Software Design Document

For

Transaction Tracker

An assistive tool for tracking expenses

Prepared by:
Rhadjni Phipps, Jason Manning,
Khamoi White, Dolieth Chambers

Contents

Project Contributions

Name / ID Number	**Document Contribution Headings**	**Software Contribution**
Rhadjni Phipps / 180918	Introductions, Special/Functionality, Product Perspective, User Classes & Characteristics, Overall Description	Expense fragment, bar graph analysis design, database schema, and Database Functionalities.
Dolieth Chambers / 180905	Product Features, Design & Implementation Constraints, System Features (Managing Transactions, Monitoring Expenses)	Affordable Fragment (Wheel and text animation), monitoring income fragment(pie chart)
Khamoi White / 180909	Assumptions & Dependencies, System Features (Calculate Cost, Graphical Representation of Expenses), External Interface Requirement	Interface Look & Feel, Settings, navigation drawer, main activity, notifications, calculator functionality
Jason Manning / 180906	System Features (Make Stochastic Decision, Managing expenses), Other Nonfunctional & Other Requirements, Use case Diagram	Daily Fragment, animation, splash screen

1. Introduction

1.1 Intended Audience and Reading Suggestions

This Software Design Document is intended for multiple interested parties. The team members of the TPGD development team will use this to aid in the development of the application by reviewing and implementing.
Downloaders and direct users who will overserve, comment upon and consult with development will also be given to the most updated version of this Software Design Document.
The main stakeholder of this Application, Thomas Xu, will have access to the final SDD.
All parties along with external stakeholders are the intended audience.

The Software Design Document contains a detailed description of the process of the software development along with the features and mandatory components. Detailed diagrams are used to show interaction within the application and entire system. This includes user interactions with hardware, software and external applications.

1.2 Project Scope

The Transaction Tracker application will enable users to have improved financial management by use of a mobile assistive tool. The application will provide a method for the user to track their expenses in a visual and textual format while also providing a method for making crucial purchase decisions.

1.3 References

The references used with the development of this SRS and the software are noted below:
2002 by Karl E. Wiegers

2. Overall Description

2.1 Product Perspective

The software, Transaction Tracker, will be a stand-alone mobile application developed using the JetBrains' IntelliJ IDEA powered software, Android Studio. The application will be used not only to display user transaction, but provide a means by which the user can add, update and delete transaction details. The application will also provide graphical representations of the user’s editable income and produce a dynamic representation of expenses.

2.2 Product Features

There will be six (6) features associated with the Transaction Tracker: managing transactions, monitor expenses, calculate cost, adding expenses, graphical representation of expenses and make stochastic decisions.

Managing Transactions– This feature allows the user to add, delete, update, cancel and view transactions. Adding a transaction entails inputting the transaction details, the date and submitting such information to the database to be stored. Deleting a transaction instructs the application to find the specified record in the database and remove it. Updating a record allows the user to change the title, cost or date of a transaction that already existed. Upon creating a transaction, the user has the option to cancel the transaction before the information is saved. Existing transactions appear on the screen once the Transaction Tracker is loaded allowing the user to view their transaction on the homepage of the navigation bar. The user will also be given the option to clear all transactions from the database in order to start anew each month/day/year.

Monitor Expenses – This feature is available on the second tab of the navigation bar. Here a user can set or update his/her targeted budget and generate a comparative analytic pie chart. The monitor expenses feature will give the user a visual representation of their budget and how close or far away they are to their target so that they can make decisions accordingly.

Calculate cost - This computation feature is available as an extended portion of the decision-making tab (third tab in the navigation bar). This is an average calculator provided within the application that will be used to compute the cost of items in order to make purchase decisions.

Make stochastic decisions - This feature gives the user random advice on item purchase. The opinion comes in the form of a rotating wheel in which the user will engage, the wheel will spin for a duration based on the touch intensity and the user will be given a random advice.

Adding Expenses - This feature will allow the user to add, delete, edit and view various expenses. When adding an expense, the user enters the title of the expense along with the amount spent. The application then sends the newly created record to the database and is stored there. This new expense is shown and sorted by date in the list of expenses. Editing an expense allows the user to change and update the title of expense as well as cost of expense. When creating an expense, the user is given an option to cancel the expense before it is saved. Deleting an expense is done by the user selecting a record to be deleted through the interface and this specific record is deleted along with its database.

Graphical Representation of Expenses - This feature allows for the creation of dynamic graphs. The generated graph shows all the expenses in a bar graph format. This bar graph is used to manage the various monthly expenses recorded within the app. This will allow the user to compare his expenses with his/her income graphically, and highlight whichever expense desired.

2.3 Special Widgets/Functionality

These are some of the specific widgets used in the creation of the Transaction Tracker.

1. **Recycler view** – This widget was used to show records from our database and thus place our transactions list. It was used instead of a traditional ListView because it was more customizable, optimized and could work with our large dataset.
2. **PiechartView** - This special widget was imported into our project from a GitHub resource. The main purpose of this widget was to display a dynamic and customizable view of the user's expenses when compared with their income.
3. **Fab** -The main purpose for using the Floating Action Button was to provide a unique method of launching certain activities and dialogs.
4. **Navigation Bar** - In order to navigate between the various Fragments of our application, this widget was used. It provided an easy method of access for all the various functionalities of the software.
5. **Wheel View** - This is a special widget that was imported to our resources. This allows to display a wheel that is clickable and spinnable.

2.4 User Classes and Characteristics

The Transaction Tracker is an application designed for adults seeking to monitor their transactions and do monthly budgeting.

Users will use the Transaction Tracker after making payments to document the amount spent in any instance and ultimately keep a record of their daily spending.

Users will use Transaction Tracker to analyze their monthly consumption value and comparative budgeted value.

Users will use the random decision making and additional calculator feature of the Transaction Tracker to help in their decision-making processes.

See Appendix A for a User Case Diagram.

2.5 Operating Environment

The Transaction Tracker can be loaded onto any android device running at minimum Android 4.0.3 (ice cream sandwich) operating system and at maximum Android 9.0 (pie) operating system. The Transaction Tracker will utilize an internal database for storing transaction details.

2.6 Design and Implementation Constraints

The following is an initial list of constraints that must be considered during the development and implementation process.

- Memory size and capacity of the handheld device will limit the amount of transaction records that can be stored at any one time.
- GUI interfaces must conform to XML format.
- Empty/void transactions cannot be added to the database.
- Information to be stored is limited to the parameters defined in the database.

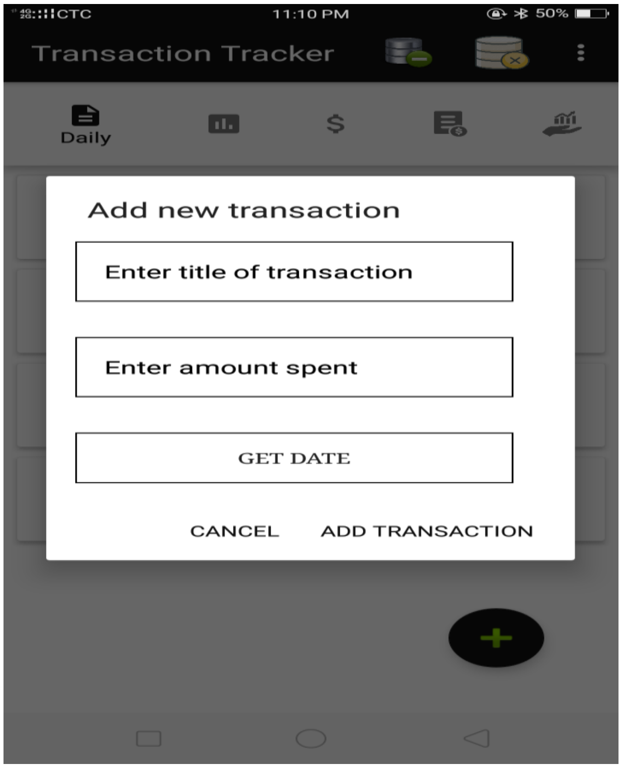
2.7 Assumptions and Dependencies

Effective operation of the Transaction Tracker will depend on the following pre-conditions being present at the time the application is implemented and initiates use.

The user’s device is compatible with versions between API version 15 and API version 28.
The user’s device has 2.6MB available space for the installation of the application.

3. System Features

The following are detailed descriptions of the four (4) features listed in Section 2.2. As the design component of development cycle progresses, additional details for these features may be added to this SRS.



3.1 Manage transactions

3.1.1 Add transaction

3.1.1.1 Description and Priority

High Priority – This will allow the user to add a transaction record to the database.

3.1.1.2 Stimulus/Response Sequences

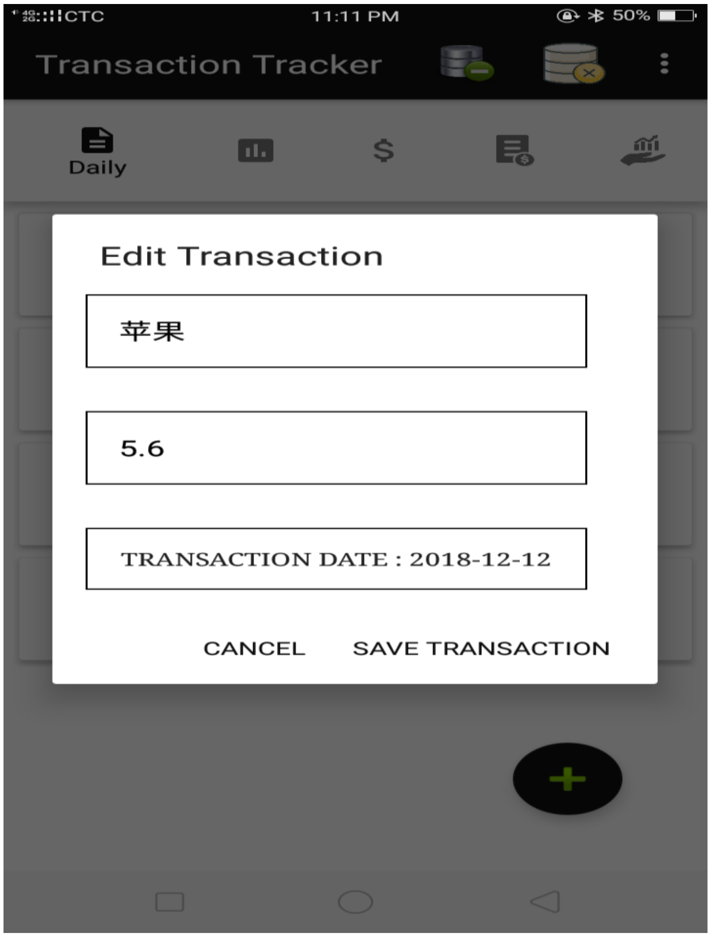
The user will select the add icon floating action tab which will open a dialog prompting the user to enter details of the transaction

3.1.1.3 Actor: User

3.1.1.4 Main Sequence

1. The user selects the add icon.
2. A dialog is fired prompting the user to enter the details of the transaction
3. User enters the name of expense and the expense amount
4. User selects the date once the date picker dialog is fired
5. User selects the add button
6. Transaction details are added to the database as a new record
7. The user is redirected to the home page and home page refreshes to include the newtransaction in the view.

3.1.2 Edit transaction



苹果

5.6

TRANSACTION DATE : 2018-12-12

3.1.2.1 Description and Priority

Medium Priority – This will allow the user to edit a transaction record in the database.

3.1.2.2 Stimulus/Response Sequences

The user will select the pencil icon which will open a dialog containing the previously saved details of the transaction where the user can then make changes to their desired information field.

3.1.2.3 Actor: User

3.1.2.4 Main Sequence

1. The user selects the pencil icon beside the transaction they want to edit.
2. A dialog is fired containing the previously saved details of the transaction in editable format.
3. User edits one or several details
4. User selects the save button
5. Transaction details are updated in the database
6. The user is redirected to the home page and the home page refreshes to include the updated information in the view.

3.1.3 Cancel transaction

3.1.3.1 Description and Priority

Medium Priority – This will allow the user to cancel the creation or updating of a transaction.

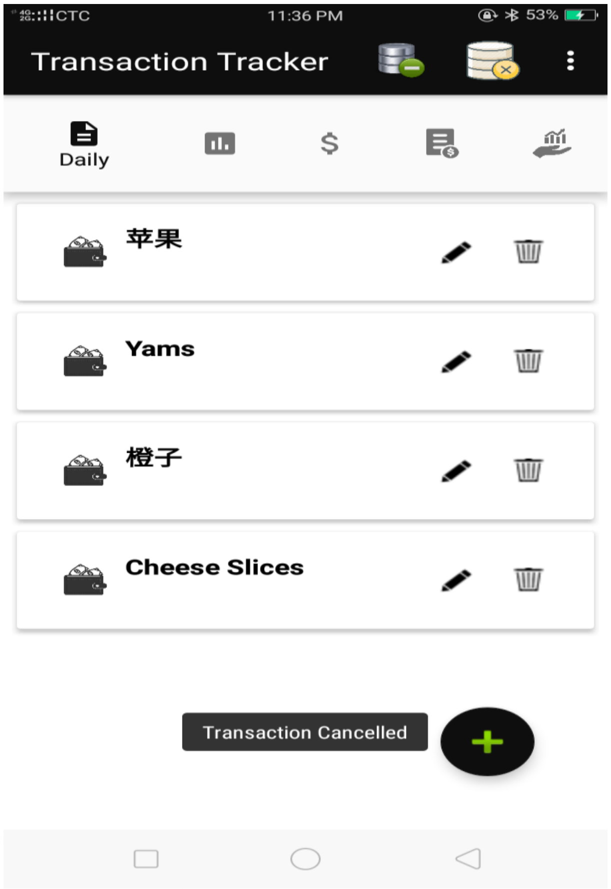
3.1.3.2 Stimulus/Response Sequences

The user will select the cancel button which will restore the system to its previous state.

3.1.3.3 Actor: User

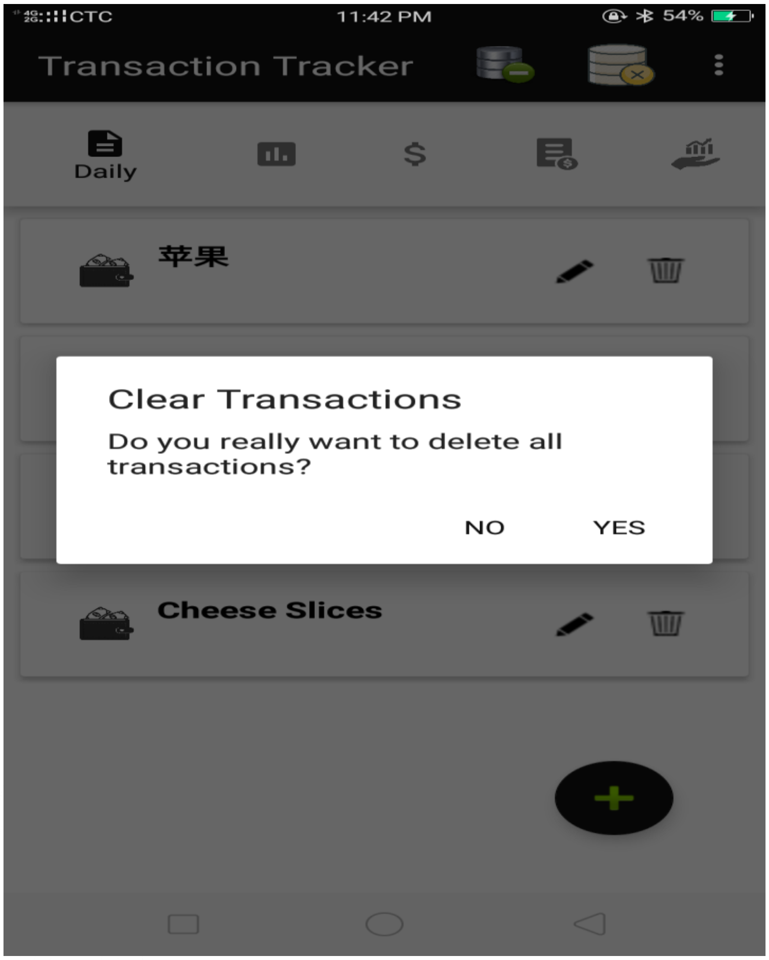
3.1.3.4 Main Sequence

1. The user selects the add or edit icon
2. A dialog is fired containing the previously saved details of the transaction in editable format or prompting the user to create a new record.
3. User selects the cancel button
4. A toast appears informing the user that the transaction has been cancelled
5. System is restore to its previous page



Transaction Cancelled

3.1.4 Delete transaction



Clear Transactions

Do you really want to delete all transactions?

3.1.4.1 Description and Priority

Medium Priority – This will allow the user to delete a transaction record from the database.

3.1.4.2 Stimulus/Response Sequences

The user will select the delete icon which will open an alert dialog requesting permission to delete the selected transaction record.

3.1.4.3 Actor: User

3.1.4.4 Main Sequence

1. The user selects the delete icon beside the transaction they want to be deleted.
2. An alert dialog is fired asking permission to delete the record from the datebase.
3. User selects the “Ok” button
4. The system searches the database table for the record matching the unique id of the transaction record selected and removes it.
5. The user is redirected to the home page and the home page refreshes loading information from the updated database records in the view.

3.1.4.5 Alternative Sequence

3. User selects “Cancel” button
4. System restores to its previous state

3.1.5 Clear database

3.1.5.1 Description and Priority

Medium Priority – This will allow the user to clear all transaction records from the database.

3.1.5.2 Stimulus/Response Sequences

The user will select the clear database button located on the menu tab which will open an alert dialog requesting permission to delete all transaction records.

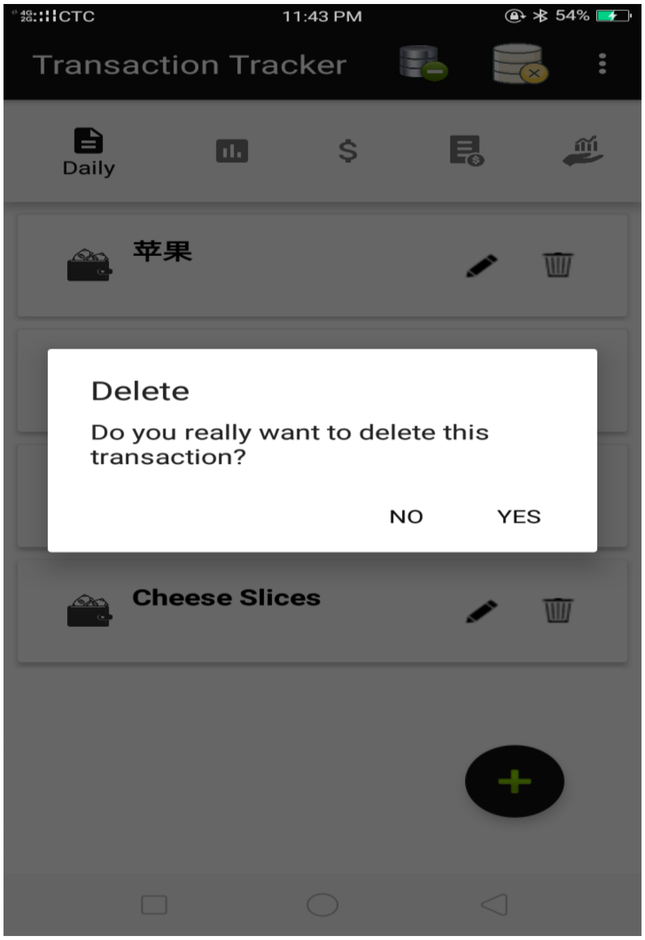
3.1.5.3 Actor: User

3.1.5.4 Main Sequence

1. The user selects the clear database button.
2. An alert dialog is fired asking permission to delete all records from the datebase.
3. User selects the “Yes” button
4. The system deletes all records from the database table.
5. The user is redirected to the home page and the home page refreshes.

3.1.5.5 Alternative Sequence

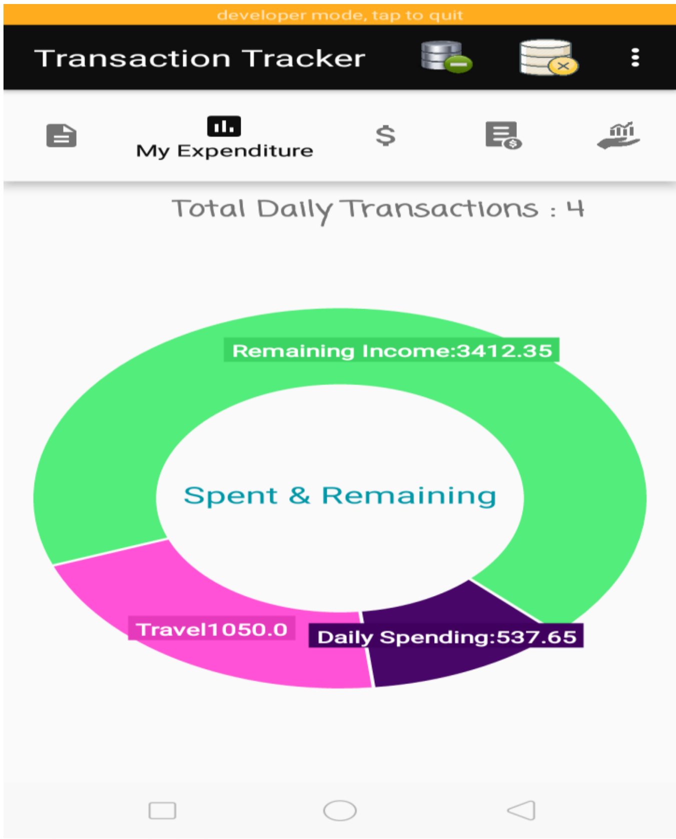
3. User selects “No” button
4. System restores to its previous state



Delete

Do you really want to delete this transaction?

3.2 Monitor expenses



3.2.1 Monitor Spending

3.2.2.1 Description and Priority

High Priority – This will allow the user to generate a pie chart visually representing the total transaction cost and the targeted budget.

3.2.2.2 Stimulus/Response Sequences

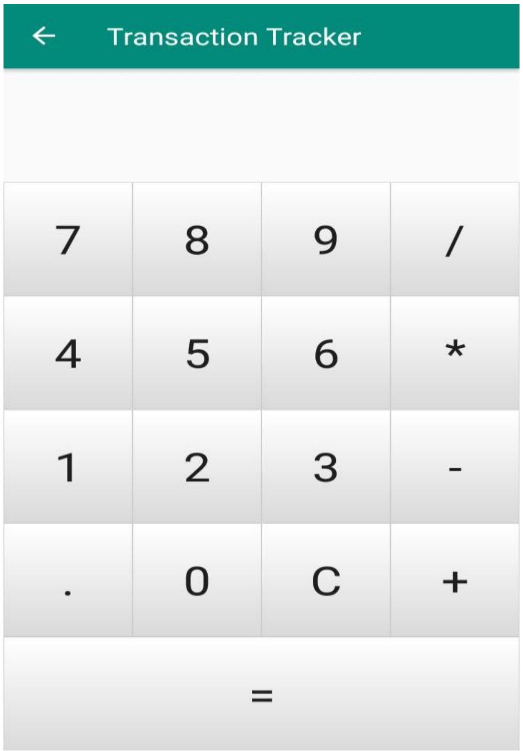
The user turns on the switch to generate a pie chart.

3.2.2.3 Actor: User

3.2.2.4 Main Sequence

1. The user enters their target budget
2. User turns on switch to generate a pie chart for statistical analysis.
3. System collects total transaction amount fron database.
4. System displays pie chart.

3.3 Calculate cost



3.3.1 Description and Priority

Low Priority – This will allow the user to calculate cost of an item without leaving the application.

3.3.2 Stimulus/Response Sequences

The user will enter the values to be calculated into the calculator feature.

3.3.3 Actor: User

3.3.4 Main Sequence

1. The user selects the calculator icon
2. An activity is fired containing all the functionalities of a basic calculator.
3. User computes the cost of item/items.
4. User makes decisions based on computations

3.4 Make stochastic decisions

3.4.1 Description and Priority

Low Priority – This will help the user to make on the forthwith decisions on purchase orders.

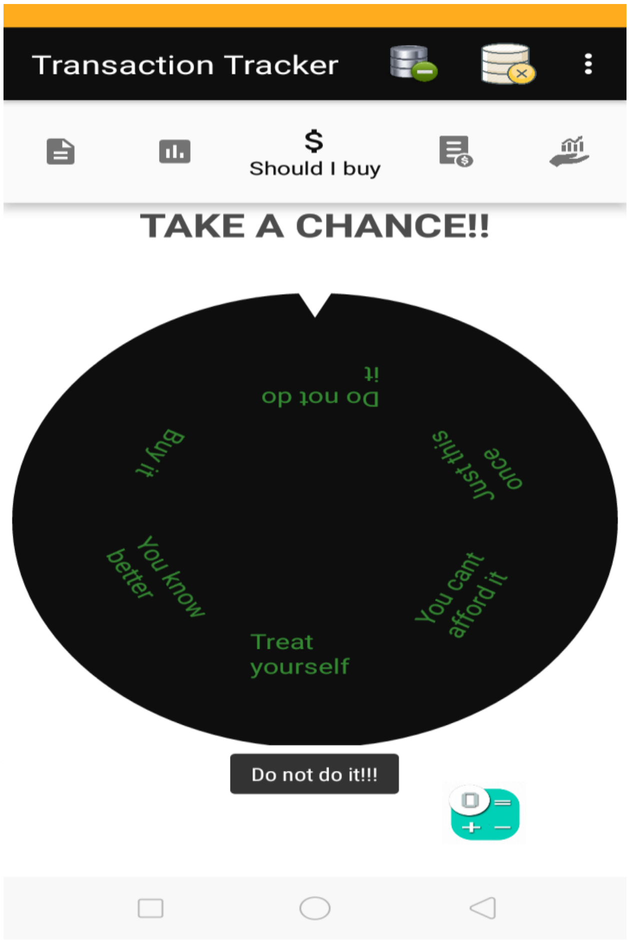
3.4.2 Stimulus/Response Sequences

The user will spin the wheel and make decisions based on the random response given.

3.4.3 Actor: User

3.4.4 Main Sequence

1. The user spins the wheel.
2. The wheel rotates based on the touch intensity of the user.
3. The user is presented with a random decision



3.5 Manage Expenses

3.5.1 Add monthly budget

3.5.1.1 Description and Priority

High Priority – This will allow the user to enter their target budget

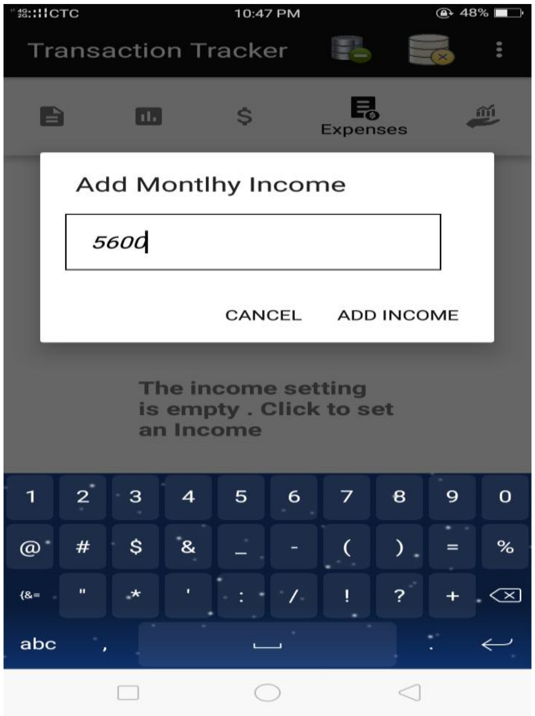
3.5.1.2 Stimulus/Response Sequences

The user will enter their target budget and system will store value in database

3.5.1.3 Actor: User

3.5.1.4 Main Sequence

1. The user enters their target budget.
2. System saves the value entered into the database.



Add Montlhy Income

5600

The income setting is empty . Click to set an Income

3.5.2 Add Expense

3.5.2.1 Description and Priority

High Priority – This will allow the user to add an expense record to the database.

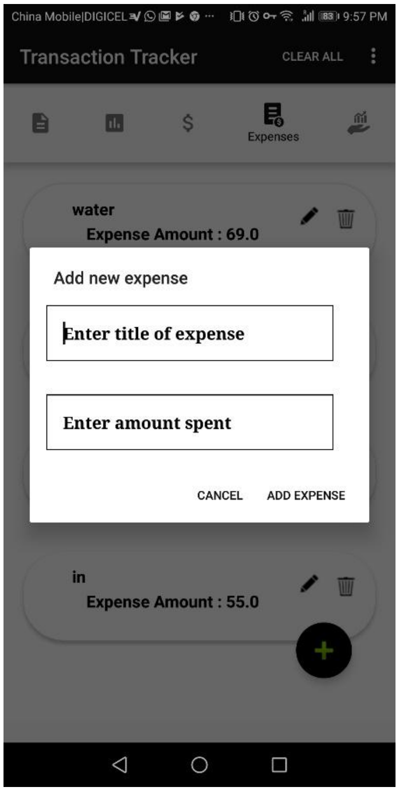
3.5.2.2 Stimulus/Response Sequences

The user will select the add icon floating action button which will open a dialog prompting the user to enter details of the expense.

3.5.2.3 Actor: User

3.5.2.4 Main Sequence

1. The user selects the add floating action button.
2. A dialog is fired prompting the user to enter the details of the expense
3. User enters the name of expense and the expense amount
4. User selects the add button
5. Expense details are added to the database as a new record
6. The user is sent back to the expenses page and the expenses page refreshes to include the new expense in the view.



water
Expense Amount : 69.0

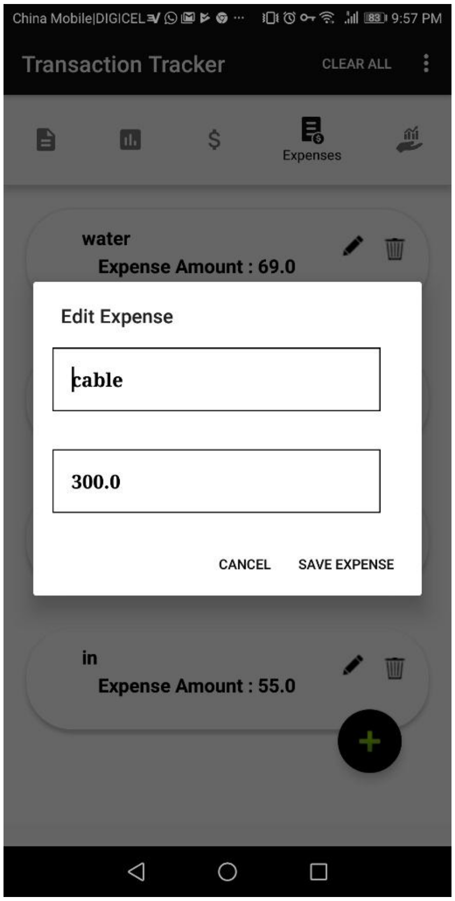
Add new expense

Enter title of expense

Enter amount spent

in
Expense Amount : 55.0

3.5.3 Edit expense



water
Expense Amount : 69.0

cable

300.0

in
Expense Amount : 55.0

3.5.3.1 Description and Priority

Medium Priority – This will allow the user to edit an expense record in the database.

3.5.3.2 Stimulus/Response Sequences

The user will select the edit icon which will open a dialog containing the previously saved details of the expense where the user can then make changes to their desired information field.

3.5.3.3 Actor: User

3.5.3.4 Main Sequence

1. The user selects the pencil icon beside the expense record they want to edit.
2. A dialog is fired containing the previously saved details of the expense in editable format.
3. The user edits at least one detail
4. User selects the save button
5. Expense details are updated in the database
6. The user is sent back to the expenses page and the expenses page refreshes to include the new expense in the view.

3.5.4 Cancel expense

3.5.4.1 Description and Priority

Medium Priority – This will allow the user to cancel the creation or updating of an expense.

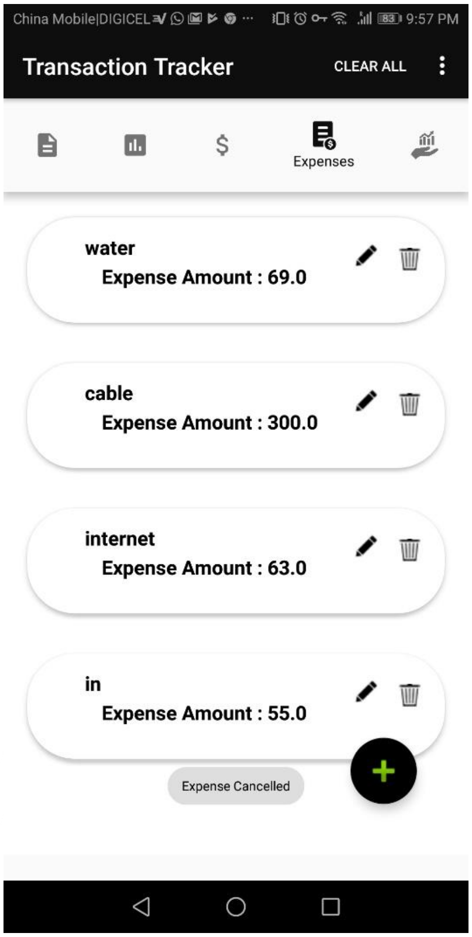
3.5.4.2 Stimulus/Response Sequences

The user will select the cancel button which will restore the system to its previous state.

3.5.4.3 Actor: User

3.5.4.4 Main Sequence

1. The user selects the add or edit icon
2. A dialog is fired containing the previously saved details of the expense in editable format or prompting the user to create a new record.
3. User selects the cancel button
4. A toast appears informing the user that the expense has been cancelled
5. System is restored to its previous page



water
Expense Amount : 69.0

cable
Expense Amount : 300.0

internet
Expense Amount : 63.0

in
Expense Amount : 55.0

Expense Cancelled

3.5.5 Delete Expense

3.5.5.1 Description and Priority

Medium Priority – This will allow the user to delete an expense record from the database.

3.5.5.2 Stimulus/Response Sequences

The user will select the delete icon which will open an alert dialog requesting permission to delete the selected expense record.

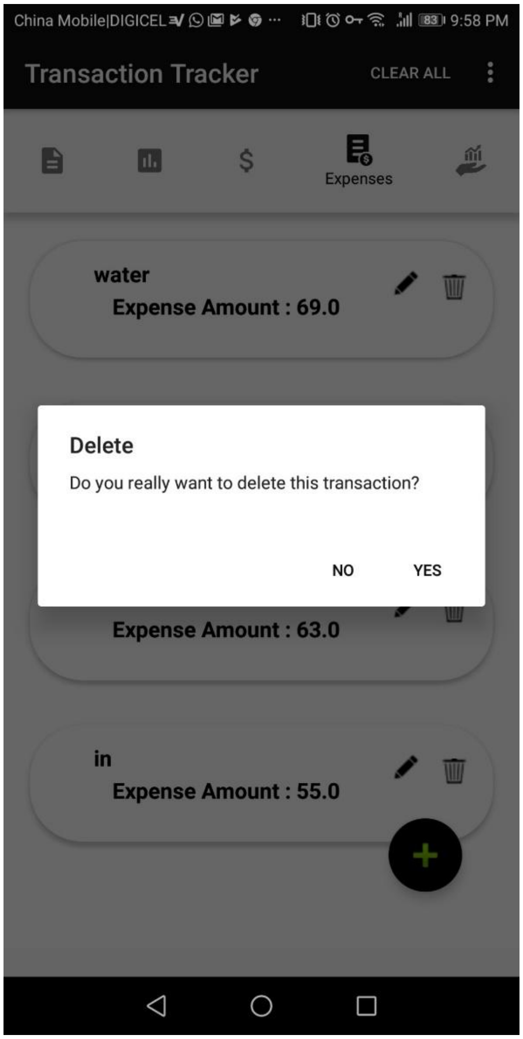
3.5.5.3 Actor: User

3.5.5.4 Main Sequence

1. The user selects the delete icon beside the expense they want to be deleted.
2. An alert dialog is fired asking permission to delete the record from the database.
3. User selects the "Ok" button
4. The system searches the database table for the record matching the unique id of the expense record selected and removes it.
5. The user is sent back to the expenses page and the expenses page refreshes to include the new expense in the view.

3.5.5.5 Alternative Sequence

3. User selects "Cancel" button
4. System restores to its previous state



water
Expense Amount : 69.0

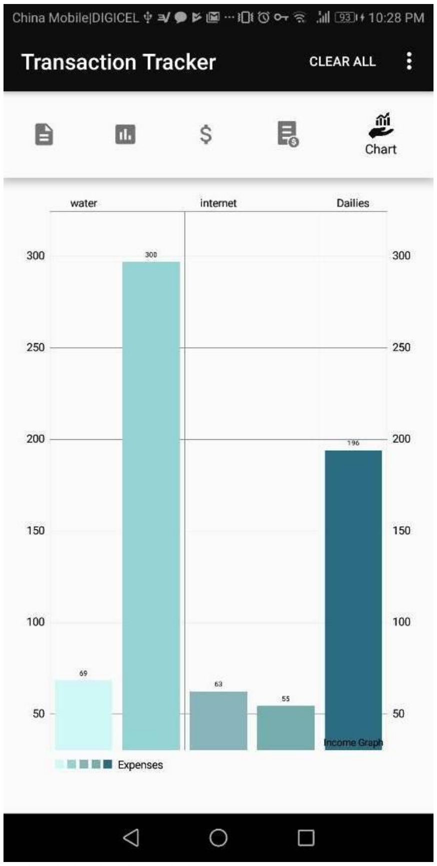
Delete

Do you really want to delete this transaction?

Expense Amount : 63.0

in
Expense Amount : 55.0

3.5.6 Generate Graph



3.5.6.1 Description and Priority

High Priority – This will allow a graph to be automatically automated.

3.5.6.2 Stimulus/Response Sequences

The user will select the graph tab which will atomically generate a graph after a few seconds.

3.5.6.3 Actor: User

3.5.6.4 Main Sequence

1. The user selects the chart option in the navigation bar.
2. The values for expenses are retrieved
3. The graphs are automatically generated from the data
4. The graph is displayed to the user

4. External Interface Requirements

4.1 User Interfaces

The User Interface will adhere to standard viewing protocols. In that, the look and feel will maintain high contrast and high visibility throughout the software. The background will be light in color with dark colored font to enhance the contrast and visibility. Icons will make use of bright colors and be of relation to the subject matter to which they navigate. The layout will be constrained to the size and scope of the display screen.

4.2 Hardware Interfaces

The Transaction Tracker will be able to be loaded onto any hand-held computing device running at minimum Android 4.0.3 (ice cream sandwich) operating system and at maximum Android 9.0 (pie) operating system.

5. Other Nonfunctional Requirements

5.1 Performance Requirements

The Transaction Tracker’s accuracy will only be constrained by the accurate entry of transaction information by the user.
The expense monitor is expected to generate with information from the database within ten (10) seconds after being prompted by the user.

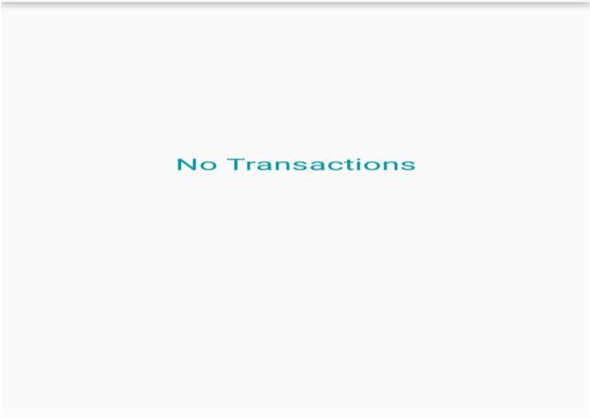
After commanding the Transaction Tracker to add a transaction the home page should dynamically be updated with the new record added to the database, as well as gives the user options of editing and deleting said transaction.

After commanding the Transaction Tracker to delete a transaction the home page should dynamically be updated removing said transaction.

6. Optional Requirements

An updated version of the Transaction Tracker may be developed to enable picture input in a future release. This will allow users to save images of their transactions into the database for easier input and usability. Picture input will allow for rapid recollection of purchases made by the user.

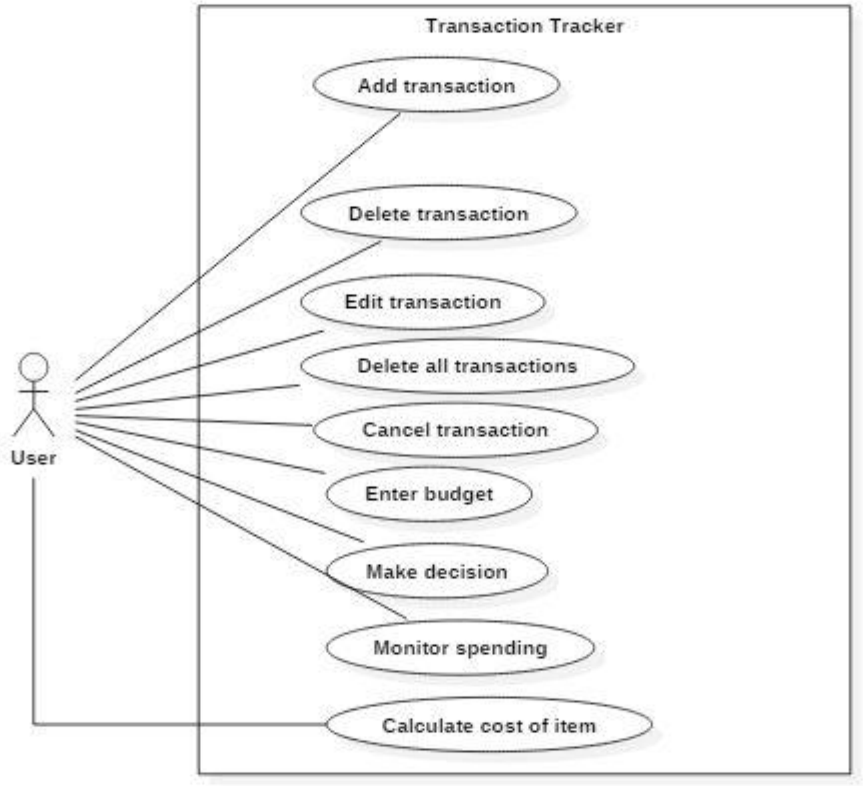
Appendix A: Additional User Interface Sample Images and Use Case


No Transactions

Welcome Page

Empty Transaction List

User Interaction Diagram


Transaction Tracker
Add transaction
Delete transaction
Edit transaction
Delete all transactions
Cancel transaction
Enter budget
Make decision
Monitor spending
Calculate cost of item
User