºF" or vice versa.
9. If a user press a gray star, then you should mark the City as favorite and change the favorite field in database as **TRUE**. You should change the color of the star button to Golden. Finally, you should put the favorite cities upwards at the top of the RecyclerView (Figure 3).
10. Long press on any of the items in the RecyclerView should delete it from both the list and database.



(a) Tap on star to Add to Favorites

(b) Added to Favorites and Sorted Up

(c) Long Click to delete a city

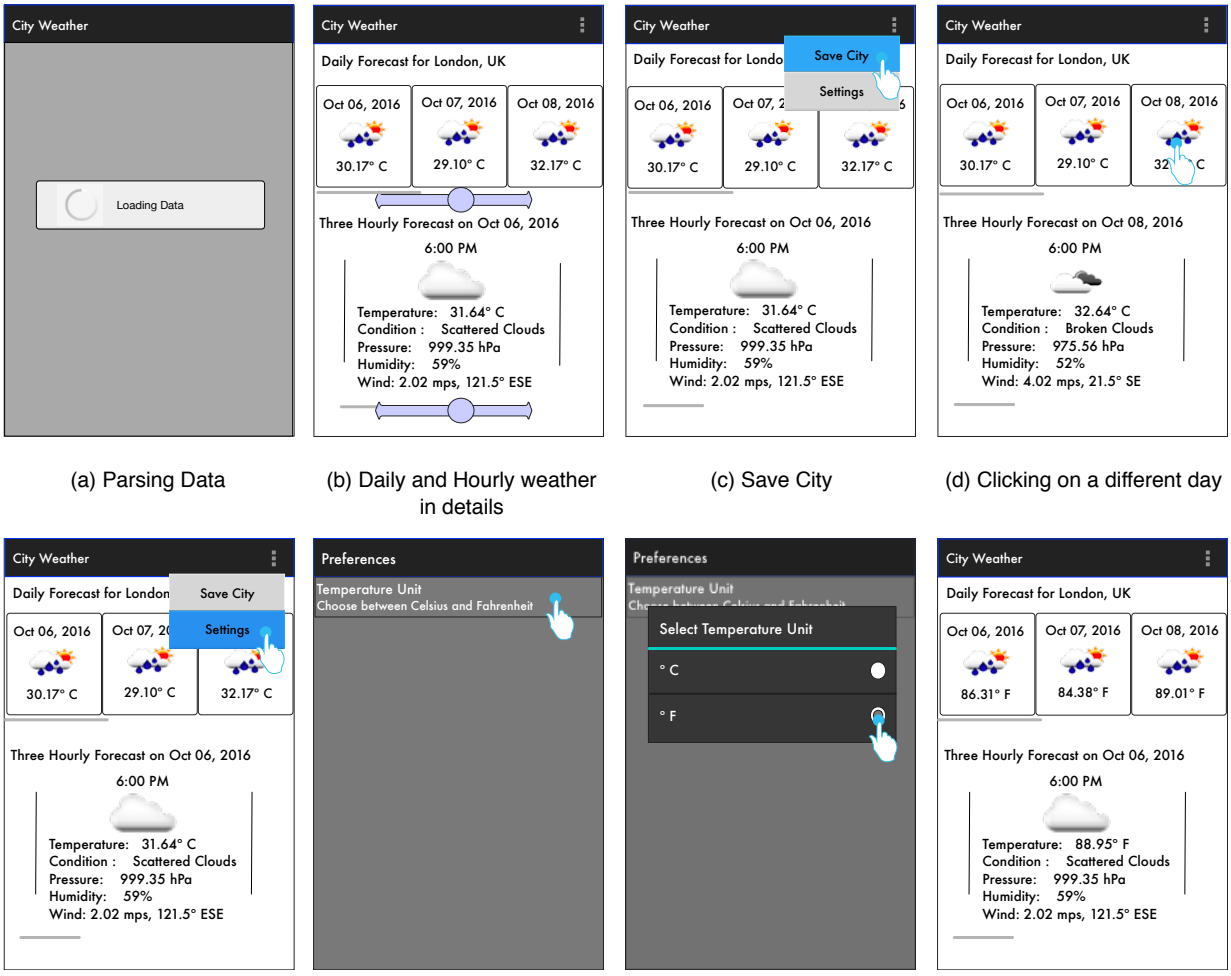
(d) city deleted

Figure 3, Main Activity

Part 2: CityWeather Activity (60 points)

Clicking on Search button from the Main Activity should start CityWeather Activity. This activity should display the detailed Weather Forecast for the searched City. The requirements are as follows:

1. When Search button is pressed you should start a Progress dialogue showing parsing progress as shown in Figure 4 (a).
2. An API call should be triggered using the provided user input. The API call should return the data for consecutive 5days starting from the current day. It will return 3 hourly data (8 weather forecasts / day roughly) per day.
3. The API call example can be: http://api.openweathermap.org/data/2.5/forecast?q=London,UK&mode=xml&appid={API_Key}. Change the mode to mode=json for JSON.
4. Parse the data using any JSON or XML parser. You will find 40 forecast items each having 3hourly data. As you are not always calling the API at midnight (12:00 AM), you may get 6 dates, which is OK. You should use Picasso library to load the images.
5. At the top of the layout, display the text indicating the city and country as "Daily Forecast for {City},{Country}" (Figure 4 (b)).



(a) Parsing Data (b) Daily and Hourly weather in details (c) Save City (d) Clicking on a different day

(e) Settings (f) Preference Activity (g) ºC to ºF (h) Unit changed

Figure 4, CityWeather Activity

6. Then you should display a Horizontal RecyclerView of items, each including the summary of Weather on that particular day. The RecyclerView should show 3 items on screen at once, and should be scrollable horizontally to display the remaining items, see Figure 4(b)).
   a) Each item should be clickable. Clicking on any item should show the detailed weather for that day in another RecyclerView below this view.
   b) Each item in the RecyclerView should display the Date, corresponding weather icon and the average temperature of the day.
   c) You need to load the weather icon image from: http://openweathermap.org/img/w/{Symbol_serial_number}.png. In JSON, you will get the Symbol serial number in the weather array, an attribute called “icon” contains it. In XML, you can find it in “var” attribute of “symbol” tag.
   d) Pick the icon of the Median forecast for each day. For example, you get 8 weather forecasts for a day. So, the Median forecast will be the 5th forecast.

e) Next, you need to display the average temperature. Get all the temperature values for each day. Calculate the average and display it. NOTE: avoid taking min and max values.

7. Below the RecyclerView, display a TextView containing “Three Hourly Forecast on {Date}”. By default, the Date will be the first date in the upper RecyclerView. If the user press on any of the rest days other than current day, the Date will be changed to that day.
8. The second RecyclerView should display three hourly forecast for the selected day. This RecyclerView should contain several items containing Time, Weather icon, Temperature, Condition, Pressure, Humidity and Wind speed and direction. This RecyclerView should be designed to display one item at once on the screen. This view should be horizontally scrollable, See Figure 4(b).
9. There are two Menu buttons in this activity.
   a) Save City: Pressing on this button should save the city’s details (cityname, country, temperature, favorite) into SQLite database (Figure 4 (c)).
      a) If the city has been previously saved, then simply update the stored temperature to reflect the new temperature. A Toast should display the message, “City Updated”.
      b) If the city has not been previously saved, then save the new city and set the favorite flag to false. **In case of temperature, always use celsius unit to save into database.** A Toast should display the message, “City Saved”.
      c) Upon returning to the Main Activity the list should show the newly added city as a saved city.
   b) Settings: Pressing on this button will do exactly same tasks with Preference Activity as Main Activity to change the units from ºC to ºF or vice versa (Figure 4 (f, g)). Temperature unit in the main activity or CityWeather Activity should be displayed based on the shared preference saved using the Preference Activity.