

GitHub Username: jrc294

Mileage Master & Expense Star

Description

Problem:

The user is a professional driver, such as a taxi or delivery driver, who needs to record distances travelled for one or more clients, along with any additional expenses incurred. The user might use the data recorded to create expense claims or client invoices based on their daily travels, and maintain a historical record for audit purposes. The app will keep a record for them to do this.

Proposed Solution:

1. Design an app that allows a user to enter daily travel trips, which includes an activity so the user can enter...
 - Date & Time of the trip
 - The client name, selected from a spinner of user defined client names. (The app will have a separate activity for client maintenance, that will record the client name, cost per mile to be charged for the client, any additional tax percentage to be tacked on, and any additional pre-defined expenses that should always be charged).
 - A start and end address for the trip.
 - Distance travelled in miles / km. (The app will be configured to use either miles or km in preferences).
 - The cost of the trip calculated by multiplying the distance travelled by the cost per mile / km attributed to the client
 - A check box to indicate if the trip has been completed
 - A free text notes field
 - Additional expenses, either pre-defined and selected from spinners, or one offs that can be added to the cost of each trip. (The app will have a separate activity for pre-defined expense maintenance, which will record the description and cost of the expense).
 - The cost of any tax, using the pre-defined rate that is assigned to the customer.
2. The user will be able to open a saved individual trip for editing or deletion.
3. The app will include a home screen with a bar chart showing miles travelled on a monthly per customer basis.
4. The app will have a search facility to allow the user to search for trips based on the following...
 - Client
 - Date range

- Trip Completed
... and will report on the total distance travelled and expenses incurred depending on the filter options selected.

Intended User

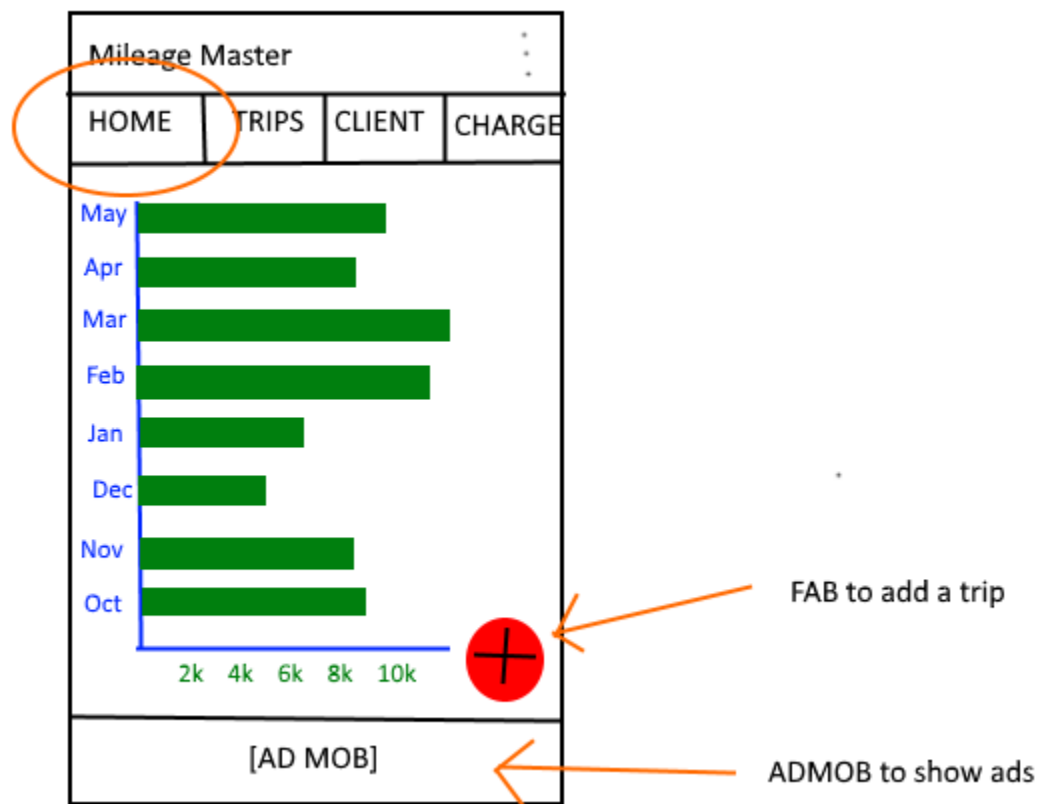
The app is intended for use by professional drivers, such as taxi drivers, delivery drivers, home care professionals, etc.

Features

- Saves information to a database with a content provider
- Retrieves historic data.

User Interface Mocks

Home Screen



The app will have four tabs consisting of HOME, TRIPS, CLIENT and CHARGE. The home screen will show a bar chart showing the distances travelled over the previous few months. A

FAB will start the activity to add a new trip. The bottom of the screen will show an ad from ad mob.

Trip Activity

Trip details top half of scroll view

Trip	
05-May-2016 ▲	14:22 ▲
ACME ▲	
From	
To	
Distance	\$0.00
<input type="checkbox"/> Trip Complete	
Standard Fee 1 ▲	\$0.00
Standard Fee 2 ▲	\$0.00
Standard Fee 3 ▲	\$0.00

Enter date and time, client, start point, end point, distance travelled, charge for mileage (auto calculated) has the trip been completed, and up to three standard charges

Trip Activity bottom half of scroll view

Trip	
Other Charge	\$0.00
Other Charge	\$0.00
Other Charge	\$0.00
Add the Tax	\$0.00
Total Cost of Trip	\$0.00
Notes	

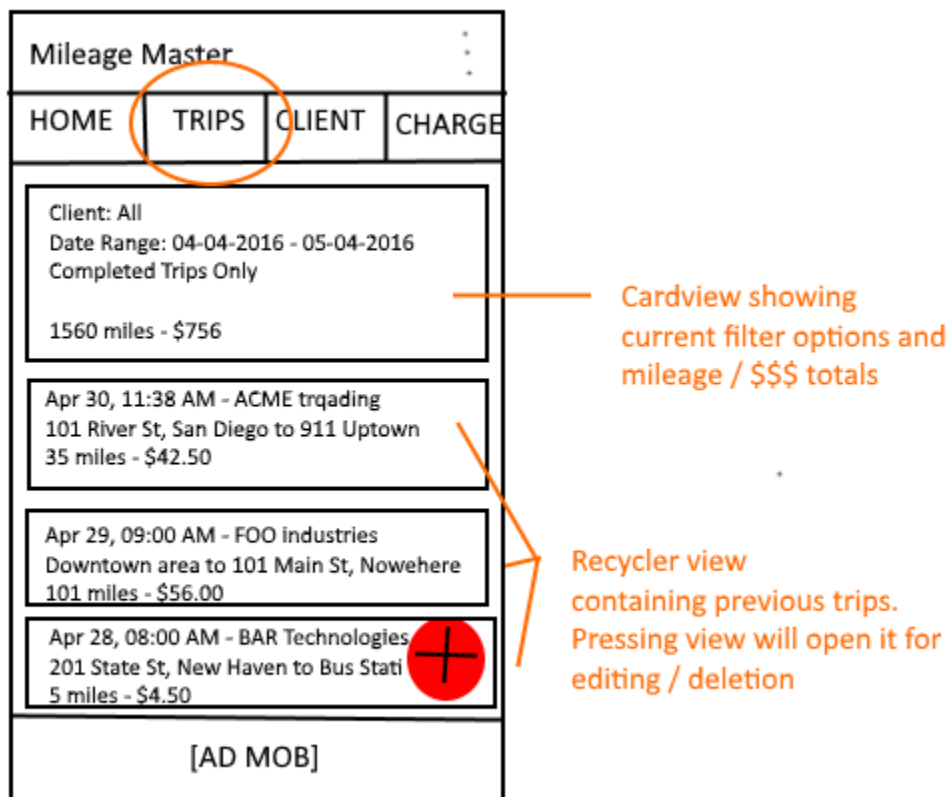
User can add up to three ad hoc charges, and pressing the 'Add the Tax' button will calculate tax based on the tax percentage for the client.

Also there is a free form text field for notes.

A scroll view will be used with a number of widgets and text fields that allow the user to add the details of their trip. They will include the...

- Date / Time from date and time pickers.
- A spinner to select the client
- Edit text fields to add the start and end places
- An edit text field to add the distance travelled
- A \$\$\$ amount that will be auto calculated by multiplying the distance travelled by the cost per mile stored against the client
- A check box to say if the trip has been completed
- Spinners to add up to three pre-defined standard charges.
- Edit text fields to add up to three free form, ad hoc charges.
- An add the tax button, which will multiply the charges total by the tax rate stored against the client
- A field that calculates all the charges for the trip.
- An edit text field for notes.

Trips List View

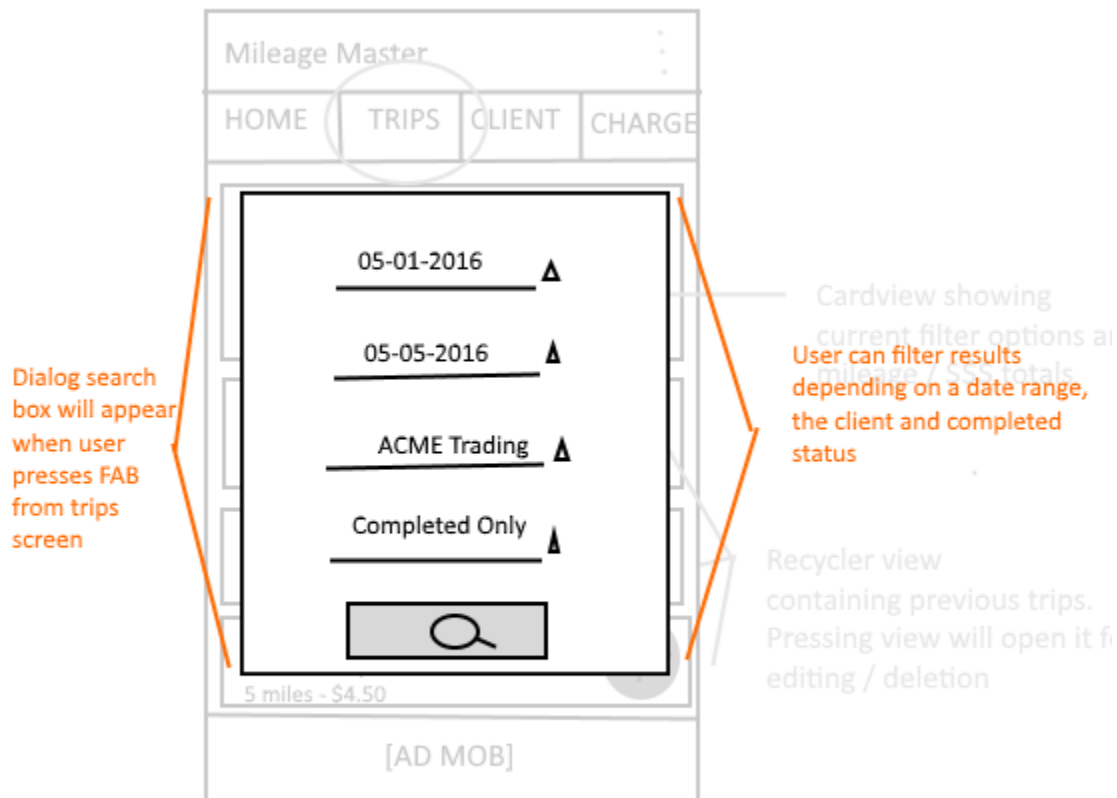


The trips list screen will load a list of recent trips in descending order. Pressing an individual trip will open the trip for editing or deletion.

Pressing the FAB will open a filter dialog box that will allow the user to restrict the trips to those limited by the filter options.

The first card view on the screen will show the current filter options and the mileage / money totals.

Trip List Filter Dialog Box



When the user presses the FAB from the trips list screen, a filter screen pop up window will appear allowing the user to filter the results based on the selection criteria. The date range fields will have date pickers, and the client and completed status widgets will use spinners.

Trip List and Detail Tablet View

Mileage Master			
Home	Trips	Clients	Charge
<div>Client: All Date Range: 04-04-2016 - 05-04-2016 Completed Trips Only 1560 miles - \$756</div> <div>Apr 30, 11:38 AM - ACME trading 101 River St, San Diego to 911 Uptown 35 miles - \$42.50</div> <div>Apr 29, 09:00 AM - FOO industries Downtown area to 101 Main St, Nowhere 101 miles - \$56.00</div>		<div>05-May-2016 ▲ 14:22 ▲</div> <div>ACME ▲</div> <div>From</div> <div>To</div> <div>Distance \$0.00</div> <div><input type="checkbox"/> Trip Complete</div>	

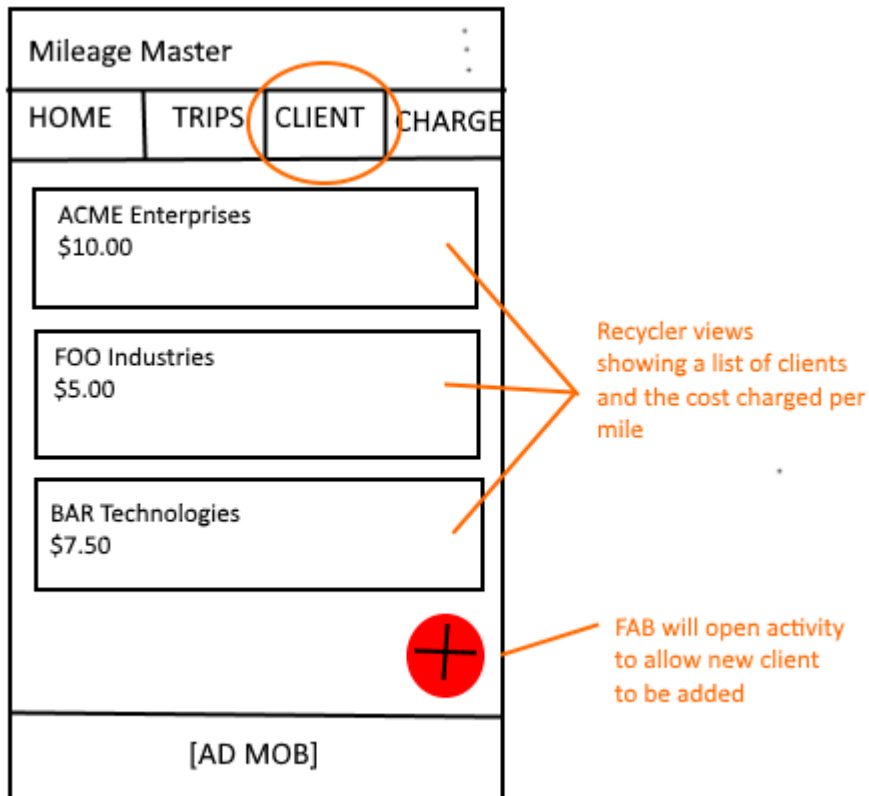
For screens beyond a certain width, combine the trips master detail view into a single item.

Settings Screen

<- Settings
Distance Units Miles

Settings screen is accessible from the main activity action bar, and will allow the user to set the distance units.

Clients List View



The clients List screen shows a list of clients that are currently defined in the application. Pressing one of the clients will open it in another activity for editing or deletion.

Pressing the FAB will allow the user to enter an additional client

Client Details View

Client Details

Name

Cost

Tax %

Enter any default charges here

Standard Charge 1 ▾

Standard Charge 2 ▾

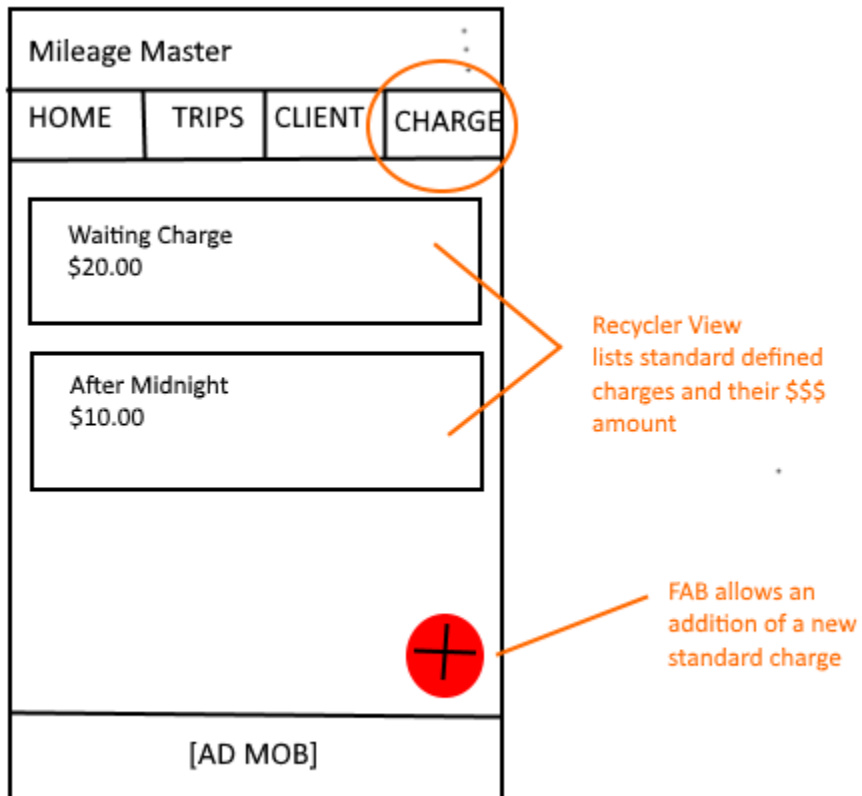
Standard Charge 3 ▾

Menu will have a delete option

User can add a new client with a name and a cost, a tax % and up to three standard charges

This activity will be displayed when the user presses the FAB button from the client list activity. It allows a client to be created, edited and deleted. The client name, mileage cost and tax %, along with up to three standard charges that will be auto selected for a trip for that client, can be stored.

Standard Charge List View



The standard charge list screen shows a list of charges that are currently defined in the application. Pressing one of the clients will open it in another activity for editing or deletion.

Pressing the FAB will allow the user to enter an additional charge

Standard Charge View

The image shows a wireframe of a 'Standard Charge' form. The form has a title bar at the top labeled 'Standard Charge' with three dots on the right. Below the title bar, there are two input fields: 'Name' and 'Cost', each with a horizontal line underneath. To the right of the form, there are two orange annotations with lines pointing to the form. The first annotation, 'Menu will have a delete option', points to the three dots in the title bar. The second annotation, 'User can store a charge and a cost', points to the 'Name' and 'Cost' input fields.

This activity will be displayed when the user presses the FAB button from the charge list activity. It allows a charge to be created, edited and deleted. The charge name and cost will be stored. The charge cannot be deleted if it is referenced by a client or trip.

Key Considerations

How will your app handle data persistence?

Data persistence will be handled via a content provider which will store and retrieve the data in four tables in the database.

1. CLIENT
2. STANDARD_CHARGE
3. TRIP_CHARGE
4. TRIP

Describe any corner cases in the UX.

None

Describe any libraries you'll be using and share your reasoning for including them.

For example, Picasso or Glide to handle the loading and caching of images.

I plan on using MPAndroidChart to present a bar chart on the Home Screen of the app which will show a graphical representation of miles travelled over the previous months.

Next Steps: Required Tasks

This is the section where you can take the main features of your app (declared above) and decompose them into tangible technical tasks that you can complete incrementally until you have a finished app.

Task 1: Create new project in Android Studio

Set up new Android project in Android Studio.

Task 2: Create content provider and database

1. Define contract for data updates and retrieval
 - a. Define database tables and columns
 - b. Define URI methods
2. Create class to extend SQLiteOpenHelper to create database
3. Create content provider for querying and updating database
4. Run unit tests to prove data can be added and retrieved from the database

Task 3: Implement UI for Main Activity

1. Configure MainActivity to include four tabs for HOME, TRIPS, CLIENTS and CHARGES
2. Configure FAB to start a TripActivity when coded.
3. Configure settings menu to open settings and show settings screen and store in shared preferences.

Task 4: Implement UI for Charges List Activity and Charge Activity

1. Configure ChargeActivityList and fragment to retrieve a list of charges from the database. This will be visible from the charges tab.
2. Configure ChargeActivity and fragment to allow charges to be saved and edited from the database. This activity will be started when either pressing the fab from the ChargeActivityList activity, or the charge item already existing in the list.

Task 5: Implement UI for Clients List Activity & Client Activity

1. Configure ClientActivityList and fragment to retrieve a list of clients from the database. This will be visible from the client tab.
2. Configure ClientActivity and fragment to allow charges to be saved and edited from the database. This activity will be started when either pressing the fab from the ClientActivityList activity, or the client item already existing in the list.

Task 6: Implement UI for Trips List Activity and Filter Pop-up Activity

1. Configure TripActivityList and fragment to retrieve a list of trips from the database. This will be visible from the trips tab.
2. Make sure that the first card view in the tab displays the filter criteria along with mileage and money totals.
3. Configure FAB to open a pop up filter screen, to limit the results retrieved by the activity.
 - a. The app will use an AsyncTask to retrieve the required data from the database.

Task 7: Implement UI for Trip Activity

1. Configure Trip Activity and fragment to be able to add, edit and delete an individual trip entry.
 - a. Use date time picker for date
 - b. Use time picker for time
 - c. Use spinners for client and standard charges.
 - d. Pre-populate standard charges depending on client
 - e. Auto calculate the cost of the mileage by multiplying the miles travelled by the cost per mile defined for the client.
 - f. Calculate a grand total for the trip by adding the mileage cost and additional charges and any tax.

Task 8: Implement MPAndroidChart in the home screen

1. Add the MPAndroidChart to the home screen to give an over view of the distances travelled in previous months.

Task 9: Create user widget

1. Create a user widget that shows a list of recent and upcoming trips. Pressing a list item will start the app trip activity.

Task 10: Add the admob advert to the bottom of the mainActivity.

1. Make sure that AdMob is included at the bottom of the main activity.

Task 11: Add Google analytics

1. Add Google Analytics
-