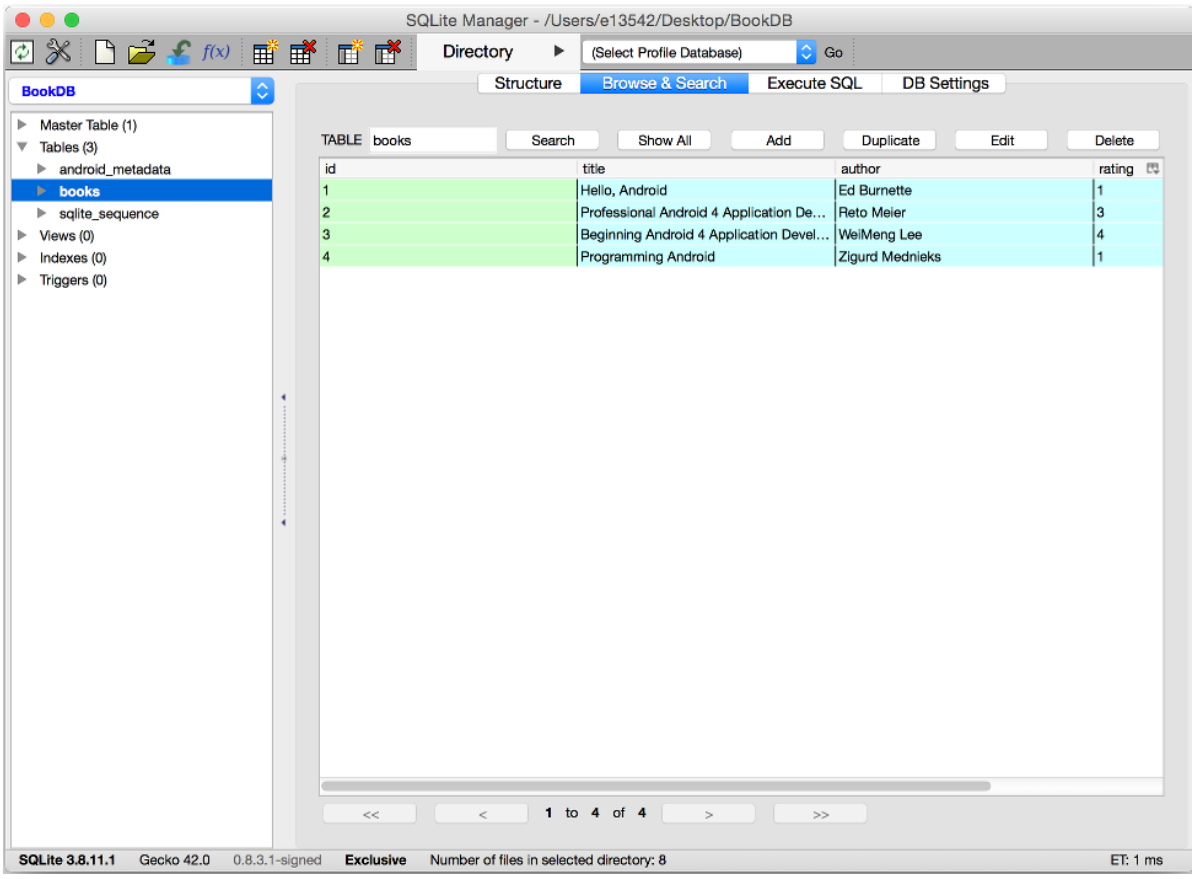
Github

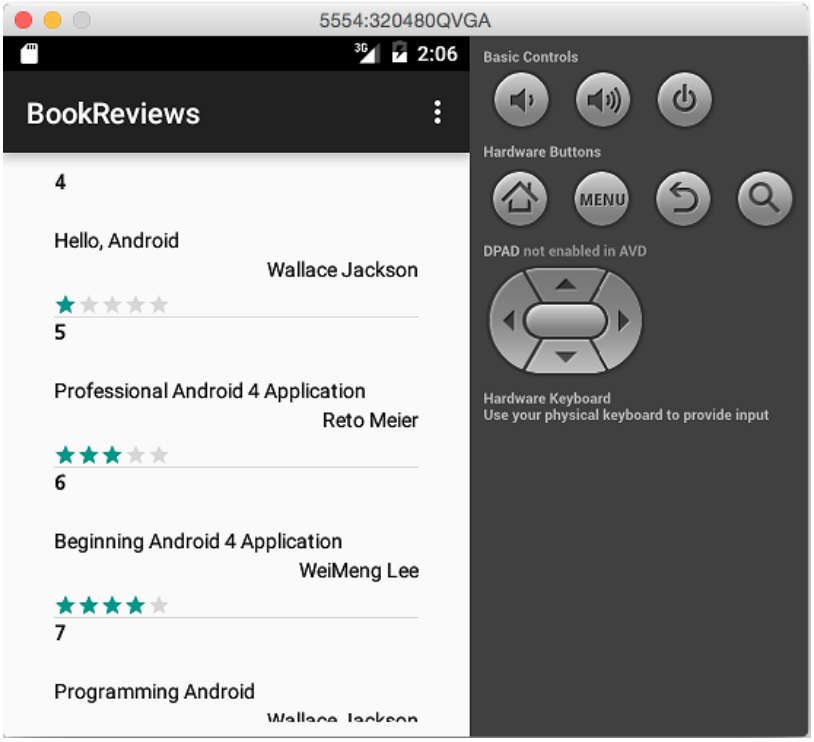
The source code for this assignment, which discussed in this document, is published online @ https://github.com/bradyhouse/ITM455/tree/master/HW7.

Screenshots

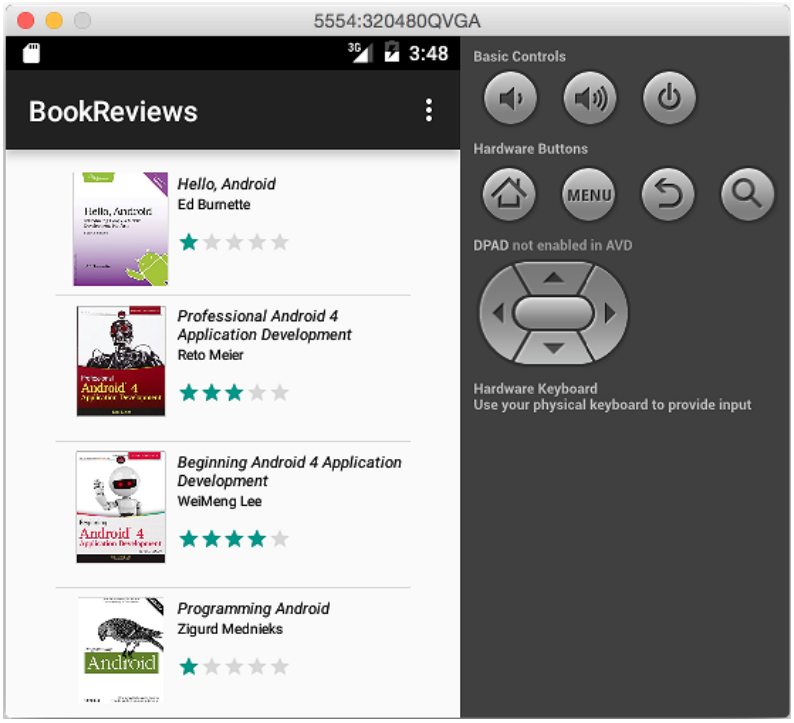
Step 6 – SQLite Manager ~ Books Table



Step 7 – Interface

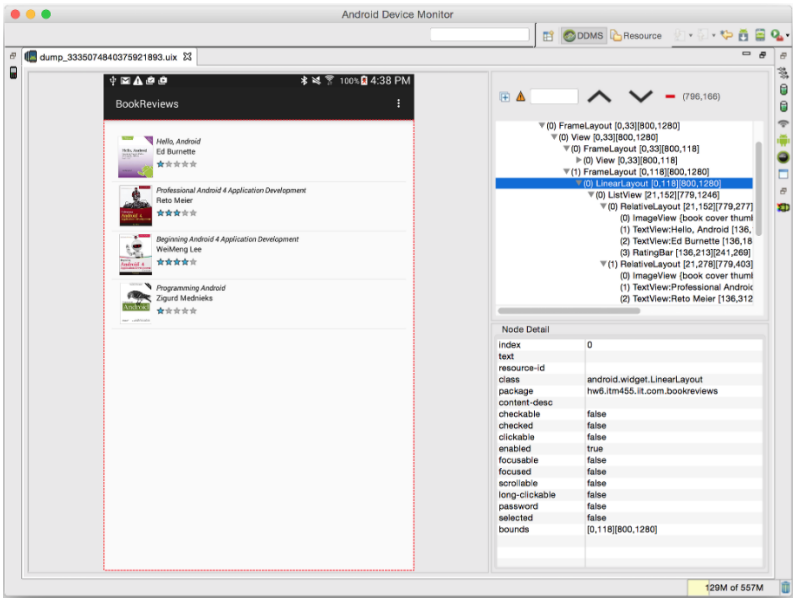

5554:320480QVGA
2:06
BookReviews
4
Hello, Android
Wallace Jackson
5
Professional Android 4 Application
Reto Meier
6
Beginning Android 4 Application
WeiMeng Lee
7
Programming Android
Basic Controls
Hardware Buttons
MENU
DPAD not enabled in AVD
Hardware Keyboard
Use your physical keyboard to provide input

Step 8 – Completed Interface (Cellphone Emulator)



Step 8 – Completed Interface (USB Device)

I also deployed the app to Galaxy Tab 4 via USB. Here is a screenshot.



Source Code

MainActivity.java

```
package hw6.itm455.iit.com.bookreviews;

import java.util.List;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.Menu;
import android.view.MenuItem;
import android.widget.ListView;

public class MainActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        SqlHelper db = new SqlHelper(this);
        if (db.getAllBooks().size() == 0) {
            db.addBook(new Book("Hello, Android", "Ed Burnette", "1"));
            db.addBook(new Book("Professional Android 4 Application Development", "Reto Meier",
"3"));
            db.addBook(new Book("Beginning Android 4 Application Development", "WeiMeng Lee",
"4"));
            db.addBook(new Book("Programming Android", "Zigurd Mednieks", "1"));
        }
        List<Book> list = db.getAllBooks();
        ListView listContent = (ListView) findViewById(R.id.list);
        ListAdapter customAdapter = new ListAdapter(this, R.layout.itemlistrow, list );
        listContent.setAdapter(customAdapter);
    }

    @Override
    public boolean onCreateOptionsMenu(Menu menu) {
        getMenuInflater().inflate(R.menu.menu_main, menu);
        return true;
    }

    @Override
    public boolean onOptionsItemSelected(MenuItem item) {
        int id = item.getItemId();
        if (id == R.id.action_settings) {
            return true;
        }
        return super.onOptionsItemSelected(item);
    }
}
```

Book.java

```
package hw6.itm455.iit.com.bookreviews;

public class Book {
    private int id;
    private String title;
    private String author;
    private String rating;
    public Book() {
    }
    public Book(String title, String author, String rating) {
        super();
        this.title = title;
        this.author = author;
        this.rating = rating;
    }
    public int getId() {
        return id;
    }
    public void setId(int id) {
        this.id = id;
    }
    public String getTitle() {
        return title;
    }
    public void setTitle(String title) {
        this.title = title;
    }
    public String getAuthor() {
        return author;
    }
    public void setAuthor(String author) {
        this.author = author;
    }
    public void setRating(String rating) {
        this.rating = rating;
    }
    public String getRating() {
        return rating;
    }
    @Override
    public String toString() {
        return "Book [id=" + id + ", title=" + title + ", author=" + author
                + ", rating=" + rating + "]";
    }
}
```

SqlHelper.java

```
package hw6.itm455.iit.com.bookreviews;

import java.util.LinkedList;
import java.util.List;

import android.content.ContentValues;
import android.content.Context;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;
import android.util.Log;

public class SqlHelper extends SQLiteOpenHelper {
    private static final String LOG_PREFIX = "BR-";
    private static final int DATABASE_VERSION = 1;
    private static final String DATABASE_NAME = "BookDB";
    private static final String TABLE_BOOKS = "books";
    private static final String KEY_ID = "id";
    private static final String KEY_TITLE = "title";
    private static final String KEY_AUTHOR = "author";
    private static final String KEY_RATING = "rating";
    public SqlHelper(Context context) {
        super(context, DATABASE_NAME, null, DATABASE_VERSION);
    }
    @Override
    public void onCreate(SQLiteDatabase db) {
        String CREATE_BOOK_TABLE = "CREATE TABLE books ( " +
                "id INTEGER PRIMARY KEY AUTOINCREMENT, " +
                "title TEXT, " +
                "author TEXT, " +
                "rating TEXT)";
        db.execSQL(CREATE_BOOK_TABLE);

    }
    @Override
    public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {
        db.execSQL("DROP TABLE IF EXISTS books");
        this.onCreate(db);
    }
    public void addBook(Book book) {
        Log.d(this.LOG_PREFIX + "addBook", book.toString());
        SQLiteDatabase db = this.getWritableDatabase();
        ContentValues values = new ContentValues();
        values.put(KEY_TITLE, book.getTitle());
        values.put(KEY_AUTHOR, book.getAuthor());
        values.put(KEY_RATING, book.getRating());
        db.insert(TABLE_BOOKS,
                null,
                values);
        db.close();
    }
    public List<Book> getAllBooks() {
        List<Book> books = new LinkedList<Book>();
        String query = "SELECT * FROM " + TABLE_BOOKS;
        SQLiteDatabase db = this.getWritableDatabase();
        Cursor cursor = db.rawQuery(query, null);
        Book book = null;
        if (cursor.moveToFirst()) {
            do {
                book = new Book();
                book.setId(Integer.parseInt(cursor.getString(0)));
                book.setTitle(cursor.getString(1));
                book.setAuthor(cursor.getString(2));
                book.setRating(cursor.getString(3));
                books.add(book);
            } while (cursor.moveToNext());
        }
```

```
        Log.d(this.LOG_PREFIX + "getAllBooks()", books.toString());
        return books; // return books
    }
    public int updateBook(Book book, String newTitle, String newAuthor) {
        SQLiteDatabase db = this.getWritableDatabase();
        ContentValues values = new ContentValues();
        values.put("title", newTitle); // get title
        values.put("author", newAuthor); // get author
        int i = db.update(TABLE_BOOKS, //table
                values, // column/value
                KEY_ID + " = ?", // selections
                new String[]{String.valueOf(4)}); //selection args
        db.close();
        Log.d(this.LOG_PREFIX + "UpdateBook", book.toString());
        return i;
    }
    public void deleteBook(Book book) {
        SQLiteDatabase db = this.getWritableDatabase();
        db.delete(TABLE_BOOKS,
                KEY_ID + " = ?",
                new String[]{String.valueOf(book.getId())});
        db.close();
        Log.d(this.LOG_PREFIX + "deleteBook", book.toString());
    }
    public int getIds(Book book) {
        String selectQuery = "SELECT id FROM books";
        SQLiteDatabase database = this.getReadableDatabase();
        Cursor c = database.rawQuery(selectQuery, null);
        c.moveToFirst();
        int total = c.getCount();
        Log.d(this.LOG_PREFIX + "getIds", Integer.toString(total));
        return total;
    }
}
```

ListAdapter.java

```
package hw6.itm455.iit.com.bookreviews;

import java.util.List;
import android.content.Context;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.ArrayAdapter;
import android.widget.ImageView;
import android.widget.RatingBar;
import android.widget.TextView;

public class ListAdapter extends ArrayAdapter<Book> {
    private List<Book> items;

    // Sequential list of icons
    private Integer[] icons = {R.drawable.book_4, R.drawable.book_5,
            R.drawable.book_6, R.drawable.book_7
    };

    public ListAdapter(Context context, int textViewResourceId) {
        super(context, textViewResourceId);
    }
    public ListAdapter(Context context, int resource, List<Book> items) {
        super(context, resource, items);
        this.items = items;
    }
    @Override
    public View getView(int position, View convertView, ViewGroup parent) {
        View v = convertView;
        if (v == null) {
            LayoutInflater vi;
            vi = LayoutInflater.from(getContext());
            v = vi.inflate(R.layout.itemlistrow, null);
        }
        Book p = getItem(position);
        if (p != null) {
            ImageView icon=(ImageView) v.findViewById(R.id.icon);
            TextView tt1 = (TextView) v.findViewById(R.id.title);
            TextView tt3 = (TextView) v.findViewById(R.id.author);
            RatingBar rb = (RatingBar) v.findViewById(R.id.rating);
            if (icon != null) {
                icon.setImageResource(this.icons[position]);
            }
            if (tt1 != null) {
                tt1.setText(p.getTitle());
            }
            if (tt3 != null) {
                tt3.setText(p.getAuthor());
            }
            if (rb != null) {
                float rating = Float.parseFloat(p.getRating());
```

```
                rb.setRating(rating);
            }
        }
        return v;
    }
}
```

Layout

activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<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:layout_centerVertical="true"
    android:paddingBottom="@dimen/activity_vertical_margin"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    android:orientation="vertical" >
    <ListView
        android:id="@+id/list"
        android:layout_width="fill_parent"
        android:layout_marginTop="10dp"
        android:layout_marginBottom="10dp"
        android:layout_height="fill_parent"
        android:layout_gravity="center" >
    </ListView>
</LinearLayout>
```

itemlistrow.xml

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="match_parent"
    android:layout_alignParentLeft="true"
    android:padding="6dip"
   >
    <ImageView
        android:id="@+id/icon"
        android:contentDescription="@string/icon_alt_text"
        android:layout_width="80dip"
        android:layout_height="80dip"
        android:layout_centerVertical="true"
        android:layout_alignTop="@+id/title"
        android:src="@drawable/book_4" />
    <TextView android:textColor="#000"
        android:id="@+id/title"
        android:layout_width="wrap_content"
        android:layout_toEndOf="@+id/icon"
        android:layout_height="wrap_content"
        android:textStyle="italic"
        android:text="title" />
    <TextView android:layout_height="wrap_content"
        android:layout_width="fill_parent"
        android:textColor="#000"
        android:id="@+id/author"
        android:layout_below="@+id/title"
        android:layout_alignLeft="@+id/title"
        android:text="author"
        android:height="20sp" />
    <RatingBar
        android:id="@+id/rating"
        style="?android:attr/ratingBarStyleSmall"
        android:layout_below="@+id/author"
        android:layout_alignLeft="@+id/author"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:paddingTop="6dip"
        android:paddingBottom="20dip"
        android:stepSize="0.25"
        android:numStars="5"
        />
</RelativeLayout>
```