SRH Hochschule Heidelberg

Monthly Expense Tracker

Mobile Application Development
Prof. Christoph Hahn

Submitted by:
Nagashree Tadapatri
11007940

Date: 20/3/2016

CONTENTS

INTRODUCTION

The objective of the mobile application Monthly Expense Tracker is to keep a tab on one's personal budget on monthly basis. The app allows organization of expenses into various categories like grocery, fuel, entertainment, parking, fixed monthly expenses and more. The user can set his monthly budget, and he will be notified whether or not he is operating within budget. The app also integrates to Google APIs to fetch user's current location to offer services like reminder on grocery list items.

The app has a simple user interface with precise information, mostly easy to comprehend icons. All views are maintained on the same level, so as to enable quick navigation from any view to another.

This mobile app is built using Ionic framework and AngularJS. The data stores employed for this app are Firebase and WebSQL. WebSQL stores the user's monthly expense data. Firebase stores user's favorite places data. The mobile app is tested OK on Android platform using Ionic lab, Genymotion emulator and an android device.

FEATURES

Following are the features of this mobile app.

❏ Logs user's expenses in different categories; eight categories.

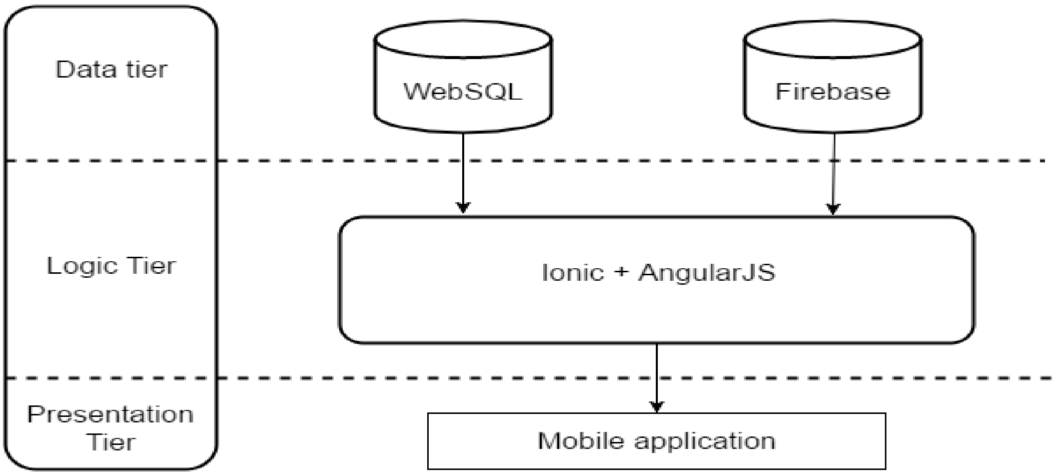
❏ Allows the user to set a target budget for a month.

❏ Keeps a tab on expenditure and notifies when the budget gets exceeded.

❏ Allows the user to save his favourite place for grocery shopping. Fires a notification when the user is within 500 meters radius from the target location, if he has items in his grocery list.

❏ All the expenses in the 'Fixed Expenses' category are replicated on the first day of the new month.

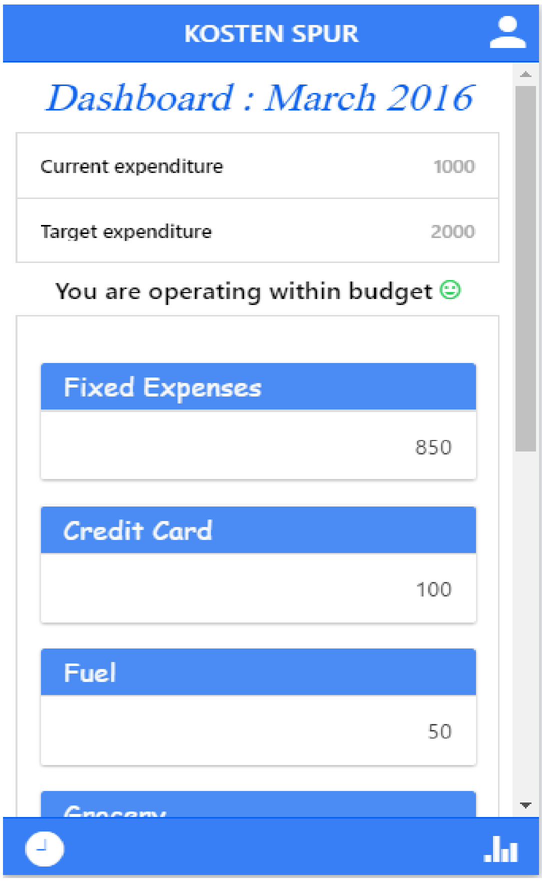
SYSTEM ARCHITECTURE



MONTHLY EXPENSE TRACKER

USER INTERFACE

User Interface screen captures



KOSTEN SPUR

Dashboard : March 2016

Current expenditure	1000
Target expenditure	2000

You are operating within budget ☺

Fixed Expenses

850

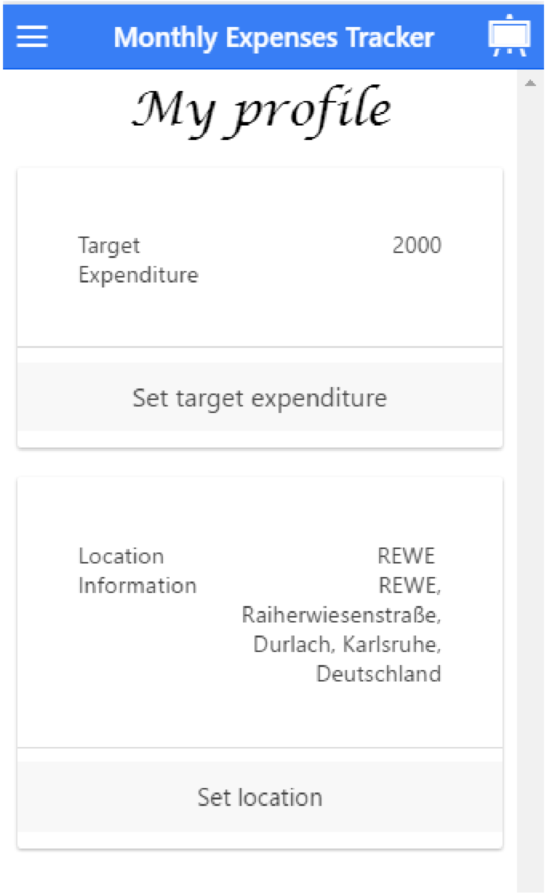
Credit Card

100

Fuel

50

Grocery



Monthly Expenses Tracker

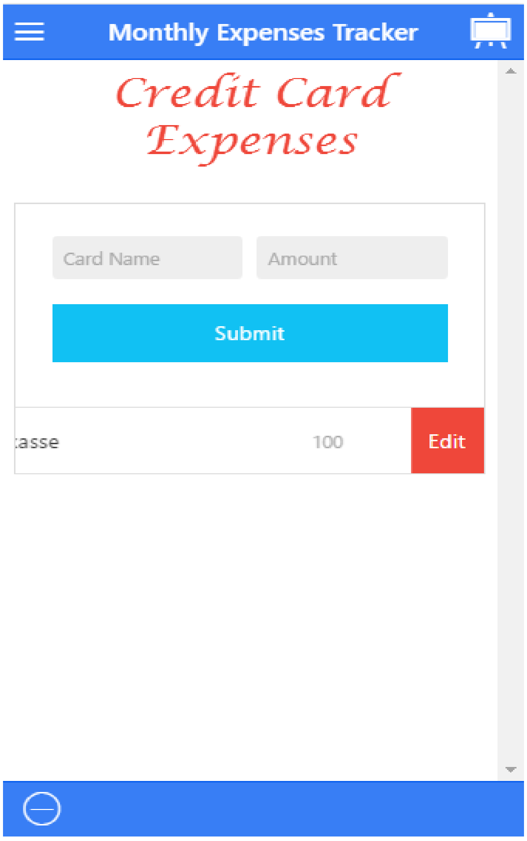
My profile

Target Expenditure	2000

Set target expenditure

Location Information	REWE REWE, Raiherwiesenstraße, Durlach, Karlsruhe, Deutschland

Set location



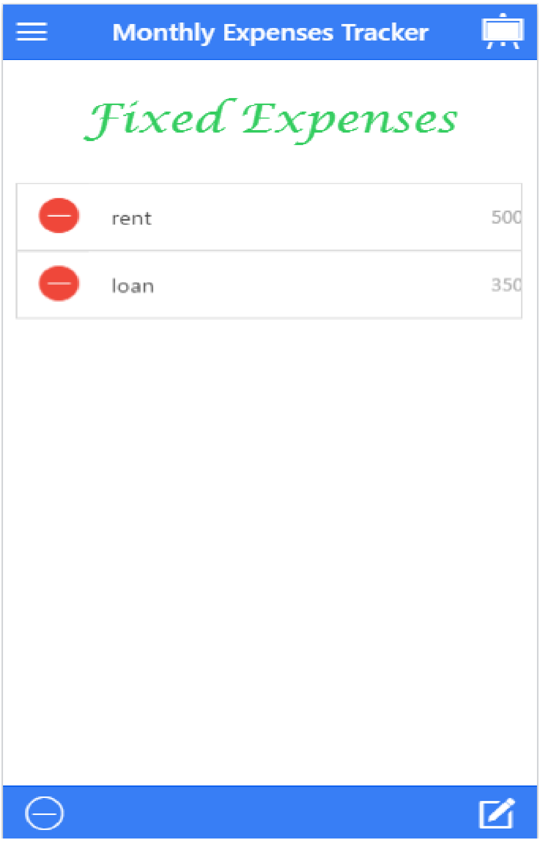
Monthly Expenses Tracker

Credit Card Expenses

Card Name | Amount

Submit

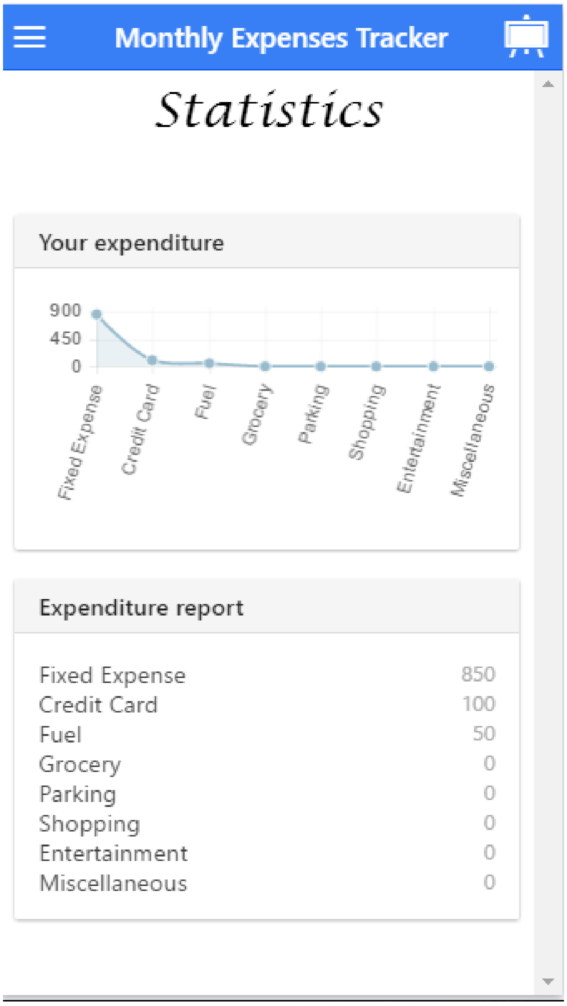
:asse	100	Edit



Monthly Expenses Tracker

Fixed Expenses

rent	500
loan	350



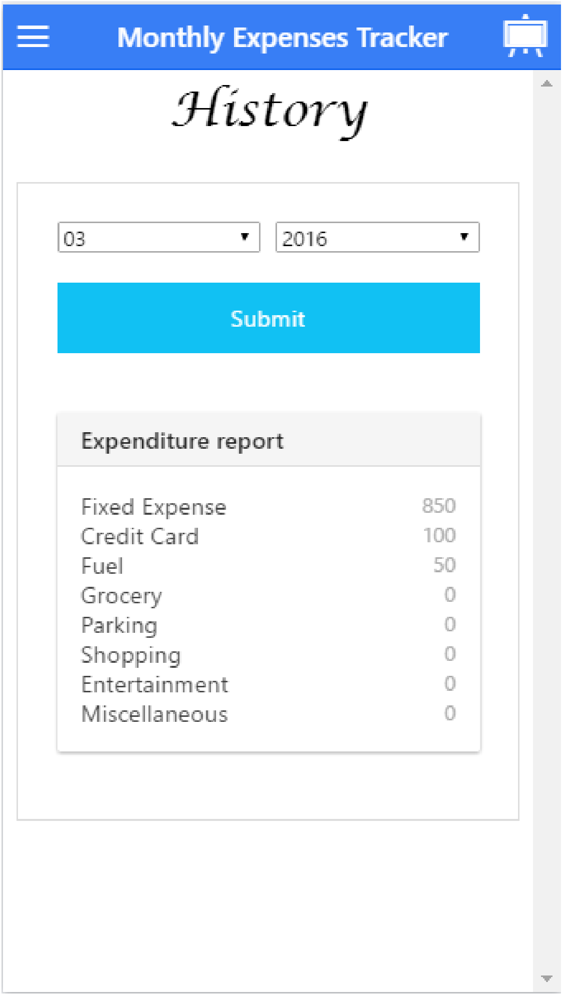
Monthly Expenses Tracker

Statistics

Your expenditure

Expenditure report

Fixed Expense	850
Credit Card	100
Fuel	50
Grocery	0
Parking	0
Shopping	0
Entertainment	0
Miscellaneous	0



Monthly Expenses Tracker

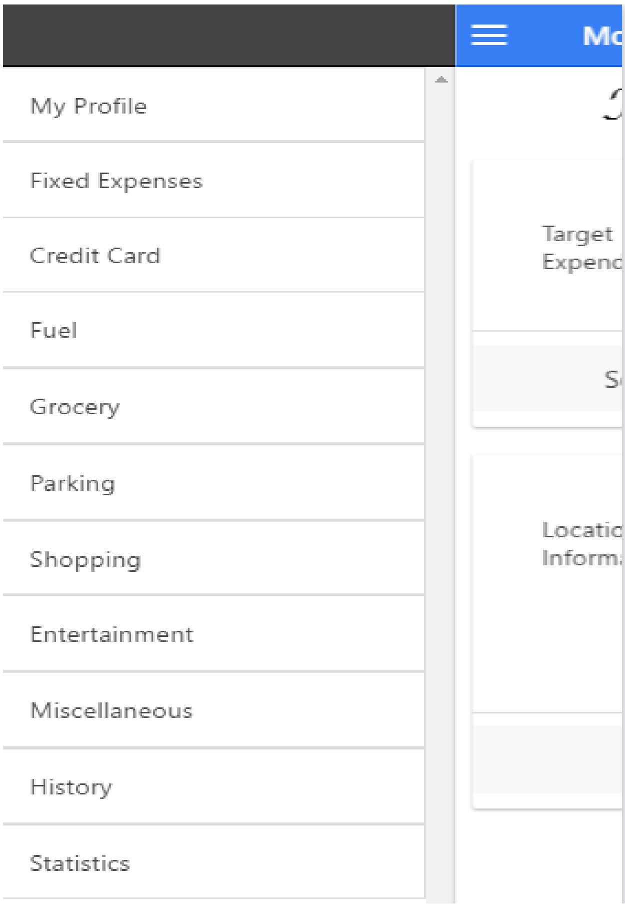
History

03 ▾ 2016 ▾

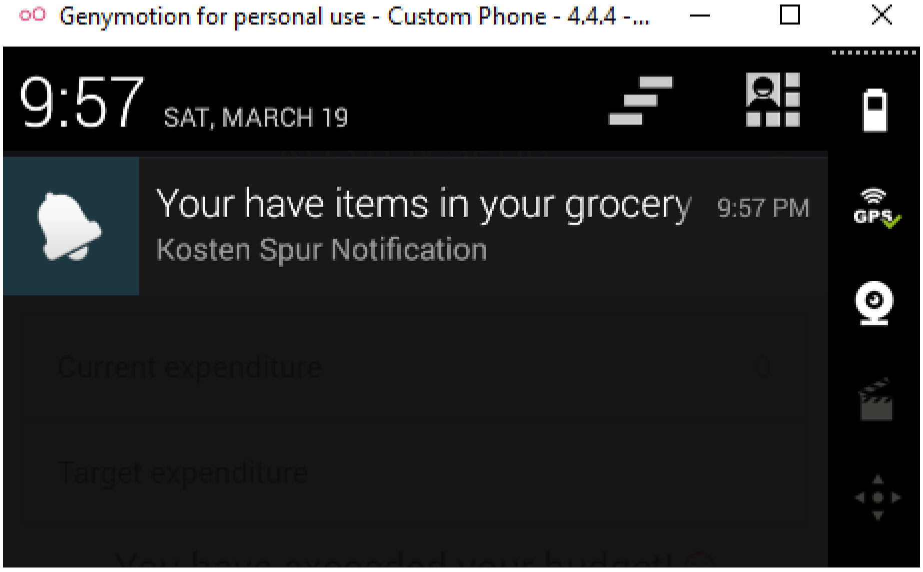
Submit

Expenditure report

Fixed Expense	850
Credit Card	100
Fuel	50
Grocery	0
Parking	0
Shopping	0
Entertainment	0
Miscellaneous	0

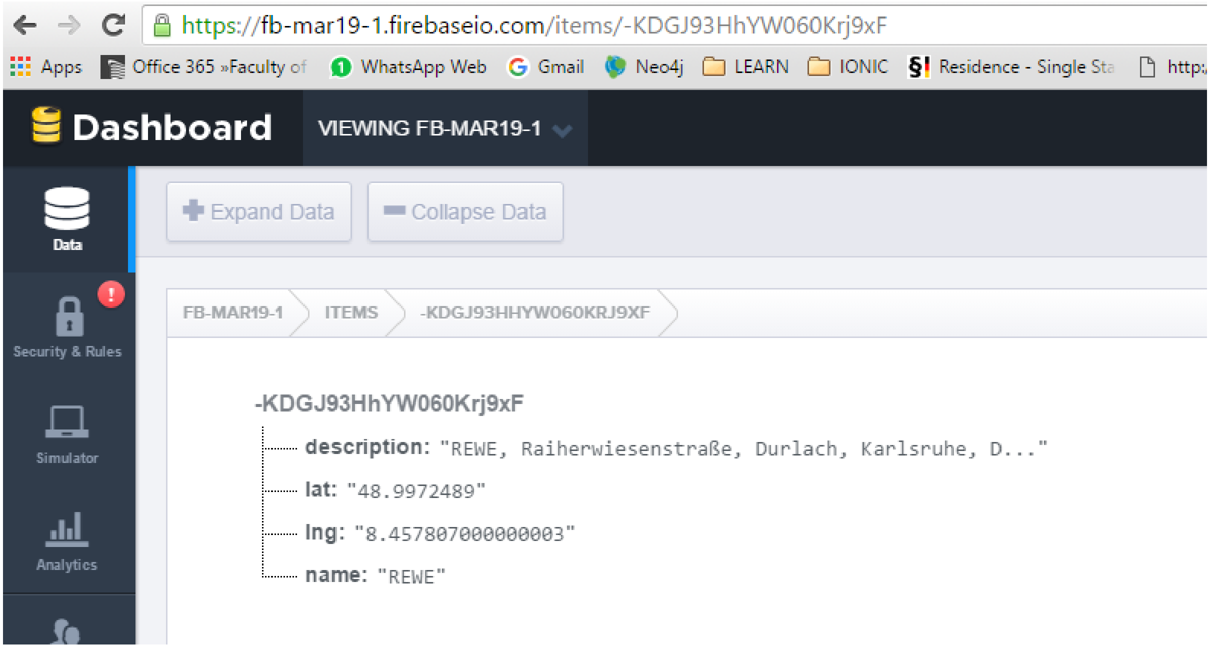


Notification when there are pending items in grocery list and the user is near target location



Your have items in your grocery 9:57 PM
Kosten Spur Notification

User’s favourite location stored in Firebase



```
-KDGJ93HhYW060Krj9xF
  description: "REWE, Raiherwiesenstraße, Durlach, Karlsruhe, D..."
  lat: "48.9972489"
  lng: "8.457807000000003"
  name: "REWE"
```

PERFORMANCE TUNING

The performance tuning measures used in the app are listed below.

- ➔ Usage of Angular template cache to ensure all views are already cached at app start-up. gulp-angular-templatecache plugin is used for this purpose.
- ➔ Enabling Ionic’s Native Scrolling instead of using the default Javascript scrolling. It can be enabled per view by placing attribute overflow-scroll=”true” on an ion-content tag. Or we can enable it for all views with $ionicConfigProvider.scrolling.jsScrolling(false);
- ➔ Adding Crosswalk plugin to obtain improvement on CSS animations. Crosswalk provides a bundled Chrome Webview with the app, resulting in better performance and predictability.
- ➔ Usage of “collection-repeat” instead of “ng-repeat” to iterate over items stored in arrays or data store.
- ➔ In case of animation usage, explicitly enabling animation serves as a performance improvement. As another option, Native transitions plugin was used which did not result in successful optimization of performance.
- ➔ Usage of Infinite-scroll and Pull-to-refresh options.

RESOURCES

List of Plug-ins, external JavaScript functions, CSS and external AngularJS Directives used

Plugins:
Cordova-plugin-geolocation
Com.telerik.plugins.nativepagetransitions
Cordova.plugin.local-notification

JavaScript functions:
Locator-tpl.js
Angular-chart.min.js
Angularfire.min.js
Chart.min.js
Firebase.js
Ng-cordova.min.js

CSS:
angular-chart

AngularJS Directives:
Locator.min.js

FUTURE SCOPE

1. Usage of Gulp Tasks and Cordova Hooks to enhance performance.
2. Allowing the user to save multiple favourite location for shopping.
3. Enabling the option of target budget for each expense category.
4. Calculation of weekly statistics of expenditure.
5. Usage of Calendar templates for better user interface.