Assignment 06

COS30017 - Software Development for Mobile Devices

Daniel Parker 971328X

October 16, 2014

1. Task 1 - Activities and Fragments

The difference between activities and fragments lies primarily in their lifecycle and where they exist in the context of an app. Firstly, Fragment is a subclass of Activity and therefore inherits all of the Activity's primary functionality. The difference is that an Activity can contain many fragments, and the fragments will provide their UI to the containing Activity. This pattern also allows for reusability of UI components in different activities.

The FragmentManager is a central class to managing fragments across an app. It allows for searching for fragments, managing the fragments on the back stack, and creating new FragmentTransactions. FragmentTransactions are an API for controlling the behaviour of Fragments in an app, and what fragments are visible and also that any changes get committed to the backstack to allow for proper back-button navigation by the user.

2. Task 2 - Simple Custom List

2.1. Screenshot

2.2. Source Code

2.2.1. Movie Model

```
package au.net.danielparker.movies;

import android.os.Parcel;
import android.os.Parcelable;
import android.util.Log;

import com.google.gson.Gson;
import com.google.gson.reflect.TypeToken;

import java.io.BufferedInputStream;
import java.io.BufferedReader;
import java.io.FileInputStream;
import java.io.IOException;
import java.io.InputStream;
import java.io.InputStreamReader;
import java.lang.reflect.Type;
import java.util.ArrayList;
import java.util.TimeZone;

/**
 * Created by danielparker on 15/10/14.
 */
public class Movie implements Parcelable {
    private String title;
    private Double rating;
    private String iconName;
```

```
public static final Parcelable.Creator<Movie> CREATOR
        = new Parcelable.Creator<Movie>() {
    public Movie createFromParcel(Parcel in) {
        return new Movie(in);
    }

    public Movie[] newArray(int size) {
        return new Movie[size];
    }
};

public Movie(Parcel in) {
    this.title = in.readString();
    this.rating = in.readDouble();
    this.iconName = in.readString();
}

public Movie(String name, Double rating, String iconName) {
    this.title = name;
    this.rating = rating;
    this.iconName = iconName;
}

public static ArrayList<Movie> loadMovies(InputStream moviesFile){
    ArrayList<Movie> movies = new ArrayList<Movie>();
    BufferedReader inputStream = new BufferedReader(new InputStreamReader(moviesFile));
    StringBuilder stringBuilder = new StringBuilder();

    try {
```

```
            String temp;
            while ((temp = inputStream.readLine()) != null) {
                stringBuilder.append(temp);
            }

            Type movieList = new TypeToken<ArrayList<Movie>>() {
            }.getType();

            movies = new Gson().fromJson(stringBuilder.toString(), movieList);
            // Read from JSON
        } catch (IOException e) {
            Log.e("MOVIES", "Error reading from JSON");
        }

        return movies;
    }

    public void writeToParcel(Parcel out, int flags) {
        out.writeString(this.title);
        out.writeDouble(this.rating);
        out.writeString(this.iconName);
    }

    public String getName() {
        return title;
    }

    public String getRating() {
        return rating.toString();
    }
```

```
    public String getIconName() {
        return iconName;
    }

    public int describeContents() {
        return 0;
    }
}
```

2.2.2. MovieList Activity

```
package au.net.danielparker.movies;

import android.app.ListActivity;
import android.content.res.AssetManager;
import android.os.Bundle;
import android.util.Log;
import android.view.Menu;
import android.view.MenuItem;
import android.view.View;
import android.widget.ArrayAdapter;
import android.widget.ListView;
import android.widget.Toast;

import java.io.IOException;
import java.util.ArrayList;


public class MovieList extends ListActivity {
```

```
private ArrayList<Movie> listData = new ArrayList<Movie>();
private ArrayAdapter<Movie> adapter;

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_movie_list);

    initializeUI();
}

@Override
public void onListItemClick(ListView l, View v, int position, long id) {
    Movie selectedItem = (Movie) getListView().getItemAtPosition(position);
    Toast ratingToast = Toast.makeText(getApplicationContext(), selectedItem.getRating(), Toast.LENGTH_LONG);
    ratingToast.show();
}

@Override
public boolean onCreateOptionsMenu(Menu menu) {
    // Inflate the menu; this adds items to the action bar if it is present.
    getMenuInflater().inflate(R.menu.movie_list, menu);
    return true;
}

@Override
public boolean onOptionsItemSelected(MenuItem item) {
    // Handle action bar item clicks here. The action bar will
    // automatically handle clicks on the Home/Up button, so long
```

```
        // as you specify a parent activity in AndroidManifest.xml.
        int id = item.getItemId();
        if (id == R.id.action_settings) {
            return true;
        }
        return super.onOptionsItemSelected(item);
    }

    public void initializeUI() {
        AssetManager assetManager = getAssets();

        try {
            this.listData = Movie.loadMovies(assetManager.open("movies.json"));
            this.adapter = new MovieListAdapter(this, this.listData);

            setListAdapter(adapter);

        } catch (IOException e) {
            Log.e("MOVIES", e.getMessage());
        }
    }
}
```


2.2.3. MovieListAdapter

```
package au.net.danielparker.movies;

import android.content.Context;
import android.content.res.Resources;
import android.view.LayoutInflater;
```

```
import android.view.View;
import android.view.ViewGroup;
import android.widget.ArrayAdapter;
import android.widget.ImageView;
import android.widget.TextView;

import java.util.ArrayList;
import java.util.ResourceBundle;

/**
 * Created by danielparker on 15/10/14.
 */
public class MovieListAdapter extends ArrayAdapter<Movie>{
    private final Context context;
    private final ArrayList<Movie> values;

    public MovieListAdapter(Context context, ArrayList<Movie> values) {
        super(context, R.layout.movie_layout, values);
        this.context = context;
        this.values = values;
    }

    @Override
    public View getView(int position, View convertView, ViewGroup parent) {
        LayoutInflater inflater = (LayoutInflater) context
                .getSystemService(Context.LAYOUT_INFLATER_SERVICE);
        View rowView = inflater.inflate(R.layout.movie_layout, parent, false);

        ImageView movieIcon = (ImageView)rowView.findViewById(R.id.icon);
        TextView movieName = (TextView)rowView.findViewById(R.id.movie_name);
```

```
        TextView movieRating = (TextView)rowView.findViewById(R.id.movie_rating);

        movieIcon.setImageResource(R.drawable.movie_icon);
        movieName.setText(values.get(position).getName());
        movieRating.setText(values.get(position).getRating());

        return rowView;
    }
}
```

2.2.4. activity_movie_list.xml

```
<LinearLayout 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"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    android:paddingBottom="@dimen/activity_vertical_margin"
    tools:context=".MovieList">

    <ListView
        android:id="@android:id/list"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content">
    </ListView>

</LinearLayout>
```

2.2.5. movie_layout.xml

```
<LinearLayout 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"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    android:paddingBottom="@dimen/activity_vertical_margin"
    tools:context=".MovieList">

    <ListView
        android:id="@android:id/list"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content">
    </ListView>

</LinearLayout>
```

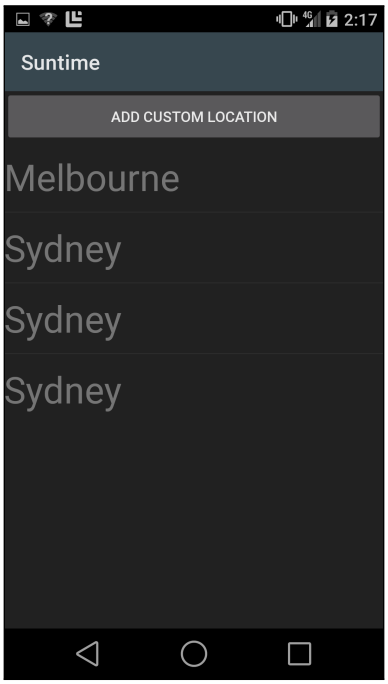
3. Task 3 - Action Bar Design Pattern

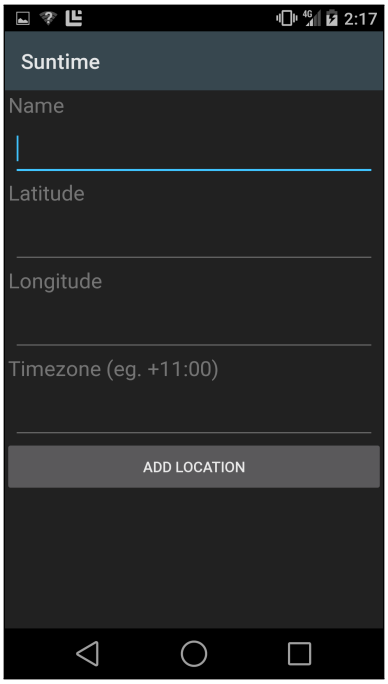
The action bar design pattern has become a very prominent visual design pattern and is preferred to the older pattern of puttings everything on a landing dashboard. The reasons for it's success and recommended use, are that it provides a dedicated space for the app logo and branding, as well as the user's current location in the app. It makes the important contextual actions for an Activity prominent, and it provides consistent navigation when switching between apps. It also works nicely across multiple device screen sizes and orientations.

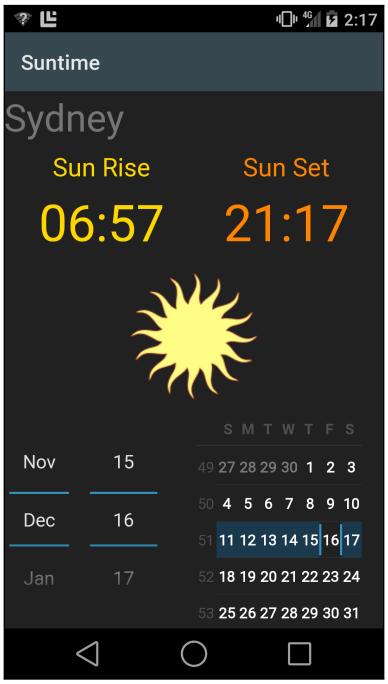
4. Task 4 - Add a Custom Geo Location

The Custom Geo Location app uses the devices internal storage to allow the user to add custom locations to get suntimes for. The modes that are used are *MODE_PRIVATE* and *MODE_APPEND*.

4.1. Screenshots




2:17
Suntime
Name
Latitude
Longitude
Timezone (eg. +11:00)
ADD LOCATION


2:17
Suntime
Sydney
Sun Rise
Sun Set
06:57
21:17
Nov
Dec
Jan
15
16
17
S M T W T F S
49 27 28 29 30 1 2 3
50 4 5 6 7 8 9 10
51 11 12 13 14 15 16 17
52 18 19 20 21 22 23 24
53 25 26 27 28 29 30 31

4.2. Source Code

4.2.1. LocationsActivity

```
package au.net.danielparker.suntime.ui;

import android.app.ListActivity;
import android.content.Intent;
import android.content.res.AssetManager;
import android.os.Bundle;
import android.util.Log;
import android.view.View;
import android.widget.ArrayAdapter;
import android.widget.ListView;
import android.widget.TextView;

import org.androidannotations.annotations.Click;
import org.androidannotations.annotations.EActivity;
import org.androidannotations.annotations.ViewById;

import java.io.IOException;
import java.io.InputStream;
import java.util.ArrayList;

import au.net.danielparker.suntime.R;
import au.net.danielparker.suntime.models.Location;

/**
 * Created by danielparker on 29/09/14.
 */
```

```
@EActivity(R.layout.activity_locations)
public class LocationsActivity extends ListActivity {
    private ArrayList<Location> listData = new ArrayList<Location>();
    private ArrayAdapter<Location> adapter;

    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);

        initialiseUI();
    }

    @Override
    public void onListItemClick(ListView l, View v, int position, long id) {
        Location selectedItem = (Location) getListView().getItemAtPosition(position);
        showSuntimesForItem(selectedItem);
    }



    private void showSuntimesForItem(Location selectedItem) {
        Intent intent = new Intent(this, SuntimeActivity.class);
        intent.putExtra("location", selectedItem);
        startActivity(intent);
    }

    private void initialiseUI() {
        this.listData = Location.loadLocations(getApplicationContext());
        adapter = new LocationsAdapter(this, this.listData);
```

```
        setListAdapter(adapter);
    }

    @Click(R.id.custom_button)
    void onCustomButtonClick() {
        Intent intent = new Intent(this, AddNewLocation_.class);
        startActivity(intent);
    }
}
```

4.2.2. AddNewLocation

```
package au.net.danielparker.suntime.ui;

import android.app.Activity;
import android.content.Intent;
import android.os.Bundle;
import android.widget.EditText;
import android.widget.Toast;

import org.androidannotations.annotations.Click;
import org.androidannotations.annotations.EActivity;
import org.androidannotations.annotations.ViewById;

import java.util.ArrayList;
import java.util.TimeZone;

import au.net.danielparker.suntime.R;
import au.net.danielparker.suntime.models.Location;
```

```java
/**
 * Created by danielparker on 15/10/14.
 */
@EActivity(R.layout.activity_add_custom)
public class AddNewLocation extends Activity {
    @ViewById(R.id.form_name)
    public EditText formName;

    @ViewById(R.id.form_latitude)
    public EditText formLat;

    @ViewById(R.id.form_longitude)
    public EditText formLon;

    @ViewById(R.id.form_timezone)
    public EditText formTimezone;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
    }

    @Click(R.id.form_submit)
    void onFormSubmit() {
        String name;
        Double latitude;
        Double longitude;
        TimeZone timeZone;
```

```
        try {
            name = formName.getText().toString();
            latitude = validateLatitude();
            longitude = validateLongitude();
            timeZone = validateTimezone();

            Location newLocation = new Location(name, latitude, longitude, timeZone);
            ArrayList<Location> newLocations = new ArrayList<Location>();
            newLocations.add(newLocation);
            Location.saveToDevice(getApplicationContext(), newLocations);

            finish();
        } catch (Exception e){
            Toast validationError = Toast.makeText(this, e.getMessage(), Toast.LENGTH_SHORT);
            validationError.show();
        }
    }

    Double validateLatitude() throws Exception {
        Double latitude;
        try {
            latitude = Double.parseDouble(formLat.getText().toString());
            if (!(-90 <= latitude && latitude <= 90)) {

            }
        } catch (Exception e) {
            throw new Exception("Latitude must be between -90 and 90");
        }

        return latitude;
```

```
    }

    Double validateLongitude() throws Exception {
        Double longitude;
        try {
            longitude = Double.parseDouble(formLat.getText().toString());
            if (!(-180 <= longitude && longitude <= 180)) {
                throw new Exception("Longitude must be between -180 and 180");
            }
        } catch (Exception e) {
            throw new Exception("Longitude must be between -180 and 180");
        }

        return longitude;
    }

    TimeZone validateTimezone() throws Exception {
        TimeZone timeZone = TimeZone.getTimeZone("GMT" + formTimezone.getText().toString());
        return timeZone;
    }
}
```

4.2.3. LocationsAdapter

```
package au.net.danielparker.suntime.ui;

import android.content.Context;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
```

```
import android.widget.ArrayAdapter;
import android.widget.TextView;

import java.util.ArrayList;

import au.net.danielparker.suntime.R;
import au.net.danielparker.suntime.models.Location;


public class LocationsAdapter extends ArrayAdapter<Location> {

    private final Context context;
    private final ArrayList<Location> values;

    public LocationsAdapter(Context context, ArrayList<Location> values) {
        super(context, R.layout.list_location_item, values);
        this.context = context;
        this.values = values;
    }

    @Override
    public View getView(int position, View convertView, ViewGroup parent) {
        LayoutInflater inflater = (LayoutInflater) context
                .getSystemService(Context.LAYOUT_INFLATER_SERVICE);
        View rowView = inflater.inflate(R.layout.list_location_item, parent, false);

        TextView locationName = (TextView)rowView.findViewById(R.id.Location_Name);
        locationName.setText(values.get(position).getName());

        return rowView;
```

```
    }
}
```

4.2.4. Location

4.2.5. activity_locations

```
<?xml version="1.0" encoding="utf-8"?>

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical" android:layout_width="match_parent"
    android:layout_height="match_parent">
    <Button
        android:id="@+id/custom_button"
        android:layout_height="wrap_content"
        android:layout_width="fill_parent"
        android:text="Add Custom Location"
        >
    </Button>
    <ListView
        android:id="@android:id/list"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content">
    </ListView>
</LinearLayout>
```

4.2.6. activity_add_custom

```
<?xml version="1.0" encoding="utf-8"?>

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical" android:layout_width="match_parent"
    android:layout_height="match_parent">

    <TextView
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:paddingLeft="@dimen/form_padding"
        android:text="@string/form_name"
        android:textSize="20sp"/>

    <EditText
        android:id="@+id/form_name"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content" />

    <TextView
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:paddingLeft="@dimen/form_padding"
        android:text="@string/form_lat"
        android:textSize="20sp"/>

    <EditText
        android:id="@+id/form_latitude"
        android:layout_width="fill_parent"
```

```
        android:layout_height="wrap_content" />

    <TextView
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:paddingLeft="@dimen/form_padding"
        android:text="@string/form_lon"
        android:textSize="20sp"/>

    <EditText
        android:id="@+id/form_longitude"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content" />

    <TextView
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:paddingLeft="@dimen/form_padding"
        android:text="@string/form_timezone"
        android:textSize="20sp"/>

    <EditText
        android:id="@+id/form_timezone"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content" />

    <Button
        android:id="@+id/form_submit"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
```

```
        android:text="@string/submit_button">
    </Button>

</LinearLayout>
```