

# **ANDROID BASED MEDICATION**

**BY**

**KHALIDA TASNEEM**

**ID: 191-15-12437**

**AND**

**Tanni Rani Saha**

**ID: 191-15-12475**

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

Supervised By

**Dr. Moushumi Zaman Bonny**

Assistant Professor

Department of CSE

Daffodil International University

Co-Supervised By

**Dr. Sheak Rashed Haider Noori**

Professor

Department of CSE

Daffodil International University



**DAFFODIL INTERNATIONAL UNIVERSITY**

**DHAKA, BANGLADESH**

**JANUARY 2023**

# APPROVAL

## APPROVAL

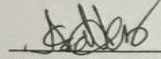
The Research Based Project Title is 'Android-based Medication', submitted by KHALIDA TASNEEM, ID No:191-15-12437 And Tanni Rani Saha, ID No:191-15-12475 to the Department of Computer Science and Engineering.

## BOARD OF EXAMINERS

**Dr. Touhid Bhuiyan**  
**Professor and Head**

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

**Chairman**

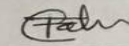


**Abdus Sattar**

**Assistant Professor**

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

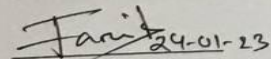
**Internal Examiner**



**Fatema Tuj Johra**  
**Senior Lecturer**

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

**Internal Examiner**



**Dr. Dewan Md Farid**  
**Professor**

Department of Computer Science and Engineering  
United International University

**External Examiner**

©Daffodil International University

i

# DECLARATION

©Daffodil International University

i

## DECLARATION

We hereby declare that this project has been done by us under the supervision of **Dr. Moushumi Zaman Bonny Assistant Professor**, Department of CSE Daffodil International University and co-supervision **Dr. Sheak Rashed Hider Noori** Department of CSE Daffodil International University. We also declare that neither this Research Based Project nor any part of this project has been submitted elsewhere for award of any degree or diploma.

### Supervised by

Mus 24.01.23

**Dr. Moushumi Zaman Bonny**  
**Assistant Professor**  
Department of Computer Science and Engineering  
Faculty of Science & Information Technology  
Daffodil International University

### Submitted by

Khalida

**KHALIDA TASNEEM**  
ID:191-15-12437  
Department of Computer Science and Engineering  
Daffodil International University

Tanni

**Tanni Rani Saha**  
ID:191-15-12475  
Department of Computer Science and Engineering  
Daffodil International University

## ACKNOWLEDGEMENT

First we express our heartiest thanks and gratefulness to almighty Allah for His divine blessing makes us possible to complete the final year research based project successfully. We really grateful and wish our profound our indebtedness to Supervisor **Dr. Moushumi Zaman Bonny Assistant Professor**, Department of CSE Daffodil International University, Dhaka. Deep Knowledge & keen interest of our supervisor in the field of “web development, Database” to carry out this paper. Her endless patience, scholarly guidance, continual encouragement, constant and energetic supervision, constructive criticism, valuable advice, reading many inferior draft and correcting them at all stage have made it possible to complete this project.

We would like to express our heartiest gratitude to Head of Department of CSE, for his kind help to finish our research based 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.

## TABLE OF CONTENTS

CONTENTS	PAGE
Approval	i
Declaration	ii
Acknowledgement	iii
Abstract	iv
<b>Chapter 1: Introduction</b>	<b>1-2</b>
1.1 Introduction	1
1.2 Motivation	1,2
1.3 Objective	2
1.4 Expected Output	2
<b>Chapter 2: Background</b>	<b>3-4</b>
2.1 Introduction	3
2.2 Related Works	3
2.3 Comparative Studies	3,4
2.4 Scope of the Problem	4
2.5 Challenges	4
<b>Chapter 3: Requirement Specification</b>	<b>5-8</b>
3.1 Business Process Modeling	5
3.2 Requirement collection and analysis	6
3.3 Use case modeling and description	7
3.4 Logical data modeling	7
3.5 Design Requirement	8

<b>Chapter 4: Design Specification</b>	<b>9-15</b>
4.1 Front-End Design	9
4.2 Interaction design and UX	12
4.3 Back End Design	12
4.4 Implementation requirement	15
<b>Chapter 5: Implementation</b>	<b>16-20</b>
5.1 Implementation of database	16
5.2 Implementation of Front End Design	17
5.3 Implementation of Interaction	18
5.4 Testing and Implementation	19
5.5 Test Results and Reports	20
<b>Chapter 6: Conclusion and Future Scope</b>	<b>21</b>
6.1 Development and Future Scope	21
6.2 Conclusion and Discussion	21
<b>References</b>	<b>21-22</b>
<b>Appendix</b>	<b>23-46</b>
<b>Figures</b>	
3.1 Business process model of Admin dashboard	5

3.2 Chart by communication with patient	7
3.3 Chart by communication with patient	7
4.1 (a),(b)Admin dashboard of complete medical care system	10
4.2 Interaction design and UX for android application	11
4.3 Back end of complete medical care web dashboard	12
4.4 Back end of complete medical care web dashboard	13
4.5 Back end of complete medical care web dashboard	13
4.6 Back end of complete medical care web dashboard	14
4.7 Back end of complete medical care web dashboard	14
4.8 INSERT INTO database using MySQL	15
5.1 Database structure of complete medical care system	16
5.2 Database structure of complete medical care system	17
5.3 Black diagram of complete medical care web development	17
5.4 Table structure using MySQL complete medical care system	18
5.5 Table structure using MySQL complete medical care system	18
5.6 Testing and Implementation	19
5.7 Test result and report	20
5.8 Test result and report	20

## **ABSTRACT**

Our Research based project is about ANDROID BASED MEDICATION. This research is up to a mobile application. Through this paper people can know about any kind of medicine information or medicine details and Doctors information.

Through this web based mobile application people get huge information about medicine, medical and Doctors. This Proposed system will save time and super supportive. By this android system people can know about any kind of medicine information and there also give doctors information and medical information. We are know that Our Country Bangladesh is a populated Country and Our Capital city Dhaka is always in too much Traffic jam that time people cannot reach for appointment or cannot get to know doctors information in time. The in-charge of hospital are busy sometimes. Because of that people are not get the hospital and Doctors information in emergency or exact time. In this situation people cannot get medical support too easily. Our system will help to know any kind of medical care and get doctors information easily just use the system Which available on smart phone.

## **Chapter 1**



# **Introduction**

## **1.1 Introduction**

In our daily life we are face to many kind of problem medical problem is one of them and it is a biggest problem for all of us .This system reduce the problem of people when they try to find a solution of medical problem. The field of medical application (apps) currently hold potential of improve the daily medical practice by implementing evidence-based medical tools.

This system have an admin panel for help to people 24 Hours. This proposal system contains an Admin panel from where each and every position of this application will be monitored. An admin can create, change, modify or delete any offensive, article or profile from this Database.

This application have an integrated system where medicine, medical and doctor information will be together.

As medical apps are increasingly used to support diagnosis and management of diseases apps that allow the user to input patient specific information along with reference material to automatically diagnose a disease or condition, the appropriate use of their reliable and valid medical information becomes crucial.

In our developing and technology dependent life we totally rely on gadgets especially smart phones. Probably everyone use smartphone. With this we get an opportunity to use technology in a better way so that it can be made useful for all of us and it plays an important part in our daily life and helps us staying fit in many ways.

## **1.2 Motivation**

In our Country When we are getting sick we are not get a medical support easily and it is so painful for all of us. By this System we can get medical help easily. We will had a helpline where a helper always to ready for help.

1

Younger to older people can call anytime and get help. A Ten years old Child also can use this application in emergency time for medical help.

©Daffodil International University

Apps have huge potential to improve patient practice, system efficiency and communication by providing a quick reference tool accessible at the point of care.

There are many apps for medical services but none are integrated like this and none of them have free features like drug details and doctor details. So this project is small but effective step for mankind as it reduces all required time and effort. This application has such skills that it will cover most of the people as users. Our main motivation is make easier and supportive for people to found medicine details and doctors information.

### **1.3 Objective**

- To make a user friendly application.
- To reduce people's time.
- To reduce people's efforts.
- To create a chance for making mankind life better.
- To ensure people get the maximum medical support.

### **1.4 Expected Output**

The main outcome of the project is to reduce people's difficulty in getting medical help doctor appointments, drug details these services are at one place. People can save their precious time and money through this application. In case of emergency they can get the doctor's details will be given in the program. Our main aim is to reduce all barriers and make people's lives easier.

## **Chapter 2**

# BACKGROUND

## 2.1 Introduction

The focal point of this chapter is to study the related work done in this domain before and to find the scope of our research and the challenges of the research.

## 2.2 Related works

Complete Medical care is a combine system for getting medical help shortly and without any kind of trouble and time killing process. This project is only applicable for Dhaka city. Complete Medical Care is a unique application from other Bangladeshi other mobile applications. There are different types of application in google play store which provide medical services with different feature.

The first one is 'Dy-named' shows the way to get medicine details but their main problem is all the process is manual and they have to input all the medicine manually. Their Database work is manual and not only that they have some limited potion of option. Dy-named works on both Android and iOS but requires a subscription plan to access.

On the other hand next application help you to find the details of medicine you have. It is really helpful but they have data limit. Means they have to input all the details about the medicine manually and they can't cover all the medicine running in this country and basically this application is fully depended on admin.

## 2.3 Comparative Studies

Medical problems are a widespread problem in this capital of Bangladesh. This problem causes many unnecessary deaths in this city. There are some mobile applications to provide individual solutions or problems but cannot work for multiple problems at the

same time. Another problem in Dhaka city is traffic congestion and lack of knowledge of people so that ambulance services, blood donors, timely diagnostic centers and medicines are not available.

## **2.4 Scope of the Problem**

Google Play Store already has many medical related apps but they are surrounded by a border. Each application is working separately so the user needs to install about six applications to get all kinds of medical help. It kills the user's phone space and time. On the other hand, complete medical care as a combination of all the possibilities together provides everything needed in just one touch. It reduces user time, effort and as a user friendly application it will include all types of users where age doesn't matter. So it will be a great opportunity for us to be one step ahead of others through this thought and action. Other limitations are opportunities to improve our application and differentiate it from others. This way we will get maximum user support and it will help us stay on top.

## **2.5 Challenges**

During the development of this project, we faced many problems and challenges. They were collecting the data for the primary, designing the application, designing the database and connecting the database to our applications. There are two parts in our application. The application are admin and user. The admin panel is web based and our main challenge is to connect the application with the admin panel. Admin has full control over application and database. It was difficult to connect to the admin panel and the application.

# Chapter 3

## Requirement Specification

### 3.1 Business process modeling

Business process model represents the system of working of the entry system. This process represents the concept level of a process. This following model the process of our system is indicate step by step. Used business process models in the system are given below.

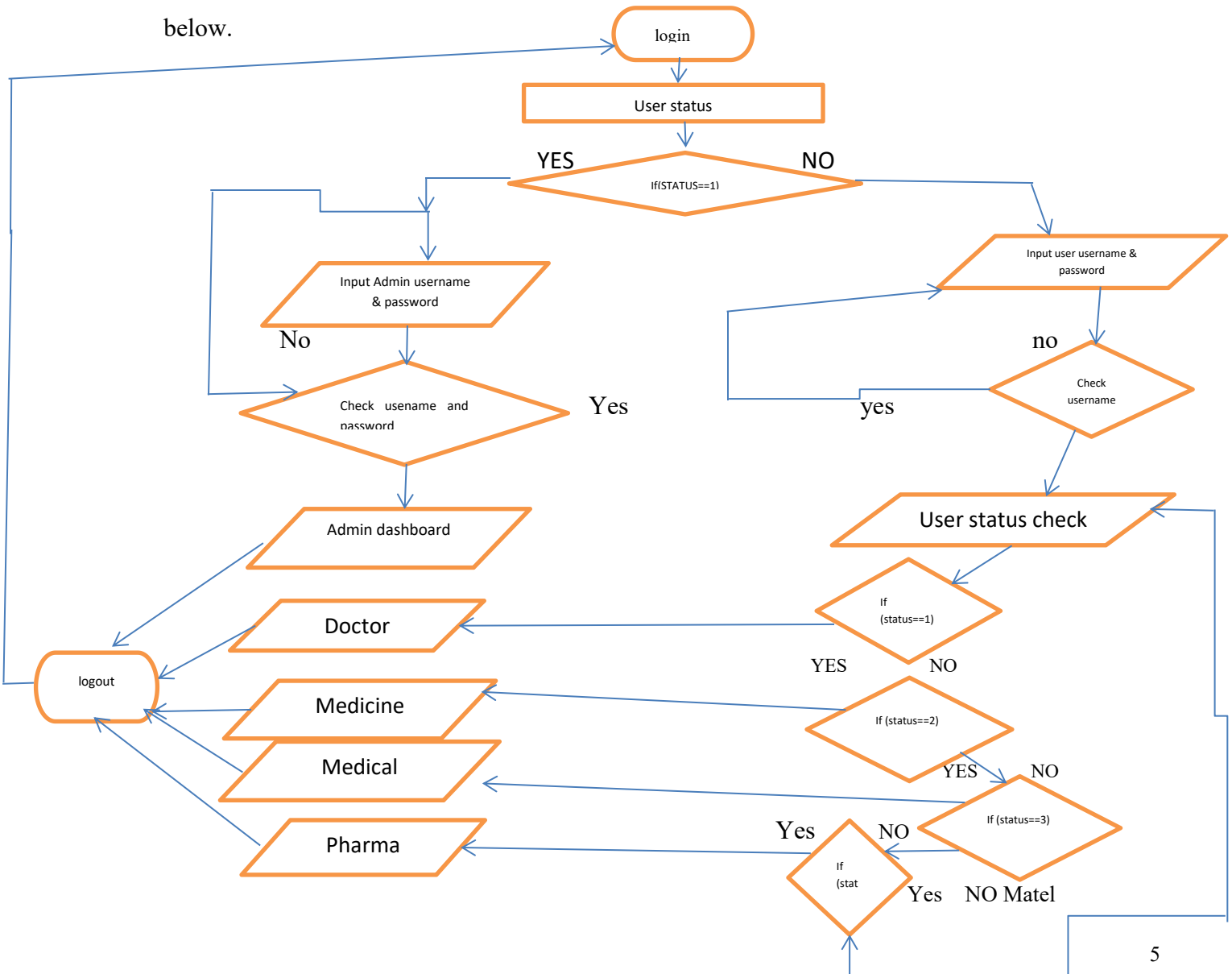


Figure3.1: Business process model of Admin dashboard

## 3.2 Requirement Collection and Analysis

### Requirement for Admin and dashboard

- Admin, Doctor, Medicine and Medical.
- Admin can view all web dashboard user.
- Admin can delete, block and approve dashboard user.
- Time convenience.
- User friendly.
- Easily accessible.

### Requirement for Android app

- Login system for app user.
- User can easily access all information.
- App running smoothly.
- Time convenience.
- User friendly.

**Communication Requirement:** In this system we provided E-mail, web browser and communication interfaces. We can use Communication models such as HTTP of FTP to give security.

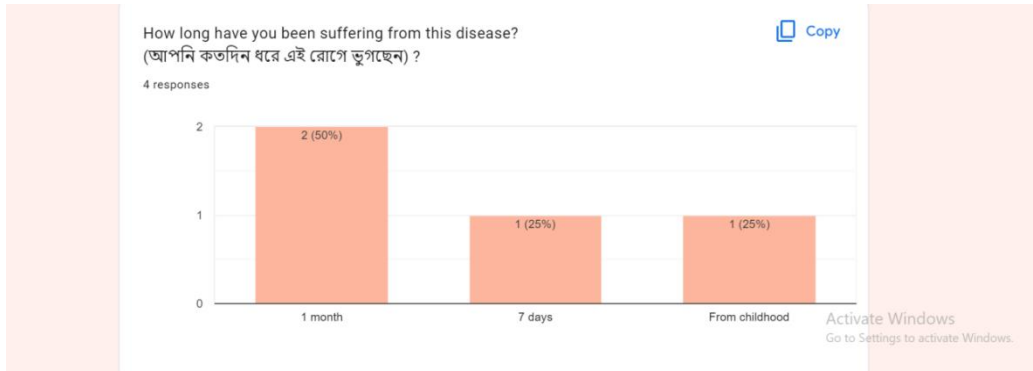


Figure 3.2: Chart by Communication with patient

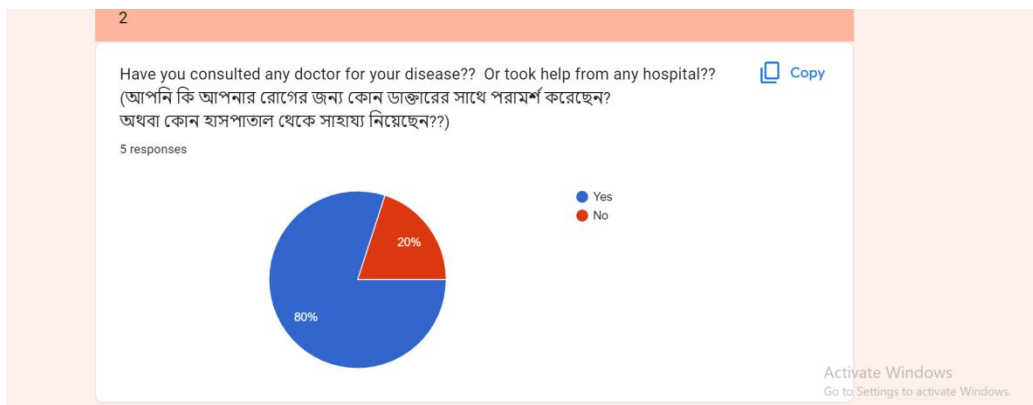


Figure 3.3: Chart by Communication with patient

### 3.3 Use Case Modeling and Description

Use case model represents various types of users and their activities of a system. This model is indicating the activities of Admin, Doctor, Medicine, Medical and the app users.

### 3.4 Logical Data Modeling

Logical data modeling is used to identify and analyze the requirements for processing by the organization's associated information systems. The main parts of an entity-relationship diagram are entities, which are concepts that can store data about them, relationships between those entities, and cardinality, which defines that relationship in terms of numbers.

### **3.5 Design Requirement**

Our system will make the task unique with the design requirements. If we want to get the actual output in a difficult time, we have created the database carefully and designed in an elegant way. There are different types of user panels (admin, user). All data requirements are arranged in different form. We have various categories for data storage so that the organization can easily benefit from this system. These system agents are able to have a great deal of control over their objectives in completing the objectives. Android app is made with unique design. In the app we have used material design for a better look.

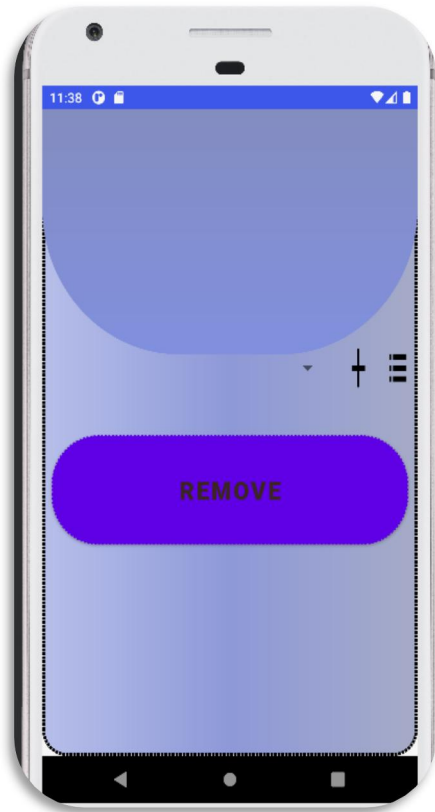


## Chapter 4

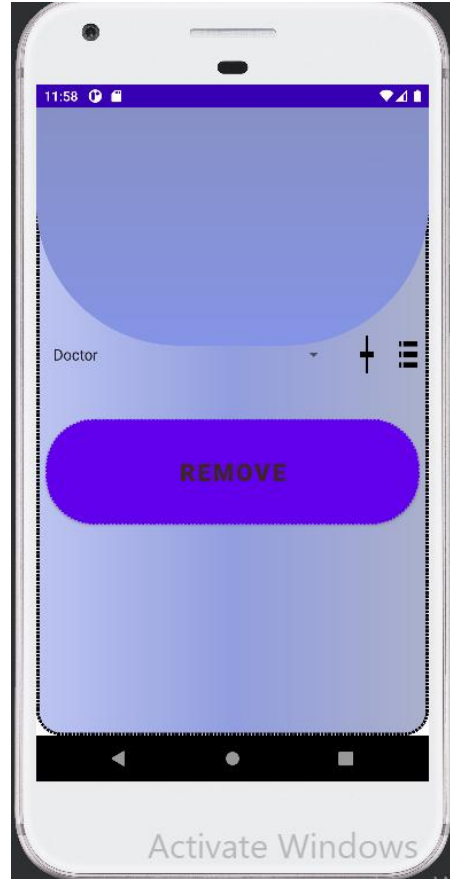
### Design Specification

#### 4.1 Front-End Design

Front end design in our project consists part: web design for web dashboard android application design part. Front-end design of web dashboard includes CSS, jQuery and JavaScript that make up a user interface and the design part of android application XML and material design. Coding patterns are placing a set of connections. By codifying these connections as a set of policies instead of a single, stand-alone. We make it feasible for other developers to extent our work while remaining faithful to these ideas. This is visible while using a website is a compound of JavaScript managed by a browser. Here added things like fonts, home, menus bar, buttons, sliders forms of different types like login, logout, add delete etc. And in android application we use different type of material design like card View, Recycle View, Navigation Drawer, different button, alert dialog, Progress Bar, Edit Text, Text View, auto Complete Text View etc. We have used the XML, JavaScript to design the system. Present the total process that we have used in our project.

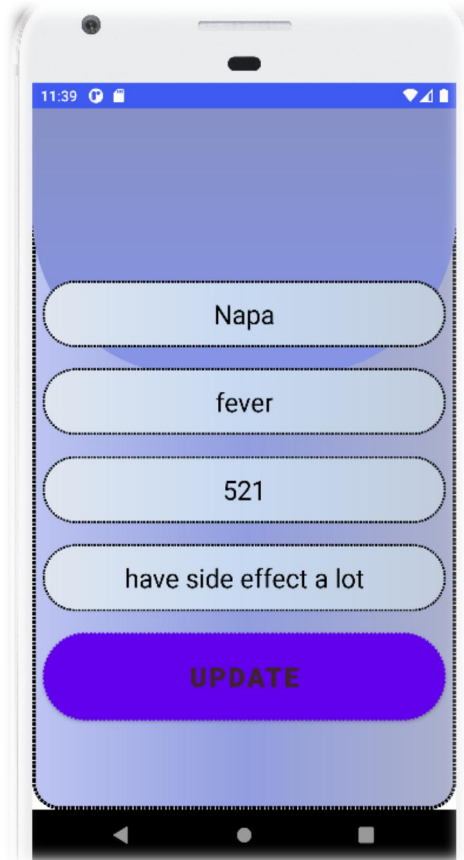


(a)



(b)

Figure 4.1: (a),(b) Admin dashboard of Complete Medical Care System



%%

Figure 4.2: Admin dashboard of Complete Medical Care System

### **Java-script**

HTML content can be changed by JavaScript. There are many JavaScript HTML methods, one of them is `getElementById()`. This example applies the method to find an HTML element and changes the element content to 'Hello JavaScript'.

### **CSS**

CSS is used to style our web pages, design and display size defined by device. By using CSS we can add lots of designs, colors, text-alignment, text-styles and everything that would make our system more charming.

## XML

XML is used for layout designing. All the UI and layout of our app is designed using XML. XML helps you to design our app, how it will look, how components like buttons, text-view will placed and their styling.

## 4.2 Interaction Design and UX

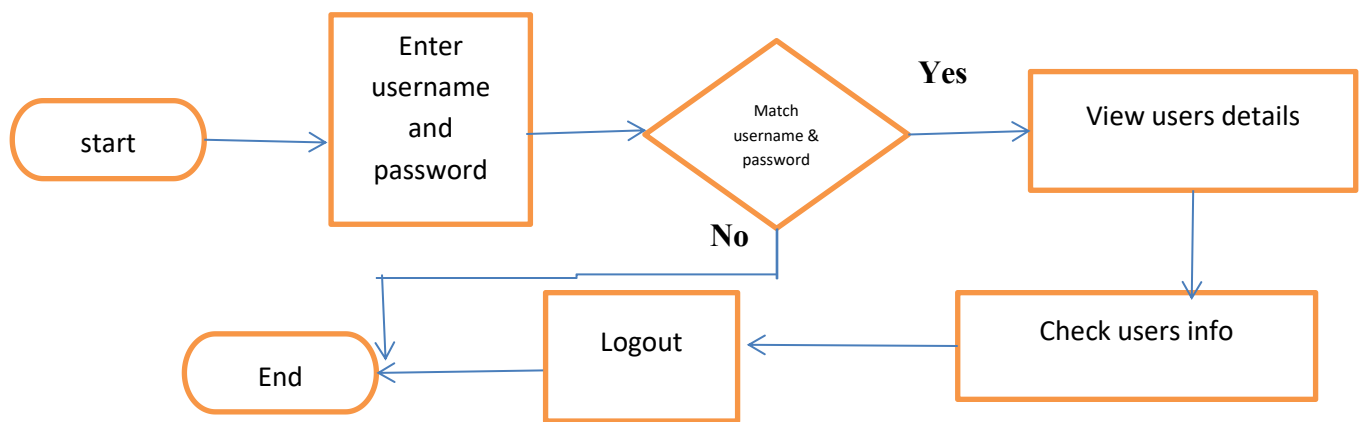


Figure 4.3: Interaction design and UX for android application

## 4.3 Back -End Design

To develop our system, we used some special virtual ambiance to make our task easier. Implementing the project to use in actual time execution it needs a server in the system then the system will be installed. Server provided by XAMPP virtual server system is used for developing purposes. Java and MySQL support us to complete our project by using as a server. We need an android mobile phone to run our application smoothly.

```
54
55
56 @Override
57 protected void onCreate(Bundle savedInstanceState) {
58     super.onCreate(savedInstanceState);
59     setContentView(R.layout.activity_add_medicin);
60
61     Intent intent = getIntent();
62     task = intent.getStringExtra("task");
63
64     if(task.equals("update")) {
65
66         medicine1 = (Medicine) intent.getSerializableExtra("object");
67     }
68
69     databaseReference = FirebaseDatabase.getInstance().getReference("path: "Medicine");
70
71     storage = FirebaseStorage.getInstance();
72     storageReference = storage.getReference("location: "Medicine");
73
74     name = findViewById(R.id.name);
75     quantity = findViewById(R.id.quantity);
76     description = findViewById(R.id.des);
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 4.4: Back end code of Complete Medical Care web dashboard

```
ava
values.xml
AndroidManifest.xml
description = findViewById(R.id.des);
diseases = findViewById(R.id.diseases);

circleImageView = findViewById(R.id.medicin_image);

if(task.equals("update")) {

    name.setText(medicine1.Name);
    quantity.setText(medicine1.Quantity + "");
    description.setText(medicine1.Description);
    diseases.setText(medicine1.Diseases);

    Picasso.get().load(Uri.parse(medicine1.medicine_image)).fit().into(circleImage

    Uri share = Uri.parse(medicine1.medicine_image.trim());

    uri = share;

}

enter = findViewById(R.id.enter);

enter.setOnClickListener(this);

if(task.equals("update")) {

Activate Windows
Go to Settings to activate Windows.
Event Log
Layout Inspector
62:46 LF UTF-8 4 spaces
```

Figure 4.5: Back end code of Complete Medical Care web dashboard

```
108 circleImageView.setOnClickListener(this);
109
110 }
111
112 @Override
113 public void onClick(View v) {
114
115     if(v.getId() == enter.getId()) {
116
117         name.setOnKeyListener(new View.OnKeyListener() {
118             @Override
119             public boolean onKey(View v, int keyCode, KeyEvent event) {
120
121                 if(keyCode == KeyEvent.KEYCODE_ENTER && event.getAction() == KeyEvent.A
122
123                     Name = name.getText().toString().trim();
124
125                     return true;
126
127                 }
128
129                 return false;
130
131             });
132
133 }
```

Figure 4.6: Back end code of Complete Medical Care web dashboard

```
135 public boolean onKey(View v, int keyCode, KeyEvent event) {
136
137     if(keyCode == KeyEvent.KEYCODE_ENTER && event.getAction() == KeyEvent.A
138
139         Quantity = Integer.parseInt(quantity.getText().toString().trim());
140
141         return true;
142
143     }
144
145     return false;
146
147 }
148
149 diseases.setOnKeyListener(new View.OnKeyListener() {
150     @Override
151     public boolean onKey(View v, int keyCode, KeyEvent event) {
152
153         if(keyCode == KeyEvent.KEYCODE_ENTER && event.getAction() == KeyEvent.A
154
155             Diseases = diseases.getText().toString().trim();
156
157             return true;
158
159         }
160
161     }
162 }
```

Figure 4.7: Back end code of Complete Medical Care web dashboard

## 4.4 Implementation Requirement

To develop our system, we used some special virtual ambiance to make our task easier. Implementing the project to use in actual time execution it needs a server in system then the system will be installed. Server provided by XAMPP virtual server system is used for developing purposes. Java and MySQL support us to complete our project by using it as a server. We need an android mobile phone to run our application smoothly.

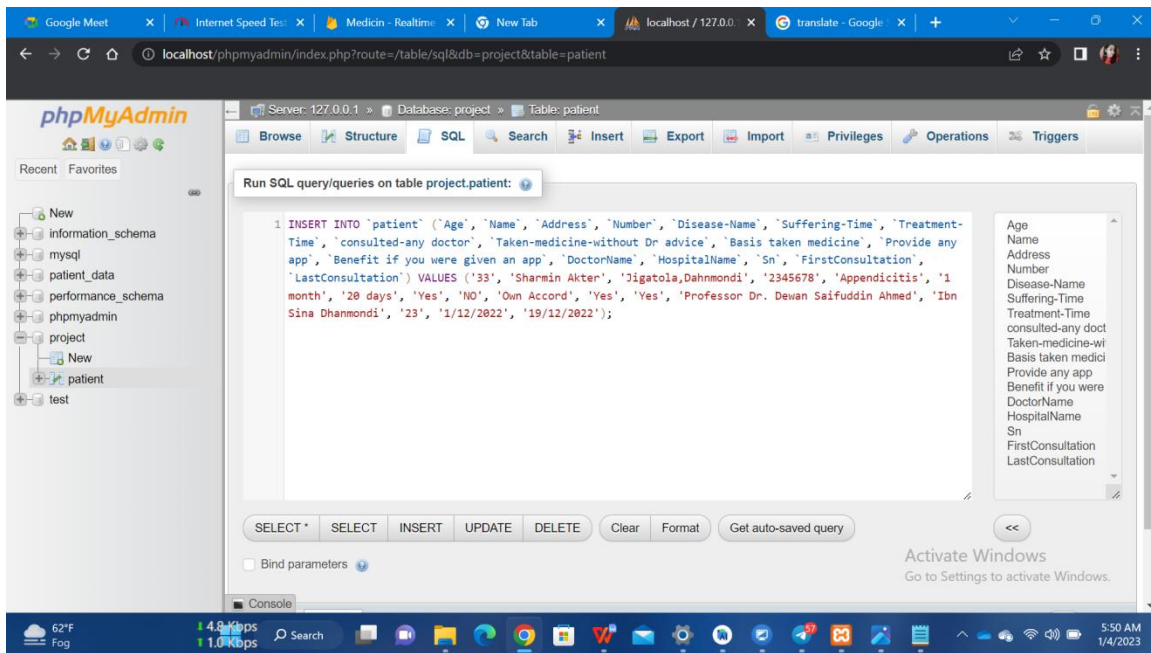


Figure 4.8: INSERT INTO database using MySQL

# Chapter 5

## Implementation

### 5.1 Implementation of database

In our system maintenance Database is the central part. Therefore, worldwide many database management systems are being used. We chose MySQL for our project. MySQL is an open source and popular database worldwide. It has become the first option for web-based applications with its safety, easy to use and performance. MySQL is the most approved for web based systems because of its open source relational database system.

Here is our database which we used as our system.

17 Tables in our database.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/>	1 Age	int(11)			Yes	NULL			Change  Drop Mc
<input type="checkbox"/>	2 Name	varchar(255)	utf8mb4_general_ci		Yes	NULL			Change  Drop Mc
<input type="checkbox"/>	3 Address	varchar(255)	utf8mb4_general_ci		Yes	NULL			Change  Drop Mc
<input type="checkbox"/>	4 Number	int(200)			Yes	NULL			Change  Drop Mc
<input type="checkbox"/>	5 Disease-Name	varchar(255)	utf8mb4_general_ci		Yes	NULL			Change  Drop Mc
<input type="checkbox"/>	6 Suffering-Time	varchar(255)	utf8mb4_general_ci		Yes	NULL			Change  Drop Mc
<input type="checkbox"/>	7 Treatment-Time	varchar(255)	utf8mb4_general_ci		Yes	NULL			Change  Drop Mc
<input type="checkbox"/>	8 consulted-any doctor	varchar(255)	utf8mb4_general_ci		Yes	NULL			Change  Drop Mc
<input type="checkbox"/>	9 Taken-medicine-without Dr advice	varchar(255)	utf8mb4_general_ci		Yes	NULL			Change  Drop Mc

Figure 5.1: Database structure of complete Medical Care System



	ID	Field Name	Field Type	Collation	Nullable	Index	Actions
<input type="checkbox"/>	9	Taken-medicine-without Dr advice	varchar(255)	utf8mb4_general_ci	Yes	NULL	Change Drop
<input type="checkbox"/>	10	Basis taken medicine	varchar(255)	utf8mb4_general_ci	Yes	NULL	Change Drop
<input type="checkbox"/>	11	Provide any app	varchar(255)	utf8mb4_general_ci	Yes	NULL	Change Drop
<input type="checkbox"/>	12	Benefit if you were given an app	varchar(255)	utf8mb4_general_ci	Yes	NULL	Change Drop
<input type="checkbox"/>	13	DoctorName	varchar(255)	utf8mb4_general_ci	Yes	NULL	Change Drop
<input type="checkbox"/>	14	HospitalName	varchar(255)	utf8mb4_general_ci	Yes	NULL	Change Drop
<input type="checkbox"/>	15	Sn	int(11)		Yes	NULL	Change Drop
<input type="checkbox"/>	16	FirstConsultation	text	utf8mb4_general_ci	Yes	NULL	Change Drop
<input type="checkbox"/>	17	LastConsultation	text	utf8mb4_general_ci	Yes	NULL	Change Drop

Figure 5.2: Database structure of complete Medical Care System

## 5.2 Implementation of Front -End Design

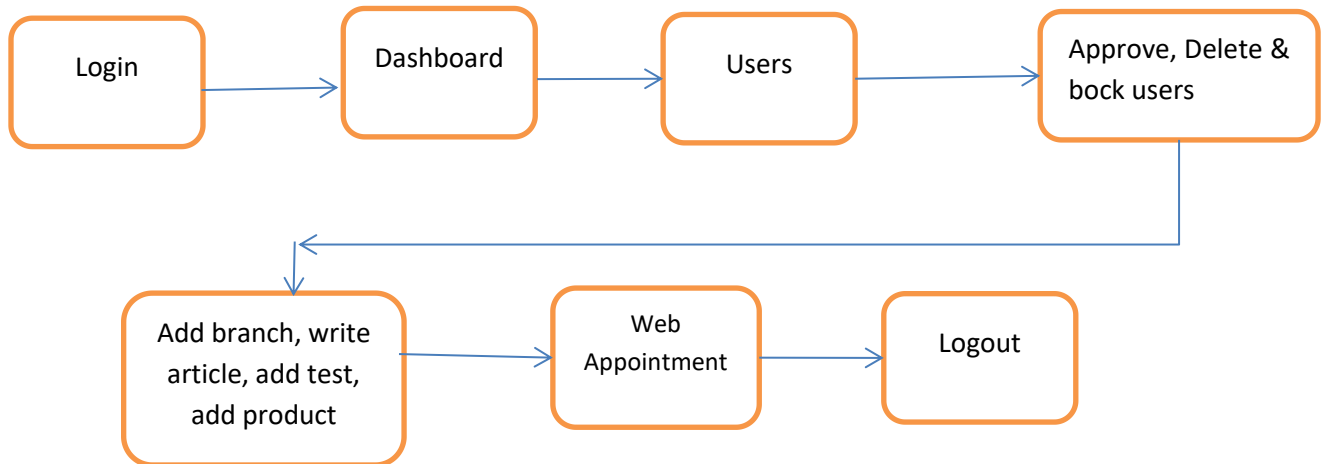


Figure 5.3: Block Diagram of complete medical care web development

### 5.3 Implementation of Interaction

For developing a project must need to Implement of Interaction. Our project is a web based Android application in this case many resources and user activity need to interact. MySQL is used as a database. Most tasks of the system are related to data. So each time connection with the server for better Performance of Organization.

Age	Name	Address	Number	Disease-Name	Suffering-Time	Treatment-Time	consulted-any doctor	Taken-medicine-without Dr advice	Basis taken medicine	Provide any app	Benefit if you were given an app	DoctorName
22	Tanni Rani Saha	Dhanmondi	1764443781	fever	10 days	3 days	No	yes	other reference	yes	yes	Najrul Isla
22	Tanni Rani Saha	Dhanmondi	1764443781	fever	10 days	3 days	No	yes	other reference	yes	yes	Najrul Isla
18	Fahim	Mirpur	97887654	Cancer	2 Months	2 Months	Yes	No	NULL	Yes	Yes	Prof. Dr. Rehana Begum
	Safa	Mohammadpur,Tajmohol road	98765433	Kidney Failure	6 months	3 Months	Yes	Yes	Other Reference	No	Yes	Dr. Saraw Iqbal

Figure 5.4: Table Structure using MySQL Complete Medical Care System

Suffering-Time	Treatment-Time	consulted-any doctor	Taken-medicine-without Dr advice	Basis taken medicine	Provide any app	Benefit if you were given an app	DoctorName	HospitalName	Sn	FirstConsultation	LastConsultation
days	3 days	No	yes	other reference	yes	yes	Najrul Islam	Square	20	0000-00-00	0000-00-00
days	3 days	No	yes	other reference	yes	yes	Najrul Islam	Square	20	0000-00-00	0000-00-00
Months	2 Months	Yes	No	NULL	Yes	Yes	Prof. Dr. Rehana Begum	Popular Diagnostic Center	12	20/10/2022	30/12/2022
Months	3 Months	Yes	Yes	Other Reference	No	Yes	Dr. Saraw Iqbal	Popular Diagnostic Center	33	7/8/2022	NULL

Figure 5.5: Table Structure using MySQL Complete Medical Care System

## 5.4 Testing and Implementation

This Process requirement is tested by us because we don't have any Professional tester. In this case, we know very well about our features and we have written down the points of features and tested Expectation of Organization. Our project testing has been implemented. Each Developing module is working very well and Verified.

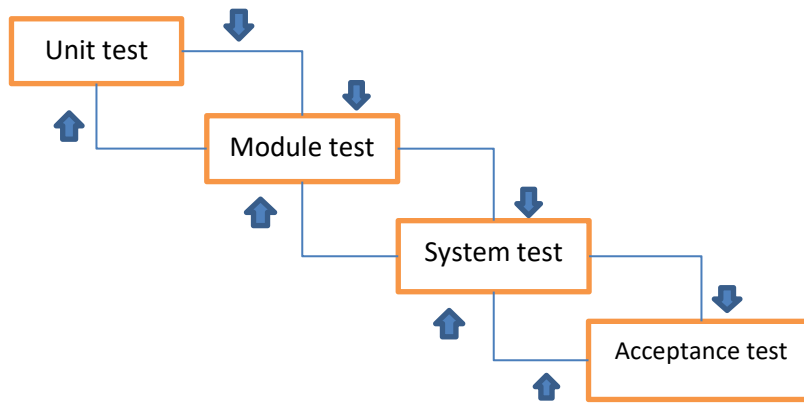


Figure 5.6: Testing and Implementation

### Unit Test

In our system we tested each part to make sure that our design is correctly implemented. We ensure that each part of our system works as per requirement.

### Module Test

We also tested a collection of dependent parts such as an object class, object data type and operation.

### System Test

System testing has been carried out remarked that this system works following expectation.

### Acceptance Test

Accepted test completed by customer. This system provides required output as expected by them.

## 5.5 Test Results and Reports

The report development is a major concern of our project. Firstly, our system is running well. In our testing report we have seen all expected output results. We didn't get any major problems when we test our Complete Medical care web dashboard and android application system. This report is obviously important and time saving for the Organization.

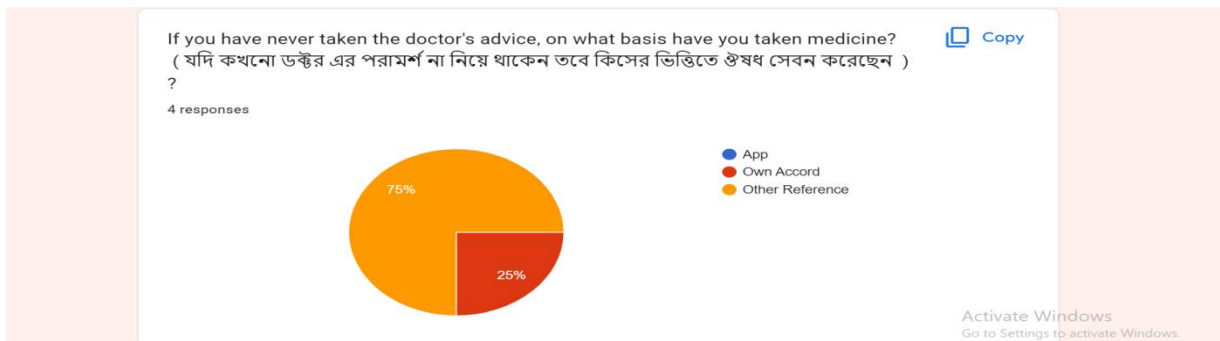


Figure 5.7: Test result and report

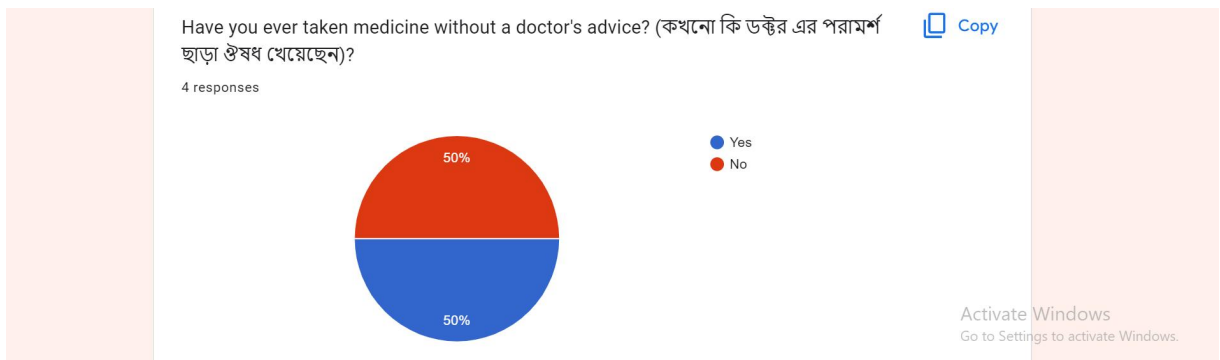


Figure 5.8 : Test result and report

## Chapter 6

### Conclusion and Future Scope

#### 6.1 Conclusion and Discussion

Treatment is always needed for everyone. But in this sector sometimes people get harassment and do not get proper information about medicine, Doctor or Medical. We can assure that our application will be a most helpful application.

Everyone needs to be well informed and concerned about the quality of care. Our system will help their quality health care. Therefore we hope that our system will help everyone.

#### 6.2 Development and Future Scope

Our project has many future opportunities. Our project is a monster project and we want to improve it so that it can be used across the country. To improve it and make it better

We need to reduce the limitations. We are planning to add more features like:

Patients can chat about sickness with their assigned doctor.

Users can comment on any type of article.

Medical examination fees can be provided on application.

Medical examination report can be provided on application.

## References:

[1] Park, KeeHyun & Lim, SeungHyeon (2012) "Construction of a Medication Reminder Synchronization System based on Data Synchronization", International Journal of Bio-Science and Bio-Technology, Vol.4, No. 4, pp1-10.

[2] "Smartphone medication adherence apps: Potential benefits to patients and providers", available at: <http://www.ncbi.nlm.nih.gov/pmc/articles/PMC3919626/>

[3] Slagle, J.M. Gordon, J.S., Harris, C.E., Davison, C.L., Culpepper, D.K., Scott P. and Johnson, K.B., (2011) "MyMediHealth" Designing a next generation system for child-centered medication management", Journal of Biomedical Informatics, Vol. 43, No. 5, pp. 27-31.

[4] Becker, E. Metsis, V., Arora, R., Vinjumur, J.K., Xu, Y. and Makedon, F. (2009) "SmartDrawer: RFID- Based smart medicine drawer for assistive environments", Proc. of Pervasive technologies related to assistive environments, June, pp 1-8.

[5] Ammouri, S. and Bilodeau, G.A. (2008) "Face and hands detection and tracking applied to the monitoring of medication intake", Proc. of Canadian Conf. on Computer and Robot Vision, May, pp. 147-154.

[6] Batz, D. Batz, M. Lobo, N.D.V. and Shah, M. (2005) "A computer vision system for monitoring medication intake", Canadian Conf. on Computer and Robot Vision, May, pp. 362-369.

[7] Prasad, B., (2013) "Social media, health care, and social networking", Gastrointest Endosc. Vol. 77, pp 492-495.

[8] Zao, J.K., Wang, M.Y., Peihuan, T. and Liu, J.W.S., (2010) "Smart Phone Based Medicine In-take Scheduler, Reminder and Monitor", IEEE e-Health Networking Applications and Services (Healthcom), pp 162 - 168.

[9] "Android", available at: [http://www.openhandsetalliance.com/android\\_overview.html](http://www.openhandsetalliance.com/android_overview.html)

[10] Mahmood, R., Mirzaei, N. Malek, S., (2014), "EvoDroid: Segmented Evolutionary Testing of Android Apps", FSE'14, November 16-21, 2014, Hong Kong, China

[11] "Medication Adherence", available at: <http://circ.ahajournals.org/content/119/23/3028.full>

[12] "Healthful Reminders for Medications, Beyond an Apple a Day", available at: [http://www.nytimes.com/2010/09/30/technology/personaltech/30smart.html?\\_r=0](http://www.nytimes.com/2010/09/30/technology/personaltech/30smart.html?_r=0)

[13] “Thinking outside the Pillbox: A System-wide Approach to Improving Patient Medication Adherence for Chronic Disease” (2009), A NEHI Research Brief July 2009, New England Healthcare Institute.

[14] Hughes, D. A., Bagust, A. Haycox, A., and Walley, T.O.M. (2001) “The impact of non-compliance on the cost effectiveness of pharmaceuticals: a review of the literature”, *Health Economics*, pp. 601– 615.

[15] Neupert, S.D., Patterson, T.R., Davis, A.A., Allarie, J.C.: Age Differences in Daily Predictors of Forgetting to Take Medication: The Importance of Context and Cognition. *Experimental Aging Research*

[16] Chia, L.R., Schlenk, E.A., Dunbar-Jacob, J.: Effect of personal and cultural beliefs on medication adherence in the elderly. *Drugs Aging* 23(3), 191–202 (2006)

## Appendix:

```
package com.example.medicin;

import androidx.annotation.NonNull;
import androidx.annotation.Nullable;
import androidx.appcompat.app.AppCompatActivity;
import android.content.ContentResolver;
import android.content.Intent;
import android.net.Uri;
import android.os.Bundle;
import android.view.KeyEvent;
import android.view.View;
import android.webkit.MimeTypeMap;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;
import com.google.android.gms.tasks.OnCompleteListener;
import com.google.android.gms.tasks.OnFailureListener;
import com.google.android.gms.tasks.OnSuccessListener;
import com.google.android.gms.tasks.Task;
import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;
import com.google.firebase.storage.FirebaseStorage;
```



```
import com.google.firebase.storage.StorageReference;

import com.google.firebase.storage.UploadTask;
import com.squareup.picasso.Picasso;

public class Add_Medicin extends AppCompatActivity implements
View.OnClickListener {

    de.hdodenhof.circleimageview.CircleImageView circleImageView;

    EditText name, quantity, description, diseases;

    Button enter;

    String Name, Description, Diseases;

    Integer Quantity;

    public static final int PICK_IMAGE = 1;
    public static final int GALLERY_REQUEST = 1;

    private Uri uri = null;

    DatabaseReference databaseReference;
```

```
FirestoreStorage storage;
StorageReference storageReference;

Medicine medicine1 = null;

String task;

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_add_medicin);

    Intent intent = getIntent();

    task = intent.getStringExtra("task");

    if(task.equals("update")) {

        medicine1 = (Medicine) intent.getSerializableExtra("object");

    }
}
```

```
databaseReference = FirebaseDatabase.getInstance().getReference("Medicine");

storage = FirebaseStorage.getInstance();
storageReference = storage.getReference("Medicine");

name = findViewById(R.id.name);
quantity = findViewById(R.id.quantity);
description = findViewById(R.id.des);
diseases = findViewById(R.id.diseases);

circleImageView = findViewById(R.id.medicin_image);

if(task.equals("update")) {

    name.setText(medicine1.Name);
    quantity.setText(medicine1.Quantity + "");
    description.setText(medicine1.Description);
    diseases.setText(medicine1.Diseases);

Picasso.get().load(Uri.parse(medicine1.medicin_image)).fit().into(circleImageView);

Uri share = Uri.parse(medicine1.medicin_image.trim());
```

```

        uri = share;
    }
    enter = findViewById(R.id.enter);

    enter.setOnClickListener(this);

    if(task.equals("update")) {

        enter.setText("Update");

    }

    circleImageView.setOnClickListener(this);

}

@Override
public void onClick(View v) {

    if(v.getId() == enter.getId()) {
        name.setOnKeyListener(new View.OnKeyListener() {
            @Override
            public boolean onKey(View v, int keyCode, KeyEvent event) {

```

```
        if(keyCode == KeyEvent.KEYCODE_ENTER && event.getAction() ==  
        KeyEvent.ACTION_DOWN) {
```

```
            Name = name.getText().toString().trim();
```

```
            return true;
```

```
        }
```

```
        return false;
```

```
    }
```

```
});
```

```
quantity.setOnKeyListener(new View.OnKeyListener() {
```

```
    @Override
```

```
    public boolean onKey(View v, int keyCode, KeyEvent event) {
```

```
        if(keyCode == KeyEvent.KEYCODE_ENTER && event.getAction() ==  
        KeyEvent.ACTION_DOWN) {
```

```
            Quantity = Integer.parseInt(quantity.getText().toString().trim());
```

```
            return true;
```

```
        }
```

```

        return false;
    }
});

diseases.setOnKeyListener(new View.OnKeyListener() {
    @Override
    public boolean onKey(View v, int keyCode, KeyEvent event) {

        if(keyCode == KeyEvent.KEYCODE_ENTER && event.getAction() ==
        KeyEvent.ACTION_DOWN) {

            Diseases = diseases.getText().toString().trim();

            return true;

        }

        return false;
    }
});

description.setOnKeyListener(new View.OnKeyListener() {
    @Override
    public boolean onKey(View v, int keyCode, KeyEvent event) {

```

```
        if(keyCode == KeyEvent.KEYCODE_ENTER && event.getAction() ==  
        KeyEvent.ACTION_DOWN) {
```

```
            Description = description.getText().toString().trim();
```

```
            return true;
```

```
        }
```

```
        return false;
```

```
    }
```

```
});
```

```
if(Description == null) {
```

```
    Description = description.getText().toString().trim();
```

```
}
```

```
if(Diseases == null) {
```

```
    Diseases = diseases.getText().toString().trim();
```

```
}
```

```
if(Quantity == null) {  
  
    Quantity = Integer.parseInt(quantity.getText().toString().trim());  
  
}  
  
if(Name == null) {  
  
    Name = name.getText().toString().trim();  
  
}  
  
if(Name == null) {  
  
    Toast.makeText(getApplicationContext(), "Name is null",  
Toast.LENGTH_SHORT).show();  
  
    return;  
  
}  
  
if(Description == null) {  
  
    Toast.makeText(getApplicationContext(), "Description is null",  
Toast.LENGTH_SHORT).show();  
  
    return;  
  
}
```



```
    }

    if(Quantity == null) {

        Toast.makeText(getApplicationContext(), "Quantity is null",
Toast.LENGTH_SHORT).show();

        return;

    }

    if(Diseases == null) {

        Toast.makeText(getApplicationContext(), "Diseases is null",
Toast.LENGTH_SHORT).show();

        return;

    }

    if(uri == null) {

        Toast.makeText(getApplicationContext(), "select an image",
Toast.LENGTH_SHORT).show();

        return;

    }

}
```

```

    }

    storageReference.child(Name).putFile(uri).addOnSuccessListener(new
    OnSuccessListener<UploadTask.TaskSnapshot>() {

        @Override

        public void onSuccess(UploadTask.TaskSnapshot taskSnapshot) {

            Toast.makeText(getApplicationContext(), "image uploaded",
            Toast.LENGTH_LONG).show();

            String ID = databaseReference.push().getKey();

            if(task.equals("update")) {

                ID = medicine1.ID;

            }

            Task<Uri> shared1 = taskSnapshot.getStorage().getDownloadUrl();

            while (!shared1.isComplete()) {

            };

            Uri shared = shared1.getResult();

```

```
        Medicine medicine = new Medicine(ID, Name, Description, Diseases,
Quantity, String.valueOf(shared));
```

```
databaseReference.child(ID).setValue(medicine).addOnCompleteListener(new
OnCompleteListener<Void>() {
```

```
    @Override
```

```
    public void onComplete(@NonNull Task<Void> task) {
```

```
        Toast.makeText(getApplicationContext(), "data is addedd",
Toast.LENGTH_LONG).show();
```

```
    }
```

```
}).addOnFailureListener(new OnFailureListener() {
```

```
    @Override
```

```
    public void onFailure(@NonNull Exception e) {
```

```
        Toast.makeText(getApplicationContext(), "Data is not addedd",
Toast.LENGTH_LONG).show();
```

```
    }
```

```
});
```

```
}
```

```
}).addOnFailureListener(new OnFailureListener() {
```

```

@Override

public void onFailure(@NonNull Exception e) {

    Toast.makeText(getApplicationContext(), "image not uploaded",
Toast.LENGTH_LONG).show();

    }

});

} else if(v.getId() == circleImageView.getId()) {

    Intent photoPickerIntent = new Intent(Intent.ACTION_GET_CONTENT);
    photoPickerIntent.setType("image/*");
    startActivityForResult(photoPickerIntent, GALLERY_REQUEST);

}

}

private String getFileExtension(Uri uri)
{
    String extension;

    ContentResolver contentResolver = getContentResolver();

    MimeTypeMap mimeTypeMap = MimeTypeMap.getSingleton();

```

```

        extension=
mimeTypeMap.getExtensionFromMimeType(contentResolver.getType(uri));

        return extension;

    }

    @Override

    protected void onActivityResult(int requestCode, int resultCode, @Nullable Intent data)
    {

        super.onActivityResult(requestCode, resultCode, data);

        if(requestCode == GALLERY_REQUEST && resultCode==RESULT_OK &&
data != null){

            uri = (data.getData());

            Picasso.get().load(uri).fit().into(circleImageView);

            //uploaded_question_name.setText((String.valueOf(uri) + getFileExtension(uri)) );

        }

    }

}

package com.example.medicin;

import androidx.annotation.NonNull;

import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;

import android.os.Bundle;

```

```
import android.os.Parcelable;

import android.view.View;

import android.view.animation.Animation;

import android.view.animation.AnimationUtils;

import android.widget.AdapterView;

import android.widget.AdapterView.OnItemClickListener;

import android.widget.AdapterView.OnItemSelectedListener;

import android.widget.ArrayAdapter;

import android.widget.Button;

import android.widget.ImageView;

import android.widget.Spinner;

import android.widget.TextView;

import android.widget.Toast;

import com.google.android.gms.tasks.OnCompleteListener;

import com.google.android.gms.tasks.OnFailureListener;

import com.google.android.gms.tasks.Task;

import com.google.firebase.database.DataSnapshot;

import com.google.firebase.database.DatabaseError;

import com.google.firebase.database.DatabaseReference;

import com.google.firebase.database.FirebaseDatabase;

import com.google.firebase.database.ValueEventListener;

import java.io.Serializable;

import java.util.ArrayList;

import java.util.List;

import java.util.Map;
```

```
public class Home extends AppCompatActivity implements View.OnClickListener {
```

```
    Spinner spinner;
```

```
    TextView add, details;
```

```
    DatabaseReference databaseReference;
```

```
    Button remove;
```

```
    ArrayAdapter<String> aa;
```

```
    List<String> list;
```

```
    Medicine medicine;
```

```
    List<Medicine> medicineList;
```

```
    @Override
```

```
    protected void onCreate(Bundle savedInstanceState) {
```

```
        super.onCreate(savedInstanceState);
```

```
        setContentView(R.layout.activity_home);
```

```
        list = new ArrayList<>();
```

```

medicineList = new ArrayList<>();

databaseReference = FirebaseDatabase.getInstance().getReference("Medicine");

spinner = findViewById(R.id.spinner);
add = findViewById(R.id.add);
details = findViewById(R.id.see);
remove = findViewById(R.id.remove);

aa = new ArrayAdapter(this,android.R.layout.simple_spinner_item,list);
aa.setDropDownViewResource(android.R.layout.simple_spinner_dropdown_item);

remove.setOnClickListener(this);

add.setOnClickListener(this);
details.setOnClickListener(this);

spinner.setOnItemClickListener(new AdapterView.OnItemClickListener() {
    @Override
    public void onItemClick(AdapterView<?> parent, View view, int position, long
id) {

        if(position - 1 >= 0) {

```



```

        medicine = medicineList.get(position - 1);

    }

}

@Override
public void onNothingSelected(AdapterView<?> parent) {

}

});

}

@Override
public void onClick(View v) {

    if(v.getId() == spinner.getId()) {

    } else if(v.getId() == add.getId()) {

        Intent intent = new Intent(Home.this, Add_Medicin.class);

```

```

intent.putExtra("task", "add");

startActivity(intent);

} else if(v.getId() == details.getId()) {

    if(medicine != null) {

        Intent intent = new Intent(Home.this, Add_Medicin.class);

        intent.putExtra("task", "update");
        intent.putExtra("object", medicine);

        Animation object
            = AnimationUtils
                .loadAnimation(
                    getApplicationContext(),

                    // blink file is in anim folder

                    R.anim.zoom);

        ImageView imageView = findViewById(R.id.imaging);
        imageView.setAnimation(object);

```

```

        startActivity(intent);

    } else {

        Toast.makeText(getApplicationContext(), "please select an list",
Toast.LENGTH_LONG).show();

    }

} else if(v.getId() == remove.getId()) {

    if(medicine != null) {

databaseReference.child(medicine.ID).removeValue().addOnCompleteListener(new
OnCompleteListener<Void>() {

    @Override

    public void onComplete(@NonNull Task<Void> task) {

        Toast.makeText(getApplicationContext(), "data removed",
Toast.LENGTH_LONG).show();

    }

}

}

onStart();

```

```

    }
    }).addOnFailureListener(new OnFailureListener() {
        @Override
        public void onFailure(@NonNull Exception e) {

            Toast.makeText(getApplicationContext(), "failed to delete",
Toast.LENGTH_LONG).show();

        }
    });

} else {

    Toast.makeText(getApplicationContext(), "select a list",
Toast.LENGTH_LONG).show();

}

}

}

@Override
protected void onStart() {

```

```

list.clear();

list.add("Medicine");

medicineList.clear();

databaseReference.addValueEventListener(new ValueEventListener() {

    @Override

    public void onDataChange(@NonNull DataSnapshot snapshot) {

        for(DataSnapshot dataSnapshot : snapshot.getChildren()) {

            //list.add(dataSnapshot.getKey());

            Map<String, String> map = (Map<String, String>) dataSnapshot.getValue();

            String id = map.get("id");

            String name = map.get("name");

            String quantity = String.valueOf(map.get("quantity"));

            String medicin_image = map.get("medicine_image");

            String description = map.get("description");

            String diseases = map.get("diseases");

            list.add(name);

```

```
        medicineList.add(new Medicine(id, name, description, diseases,  
Integer.parseInt(quantity), medicin_image));
```

```
    }
```

```
    try {
```

```
        spinner.setAdapter(aa);
```

```
    } catch (Exception e) {
```

```
    }
```

```
}
```

```
@Override
```

```
public void onCancelled(@NonNull DatabaseError error) {
```

```
}
```

```
});
```

```
super.onStart();
```

```
}
```

```
@Override
protected void onResume() {

    Animation object
        = AnimationUtils
        .loadAnimation(
            getApplicationContext(),

            // blink file is in anim folder
            R.anim.zoom);

    ImageView imageView = findViewById(R.id.imaging);

    imageView.setAnimation(object);

    super.onResume();
}
}
```

ORIGINALITY REPORT

<b>19%</b>	<b>19%</b>	<b>5%</b>	<b>%</b>
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

<b>1</b>	<b>dspace.daffodilvarsity.edu.bd:8080</b> Internet Source	<b>13%</b>
<b>2</b>	<b>ami.info.umfcluj.ro</b> Internet Source	<b>3%</b>
<b>3</b>	<b>www.slideshare.net</b> Internet Source	<b>2%</b>
<b>4</b>	<b>ijesc.org</b> Internet Source	<b>&lt;1%</b>

Exclude quotes Off

Exclude matches Off

Exclude bibliography Off