MONEY MAP

STUDENTS

ROLL NO:-31247 **NAME**:-SAHIL S. PATIL
ROLL NO:-31251 **NAME**:-PRALHAD A. KULKARNI
ROLL NO:-31254 **NAME**:-SRUSHTI S. RAYBHOGE

MOTIVATION/NEED

- Whenever a student moves away from his family to stay in a hostel he/she always faces some problems.
- One of the most frequently occurring problem is to manage his/her money.
- It is frequently heard that the student is uncertain where he/she has exactly spent his/her money after a certain period of time.
- Our main motive is to make the student realize where and how much he is spending.
- This also makes the student efficient in managing his/her money.
- It may also happen that the student has given some amount to someone or has taken an amount from someone and due to his/her busy schedule he/she might forget to pay or receive the amount.
- Thus a log about the peer transactions that took place must be maintained and the user may delete it when the transaction pays off.

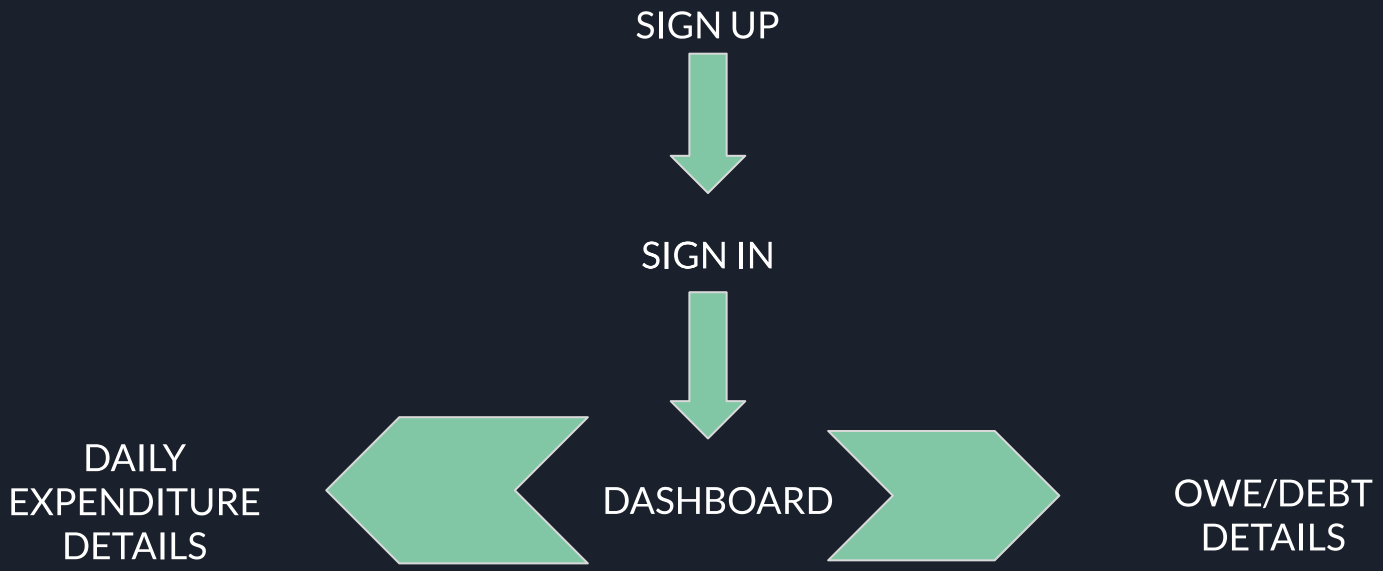
SURVEY

- A survey on students was done just to check whether they knew when and for what reason had they spent a large amount of money.
- Most of the student were able to answer this question with ease.
- But when asked to calculate last week's total expenditure approximately most of them failed to recollect all the transactions.
- This shows that a big transaction is always remembered but small transactions which may also amount more than the big transactions are often neglected by the students.
- To avoid this a regular log should be maintained which will include both the big as well as the small transactions.
- Our app provides the same.

PROBLEM STATEMENT (SCOPE)

To create a app which makes the life of a student easier to maintain a log of his/her expenditure including the amount he/she owes or is in debt , thus making it easier for student to analyze the transactions and to avoid any unnecessary transactions in the near future.

ARCHITECTURE DESIGN



SIGN UP

SIGN IN

DASHBOARD

DAILY EXPENDITURE DETAILS

OWE/DEBT DETAILS

DASHBOARD DISPLAYS DAILY TOTAL EXPENDITURE AND MONTHLY GRAPH OF EXPENDITURE

HARDWARE / SOFTWARE REQUIREMENTS

- Internet Connection
- Recycler view
- MPgraph
- Firebase Authentication connection
- Firebase Database connection
- Android Version greater than Marshmellow

ALGORITHM

- If expenditure is added it reflects on todays list in expenditure manager activity and also the graph in the dashboard
- If owe is added it reflects on the owe/debt manager activity and also the graph by decreasing the amount in today's expenditure.
- If debt is added it reflects on the owe/debt manager activity and also the graph by increasing the amount in today's expenditure.
- If the owe is removed from the owe debt activity the amount is subtracted to the graph of daily expenses.
- If the debt is removed from the owe debt activity the amount is added to the graph of daily expenses.

HIGH-LEVEL DESIGN

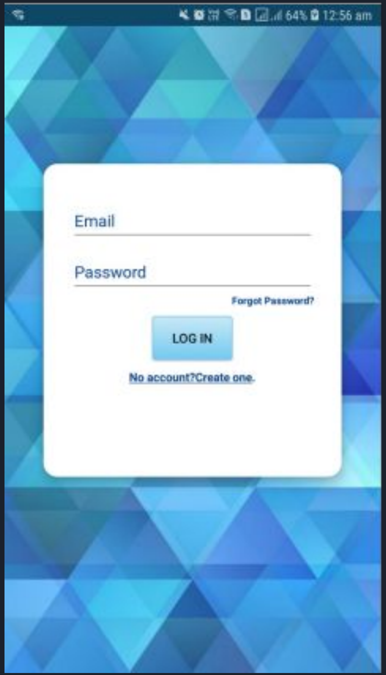
ACTIVITY FLOW

- Login page is displayed to login to our database.
- If you do not have an account you can create a new account by entering your information in create new account page.
- The user will enter the correct login credentials to start his session.
- The front page after opening the app is the dashboard , which consists of the daily expenses graph,todays total expenses and two buttons to direct the user towards expense manager and owe/debt manager.
- The owe/debt manager has the list of the people who are the part of the transaction and a button to add another element in the list.
- The element in the list is deleted after keeping it touched for long time.
- Similarly the expenditure manager will also have the same elements like the owe/debt manager(except the subparts of the element of the list).
- The dashboard will also contain a logout and instruction button.

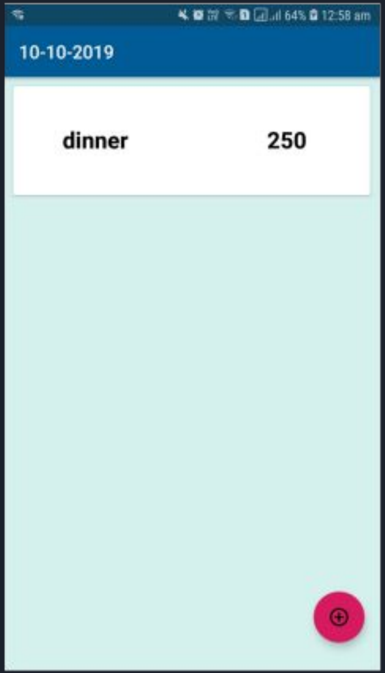
TEST CASES

- Adding 25 to the expenses manager section then the graph on the dashboard will respond by setting the point up by 25 on that day.
- Deleting the expenses manager value the graph respond by decreasing the height of point from x axis.
- Adding 25 to owe in the list, the value is added to the daily expense.
- Adding 25 to debt in the list, the value is subtracted from the daily expense.
- Deleting 25 of debt from owe/debt , 25 is subtracted from the the daily expense.
- Deleting 25 of owe from owe/debt , 25 is added to the daily expense.
- Changes get reflected on the graph as well as the total expense.

SCREENSHOTS



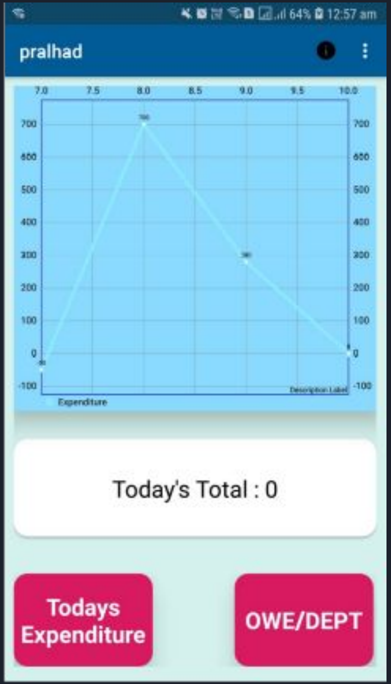
LOGIN



10-10-2019

dinner 250

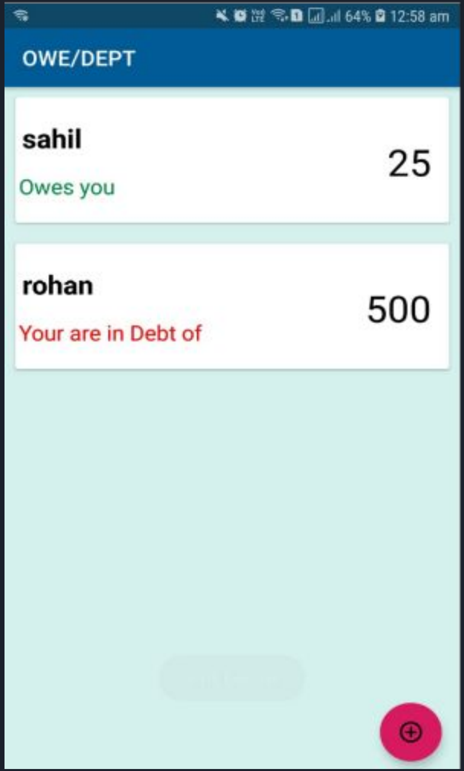
DAILY EXPENDITURE



Today's Total : 0

DASHBOARD

SCREENSHOTS



sahil

Owes you

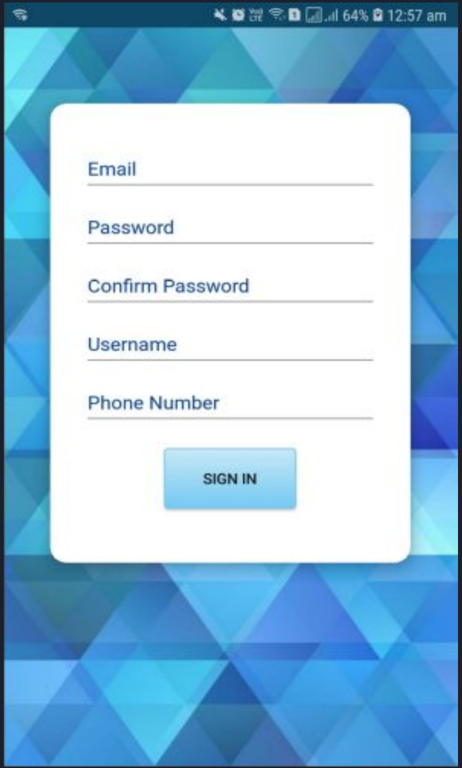
25

rohan

Your are in Debt of

500

OWE/DEBT



SIGN IN

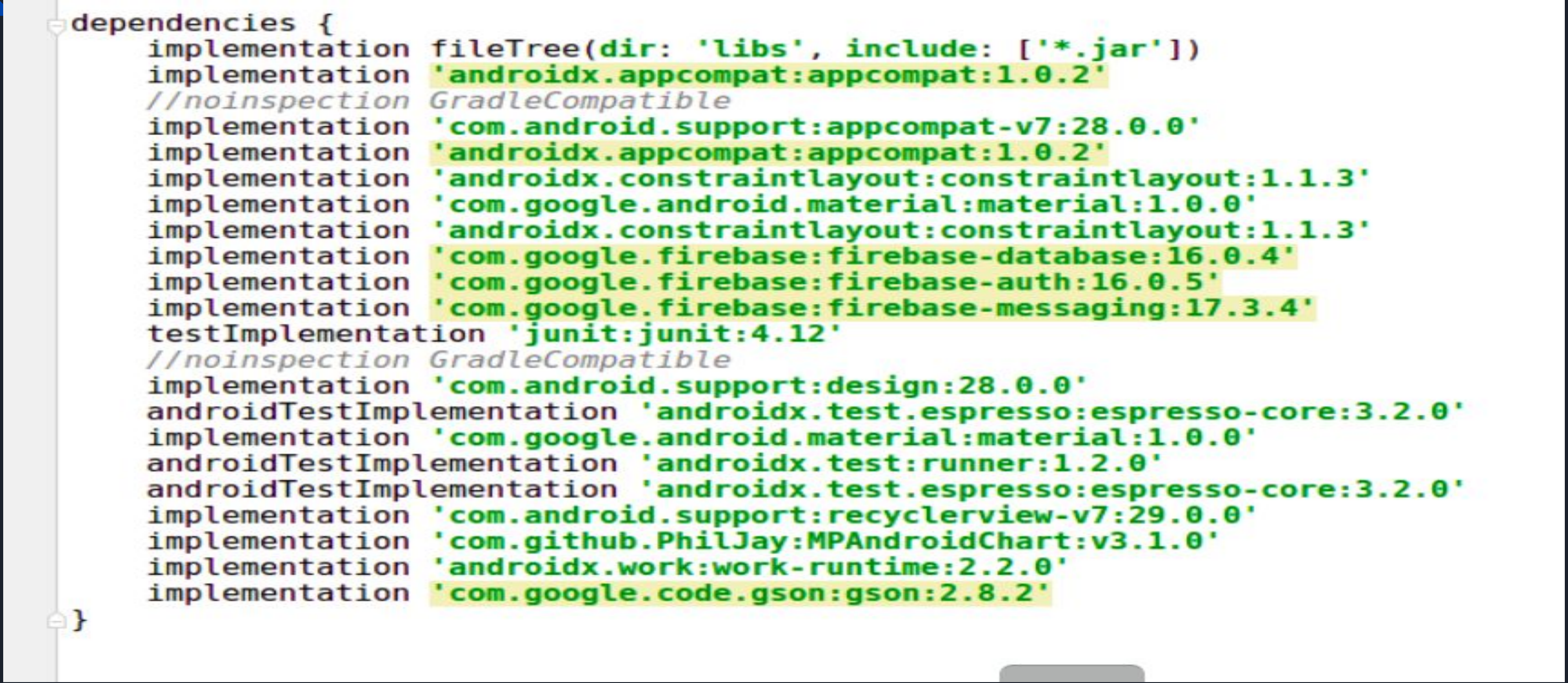
CONCLUSION

Thus we developed an app which which makes the life of a student easier to maintain a log of his/her expenditure including the amount he/she owes or is in debt , thus making it easier for student to analyze the transactions and to avoid any unnecessary transactions in the near future.

FUTURE SCOPE

- Creating an option to the user to pay for his/her expenses through the UPIs availabel
- Sending notification after a specific day if the debt is not payed or owe is not received.
- Dividing the group expenses into fair parts.
- Allow the user to take notes.
- Set a reminder for the user.

DEPENDENCIES ADDED



```
dependencies {
    implementation fileTree(dir: 'libs', include: ['*.jar'])
    implementation 'androidx.appcompat:appcompat:1.0.2'
    //noinspection GradleCompatible
    implementation 'com.android.support:appcompat-v7:28.0.0'
    implementation 'androidx.appcompat:appcompat:1.0.2'
    implementation 'androidx.constraintlayout:constraintlayout:1.1.3'
    implementation 'com.google.android.material:material:1.0.0'
    implementation 'androidx.constraintlayout:constraintlayout:1.1.3'
    implementation 'com.google.firebase:firebase-database:16.0.4'
    implementation 'com.google.firebase:firebase-auth:16.0.5'
    implementation 'com.google.firebase:firebase-messaging:17.3.4'
    testImplementation 'junit:junit:4.12'
    //noinspection GradleCompatible
    implementation 'com.android.support:design:28.0.0'
    androidTestImplementation 'androidx.test.espresso:espresso-core:3.2.0'
    implementation 'com.google.android.material:material:1.0.0'
    androidTestImplementation 'androidx.test:runner:1.2.0'
    androidTestImplementation 'androidx.test.espresso:espresso-core:3.2.0'
    implementation 'com.android.support:recyclerview-v7:29.0.0'
    implementation 'com.github.PhilJay:MPAndroidChart:v3.1.0'
    implementation 'androidx.work:work-runtime:2.2.0'
    implementation 'com.google.code.gson:gson:2.8.2'
}
```

REPOSITORIES USED

```
google()
jcenter()
maven {url 'https://jitpack.io'}
```

REFERENCES

- https://firebase.google.com/?gclid=EAIaIQobChMIkt-zm_qP5QIVjRaPCh3K-wsHEAAYASAAEgLT9fD_BwE
- https://www.youtube.com/watch?v=yrbgN2UvKGQ
- https://www.sitepoint.com/mastering-complex-lists-with-the-android-recyclerview/
- https://material-ui.com/customization/color/
- https://firebase.google.com/docs/guides
- https://developer.android.com/docs