Report On

**EXPENSE TRACKER**

**MCC**

By

|  |  |  |  |
| --- | --- | --- | --- |
| **MEGH SHAH** | **TE CMPN A** | **TCA 4** | **63** |
| **AKSHAY SHALIGRAM** | **TE CMPN A** | **TCA 4** | **64** |
| **PRIYANSHI SHUKLA**  **HIMANSHU SINGH**  **AYUSH TICKOO** | **TE CMPN A**  **TE CMPN A**  **TE CMPN A** | **TCA 4**  **TCA 4**  **TCA 4** | **66**  **67**  **69** |



Department of Computer Engineering

St. Francis Institute of Technology

UniversityofMumbai

(2015-2016)

INTERNAL EXAMINER: EXTERNAL EXAMINER:

**DECLARATION**

This to certify that the Project report on ‘**EXPENSE TRACKER’**has been carried out by **MEGH SHAH (TE CMPN A 63), AKSHAY SHALIGRAM (TE CMPN A 64),PRIYANSHI SHUKLA (TE CMPN A 66), HIMANSHU SINGH (TE CMPN A 67)** and **AYUSH TICKOO (TE CMPN A 69)** who are bonafide students of St. Francis Institute of Technology, Mumbai in partial fulfillment of the requirement of T. E. degree in Computer Engineering at St. Francis Institute of Technology, Mumbai, India. It is also certified that this work has not been presented anywhere else for award of any other degree or diploma prior to this.

INDEX

1Introduction……………………………………………………………………………………….……4

1.1Problem Statement…………………………………………………………………………………….4

2 MODULES………………………………………………………………………………………………….…5

3 IMPLEMENTATION………………………………………………………………………...……………...6

3.1 TECHNOLOGIES USED…………………...…………………………...................................................................6

3.2 CODE…………………………………………………………………………………………………………………………………….6

**4RESULT………………………………………………………………………………………………………..12**

4.1SCREENSHOTS**………………………………………………………….…………………………………………………………14**

**Introduction:**

With ever increasing expenses it is a necessity to keep a track of the expenses so as to better manage them and use the finances as optimally as possible.

**Problem Statement:**

* The aim is to build a type of tracker that keeps a record of monthly expenses.
* The *ExpenseTracker* is aimed to be a comprehensive tracker that helps manage day to day finance by keeping a track of all your expenses throughout the month.
* This application will act as a personal finance data evidence for the user, where the user can log all the expenses in categories that user creates and then accordingly plan his/her further expenditures.
* It will also act as a calculator for tracking and balancing group expenses.
* The Custom Split option will tell exactly how much each person in the group paid and consumed.
* Those who paid too little pay the difference and those who paid too much take out theirs, and the score is settled.

**Modules:**

* **New Calculation:**

This helps user to create a finance log for any particular month.

The user puts in the month name, currency, type of expenses list as input.

After entering necessary input the user gets a display of all the categories that he has made and all the expenditures done in each category is also shown for the mentioned month.

* **New Expense:**

This helps the user add in new expenditures by specifying the name, amount spent (along with currency type), date and type of the expenditure.

This also allows user to add the option of Custom Split in order to put in the split ratio of expenditure among 2 or more people in a group.

* **Deletion:**

As the name suggests, this helps to delete a particular entry of expenditure or delete an entire month of expenditure.

* **Summary:**

This module gives a brief description of the expenditure in a particular month.

It states the initial and final date entries, duration, expenses and total amount spent.

* **MoneyBalance:**

This module gives an overview of expenditure entries made in all the months up until the most recent one.

**Implementation Details:**

* **Technologies used:**

-Android Studio is used for developing the user interface. Android platform is chosen as the target OS for running the application because of its widespread reach to the audience.

-Database is stored as internal storage.

-Internet connection is not mandatory for the application to run.

* **Code:**

CalculationListActivity.java:

package ivl.android.moneybalance;

import ivl.android.moneybalance.dao.CalculationDataSource;

import ivl.android.moneybalance.dao.DataBaseHelper;

import ivl.android.moneybalance.data.Calculation;

import ivl.android.moneybalance.data.Expense;

import ivl.android.moneybalance.data.Person;

import java.text.SimpleDateFormat;

import java.util.List;

import android.app.AlertDialog;

import android.content.Context;

import android.content.DialogInterface;

import android.content.Intent;

import android.database.Cursor;

import android.os.Bundle;

import android.support.v4.view.WindowCompat;

import android.support.v4.widget.CursorAdapter;

import android.support.v7.app.ActionBarActivity;

import android.view.ContextMenu;

import android.view.ContextMenu.ContextMenuInfo;

import android.view.LayoutInflater;

import android.view.Menu;

import android.view.MenuItem;

import android.view.View;

import android.view.ViewGroup;

import android.widget.AdapterView;

import android.widget.AdapterView.OnItemClickListener;

import android.widget.ListView;

import android.widget.TextView;

public class CalculationListActivity extends ActionBarActivity implements OnItemClickListener {

private final DataBaseHelper dbHelper = new DataBaseHelper(this);

private final CalculationDataSource dataSource = new CalculationDataSource(dbHelper);

private Cursor cursor;

private ListView listView;

private CalculationAdapter adapter;

private static final int ITEM\_DELETE = 0;

private static final int ITEM\_SUMMARY = 1;

private class CalculationAdapter extends CursorAdapter {

private final String summaryFormat = getResources().getString(R.string.expenses\_summary\_format);

private final String dateRangeFormat = getResources().getString(R.string.date\_range\_format);

CalculationAdapter(Context context) {

super(context, null, 0);

}

@Override

public View newView(Context context, Cursor cursor, ViewGroup parent) {

// TODO: Use view holder

LayoutInflater inflater = (LayoutInflater) context.getSystemService(Context.LAYOUT\_INFLATER\_SERVICE);

return inflater.inflate(R.layout.calculation\_list\_row, parent, false);

}

@Override

public void bindView(View view, Context context, Cursor cursor) {

Calculation calculation = dataSource.fromCursor(cursor);

StringBuilder personNames = new StringBuilder();

for (Person person : calculation.getPersons()) {

if (personNames.length() > 0) {

personNames.append(", ");

}

personNames.append(person.getName());

}

TextView titleView = (TextView) view.findViewById(R.id.calculation\_title);

titleView.setText(calculation.getTitle());

TextView personsView = (TextView) view.findViewById(R.id.calculation\_persons);

personsView.setText(personNames);

TextView datesView = (TextView) view.findViewById(R.id.calculation\_dates);

TextView summaryView = (TextView) view.findViewById(R.id.calculation\_summary);

List<Expense> expenses = calculation.getExpenses();

if (expenses.size() == 0) {

datesView.setVisibility(View.GONE);

summaryView.setText(R.string.no\_expenses);

} else {

SimpleDateFormat format = new SimpleDateFormat("yyyy-MM-dd");

String firstDate = format.format(calculation.getFirstDate().getTime());

String lastDate = format.format(calculation.getLastDate().getTime());

datesView.setText(String.format(dateRangeFormat, firstDate, lastDate));

datesView.setVisibility(View.VISIBLE);

int count = calculation.getExpenses().size();

CurrencyHelper helper = calculation.getMainCurrency().getCurrencyHelper();

String total = helper.format(calculation.getExpenseTotal());

summaryView.setText(String.format(summaryFormat, count, total));

}

}

};

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

supportRequestWindowFeature(WindowCompat.FEATURE\_ACTION\_BAR);

setContentView(R.layout.calculation\_list);

listView = (ListView) findViewById(R.id.calculation\_list);

adapter = new CalculationAdapter(this);

listView.setAdapter(adapter);

listView.setOnItemClickListener(this);

registerForContextMenu(listView);

setContentView(listView);

refresh();

}

private void refresh() {

cursor = dataSource.listAll();

adapter.changeCursor(cursor);

listView.setAdapter(adapter);

}

@Override

public void onItemClick(AdapterView<?> l, View v, int position, long id) {

cursor.moveToPosition(position);

Calculation calculation = dataSource.fromCursor(cursor);

Intent intent = new Intent(this, ExpenseListActivity.class);

intent.putExtra(ExpenseListActivity.PARAM\_CALCULATION\_ID, calculation.getId());

startActivity(intent);

}

@Override

public boolean onCreateOptionsMenu(Menu menu) {

getMenuInflater().inflate(R.menu.calculation\_list\_options, menu);

return true;

}

@Override

public boolean onOptionsItemSelected(MenuItem item) {

switch (item.getItemId()) {

case R.id.new\_calculation:

startActivity(new Intent(this, CalculationEditorActivity.class));

return true;

default:

return super.onOptionsItemSelected(item);

}

}

@Override

public void onCreateContextMenu(ContextMenu menu, View v, ContextMenuInfo menuInfo) {

if (v.getId() == R.id.calculation\_list) {

AdapterView.AdapterContextMenuInfo info = (AdapterView.AdapterContextMenuInfo) menuInfo;

cursor.moveToPosition(info.position);

Calculation calculation = dataSource.fromCursor(cursor);

menu.setHeaderTitle(calculation.getTitle());

menu.add(0, ITEM\_DELETE, 0, R.string.menu\_delete);

menu.add(0, ITEM\_SUMMARY, 0, R.string.calculation\_summary);

}

}

@Override

public boolean onContextItemSelected(MenuItem item) {

AdapterView.AdapterContextMenuInfo info = (AdapterView.AdapterContextMenuInfo) item.getMenuInfo();

cursor.moveToPosition(info.position);

long calculationId = cursor.getLong(0);

if (item.getItemId() == ITEM\_DELETE) {

confirmAndDelete(dataSource.get(calculationId));

return true;

} else if (item.getItemId() == ITEM\_SUMMARY) {

Intent intent = new Intent(this, SummaryActivity.class);

intent.putExtra(ExpenseListActivity.PARAM\_CALCULATION\_ID, calculationId);

startActivity(intent);

} else {

return false;

}

return true;

}

private void confirmAndDelete(final Calculation calculation) {

AlertDialog.Builder dialog = new AlertDialog.Builder(this);

dialog.setIcon(android.R.drawable.ic\_delete);

dialog.setTitle(calculation.getTitle());

dialog.setMessage(R.string.confirm\_delete\_calculation);

dialog.setPositiveButton(android.R.string.yes, new DialogInterface.OnClickListener() {

@Override

public void onClick(DialogInterface dialog, int which) {

dataSource.delete(calculation.getId());

refresh();

}

});

dialog.setNegativeButton(android.R.string.no, null);

dialog.show();

}

@Override

protected void onPause() {

super.onPause();

dbHelper.close();

}

@Override

protected void onResume() {

refresh();

super.onResume();

}

@Override

public void onBackPressed() {

new AlertDialog.Builder(this).setIcon(android.R.drawable.ic\_dialog\_alert).setTitle("Exit")

.setMessage("Are you sure ?")

.setPositiveButton("Yes", new DialogInterface.OnClickListener() {

@Override

public void onClick(DialogInterface dialog, int which) {

Intent intent = new Intent(Intent.ACTION\_MAIN);

intent.addCategory(Intent.CATEGORY\_HOME);

intent.setFlags(Intent.FLAG\_ACTIVITY\_NEW\_TASK);

startActivity(intent);

finish();

}

}).setNegativeButton("No", null).show();

}

}

**Result:**

* Snapshots:

 

 

 

 

 

 