

GRADUATION PROJECT

2021 - 2022



DLEEL APPLICATION

By

Albatol Bassam Almeshali

Dhay Fozan Al-Fozan

Rania suliman almeghem

Ghaida Abdulaziz albedaiwi

Lama Abdullah ALshreef

Supervised by: Dr.Faten kareem

List of contents

| | |
|--|----|
| | i |
| List of Tables | iv |
| List of Figures | v |
| Chapter 1 Introduction..... | 1 |
| 1.1 Problem Statement & Significance..... | 1 |
| 1.2 Proposed solution | 1 |
| 1.3 Project Domain & limitations..... | 1 |
| Chapter 2 Background information & Related work..... | 3 |
| 2.1 Background information | 3 |
| 2.2 Related work survey..... | 3 |
| 2.2.1 Sahel application..... | 4 |
| 2.2.2 Metrash2 application..... | 5 |
| 2.2.3 Balade application..... | 6 |
| 2.2.4 Omanna application..... | 7 |
| 2.3 Proposed & Similar System comparison..... | 8 |
| 2.4 Survey..... | 9 |
| Chapter 3 System analysis..... | 11 |
| 3.1.1 Functional requirements:..... | 11 |
| 3.1.2 Non-Functional Requirements:..... | 11 |
| 3.2 requirements analysis..... | 12 |
| 3.3 Use case diagram | 17 |
| 3.4 Sequence diagram..... | 18 |
| [6] [8]..... | 18 |
| 3.5 Class diagram | 23 |
| Chapter 4 System design..... | 25 |
| 4.1 System architecture | 25 |
| [9] | 25 |
| 4.2 User interface Design:..... | 26 |
| Chapter 5 Implementation | 34 |
| 5.1 Implementation Requirements..... | 34 |
| 5.2.1.Hardware Requirements..... | 34 |
| 5.2.2. Software Requirements | 34 |

| | |
|---|----|
| 5.3 Implementation details..... | 35 |
| 5.4 I/O screens | 77 |
| 5.5 Summary | 82 |
| Chapter 6 Testing:..... | 84 |
| 6.1 Test Plan..... | 84 |
| 6.2.1 Testing Strategy | 84 |
| 6.2.2. Schedule of testing activities | 85 |
| 6.2.3 Test Cases..... | 85 |
| 6.3 TEST Result:..... | 94 |
| Chapter 7 conclusion | 96 |
| 7.1 Evaluation: | 96 |
| 7.2 Future work..... | 96 |
| References | 97 |
| Appendices..... | 98 |

List of Tables

| | |
|--|----|
| Table 2- 1 comparison between DLEEL and the similar applications | 8 |
| Table 3- 1 Actors | 12 |
| Table 3- 2 “Register” Use case description..... | 12 |
| Table 3- 3 “Login” Use case description..... | 13 |
| Table 3- 4 “Search” Use case description | 13 |
| Table 3- 5 “Change Language” Use case description..... | 14 |
| Table 3- 6 “Browse services” Use case description | 14 |
| Table 3- 7 “Browse services” Use case description | 15 |
| Table 3- 8 “Recently Searches” Use case description..... | 15 |
| Table 3- 9 “Favorite Services” Use case description..... | 16 |
| Table 3- 10 “Logout” Use case description..... | 16 |
| Table 5- 1-<<Login.Java>> Code..... | 35 |
| Table 5- 2-<<Login.Xml>> Code | 37 |
| Table 5- 3-<<Register.Java>>Code | 40 |
| Table 5- 4-<<Register.Xml>>Code..... | 42 |
| Table 5- 5-<<main.java>>Code | 45 |
| Table 5- 6-<<main.Xml>> Code | 49 |
| Table 5- 7-<<Profile.Java>> Code..... | 55 |
| Table 5- 8-<<Profile.Xml>> Code | 58 |
| Table 5- 9-<<Menue.Java>> Code..... | 60 |
| Table 5- 10-<<Menue.Xml>> Code | 63 |
| Table 5- 11-<<HealthService.Java>> Code..... | 68 |
| Table 5- 12-<<HealthService.Xml>> Code..... | 70 |
| Table 5- 13-<<Testing_For_Covid.Java>> Code | 72 |
| Table 5- 14-<<Testing_For_Covid.Xml>> Code | 75 |
| Table 6- 1-<<Login>>test case | 85 |
| Table 6- 2-<<REGESTER>> test case..... | 88 |
| Table 6- 3-<<Edit password>> test case | 90 |
| Table 6- 4-<<favorite service>> test case | 91 |
| Table 6- 5-<<Edit profile>>test case | 93 |

List of Figures

| | |
|---|----|
| Figure 2- 1 Shel Application | 4 |
| Figure 2- 2 Metrash2 Application | 5 |
| Figure 2- 3 Balade Application | 6 |
| Figure 2- 4 Omanna Application | 7 |
| | |
| Figure 3- 1-DLEEL -use case diagram | 17 |
| Figure 3- 2-<Login use case > sequence diagram | 18 |
| Figure 3- 3-<search> sequence diagram..... | 19 |
| Figure 3- 4-<Browse service> sequence diagram | 19 |
| Figure 3- 5-<favorite list> sequence diagram | 20 |
| Figure 3- 6-<Modifying information> sequence diagram | 20 |
| Figure 3- 8-DLEEL- class diagram | 23 |
| | |
| Figure 4- 1-Architecture design | 25 |
| Figure 4- 2 -<main page> interface | 26 |
| Figure 4- 3 -<Login> interface | 26 |
| Figure 4- 4-<Sign-up> interface..... | 26 |
| Figure 4- 5 quick access toolbar..... | 27 |
| Figure 4- 6-<homepage> interface..... | 27 |
| Figure 4- 7-<menu services > interface..... | 28 |
| Figure 4- 8-<search> interface | 28 |
| Figure 4- 9-< instruction > interface | 29 |
| Figure 4- 10-<favorite services for member> interface | 29 |
| Figure 4- 11-<favorite services for guest> interface | 30 |
| Figure 4- 12-<profile> interface | 30 |
| Figure 4- 13-<History page> interface | 31 |
| Figure 4- 14-<edit profile> interface..... | 31 |
| Figure 4- 15-<edit Username> interface..... | 32 |
| Figure 4- 16-<edit password> interface | 32 |
| | |
| Figure 5- 1-<<main page>>interface | 77 |
| Figure 5- 2-<<Register>>interface | 78 |
| Figure 5- 3-<<login>>interface | 78 |
| Figure 5- 4-<<home page>>interface | 79 |
| Figure 5- 5-<<menu>>interface | 79 |
| Figure 5- 6<<step>>interface | 80 |
| Figure 5- 7-<<favorite service>>interface | 80 |
| Figure 5- 8-<<profile>> interface | 81 |
| Figure 5- 9-<<Edit password>> interface | 81 |

Abstract :

Since technology become as daily essential and the mobile applications has been growing which made human depend on doing their needs by their mobile, including their education, transportation, shopping, health and government services. Within DLEEL applications it serve mostly citizens people, DLEEL is an application to help people by searching for specific service related to government when they need a service and they don't know how to reach it. DLEEL will provide them a video and steps to reach their needs

Chapter 1 : Introduction

Chapter 1 Introduction

1.1 Problem Statement & Significance

In light of the rapid digital development in the Kingdom of Saudi Arabia, the percentage of paper transaction in 2020 reached 0% Mobile applications, therefore, are considered to be important in many ways. Government mobile applications are important because they allow the citizens to access their information and request services for themselves such as, showing the digital ID, health statistics, digital passport, digital driving license, national address, booking an appointment in some government agencies and other services.

Government mobile application contributed to raising the efficiency, improving performance, and providing many facilities and opportunities for citizens and residents. Therefore, the need for a guide explaining how to use these applications has become very important.

1.2 Proposed solution

In this project, we proposed DLEEL a “guide application” in order to clarify the necessary steps to complete the procedures for e-government services, through a simplified scenario and an explanatory video.

1.3 Project Domain & limitations

Citizens and residents who can seamlessly use technology can benefit from the DLEEL app. The application is designed in order to fulfil the requirements of both citizens and residents. It focuses specifically on clarifying services and procedures that meet user requirement regarding government applications. The mobile application will be designed for Android platform only. It serves adults of the age of 18 and over.

Chapter 2 : Background information & Related work

Chapter 2 Background information & Related work

In this chapter, we provide an overview of the platform and programming language we chose to build the applications, why we chose it as well as a number of similar applications that have been proposed and highlight their strengths and weaknesses. Finally, we compared these systems to DLEEL app.

2.1 Background information

In this project we choose android platform to build DLEEL application because Android system supports more than one device. Manufacturers that make devices supporting Android apps outnumber those developing to suit IOS specifies apps. Java programming language will be used because Java is platform-independent can be easily migrated from one computer system to another. The ability to run the same program on many different system is critical to World Wide Web to software, and Java succeeds by being platform-independent at both the source code and binary levels. [1]

2.2 Related work survey

In this section, we review a set of work related to government platforms applications and website. We discussed their main features and disadvantages.

2.2.1 Sahel application

The “Sahel” application is a unified government application in Kuwait country for electronic services for various government agencies, through which the citizens and resident perform services and transactions more easily, quickly and effectively, according to distinguished quality standards, to provide a new experience for the completion of government transactions. [2]

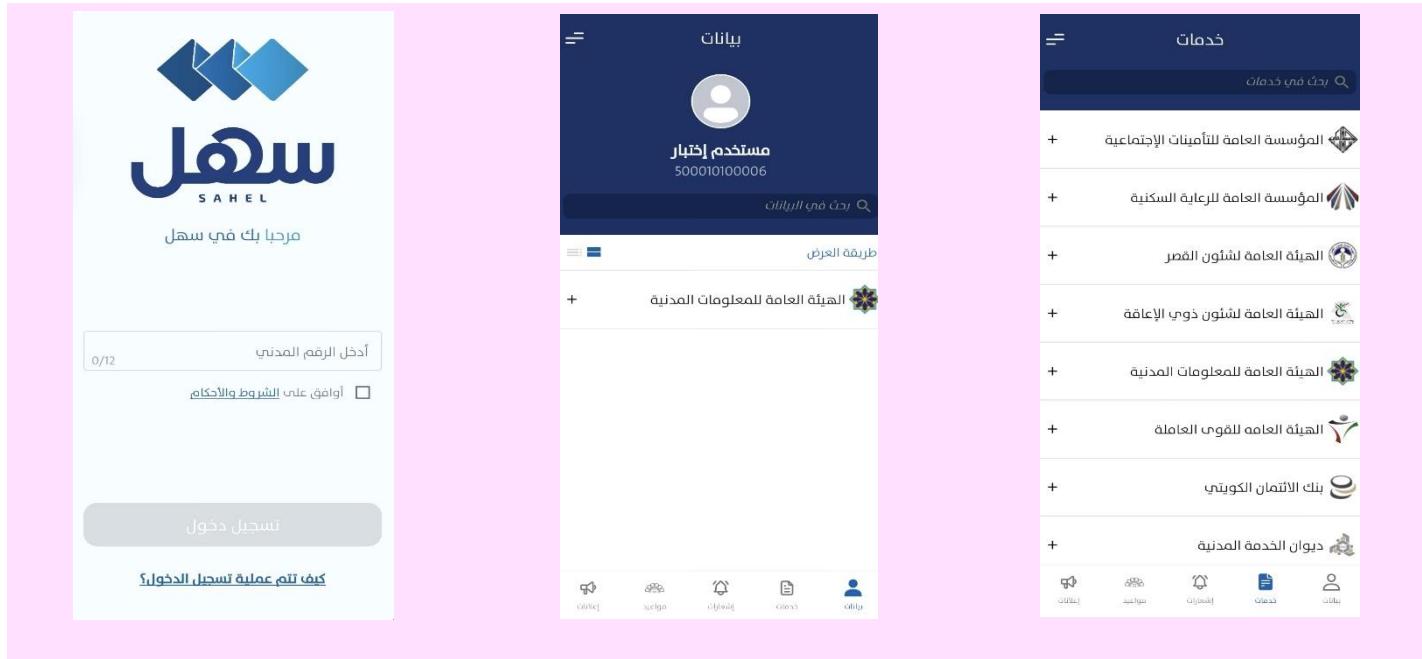


Figure 2- 1 Shel Application

Features and service:

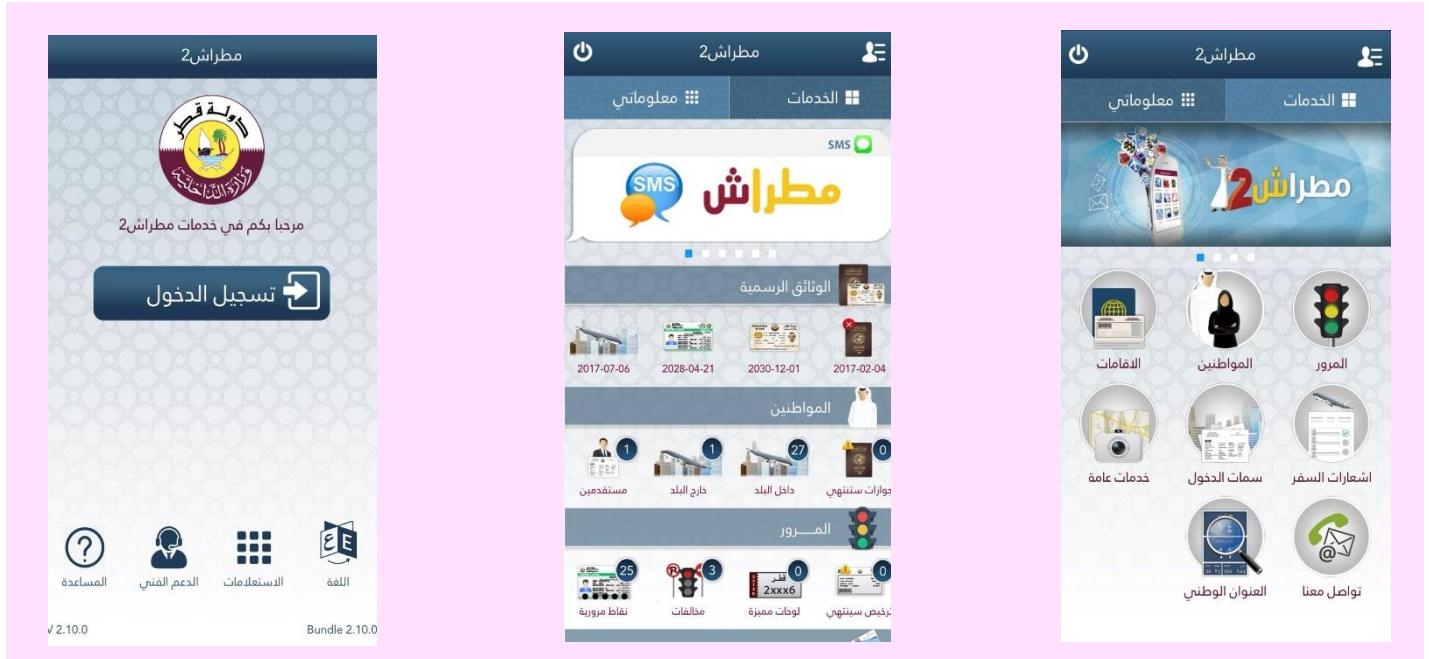
- Allow for quick and improved services of government agencies.
- Simplify government processes and make it easier for citizens and residents to complete government transactions.
- Reduce the number of auditors required in government agencies and institutions.
- Facilitate conducting government transactions through electronic applications.
- Connect services for all government agencies and institutions through a single application.
- Save time and costs for citizens and residents to complete their government transactions.
- Eliminate bureaucracy and reduce paperwork.
- Promote integrity and transparency through digital transformation.
- Be the starting point in achieving digital transformation in the State of Kuwait.

Disadvantages:

-Interface of this app not clear and complex.

2.2.2 Metrash2 application

Metrash2 application lunched by General department of information system ministry of interior in Qatar for phones to obtain the services of the ministry, from anywhere and at any time. [3]



Features and services:

- Implementation of full procedures, including payment of service fees through electronic payment.
- Interactive (Contact Us) service with the public.
- You can now activate the service on more than one device.

Disadvantage:

- Technical difficulties encountered by a number of citizens in accessing the Metrash2 application due to the great pressure.

2.2.3 Balade application

The “Balade” portal was developed by a project with entities affiliated with the Balade application to be the national portal that supports the local community through assistance services that help in submitting application through electronic information services, which help in linking businesses to the neighboring area. [4]



Features and services:

- Interactive services that help in promoting the concept of community partnership to improve the quality of services provided to beneficiaries.
- Electronic services that help submit electronic applications for obtaining the most used licenses
- Information services that help beneficiaries make relevant decisions in a way that contributes to improving level of satisfaction with municipal sector authorities.

Disadvantage:

- Difficult to download the app.-Not available for all devices.

2.2.4 Omanna application

Omanna application is a government application is a gateway to government applications and services in sultanate of Oman making it easier for citizens, residents and visitors to carry out government services such as education and health. [5]

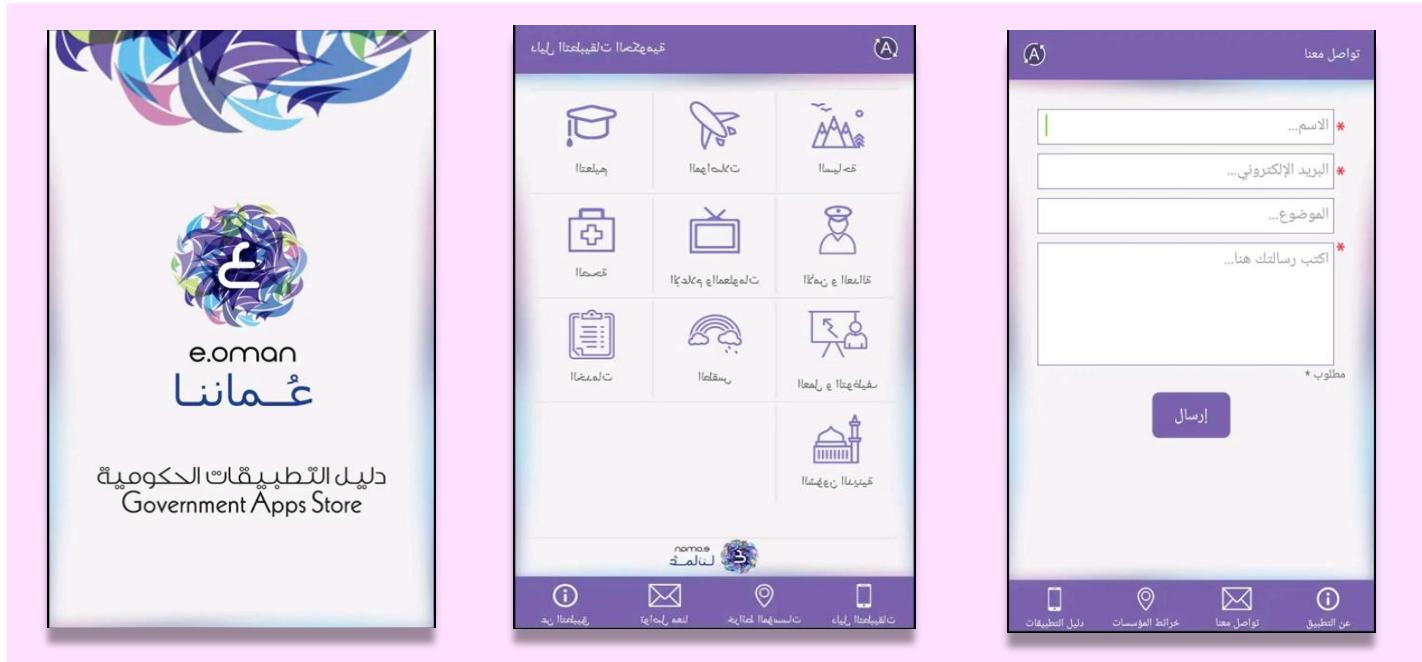


Figure 2- 4 Omanna Application

Features and services:

- Provides easy and convenient access to services and provides information for more options.
- Provides a single entry to all services it contains the security is high, as it verifies the information on the identity method and phone number.
- All services are characterized by the presence of easy-to-use forms, automatic updating of documents, and online payment.

Disadvantages:

- It's does not show the basic requirement needed to complete the services.
- It's does not show the steps to complete the services.
- simple interface.

2.3 Proposed & Similar System comparison

Table 2-1 summaries the different government platforms and compare their features with DLEEL application.

| Application Features | Unified National platform (KSA) | Sahel (Kuwait) | Metrach2 (Qatar) | Omanna (Oman) | Balade | DLEEL |
|--|---------------------------------|----------------|------------------|---------------|--------|-------|
| Comprehensive services for all government agencies | yes | yes | no | Yes | no | yes |
| Give an introduction about the services when choosing it | yes | yes | no | Yes | yes | yes |
| Show the basic requirements needed to complete the service | yes | - | no | Yes | yes | yes |
| Displays the steps to complete the service | yes | - | no | Yes | yes | yes |
| Connecting to the technical support of the services | yes | - | no | No | no | yes |
| Transfer to the Ministry's website | yes | - | yes | No | no | yes |
| Transfer to the application | yes | yes | no | No | no | yes |

Table 2- 1 comparison between DLEEL and the similar applications

2.4 Survey

We have published a survey and answered by 200 adult citizens to collect requirements, knowing what they need and the importance of the application. The answers showed that the application will be useful for a large segment of society as it will help them know how to complete the government operations they want.

Below are the questions that were put into the survey:

1-Are you experiencing difficulties in submitting government transactions?

2-Does the multiplicity of government applications cause you to be confused in deciding which application you need?

3- Have you ever needed to complete one of your transactions but didn't know the application of this service?

4-Have you ever needed a video explaining how to use a particular service?

5- Have you ever had a problem using a government app? Do you have an excellent background on government application?

6- Have you ever had a problem not knowing which app to use?

7- Does having a directory application for government applications make it easier for you to find the services you need?

8- Does having a directory application for government applications make it easier for you to find the services you need?

9- Do you support an app that collects government applications?

10- Have you ever been defrauded by an informal link to an e-service?

To see the results and more details, see the appendices.

Chapter 3 : System analysis

Chapter 3 System analysis

3.1 Requirement specification

3.1.1 Functional requirements:

- User can create an account.
- User can login.
- User can search for services.
- User browse the services easily.
- User can choose the application they need.
 - The system will browse the information related to the application.
 - The system will provide the formal link to download the application.
 - The system will shows explanatory video.
 - The system will show them the steps they need to complete the task.
- User can add the application to favourite list.
- User can modify his information.
- User can logout from the application.

3.1.2 Non-Functional Requirements:

- Security and Reliability.
- User Interface & Usability.
- Maintainability.
- Ethical.
- Scalability and Capacity.
- Performance.
- Availability.

3.2 requirements analysis

Table 3- 1 Actors

| Participating actors and roles | Use case name | Use case overview |
|--------------------------------|--------------------|--|
| Guest | Register | User create account in the application. |
| User | Search | User can search for services. |
| | Browse services | User can browse the menu easily. |
| Member | Login | User can login into application with password and username |
| | Favourite list | User can add services to favourite list. |
| | History page | User can browse what they recently searched. |
| | Modify information | User can edit their information. |
| | Log out | User can logout from application. |

Table 3- 2 "Register" Use case description

| User case name | Register |
|------------------------|--|
| Actor | Guest. |
| Description | This use case allow user to create an account. |
| Precondition | The user must connect to the internet. |
| Event flow | 1-The user inserts all requirement information. 2-The user click on sign-up button. 3-The system display success message. |
| Alternative event flow | A1: Some fields not filled 3.The system displays an error message showing that there is field is empty. A2: Invalid Email 3.The system displays a message indicating that the Email is already taken. |
| Post condition | A standard user account will be created successfully |

Table 3- 3 "Login" Use case description

| User case name | Login |
|------------------------|--|
| Actor | Member |
| Description | This use case allow user to Login into application. |
| Precondition | The user has an account |
| Event flow | 1-The user enters all required information in the login page. 2-The user clicks on the "login" button 3- The system displays success message 4-The system displays the Homepage |
| Alternative event flow | A1: Invalid Email or password 3. The system displays a message indicating that the Email and/or the password is incorrect. A2: Email or password is empty 3.The system displays a message indicating that the Email and/or password is empty. |
| Post condition | The user is logged in successfully |

Table 3- 4 "Search" Use case description

| User case name | Search |
|----------------|---|
| Actor | User |
| Description | User can search for services |
| Event flow | 1-The user enters what they need in search box. 2-The system display the services they searched about. |
| Post condition | The user searched successfully |

Table 3- 6 “Browse services” Use case description

| User case name | Browse services |
|----------------|--|
| Actor | User. |
| Description | This use case allow user to browse menu of the sections in the application. |
| Event flow | 1-The user clicks on the menu button 2-select section 3-browse service |
| Post condition | The user browses the menu |

Table 3- 7“Browse services” Use case description

| User case name | Modify information |
|------------------------|---|
| Actor | Member |
| Description | This use case allow user to modify their information |
| Precondition | The user is logged in |
| Event flow | 1-The user click my account button 2-the system display profile page 3- the user click edit my password button 4- enter new password 5- the user click Ok |
| Alternative event flow | A1: Some fields not filled The system displays an error message showing that there is field is empty. |
| Post condition | Successfully modified the password |

Table 3- 8“Recently Searches” Use case description

| User case name | History page |
|----------------|---|
| Actor | Member |
| Description | This use case allow user to browse recently searched |
| Precondition | The user is logged in |
| Event flow | 1-user click my account button 2-System display profile page 3-User click the history button 4-System display history page |
| Post condition | The user browsed recently searched |

Table 3- 9 "Favorite Services" Use case description

| User case name | Favourite list |
|----------------|--|
| Actor | Member |
| Description | This use case allow user to add the services to bookmark |
| Precondition | The user is logged in |
| Event flow | 1-user click the favourite button 2-System display favourite page |
| Post condition | The user added the services to the favourite list |

Table 3- 10 "Logout" Use case description

| User case name | Logout |
|----------------|--|
| Actor | Member |
| Description | User can logout from application |
| Precondition | The user is logged in |
| Event flow | 1-user click my account button 2-system display profile page 3-user click the logout button 4-user logout |
| Post condition | The user has been successfully logged out |

3.3 Use case diagram

Use case diagrams can be summed up the details of system actors and how they interact with the system to build one, you will use a set of specialized symbols and connectors. It can help the team discuss and represent scenarios, goals, system scope and when to use

[6] [7]

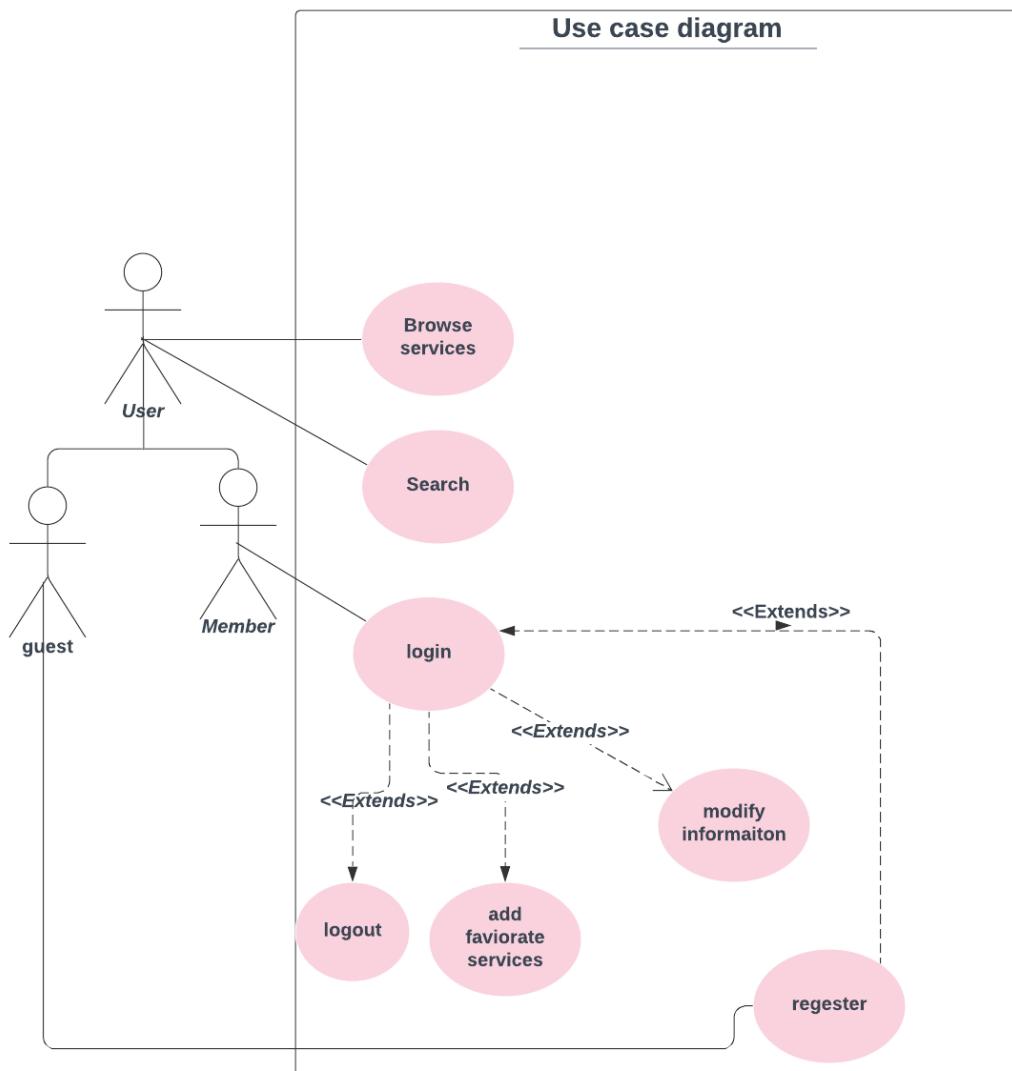


Figure 3- 1-DLEEL -use case diagram

3.4 Sequence diagram

From a dynamic perspective, we used the UML sequence diagram to model the interaction between the objects and classes involved in the scenario. Describes the sequence of messages interchanged over time and needed to carry out the use case. Hereafter, we present the sequence diagrams of the main use cases of DLEEL Application.

[6] [8]

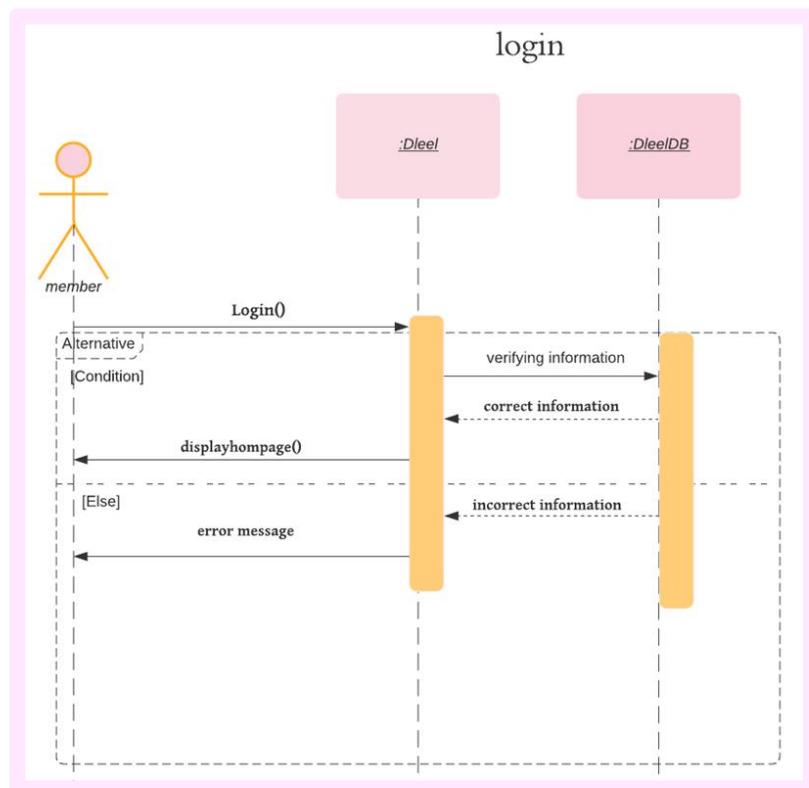


Figure 3- 2-<Login use case > sequence diagram

Figure 3-2 depicts the « Login in » sequence diagram. The user should enter their authentication parameters, i.e., its Email and password. If the parameters are correct, he can access to the homepage. Otherwise, an error message will be displayed.

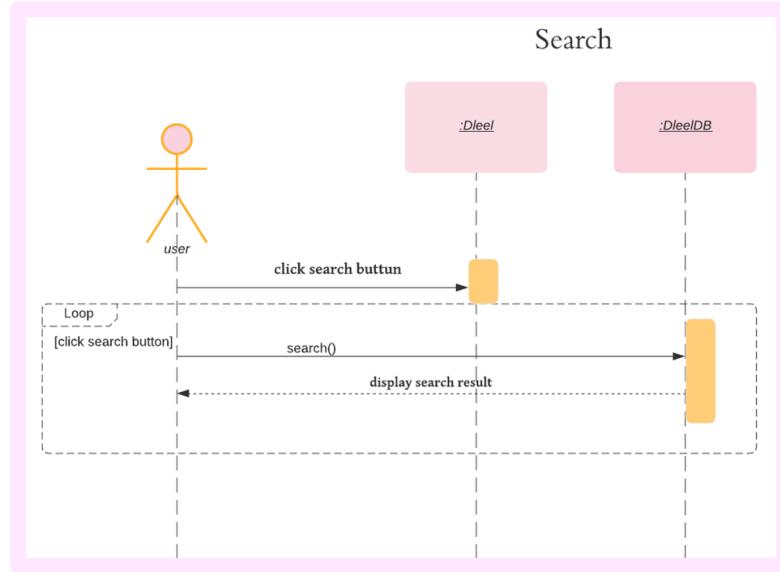


Figure 3- 3-<search> sequence diagram

Figure 3-3 shows the search when the Member enters the text the topic that he/she is searching for then presses the search button.

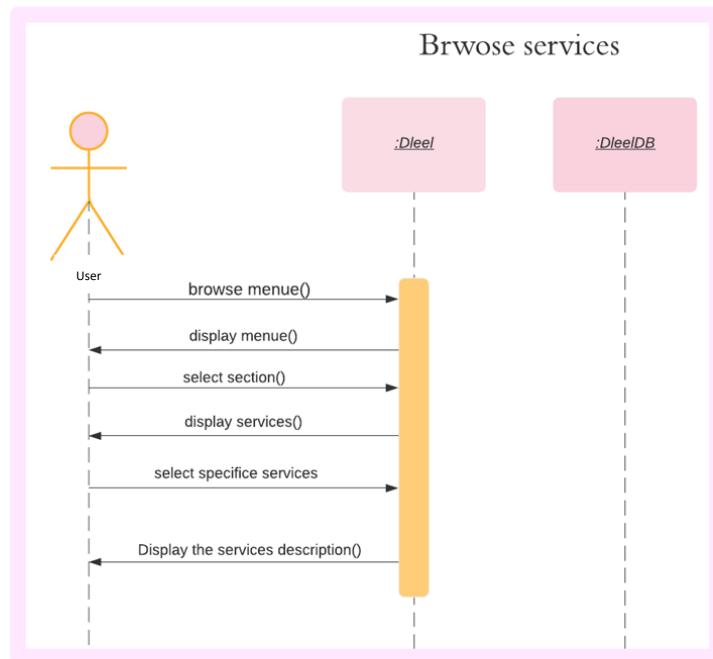


Figure 3- 4-<Browse service> sequence diagram

The Figure 3-4 The user can browse the services by pressing the menu button.

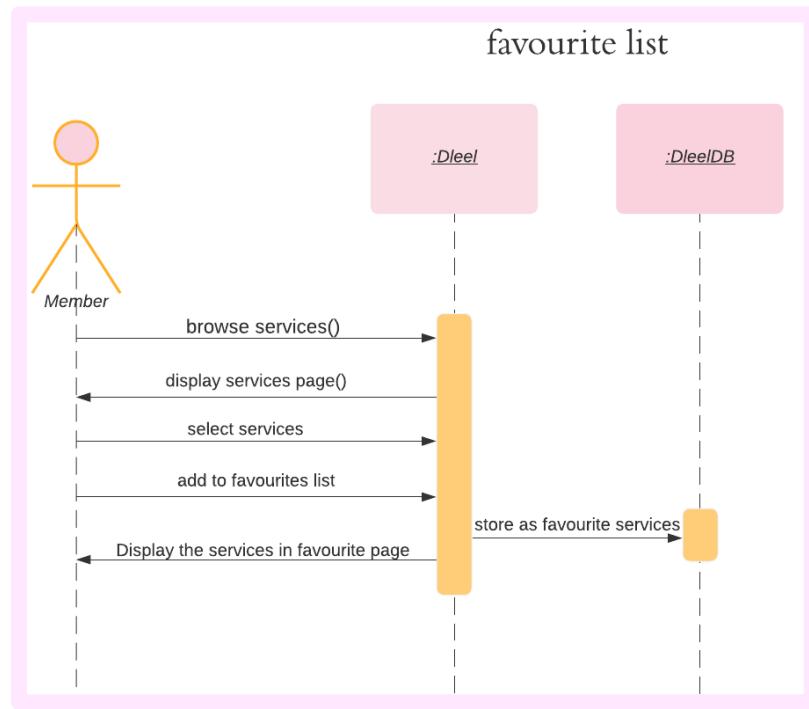


Figure 3- 5-<favorite list> sequence diagram

Figure 3-5 the application also provides the ability to add a service to the favourites list. After browsing the services, the user can choose a specific service and add it to the list, the application saves it in the database and then displays it on the favourites list page.

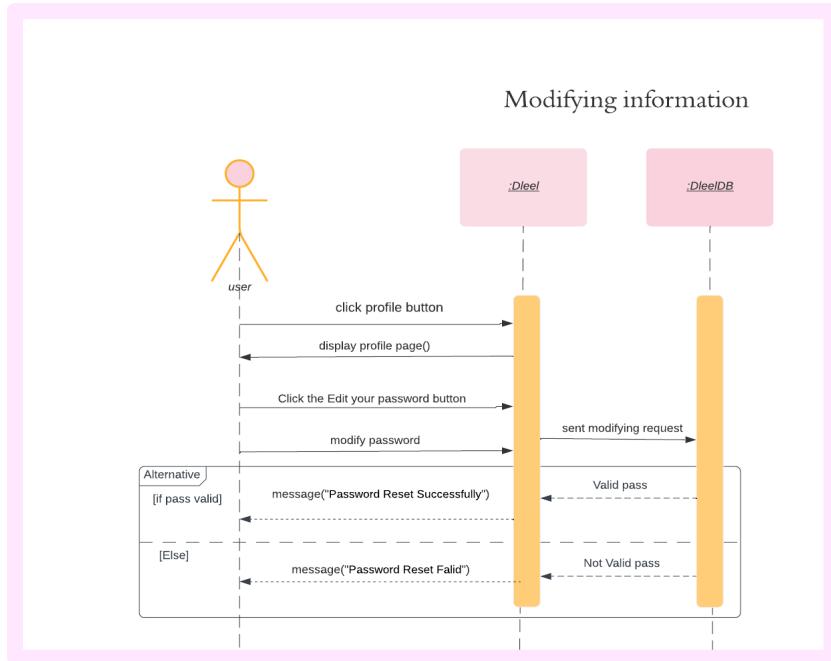


Figure 3- 6-<Modifying information> sequence diagram

Figure 3-6 The DLEEL application allows easy modification of the member's information. The member presses the modification button and then changes his information. Then the application stores the information in the database and then sends a message to the member that the modification is successful.

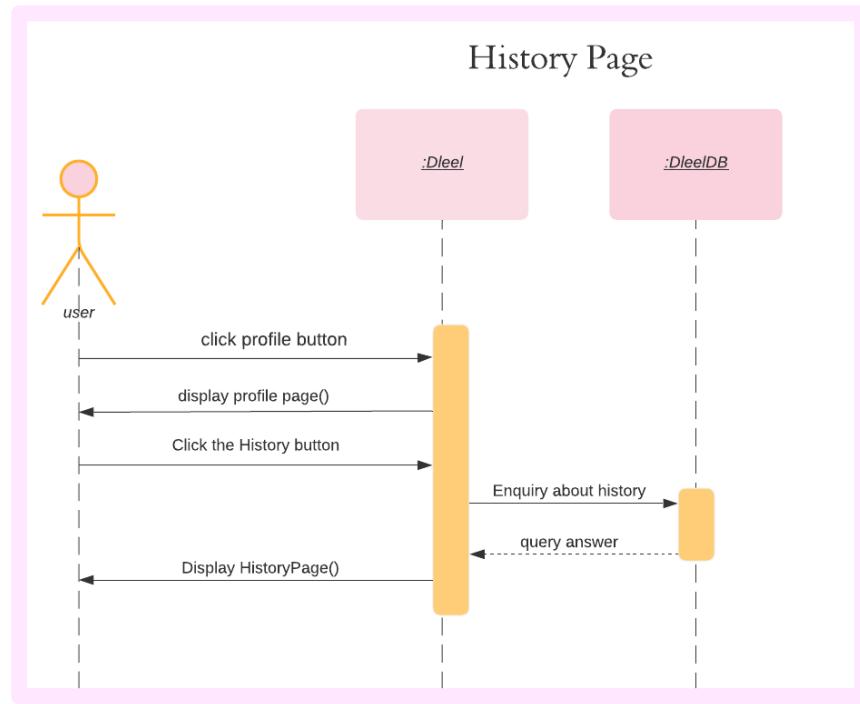


Figure 3- 7-<History page> sequence diagram

Figure 3-7 The user can view recent searches by clicking on the profile button and then clicking on the history page. The application displays the results after fetching them from the database.

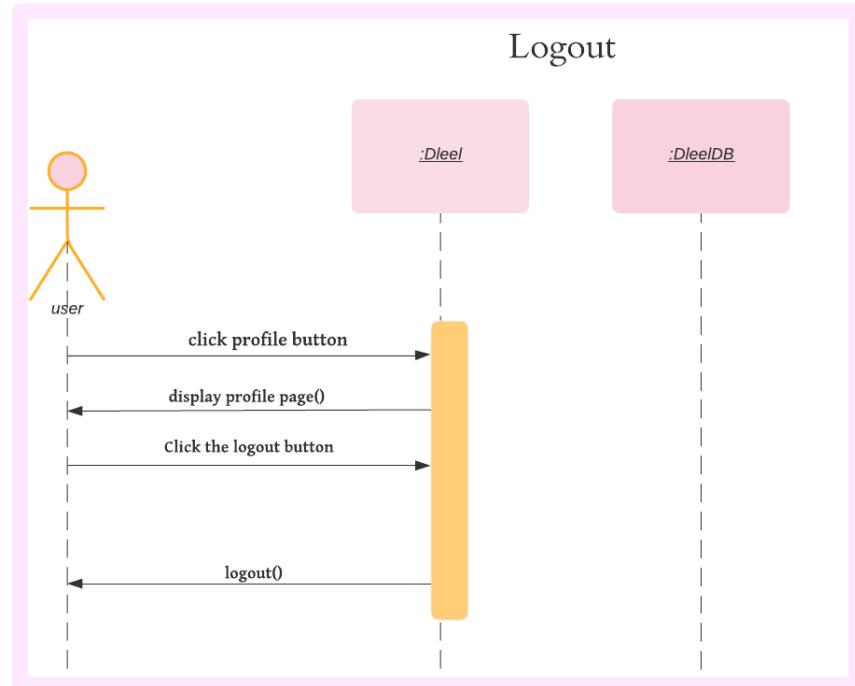


Figure 3- 8-<Logout> sequence diagram

Figure 3-8 The user logs out of the program from the profile page.

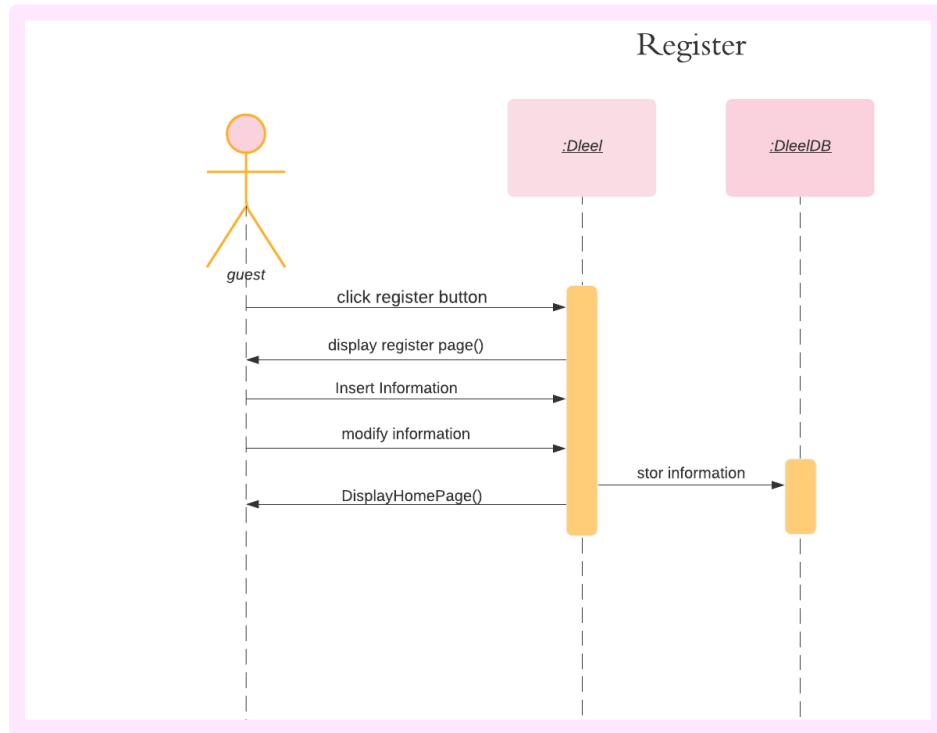


Figure 3- 9-<Register> sequence diagram

As shown in Figure 3-10 the Register sequence diagram starts when the Member presses the registration button then enters the required information including the Username, the password, the phone number and the full name...etc.

3.5 Class diagram

to Form a structural point of view, and to model the structure of our application by showing the classes, their attributes, operation, and the relationships among them. Figure 3-11 shows DLEEL class diagram.

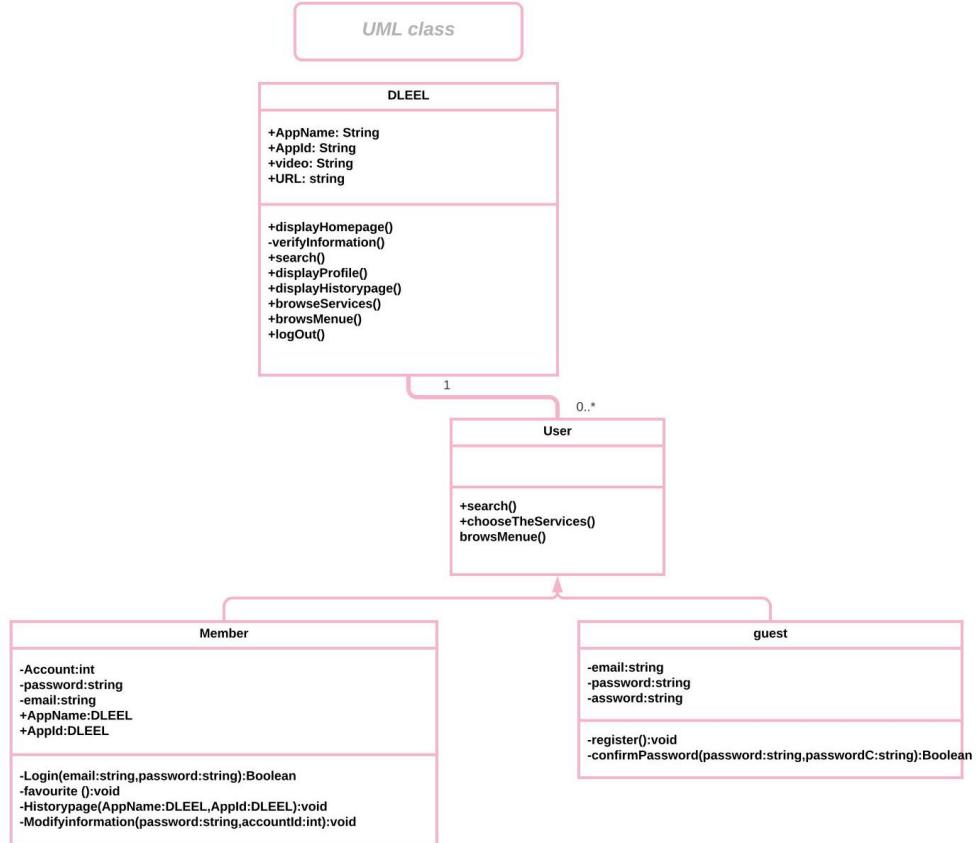


Figure 3- 7-DLEEL- class diagram

Chapter 4 : System design

Chapter 4 System design

In this chapter we will show system architecture.

4.1 System architecture

This chapter focuses on the implementation of DALEEL application. More specifically, Section 5.2 describes the implementation requirements including software and hardware requirements. Section 5.3 depicts the implementation details. Section 5.4 shows the main interfaces' screens of DALEEL. A system architecture is a conceptual model that defines the structural behavior and other views of a system. An architectural description is a formal description and representation of a system, organized in a way that supports reasoning about the structure and behavior of the system.

[9]

-Presentation layer

The presentation layer is the only layer directly connected to the user so in this matter, it is also very important aspect of marketing purpose. The presentation layer is mainly used to obtain user data and then pass it to the business logic layer for further processing. When data is received in a value object, it is responsible for representing the value object in an appropriate form that the user can understand.

-Business logic layer

This is the middle layer-the heart of the application, it contains all business logic of the application, describes how business objects interact with each other, where the presentation layer and data access layer can indirectly communicate with each other.

-Data access layer

The data access layer enforces rules regarding accessing data, providing simplified access to data stored in persistent storage, such as SQL server. It is noteworthy that this layer only focuses on data access instead of data storage. It may create a bit of confusion with the Data-tier in a three-tier architecture

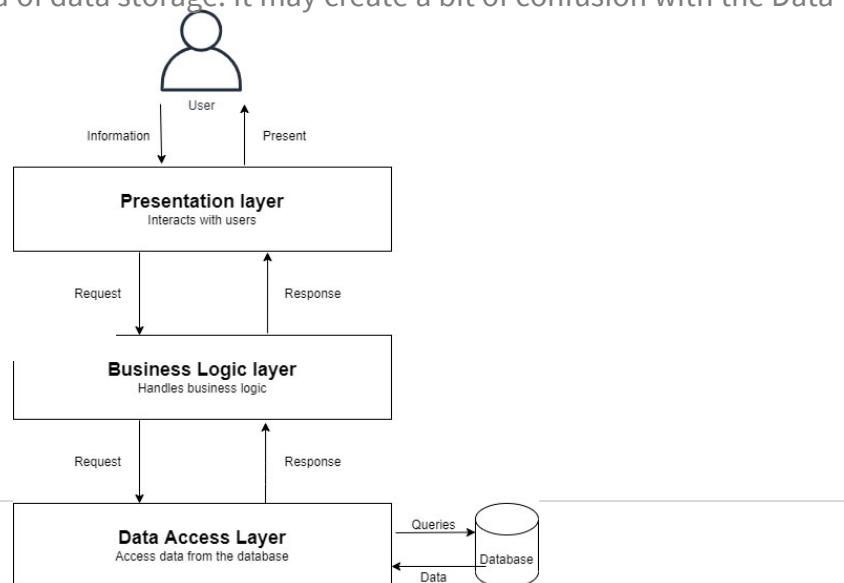


Figure 4- 1-Architecture design

4.2 User interface Design:

In this section, we will expose the main interfaces of DLEEL Application.



Figure 4- 2 -<main page> interface

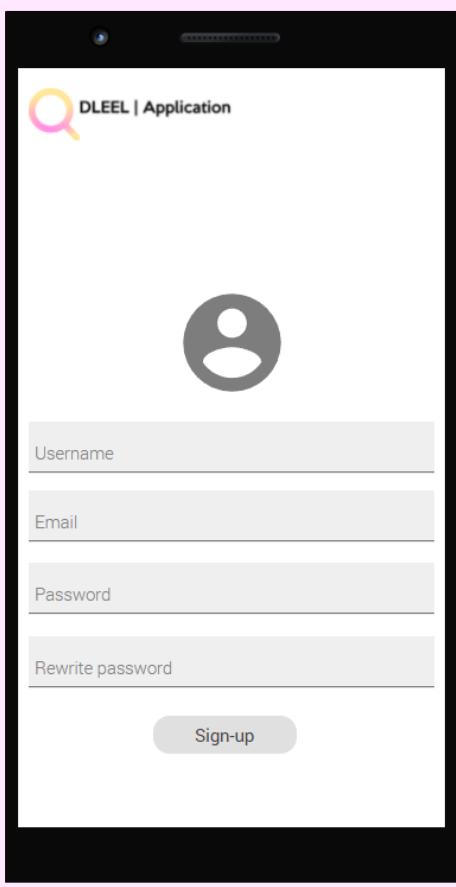


Figure 4- 4-<Sign-up> interface

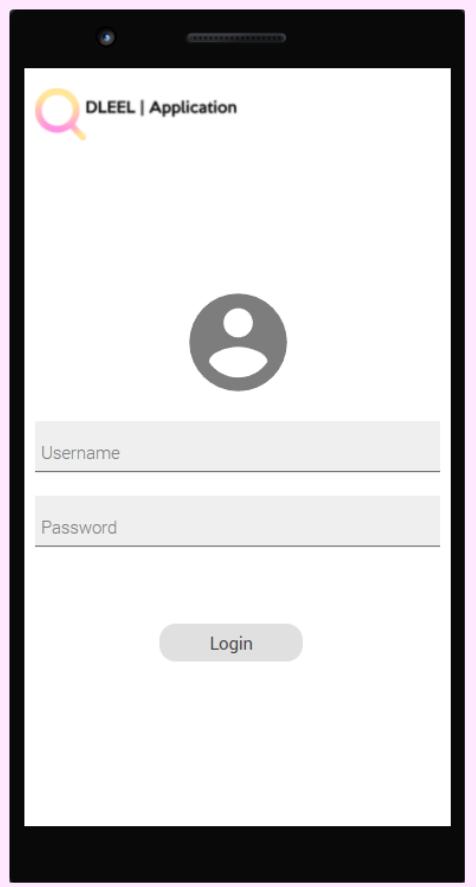


Figure 4- 3 -<Login> interface

If the authentication process successfully performed, the homepage will be displayed and will be provide an access to toolbar for member and guest, as Figure 4-5.



Figure 4- 5 quick access toolbar

- 1- Profile: provide the Language, History page and edit profile.
- 2- Favorite list: will show favorite services to user.
- 3- Menu Services: user can browse services or search for services.
- 4- Homepage: User can check the common application and search.

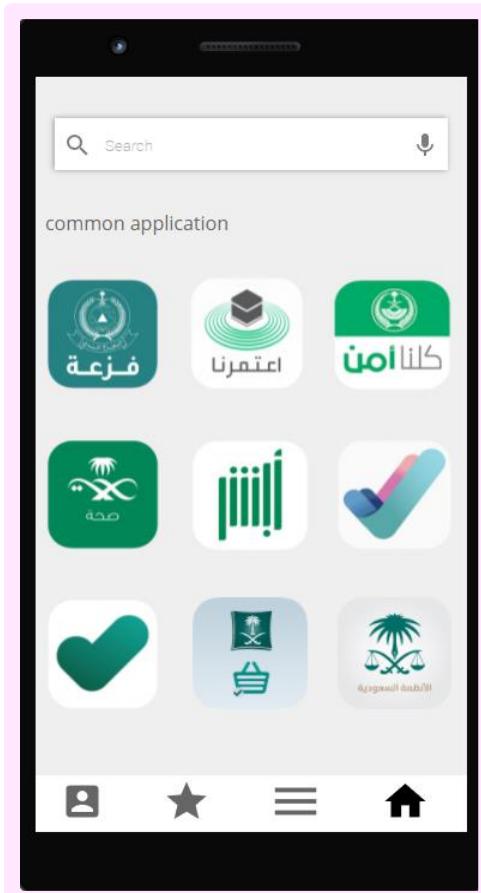


Figure 4- 6-<homepage> interface

Figure 4-6 shows DLEEL home page it contains most important application for most of users.

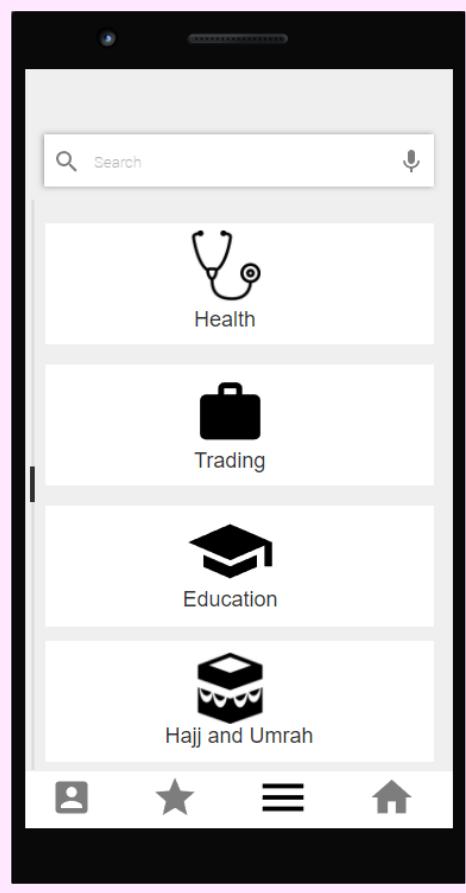


Figure 4- 7-<menu services > interface

Figure 4-7 shows the list of government services departments.

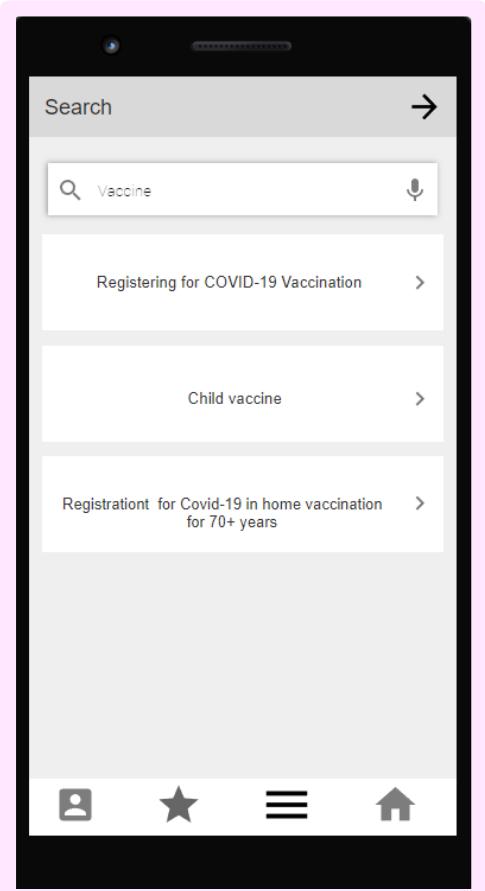


Figure 4- 8-<search> interface

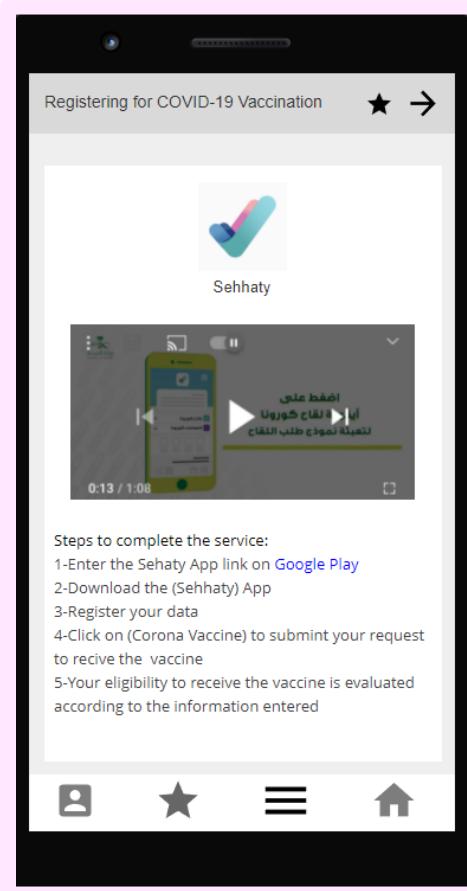


Figure 4- 9-< instruction > interface

Figure 4-9 If the user chooses a service, the application displays details of how to complete the procedures for this service, step by step, in addition to providing an explanatory video.

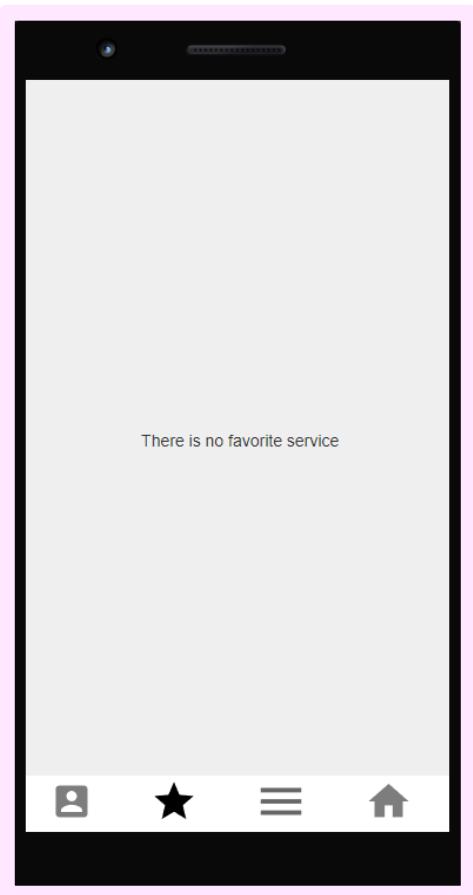


Figure 4- 10-<favorite services for member> interface

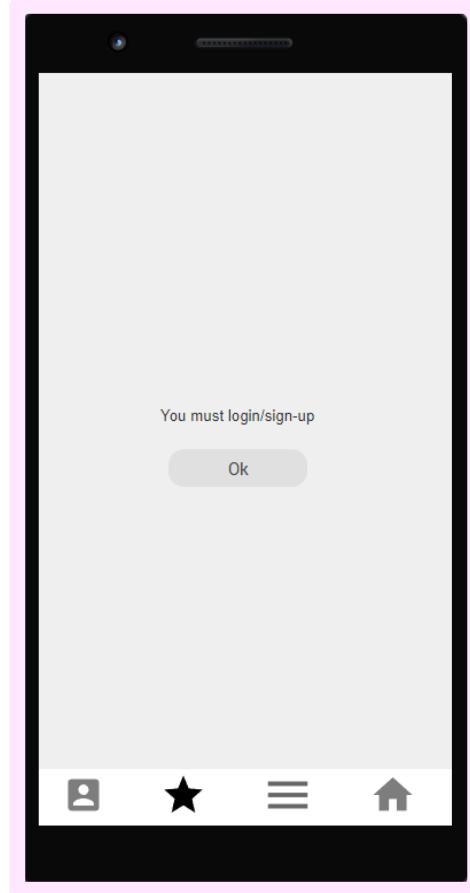


Figure 4- 11-<favorite services for guest> interface

Figure 4-11 the user must login/sign-up to use the feature for accessing preference service faster and easier

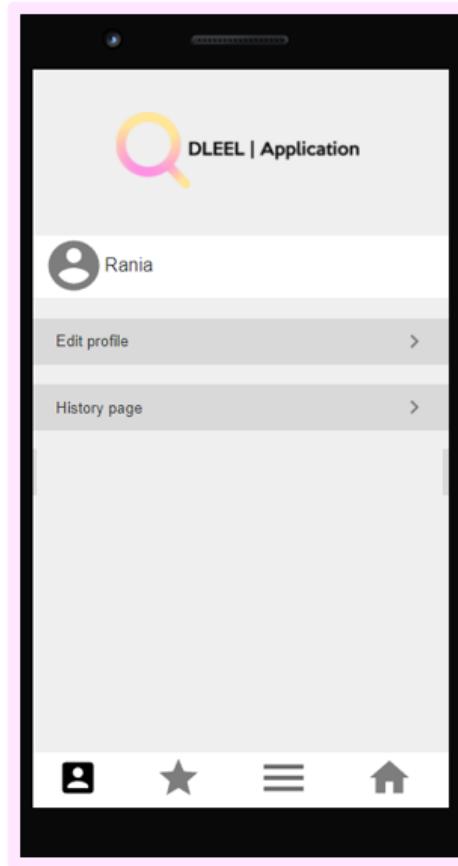


Figure 4- 12-<profile> interface

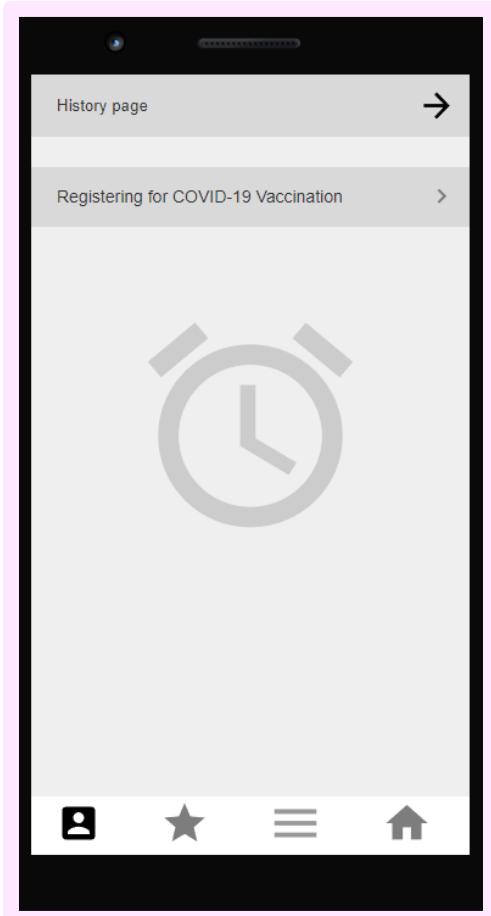


Figure 4- 13-<History page> interface

Figure 4-13 shows the recently searched that the user has searched for.

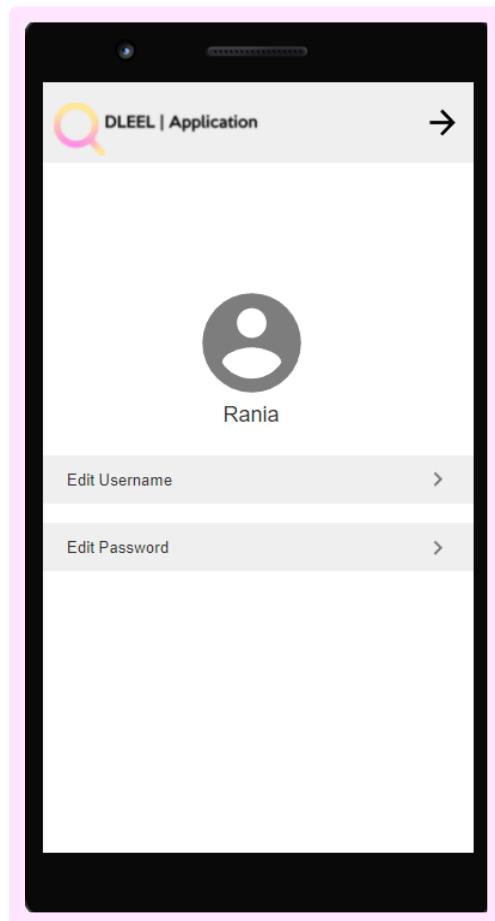


Figure 4-14 The user can edit their Username or Password.

Figure 4- 14-<edit profile> interface

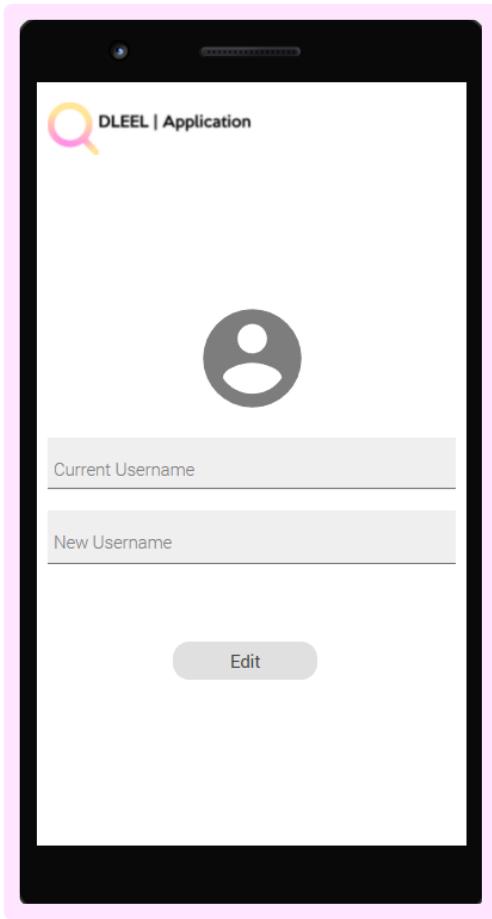


Figure 4- 15-<edit Username> interface

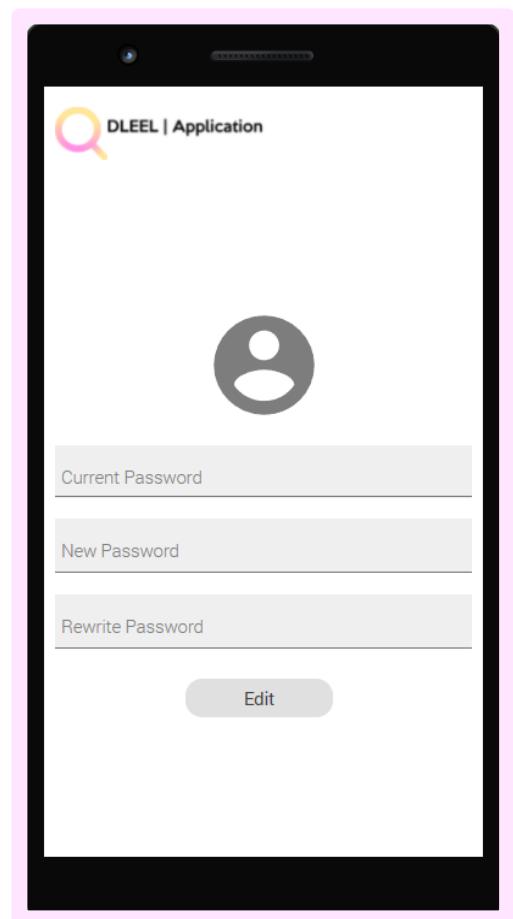


Figure 4- 16-<edit password> interface

Chapter 5 : Implementation

Chapter 5 Implementation

In this chapter, we focused on implementing a DLEEL application. We describe the implementation requirements, including software and hardware, and show the implementation details and the main interfaces of the DLEEL application.

5.1 Implementation Requirements

This section mentioned the implementation requirements for hardware and software

5.2.1.Hardware Requirements.

To implement DLEEL, we used a laptop with HDD 1TB storage, Memory with 16.0 GB storage and Intel core i7-1165G7, up to 2.80GHz CPU. Additionally, we used an android device to install and run DLEEL application

5.2.2. Software Requirements

While implementing a application, we used software. We present it as follows:

- Android Studio:**

Android Studio is an integrated development environment (IDE) from Google that makes it easier for developers to write the source code for Android applications, and also allows the developer to preview the appearance of his application on various screen sizes immediately during development and uses Java and Kotlin

- The Firebase platform :** is a platform launched by Google specialized in creating new applications. The platform contains a suite of applications that contains many services that developers usually have to build themselves, we used it for adding, removing, and modifying information in the database to store, retrieve, and update the information on our database.

5.3 Implementation details

Table 5- 1-<<Login.Java>> Code

```
1 package com.example.login_first;
2
3 import androidx.annotation.NonNull;
4 import androidx.appcompat.app.AppCompatActivity;
5
6 import android.content.Intent;
7 import android.os.Bundle;
8 import android.text.TextUtils;
9 import android.view.View;
10 import android.widget.Button;
11 import android.widget.TextView;
12 import android.widget.Toast;
13
14 import com.google.android.gms.tasks.OnCompleteListener;
15 import com.google.android.gms.tasks.Task;
16 import com.google.android.material.textfield.TextInputEditText;
17 import com.google.firebase.auth.AuthResult;
18 import com.google.firebase.auth.FirebaseAuth;
19
20 public class LoginActivity extends AppCompatActivity {
21
22     TextInputEditText etLoginEmail;
23     TextInputEditText etLoginPassword;
24     TextView tvRegisterHere;
```

```

25     Button btnLogin;
26     Button skipBtn;
27
28     FirebaseAuth mAuth;
29
30     @Override
31     protected void onCreate(Bundle savedInstanceState) {
32         super.onCreate(savedInstanceState);
33         setContentView(R.layout.activity_login);
34
35         etLoginEmail = findViewById(R.id.etLoginEmail);
36         etLoginPassword = findViewById(R.id.etLoginPass);
37         tvRegisterHere = findViewById(R.id.tvRegisterHere);
38         btnLogin = findViewById(R.id.btnLogin);
39         skipBtn = findViewById(R.id.SkipBtn);
40
41         btnLogin.setOnClickListener(view -> {
42             loginUser();
43         });
44         tvRegisterHere.setOnClickListener(view ->{
45             startActivity(new Intent(packageContext: LoginActivity.this, RegisterActivity.class));
46         });
47
48         skipBtn.setOnClickListener(new View.OnClickListener() {
49
50             skipBtn.setOnClickListener(new View.OnClickListener() {
51                 @Override
52                 public void onClick(View view) {
53                     startActivity(new Intent(packageContext: LoginActivity.this, MainActivity.class));
54                 }
55             });
56         });
57     private void loginUser(){
58         FirebaseAuth mAuth = FirebaseAuth.getInstance();
59         String email = etLoginEmail.getText().toString();
60         String password = etLoginPassword.getText().toString();
61
62         if (TextUtils.isEmpty(email)){
63             etLoginEmail.setError("Email cannot be empty");
64             etLoginEmail.requestFocus();
65         }else if (TextUtils.isEmpty(password)){
66             etLoginPassword.setError("Password cannot be empty");
67             etLoginPassword.requestFocus();
68         }else{
69             mAuth.signInWithEmailAndPassword(email,password).addOnCompleteListener(new OnCompleteListener<AuthResult>() {
70
71                 @Override
72                 public void onComplete(@NonNull Task<AuthResult> task) {
73                     if (task.isSuccessful()){
74                         Toast.makeText(context: LoginActivity.this, text: "User logged in successfully", Toast.LENGTH_SHORT).show();
75                         startActivity(new Intent(packageContext: LoginActivity.this, MainActivity.class));
76                     }else{
77                         Toast.makeText(context: LoginActivity.this, text: "Log in Error: " + task.getException().getMessage(), Toast.LENGTH_SHORT).show();
78                     }
79                 }
80             });
81         }
82     }

```

Table 5- 2-<<Login.Xml>> Code

```
1  <?xml version="1.0" encoding="utf-8"?>
2  <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
3      xmlns:app="http://schemas.android.com/apk/res-auto"
4      xmlns:tools="http://schemas.android.com/tools"
5      android:layout_width="match_parent"
6      android:layout_height="match_parent"
7      android:orientation="vertical"
8      android:padding="24dp"
9      tools:context=".LoginActivity">
10
11     <ImageView
12         android:id="@+id/imageView"
13         android:layout_width="200dp"
14         android:layout_height="200dp"
15         android:layout_gravity="center"
16         android:layout_marginTop="25dp"
17         android:src="@drawable/welcoming_img"/>
18
19     <LinearLayout
20         android:layout_width="match_parent"
21         android:layout_height="match_parent"
22         android:orientation="vertical"
23         android:layout_gravity="center"
24         android:layout_marginTop="25dp">
25
26         <TextView
27             android:layout_width="match_parent"
28             android:layout_height="wrap_content"
29             android:text="Login"
30             android:textSize="30sp"
31             android:textStyle="bold"
32             android:textColor="#D88BD1"
33             android:gravity="center_horizontal"
34             android:layout_marginVertical="16dp"/>
35
36
```

```
37     <com.google.android.material.textfield.TextInputLayout  
38         android:layout_width="match_parent"  
39         style="@style/Widget.MaterialComponents.TextInputLayout.OutlinedBox"  
40         android:layout_height="wrap_content">  
41  
42         <com.google.android.material.textfield.TextInputEditText  
43             android:id="@+id/etLoginEmail"  
44             android:layout_width="match_parent"  
45             android:layout_height="wrap_content"  
46             android:hint="Email"  
47             android:drawableStart="@drawable/ic_email"  
48             android:drawablePadding="10dp"  
49             android:inputType="textEmailAddress"/>  
50     </com.google.android.material.textfield.TextInputLayout>  
51  
52 
```

```
53     <com.google.android.material.textfield.TextInputLayout  
54         android:layout_width="match_parent"  
55         style="@style/Widget.MaterialComponents.TextInputLayout.OutlinedBox"  
56         android:layout_height="wrap_content"  
57         app:passwordToggleEnabled="true"  
58         android:layout_marginTop="12dp">  
59  
60         <com.google.android.material.textfield.TextInputEditText  
61             android:id="@+id/etLoginPass"  
62             android:layout_width="match_parent"  
63             android:layout_height="wrap_content"  
64             android:hint="Password"  
65             android:drawableStart="@drawable/ic_lock"  
66             android:drawablePadding="10dp"  
67             android:inputType="textPassword"/>  
68     </com.google.android.material.textfield.TextInputLayout>  
69 
```

```
70 |     <com.google.android.material.button.MaterialButton  
71 |         android:id="@+id/btnLogin"  
72 |         android:layout_width="match_parent"  
73 |         android:layout_height="wrap_content"  
74 |         android:layout_marginTop="24dp"  
75 |         android:text="Login"  
76 |         android:backgroundTint="#B3B3B3"  
77 |         android:textColor="#4A4A4A"  
78 |     />  
79 |  
80 |     <LinearLayout  
81 |         android:layout_width="match_parent"  
82 |         android:layout_height="wrap_content"  
83 |         android:layout_marginTop="15dp"  
84 |         android:gravity="center"  
85 |         android:orientation="horizontal">  
86 |
```

```
87 |     <TextView  
88 |         android:layout_width="wrap_content"  
89 |         android:layout_height="wrap_content"  
90 |         android:text="Not registered yet"  
91 |         android:textSize="16sp"/>  
92 |  
93 |     <TextView  
94 |         android:id="@+id/tvRegisterHere"  
95 |         android:layout_width="wrap_content"  
96 |         android:layout_height="wrap_content"  
97 |         android:layout_marginLeft="8dp"  
98 |         android:text="Register here"  
99 |         android:textColor="#D88BD1"  
100 |         android:textStyle="bold"  
101 |         android:textSize="20sp"/>  
102 |  
103 |     </LinearLayout>  
104 |
```

```
105     <Button  
106         android:id="@+id/SkipBtn"  
107         android:layout_width="match_parent"  
108         android:layout_height="wrap_content"  
109         android:text="As a Guest"  
110         android:layout_marginTop="24dp"  
111         android:backgroundTint="#B3B3B3"  
112         android:textColor="#4A4A4A"/>  
113     </LinearLayout>  
114 </LinearLayout>  
115 </LinearLayout>
```

Table 5- 3-<>Register.Java<>Code

```
1 package com.example.login_first;  
2  
3 import androidx.annotation.NonNull;  
4 import androidx.appcompat.app.AppCompatActivity;  
5  
6 import android.content.Intent;  
7 import android.os.Bundle;  
8 import android.text.TextUtils;  
9 import android.widget.Button;  
10 import android.widget.EditText;  
11 import android.widget.Toast;  
12  
13 import com.google.android.gms.tasks.OnCompleteListener;  
14 import com.google.android.gms.tasks.Task;  
15 import com.google.android.material.textfield.TextInputEditText;  
16 import com.google.firebase.auth.AuthResult;  
17 import com.google.firebase.auth.FirebaseAuth;  
18  
19 public class RegisterActivity extends AppCompatActivity {  
20  
21     TextInputEditText etRegEmail;  
22     TextInputEditText etRegPassword;  
23     TextInputEditText etRegCPassword;  
24     TextView tvLoginHere;
```

```
25     Button btnRegister;
26
27     FirebaseAuth mAuth;
28
29     @Override
30     protected void onCreate(Bundle savedInstanceState) {
31         super.onCreate(savedInstanceState);
32         setContentView(R.layout.activity_register);
33
34         etRegEmail = findViewById(R.id.etRegEmail);
35         etRegPassword = findViewById(R.id.etRegPass);
36         etRegCPassword = findViewById(R.id.etRegCPass);
37         tvLoginHere = findViewById(R.id.tvLoginHere);
38         btnRegister = findViewById(R.id.btnRegister);
39
40         mAuth = FirebaseAuth.getInstance();
41
42         btnRegister.setOnClickListener(view ->{
43             createUser();
44         });
45
46         tvLoginHere.setOnClickListener(view ->{
47             startActivity(new Intent(getApplicationContext(), RegisterActivity.this, LoginActivity.class));
48         });
49     }

```

```
50
51     private void createUser(){
52         String email = etRegEmail.getText().toString();
53         String password = etRegPassword.getText().toString();
54         String cpassword = etRegCPassword.getText().toString();
55
56         if (TextUtils.isEmpty(email)){
57             etRegEmail.setError("Email can't be empty");
58             etRegEmail.requestFocus();
59         }else if (TextUtils.isEmpty(password)){
60             etRegPassword.setError("Password can't be empty");
61             etRegPassword.requestFocus();
62         }else if (TextUtils.isEmpty(cpassword)){
63             etRegCPassword.setError("Confirm Password can't be empty");
64             etRegCPassword.requestFocus();
65         }else if (!password.equals(cpassword)){
66             etRegPassword.setError("Password doesn't match");
67             etRegCPassword.requestFocus();
68         }else{
69             mAuth.createUserWithEmailAndPassword(email, password).addOnCompleteListener(new OnCompleteListener<AuthResult>() {

```

```
70             @Override
71             public void onComplete(@NonNull Task<AuthResult> task) {
72                 if (task.isSuccessful()){
73                     Toast.makeText(context, "User registered successfully", Toast.LENGTH_SHORT).show();
74                     startActivity(new Intent(getApplicationContext(), RegisterActivity.this, LoginActivity.class));
75                 }else{
76                     Toast.makeText(context, "Registration Error: " + task.getException().getMessage(), Toast.LENGTH_SHORT).show();
77                 }
78             }
79         };
80     }
81
82 }
```

Table 5- 4-<>Register.Xml>>Code

```
1  <?xml version="1.0" encoding="utf-8"?>
2  <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
3      xmlns:app="http://schemas.android.com/apk/res-auto"
4      xmlns:tools="http://schemas.android.com/tools"
5      android:layout_width="match_parent"
6      android:layout_height="match_parent"
7      android:orientation="vertical"
8      android:padding="24dp"
9      tools:context=".RegisterActivity">
10
11     <ImageView
12         android:id="@+id/imageView"
13         android:layout_width="200dp"
14         android:layout_height="200dp"
15         android:layout_gravity="center"
16         android:layout_marginTop="25dp"
17         android:src="@drawable/welcoming_img"/>
18
19
20     <LinearLayout
21         android:layout_width="match_parent"
22         android:layout_height="match_parent"
23         android:orientation="vertical"
24         android:layout_gravity="center"
25         android:layout_marginTop="25dp">
26
27         <TextView
28             android:layout_width="match_parent"
29             android:layout_height="wrap_content"
30             android:text="Register"
31             android:textSize="30sp"
32             android:textStyle="bold"
33             android:textColor="#D88BD1"
34             android:gravity="center_horizontal"
35             android:layout_marginVertical="16dp"/>
36
```

```
37     <com.google.android.material.textfield.TextInputLayout
38         android:layout_width="match_parent"
39         style="@style/Widget.MaterialComponents.TextInputLayout.OutlinedBox"
40         android:layout_height="wrap_content">
41
42         <com.google.android.material.textfield.TextInputEditText
43             android:id="@+id/etRegEmail"
44             android:layout_width="match_parent"
45             android:layout_height="wrap_content"
46             android:hint="Email"
47             android:drawableStart="@drawable/ic_email"
48             android:drawablePadding="10dp"
49             android:inputType="textEmailAddress"/>
50     </com.google.android.material.textfield.TextInputLayout>
51
52
53     <com.google.android.material.textfield.TextInputLayout
54         android:layout_width="match_parent"
55         style="@style/Widget.MaterialComponents.TextInputLayout.OutlinedBox"
56         android:layout_height="wrap_content"
57         app:passwordToggleEnabled="true"
58         android:layout_marginTop="12dp">
```

```
59
60
61         <com.google.android.material.textfield.TextInputEditText
62             android:id="@+id/etRegPass"
63             android:layout_width="match_parent"
64             android:layout_height="wrap_content"
65             android:hint="Password"
66             android:drawableStart="@drawable/ic_lock"
67             android:drawablePadding="10dp"
68             android:inputType="textPassword"/>
69     </com.google.android.material.textfield.TextInputLayout>
70
71     <com.google.android.material.textfield.TextInputLayout
72         android:layout_width="match_parent"
73         style="@style/Widget.MaterialComponents.TextInputLayout.OutlinedBox"
74         android:layout_height="wrap_content"
75         app:passwordToggleEnabled="true"
76         android:layout_marginTop="12dp">
```

```
79          <com.google.android.material.textfield.TextInputEditText  
80              android:id="@+id/etRegCPass"  
81              android:layout_width="match_parent"  
82              android:layout_height="wrap_content"  
83              android:hint="Confirm Password"  
84              android:drawableStart="@drawable/ic_lock"  
85              android:drawablePadding="10dp"  
86              android:inputType="textPassword"/>  
87      </com.google.android.material.textfield.TextInputLayout>  
88  
89      <com.google.android.material.button.MaterialButton  
90          android:id="@+id/btnRegister"  
91          android:layout_width="match_parent"  
92          android:layout_height="wrap_content"  
93          android:layout_marginTop="24dp"  
94          android:text="Register"  
95          android:backgroundTint="#B3B3B3"  
96          android:textColor="#4A4A4A"/>  
97
```

```
98          <LinearLayout  
99              android:layout_width="match_parent"  
100             android:layout_height="wrap_content"  
101             android:layout_marginTop="15dp"  
102             android:gravity="center"  
103             android:orientation="horizontal">  
104  
105             <TextView  
106                 android:layout_width="wrap_content"  
107                 android:layout_height="wrap_content"  
108                 android:text="Already registered"  
109                 android:textSize="16sp"/>  
110  
111             <TextView  
112                 android:id="@+id/tvLoginHere"  
113                 android:layout_width="wrap_content"  
114                 android:layout_height="wrap_content"  
115                 android:layout_marginLeft="8dp"  
116                 android:text="Login here"  
117                 android:textColor="#D88BD1"  
118                 android:textStyle="bold"  
119                 android:textSize="20sp"/>  
120         </LinearLayout>  
121     </LinearLayout>  
122  
123 </LinearLayout>  
124  
125 </LinearLayout>
```

Table 5- 5-<<main.java>>Code

```
1 package com.example.login_first;
2
3 import androidx.appcompat.app.AppCompatActivity;
4
5 import android.content.Intent;
6 import android.net.Uri;
7 import android.os.Bundle;
8 import android.view.View;
9 import android.widget.ImageButton;
10
11 import com.google.firebase.auth.FirebaseAuth;
12
13 public class MainActivity extends AppCompatActivity {
14     public ImageButton Menu_home;
15     public ImageButton Star_home;
16     public ImageButton Profile_home;
17     public ImageButton Home_home;
18
19     ImageButton tawklna;
20     ImageButton sahaty;
21     ImageButton Etmrna;
22     ImageButton absher;
23     ImageButton fazaa;
24     ImageButton amn;
25     ImageButton seha;
26
27     ImageButton balag;
28     ImageButton andhma;
29     ImageButton balady;
30     ImageButton asafny;
31     ImageButton qayas;
32
33     FirebaseAuth mAuth;
34
35     @Override
36     protected void onCreate(Bundle savedInstanceState) {
37         super.onCreate(savedInstanceState);
38         setContentView(R.layout.activity_main);
39
40         mAuth = FirebaseAuth.getInstance();
41
42         tawklna=findViewById(R.id.tawklna);
43         sahaty=findViewById(R.id.sahaty);
44         Etmrna=findViewById(R.id.Etmrna);
45         absher=findViewById(R.id.absher);
46         fazaa=findViewById(R.id.fazaa);
47         amn=findViewById(R.id.amn);
48         seha=findViewById(R.id.seha);
49         balag=findViewById(R.id.balag);
50         andhma=findViewById(R.id.andhma);
51         balady=findViewById(R.id.balady);
```

```

51         asafny=findViewById(R.id.asafny);
52         qayas=findViewById(R.id.qayas);
53
54
55
56         Menu_home=(ImageButton) findViewById(R.id.menu_btn_home);
57         Menu_home.setOnClickListener(new View.OnClickListener() {
58             @Override
59             public void onClick(View view) {
60                 Intent intent=new Intent( packageContext: MainActivity.this,MenuActivity.class);
61                 startActivity(intent);
62             }
63         });
64
65         Star_home=(ImageButton) findViewById(R.id.star_btn_home);
66         Star_home.setOnClickListener(new View.OnClickListener() {
67             @Override
68             public void onClick(View view) {
69                 if(mAuth.getCurrentUser()==null){
70                     Intent intent=new Intent( packageContext: MainActivity.this,GuestProfile.class);
71                     startActivity(intent);
72                 }
73                 else
74                     {Intent intent=new Intent( packageContext: MainActivity.this,FavoriteActivity.class);
75                     startActivity(intent);};
76             }
77         });
78
79         //skip
80         Profile_home=(ImageButton) findViewById(R.id.profile_btn_home);
81         Profile_home.setOnClickListener(new View.OnClickListener() {
82             @Override
83             public void onClick(View view) {
84                 if(mAuth.getCurrentUser()==null){
85                     Intent intent=new Intent( packageContext: MainActivity.this,GuestProfile.class);
86                     startActivity(intent);
87                 }
88                 else
89                     {Intent intent=new Intent( packageContext: MainActivity.this,ProfileActivity.class);
90                     startActivity(intent);};
91             }
92         });
93         tawkklna.setOnClickListener(new View.OnClickListener() {
94             @Override
95             public void onClick(View view) {
96                 gotoUrl( s: "https://play.google.com/store/apps/details?id=sa.gov.nic.tawakkalna&hl=ar&gl=sa");
97             }
98         });
99         sahaty.setOnClickListener(new View.OnClickListener() {
100            @Override

```

```
101 ❶↑    public void onClick(View view) {
102      |     gotoUrl( s: "https://play.google.com/store/apps/details?id=com.lean.sehhaty&hl=ar&gl=sa");
103      |
104      }});
105 ❷↑    Etmrna.setOnClickListener(new View.OnClickListener() {
106      @Override
107 ❸↑    public void onClick(View view) {
108          |     gotoUrl( s: "https://play.google.com/store/apps/details?id=com.sejel.eatamrna&hl=ar&gl=sa");
109          |
110      });
111 ❹↑    absher.setOnClickListener(new View.OnClickListener() {
112      @Override
113 ❺↑    public void onClick(View view) {
114          |     gotoUrl( s: "https://play.google.com/store/apps/details?id=sa.gov.moi&hl=ar&gl=sa");
115          |
116      });
117 ❻↑    fazaa.setOnClickListener(new View.OnClickListener() {
118      @Override
119 ❻↑    public void onClick(View view) {
120          |     gotoUrl( s: "https://play.google.com/store/apps/details?id=sa.cd.faza&hl=ar&gl=sa");
121          |
122      });
123 ❼↑    amn.setOnClickListener(new View.OnClickListener() {
124      @Override
125
126 ❼↑    public void onClick(View view) {
127          |     gotoUrl( s: "https://play.google.com/store/apps/details?id=sa.gov.moi.securityinform&hl=ar&gl=sa");
128          |
129      });
130
131 ❼↑    seha.setOnClickListener(new View.OnClickListener() {
132      @Override
133 ❼↑    public void onClick(View view) {
134          |     gotoUrl( s: "https://play.google.com/store/apps/details?id=com.moh.mohformulary&hl=ar&gl=sa");
135          |
136      });
137
138 ❼↑    });
139
140 ❼↑    balagh.setOnClickListener(new View.OnClickListener() {
141      @Override
142 ❼↑    public void onClick(View view) {
143          |     gotoUrl( s: "https://play.google.com/store/apps/details?id=com.mci.balagh&hl=ar&gl=sa");
144          |
145      });
146
147 ❼↑    andhma.setOnClickListener(new View.OnClickListener() {
148      @Override
149 ❼↑    public void onClick(View view) {
150          |     gotoUrl( s: "https://play.google.com/store/apps/details?id=mobi.bip_sa.bipguid&hl=ar&gl=sa");
```

```
151     }
152     });
153
154     balady.setOnClickListener(new View.OnClickListener() {
155         @Override
156         public void onClick(View view) {
157             gotoUrl("https://play.google.com/store/apps/details?id=momra.balady.android&hl=ar&gl=sa");
158         }
159     });
160
161     asafny.setOnClickListener(new View.OnClickListener() {
162         @Override
163         public void onClick(View view) {
164             gotoUrl("https://play.google.com/store/apps/details?id=com.srca.apps.sos&hl=ar&gl=sa");
165         }
166     });
167
168     qayas.setOnClickListener(new View.OnClickListener() {
169         @Override
170         public void onClick(View view) {
171             gotoUrl("https://play.google.com/store/apps/details?id=com.alho00ot.qyas&hl=ar&gl=sa");
172         }
173     });
174 }
175
176 private void gotoUrl(String s) {
177     Uri uri=Uri.parse(s);
178     startActivity(new Intent(Intent.ACTION_VIEW,uri));
179 }
```

Table 5- 6-<<main.Xml>> Code

```
1  <?xml version="1.0" encoding="utf-8"?>
2  <androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
3      xmlns:app="http://schemas.android.com/apk/res-auto"
4      xmlns:tools="http://schemas.android.com/tools"
5      android:layout_width="match_parent"
6      android:layout_height="match_parent"
7      android:orientation="horizontal"
8      tools:context=".MainActivity"
9      >
10
11
12      <TextView
13          android:id="@+id/textView"
14          android:layout_width="wrap_content"
15          android:layout_height="wrap_content"
16          android:fontFamily="sans-serif-condensed-medium"
17          android:gravity="left"
18          android:text="Common Applications"
19          android:textColor="@color/black"
20          android:textSize="24sp"
21          android:textStyle="bold"
22          app:layout_constraintBottom_toBottomOf="parent"
23          app:layout_constraintEnd_toEndOf="parent"
24
25
26
27          app:layout_constraintHorizontal_bias="0.077"
28          app:layout_constraintStart_toStartOf="parent"
29          app:layout_constraintTop_toTopOf="parent"
30          app:layout_constraintVertical_bias="0.241" />
31
32
33      <ImageButton
34          android:id="@+id/tawklna"
35          android:layout_width="wrap_content"
36          android:layout_height="wrap_content"
37          android:background="#A6CAC7C7"
38          app:layout_constraintBottom_toBottomOf="parent"
39          app:layout_constraintEnd_toEndOf="parent"
40          app:layout_constraintHorizontal_bias="0.048"
41          app:layout_constraintStart_toStartOf="parent"
42          app:layout_constraintTop_toTopOf="parent"
43
44          app:layout_constraintVertical_bias="0.446"
45          app:srcCompat="@drawable/tawklna" />
46
47      <ImageButton
48          android:id="@+id/sahaty"
49          android:layout_width="wrap_content"
```

```
47     android:layout_height="wrap_content"
48     android:background="#A6CAC7C7"
49     app:layout_constraintBottom_toBottomOf="parent"
50     app:layout_constraintEnd_toEndOf="parent"
51     app:layout_constraintHorizontal_bias="0.355"
52     app:layout_constraintStart_toStartOf="parent"
53     app:layout_constraintTop_toTopOf="parent"
54     app:layout_constraintVertical_bias="0.446"
55     app:srcCompat="@drawable/sehaty" />
56
57     <ImageButton
58         android:id="@+id/Etmrna"
59         android:layout_width="wrap_content"
60         android:layout_height="wrap_content"
61         android:background="#A6CAC7C7"
62         app:layout_constraintBottom_toBottomOf="parent"
63         app:layout_constraintEnd_toEndOf="parent"
64         app:layout_constraintHorizontal_bias="0.647"
65         app:layout_constraintStart_toStartOf="parent"
66         app:layout_constraintTop_toTopOf="parent"
67         app:layout_constraintVertical_bias="0.446"
68         app:srcCompat="@drawable/eatmarna" />
69
70     <ImageButton
71         android:id="@+id/absher"
72         android:layout_width="wrap_content"
73         android:layout_height="wrap_content"
74         android:background="@color/white"
75         app:layout_constraintBottom_toBottomOf="parent"
76         app:layout_constraintEnd_toEndOf="parent"
77         app:layout_constraintHorizontal_bias="0.951"
78         app:layout_constraintStart_toStartOf="parent"
79         app:layout_constraintTop_toTopOf="parent"
80         app:layout_constraintVertical_bias="0.446"
81         app:srcCompat="@drawable/absher01" />
82
83     <ImageButton
84         android:id="@+id/fazaa"
85         android:layout_width="wrap_content"
86         android:layout_height="wrap_content"
87         android:background="#A6CAC7C7"
88         app:layout_constraintBottom_toBottomOf="parent"
89         app:layout_constraintEnd_toEndOf="parent"
90         app:layout_constraintHorizontal_bias="0.048"
91         app:layout_constraintStart_toStartOf="parent"
92         app:layout_constraintTop_toTopOf="parent"
```

```
93         app:layout_constraintVertical_bias="0.619"
94         app:srcCompat="@drawable/fazaa" />
95
96     <ImageButton
97         android:id="@+id/amn"
98         android:layout_width="wrap_content"
99         android:layout_height="wrap_content"
100        android:background="#A6CAC7C7"
101        app:layout_constraintBottom_toBottomOf="parent"
102        app:layout_constraintEnd_toEndOf="parent"
103        app:layout_constraintHorizontal_bias="0.355"
104        app:layout_constraintStart_toStartOf="parent"
105        app:layout_constraintTop_toTopOf="parent"
106        app:layout_constraintVertical_bias="0.619"
107        app:srcCompat="@drawable/amn" />
108
109    <ImageButton
110        android:id="@+id/seha"
111        android:layout_width="wrap_content"
112        android:layout_height="wrap_content"
113        android:background="#A6CAC7C7"
114        app:layout_constraintBottom_toBottomOf="parent"
115        app:layout_constraintEnd_toEndOf="parent"
116
117        app:layout_constraintHorizontal_bias="0.647"
118        app:layout_constraintStart_toStartOf="parent"
119        app:layout_constraintTop_toTopOf="parent"
120        app:layout_constraintVertical_bias="0.619"
121        app:srcCompat="@drawable/seha" />
122
123    <ImageButton
124        android:id="@+id/balag"
125        android:layout_width="wrap_content"
126        android:layout_height="wrap_content"
127        android:background="#A6CAC7C7"
128        app:layout_constraintBottom_toBottomOf="parent"
129        app:layout_constraintEnd_toEndOf="parent"
130        app:layout_constraintHorizontal_bias="0.951"
131        app:layout_constraintStart_toStartOf="parent"
132        app:layout_constraintTop_toTopOf="parent"
133        app:layout_constraintVertical_bias="0.619"
134        app:srcCompat="@drawable/balag" />
135
136    <ImageButton
137        android:id="@+id/andhma"
138        android:layout_width="wrap_content"
139        android:layout_height="wrap_content"
```

```
139     android:background="#A6CAC7C7"
140     app:layout_constraintBottom_toBottomOf="parent"
141     app:layout_constraintEnd_toEndOf="parent"
142     app:layout_constraintHorizontal_bias="0.048"
143     app:layout_constraintStart_toStartOf="parent"
144     app:layout_constraintTop_toTopOf="parent"
145     app:layout_constraintVertical_bias="0.787"
146     app:srcCompat="@drawable/andhma" />
147
148     <ImageButton
149         android:id="@+id/balady"
150         android:layout_width="wrap_content"
151         android:layout_height="wrap_content"
152         android:background="#A6CAC7C7"
153         app:layout_constraintBottom_toBottomOf="parent"
154         app:layout_constraintEnd_toEndOf="parent"
155         app:layout_constraintHorizontal_bias="0.355"
156         app:layout_constraintStart_toStartOf="parent"
157         app:layout_constraintTop_toTopOf="parent"
158         app:layout_constraintVertical_bias="0.781"
159         app:srcCompat="@drawable/balady" />
160
161     <ImageButton
162         android:id="@+id/asafny"
163         android:layout_width="wrap_content"
164         android:layout_height="wrap_content"
165         android:background="#A6CAC7C7"
166         android:backgroundTint="@color/white"
167         app:layout_constraintBottom_toBottomOf="parent"
168         app:layout_constraintEnd_toEndOf="parent"
169         app:layout_constraintHorizontal_bias="0.647"
170         app:layout_constraintStart_toStartOf="parent"
171         app:layout_constraintTop_toTopOf="parent"
172         app:layout_constraintVertical_bias="0.781"
173         app:srcCompat="@drawable/asafny" />
174
175     <ImageButton
176         android:id="@+id/gayas"
177         android:layout_width="wrap_content"
178         android:layout_height="wrap_content"
179         android:background="#A6CAC7C7"
180         app:layout_constraintBottom_toBottomOf="parent"
181         app:layout_constraintEnd_toEndOf="parent"
182         app:layout_constraintHorizontal_bias="0.954"
183         app:layout_constraintStart_toStartOf="parent"
184         app:layout_constraintTop_toTopOf="parent"
```

```
185     app:layout_constraintVertical_bias="0.781"
186     app:srcCompat="@drawable/qayas" />
187
188     <TextView
189         android:id="@+id/textView2"
190         android:layout_width="wrap_content"
191         android:layout_height="wrap_content"
192         android:text="Welcome to Oleel application!"
193         android:textAlignment="center"
194         android:textColor="@color/black"
195         android:textSize="34sp"
196         android:textStyle="bold"
197         app:layout_constraintBottom_toBottomOf="parent"
198         app:layout_constraintEnd_toEndOf="parent"
199         app:layout_constraintHorizontal_bias="0.0"
200         app:layout_constraintStart_toStartOf="parent"
201         app:layout_constraintTop_toTopOf="parent"
202         app:layout_constraintVertical_bias="0.091" />
203
204     <LinearLayout
205         android:id="@+id/homeNavBar"
206         android:layout_width="match_parent"
207
208         android:layout_height="70dp"
209         android:background="#FFFFFF"
210         android:orientation="horizontal"
211         app:layout_constraintBottom_toBottomOf="parent"
212         app:layout_constraintEnd_toEndOf="parent"
213         app:layout_constraintHorizontal_bias="1.0"
214         app:layout_constraintStart_toStartOf="parent"
215         app:layout_constraintTop_toTopOf="parent"
216         app:layout_constraintVertical_bias="1.0">
217             <ImageButton
218                 android:id="@+id/profile_btn_home"
219                 android:layout_width="wrap_content"
220                 android:layout_height="wrap_content"
221                 android:layout_gravity="center"
222                 android:layout_weight="1"
223                 android:backgroundTint="#05FFFFFF"
224                 app:srcCompat="@drawable/ic__person_gray" />
225
226             <ImageButton
227                 android:id="@+id/star_btn_home"
228                 android:layout_width="wrap_content"
229                 android:layout_height="wrap_content"
```

```
231           android:layout_gravity="center"
232           android:layout_weight="1"
233           android:backgroundTint="#05FFFFFF"
234 ★         app:srcCompat="@drawable/ic_star_gray" />
235
236     <ImageButton
237         android:id="@+id/menu_btn_home"
238         android:layout_width="wrap_content"
239         android:layout_height="wrap_content"
240         android:layout_gravity="center"
241         android:layout_weight="1"
242         android:backgroundTint="#05FFFFFF"
243         app:srcCompat="@drawable/ic_menu_gray" />
244
245     <ImageButton
246         android:id="@+id/home_btn_home"
247         android:layout_width="wrap_content"
248         android:layout_height="wrap_content"
249         android:layout_gravity="center"
250         android:layout_weight="1"
251         android:backgroundTint="#05FFFFFF"
252         app:srcCompat="@drawable/ic_home" />
253     </LinearLayout>
254
255 </androidx.constraintlayout.widget.ConstraintLayout>
```

Table 5- 7-<<Profile.Java>> Code

```
1  package com.example.login_first;
2
3  import androidx.annotation.NonNull;
4  import androidx.appcompat.app.AlertDialog;
5  import androidx.appcompat.app.AppCompatActivity;
6
7  import android.content.DialogInterface;
8  import android.content.Intent;
9  import android.os.Bundle;
10 import android.view.View;
11 import android.widget.Button;
12 import android.widget.EditText;
13 import android.widget.ImageButton;
14 import android.widget.Toast;
15
16 import com.google.android.gms.tasks.OnFailureListener;
17 import com.google.android.gms.tasks.OnSuccessListener;
18 import com.google.firebase.auth.FirebaseAuth;
19 import com.google.firebase.auth.FirebaseUser;
20
21 public class ProfileActivity extends AppCompatActivity {
22
23     Button btnLogOut;
24     FirebaseAuth mAuth;
25     Button btredit;
```



```
26     ImageButton Home_profile;
27     ImageButton Star_profile;
28     ImageButton Profile_profile;
29     ImageButton Menu_profile;
30
31     Button resetPassLocal;
32     FirebaseAuth fAuth;
33     FirebaseUser user;
34
35     @Override
36     protected void onCreate(Bundle savedInstanceState) {
37         super.onCreate(savedInstanceState);
38         setContentView(R.layout.activity_profile);
39
40         btnLogOut = findViewById(R.id.btnLogout);
41         mAuth = FirebaseAuth.getInstance();
42         resetPassLocal = findViewById(R.id.resetPasswordLocal);
43         fAuth = FirebaseAuth.getInstance();
44         user = fAuth.getCurrentUser();
45
46
47         btnLogOut.setOnClickListener(view ->{
48             mAuth.signOut();
49             startActivity(new Intent(packageContext: ProfileActivity.this, LoginActivity.class));
50 }
```

```

51     });
52
53     Home_profile=(ImageButton) findViewById(R.id.home_btn_profile);
54     Home_profile.setOnClickListener(new View.OnClickListener() {
55
56         @Override
57         public void onClick(View view) {
58             Intent intent = new Intent ( packageContext: ProfileActivity.this,MainActivity.class);
59             startActivity(intent);
60         }
61     });
62
63     Menu_profile=(ImageButton) findViewById(R.id.menu_btn_profile);
64     Menu_profile.setOnClickListener(new View.OnClickListener() {
65
66         @Override
67         public void onClick(View view) {
68             Intent intent=new Intent( packageContext: ProfileActivity.this,MenuActivity.class);
69             startActivity(intent);
70         }
71     });
72
73     Star_profile=(ImageButton) findViewById(R.id.star_btn_profile);
74     Star_profile.setOnClickListener(new View.OnClickListener() {
75
76         @Override
77         public void onClick(View view) {
78             Intent intent=new Intent( packageContext: ProfileActivity.this,FavoriteActivity.class);
79             startActivity(intent);
80         }
81     });
82
83     resetPassLocal.setOnClickListener(new View.OnClickListener() {
84
85         @Override
86         public void onClick(View v) {
87
88             final EditText resetPassword = new EditText(v.getContext());
89             final AlertDialog.Builder passwordResetDialog = new AlertDialog.Builder(v.getContext());
90             passwordResetDialog.setTitle("Reset Password ?");
91             passwordResetDialog.setMessage("Enter New Password > 6 Characters long.");
92             passwordResetDialog.setView(resetPassword);
93
94             passwordResetDialog.setPositiveButton("Yes", new DialogInterface.OnClickListener() {
95
96                 @Override
97                 public void onClick(DialogInterface dialog, int which) {
98                     // extract the email and send reset link
99                     String newPassword = resetPassword.getText().toString();
100                    user.updatePassword(newPassword).addOnSuccessListener<Void>() {
101
102                         @Override
103                         public void onSuccess(Void aVoid) {
104                             Toast.makeText( context: ProfileActivity.this, text: "Password Reset Successfully.", Toast.LENGTH_SHORT).show();
105                         }
106                     }
107                 }
108             });
109         }
110     });

```

```
101     });
102
103     passwordResetDialog.setOnFailureListener(new OnFailureListener() {
104         @Override
105         public void onFailure(@NonNull Exception e) {
106             Toast.makeText(context: ProfileActivity.this, text: "Password Reset Failed.", Toast.LENGTH_SHORT).show();
107         }
108     });
109
110     passwordResetDialog.setNegativeButton(text: "No", new DialogInterface.OnClickListener() {
111         @Override
112         public void onClick(DialogInterface dialog, int which) {
113             // close
114         }
115     });
116
117     passwordResetDialog.create().show();
118
119     });
120 };
121
122
123     });
124
125     @Override
126     protected void onStart() {
127         super.onStart();
128         FirebaseUser user = mAuth.getCurrentUser();
129         if (user == null){
130             startActivity(new Intent(packageContext: ProfileActivity.this, LoginActivity.class));
131         }
132     }
133
134 }
```

Table 5- 8-<<Profile.Xml>> Code

```
1  <?xml version="1.0" encoding="utf-8"?>
2  <androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
3      xmlns:app="http://schemas.android.com/apk/res-auto"
4      xmlns:tools="http://schemas.android.com/tools"
5      android:layout_width="match_parent"
6      android:layout_height="match_parent"
7      tools:context=".ProfileActivity"
8      >
9
10     <LinearLayout
11         android:id="@+id/ProfileeNavBar"
12         android:layout_width="match_parent"
13         android:layout_height="100dp"
14         android:background="#FFFFFF"
15         android:orientation="horizontal"
16         app:layout_constraintBottom_toBottomOf="parent"
17         app:layout_constraintEnd_toEndOf="parent"
18         app:layout_constraintHorizontal_bias="1.0"
19         app:layout_constraintStart_toStartOf="parent"
20         app:layout_constraintTop_toTopOf="parent"
21         app:layout_constraintVertical_bias="1.0">
22         <ImageButton
23             android:id="@+id/profile_btn_profile"
24
25             android:layout_width="wrap_content"
26             android:layout_height="wrap_content"
27             android:layout_gravity="center"
28             android:layout_weight="1"
29             android:backgroundTint="#05FFFFFF"
30             app:srcCompat="@drawable/ic_person" />
31
32         <ImageButton
33             android:id="@+id/star_btn_profile"
34             android:layout_width="wrap_content"
35             android:layout_height="wrap_content"
36             android:layout_gravity="center"
37             android:layout_weight="1"
38             android:backgroundTint="#05FFFFFF"
39             app:srcCompat="@drawable/ic_star_gray" />
40
41         <ImageButton
42             android:id="@+id/menu_btn_profile"
43             android:layout_width="wrap_content"
44             android:layout_height="wrap_content"
45             android:layout_gravity="center"
46             android:layout_weight="1"
```

```
47           android:backgroundTint="#05FFFFFF"
48           app:srcCompat="@drawable/ic_menu_gray" />
49
50       <ImageButton
51           android:id="@+id/home_btn_profile"
52           android:layout_width="wrap_content"
53           android:layout_height="wrap_content"
54           android:layout_gravity="center"
55           android:layout_weight="1"
56           android:backgroundTint="#05FFFFFF"
57           app:srcCompat="@drawable/ic_home_gray" />
58     </LinearLayout>
59
60     <TextView
61         android:id="@+id/textView"
62         android:layout_width="wrap_content"
63         android:layout_height="wrap_content"
64         android:text="Welcome, You are logged in "
65         android:textSize="30sp"
66         android:textStyle="bold"
67         app:layout_constraintBottom_toBottomOf="parent"
68         app:layout_constraintEnd_toEndOf="parent"
69         app:layout_constraintHorizontal_bias="0.5"
70
71             app:layout_constraintStart_toStartOf="parent"
72             app:layout_constraintTop_toTopOf="parent" />
73
74     <Button
75         android:id="@+id	btnLogout"
76         android:layout_width="match_parent"
77         android:layout_height="wrap_content"
78         android:layout_marginTop="16dp"
79         android:text="Logout"
80         android:backgroundTint="#B3B3B3"
81         android:textColor="#4A4A4A"
82         app:layout_constraintEnd_toEndOf="parent"
83         app:layout_constraintHorizontal_bias="0.5"
84         app:layout_constraintStart_toStartOf="parent"
85         app:layout_constraintTop_toBottomOf="@+id/textView" />
86
87     <Button
88         android:id="@+id/resetPasswordLocal"
89         android:layout_width="match_parent"
90         android:layout_height="wrap_content"
91         android:layout_marginTop="16dp"
92         android:text="Edit your password"
93         app:layout_constraintEnd_toEndOf="parent"
```

```
93 |         android:backgroundTint="#B3B3B3"
94 |         android:textColor="#4A4A4A"
95 |
96 |
97 |     <ImageView
98 |         android:id="@+id/imageView3"
99 |         android:layout_width="wrap_content"
100 |         android:layout_height="wrap_content"
101 |         app:layout_constraintBottom_toBottomOf="parent"
102 |         app:layout_constraintEnd_toEndOf="parent"
103 |         app:layout_constraintHorizontal_bias="0.497"
104 |         app:layout_constraintStart_toStartOf="parent"
105 |         app:layout_constraintTop_toTopOf="parent"
106 |         app:layout_constraintVertical_bias="0.261"
107 |         app:srcCompat="@drawable/ic_baseline_person" />
108 |
109 |     </androidx.constraintlayout.widget.ConstraintLayout>
110 |
111 |
112 | 
```

Table 5- 9-<<Menue.Java>> Code

```
1  package com.example.login_first;
2
3  import androidx.appcompat.app.AppCompatActivity;
4
5  import android.content.Intent;
6  import android.os.Bundle;
7  import android.view.View;
8  import android.widget.ImageButton;
9
10 import com.google.firebase.auth.FirebaseAuth;
11
12 public class MenueActivity extends AppCompatActivity {
13     public ImageButton Home_menu;
14     public ImageButton Star_menu;
15     public ImageButton Profile_menu;
16     public ImageButton Menu_menu;
17     public ImageButton Healthser;
18     public ImageButton Hajjser;
19     public ImageButton Educationser;
20     public ImageButton businesser;
21     public ImageButton JusticeandlawSer;
22
23
24     FirebaseAuth mAuth;
25     @Override
```

```
26 ❶  protected void onCreate(Bundle savedInstanceState) {  
27      super.onCreate(savedInstanceState);  
28      setContentView(R.layout.activity_menu);  
29  
30      mAuth = FirebaseAuth.getInstance();  
31  
32      Healthser=findViewById(R.id.health_1);  
33      Healthser.setOnClickListener(new View.OnClickListener() {  
34          @Override  
35          ❶  public void onClick(View view) {  
36              Intent intent = new Intent( packageContext: MenuActivity.this,HealthService.class);  
37              startActivity(intent);  
38          }  
39      });  
40  
41      Hajjser=findViewById(R.id.HajjandUmrah_Services);  
42      Hajjser.setOnClickListener(new View.OnClickListener() {  
43          @Override  
44          ❶  public void onClick(View view) {  
45              Intent intent = new Intent( packageContext: MenuActivity.this,HajjAndUmrahService.class);  
46              startActivity(intent);  
47          }  
48      });  
49  
50      Educationser=findViewById(R.id.education_Services);  
51      Educationser.setOnClickListener(new View.OnClickListener() {  
52          @Override  
53          ❶  public void onClick(View view) {  
54              Intent intent = new Intent( packageContext: MenuActivity.this,Educaiton_Services.class);  
55              startActivity(intent);  
56          }  
57      });  
58  
59      businesser=findViewById(R.id.business_Services);  
60      businesser.setOnClickListener(new View.OnClickListener() {  
61          @Override  
62          ❶  public void onClick(View view) {  
63              Intent intent = new Intent( packageContext: MenuActivity.this,BusinessService.class);  
64              startActivity(intent);  
65          }  
66      });  
67      JusticeandlawSer=findViewById(R.id.Justiceandlaw_Services);  
68      JusticeandlawSer.setOnClickListener(new View.OnClickListener() {  
69          @Override  
70          ❶  public void onClick(View view) {  
71              Intent intent = new Intent( packageContext: MenuActivity.this,JusticeAndLawService.class);  
72              startActivity(intent);  
73          }  
74      });  
75
```

```
76
77
78
79     Home_menu=(ImageButton) findViewById(R.id.home_btn_menu);
80     Home_menu.setOnClickListener(new View.OnClickListener() {
81
82         @Override
83         public void onClick(View view) {
84             Intent intent = new Intent( packageContext: MenuActivity.this,MainActivity.class);
85             startActivity(intent);
86         }
87     });
88
89     Star_menu=(ImageButton) findViewById(R.id.star_btn_menu);
90     Star_menu.setOnClickListener(new View.OnClickListener() {
91
92         @Override
93         public void onClick(View view) {
94             if(mAuth.getCurrentUser()==null){
95                 Intent intent=new Intent( packageContext: MenuActivity.this,GuestProfile.class);
96                 startActivity(intent);
97             }
98             else
99                 {Intent intent=new Intent( packageContext: MenuActivity.this,FavoriteActivity.class);
100                  startActivity(intent);}
101
102     });
103
104     Profile_menu=(ImageButton) findViewById(R.id.profile_btn_menu);
105     Profile_menu.setOnClickListener(new View.OnClickListener() {
106
107         @Override
108         public void onClick(View view) {
109             if(mAuth.getCurrentUser()==null){
110                 Intent intent=new Intent( packageContext: MenuActivity.this,GuestProfile.class);
111                 startActivity(intent);
112             }
113             else
114                 {Intent intent=new Intent( packageContext: MenuActivity.this,ProfileActivity.class);
115                  startActivity(intent);}
116         }
117     });
118 }
119 }
```

Table 5- 10-<<Menue.Xml>> Code

```
1  <?xml version="1.0" encoding="utf-8"?>
2  <androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
3      xmlns:app="http://schemas.android.com/apk/res-auto"
4      xmlns:tools="http://schemas.android.com/tools"
5      android:layout_width="match_parent"
6      android:layout_height="match_parent"
7      tools:context=".MenueActivity">
8
9
10     <ScrollView
11         android:id="@+id/scrollMenu"
12         android:layout_width="411dp"
13         android:layout_height="648dp"
14         android:layout_marginTop="90dp"
15         app:layout_constraintBottom_toBottomOf="parent"
16         app:layout_constraintEnd_toEndOf="parent"
17         app:layout_constraintStart_toStartOf="parent"
18         app:layout_constraintTop_toTopOf="parent">
19
20         <LinearLayout
21             android:id="@+id/linearScroll"
22             android:layout_width="match_parent"
23             android:layout_height="wrap_content"
24
25             android:orientation="vertical">
26
27             <LinearLayout
28                 android:id="@+id/Linear_health_menue"
29                 android:layout_width="match_parent"
30                 android:layout_height="140dp"
31                 android:background="#F7F7F8"
32                 android:orientation="vertical">
33
34                 <ImageButton
35                     android:id="@+id/health_1"
36                     android:layout_width="match_parent"
37                     android:layout_height="90dp"
38                     android:layout_marginStart="10dp"
39                     android:layout_marginTop="10dp"
40                     android:layout_marginEnd="10dp"
41                     android:layout_marginBottom="10dp"
42                     android:background="#F7F7F8"
43                     android:gravity="center"
44                     android:src="@drawable/stethoscope"
45                     android:textAlignment="center" />
46
47             <TextView
```

```
47         android:layout_width="wrap_content"
48         android:layout_height="wrap_content"
49         android:layout_gravity="center"
50         android:text="Health"
51         android:textColor="#3E3E3E"
52         android:textSize="20sp" />
53
54     </LinearLayout>
55
56     <LinearLayout
57         android:id="@+id/Linear_trading_menu"
58         android:layout_width="match_parent"
59         android:layout_height="140dp"
60         android:layout_marginTop="30dp"
61         android:background="#F7F7F8"
62         android:orientation="vertical">
63
64         <ImageButton
65             android:id="@+id/business_Services"
66             android:layout_width="match_parent"
67             android:layout_height="90dp"
68             android:layout_marginStart="10dp"
69             android:layout_marginTop="10dp"
70
71             android:layout_marginEnd="10dp"
72             android:layout_marginBottom="10dp"
73             android:background="#F7F7F8"
74             android:gravity="center"
75             android:src="@drawable/tradeing" />
76
77         <TextView
78             android:layout_width="wrap_content"
79             android:layout_height="wrap_content"
80             android:layout_gravity="center"
81             android:text="Business and Entrepreneurship"
82             android:textColor="#3E3E3E"
83             android:textSize="20sp" />
84
85     </LinearLayout>
86
87     <LinearLayout
88         android:id="@+id/Linear_education_menu"
89         android:layout_width="match_parent"
90         android:layout_height="140dp"
91         android:layout_marginTop="30dp"
92         android:background="#F7F7F8"
```

```
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
```

```
    android:orientation="vertical">

        <ImageButton
            android:id="@+id/education_Services"
            android:layout_width="match_parent"
            android:layout_height="90dp"
            android:layout_marginStart="10dp"
            android:layout_marginTop="10dp"
            android:layout_marginEnd="10dp"
            android:layout_marginBottom="10dp"
            android:background="#F7F7F8"
            android:gravity="center"
            android:src="@drawable/education" />

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_gravity="center"
            android:text="Education"
            android:textColor="#3E3E3E"
            android:textSize="20dp" />
    </LinearLayout>

<LinearLayout
    android:id="@+id/Linear_HajjandUmrah_menu"
    android:layout_width="match_parent"
    android:layout_height="140dp"
    android:layout_marginTop="30dp"
    android:background="#F7F7F8"
    android:orientation="vertical">

    <ImageButton
        android:id="@+id/HajjandUmrah_Services"
        android:layout_width="match_parent"
        android:layout_height="90dp"
        android:layout_marginStart="10dp"
        android:layout_marginTop="10dp"
        android:layout_marginEnd="10dp"
        android:layout_marginBottom="10dp"
        android:background="#F7F7F8"
        android:gravity="center"
        android:src="@drawable/kaaba" />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
```

```
137           android:layout_height="wrap_content"
138           android:layout_gravity="center"
139           android:text="Hajj and Umrah"
140           android:textColor="#3E3E3E"
141           android:textSize="20dp" />
142       </LinearLayout>
143
144
145   <LinearLayout
146       android:id="@+id/Linear_Justceandlaw_menu"
147       android:layout_width="match_parent"
148       android:layout_height="140dp"
149       android:layout_marginTop="30dp"
150       android:background="#F7F7F8"
151       android:orientation="vertical">
152
153     <ImageButton
154         android:id="@+id/JusticeandLaw_Services"
155         android:layout_width="match_parent"
156         android:layout_height="90dp"
157         android:layout_marginStart="10dp"
158         android:layout_marginTop="10dp"
159         android:layout_marginEnd="10dp"
160
161
162     <Image
163         android:layout_marginBottom="10dp"
164         android:background="#F7F7F8"
165         android:gravity="center"
166         android:src="@drawable/balance" />
167
168     <TextView
169         android:layout_width="wrap_content"
170         android:layout_height="wrap_content"
171         android:layout_gravity="center"
172         android:text="Justice and Law"
173         android:textColor="#3E3E3E"
174         android:textSize="20dp" />
175     </LinearLayout>
176     <LinearLayout
177         android:id="@+id/Empty"
178         android:layout_width="match_parent"
179         android:layout_height="140dp"
180         android:layout_marginTop="30dp"
181         android:background="#F7F7F8"
182         android:orientation="vertical">
```

```
183     |     </LinearLayout>
184     |     |
185     |     </LinearLayout>
186     |
187     |     </ScrollView>
188     |   />
189     |   <LinearLayout
190     |     android:id="@+id/menuNavBar"
191     |     android:layout_width="match_parent"
192     |     android:layout_height="70dp"
193     |     android:background="#FFFFFF"
194     |     android:orientation="horizontal"
195     |     app:layout_constraintBottom_toBottomOf="parent"
196     |     app:layout_constraintEnd_toEndOf="parent"
197     |     app:layout_constraintHorizontal_bias="1.0"
198     |     app:layout_constraintStart_toStartOf="parent"
199     |     app:layout_constraintTop_toTopOf="parent"
200     |     app:layout_constraintVertical_bias="1.0">
201     |       <ImageButton
202     |         android:id="@+id/profile_btn_menu"
203     |         android:layout_width="wrap_content"
204     |         android:layout_height="wrap_content"
205     |         android:layout_gravity="center"
206     |         android:layout_weight="1"
207     |         android:backgroundTint="#05FFFFFF"
208     |         app:srcCompat="@drawable/ic_person_gray" />
209
210
211     |       <ImageButton
212     |         android:id="@+id/star_btn_menu"
213     |         android:layout_width="wrap_content"
214     |         android:layout_height="wrap_content"
215     |         android:layout_gravity="center"
216     |         android:layout_weight="1"
217     |         android:backgroundTint="#05FFFFFF"
218     |         app:srcCompat="@drawable/ic_star_gray" />
219
220
221     |       <ImageButton
222     |         android:id="@+id/menu_btn_menu"
223     |         android:layout_width="wrap_content"
224     |         android:layout_height="wrap_content"
225     |         android:layout_gravity="center"
226     |         android:layout_weight="1"
227     |         android:backgroundTint="#05FFFFFF"
228     |         app:srcCompat="@drawable/ic_menu" />
```

```
229     <ImageButton  
230         android:id="@+id/home_btn_menu"  
231         android:layout_width="wrap_content"  
232         android:layout_height="wrap_content"  
233         android:layout_gravity="center"  
234         android:layout_weight="1"  
235         android:backgroundTint="#05FFFFFF"  
236         app:srcCompat="@drawable/ic_home_gray" />  
237     </LinearLayout>  
238  
239     <ImageView  
240         android:layout_width="match_parent"  
241         android:layout_height="80dp"  
242         android:layout_marginBottom="7dp"  
243         android:src="@drawable/dleel_img"  
244         app:layout_constraintBottom_toTopOf="@+id/scrollMenu"  
245         app:layout_constraintEnd_toEndOf="parent"  
246         app:layout_constraintStart_toStartOf="parent"  
247         app:layout_constraintTop_toTopOf="parent" />  
248  
249  
250 </androidx.constraintlayout.widget.ConstraintLayout>
```

Table 5- 11-<<HealthService.Java>> Code

```
1 package com.example.login_first;  
2  
3 import androidx.appcompat.app.AppCompatActivity;  
4 import android.app.SearchManager;  
5 import android.content.Intent;  
6  
7 import androidx.appcompat.widget.SearchView;  
8  
9 import android.widget.ImageButton;  
10 import android.widget.SearchView.OnQueryTextListener;  
11 import android.os.Bundle;  
12 import android.view.View;  
13 import android.widget.AdapterView;  
14 import android.widget.ArrayAdapter;  
15 import android.widget.ListView;  
16  
17 import android.widget.Toast;  
18  
19 public class HealthService extends AppCompatActivity {  
20     SearchView searchView;  
21     ListView listView;  
22     String[]nameList = {"Registration for covid 19 vaccine", "Testing for COVID-19", "Medical Appointment Service (Mawid)",  
23                     "Medical Appointment Cancellation Service", "Premarital Medical Examination Program Services"};  
24     ArrayAdapter<String> arrayAdapter;  
25     ImageButton Back_Bar_1;
```

```

26
27     @Override
28     protected void onCreate(Bundle savedInstanceState) {
29         super.onCreate(savedInstanceState);
30         setContentView(R.layout.activity_health_service);
31         searchView = (SearchView) findViewById(R.id.search_Baar);
32         listView = findViewById(R.id.list_item);
33         arrayAdapter= new ArrayAdapter<String>( context: this, android.R.layout.simple_list_item_1,android.R.id.text1, nameList);
34         listView.setAdapter(arrayAdapter);
35         listView.setOnItemClickListener(new AdapterView.OnItemClickListener() {
36             @Override
37             public void onItemClick(AdapterView<?> adapterView, View view, int i, long l) {
38                 Toast.makeText( context: HealthService.this, text: "you Click - "+adapterView.getItemAtPosition(i).toString(),
39                             Toast.LENGTH_SHORT).show();
39             }
40         });
41     };
42     //then Search view set filter...
43
44     searchView.setOnQueryTextListener(new SearchView.OnQueryTextListener() {
45         @Override
46         public boolean onQueryTextSubmit(String query) {
47             HealthService.this.arrayAdapter.getFilter().filter(query);
48             return false;
49         }
50     });
51
52     @Override
53     public boolean onQueryTextChange(String newText) {
54         HealthService.this.arrayAdapter.getFilter().filter(newText);
55         return false;
56     }
57     ArrayAdapter arrayAdapter1 =new ArrayAdapter(getApplicationContext(), android.R.layout.simple_list_item_1, nameList);
58     listView.setAdapter(arrayAdapter1);
59     listView.setOnItemClickListener(new AdapterView.OnItemClickListener() {
60         @Override
61         public void onItemClick(AdapterView<?> adapterView, View view, int position, long id) {
62             if(position==0){
63                 //click on Registration for covid 19 vaccine
64                 startActivity(new Intent( packageContext: HealthService.this,Covid_19_Vaccine.class));
65             }else if (position==1){
66                 //click on Testing for COVID-19
67                 startActivity(new Intent( packageContext: HealthService.this,testing_for_covid19.class));
68             }else if(position==2){
69                 // click on (Mawid)
70                 startActivity(new Intent( packageContext: HealthService.this,Medical_appointment_mawid.class));
71             }
72             }else if(position==3){
73                 // click on Medical Appointment Cancellation Service
74                 startActivity(new Intent( packageContext: HealthService.this,Medical_appointment_cancellation.class));
75             }

```

```

76     }else{
77         // click on Premarital Medical Examination
78
79         startActivity(new Intent( packageContext: HealthService.this,MenuActivity.class));
80
81     }
82 }
83 );
84 Back_Bar_1 =(ImageButton) findViewById(R.id.Back_Bar_1);
85 Back_Bar_1.setOnClickListener(new View.OnClickListener() {
86     @Override
87     public void onClick(View view) {
88         Intent intent = new Intent( packageContext: HealthService.this,MenuActivity.class);
89         startActivity(intent);
90     }
91 });
92
93 }
94
95
96

```

Table 5- 12-<<HealthService.Xml>> Code

```

1 <?xml version="1.0" encoding="utf-8"?>
2 <RelativeLayout
3     xmlns:android="http://schemas.android.com/apk/res/android"
4     xmlns:app="http://schemas.android.com/apk/res-auto"
5     xmlns:tools="http://schemas.android.com/tools"
6     android:layout_width="match_parent"
7     android:orientation="vertical"
8     android:layout_height="match_parent"
9     tools:context=".HealthService">
10
11     <ImageView
12         android:id="@+id/imgBackBar1"
13         android:layout_width="match_parent"
14         android:layout_height="80dp"
15         android:layout_marginBottom="7dp"
16         android:background="#EFEFEF"
17         android:src="@drawable/dleel_img"
18     />
19
20     <ImageButton
21         android:id="@+id/Back_Bar_1"
22         android:layout_width="wrap_content"
23         android:layout_height="wrap_content"

```

```
24 |         android:layout_alignLeft="@+id/imgBackBar1"
25 |         android:layout_marginStart="15dp"
26 |         android:layout_marginTop="15dp"
27 |         android:layout_marginEnd="15dp"
28 |         android:layout_marginBottom="15dp"
29 |         android:backgroundTint="#05FFFFFF"
30 |         app:srcCompat="@drawable/ic_arrow_back"
31 |         tools:ignore="SpeakableTextPresentCheck" />
32 |
33 |     <ScrollView
34 |         android:layout_below="@+id/imgBackBar1"
35 |         android:id="@+id/scroll_OF_Health"
36 |         android:layout_width="match_parent"
37 |         android:layout_height="match_parent"
38 |
39 |     >
40 |     <RelativeLayout
41 |         android:layout_width="match_parent"
42 |         android:layout_height="wrap_content" >
43 |
44 |         <androidx.appcompat.widget.SearchView
45 |             android:id="@+id/search_Baar"
46 |             android:layout_width="match_parent"
47 |
48 |             android:layout_height="wrap_content"
49 |             android:layout_marginStart="2dp"
50 |             android:layout_marginTop="2dp"
51 |             android:layout_marginEnd="2dp"
52 |             android:layout_marginBottom="2dp"
53 |             app:iconifiedByDefault="false"
54 |             app:queryHint="Search name">
55 |
56 |         </androidx.appcompat.widget.SearchView>
57 |         <ListView
58 |
59 |             android:id="@+id/list_item"
60 |             android:layout_width="match_parent"
61 |             android:layout_height="800dp"
62 |             android:layout_below="@+id/search_Baar"
63 |             android:layout_marginStart="2dp"
64 |             android:layout_marginTop="25dp"
65 |             android:layout_marginEnd="2dp"
66 |             android:layout_marginBottom="2dp"
67 |             android:divider="#E1E1E1"
68 |             android:dividerHeight="30.0sp">
69 |
70 |         </ListView>
71 |     </RelativeLayout>
72 |     </ScrollView>
73 |
74 | </RelativeLayout>
75 |
```

Table 5- 13-<<Testing_For_Covid.Java>> Code

```
1 package com.example.login_first;
2
3 import androidx.annotation.NonNull;
4 import androidx.appcompat.app.AppCompatActivity;
5
6 import android.content.Intent;
7 import android.net.Uri;
8 import android.os.Bundle;
9 import android.util.Log;
10 import android.view.View;
11 import android.widget.ImageButton;
12 import android.widget.MediaController;
13 import android.widget.TextView;
14 import android.widget.Toast;
15 import android.widget.VideoView;
16
17 import com.google.android.gms.tasks.OnCompleteListener;
18 import com.google.android.gms.tasks.Task;
19 import com.google.firebase.auth.FirebaseAuth;
20 import com.google.firebaseio.database.DataSnapshot;
21 import com.google.firebaseio.database.DatabaseError;
22 import com.google.firebaseio.database.DatabaseReference;
23 import com.google.firebaseio.database.FirebaseDatabase;
24 import com.google.firebaseio.database.Query;
25 import com.google.firebaseio.database.ValueEventListener;
26
27 import java.util.HashMap;
28 import java.util.Map;
29
30 public class testing_for_covid19 extends AppCompatActivity {
31     public ImageButton BackTOhealthlist2;
32     public ImageButton Sehhaty_icon2;
33     FirebaseAuth mAuth;
34     DatabaseReference RootRef;
35     String currentUserID;
36     boolean bookmarked=false;
37     TextView textView;
38     ImageButton imageView;
39
40     @Override
41     protected void onCreate(Bundle savedInstanceState) {
42         super.onCreate(savedInstanceState);
43         setContentView(R.layout.activity_testing_for_covid19);
44         mAuth = FirebaseAuth.getInstance();
45         currentUserID = mAuth.getCurrentUser().getUid();
46         RootRef = FirebaseDatabase.getInstance().getReference();
47
48         textView=findViewById(R.id.covid);
49         imageView=findViewById(R.id.star_testing_covid19);
50 }
```

```

51     RootRef.child("Favorites").child(currentUserID).orderByChild("text").equalTo(textView.getText().toString())
52         .addValueEventListener(new ValueEventListener() {
53             @Override
54             public void onDataChange(@NonNull DataSnapshot snapshot) {
55                 if(snapshot.exists()){
56                     // Drawable myIcon = AppCompatResources.getDrawable(getApplicationContext(),
57                     // R.drawable.ic_baseline_star_rate_24);
58                     ImageButton.setImageResource(R.drawable.ic_star_pink);
59                     bookmarked=true;
60                 }
61             }
62
63             @Override
64             public void onCancelled(@NonNull DatabaseError error) {
65
66             }
67         });
68
69
70         ImageButton.setOnClickListener(new View.OnClickListener() {
71             @Override
72             public void onClick(View view) {
73                 if(!bookmarked) {
74
75                     DatabaseReference userMessageKeyRef = RootRef.child(currentUserID).child("Favorites").push();
76
77                     String messagePushID = userMessageKeyRef.getKey();
78
79                     String path = "Favorites/" + currentUserID;
80
81                     Map messageTextBody = new HashMap();
82                     messageTextBody.put("text", textView.getText().toString());
83                     Map messageBodyDetails = new HashMap();
84                     messageBodyDetails.put(path + "/" + messagePushID, messageTextBody);
85
86                     RootRef.updateChildren(messageBodyDetails).addOnCompleteListener(new OnCompleteListener<Void>() {
87                         @Override
88                         public void onComplete(@NonNull Task<Void> task) {
89                             if(task.isSuccessful()) {
90                                 ImageButton.setImageResource(R.drawable.ic_star_pink);
91                                 bookmarked=true;
92                                 Toast.makeText(getApplicationContext(), "Added to favorites", Toast.LENGTH_SHORT).show();
93                             } else {
94                                 Toast.makeText(context, "Error", Toast.LENGTH_SHORT).show();
95                             }
96                         }
97                     });
98                 } else{
99                     Query applesQuery = RootRef.child("Favorites").child(currentUserID).orderByChild("text")
100                         .equalTo(textView.getText().toString());

```

```
101     applesQuery.addListenerForSingleValueEvent(new ValueEventListener() {
102
103     @Override
104     public void onDataChange(DataSnapshot dataSnapshot) {
105         for (DataSnapshot appleSnapshot: dataSnapshot.getChildren()) {
106             Toast.makeText( context: testing_for_covid19.this, text: "removed from favorites", Toast.LENGTH_SHORT).show();
107             appleSnapshot.getRef().removeValue();
108             imageButton.setImageResource(R.drawable.ic_star);
109             bookmarked=false;
110         }
111     }
112
113     @Override
114     public void onCancelled(DatabaseError databaseError) {
115         Log.e( tag: "Result of removing", msg: "onCancelled", databaseError.toException());
116     }
117 });
118 });
119 });
120
121 //Testing_for_COVID19
122 BackTOhealthlist2 =(ImageButton) findViewById(R.id.lefBACKtoHealthList_2);
123 BackTOhealthlist2.setOnClickListener(new View.OnClickListener() {
124     @Override
125     public void onClick(View view) {
126
127         Intent intent = new Intent( packageContext: testing_for_covid19.this, HealthService.class);
128         startActivity(intent);
129     });
130
131     /////video
132     VideoView videoView = findViewById(R.id.videoTestForCOVID);
133     videoView.setVideoPath("android.resource://"+getPackageName()+"//"+R.raw.test_for_covid19);
134     MediaController mediaController =new MediaController( context: this);
135     mediaController.setAnchorView(videoView);
136     videoView.setMediaController(mediaController);
137     /// Sehhaty Link to google play
138     Sehhaty_icon2 = findViewById(R.id.sehhatyapp2);
139     Sehhaty_icon2.setOnClickListener(new View.OnClickListener() {
140
141         @Override
142         public void onClick(View view) {
143             GotoUrl( s: "https://play.google.com/store/apps/details?id=moh.gov.sa.mawid&hl=ar&gl=US");
144         });
145     }private void GotoUrl(String s){
146
147         Uri uri= Uri.parse(s);
148         startActivity(new Intent(Intent.ACTION_VIEW,uri));
149     }
150 }
```

Table 5- 14-<<Testing_For_Covid.Xml>> Code

```
1  <?xml version="1.0" encoding="utf-8"?>
2  <RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
3      xmlns:app="http://schemas.android.com/apk/res-auto"
4      xmlns:tools="http://schemas.android.com/tools"
5      android:layout_width="match_parent"
6      android:layout_height="match_parent"
7      tools:context=".testing_for_covid19">
8
9      <TextView
10         android:id="@+id/covid"
11         android:layout_width="match_parent"
12         android:layout_height="80dp"
13         android:background="#EFEFEF"
14         android:gravity="center"
15         android:text=" appointment for a COVID-19 test "
16         android:textSize="18dp" />
17
18      <ImageButton
19          android:id="@+id/star_testing_covid19"
20          android:layout_width="wrap_content"
21          android:layout_height="wrap_content"
22          android:layout_alignRight="@+id/covid"
23          android:layout_marginStart="15dp"
24
25          android:layout_marginTop="15dp"
26          android:layout_marginEnd="15dp"
27          android:layout_marginBottom="15dp"
28          android:backgroundTint="#05FFFFFF"
29          app:srcCompat="@drawable/ic_star"
30          tools:ignore="SpeakableTextPresentCheck" />
31
32      <ImageButton
33          android:id="@+id/lefBACKtoHealthList_2"
34          android:layout_width="wrap_content"
35          android:layout_height="wrap_content"
36          android:layout_alignLeft="@+id/covid"
37          android:layout_marginStart="15dp"
38          android:layout_marginTop="15dp"
39          android:layout_marginEnd="15dp"
40          android:layout_marginBottom="15dp"
41          android:backgroundTint="#05FFFFFF"
42          app:srcCompat="@drawable/ic_arrow_back"
43          tools:ignore="SpeakableTextPresentCheck" />
44
45      <ImageButton
46          android:id="@+id/sehhatyapp2"
47          android:layout_width="wrap_content"
```

```
47     android:layout_height="wrap_content"
48     android:layout_below="@+id/covid"
49     android:layout_marginLeft="160dp"
50     android:layout_marginTop="30dp"
51     android:backgroundTint="#00FFFFFF"
52     app:srcCompat="@drawable/sehhaty"
53     tools:ignore="SpeakableTextPresentCheck" />
54
55     <TextView
56         android:layout_width="wrap_content"
57         android:layout_height="wrap_content"
58         android:layout_below="@+id/sehhatyapp2"
59         android:layout_marginLeft="160dp"
60         android:text="Sehhaty"
61         android:textSize="20sp"
62         android:textStyle="bold" />
63
64     <FrameLayout
65         android:id="@+id/frameOfvideo"
66         android:layout_width="match_parent"
67         android:layout_height="wrap_content"
68         android:layout_below="@+id/sehhatyapp2"
69         android:layout_centerHorizontal="true"
70
71         android:layout_marginTop="30dp">
72
73         <VideoView
74             android:id="@+id/videoTestForCOVID"
75             android:layout_width="match_parent"
76             android:layout_height="200dp"
77             android:layout_gravity="center"
78         />
79     </FrameLayout>
80
81     <ScrollView
82         android:id="@+id/scroll"
83         android:layout_width="match_parent"
84         android:layout_height="wrap_content"
85         android:layout_below="@+id/frameOfvideo"
86         tools:ignore="SpeakableTextPresentCheck">
87
88         <LinearLayout
89             android:layout_width="match_parent"
90             android:layout_height="wrap_content"
91             android:orientation="vertical"
92             >
93             <ImageView
```

```

93
94
95
96
97
98  android:id="@+id/steps_img"
99     android:layout_width="280dp"
100    android:layout_height="350dp"
101    android:layout_weight="2"
102    android:layout_marginRight="200dp"
103    android:src="@drawable/steps_test_covid19" />
104
105 </LinearLayout>
106 </ScrollView>
107
108 </RelativeLayout>

```

5.4 I/O screens

Figure 5-1 Shows the shared interface between all the users, the user needs to pick one of the following Login, Sign-up or a guest. the system validated them and check by the user what is the type of the user. If the user already has an account will login and it will be an active account, if the user does not have an account he can easily create one by clicking on Sign-up button to have an account in DLEEL.

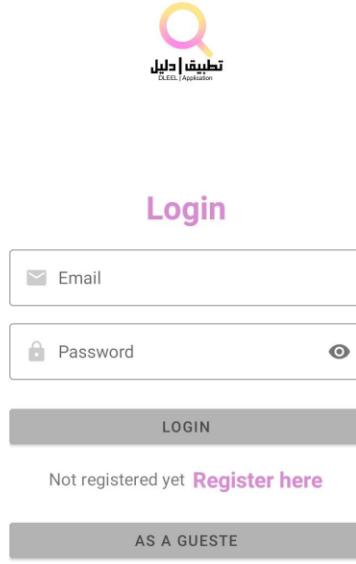


Figure 5-1-<<main page>>interface

Figure 5-2 Shows Sign-up interfaces it contains Email, Password and Rewrite email, After the user registered, the account will be active and the user can use his/her account and login without any constraint.



Register

✉

🔒
👁

🔒
👁

REGISTER

Already registered [Login here](#)



Figure 5- 2-<<Register>>interface

Figure 5-3 Shows Login interface it contains username and password fields the user need to enter it username and password, the system have to check if it is valid or invalid to login successfully.

Login

✉

🔒
👁

LOGIN

Not registered yet [Register here](#)

AS A GUEST

Figure 5- 3-<<login>>interface



Welcome to Dleel application!

Common Applications



Figure 5- 4-<<home page>>interface

Figure 5-4 The home page displays the most important popular applications, and when the user clicks on the application, it will be transferred to the link to download the application from Google Play.



Figure 5-5 Shows the menu interfaces the user can scroll and check the service type he/she wants, and by clicking on the service type “Health” new interface will appear and it will show the health services as shown “figure”. Or the user can search for specific service by writing it on the Search field as shown “figure”

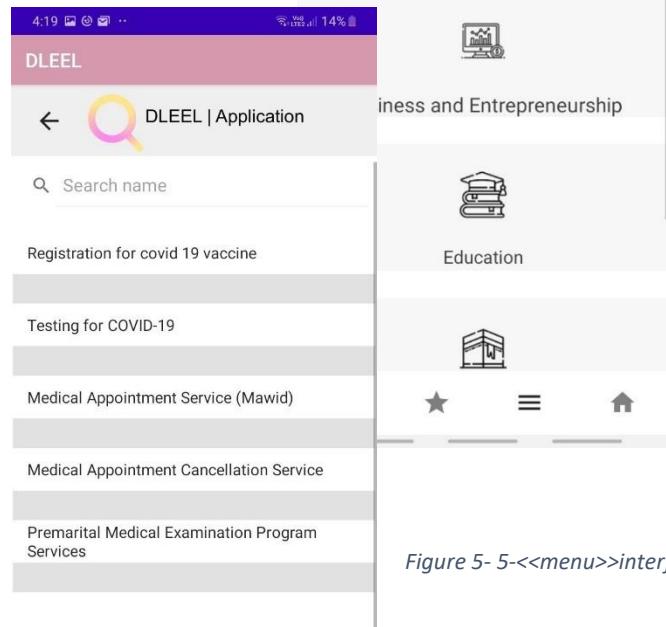


Figure 5- 5-<<menu>>interface

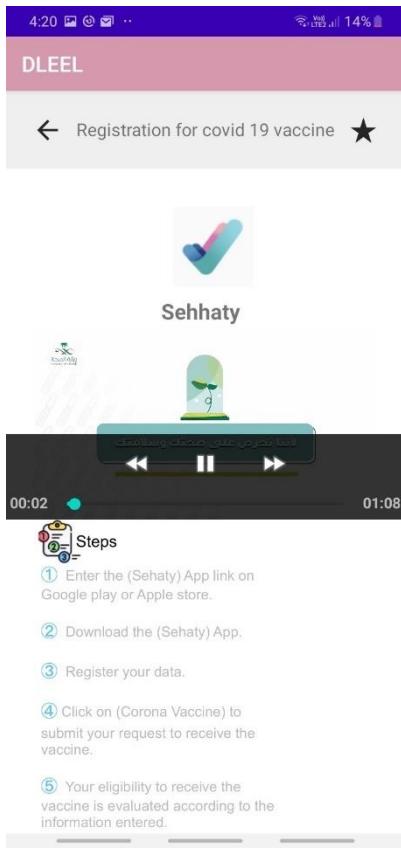


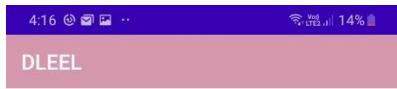
Figure 5-6 Shows service step page interface, video and steps will appear for the user. The user can watch the video about the steps or he can read the steps below the video.

Figure 5- 6<<step>>interface



Figure 5-7 the user must login/sign-up to use the feature for accessing preference service faster and easier

Figure 5- 7-<<favorite service>>interface



Welcome, You are logged in

LOGOUT

EDIT YOUR PASSWORD



Figure 5- 8-<<profile>> interface

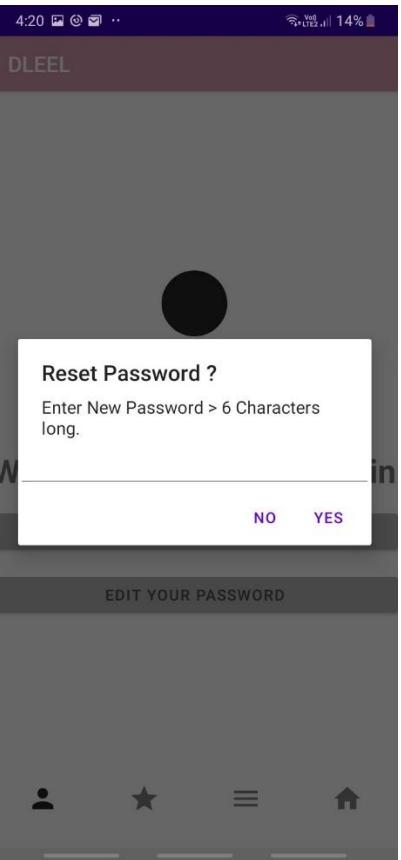


Figure 5-9 Shows Edit password page ,If users want to change their password, they have to use 6 character to be able to edit it.

5.5 Summary

In this chapter, we focused on the implementation of DLEEL. First, we described the implementation requirements from both of hardware and software view then we depicted implementation details and Finally, we showed the main interfaces' screens of DLEEL

Chapter 6 : Testing

Chapter 6 Testing:

In this chapter, we focused on testing the DLEEL application. We describe the testing plan and strategies and present the schedule of testing activities as well as show the application test cases and test results for the DLEEL application.

6.1 Test Plan

A Test Plan refers to a detailed document that catalogs the test strategy, objectives, schedule for completing that particular project. It is a blueprint for running the tests needed to ensure the software is working properly. The test plan is important for a DLEEL because it details aspects such as testing scope, strategy and to ensure that it meets our requirements.

6.2.1 Testing Strategy

While testing our DLEEL application, we covered several testing that can be present it as follows:

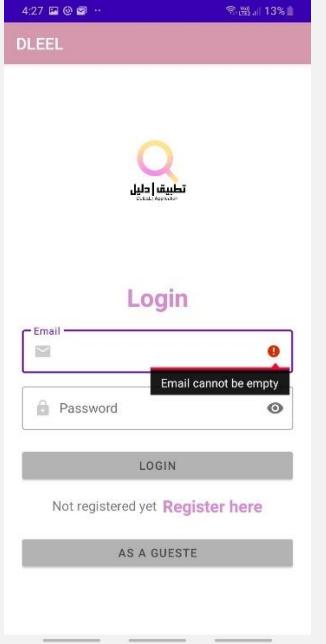
- **Unit testing :** is the process of testing program components or individual units, such as methods, object classes and individual functions . We have individually tested each unit to ensure it works by dividing the program into activities and each member of the team tested one activity to make sure that it worked and was free from logical and syntax errors.
- **Functionality Testing :** is the process of testing system against to the functional requirements/specifications by selecting function or feature and give them input and see the output if it meets the functional requirements/specifications that already been identified or not.
- **Database Testing :** Is the process of testing all system. We tested the database process of Login-sign up-favorite list we make sure it work correctness.

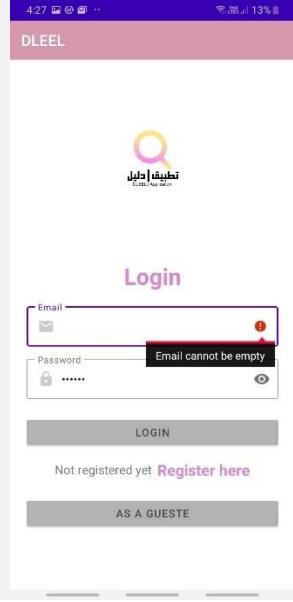
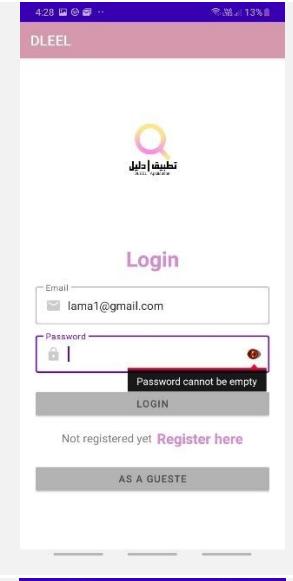
6.2.2. Schedule of testing activities

| Test Activity | Test completed |
|-----------------------|----------------|
| Unite testing | 3/4/2022 |
| Functionality Testing | 10/4/2022 |
| Error Testing | 15/4/2022 |
| Database Testing | 21/4/2022 |

6.2.3 Test Cases

Table 6- 1-<<Login>>test case

| ID | Description | Status | Expected Result | Screenshot |
|-------|---|--------|---|---|
| TC1_1 | The user did not fill any of the fields | Fail | An Error message is displayed “Email cannot be empty” |  <p>A screenshot of a mobile application's login screen. At the top, there is a navigation bar with the text 'DLEEL' and some icons. Below the navigation bar is a logo consisting of a yellow magnifying glass icon with the text 'نطاق احلي' (Netaq Alali) underneath it. The main screen has a light blue background with the word 'Login' centered at the top. Below 'Login' are two input fields: 'Email' and 'Password'. The 'Email' field has a red error message bubble above it containing the text 'Email cannot be empty'. Below the input fields are two large grey buttons labeled 'LOGIN' and 'AS A GUEST'. At the bottom of the screen, there is a link 'Not registered yet? Register here'.</p> |

| | | | | |
|-------|---|------|---|---|
| TC1_2 | The user did not enter the email | Fail | An Error message is displayed “Email cannot be empty” |  <p>The screenshot shows the DLEEL app's login interface. At the top, there is a purple header bar with the app's logo and name. Below it is a white background with a yellow circular logo containing a stylized letter 'Q'. The word 'Login' is centered above two input fields: 'Email' and 'Password'. The 'Email' field is empty and highlighted with a purple border. To its right, a small red circle with a white exclamation mark indicates an error. A black rectangular box contains the error message 'Email cannot be empty'. Below the input fields is a large grey 'LOGIN' button. Underneath the button, there is a link 'Not registered yet? Register here' and a 'AS A GUEST' button.</p> |
| TC1_3 | The user did not enter the password | Fail | An Error message is displayed “Password cannot be empty” |  <p>The screenshot shows the DLEEL app's login interface. At the top, there is a purple header bar with the app's logo and name. Below it is a white background with a yellow circular logo containing a stylized letter 'Q'. The word 'Login' is centered above two input fields: 'Email' and 'Password'. The 'Email' field contains the value 'lama1@gmail.com'. The 'Password' field is empty and highlighted with a purple border. To its right, a small red circle with a white exclamation mark indicates an error. A black rectangular box contains the error message 'Password cannot be empty'. Below the input fields is a large grey 'LOGIN' button. Underneath the button, there is a link 'Not registered yet? Register here' and a 'AS A GUEST' button.</p> |
| TC1_4 | If the user write the email format in incorrect way | Fail | An Error message is displayed “Log in Error : the email address is badly formatted” |  <p>The screenshot shows the DLEEL app's login interface. At the top, there is a purple header bar with the app's logo and name. Below it is a white background with a yellow circular logo containing a stylized letter 'Q'. The word 'Login' is centered above two input fields: 'Email' and 'Password'. The 'Email' field contains the value 'lama1@gmail.com'. The 'Password' field contains the value '.....'. Below the input fields is a large grey 'LOGIN' button. Underneath the button, there is a link 'Not registered yet? Register here' and a 'AS A GUEST' button. A black rectangular box at the bottom contains the error message 'Log in Error: The email address is badly formatted.'</p> |

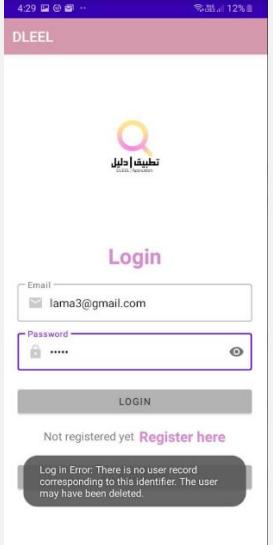
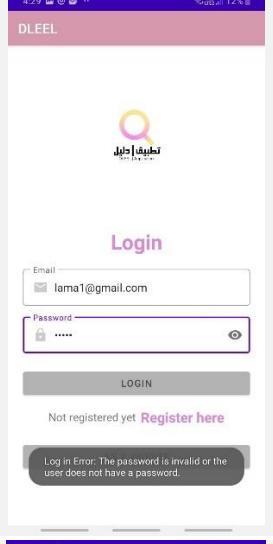
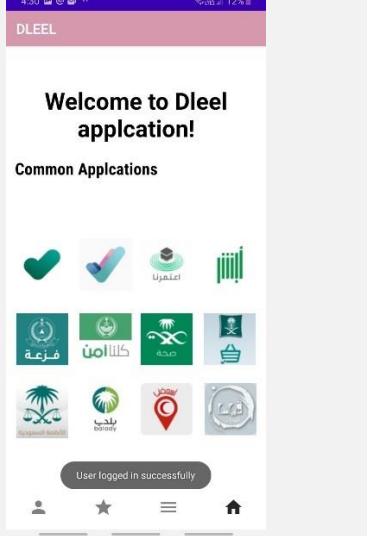
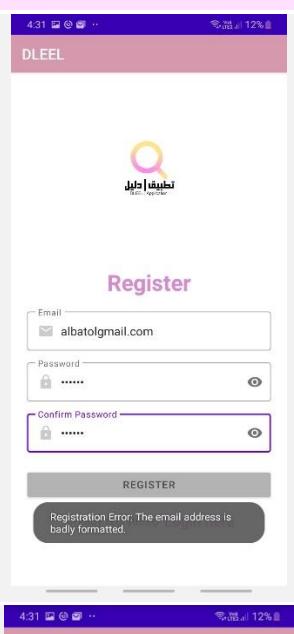
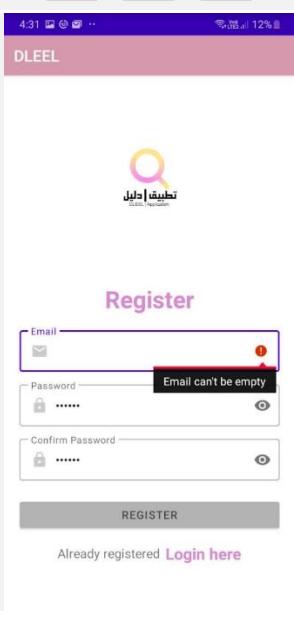
| | | | | |
|-------|--|---------|---|---|
| TC1_5 | If the user enter wrong email | Fail | An Error message is displayed “Log in Error : There is no user record corresponding to this identifier . The user may have been deleted ” |  |
| TC1_6 | If The user enters incorrect password | Fail | An error message is displayed “The password is invalid or the user does not have password” |  |
| TC1_7 | The user enters Email and Password correct | Success | The user logged in successfully displayed messages “User logged in successfully” And The Home page will appear |  |

Table 6- 2-<<REGESTER>> test case

| ID | Description | Status | Expected Result | Screenshot |
|-------|---------------------------------------|--------|--|--|
| TC2_1 | The user wrote incorrect Email format | Fail | An Error message is displayed “Registration Error: the email address is badly format” |  |
| TC2_2 | The user did not enter the email | Fail | An Error message is displayed “Email cannot be empty” |  |

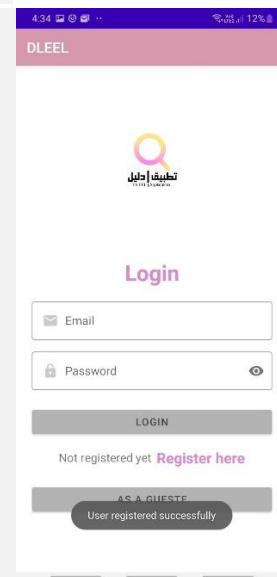
| | | | | |
|-------|--|---------|---|---|
| TC2_3 | The user did not enter the password | Fail | An Error message is displayed “Password cannot be empty” |  |
| TC2_4 | The user enter invalid password | Fail | An Error message is displayed “the given password is invalid password should be at least 6 Characters ” |  |
| TC2_5 | The user enters Email and Password correct | Success | The user logged in successfully displayed messages “ User registered successfully” |  |

Table 6- 3-<<Edit password>> test case

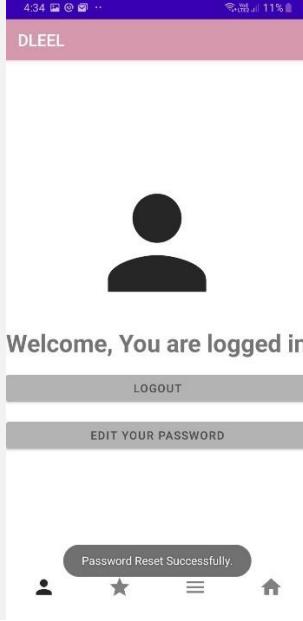
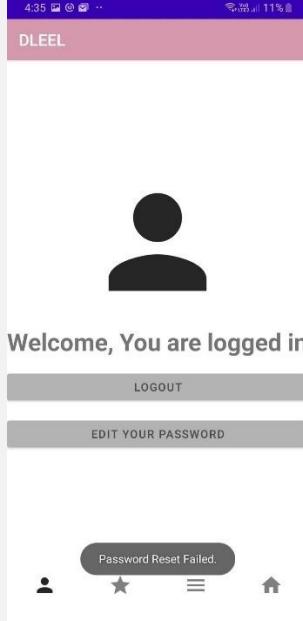
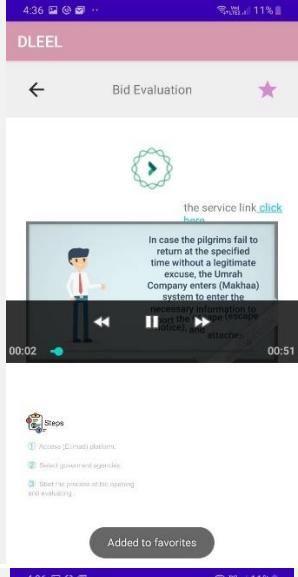
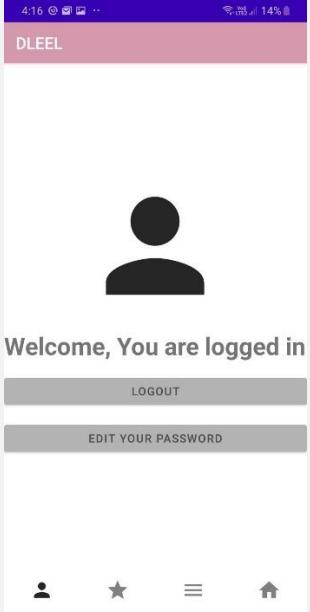
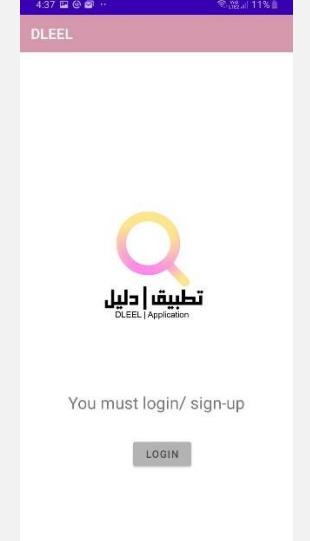
| ID | Description | Status | Expected Result | Screenshot |
|-------|--------------------------------------|---------|--|--|
| TC3_1 | The user wrote 6 OR more character | Success | A message is displayed "Password Reset Successfully." |  |
| TC3_2 | The user wrote less than 6 character | Fail | An Error message is displayed "Password Reset Failed." |  |

Table 6- 4-<<favorite service>> test case

| ID | Description | Status | Expected Result | Screenshot |
|-------|---|---------|---|--|
| TC4_1 | The user logged in and added the service to Favorite list | Success | A message is displayed "Added to favorites" |  |
| TC4_2 | The user logged in and removed the service from Favorite list | Success | A message is displayed "removed from favorites" |  |

| | | | | | | |
|-------|--|------|---|--|--|--|
| TC4_3 | The guest added the service to Favorite list | Fail | A message is displayed “you must login sign-up” | |  You must login/ sign-up <input type="button" value="LOGIN"/> | |
|-------|--|------|---|--|--|--|

Table 6- 5-<<Edit profile>>test case

| ID | Description | Status | Expected Result | Screenshot |
|----|---|---------|---|--|
| | The user logged in and clicks on Profile icon | Success | The profile page appear |  |
| | The gest clicks on Profile icon | Fail | A message is displayed “you must login sign-up” with button “login” |  |

Problems and modifications

- After we did several tests including database, unit test and functionality test we make sure we made an application easy to use and free from errors And it's requirement is complete , we faced some of problems and we solved it
- We dispensed with the history page activity because we realized that the favorites list activity has almost the same purpose, and we can suffice with one to help the user reach the service they want to return to.
- One of problems was after we finished designing the XML interface, we unified all the object names and then linked the interfaces together, which showed many problems that we can solve only by reworking a clear and detailed plan.
- In some services, there was no video to explain the service, so we produce it ourselves to make sure that the service was clear.

- Another problems was databases ,We were confused about which database is better for us as a beginner to use, but after research and study, we identified the database that we want to work on
- We encountered a problem when the user was a guest. We could not do it until after we finished the user. Then our ideas were arranged and we learned how to activate the guest.

6.3 TEST Result:

After we did several tests including database, unit test and functionality test we make sure we made an application easy to use and free from errors And its requirement is complete.

Chapter 7 : Conclusion

Chapter 7 conclusion

7.1 Evaluation:

The goal of our project is to design mobile application for serving Citizens. first, we have started with background information and related works and compared our applications with their works, second we focused on the requirements specifications included functional and non-functional requirements of our application, requirements analysis been performed from both structural and functional using UML diagrams. Then we started doing the implementation and the testing phases, with taking screenshots of the main I/O from DLEEL application and made sure that the usability is well-defined based on the plan

7.2 Future work

As future work, we are looking forward to improve our application DLEEL by adding more features such as voice reader for the steps, pdf page for each service with more details and we are planning to make DLEEL Application on other platforms such as IOS and website.

References

- [1] "HOKUAPPS," [Online]. Available: <https://www.hokuapps.com/blogs/10-reasons-choose-android-mobile-application-development/>.
- [2] "Sahel," [Online]. Available: <https://play.google.com/store/apps/details?id=kw.gov.sahel&hl=ar&gl=US>.
- [3] "Metrash2," [Online]. Available: <https://play.google.com/store/apps/details?id=com.vipera.ts.starter.Metrash2&hl=ar&gl=US>.
- [4] "Balade," [Online]. Available: <https://play.google.com/store/apps/details?id=com.balade>.
- [5] "Omanna," [Online]. Available: <https://play.google.com/store/apps/details?id=com.ita.omanuna>.
- [6] I. Sommerville, Software Engineering, Addison Wesley.
- [7] "Geeks for geeks," [Online]. Available: <https://www.geeksforgeeks.org/designing-use-cases-for-a-project/>.
- [8] "geeks for geeks," [Online]. Available: <https://www.geeksforgeeks.org/unified-modeling-language-uml-sequence-diagrams/>.
- [9] "Enlab," [Online]. Available: <https://enlabsoftware.com/development/how-to-build-and-deploy-a-three-layer-architecture-application-with-c-sharp-net-in-practice.html>.

Appendices

Appendix A

Here we review the survey questions graph we published to gather requirements.

■ Yes ■ No ■ Sometimes

1-Are you experiencing difficulties in submitting government transactions?

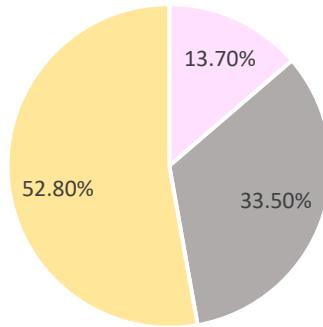


Figure 3-1 -Survey-Question 1

2-Does the multiplicity of government applications cause you to be confused in deciding which application you need?

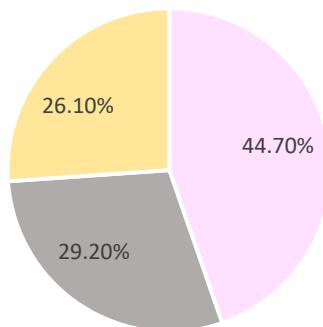
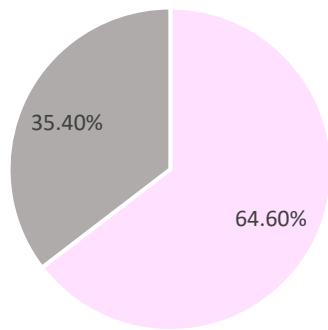
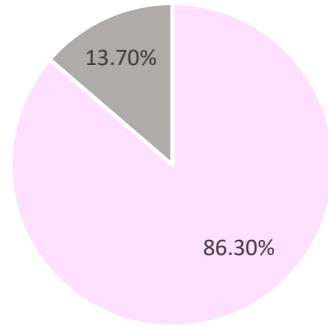


Figure 3-2 -Survey-Question 2

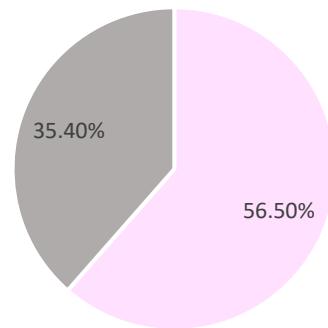
3-Have you ever needed to complete one of your transactions but didn't know the application of this service?



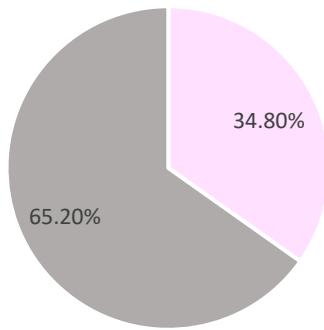
4-Have you ever needed a video explaining how to use a particular service?



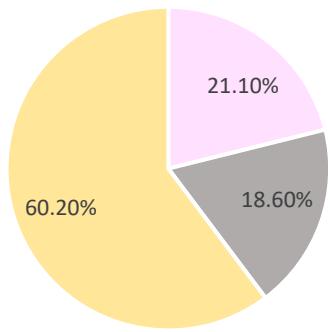
5-Have you ever had a problem using a government app? Do you have an excellent background on government application?



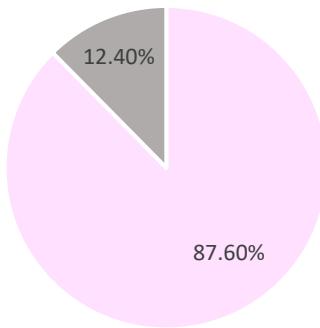
6-Have you ever had a problem not knowing which app to use?



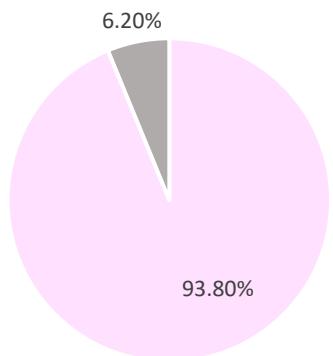
7-Do you have an excellent government applications?



8-Does having a directory application for government applications make it easier for you to find the services you need?



9-Do you support an app that collects government applications?



10-Have you ever been defrauded by an informal link to an e-service?

