Jake Lambert

Student Number: Q10158341

\_

Topic: Budget Planner

SOUTHAMPTON SOLENT UNIVERSITY

SCHOOL OF MEDIA ARTS & TECHNOLOGY

BSc (Hons) Computing

Academic Year 2017- 2018

Table of Contents

[1 Introduction 3](#_Toc513465235)

[1.1 Background 3](#_Toc513465236)

[2 Aims and Objectives 5](#_Toc513465237)

[2.1 Aims 5](#_Toc513465238)

[2.2 Objective 5](#_Toc513465239)

[3 Incorporated Technologies 6](#_Toc513465240)

[3.1 Navigation Drawer with Fragments 6](#_Toc513465241)

[3.2 Toolbar with Action Buttons, Views and Icons 9](#_Toc513465242)

[3.3 SQLite Database with CRUD Operations 11](#_Toc513465243)

[4 Mobile Application Workflow 13](#_Toc513465244)

[5 Evaluation and Conclusion 14](#_Toc513465245)

[5.1 Evaluation 14](#_Toc513465246)

[5.2 Conclusion 14](#_Toc513465247)

[6 Recommendations for Further Work 15](#_Toc513465248)

[7 References 16](#_Toc513465249)

[8 Appendix A 17](#_Toc513465250)

[9 Appendix B 18](#_Toc513465251)

[9.1 MainActivity.java 18](#_Toc513465252)

[9.2 MyHelper.java 22](#_Toc513465253)

[9.3 TransactionsFragment.java 23](#_Toc513465254)

[9.4 MonthsFragment.java 24](#_Toc513465255)

[9.5 activity\_main.xml 25](#_Toc513465256)

[9.6 fragment\_dashboard.xml 26](#_Toc513465257)

[9.7 fragment\_months.xml 27](#_Toc513465258)

[9.8 fragment\_transactions.xml 28](#_Toc513465259)

[9.9 nav\_header.xml 31](#_Toc513465260)

[9.10 drawer\_menu.xml 32](#_Toc513465261)

[9.11 menu.xml 33](#_Toc513465262)

[9.12 styles.xml 34](#_Toc513465263)

[9.13 AndroidManifest.xml 35](#_Toc513465264)

# 1 Introduction

## 1.1 Background

For this project we have developed an Android mobile application for budgeting money. This app will give the user the ability to plan out their incomings and outgoings for a given month, it features a visual experience by reading numbers to see where your money is going. This application was developed in Android Studio and has a good user interface and functionality.

The reason a budget planner was chosen is that not everyone has mobile banking on their phone, so this application is a good alternative to manage your money. In a recent study, for Mobile App Categories on Which Internet Users Spend the Most Time, they found that only 20% spend time on their mobile banking (eMarketer 2017a). Also, they found that the bank app download rate was highest amongst youngsters, aged 18 - 24, 96% of whom had installed a banking app (eMarketer 2017b). This means we can target this age group for our application, as the majority will have mobiles these days therefore likely to download our budget planner application.

To have easy navigation and be a user-friendly application, we have extended the budget planner app to incorporate a few Android mobile technologies. The application allows the user to enter in the money they spend each month and the money they earn each month, which then gets stored into a database using SQLite (Android Developer 2016e). This feature uses the CRUD operators (create, read, update, and delete) to retrieve and input data.

We searched the app store and found there is a lot of budget planner apps with all different features, we can take little features from each app and put our own twist on it.

# 2 Aims and Objectives

## 2.1 Aims

The aim of this project was to combine Android mobile technologies in developing the application, to further our knowledge in mobile application architecture and android API functionalities. Our project outcome was to create a fully functioning budget planner and to learn new Android technologies in the development of this project.

The three topics from the Engineering Mobile Applications course; Android 5 UIs, SQLite, and Dynamic User Interface with Fragments will be the Android technologies used to create this project. Furthermore, we will be extending these technologies with further online research to make our application have advanced features.

## 2.2 Objective

This project will be divided into multiple objectives to achieve the desired fully functional application.

1. Navigate pages through the Navigation Drawer with Fragments
2. Use a Toolbar to include features
3. Use Action Buttons with Icons to display in the Toolbar
4. Insert data into the SQLite database using the Floating Action Button
5. Use the CRUD operations to create, look up, update, and delete data
6. Search each month with the Search View
7. Use a Spinner to display specific options for the user to choose

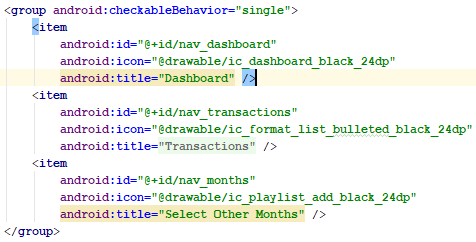
# 3 Incorporated Technologies

## 3.1 Navigation Drawer with Fragments

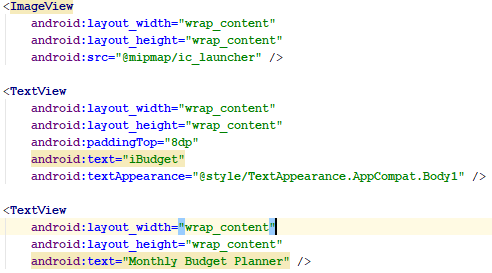
A navigation drawer is a great UI panel to keep things simple, it is hidden until the user swipes their finger to display the navigation menu (Android Developer 2016b).

We setup a menu file *drawer\_menu* to get our items into our navigation drawer.

We setup the items into a group and set the checkableBehaviour to “single” so only one item can be selected at a time. We assign an ID, icon and title to each item.



We must create the head of our navigation drawer. So, we need to setup a new layout file *nav\_header.*

We set the height of this header and add an image view and two text views to fill out the header with information to the user.   


Now we need to setup the drawer into the main layout file *activity\_main.* We change the route layout to a drawer layout, add in our toolbar and then add a FrameLayout which will hold our fragments (Android Developer 2016a).

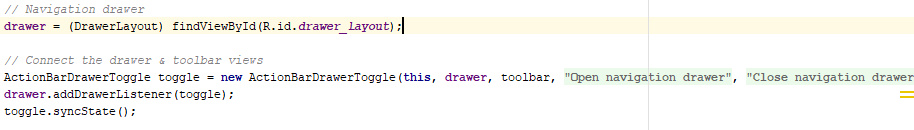


In our *MainActivity,* we set our toolbar as our action bar.

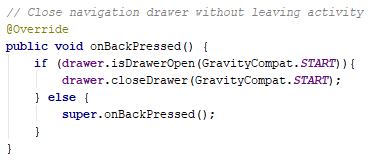


We setup a variable for our drawable layout, we assign this variable to the *drawer\_layout.* To get the menu button (hamburger icon) in the top right-hand corner to open our navigation drawer, we use the ActionBarDrawerToggle class where we pass in our drawer and toolbar variables as these are the two views that will be connected and synchronised. This code sets up a rotating hamburger icon.

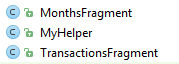
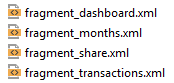




To close our navigation drawer without leaving the activity we need to call *OnBackPressed()*.



We need to make our items react to a click. Firstly, we need to create a fragment layout file for each item in the navigation drawer. Then setup a java class for each fragment item.

In each java class we call the LayoutInflater to link to their respective fragments layout files.



In our *MainActivity* we need to listen to click events on the navigation drawer, so we get the NavigationView from the *activity\_main* layout file.

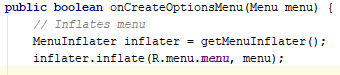


When then test to see which item is selected with *onNavigationItemSelected,* we test by the id of each item in the menu file.

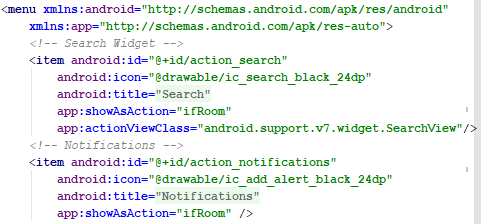


## 3.2 Toolbar with Action Buttons, Views and Icons

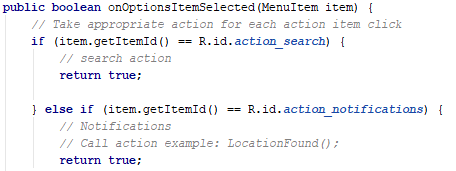
We have showed in the previous section how we setup the toolbar and how we call it in the *onCreate* method in our *MainActivity (Android Developer 2016d).* We do the standard inflating of our menu in the *MainActivity.*



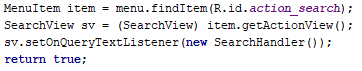
In the *menu* file, we add in our menu items. We assign them with an ID, icon, title and state whether we want them to be in the overflow or not using *showAsAction.* The icons we use in our toolbar are from [[1]](#footnote-1).



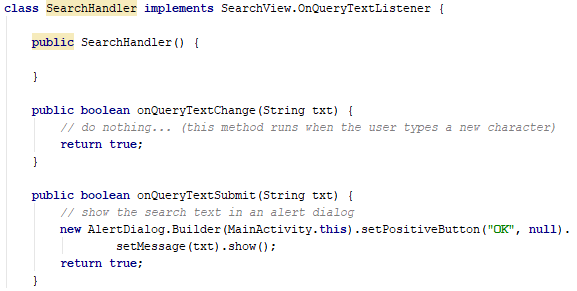
To take appropriate for each item selected we use onOptionsItemSelected.



For the search button we wanted to implement a *SearchView* as our *ActionView*, we need to attach event listeners to our *SearchView* when it is created. This will be in the *onCreateOptionsMenu*() method as that is when the XML is inflated. We use the *findItem*() method to find the menu item by it ID, then get the action view associated with that men item with the *MenuItemCompat.getActionView()(Android Developer 2016f).*



We then attach an *OnQueryTextListener* event handler to the *SearchView*. SearchHandler() is an inner class to implement the OnQueryTextListener interface.



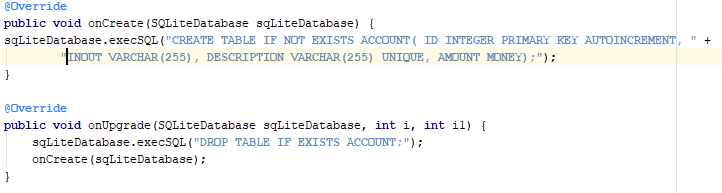
## 3.3 SQLite Database with CRUD Operations

Firstly, we need to create a layout file, so the user can enter data and submit actions using buttons. In the *fragments\_transactions* file we have:

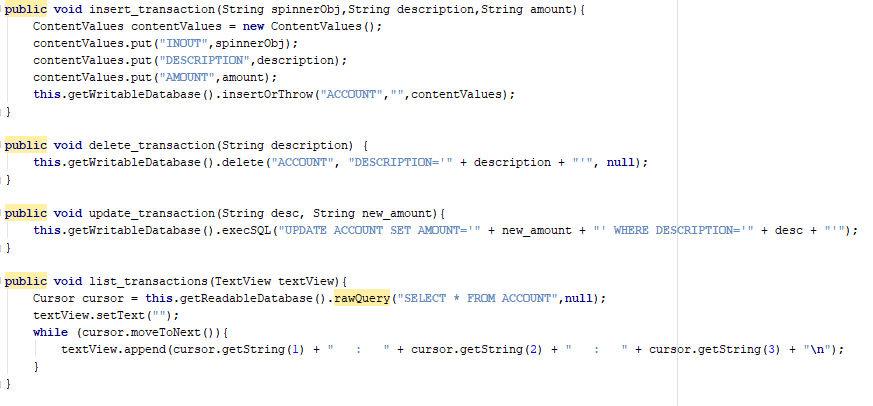
* Spinner
* Two edit texts
* Three buttons for Read, Update and Delete
* Floating action button for Insert

The three buttons have the same onClick listener that is defined as btn\_click.

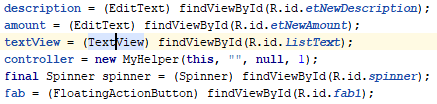
We setup an Activity class called *MyHelper*, this is where we control the access to the database via CRUD. In the onCreate() method we setup our database and database table (Zhao 2018).



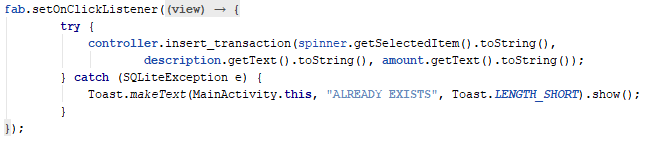
Next, we have the methods for the CRUD operations. For insert you simply pass in the variables and send them to the database using ContentValues. For delete you pass in the one variable then delete that row from the database. For update you pass in the *desc* and *new*\_*amount* variables and set the new\_amount where the *desc* variable matches to the database. For read we pass in the text view and read all data entries and display them in the text view.



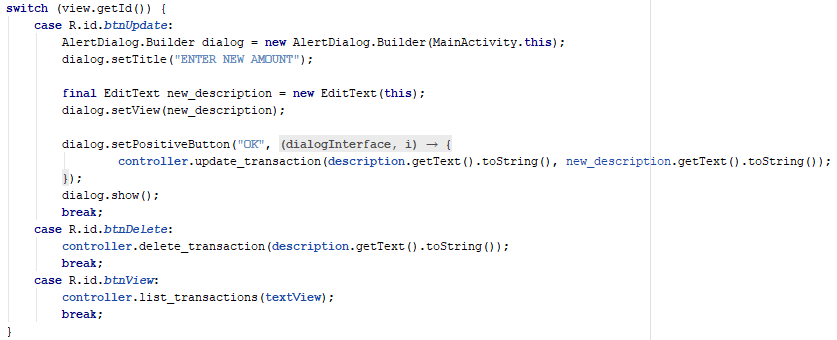
In our *MainActivity* we reference our MyHelper class and name the variable *controller*. We use a method that responds to the onClick of the buttons. In this method we link to the layout file using findViewById to store into variables



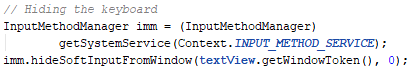
We set an on click listener to the floating action button variable. We use a try/catch block to send the data over to out MyHelper class, if there is an exception then we display an error message.



We run a switch statement to test to see which button has been clicked, then the data gets sent over to the MyHelper class.



This block of code hides the keyboard upon the user clicking a button, this prevents obscuring the users view.



# 4 Mobile Application Workflow

Once the application is open, the user will be presented with the home page (Appendix 9.1). To navigate to the Transactions page, the hamburger icon in the top left corner will have to be clicked, then the Transaction menu item will load in the transactions fragment page.

**Figure 1 - Home page w/ menu open**

On this page it will display the month. When the user wants to insert a transaction, they can choose an option from the dropdown menu and then enter a description of what that transaction is and the amount of it. To insert the transaction, you will need to click the floating action button, where it sends over the data and determined whether to insert or throw an exception.

**Figure 2 – Transactions page with data filled in**

This is what displays to the user if they try to input a transaction with the same description.

**Figure 3 – Already exists!**

When the user wants to view the current saved transactions, they will have to simply click the view button. This sends the text view over to the MyHelper class where it runs a select statement for all transactions.

**Figure 4 – Display all records**

To update a transaction, the user will have to specify the transaction by the description input field and click the update button. The user will then be displayed an edit text telling them to type in a new amount. To clarify, the update button is to only update the amount of the transaction.

**Figure 5 – Display the New Amount edit text**

To view the updated transaction the user can click the view button

**Figure 6 – Display the updated record**

To delete a transaction, the user will have to specify the transaction by the description input field and then click the delete button, thus that transaction will be deleted from the database.

**Figure 7 – Show deleting a record**

To show the deleted transaction has gone, the user can click the view button.

**Figure 8 – Show deleted record has gone**

# 5 Evaluation and Conclusion

This chapter will look at our previously set objectives and see whether they have been implemented successfully, then secondly will evaluate the project outcome.

## Evaluation

*Navigate pages through the Navigation Drawer with Fragments*

This objective was fulfilled by the DrawerLayout widget in the main activity layout and the navigation header layout file. We referenced these in our main activity where we ran a switch statement to check what item was selected, upon this it would load in the selected fragment class.

*Use a Toolbar to include features*

We have features on our toolbar, we have our toolbar setup in the main activity layout file. We have referenced out toolbar in the main activity and then inflated the toolbar.

*Use Action Buttons with Icons to display in the Toolbar*

In our menu file we have specified the menu items and assigned them all with icons. Also, we have specified whether they are to be in the overflow or not. In our main activity we have used an if statement to react to the action buttons being selected.

*Insert data into the SQLite database using the Floating Action Button*

This application allows the user to insert data using the floating action button, upon the user entering the details then sending them over to the MyHelper class, it will determine whether it is an existing record, if not then it will insert the record.

*Use the CRUD operations to create, look up, update, and delete data*

These CRUD operations were successfully fulfilled, they each have a button when clicked it sends the specified data the MyHelper class which determines the action required.

*Search each month with the Search View*

This was not fulfilled, we setup the search view where the user can type in anything, but there is nothing to search for. Due to time we couldn’t fulfil this objective.

*Use a Spinner to display specific options for the user to choose*

We successfully got the spinner working, it just displays the two options to the user. We used a spinner as it was the best tool for us to make sure the right data s inserted into the database.

## Conclusion

We didn’t manage to achieve all our objective but manged to implement 6/7 objectives into our application. The focus of our application to use CRUD operations in an SQLite database were accomplished, we have a good and clear system to use to keep up to date with managing your money.

Gantt chart

The report demonstrated the implementation with use of many Android APIs, combining these technologies to create a fully functional budget planner application. We have presented the development process with evidence and references. We have recommendations for further work in the next chapter.

# Recommendations for Further Work

Although most of our objectives were fulfilled apart from one, a budget planner application can use a few more Android technologies to add more advanced features. These can be implemented on top of the current application features.

Firstly, a great feature to use with SQLite would be 2D graphics, a topic we learnt from the engineering Mobile Applications course (Android Developers 2016). Although the user can see their transactions by looking at numbers, it would enhance their visual experience if they were able to view a bar chart of their transactions. You can form the bar chart from for example, the amount spent on food each month, then you can see if your food shops are on the increase or decrease. Overall, this would be a great way for the user to keep track of their transaction monthly. As you would normally write a rectangle using 2D Graphics by numbered values, instead you can get these values straight from the SQLite database (MJRoBot 2018).

Another Android technology to use would be Notifications (Android Developer 2016c). The application could notify the user that their phone bill will be coming out as a friendly reminder, so just to make sure they are aware and do not overspend. You can set it up for a specific day each month to notify the user.

Lastly, you can make this application more user-friendly if we had all the action linking to another page, but we don’t due to only having a few features.

# 7 References

ANDROID DEVELOPER, 2016a. *Building a Flexible UI*[viewed 3 May, 2018]. Available from: <https://developer.android.com/training/basics/fragments/fragment-ui>

ANDROID DEVELOPER, 2016b. *Create a navigation drawer*[viewed 4 May, 2018]. Available from: <https://developer.android.com/training/implementing-navigation/nav-drawer>

ANDROID DEVELOPER, 2016c. *Notifications Overview*[viewed Apr 23, 2018]. Available from: <https://developer.android.com/guide/topics/ui/notifiers/notifications>

ANDROID DEVELOPER, 2016d. *Set up the app bar*[viewed 2 May, 2018]. Available from: <https://developer.android.com/training/appbar/setting-up>

ANDROID DEVELOPER, 2016e. *SQLiteDatabase*Available from: <https://developer.android.com/reference/android/database/sqlite/SQLiteDatabase>

ANDROID DEVELOPER, 2016f. *Use action views and action providers*[viewed May 3, 2018]. Available from: <https://developer.android.com/training/appbar/action-views>

ANDROID DEVELOPERS, 2016. *Drawables overview*[viewed May 4, 2018]. Available from: <https://developer.android.com/guide/topics/graphics/drawables>

EMARKETER, 2017a. *Mobile App Categories on Which US Internet Users Spend the Most Time*[viewed May 2, 2018]. Available from: <https://www.emarketer.com/Chart/Mobile-App-Categories-on-Which-US-Internet-Users-Spend-Most-Time-Q2-2017-of-respondents/211534>

EMARKETER, 2017b. *Most People Have a Mobile Banking App, but Do They Use It*[viewed May 3, 2018]. Available from: <https://www.emarketer.com/Article/Most-People-Have-Mobile-Banking-App-Do-They-Use-It/1016690>

MJROBOT, *From Data to Graph: A Web Journey With Flask and SQLite*[viewed May 2, 2018]. Available from: <https://www.hackster.io/mjrobot/from-data-to-graph-a-web-journey-with-flask-and-sqlite-4dba35>

ZHAO, J., 2018. *Android SQLite CRUD Operations&nbsp;*[viewed May 2, 2018]. Available from: <https://www.dev2qa.com/android-sqlite-crud-operations-examples/>

# 8 Appendix A

# 9 Appendix B

## 9.1 MainActivity.java

package com.example.a0lambj41.budgetplannerapp;  
  
import android.content.Context;  
import android.content.DialogInterface;  
import android.content.Intent;  
import android.database.sqlite.SQLiteException;  
import **android.support.annotation.NonNull**;  
import android.support.design.widget.FloatingActionButton;  
import android.support.design.widget.NavigationView;  
import android.support.v4.view.GravityCompat;  
import android.support.v4.widget.DrawerLayout;  
import android.support.v7.app.ActionBarDrawerToggle;  
import android.support.v7.app.AlertDialog;  
import android.support.v7.app.AppCompatActivity;  
import android.os.Bundle;  
import android.support.v7.widget.SearchView;  
import android.support.v7.widget.Toolbar;  
import android.view.Menu;  
import android.view.MenuInflater;  
import android.view.MenuItem;  
import android.view.View;  
import android.view.inputmethod.InputMethodManager;  
import android.widget.EditText;  
import android.widget.Spinner;  
import android.widget.TextView;  
import android.widget.Toast;  
  
  
public class MainActivity extends AppCompatActivity implements NavigationView.OnNavigationItemSelectedListener {  
  
 FloatingActionButton fab;  
 private DrawerLayout drawer;  
  
 EditText transaction, description, amount;  
 TextView textView;  
 MyHelper controller;  
 Spinner spinner;  
  
 **@Override** protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
  
 // Toolbar  
 Toolbar toolbar = (Toolbar) findViewById(R.id.*toolbar1*);  
 setSupportActionBar(toolbar);  
  
 // Navigation drawer  
 drawer = (DrawerLayout) findViewById(R.id.*drawer\_layout*);  
 // Listen to click events on navigation view  
 NavigationView navigationView = (NavigationView) findViewById(R.id.*nav\_view*);  
 navigationView.setNavigationItemSelectedListener(this);  
  
 // Connect the drawer & toolbar views  
 ActionBarDrawerToggle toggle = new ActionBarDrawerToggle(this, drawer, toolbar, R.string.*navigation\_drawer\_open*, R.string.*navigation\_drawer\_close*);  
 drawer.addDrawerListener(toggle);  
 toggle.syncState();  
  
 }  
  
 public void btn\_click(View view) {  
  
 // SQLite g  
 description = (EditText) findViewById(R.id.*etNewDescription*);  
 amount = (EditText) findViewById(R.id.*etNewAmount*);  
 textView = (TextView) findViewById(R.id.*listText*);  
  
 controller = new MyHelper(this, **""**, null, 1);  
  
 final Spinner spinner = (Spinner) findViewById(R.id.*spinner*);  
  
 // Floating action button  
 fab = (FloatingActionButton) findViewById(R.id.*fab1*);  
 fab.setOnClickListener(new View.OnClickListener() {  
 public void onClick(View view) {  
 try {  
 controller.insert\_transaction(spinner.getSelectedItem().toString(),  
 description.getText().toString(), amount.getText().toString());  
 } catch (SQLiteException e) {  
 Toast.*makeText*(MainActivity.this, **"ALREADY EXISTS"**, Toast.*LENGTH\_SHORT*).show();  
 }  
 }  
 });  
  
 // Hiding the keyboard  
 InputMethodManager imm = (InputMethodManager)  
 getSystemService(Context.*INPUT\_METHOD\_SERVICE*);  
 imm.hideSoftInputFromWindow(textView.getWindowToken(), 0);  
  
 switch (view.getId()) {  
 case R.id.*btnUpdate*:  
 AlertDialog.Builder dialog = new AlertDialog.Builder(MainActivity.this);  
 dialog.setTitle(**"ENTER NEW AMOUNT"**);  
  
 final EditText new\_description = new EditText(this);  
 dialog.setView(new\_description);  
  
 dialog.setPositiveButton(**"OK"**, new DialogInterface.OnClickListener() {  
 **@Override** public void onClick(DialogInterface dialogInterface, int i) {  
 controller.update\_transaction(description.getText().toString(), new\_description.getText().toString());  
 }  
 });  
 dialog.show();  
 break;  
 case R.id.*btnDelete*:  
 controller.delete\_transaction(description.getText().toString());  
 break;  
 case R.id.*btnView*:  
 controller.list\_transactions(textView);  
 break;  
 }  
 }  
  
  
 // Close navigation drawer without leaving activity  
 **@Override** public void onBackPressed() {  
 if (drawer.isDrawerOpen(GravityCompat.*START*)) {  
 drawer.closeDrawer(GravityCompat.*START*);  
 } else {  
 super.onBackPressed();  
 }  
 }  
  
 **@Override** public boolean onCreateOptionsMenu(Menu menu) {  
 // Inflates menu  
 MenuInflater inflater = getMenuInflater();  
 inflater.inflate(R.menu.*menu*, menu);  
  
 MenuItem item = menu.findItem(R.id.*action\_search*);  
 SearchView sv = (SearchView) item.getActionView();  
 sv.setOnQueryTextListener(new SearchHandler());  
 return true;  
 }  
  
 **@Override** public boolean onNavigationItemSelected(**@NonNull** MenuItem item) {  
 switch (item.getItemId()) {  
 case R.id.*nav\_dashboard*:  
 Intent intent = new Intent(this, MainActivity.class);  
 startActivityForResult(intent, 0);  
 return true;  
 case R.id.*nav\_transactions*:  
 getSupportFragmentManager().beginTransaction().replace(R.id.*fragment\_container*, new TransactionsFragment()).commit();  
 break;  
 case R.id.*nav\_months*:  
 getSupportFragmentManager().beginTransaction().replace(R.id.*fragment\_container*, new MonthsFragment()).commit();  
 break;  
 case R.id.*nav\_share*:  
 Toast.*makeText*(this, **"Share"**, Toast.*LENGTH\_SHORT*).show();  
 break;  
 }  
  
 drawer.closeDrawer(GravityCompat.*START*);  
 return true;  
 }  
  
  
 // Handles the SearchView  
 class SearchHandler implements SearchView.OnQueryTextListener {  
  
 public SearchHandler() {  
  
 }  
  
 public boolean onQueryTextChange(String txt) {  
 // do nothing... (this method runs when the user types a new character)  
 return true;  
 }  
  
 public boolean onQueryTextSubmit(String txt) {  
 // show the search text in an alert dialog  
 new AlertDialog.Builder(MainActivity.this).setPositiveButton(**"OK"**, null).  
 setMessage(txt).show();  
 return true;  
 }  
 }  
  
 // On selecting action bar icons  
 **@Override** public boolean onOptionsItemSelected(MenuItem item) {  
 // Take appropriate action for each action item click  
 if (item.getItemId() == R.id.*action\_search*) {  
 // search action  
 return true;  
  
 } else if (item.getItemId() == R.id.*action\_notifications*) {  
 // Notifications  
 // Call action example: LocationFound();  
 return true;  
 } else if (item.getItemId() == R.id.*action\_transactions*) {  
 // Transactions  
 // Call action example: LocationFound();  
 return true;  
 } else if (item.getItemId() == R.id.*action\_analytics*) {  
 // Analytics  
 return true;  
 } else if (item.getItemId() == R.id.*action\_rate\_us*) {  
 // Rate us  
 return true;  
 }  
 return false;  
 }  
  
}

## 9.2 MyHelper.java

package com.example.a0lambj41.budgetplannerapp;  
  
import android.content.ContentValues;  
import android.content.Context;  
import android.database.Cursor;  
import android.database.sqlite.SQLiteDatabase;  
import android.database.sqlite.SQLiteOpenHelper;  
import android.widget.TextView;  
  
*/\*\*  
 \* Created by 0lambj41 on 25/04/2018.  
 \*/*public class MyHelper extends SQLiteOpenHelper{  
 public MyHelper(Context context, String name, SQLiteDatabase.CursorFactory factory, int version) {  
 super(context, **"budgetplanner.db"**, factory, version);  
 }  
  
 **@Override** public void onCreate(SQLiteDatabase sqLiteDatabase) {  
 sqLiteDatabase.execSQL(**"CREATE TABLE IF NOT EXISTS ACCOUNT( ID INTEGER PRIMARY KEY AUTOINCREMENT, INOUT VARCHAR(255), DESCRIPTION VARCHAR(255) UNIQUE, AMOUNT MONEY);"**);  
 }  
  
 **@Override** public void onUpgrade(SQLiteDatabase sqLiteDatabase, int i, int i1) {  
 sqLiteDatabase.execSQL(**"DROP TABLE IF EXISTS ACCOUNT;"**);  
 onCreate(sqLiteDatabase);  
 }  
  
 public void insert\_transaction(String spinnerObj,String description,String amount){  
 ContentValues contentValues = new ContentValues();  
 contentValues.put(**"INOUT"**,spinnerObj);  
 contentValues.put(**"DESCRIPTION"**,description);  
 contentValues.put(**"AMOUNT"**,amount);  
 this.getWritableDatabase().insertOrThrow(**"ACCOUNT"**,**""**,contentValues);  
 }  
  
 public void delete\_transaction(String description) {  
 this.getWritableDatabase().delete(**"ACCOUNT"**, **"DESCRIPTION='"** + description + **"'"**, null);  
 }  
  
 public void update\_transaction(String desc, String new\_amount){  
 this.getWritableDatabase().execSQL(**"UPDATE ACCOUNT SET AMOUNT='"** + new\_amount + **"' WHERE DESCRIPTION='"** + desc + **"'"**);  
 }  
  
 public void list\_transactions(TextView textView){  
 Cursor cursor = this.getReadableDatabase().rawQuery(**"SELECT \* FROM ACCOUNT"**,null);  
 textView.setText(**""**);  
 while (cursor.moveToNext()){  
 textView.append(cursor.getString(1) + **" : "** + cursor.getString(2) + **" : "** + cursor.getString(3) + **"**\n**"**);  
 }  
 }  
}

## 9.3 TransactionsFragment.java

package com.example.a0lambj41.budgetplannerapp;  
  
import android.os.Bundle;  
import **android.support.annotation.Nullable**;  
import android.support.v4.app.Fragment;  
import android.view.LayoutInflater;  
import android.view.View;  
import android.view.ViewGroup;  
import android.widget.ArrayAdapter;  
import android.widget.Spinner;  
  
import static com.example.a0lambj41.budgetplannerapp.R.id.*spinner*;  
  
*/\*\*  
 \* Created by 0lambj41 on 17/04/2018.  
 \*/*public class TransactionsFragment extends Fragment {  
  
 **@Nullable  
 @Override** public View onCreateView(LayoutInflater inflater, **@Nullable** ViewGroup container, **@Nullable** Bundle savedInstanceState) {  
 return inflater.inflate(R.layout.*fragment\_transactions*, container, false);  
 }  
  
 public void onActivityCreated(Bundle savedInstanceState) {  
 super.onActivityCreated(savedInstanceState);  
  
 }  
}

## 9.4 MonthsFragment.java

package com.example.a0lambj41.budgetplannerapp;  
  
import android.os.Bundle;  
import **android.support.annotation.Nullable**;  
import android.support.v4.app.Fragment;  
import android.view.LayoutInflater;  
import android.view.View;  
import android.view.ViewGroup;  
  
*/\*\*  
 \* Created by 0lambj41 on 17/04/2018.  
 \*/*public class MonthsFragment extends Fragment {  
  
 **@Nullable  
 @Override** public View onCreateView(LayoutInflater inflater, **@Nullable** ViewGroup container, **@Nullable** Bundle savedInstanceState) {  
 return inflater.inflate(R.layout.*fragment\_months*, container, false);  
 }  
}

## 9.5 activity\_main.xml

<?xml version=**"1.0"** encoding=**"utf-8"**?>  
  
<android.support.v4.widget.DrawerLayout xmlns:android=**"http://schemas.android.com/apk/res/android"** xmlns:app=**"http://schemas.android.com/apk/res-auto"** xmlns:tools=**"http://schemas.android.com/tools"** android:id=**"@+id/drawer\_layout"** android:layout\_width=**"match\_parent"** android:layout\_height=**"match\_parent"** android:fitsSystemWindows=**"true"** tools:openDrawer=**"start"** tools:context=**"com.example.a0lambj41.budgetplannerapp.MainActivity"**>  
  
<RelativeLayout xmlns:android=**"http://schemas.android.com/apk/res/android"** android:layout\_width=**"match\_parent"** android:layout\_height=**"match\_parent"** xmlns:app=**"http://schemas.android.com/apk/res-auto"** android:scrollbars=**"vertical|horizontal"** >  
  
 <android.support.v7.widget.Toolbar  
 android:layout\_width=**"match\_parent"** android:layout\_height=**"wrap\_content"** android:background=**"@color/colorPrimary"** android:text=**"@string/app\_name"** app:theme=**"@style/ThemeOverlay.AppCompat.Dark.ActionBar"** app:popupTheme=**"@style/ThemeOverlay.AppCompat.Light"** android:id=**"@+id/toolbar1"** android:elevation=**"4dp"**/>  
  
 <FrameLayout  
 android:id=**"@+id/fragment\_container"** android:layout\_width=**"match\_parent"** android:layout\_height=**"match\_parent"** android:layout\_below=**"@id/toolbar1"**>  
 </FrameLayout>  
<LinearLayout  
 android:layout\_width=**"match\_parent"** android:layout\_height=**"match\_parent"** android:orientation=**"vertical"** android:id=**"@+id/rectangle"** >  
  
</LinearLayout>  
  
  
</RelativeLayout>  
  
 <android.support.design.widget.NavigationView  
 android:id=**"@+id/nav\_view"** android:layout\_width=**"wrap\_content"** android:layout\_height=**"match\_parent"** android:layout\_gravity=**"start"** app:headerLayout=**"@layout/nav\_header"** app:menu=**"@menu/drawer\_menu"**>  
  
 </android.support.design.widget.NavigationView>  
  
</android.support.v4.widget.DrawerLayout>

## 9.6 fragment\_dashboard.xml

<?xml version=**"1.0"** encoding=**"utf-8"**?>  
<RelativeLayout xmlns:android=**"http://schemas.android.com/apk/res/android"** android:layout\_width=**"match\_parent"** android:layout\_height=**"match\_parent"** android:background=**"@android:color/holo\_blue\_bright"**>  
  
 <TextView  
 android:layout\_width=**"wrap\_content"** android:layout\_height=**"wrap\_content"** android:text=**"Dashboard Fragment"** android:textSize=**"28sp"** android:layout\_centerInParent=**"true"**/>  
  
</RelativeLayout>

## 9.7 fragment\_months.xml

<?xml version=**"1.0"** encoding=**"utf-8"**?>  
<RelativeLayout xmlns:android=**"http://schemas.android.com/apk/res/android"** android:layout\_width=**"match\_parent"** android:layout\_height=**"match\_parent"** android:background=**"@android:color/holo\_purple"**>  
  
 <TextView  
 android:layout\_width=**"wrap\_content"** android:layout\_height=**"wrap\_content"** android:text=**"Months Fragment"** android:textSize=**"28sp"** android:layout\_centerInParent=**"true"**/>  
  
</RelativeLayout>

## 9.8 fragment\_transactions.xml

<?xml version=**"1.0"** encoding=**"utf-8"**?>  
<RelativeLayout xmlns:android=**"http://schemas.android.com/apk/res/android"** xmlns:app=**"http://schemas.android.com/apk/res-auto"** android:layout\_width=**"match\_parent"** android:layout\_height=**"match\_parent"** xmlns:tools=**"http://schemas.android.com/tools"** android:background=**"@android:color/white"** android:padding=**"16dp"** tools:context=**"com.example.a0lambj41.budgetplannerapp.TransactionsFragment"**>  
  
 <TextView  
 android:id=**"@+id/textView"** android:layout\_width=**"wrap\_content"** android:layout\_height=**"wrap\_content"** android:text=**"May"** android:textColor=**"@android:color/black"** android:textSize=**"20sp"** android:textStyle=**"bold"** android:layout\_marginTop=**"12dp"** android:layout\_alignParentTop=**"true"** android:layout\_alignParentLeft=**"true"** android:layout\_alignParentStart=**"true"** />  
  
 <TextView  
 android:id=**"@+id/textView2"** android:layout\_width=**"wrap\_content"** android:layout\_height=**"wrap\_content"** android:text=**"Transaction"** android:textColor=**"@android:color/black"** android:textSize=**"16sp"** android:textStyle=**"italic"** android:layout\_marginTop=**"13dp"** android:layout\_below=**"@+id/textView"** android:layout\_alignLeft=**"@+id/textView"** android:layout\_alignStart=**"@+id/textView"** />  
  
 <Spinner  
 android:layout\_width=**"match\_parent"** android:layout\_height=**"wrap\_content"** android:id=**"@+id/spinner"** android:entries=**"@array/spinner\_array"** android:layout\_below=**"@+id/textView2"** android:layout\_alignParentLeft=**"true"** android:layout\_alignParentStart=**"true"** android:layout\_marginTop=**"11dp"** />  
  
 <TextView  
 android:id=**"@+id/textView3"** android:layout\_width=**"wrap\_content"** android:layout\_height=**"wrap\_content"** android:text=**"Description"** android:textColor=**"@android:color/black"** android:textSize=**"16sp"** android:textStyle=**"italic"** android:layout\_marginTop=**"12dp"** android:layout\_below=**"@+id/spinner"** android:layout\_alignLeft=**"@+id/spinner"** android:layout\_alignStart=**"@+id/spinner"** />  
  
 <EditText  
 android:id=**"@+id/etNewDescription"** android:layout\_width=**"match\_parent"** android:layout\_height=**"wrap\_content"** android:hint=**"Salary, Rent.. "** android:layout\_below=**"@+id/textView3"** android:layout\_alignLeft=**"@+id/textView3"** android:layout\_alignStart=**"@+id/textView3"** />  
  
 <TextView  
 android:id=**"@+id/textView4"** android:layout\_width=**"wrap\_content"** android:layout\_height=**"wrap\_content"** android:text=**"Amount"** android:textColor=**"@android:color/black"** android:textSize=**"16sp"** android:textStyle=**"italic"** android:layout\_marginTop=**"14dp"** android:layout\_below=**"@+id/etNewDescription"** android:layout\_alignLeft=**"@+id/etNewDescription"** android:layout\_alignStart=**"@+id/etNewDescription"** />  
  
 <EditText  
 android:id=**"@+id/etNewAmount"** android:layout\_width=**"match\_parent"** android:layout\_height=**"wrap\_content"** android:hint=**"£600"** android:inputType=**"numberDecimal"** android:layout\_below=**"@+id/textView4"** android:layout\_alignLeft=**"@+id/textView4"** android:layout\_alignStart=**"@+id/textView4"** />  
  
 <Button  
 android:id=**"@+id/btnUpdate"** android:layout\_width=**"wrap\_content"** android:layout\_height=**"wrap\_content"** android:text=**"Update"** android:onClick=**"btn\_click"** android:layout\_below=**"@+id/etNewAmount"** />  
  
 <Button  
 android:id=**"@+id/btnDelete"** android:layout\_width=**"wrap\_content"** android:layout\_height=**"wrap\_content"** android:text=**"Delete"** android:onClick=**"btn\_click"** android:layout\_below=**"@+id/etNewAmount"** android:layout\_toRightOf=**"@+id/btnUpdate"** android:layout\_toEndOf=**"@+id/btnUpdate"** />  
  
 <Button  
 android:id=**"@+id/btnView"** android:layout\_width=**"wrap\_content"** android:layout\_height=**"wrap\_content"** android:text=**"View"** android:onClick=**"btn\_click"** android:layout\_alignBaseline=**"@+id/btnDelete"** android:layout\_alignBottom=**"@+id/btnDelete"** android:layout\_toRightOf=**"@+id/btnDelete"** android:layout\_toEndOf=**"@+id/btnDelete"** />  
  
 <android.support.design.widget.FloatingActionButton  
 android:id=**"@+id/fab1"** android:layout\_width=**"wrap\_content"** android:layout\_height=**"wrap\_content"** android:layout\_alignParentRight=**"true"** android:layout\_alignParentBottom=**"true"** android:layout\_marginRight=**"24dp"** android:layout\_marginBottom=**"24dp"** app:elevation=**"6dp"** app:pressedTranslationZ=**"12dp"** android:src=**"@drawable/ic\_save\_black\_24dp"**/>  
  
 <TextView  
 android:layout\_width=**"match\_parent"** android:layout\_height=**"match\_parent"** android:textAppearance=**"?android:attr/textAppearanceLarge"** android:id=**"@+id/listText"** android:gravity=**"top|center"** android:layout\_marginTop=**"17dp"** android:layout\_below=**"@+id/btnUpdate"** android:layout\_alignLeft=**"@+id/btnUpdate"** android:layout\_alignStart=**"@+id/btnUpdate"** />  
  
  
  
  
  
</RelativeLayout>

## 9.9 nav\_header.xml

<?xml version=**"1.0"** encoding=**"utf-8"**?>  
<LinearLayout xmlns:android=**"http://schemas.android.com/apk/res/android"** android:layout\_width=**"match\_parent"** android:layout\_height=**"176dp"** android:background=**"@color/colorPrimary"** android:gravity=**"bottom"** android:orientation=**"vertical"** android:padding=**"16dp"** android:theme=**"@style/ThemeOverlay.AppCompat.Dark"**>  
  
 <ImageView  
 android:layout\_width=**"wrap\_content"** android:layout\_height=**"wrap\_content"** android:src=**"@mipmap/ic\_launcher"** />  
  
 <TextView  
 android:layout\_width=**"wrap\_content"** android:layout\_height=**"wrap\_content"** android:paddingTop=**"8dp"** android:text=**"iBudget"** android:textAppearance=**"@style/TextAppearance.AppCompat.Body1"** />  
  
 <TextView  
 android:layout\_width=**"wrap\_content"** android:layout\_height=**"wrap\_content"** android:text=**"Monthly Budget Planner"** />  
  
</LinearLayout>

## 9.10 drawer\_menu.xml

<?xml version=**"1.0"** encoding=**"utf-8"**?>  
<menu xmlns:android=**"http://schemas.android.com/apk/res/android"** xmlns:tools=**"http://schemas.android.com/tools"** tools:showIn=**"navigation\_view"**>  
  
 <group android:checkableBehavior=**"single"**>  
 <item  
 android:id=**"@+id/nav\_dashboard"** android:icon=**"@drawable/ic\_dashboard\_black\_24dp"** android:title=**"Dashboard"** />  
 <item  
 android:id=**"@+id/nav\_transactions"** android:icon=**"@drawable/ic\_format\_list\_bulleted\_black\_24dp"** android:title=**"@string/action\_transactions"** />  
 <item  
 android:id=**"@+id/nav\_months"** android:icon=**"@drawable/ic\_playlist\_add\_black\_24dp"** android:title=**"Select Other Months"** />  
 </group>  
  
 <item android:title=**"Communicate"**>  
 <menu>  
 <item  
 android:id=**"@+id/nav\_share"** android:icon=**"@drawable/ic\_share\_black\_24dp"** android:title=**"share"** />  
 </menu>  
 </item>  
  
</menu>

## 9.11 menu.xml

<?xml version=**"1.0"** encoding=**"utf-8"**?>  
<menu xmlns:android=**"http://schemas.android.com/apk/res/android"** xmlns:app=**"http://schemas.android.com/apk/res-auto"**>  
 <!-- Search Widget -->  
 <item android:id=**"@+id/action\_search"** android:icon=**"@drawable/ic\_search\_black\_24dp"** android:title=**"@string/action\_search"** app:showAsAction=**"ifRoom"** app:actionViewClass=**"android.support.v7.widget.SearchView"**/>  
 <!-- Notifications -->  
 <item android:id=**"@+id/action\_notifications"** android:icon=**"@drawable/ic\_add\_alert\_black\_24dp"** android:title=**"@string/action\_notifications"** app:showAsAction=**"ifRoom"** />  
 <!-- Transactions -->  
 <item android:id=**"@+id/action\_transactions"** android:icon=**"@drawable/ic\_payment\_black\_24dp"** android:title=**"@string/action\_transactions"** app:showAsAction=**"ifRoom"** />  
 <!-- Analytics -->  
 <item android:id=**"@+id/action\_analytics"** android:icon=**"@drawable/ic\_pie\_chart\_outlined\_black\_24dp"** android:title=**"@string/action\_analytics"** app:showAsAction=**"never"**/>  
 <!-- Rate us -->  
 <item android:id=**"@+id/action\_rate\_us"** android:icon=**"@drawable/ic\_star\_rate\_black\_18dp"** android:title=**"@string/action\_rate\_us"** app:showAsAction=**"never"** />  
  
</menu>

## 9.12 styles.xml

<resources>  
  
 <!-- Base application theme. -->  
 <style name=**"AppTheme"** parent=**"Theme.AppCompat.Light.NoActionBar"**>  
 <!-- Customize your theme here. -->  
 <item name=**"colorPrimary"**>@color/colorPrimary</item>  
 <item name=**"colorPrimaryDark"**>@color/colorPrimaryDark</item>  
 <item name=**"colorAccent"**>@color/colorAccent</item>  
 </style>  
  
 <style name=**"AppTheme.NoActionBar"**>  
 <item name=**"windowActionBar"**>false</item>  
 <item name=**"windowNoTitle"**>true</item>  
 </style>  
  
</resources>

## 9.13 AndroidManifest.xml

<?xml version=**"1.0"** encoding=**"utf-8"**?>  
<manifest xmlns:android=**"http://schemas.android.com/apk/res/android"** package=**"com.example.a0lambj41.budgetplannerapp"** >  
  
 <application  
 android:allowBackup=**"true"** android:icon=**"@mipmap/ic\_launcher"** android:label=**"@string/app\_name"** android:supportsRtl=**"true"** android:theme=**"@style/Theme.AppCompat.Light.NoActionBar"**>  
 <activity android:name=**".MainActivity"** >  
 <intent-filter>  
 <action android:name=**"android.intent.action.MAIN"** />  
  
 <category android:name=**"android.intent.category.LAUNCHER"** />  
 </intent-filter>  
 </activity>  
 </application>  
  
</manifest>

1. [https://material.io/icons/](https://material.io/icons/%20#ic_create) [↑](#footnote-ref-1)