Android Search

Android Search Framework

- The android platform provides us with a search framework that helps us implementing search mechanism in our application.

Search Input

- We can allow the user to enter the searched term either through a search dialogue shown on top of the screen or through a search view widget.

Search Input

- Choosing to allow the user to search using a search dialogue we should include in the application UI kind of a button (or something similar) that when the user presses it the search dialogue will be popped up on the screen.

Searchable Configuration

- When developing an application that includes a searching mechanism we should configure that searching using an xml file.
- The name of this file is usually `searchable.xml` and you should save it within the rex\xml folder.

Searchable Configuration

```
<?xml version="1.0" encoding="utf-8"?>
<searchable
xmlns:android="http://schemas.android.com/apk/res/android"
    android:label="@string/app_label"
    android:hint="@string/search_hint" >
</searchable>
```

The android:hint attribute provides a hint string in the search box before the user enters the query. The hint provides the users with an important clues to users about what they can search. You better format the string for android:hint as "Search _____" (e.g. "Search Books" or "Search Movies".

The android:label attribute points to a string resource that should be the application name. This label becomes visible to the user when he enables search suggestions in the quick search box.

Searchable Configuration

- The <searchable> can accept many other attributes that allow us to configure the search and add special features to it, such as voice search and search suggestions.
- You can find detailed information about the possibilities at http://developer.android.com/guide/topics/search/searchable-config.html

The Searchable Activity

- This is the activity that receives the search query, searches the application data, and displays the search results. When the user executes a search in the search dialog or widget, the system starts the searchable activity and send it the search query.
- The search query is bundled within the intent the system created in order to start the searchable activity. The action of that intent should be ACTION_SEARCH.

The Searchable Activity

- This searchable activity will be defined in the manifest file just as any other activity.

```
<application ... >

    <activity android:name=".OurSearchableActivity" >
        <intent-filter>
            <action android:name="android.intent.action.SEARCH" />
        </intent-filter>
        <meta-data android:name="android.app.searchable"
                   android:resource="@xml/searchable"/>
    </activity>
    ...
</application>
```

The Other Activities

- Activities that allow the user to perform the search through a search dialogue will be defined in the following way:

```
<application ... >
    ...
    <activity android:name=".OtherActivity" ... >
        <meta-data android:name="android.app.default_searchable"
                   android:value=".SearchableActivity" />
    </activity>
    ...
</application>
```

Sample

SearchActivity.java

```
public class SearchActivity extends Activity
{

    @Override
    protected void onCreate(Bundle savedInstanceState)
    {
        Map<String,String> map = new Hashtable<String,String>();
        map.put("Shalom", "Hello in Hebrew");
        map.put("Salam","Hello in Arabic");
        map.put("Bonjour", "Hello in French");
        map.put("Hola","Hello in Spanish");
        String result = "found nothing!";
        super.onCreate(savedInstanceState);
        Intent intent = getIntent();
```

Sample

```
        if (Intent.ACTION_SEARCH.equals(intent.getAction()))
        {
            String query = intent.getStringExtra(SearchManager.QUERY);
            Log.i("search","query="+query);
            String str = map.get(query);
            if(str!=null)
            {
                result = "result: "+query+" is "+str;
            }
        }
        TextView text = new TextView(this);
        text.setText(result);
        setContentView(text);
    }
}
```

Sample

MainActivity.java

```
public class MainActivity extends Activity
{

    @Override
    protected void onCreate(Bundle savedInstanceState)
    {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        Button bt = (Button) findViewById(R.id.btSearch);
        bt.setOnClickListener(new View.OnClickListener()
        {
            @Override
            public void onClick(View arg0)
            {
                onSearchRequested ();
            }
        });
    }

}
```

Sample

activity_main.xml

```xml
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity" >

    <Button
        android:id="@+id/btSearch"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentTop="true"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="74dp"
        android:text="@string/search_text" />

</RelativeLayout>
```

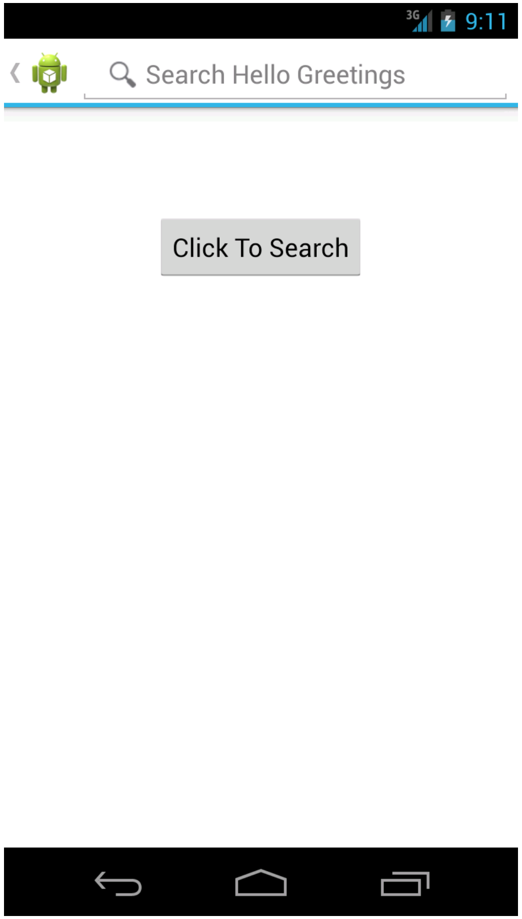
Sample

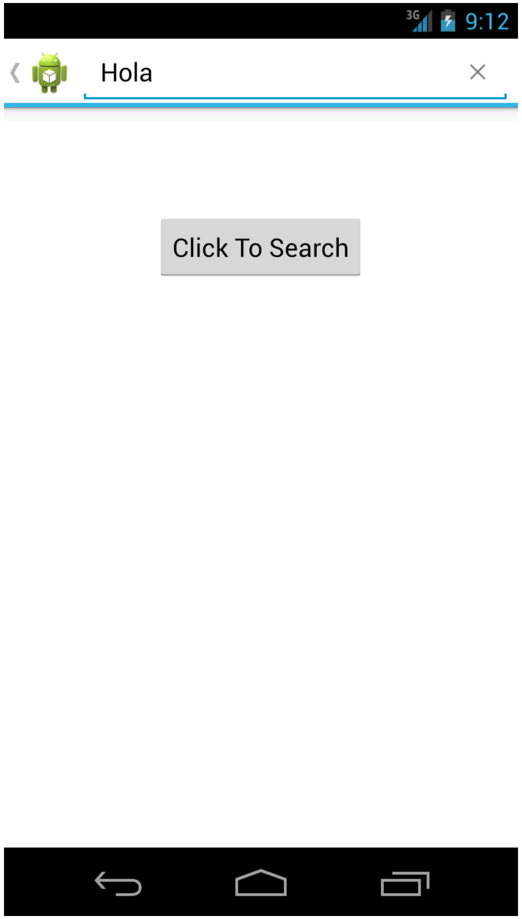
```xml
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.lifemichael.searchsample"
    android:versionCode="1"
    android:versionName="1.0" >

    <uses-sdk
        android:minSdkVersion="8"
        android:targetSdkVersion="17" />

    <application
        android:allowBackup="true"
        android:icon="@drawable/ic_launcher"
        android:label="@string/app_name"
        android:theme="@style/AppTheme" >
        <meta-data
            android:name="android.app.default_searchable"
            android:value=".SearchActivity" />
```

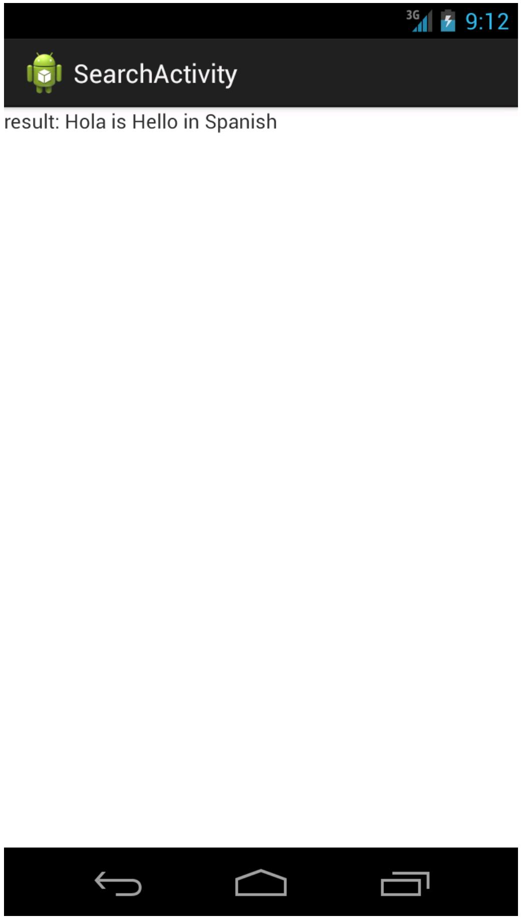
Sample

```
        <activity
            android:name="com.lifemichael.searchsample.MainActivity"
            android:label="@string/app_name" >
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
        <activity
            android:name="com.lifemichael.searchsample.SearchActivity"
            android:label="@string/title_activity_search" >
            <intent-filter>
                <action android:name="android.intent.action.SEARCH" />
            </intent-filter>
            <meta-data
                android:name="android.app.searchable"
                android:resource="@xml/searchable" />
        </activity>
    </application>
</manifest>
```

Sample







result: Hola is Hello in Spanish