

# **TO-LET FINDER**

**By**

**Md. Mesbah Hossain**

**153-15-6549**

**And**

**Md. Mehedi Hasan**

**153-15-6550**

**And**

**Hasan Md. Alamin Patwary**

**153-15-6678**

This Report Presented in Partial Fulfillment of the Requirements for the Degree of Bachelor of Science in Computer Science and Engineering.

Supervised By

**Md. Riazur Rahman**

Senior Lecturer

Department of CSE

Daffodil International University



**DAFFODIL INTERNATIONAL UNIVERSITY**

**DHAKA, BANGLADESH**

**SEPTEMBER 2019**

## **APPROVAL**

This Project titled “To-Let Finder”, submitted by Md. Mesbah Hossain, ID No: 153-15-6549, Md. Mehedi Hasan, ID No: 153-15-6550 and Hasan Md. Alamin Patwary, ID No: 153-15-6678 to the Department of Computer Science and Engineering, Daffodil International University has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of B.Sc. in Computer Science and Engineering and approved as to its style and contents. The presentation has been held on 12.09.2019.

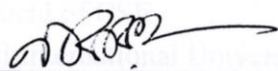
### **BOARD OF EXAMINERS**



**Dr. Syed Akhter Hossain**  
**Professor and Head**

Department of Computer Science and Engineering  
Faculty of Science & Information Technology  
Daffodil International University

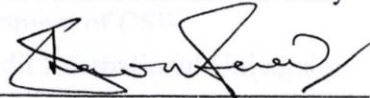
**Chairman**



**Narayan Ranjan Chakraborty**  
**Assistant Professor**

Department of Computer Science and Engineering  
Faculty of Science & Information Technology  
Daffodil International University

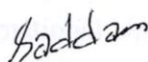
**Internal Examiner**



**Shaon Bhatta Shuvo**  
**Senior Lecturer**

Department of Computer Science and Engineering  
Faculty of Science & Information Technology  
Daffodil International University

**Internal Examiner**



**Dr. Md. Saddam Hossain**  
**Assistant Professor**

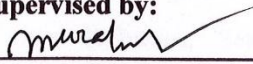
Department of Computer Science and Engineering  
United International University

**External Examiner**

## DECLARATION

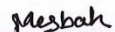
We hereby declare that, this project has been done by us under the supervision of **Md. Riazur Rahman, Senior Lecturer, Department of CSE** Daffodil International University. We also declare that neither this project nor any part of this project has been submitted elsewhere for award of any degree or diploma.

**Supervised by:**


  
\_\_\_\_\_

**Md. Riazur Rahman**  
Senior Lecturer  
Department of CSE  
Daffodil International University

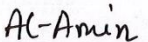
**Submitted by:**

  
\_\_\_\_\_

**Md. Mesbah Hossain**  
ID: 153-15-6549  
Department of CSE  
Daffodil International University

  
\_\_\_\_\_

**Md. Mehedi Hasan**  
ID: 153-15-6550  
Department of CSE  
Daffodil International University

  
\_\_\_\_\_

**Hasan Md. Alamin Patwary**  
ID: 153-15-6678  
Department of CSE  
Daffodil International University

## ACKNOWLEDGEMENT

First of all, we express our deepest gratitude and thanks to the almighty god for His divine blessing to enable us to complete the final year project successfully.

We are deeply grateful and express our profound indebtedness to **Supervisor Md. Riazur Rahman, Senior Lecturer**, Department of CSE Daffodil International University, Dhaka. Our supervisor gave us continual encouragement and constructive criticism. His deep knowledge and keen interest in the field of ‘Android App Development’, endless patience, scholarly guidance, constant and energetic supervision and valuable advice have made it possible to complete this project.

We wish to express our heartiest gratitude to Head, Department of CSE, for his kind help to finish our project and also to other faculty member and the staff of CSE department of Daffodil International University.

We would like to thank our entire course mate in Daffodil International University, who took part in this discuss while completing the course work.

Finally, we must acknowledge with due respect the constant support and patients of our parents.

## **ABSTRACT**

Bangladesh is a very small country with over population, specially Dhaka city. Most of people who live here come from outside of Dhaka city. People come here mostly for studying and to find jobs. But firstly, they need to find a good place to live in this city. And this is a very tough thing to find a good place to live in Dhaka city. It costs huge amount of time to travel through the city and find to-lets. This project will help people to save their time, money and other sufferings while finding to-let. This is basically an android application. So, an android device will be needed to run this application. This is a very user-friendly application so users won't face any difficulties using this app.

The aim of our project is to help people find to-lets online, sitting their home or anywhere using their android devices. We followed the basic procedure of making an android application using the android application development tool "Android Studio". As a result, we finally get to see to-let sings in the main page which is a google map activity. And then we can also add our own post of to-let and it will automatically be added in the map. After the development of this application we have tested it in different android devices by different user and found it very efficient and well build.

## TABLE OF CONTENTS

<b>CONTENTS</b>	<b>PAGE</b>
Board of examiners	i
Declaration	ii
Acknowledgements	iii
Abstract	iv
<b>CHAPTERS</b>	
<b>CHAPTER 1: Introduction</b>	<b>01 - 03</b>
1.1 Introduction	01
1.2 Motivation	01
1.3 Objectives	02
1.4 About our app	02
1.5 Expected Outcomes	02
<b>CHAPTER 2: Background</b>	<b>04 - 05</b>
2.1 Introduction	04
2.2 Related Work	04
2.3 Our Project Work	05
2.4 Scope of the Problems	05
2.5 Challenges	05

**CHAPTER 3: Requirement Specifications** **06 - 08**

3.1 Introduction	06
3.2 Requirement Collection and Analysis	06
3.3 Use Case Diagram	07
3.4 Business Process Model	08

**CHAPTER 4: Design Specifications** **09 - 21**

4.1 Front End Design	09
4.2 Back End Design	09
4.3 UI Design	10
4.4 Activity Layouts	10
4.4.1 Splash Screen	12
4.4.2 User Verification Activity	13
4.4.3 New Profile Activity	14
4.4.4 Main Activity	15
4.4.5 Ad Post Activity	16
4.4.6 To-let List Activity	17
4.4.7 Post Details Activity	18
4.4.8 Navigation Drawer Activity	19
4.5 Implementation Tools	20

4.5.1 Android Studio	20
4.5.2 Android SDK	20
4.5.3 JDK	20
4.5.4 Android Device	21
4.5.5 Permission	21
4.5.2 Android SDK	21
<b>CHAPTER 5: Testing and Result</b>	<b>22</b>
5.1 Testing	22
5.2 Test Result	22
<b>CHAPTER 6: Discussion and Future Plan</b>	<b>23</b>
6.1 Discussion	23
6.2 Limitations	23
6.3 Future Plan	23
<b>CHAPTER 7: Conclusion</b>	<b>24</b>
7.1 Conclusion	24
<b>APPENDIX</b>	<b>25 - 28</b>
<b>REFERENCES</b>	<b>29</b>



## LIST OF FIGURES

<b>FIGURES</b>	<b>PAGE</b>
Figure 3.3.1: Use Case Diagram	07
Figure 3.4.1: Business Process Model	08
Figure 4.4.0: Activity Layout Flowchart	11
Figure 4.4.1: Splash Screen	12
Figure 4.4.2: User Verification Activity	13
Figure 4.4.3: New Profile Activity	14
Figure 4.4.4: Main Activity	15
Figure 4.4.5: Post New Ad Activity	16
Figure 4.4.6: To-let List Activity	17
Figure 4.4.7: Post Details Activity	18
Figure 4.4.8: Navigation Drawer Activity	19
Figure A-1: App level gradle file	25
Figure A-2: XML code of Ad Post activity	25
Figure A-3: Java code of the Main activity	26
Figure A-4: Java code of the Map activity	26
Figure A-5: Firebase authentication table	27
Figure A-6: Firebase Realtime database table	27
Figure A-7: Firebase storage table	28

# CHAPTER 1

## Introduction

### 1.1 Introduction

Bangladesh is a densely over populated country. And most of the cities in Bangladesh they don't have enough workplaces to give people jobs. Not only that they also have deficiency of education institutions as well. So, people from small towns or villages come to big cities to find jobs and educations. Especially in Dhaka city, people come from all over the country. But when someone comes to a new place, at first, he/she has to find a place to live. But this is a very difficult and troublesome work for someone to find a good place to live in big cities. It costs a huge amount of time and energy to travel through the city and find to-lets. And those who have jobs or other works in day times we can't even think about the situations that they go through. And as students we also have to go through a lot in order to find to-lets. If we have a way to find to-lets from anywhere at any time without wasting our time and sufferings who wouldn't want that. Our project does the exact thing for the users.

### 1.2 Motivation

We have been living in Dhaka city for almost 4 years. Since we came here, we are seeing the sufferings of our friends, brothers and other familiar persons whenever they have to change their current places and find a new place for living such as hostels, messes, sublets or flats. We also had to face those problems many times ourselves. This wastes lots of our time, money and energy. We also saw that those who have empty flats or rooms that they want to provide others on rent have to make ads, print them on posters and stick them on the walls of other buildings. This is also a very troublesome work. So, from that we have come up with this idea to make an android application which will solve these problems.

### **1.3 Objectives**

- User authentication.
- Show to-lets around the user on google map.
- Show the list of to-lets.
- Find to-lets using filters.
- Filter to-let by date, area or amount of rent.
- Show all the details of to-let with image.
- Show the details of owners with contact number.
- Make posts of to-lets.
- Add posts to favorites.
- Delete own posts.

### **1.4 About Our App**

This is actually an android app. And it needs internet connection and location information to work fully. Using this app, we can find to-lets around us, know every detail we need to know about the to-let like is it for bachelor or for family. Then we can have the information about the owner as well as the contact number of the owner to contact with him/her. We can also be able to see the picture of the room if the owner provides that. The user can also post ads of his/her own to-lets. User can save the location of his/her to-let manually on the google map.

### **1.5 Expected Outcomes**

- User has to be verified using phone number at first.
- User has to make a profile.
- User will see his/her current location on google map.
- User will see markers of to-lets around him/her on google map.
- Color of the markers will be varied based on the category of the to-let.
- Category of a to-let can be seen by tapping on the marker of the to-let.

- User can see all the details that are needed about a to-let.
- User can see the information of the owner of a to-let as well.
- User can make posts of to-let.
- User can have the location selected of his/her current position automatically or any other position manually on google map.

## **CHAPTER 2**

### **Background**

#### **2.1 Introduction**

Ours is an android based project. So, in order to work on this project and to make our application we had to work with android studio. But we don't know that much about android app development. But we had faith on ourselves and we had that will power that helped us completing such a difficult task like this by ourselves. All we had was the internet and YouTube tutorials [1, 2].

#### **2.2 Related Work**

When we came up with the idea of our final project and decide to make an android app, we looked through google play-store to find out if there is an app existing like ours or not. And the thing is we had found one but that app was not that well developed and decorated. And the layouts and the format of that app is totally different than ours. Our app is very unique and user friendly. Our apps main activity layout is more like "Uber" or "Pathao" app but their purpose is totally different. Our app serves totally a different purpose than "Uber" or "Pathao". We tried our level best to create something totally different and unique.

#### **2.3 Our Work**

We decided to make an android application. This application will help people find to-let around them or anywhere in city they want. In our project we have tried to focus on what the user needs. That's why we tried to make our app the most user friendly. It is an android app that's why we had to use "Java" and "XML" programming languages. We have used the Firebase Realtime Database and Firebase Storage for our storage purpose. And we have used the Firebase Authentication service to verify the user [3-5].

## **2.4 Scope of Problem**

Now a days there still many android users exists who do not understand much about how to use many android applications. So, this is a problem. But our app is so user friendly that we hope there won't be any case like this with ours app.

Like any other software or application android applications also have many limitations. And we also know that android version and android API updates constantly. With every update many things change. So, in that case many new problems can arise in our app. But we are hopeful we will be able to overcome those problems easily.

## **2.5 Challenges**

In order to make an app which is very different and unique than others the developers have to face many challenges and obstacles. And we are the newest in this field. So, in our case challenges are more. Our app is a helpful app. It's like a bridge between the to-let seeker and the to-let provider. But in this modern world of Facebook, YouTube, WhatsApp and Instagram it is very tough to get people use our app at first.

Then again, we are very concern about the security of the users. We want to provide absolutely valid information through our app. So, this is a big challenge for us. Then to make our application more user friendly we had to work very hard on the user interface. Had to some research and survey. Had to change the interface and the layout designs many times. This was a very difficult and challenging task. But the biggest challenge for us was to create and sort the database. We had to create the database very deliberately so that it could work efficiently. And we had to change the database so many times in order to do that. There were many other small challenges that we had to face during the development of our application.

## **CHAPTER 3**

### **Requirement Specifications**

#### **3.1 Introduction**

In order to make an app or any software or any other thing, at first, we have to know about the requirements. And as we were making this kind of application for the first time, we had to do some research and asked some people about their requirements on this kind of applications. We have described the requirement related work in this chapter.

#### **3.2 Requirement Collection and Analysis**

Searching for to-let throughout the city is a very common scenario. Normal people can't do anything about it. But we are living in the world of internet. There's nothing that we cannot find on the internet. So, why not to-let. From that thought we decided to make a platform which will fulfill all the necessary requirements of the user.

The most essential requirements we found that had to be fulfilled are:

- Creating a mobile platform which will be android based.
- People can search for to-let.
- To-let providers can post ads with all the necessary details.
- People can get the location of the to-let without going there physically.
- People can see the image of the room before going there.
- To-lets around someone's current location should be shown on google map.

After gathering all the requirements, we started to analyze them and discuss ourselves how to fulfill all the requirements with an android application. It took a lot of time and we have learnt a lot while doing these.

### 3.3 Use Case Diagram

Use case diagram is a very essential thing to understand the relationship between the user and the application. We get to understand how the user and the application will interact with each other through the use case diagram.

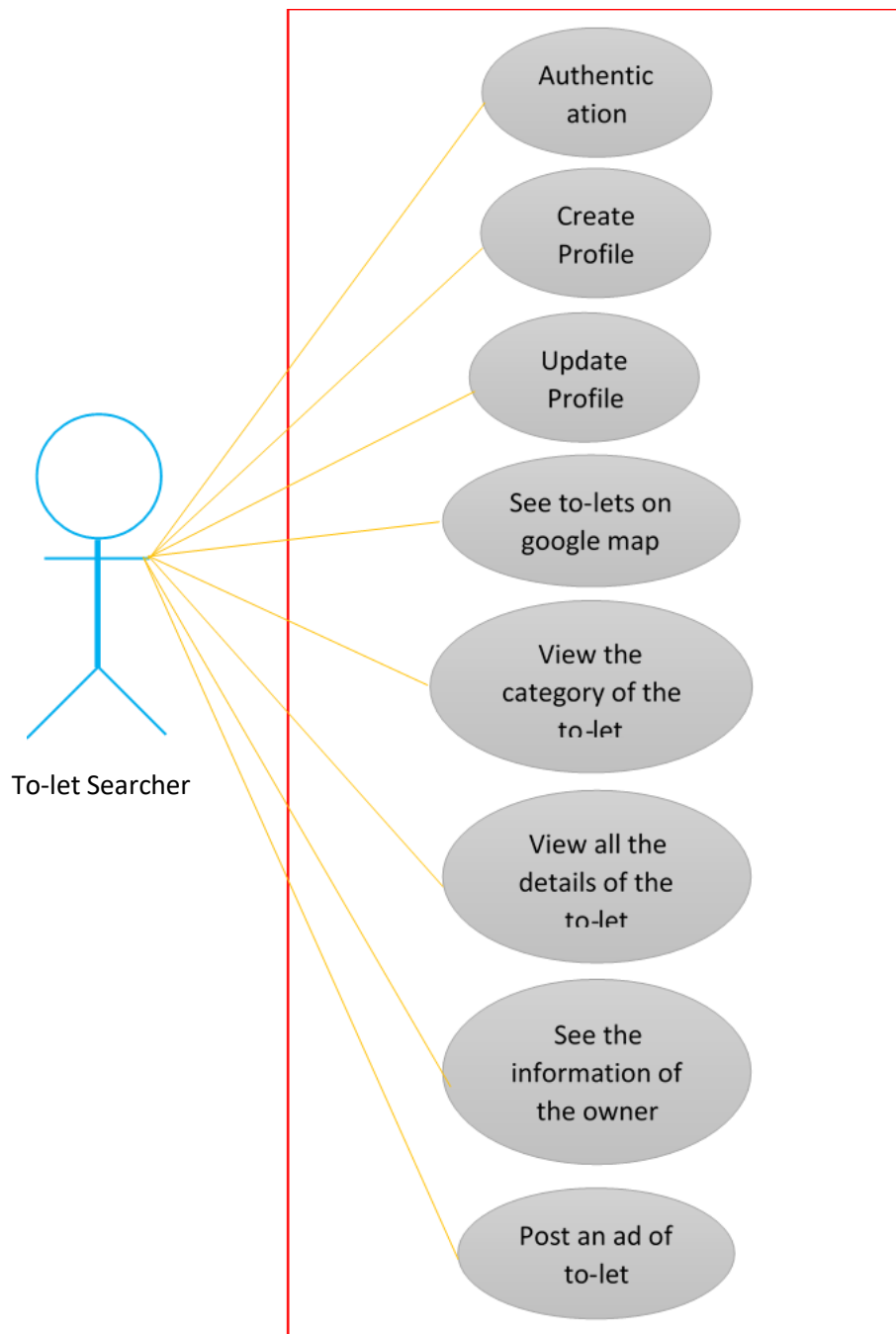


Figure 3.3.1: Use Case Diagram



### 3.4 Business Process Modeling

Business process modeling is the graphical representation of a systems workflow. This is usually done through different graphing methods, such as the flowchart, data-flow diagram, etc.

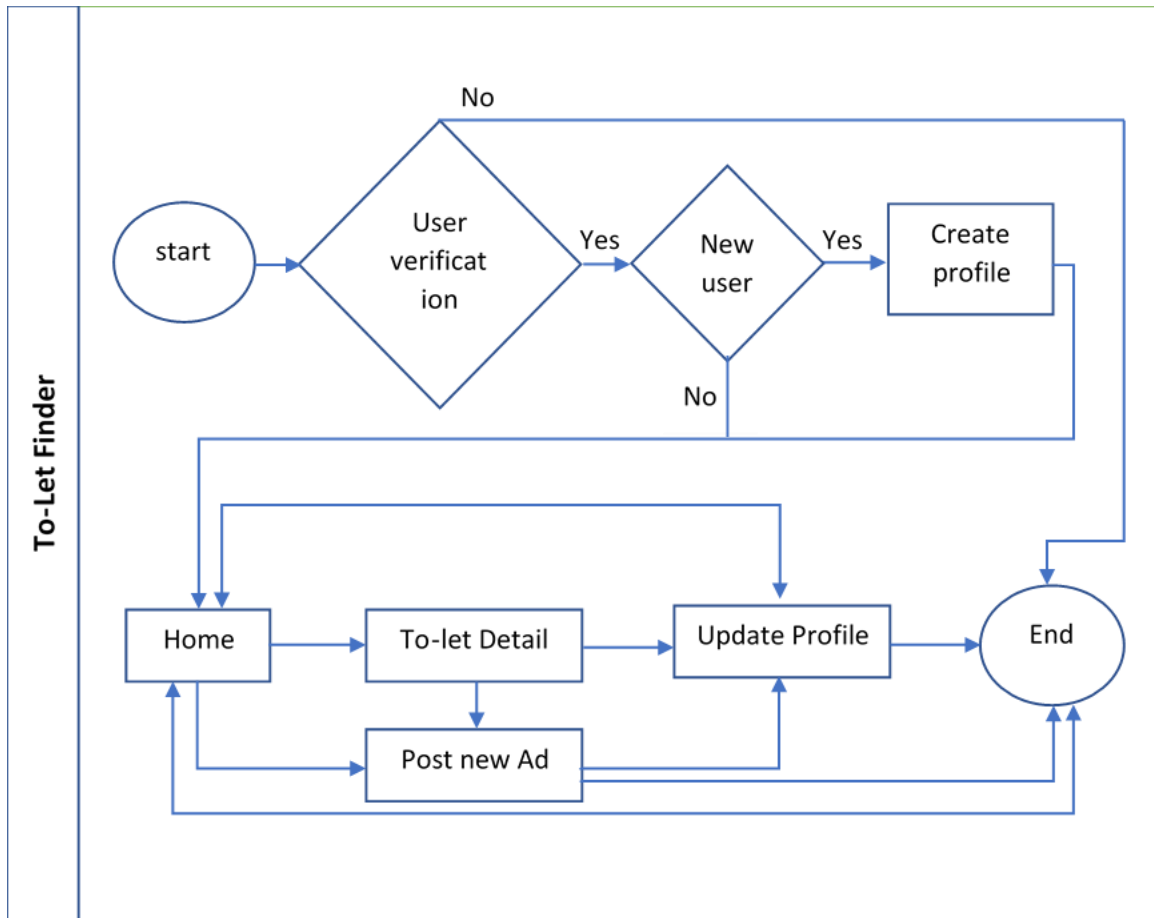


Figure 3.4.1: Business Process Model

## **CHAPTER 4**

### **Design Specifications**

#### **4.1 Front End Design**

Basically, front-end design means the graphical form of a system. It includes the overall interface users see when they enter in a system or open an app. It is a very important part for every system because the design attracts the user on the first place. So, the front-end design should be very beautiful, creative and attractive so that it can attract people and make them interested on the app. We had to work on the front-end design of our project so hard because we thought that the design should be simple as well as attractive. Because if the front-end design or in other word the user interface is complex then many user won't be able to understand how to use the app and can become frustrated and lose interest on the app. The front-end design makes the interaction between the system and the user.

#### **4.2 Back End Design**

Back-end design also known as server-side design determines the behavior of a system or application. It is the main portion or the core part of a project which controls everything. It is the most important and critical part of a project. The structure, content management and security of the system depends on the back-end design.

For our applications back-end design we have used Java language for the activities and XML language for creating the user interface to make the interaction of front-end with the database.

#### **4.3 UI Design**

A user uses an application through the user interface (UI) because it is what a user sees when he/she open an application. It basically refers to the graphical form or the front-end design of an application. After finalizing the utility requirement, we had to focus on the UI design. We had to keep in mind that the UI should be simple and user friendly.

## **4.4 Activity Layouts**

1. Splash Screen
2. Add Phone Number
3. Phone Number Verification
4. Create Profile
5. Main Map Activity with To-lets
6. List of To-lets
7. Lo-let Details
8. Update Profile
9. Add New Post of To-let
10. Location Selection

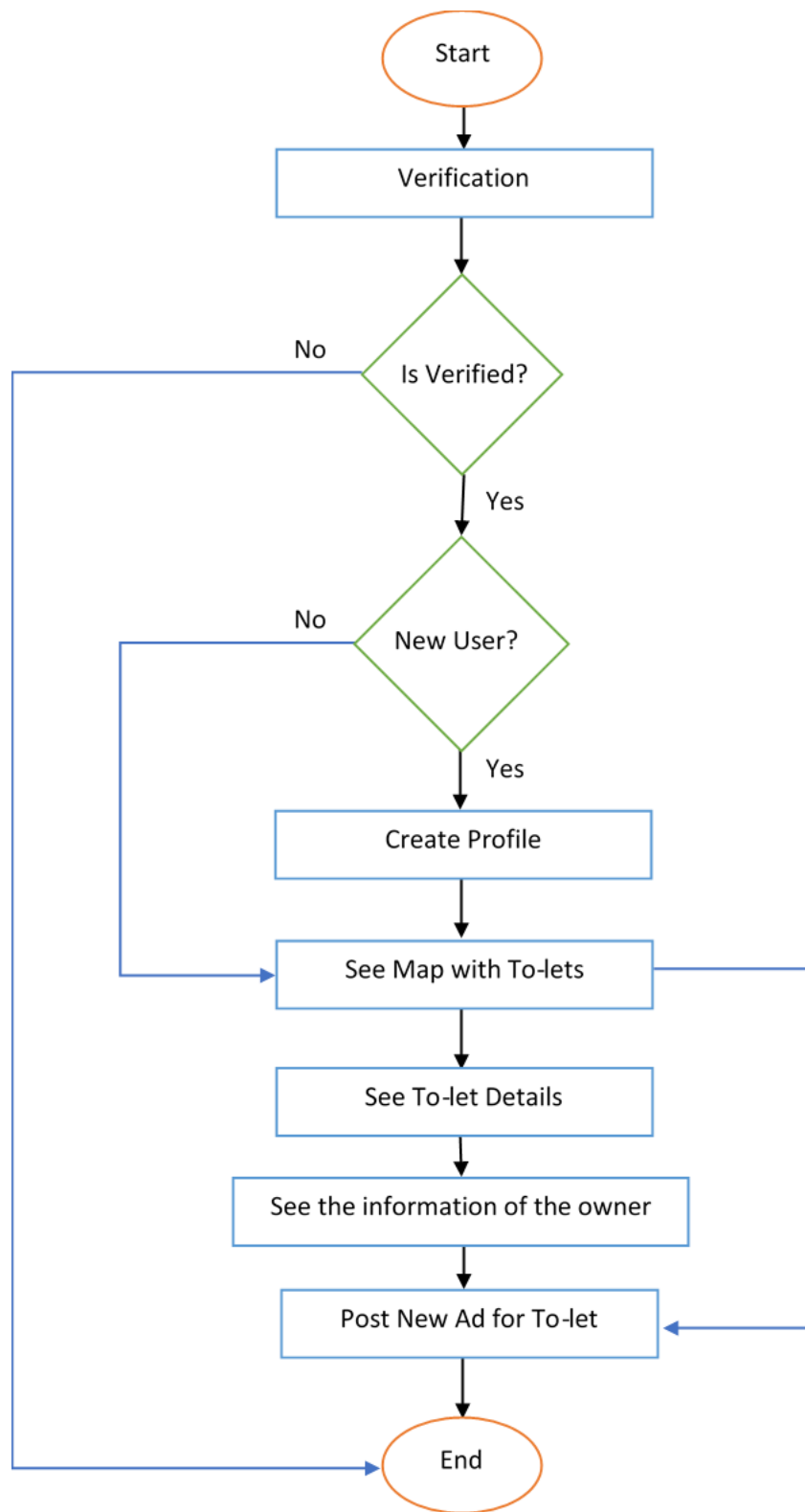


Figure 4.4.0: Activity Layout Flowchart

#### 4.4.1 Splash Screen

This is the screen shot of our splash screen. This screen will come first whenever anyone opens the app. This screen will stay for 1 second only. Then the home page will appear.



Figure 4.4.1: Splash Screen

## 4.4.2 User Verification Activity

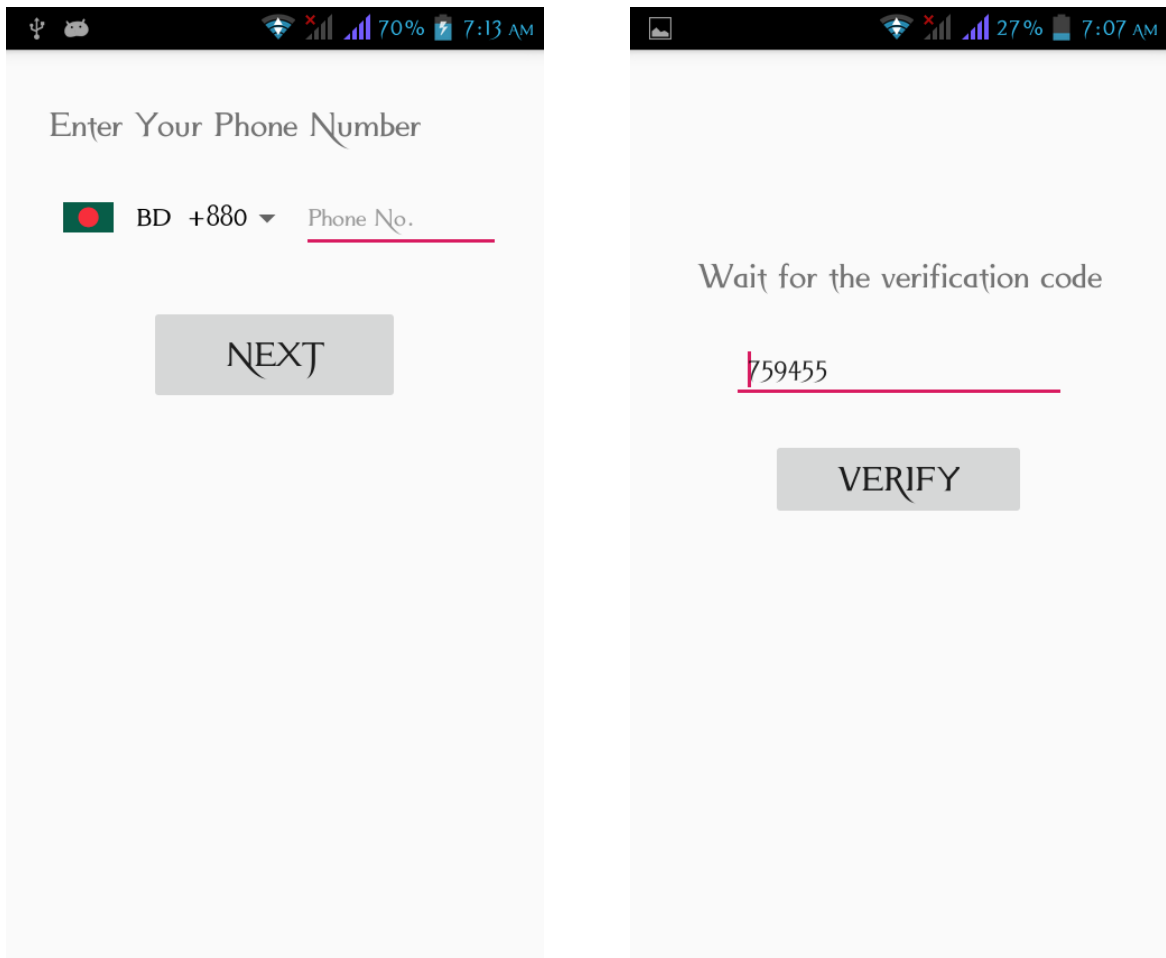


Figure 4.4.2: User Verification Activity

### 4.4.3 New Profile Activity

Here new users have to create a profile.

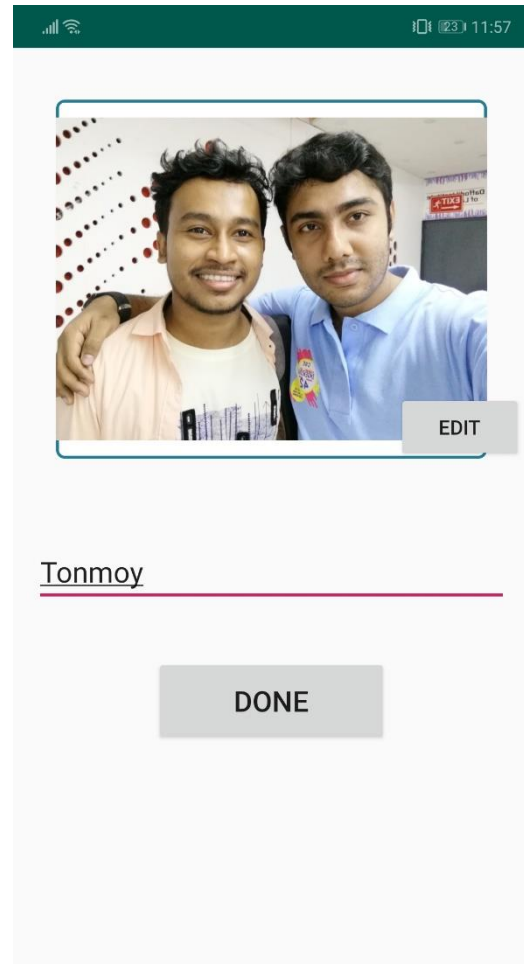
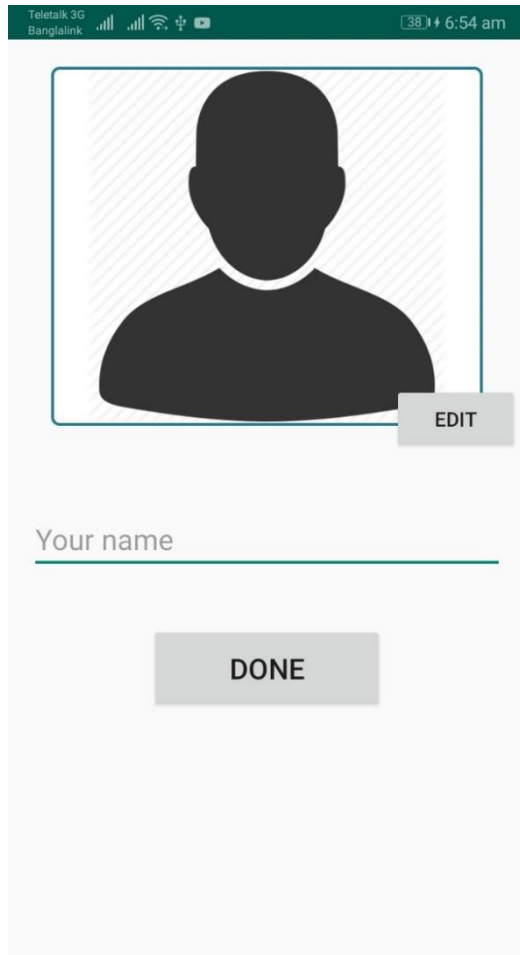


Figure 4.4.3: New Profile Activity

#### 4.4.4 Main Activity

This is the main activity or we can say the home page of our application. Here we can see the user's current location and to-let markers around him.

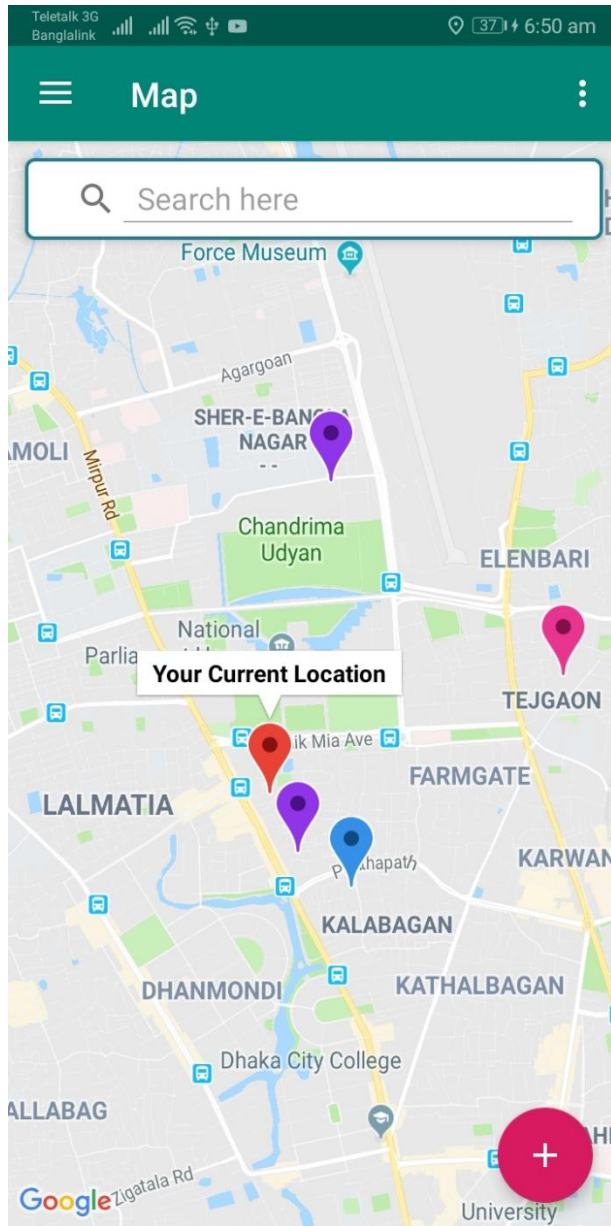


Figure 4.4.4: Main Activity



## 4.4.5 Ad Post Activity

In this page user can post his own to-let ad. After clicking on the select location button the map activity will open where the user can search and select the location.

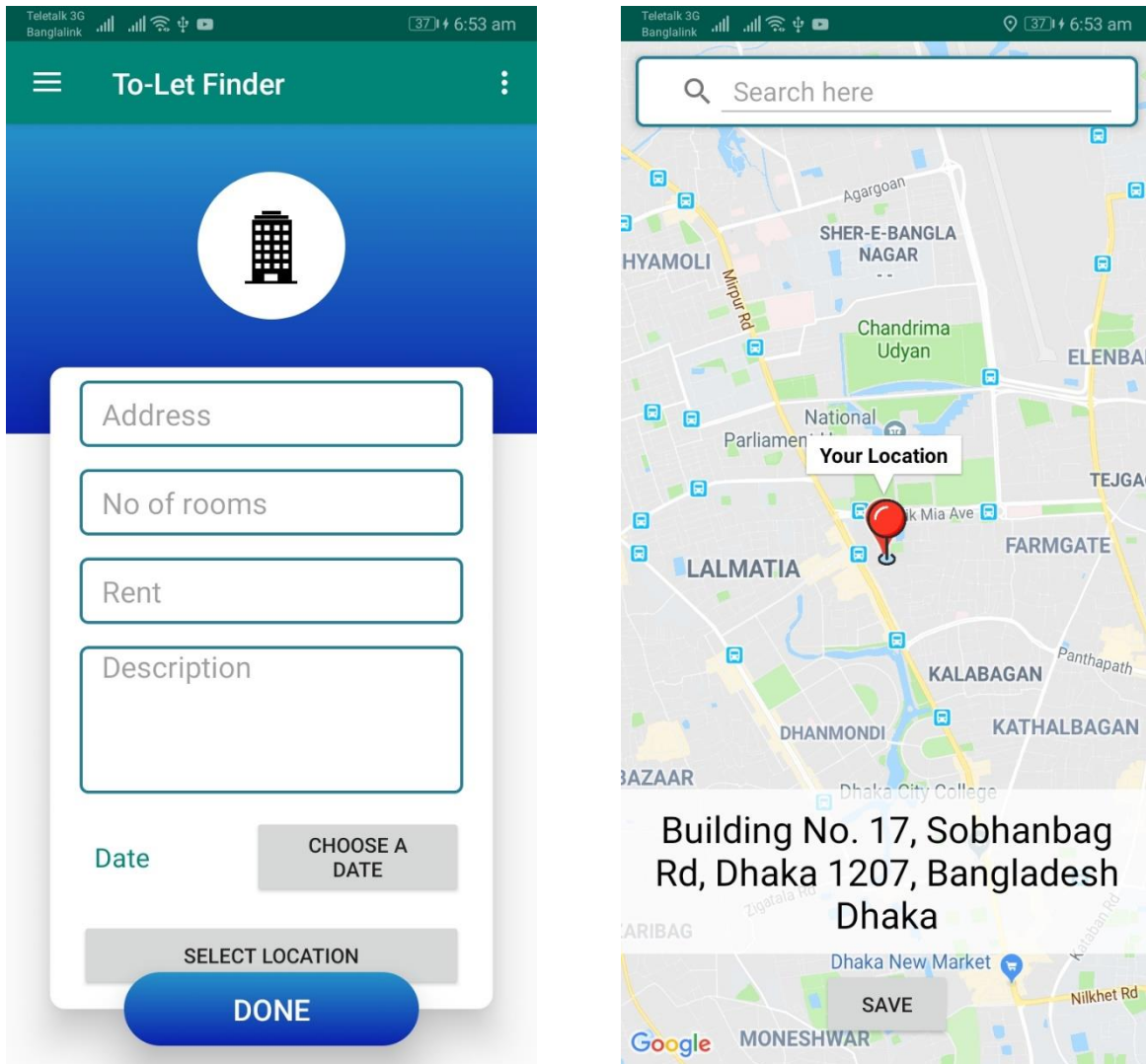


Figure 4.4.5: Post New Ad Activity

## 4.4.6 To-let List Activity

In this activity user can see all the to-lets in the form of a list view.

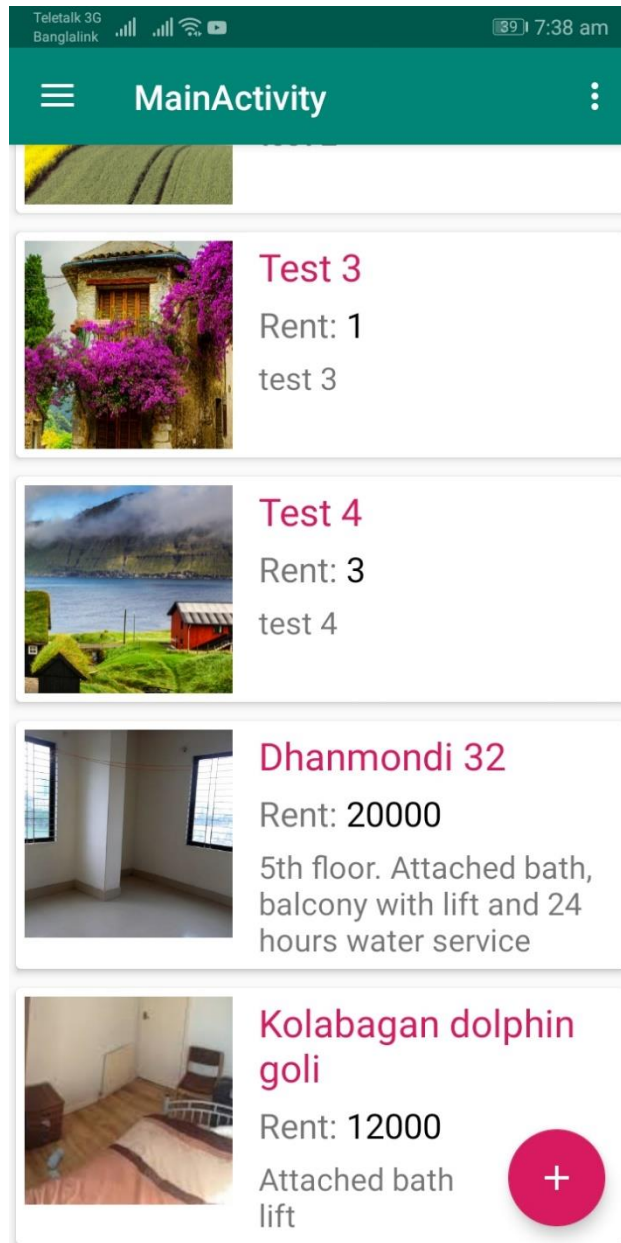


Figure 4.4.6: To-let List Activity

#### 4.4.7 Post Details Activity

In this page user can see all the details of a to-let.

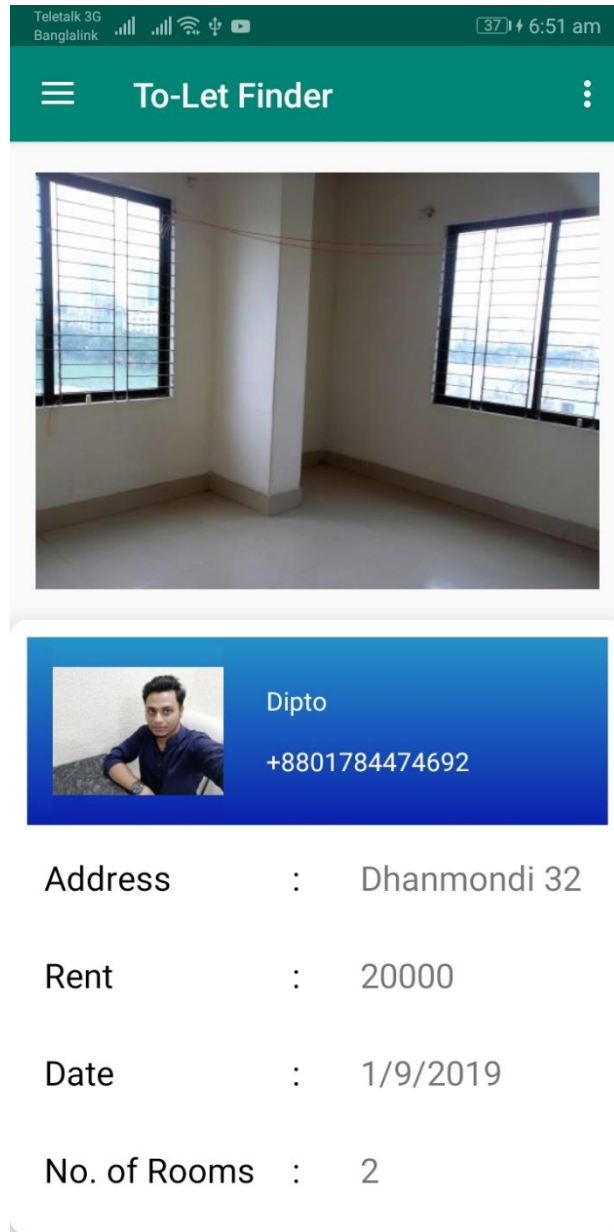


Figure 4.4.7: Post Details Activity

## 4.4.8 Navigation Drawer Activity

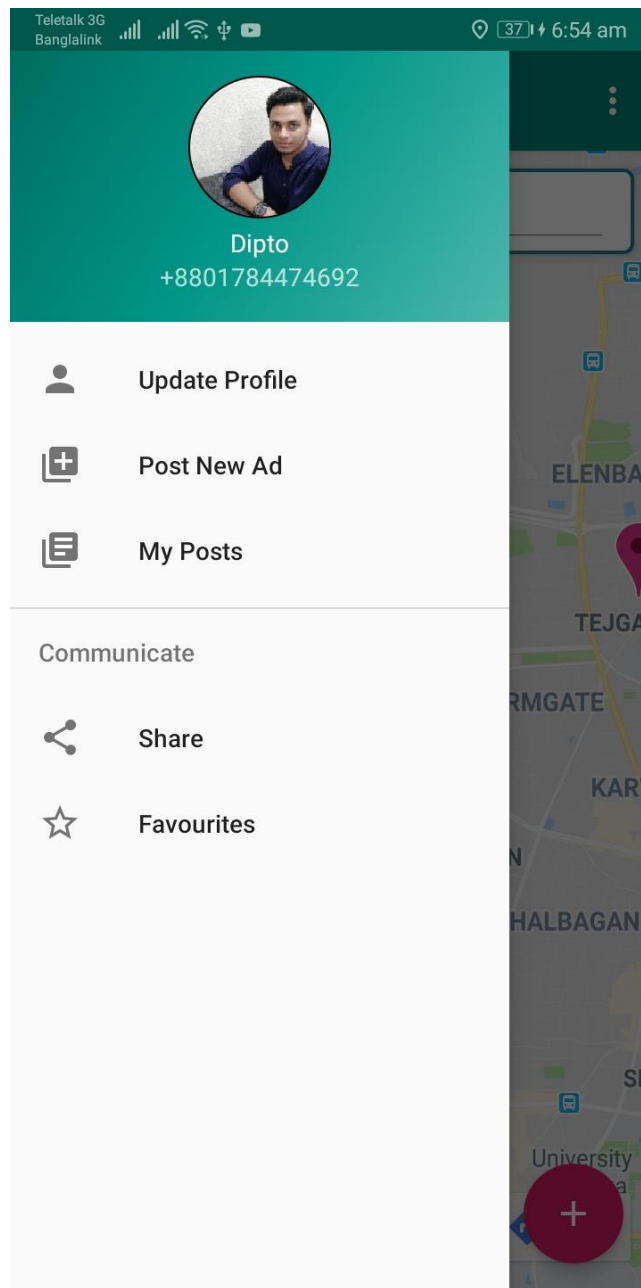


Figure 4.4.8: Navigation Drawer Activity

## **4.5 Implementation Tools**

We needed to use a lot of tools and components to develop our project and to build our app. We are going to discuss about all the necessary tools and components that use have used to build our project in this section.

### **4.5.1 Android Studio**

Android studio is the official development tool for Google's android operating system [6]. It is designed particularly for the development of android applications. So, we had to use Android Studio to build our application.

### **4.5.2 Android SDK**

Android SDK is the software development kit for android applications [7]. It provides a set of tools that the developers need to develop android applications. The tools that android SDK provides are following:

- Required libraries
- Debugger
- An emulator
- Relevant documentation for the Android application program interfaces (APIs)
- Sample source code
- Tutorials for the Android OS

### **4.5.3 JDK**

JDK means java development kit which is an environment for software development [8]. To develop java applications, developers need to use JDK. It includes the following components:

- Java Runtime Environment (JRE)
- Interpreter/loader (java)
- Compiler (javac)
- Archiver (jar)
- Documentation generator (javadoc)

### **4.5.4 Android Device**

An android device is one of the most important components for developing and android application. It is needed to run and test the application. Without testing the application on an actual device, we cannot be sure that the application is working correctly.

### **4.5.5 Permission**

For the development of our application we had to request for two permissions. One is permission for Internet access and other is permission for Location access.

For Internet access we had to add the following permission to our manifest file:

“android.permission.INTERNET”

For Location access we used the following permission:

“android.permission.ACCESS\_FINE\_LOCATION”

## **CHAPTER 5**

### **Testing and Result**

#### **5.1 Testing**

After successfully building our application it was time to implement and test the application and see the result. Since we have created the app, we were very familiar with the apps' user interface. So, it wouldn't be fair to test the app ourselves. That's why to test the app properly we ran a survey among our friends and class mates and some other people of different ages. After the survey as a result we saw that our app was very user friendly and everybody was being able to use its features at ease.

#### **5.2 Test Result**

After finishing the testing process, we finally came up with the following results:

- Good quality application.
- Innovative way of finding to-let.
- User friendly application.
- Easy to use.
- Small but useful application.

## **CHAPTER 6**

### **Discussion and Future Plan**

#### **6.1 Discussion**

After successfully finishing the development and testing of the project we can say that we have gained a lot of experiences. And these experiences have increased our confidence level. At first, we were afraid that what if we couldn't finish the project. We had doubts on ourselves about completing the whole project by ourselves. But now all the doubts and fears are gone and we have the confidence to make more better project than this in future.

#### **6.2 Limitations**

Though we have successfully built the application there still some limitations exist. Such as we couldn't provide any option to the user to communicate with the to-let owner through our app. Then there are many other small limitations which we will try overcome in future.

#### **6.3 Future Plan**

Making an application or any system absolutely perfect is impossible because nothing is absolutely perfect in the world. And every person has his/her own perspective which is different than others. So, one thing can be seemed perfect to me but not to others. But there is always scope to make things better. That's why we are determined to make improvement to our project, overcome all the limitations and fulfill all the needs of a user in future.

Some of the features that we want to include in future are:

- Search to-let by different categories.
- Filtering option for to-lets in the main activity.
- Automatically delete posts after a certain time, etc.



## **CHAPTER 7**

### **Conclusion**

#### **7.1 Conclusion**

After finishing the project, we are quite confident that we have achieved our goal. To-let finding is a day to day scenario in Dhaka city. It wastes a lot of time and causes a lot of sufferings to people specially to them who has jobs and to the females. Everybody wants an easy solution of this problem and to get rid of their sufferings. But there are no such good platforms available in our country. Our application can be a life saver in this critical situation. We are hopeful that our app will be able to release the users from their sufferings and also fulfill all of their needs at any time. When someone post an ad for to-let they get to choose the location from a google map. And after entering all the necessary data and save them the data will automatically be saved in firebase database. Then whenever a user enters our app after verification, he/she will get to see the to-lets on the google map around his/her current location. Then he/she can see all details of any to-let and have the contact no of the owner as well. So, at this point we think our project will be able to satisfy all the requirements of a user and fulfill its goal.

We had to face several obstacles during making of this project. But they couldn't stop us rather than made us more determined to finish the project at any cost. It made learn many things that we didn't know before and made us more confidence. Al last, we hope that those people who have to find new place for living but don't have that much time to search for to-lets throughout the city can be highly benefited by our app.

# APPENDIX

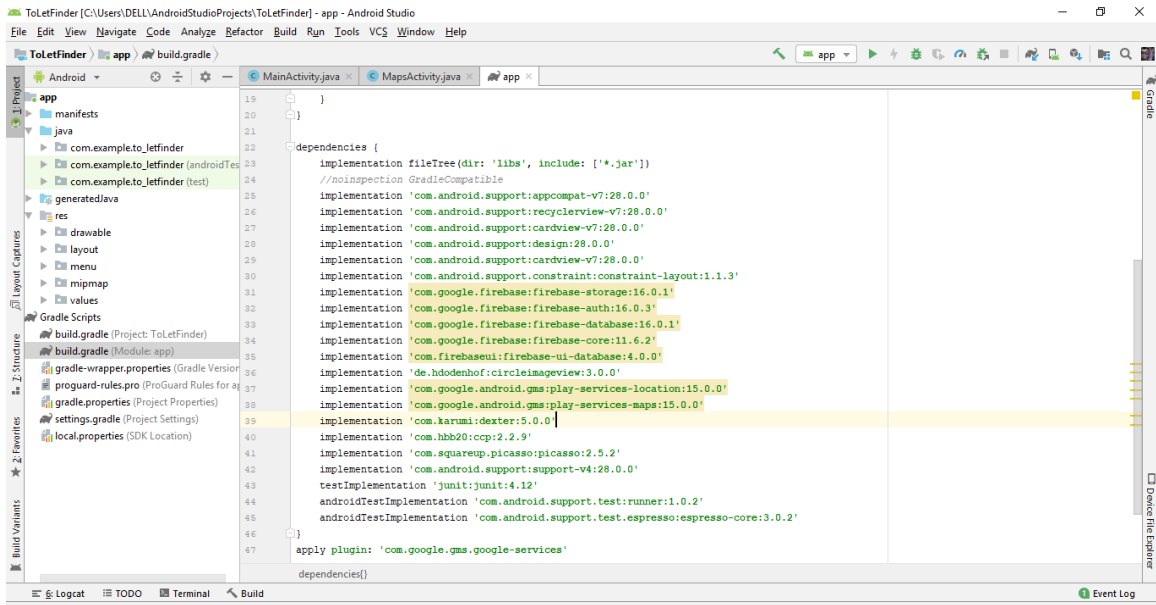


Figure A-1: App level gradle file

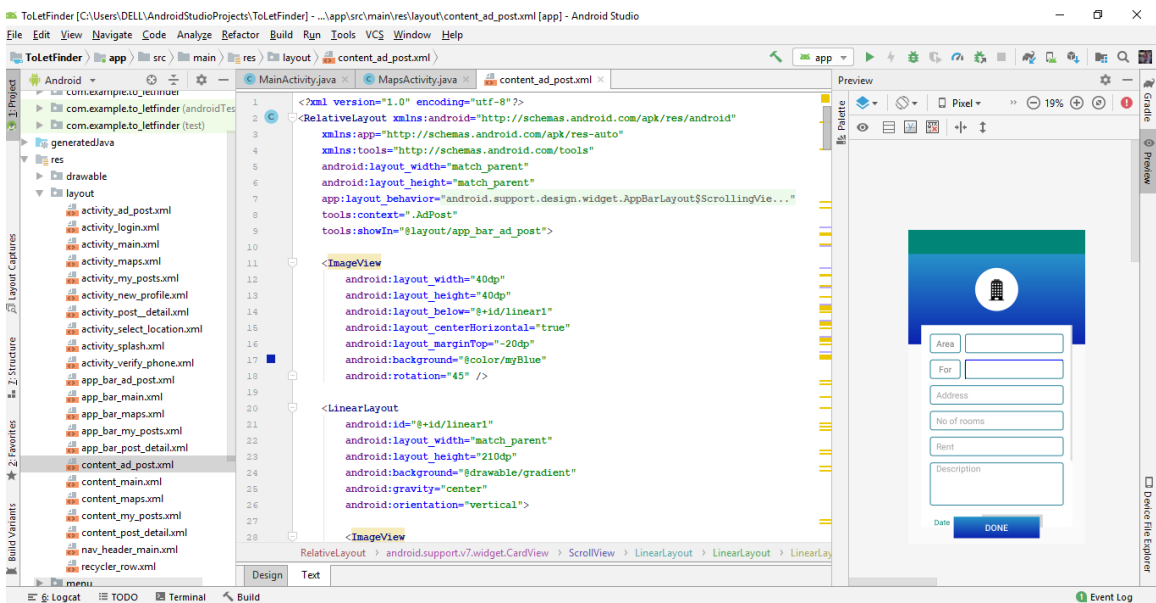


Figure A-2: XML code of Ad Post activity

```

1 package com.example.to_letfinder;
2
3 import ...
4
5 public class MainActivity extends AppCompatActivity
6     implements NavigationView.OnNavigationItemSelectedListener, View.OnClickListener {
7     private FloatingActionButton fab;
8     private RecyclerView mRecyclerView;
9     private ViewPagerAdapter viewPagerAdapter;
10    private DatabaseReference mRef;
11    private List<Data> dataSetList;
12    private ProgressBar progressBar;
13    private CircleImageView FP;
14    private TextView name, phoneNo;
15
16    @Override
17    protected void onCreate(Bundle savedInstanceState) {
18        super.onCreate(savedInstanceState);
19        setContentView(R.layout.activity_main);
20
21        progressBar = (ProgressBar) findViewById(R.id.probar_circle);
22        mRecyclerView = findViewById(R.id.recyclerview);
23        mRecyclerView.setHasFixedSize(true);
24        mRecyclerView.setLayoutManager(new LinearLayoutManager(this));
25        dataSetList = new ArrayList<>();
26        mRef = FirebaseDatabase.getInstance().getReference().child("Post Detail");
27        mRef.keepSynced(true);
28
29        mRef.addValueEventListener(new ValueEventListener() {
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000

```

Figure A-3: Java code of the Main activity

```

256 public void onDataChange(@NonNull DataSnapshot dataSnapshot) {
257     for (DataSnapshot shot : dataSnapshot.getChildren()) {
258         if (shot.child("latitude").getValue() != null && shot.child("longitude")
259             .getValue() != null && shot.child("category").getValue() != null) {
260             String lat = shot.child("latitude").getValue().toString();
261             String lng = shot.child("longitude").getValue().toString();
262             String ctg = shot.child("category").getValue().toString();
263             Double latitude = Double.parseDouble(lat);
264             Double longitude = Double.parseDouble(lng);
265             LatLng toLet = new LatLng(latitude, longitude);
266             if (text != null) {
267                 switch (text) {
268                     case "Family":
269                         switch (ctg) {
270                             case "Family":
271                                 MarkerOptions options = new MarkerOptions().position(toLet).title("To-let Here").snippet(ctg);
272                                 currentMarker = mMap.addMarker(options);
273                                 currentMarker.setIcon(BitmapDescriptorFactory.defaultMarker(BitmapDescriptorFactory.HUE_ROSE));
274                                 if (marker2 != null) {
275                                     marker2.remove();
276                                 }
277                                 marker2 = currentMarker;
278                                 break;
279                             case "Bachelor":
280                                 mMap.addMarker(new MarkerOptions().position(toLet).title("To-let Here").snippet(ctg).visible(false)
281                                     .setIcon(BitmapDescriptorFactory.defaultMarker(BitmapDescriptorFactory.HUE_VIOLET)));
282                                 break;
283                             default:
284                                 mMap.addMarker(new MarkerOptions().position(toLet).title("To-let Here").snippet(ctg).visible(false));
285                                 break;
286                         }
287                     }
288             }
289         }
290     }
291 }
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000

```

Figure A-4: Java code of the Map activity

The screenshot shows the Firebase Authentication console. The left sidebar contains navigation options: Project Overview, Develop (Authentication, Database, Storage, Hosting, Functions, ML Kit), Quality (Crashlytics, Performance, Test L...), Analytics (Dashboard, Events, Conversions,...), and Spark (Free \$0/month, Upgrade). The main content area is titled 'Authentication' and has tabs for Users, Sign-in method, Templates, and Usage. A search bar at the top allows searching by email address, phone number, or user UID. Below the search bar is a table with the following data:

Identifier	Providers	Created	Signed In	User UID ↑
+8801516726734	Phone	Jul 31, 2019	Jul 31, 2019	W9jfUblBjZ05n0VCjHgPVJcE9a2
+8801835222980	Phone	Jul 26, 2019	Aug 1, 2019	gGOPbGWBWVQXByMsFi2hztw...
+8801784474692	Phone	Aug 2, 2019	Aug 2, 2019	qrvtTwT07YY2AE1yFhvPZzqefM2

At the bottom of the table, it indicates 'Rows per page: 50' and '1-3 of 3'.

Figure A-5: Firebase authentication table

The screenshot shows the Firebase Realtime Database console. The left sidebar is similar to the previous image, but the 'Database' tab is selected. The main content area is titled 'Database' and has tabs for Data, Rules, Backups, and Usage. The URL bar shows 'https://to-let-finder.firebaseio.com/'. The data is displayed as a JSON object:

```

to-let-finder
├── Data
│   └── Post Detail
│       └── -LkujVSn0EDqzK9Joy0o
│           ├── daddress: "Purbo Rajabazar"
│           ├── darea: "Dhanmondi"
│           ├── ddescription: "....."
│           ├── deadline: "1/7/2019"
│           ├── dnoofrooms: "3"
│           ├── drent: "18000"
│           ├── imageUrl: "https://firebasestorage.googleapis.com/v0/b/to-..
│           ├── phoneNo: "+8801835222980
│           ├── ppUrl: "https://firebasestorage.googleapis.com/v0/b/to-..
│           └── uid: "gGOPbGWBWVQXByMsFi2hztwW..."

```

Figure A-6: Firebase Realtime database table

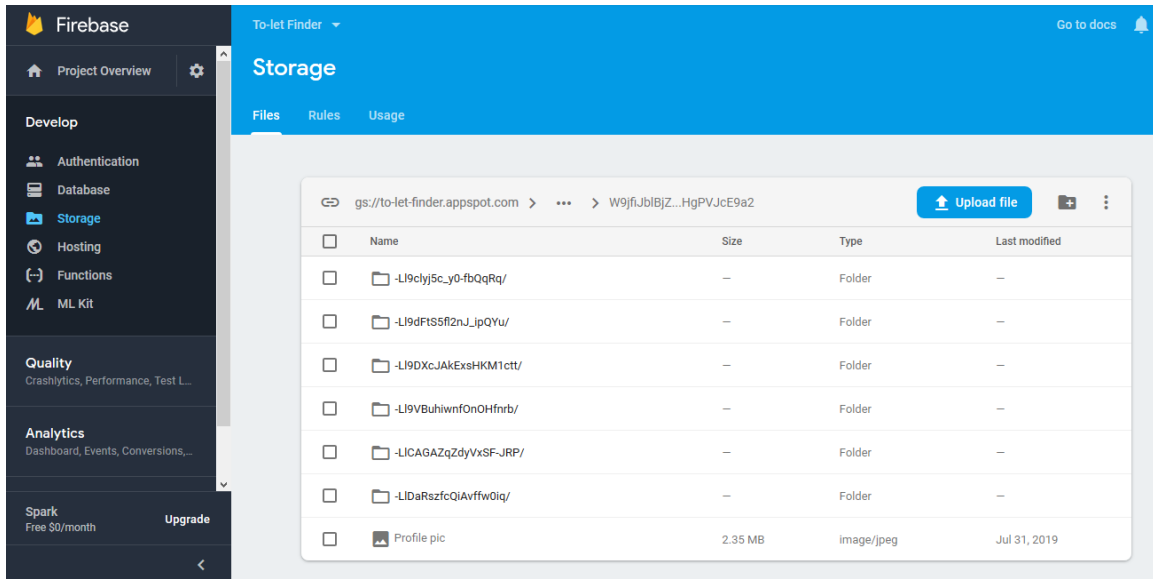


Figure A-6: Firebase storage table

## REFERENCES

[1] Android Studio tutorial on firebase application, available at:

<https://www.youtube.com/watch?v=JVokoelQ1RI&list=PLrnPJCHvNZuBf5KH4XXOthtgo6E4Eplj8>

[Last accessed: July 28, 2019]

[2] Android Studio tutorial on simple login application, available at:

[https://www.youtube.com/watch?v=IF5m4o\\_CuNg&list=PLbte\\_tgDKVWQOCRIzkgEQ8umdn\\_S6ZnHr](https://www.youtube.com/watch?v=IF5m4o_CuNg&list=PLbte_tgDKVWQOCRIzkgEQ8umdn_S6ZnHr)

[Last accessed: July 22, 2019]

[3] Database and storage, available at

<https://console.firebase.google.com/project/to-let-finder/database/to-let-finder/data>

[Last accessed: August 4, 2019]

[4] Google APIs and Services, available at:

<https://console.developers.google.com/projectselector2/apis/dashboard?pli=1&supportedpurview=project&project&folder&organizationId> [Last accessed: July 1, 2019]

[5] Google Cloud Platform, available at:

<https://console.cloud.google.com/getting-started?pli=1>

[Last accessed: August 1, 2019]

[6] Detail information about android studio, available at:

[https://en.wikipedia.org/wiki/Android\\_Studio](https://en.wikipedia.org/wiki/Android_Studio)

[Last accessed: August 2, 2019]

[7] Definition and details about android SDK, available at:

<https://www.techopedia.com/definition/4220/android-sdk>

[Last accessed: August 2, 2019]

[8] Definition and details about JDK, available at:

<https://www.techopedia.com/definition/5594/java-development-kit-jdk> [Last accessed: August 2, 2019]