Project name: dewe  
students: Shehab Abouelhassan (Leader) , Amin asjad (contributor )

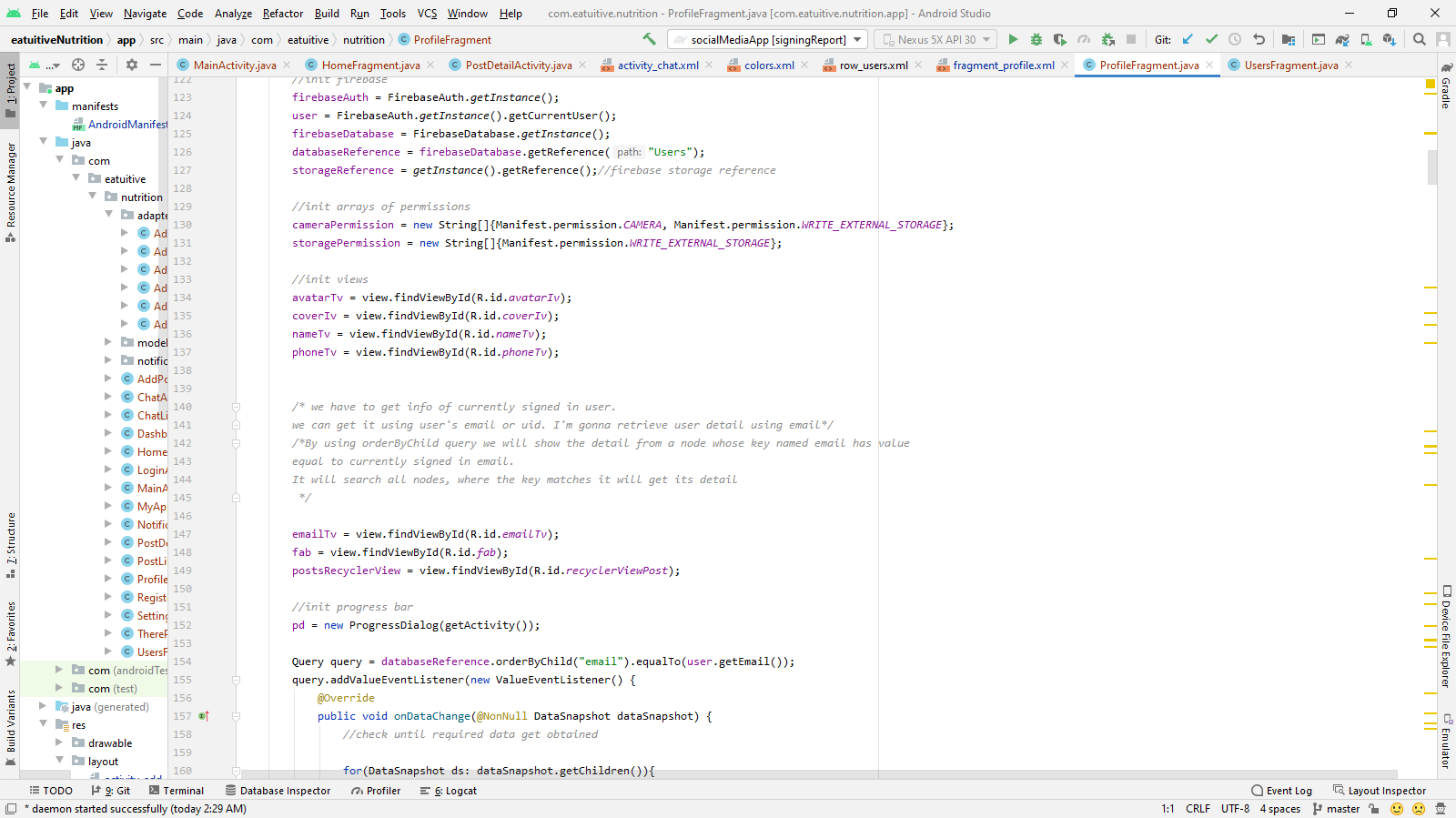
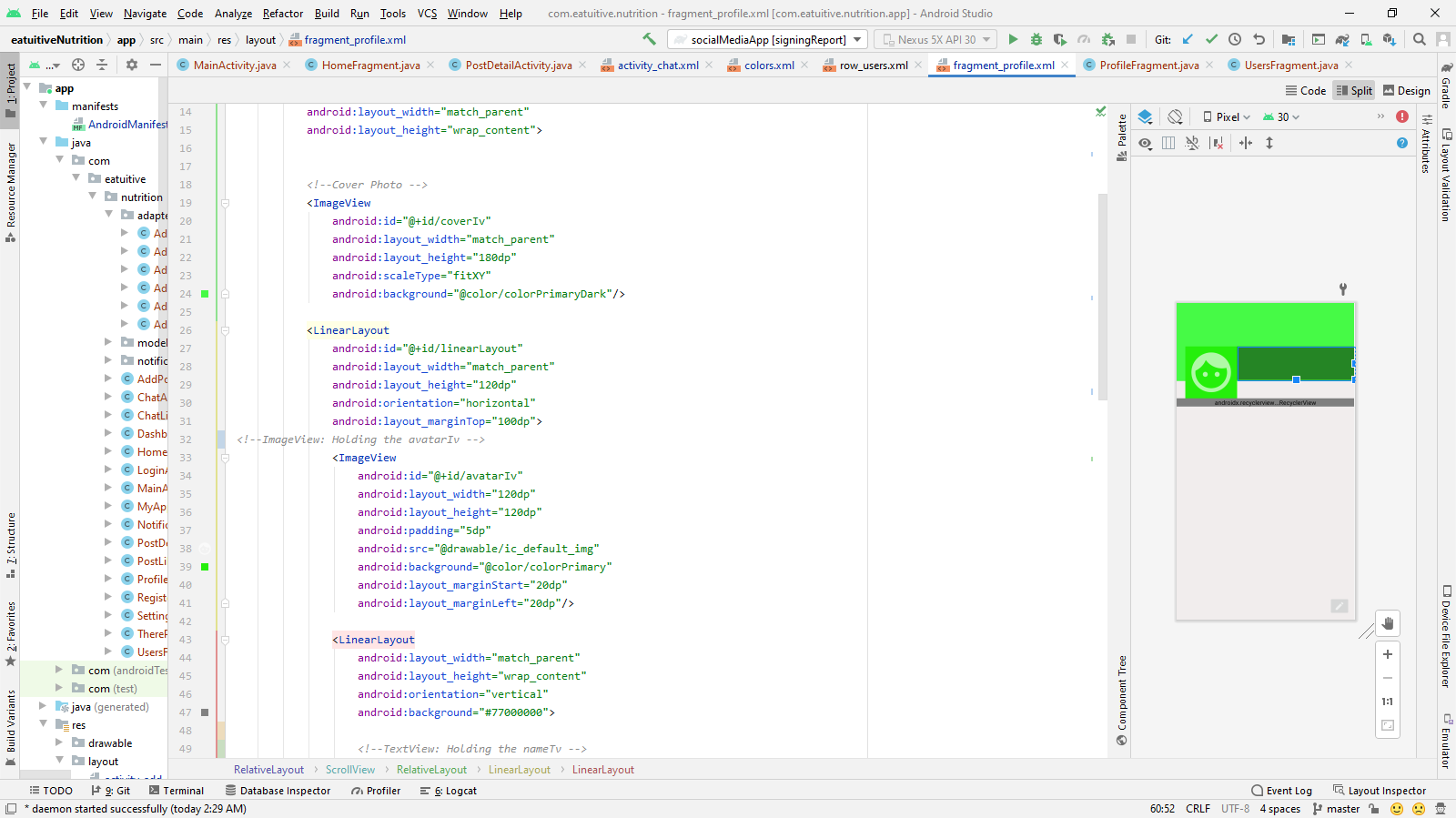
Neptun code: YOOQ6Z , JDKQJC

Reading and exploring the README.dm file is Strongly suggested for better understanding, evaluation and assessment of this project. Along side with the Final\_Report file\*\*\*\*\*\*\*\*\*

Client-side Application Logic (Parent)

Some of the #34 steps/parts of the development process and implementation for this App, Are being illustrated here, accompanied by some screenshots for both java and xml activities. i.e

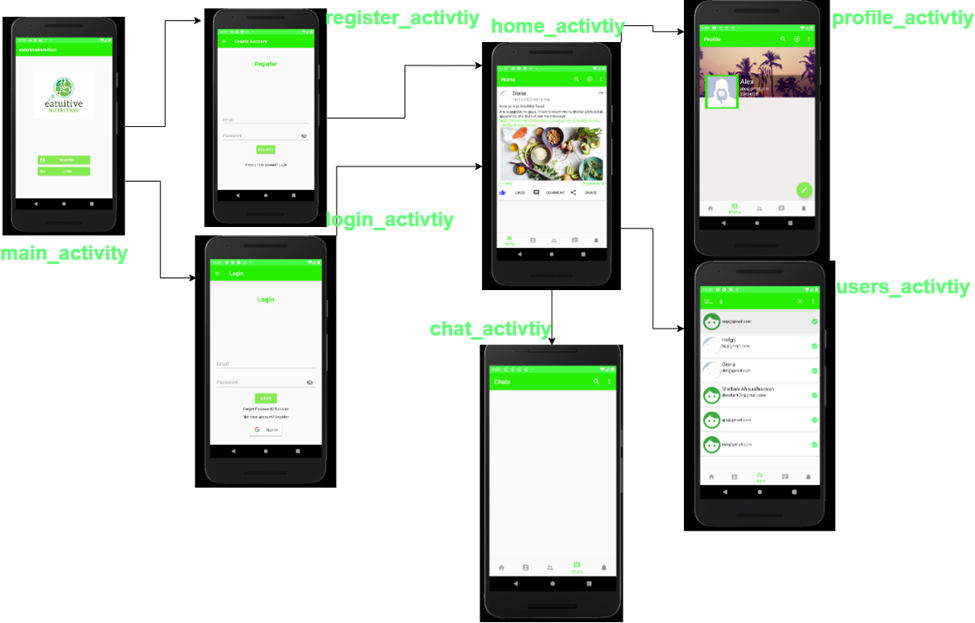
*/\*In this Part(06):  
 \* Design user profile  
 \* Get user info from firebase  
 \* Show user info  
  
In this Part(07):  
 --Edit Profile:  
 1) Name  
 2) Phone  
 3) Profile Picture  
 4) Cover Photo  
 --Requirements:  
 1) Camera & Storage Permissions (to pick image from camera or gallery)  
 2) Firebase Storage libraries (to upload & receive profile/cover photo)  
 --UI Update:  
 1) Add ImageView (for cover photo)  
 2) Add FloatingActionButton to show dialog containing options to Edit Profile  
 3) Add Default Image for Profile Picture\*/*

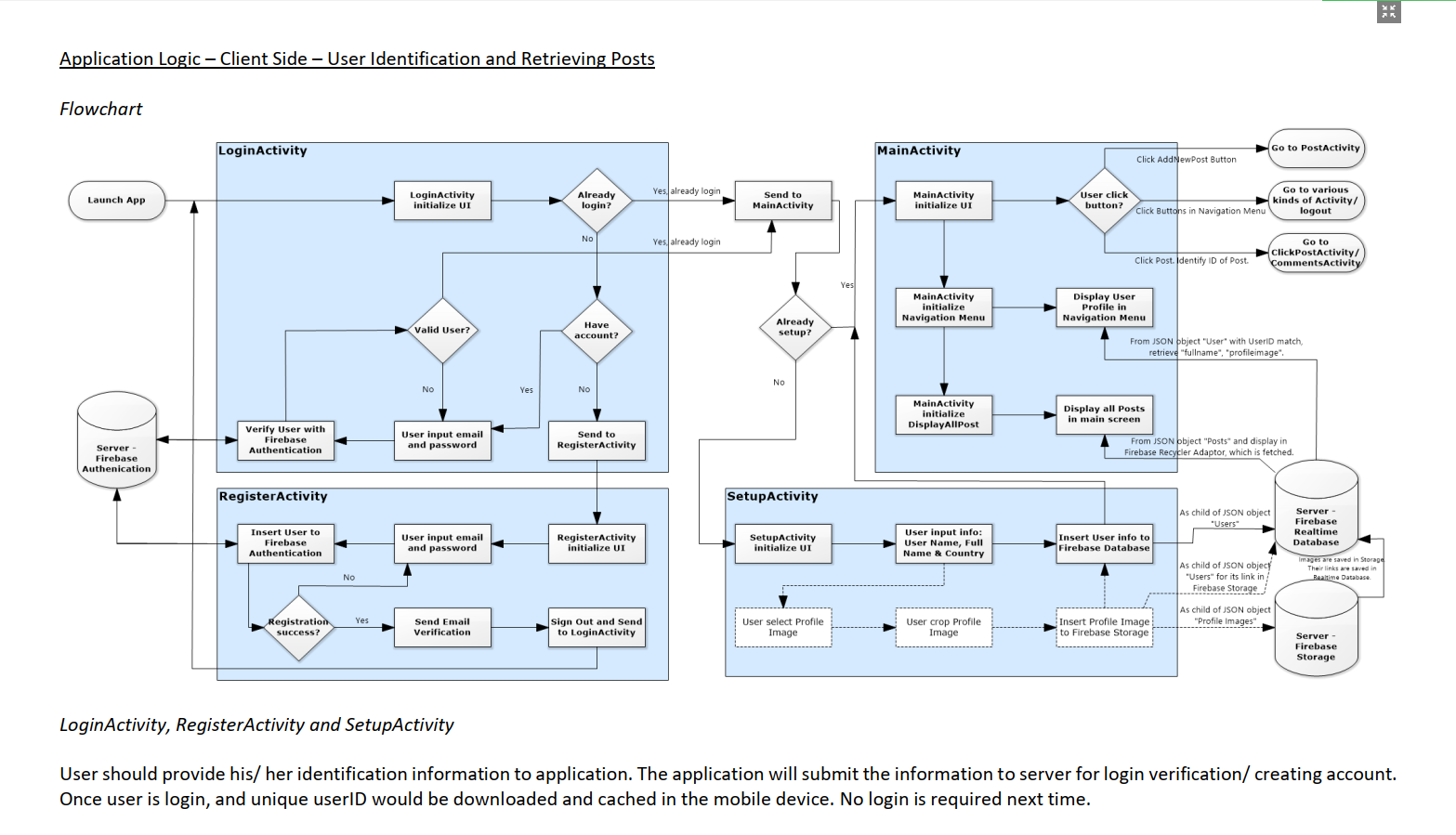
> (child)

## Overview

Our application is an Instagram-like App which supports below functions  
ω User identification by login, then retrieving posts from server  
ω Clients user profile  
ω View, comment, like and create posts  
ω Sending messages to friends

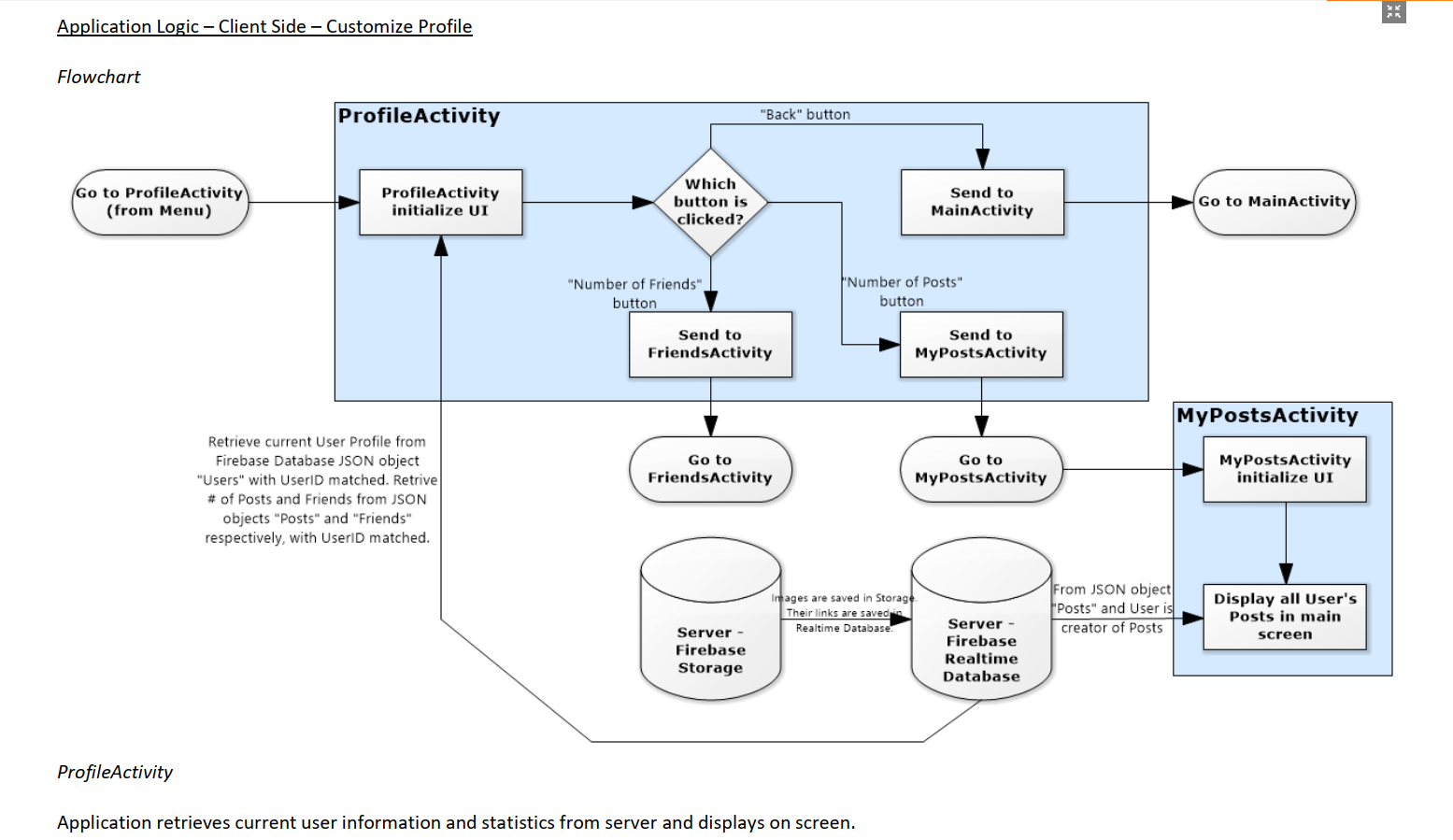


## User Identification and Retrieving Posts



*MainActivity*Application would download all posts if users submit a valid userID to server, i.e. user successfully login. The posts are fetched and showed using RecyclerView,  
which allows user to view posts even if the connection with server is lost.  
*MainActivity – Navigation Menu*User could access other functions by clicking the menu.

## Customize Profile

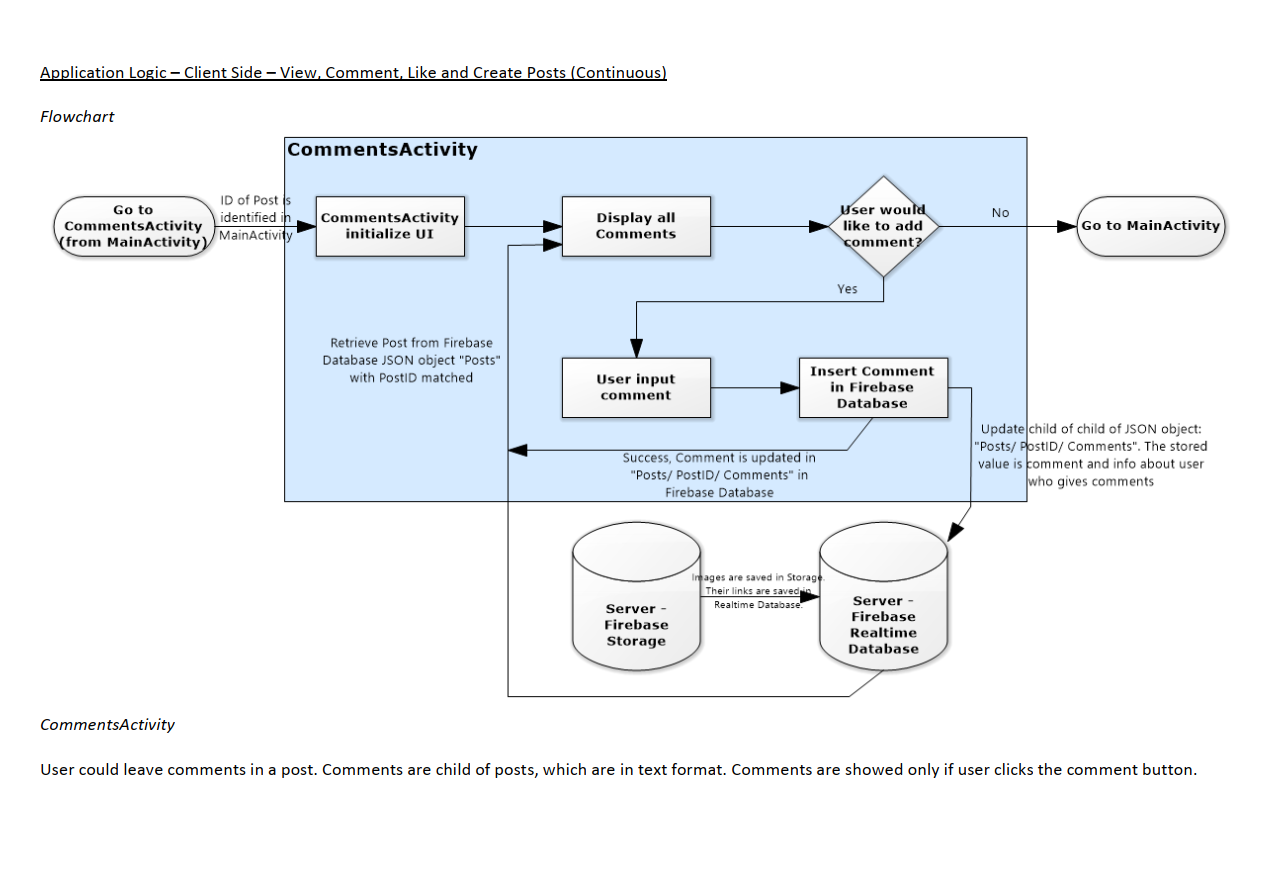


PostsActivity

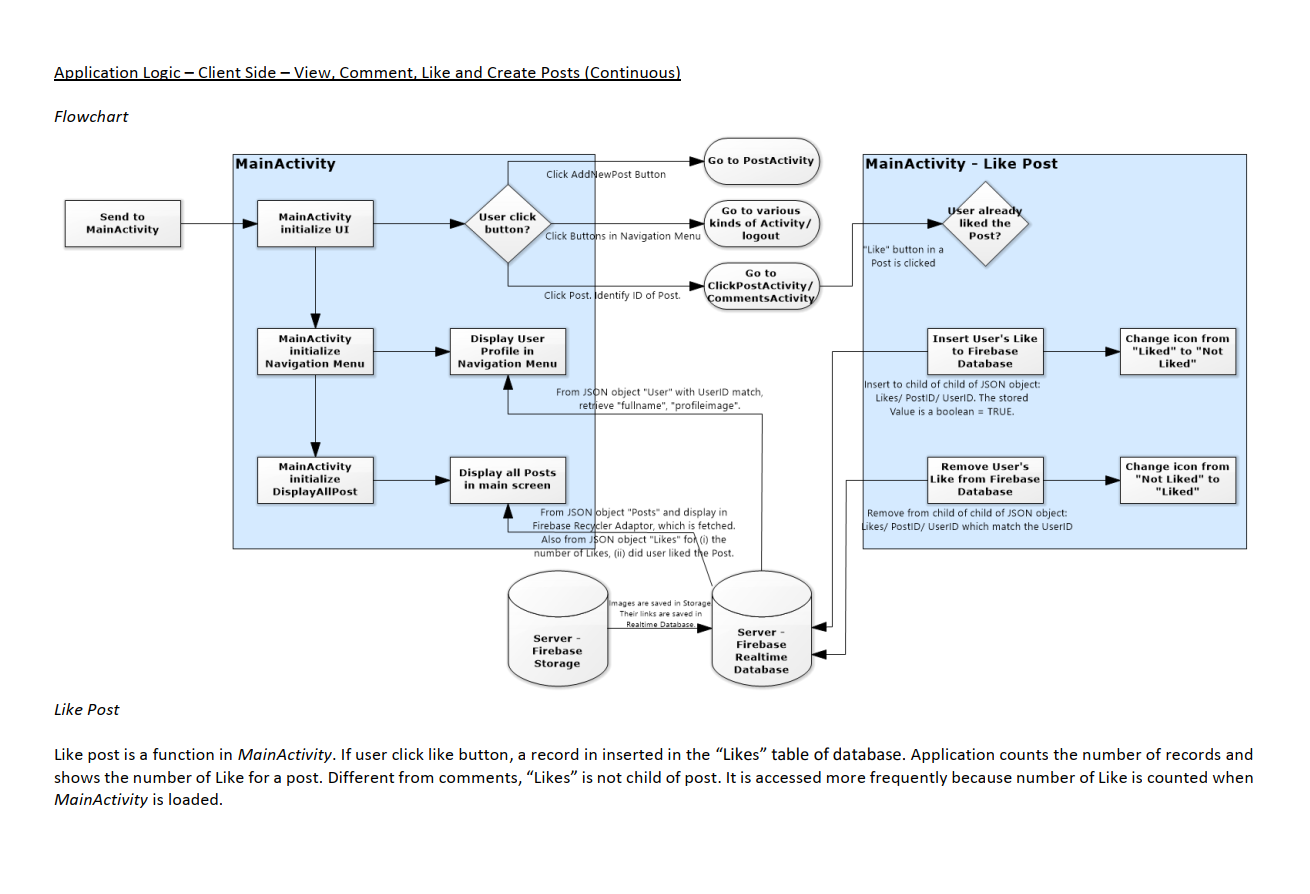
Application requests information of post created by current user by query (supported by Firebase Server) and displays on screen.

## View, Comment, Like and Create Posts

*PostsActivity*All posts are show in *MainActivity*. By clicking a post, creator of the post could send request to server for edit or delete the it. If other users click a post, they could  
see the post in full screen. Edit and deleted functions are invisible for them



*CommentsActivity*User could leave comments in a post. Comments are child of posts, which are in text format. Comments are showed only if user clicks the comment button.

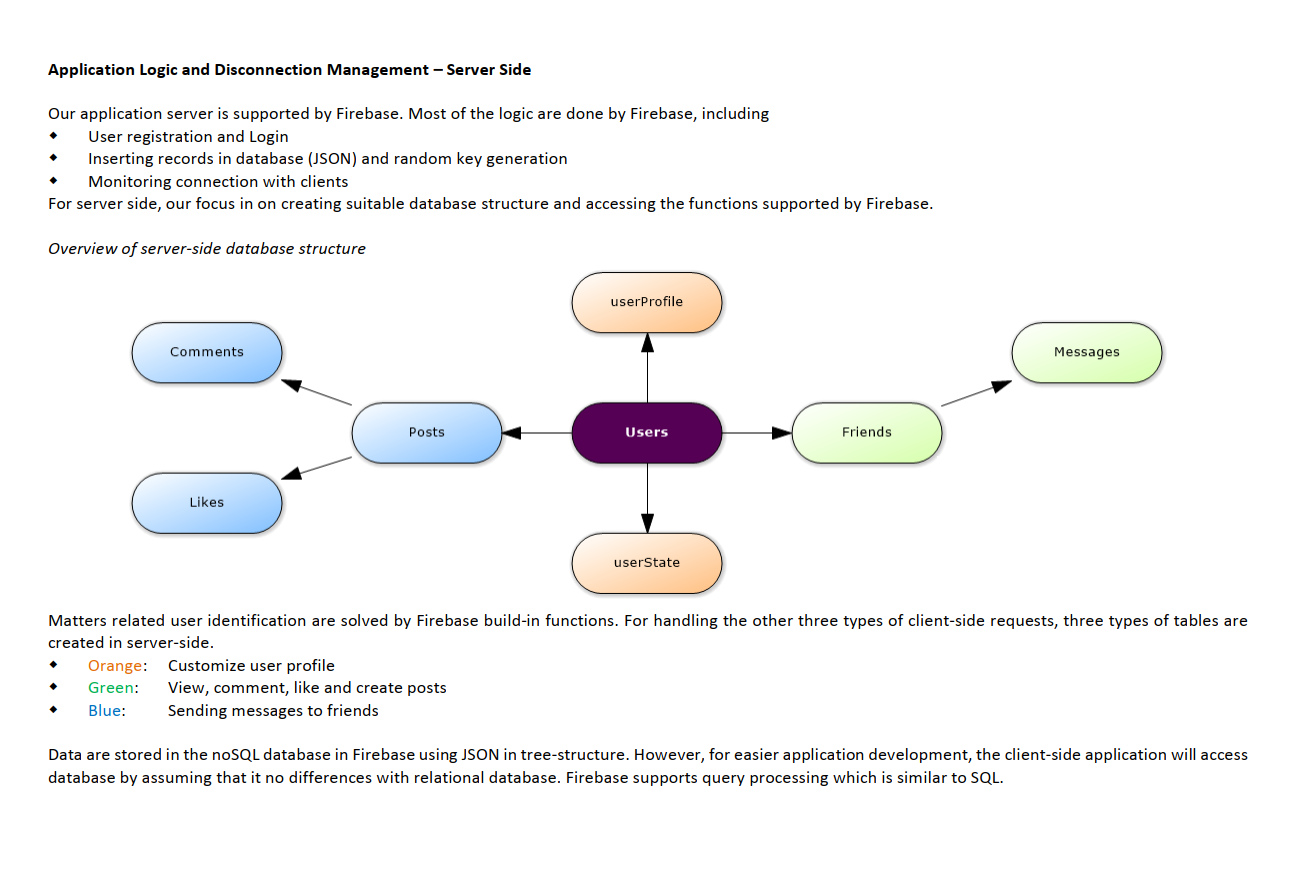


*Like Post*Like post is a function in *MainActivity*. If user click like button, a record in inserted in the “Likes” table of database. Application counts the number of records and  
shows the number of Like for a post. Different from comments, “Likes” is not child of post. It is accessed more frequently because number of Like is counted when  
*MainActivity* is loaded

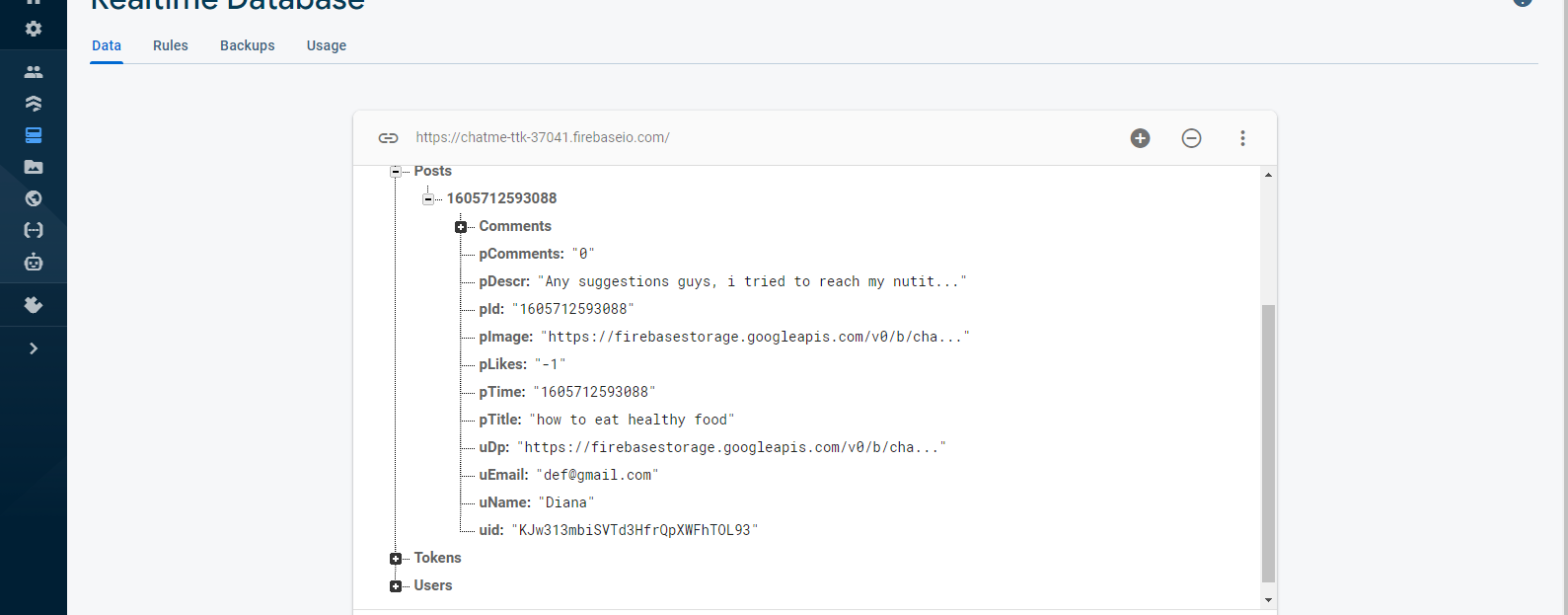
## Friends and Message

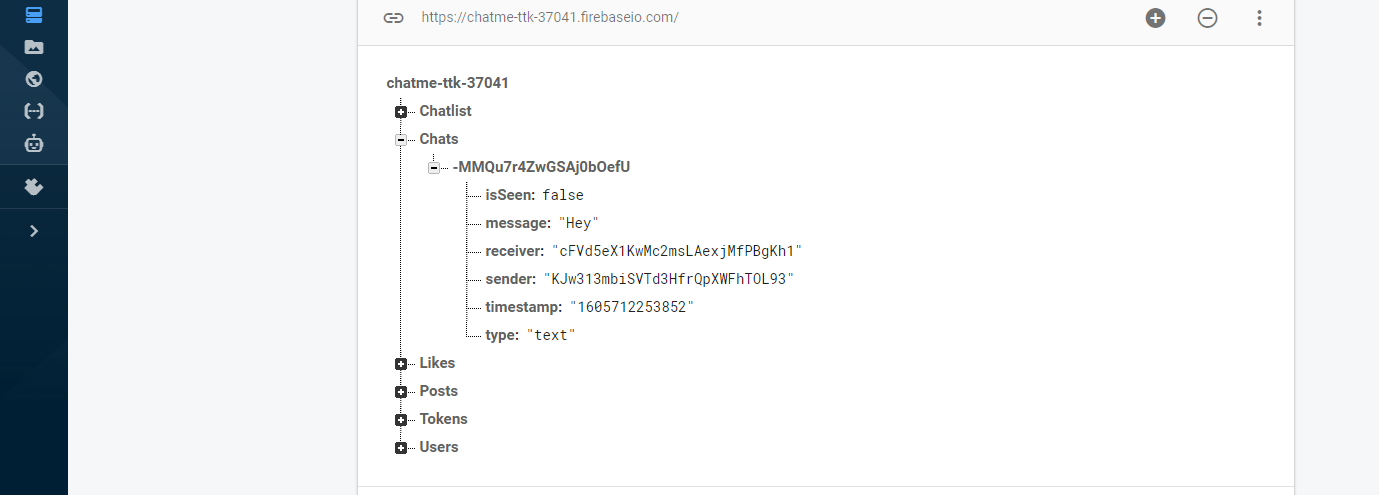


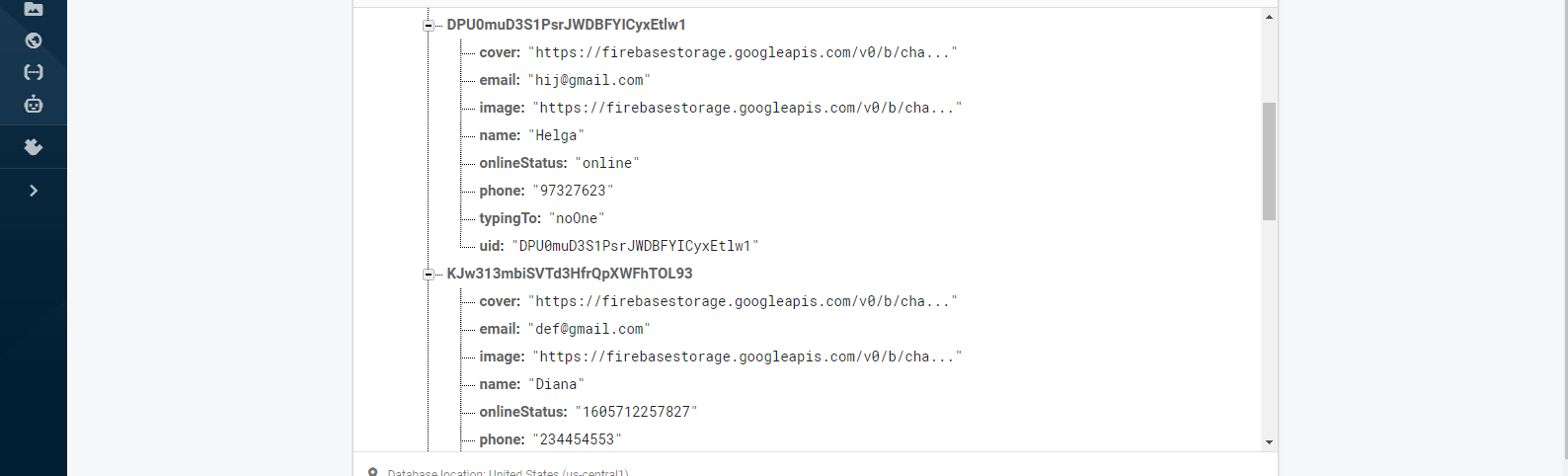
## Disconnection Management











## Server-side Database

