CS493

Regis University Senior Capstone

Week Four

Intructor: Mr. Allan Rossi

Author: Mr. Jason N Mansfield

The Object Based Study Application

July 28, 2013

Contents

July 28, 2013

1 Continued efforts with SQLite3

I have been capable of investigating and utilizing the concept of CRUD based operation (Tamada, 2012) using the android API. I have created a rough Database handler which extends SQLiteOpenHelper(*Saving Data in SQL Databases* 2013; *SQLiteOpenHelper* 2013). This extension has allowed me to create a SQLite database and table directly from the application itself launching. I have discovered that I can easily view the SQLite databases, tables and content within using adb. At this point I am fairly confident I will have a fully functional database within the week.

2 Continued efforts with ListView

ListView is the other piece of the pie, so to speak. The ListView (*ListView* 2013) is specifically suited for the SQLite database. Simply put ListView is a String array which can be used to cycle through and create a View for the customer or developer.

```
final ListView listview = (ListView) findViewById(R.id.listview);
String[] values = new String[] {"Object 1", "Object 2", "Object 3", "Object 4"};

final ArrayList<String> list = new ArrayList<String>();
for(int i = 0; i < values.length; ++i){
list.add(values[i]);
}
```

My next step is to take the above code, or similar code, and retrieve the database information to place within it. I have created some constructors which should be helpful in filling the needed data into a String array. For example:

```
public int getID(){
return this._id;
}
public void setID(int id){
this._id = id;
}
```

Once I have succeeded with passing information from the SQLite database to the View, then I feel this application will truly begin growing.

3 Some information regarding Gradle

Gradle is the new official build system for android applications (*Gradle: The New Android Build System* 2013). The new Android Studio is currently offered to developers with a warning that it has its flaws. One of the larger issues appears to be with the android.R library (*R* 2013). Many times the errors revolve around the R library due to poor editing in the layout:

```
< 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"
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=".ListHandlerActivity">

<ListView xmlns:android="http://schemas.android.com/apk/res/android"
android:id="@+id/listview"
android:layout_width="wrap_content"
android:layout_height="wrap_content" />

</RelativeLayout>
```

However, even once the error is fixed in the xml file, or sometimes without a mistake at all, R simply cannot seem to see the layout files. This is where Gradle comes to the rescue. The quickest way to deal with strange errors in the current Android Studio is to run CLI command in the root of the Project folder. For example:

```
./gradlew clean
./gradlew assemble
```

Once these Gradle type command are run, assuming the xml file was written correctly or corrected the layouts become recognized once again. This is of course a extremely limited view of the power Gradle holds. Gradle is a Domain Specific Language (*Overview* 2013) which is used for automation purposes. While I am still new to this build system I can see how it could

be used to copy previous build approaches and increase productivity. I have had some difficulty with a few of the Gradle concepts but I intend to explore the Gradle approach more in-depth over the next week. I feel its important to understand the build system for debugging purposes and clean code. Hopefully, I can add more about this exciting new tool on my next weekly summary.

Bibliography

Gradle: The New Android Build System (2013). Gradleware Inc. URL: http://www.gradleware.com/resources/tech/android.

ListView (2013). Android Open Source Project. URL: http://developer.android.com/guide/topics/ui/layout/listview.html#Example.

Overview (2013). Gradleware Inc. URL: http://www.gradle.org/overview.

R (2013). Android Open Source Project. URL: http://developer.android.com/reference/android/R.html.

Saving Data in SQL Databases (2013). Android Open Source Project. URL: http://developer.android.com/training/basics/data-storage/databases.html.

SQLiteOpenHelper (2013). Android Open Source Project. URL: http://developer.android.com/reference/android/database/sqlite/SQLiteOpenHelper.html.

Tamada, Ravi (2012). *Android SQLite Database Tutorial*. URL: http://www.androidhive.info/2011/11/android-sqlite-database-tutorial/.