

Kingdom of Saudi Arabia  
Majmaah University  
Ministry of Higher Education  
College of Science in Zolfi  
Dept. of Computer Science



**Bank to reserve an appointment  
Dept. of Computer Science**

**By:**  
Tahany Abdallah Ahmed AL-Zuwayid

**Graduation project**

**Submitted in partial fulfilment of the requirements  
for the award of  
Bachelor degree  
of the  
Majmaa University**

**Supervisor: T.Chafika Laabidi Ouni**

**Faculty of science  
Dept. of Computer Science**

**2018**

## **Acknowledgements**

In the name of Allah the most merciful and beneficent, First, we thank Allah who is the first and final assistant and the only one who gives us the grace and strength to walk this path and guiding us to the correct way. We would like to express our thanks to our families who support us through all steps of our life and still supporting us until the end.

We would like to express our sincere gratitude to our supervisor for her continuous support to help us, for her patience, motivation, and immense knowledge. Her guidance helped us all the time also we would like to thank all the people who directly or indirectly supported us during the completion of project.

## **Abstract**

Bank Customers Appointments is a mobile application developed to make customer management appointments easy and convenient to save time and effort to deliver services smoothly.

Bank Customers Appointments app is a simple application for processing electronic appointments where the customer can book appointments and check their availability through the application and choose the service they want in easy way without any effort.

Questionnaire prepared a to identify customers problems through many questions about the problem facing customers with bank appointments and how spend a long time to waiting appointments. These problems have been identified and the solution to these problems was developed through the development of the app.

**MAJMAAH UNIVERSITY,  
COLLEGE OF SCIENCE AL ZULFI,  
DEPARTMENT OF COMPUTER SCIENCE AND INFORMATION**

**(CERTIFICATE BY STUDENT)**

This is to certify that the project titled “**Bank to reserve an appointment**” submitted by me  
(**Tahany Abdallah Ahmed AL-Zuwayid, 351202305**) under the supervision of **T.Chafika  
Laabidi Ouni** for award of Bachelor degree of the Majmaah University carried out during the  
Semester 1, 2018-19 embodies my original work.

Signature in full: -----

Name in block letters:

Student ID:

Date:

## Contents

<b>Title</b>	<b>Number</b>
<b>Acknowledgements</b>	<b>i</b>
<b>Abstract</b>	<b>ii</b>
<b>Certificate</b>	<b>iii</b>
<b>1.Introduction</b>	<b>1</b>
<b>1.1.Overview</b>	<b>1</b>
<b>1.1.1. Abstract system description</b>	<b>1</b>
<b>1.2. Problem definition</b>	<b>2</b>
<b>1.2.1. Goals</b>	<b>2</b>
<b>1.2.2. Literature review</b>	<b>2</b>
<b>1.2.3. Collect of data</b>	<b>5</b>
<b>1.2.4. Objectives</b>	<b>5</b>
<b>1.2.5. Critical success factors</b>	<b>6</b>
<b>1.2.6. Organization chart and responsibilities</b>	<b>6</b>
<b>1.3. General rules (assumptions)</b>	<b>6</b>
<b>2. System analysis</b>	<b>7</b>
<b>2.1. Introduction</b>	<b>7</b>
<b>2.2. Description of Data Flow Diagram (DFD)</b>	<b>7</b>
<b>2.2.1. Context Diagram</b>	<b>8</b>
<b>2.2.2. Overview diagram (level 0)</b>	<b>9</b>
<b>2.2.3. Detailed DFDs</b>	<b>10</b>
<b>2.3. Entity Relationship Diagram (ERD)</b>	<b>10</b>
<b>2.3.1. Description of Entities</b>	<b>11</b>
<b>2.3.2. Description of relations</b>	<b>11</b>
<b>2.3.3. Drawing ERD</b>	<b>11</b>
<b>2.4. Structure Diagram</b>	<b>13</b>
<b>2.4.1. Class Diagram</b>	<b>13</b>
<b>2.5. Behavior diagram</b>	<b>14</b>
<b>2.5.1. use case Diagram</b>	<b>14</b>
<b>a) Customer Use Case</b>	<b>15</b>
<b>b) Bank employee Use Case</b>	<b>16</b>
<b>2.5.2. Activity Diagram</b>	<b>17</b>
<b>2.5.3. State Diagram</b>	<b>18</b>
<b>2.6. Interaction Diagram</b>	<b>19</b>
<b>2.6.1. Sequence Diagram</b>	<b>19</b>
<b>3. System design</b>	<b>20</b>

<b>3.1. Description of procedures and function</b>	<b>20</b>
<b>3.2. Relation database schema</b>	<b>23</b>
<b>3.2.1. Tables</b>	<b>23</b>
<b>3.2.2. Attributes</b>	<b>23</b>
<b>3.2.3. Relations</b>	<b>25</b>
<b>3.3. Hardware and software requirements</b>	<b>25</b>
<b>3.4. Screen</b>	<b>26</b>
<b>4. Implementation</b>	<b>31</b>
<b>4.1 Introduction</b>	<b>31</b>
<b>4.2 Procedures</b>	<b>31</b>
<b>4.3 Reports</b>	<b>37</b>
<b>4.4 Layouts</b>	<b>38</b>
<b>4.5 Reports layouts</b>	<b>46</b>
<b>5. Conclusion and future work</b>	<b>48</b>
<b>6. References</b>	<b>49</b>
<b>7. Appendixes</b>	<b>50</b>

## Table of Figures

<b>Title</b>	<b>Number</b>
<b>Figure 1-1 Absher app</b>	<b>3</b>
<b>Figure 1-2 Mawid app</b>	<b>4</b>
<b>Figure 2-1 context Diagram</b>	<b>8</b>
<b>Figure 2-2 Overview diagram (level 0)</b>	<b>9</b>
<b>Figure 2-3 Detailed DFD</b>	<b>10</b>
<b>Figure 2-4 ERD diagram</b>	<b>12</b>
<b>Figure 2-5 Class Diagram</b>	<b>13</b>
<b>Figure 2-6 Use case diagram</b>	<b>14</b>
<b>Figure 2-7 Customer Use Case</b>	<b>15</b>
<b>Figure 2-8 Bank Use Case</b>	<b>16</b>
<b>Figure 2-9 Activity Diagram</b>	<b>17</b>
<b>Figure 2-10 State Diagram</b>	<b>18</b>
<b>Figure 2-11 Sequence Diagram</b>	<b>19</b>
<b>Figure 3-1 Customer Login and Registration</b>	<b>20</b>
<b>Figure 3-2 Book Appointment</b>	<b>21</b>
<b>Figure 3-3 bank employee</b>	<b>22</b>
<b>Figure 3-4 Login And Registration Screen</b>	<b>26</b>
<b>Figure 3-5 Customer Type and Service Information Screen</b>	<b>27</b>
<b>Figure 3-6 book appointment screen</b>	<b>28</b>
<b>Figure 3-7 Screen confirm appointment</b>	<b>29</b>
<b>Figure 3-8 Screen about all appointment for each customer</b>	<b>30</b>
<b>Figure 4-1 appointment Table</b>	<b>37</b>
<b>Figure 4-2 bank employee Table</b>	<b>37</b>
<b>Figure 4-3 Customer Table</b>	<b>37</b>
<b>Figure 4-4 main services Table</b>	<b>37</b>
<b>Figure 4-5 sub services Table</b>	<b>38</b>
<b>Figure 4-6 Welcome Screen</b>	<b>38</b>
<b>Figure 4-7 Customer Screen</b>	<b>39</b>
<b>Figure 4-8 Register Screen</b>	<b>39</b>
<b>Figure 4-9 Employee login Screen</b>	<b>40</b>
<b>Figure 4-10 Appointment for customer Screen</b>	<b>40</b>
<b>Figure 4-11 New appointment Screen</b>	<b>41</b>
<b>Figure 4-12 Select service Screen</b>	<b>41</b>

<b>Figure 4-13 Appointment screen</b>	<b>42</b>
<b>Figure 4-14 Time of the appointment Screen</b>	<b>42</b>
<b>Figure 4-15 Cancel appointment Screen</b>	<b>43</b>
<b>Figure 4-16 Booking procedures Screen</b>	<b>43</b>
<b>Figure 4-17 change password Screen</b>	<b>44</b>
<b>Figure 4-18 User management Screen</b>	<b>44</b>
<b>Figure 4-19 service management Screen</b>	<b>45</b>
<b>Figure 4-20 appointment management Screen</b>	<b>45</b>

## Table of Tables

<b>Title</b>	<b>Number</b>
<b>Table 1-1 comparison</b>	<b>4</b>
<b>Table 3-1 customer Table</b>	<b>23</b>
<b>Table 3-2 Bank employee Table</b>	<b>23</b>
<b>Table 3-3 Appointment Table</b>	<b>24</b>
<b>Table 3-4 Main Services Table</b>	<b>24</b>
<b>Table 3-5 sub Services Table</b>	<b>24</b>

# **1. Introduction**

## **1.1. Overview**

Nowadays, technology is taking a major role in almost every aspect of our life, especially when it comes to any office or work that is done manually. It's not acceptable anymore and money and effort wasting to keep doing work manually in any field, like banks.

Any bank has a wide range of services to offer for its customers. These services need to be tracked and achieved in a way that saves money, effort and time. The important work is to make is customer appointments managing to be easily and conveniently to save time and effort to deliver services in a seamless manner.

The bank need to build appointment management application that is capable to provide functionality to facilitate the customers appointment management and organizing. Bank Customers Appointments app is a simple application for processing electronic appointments where the customer can book appointments and check their availability through the application and choose the service they want in easy way without any effort.

### **1.1.1. Abstract system description**

Bank Customers Appointments app will specially design for banks to enable both banks employs and customers to accomplish their daily work efficiently. By offering management features for the customers appointments, Users must first register in app using ID number and name. Then book appointments to benefit from the services available in several classifications so that the user can book an appointment either is customers or not, so that the app displays the times available for booking according to the service chosen by the user. The user will then book the appropriate time. Also user can be able to cancel the appointment if needed it's.

## **1.2. Problem definition**

The banking sector generally has a large segment of customers. Customers perform many procedures inside the bank daily, which in most times caused overcrowding inside bank and leads to wasting time for the customer. The customer may wait for several hours if the bank is overcrowded. To solve the problem, Bank Customers Appointments app prepared to organize appointments.

### **1.2.1. Goals**

The goal of project It optimizes customer appointment booking experience by managing appointment scheduling from mobile apps to offers bank's customers a convenient way to save time and effort.

### **1.2.2. Literature review**

We study some application list below tom make comparisons between them.

#### **Absher app**

Absher is the official individual's eservices Mobile Application that provide the services of Absher portal in Kingdom of Saudi Arabia [1]. The main screen of Absher app is shown by the following Figure 1.1

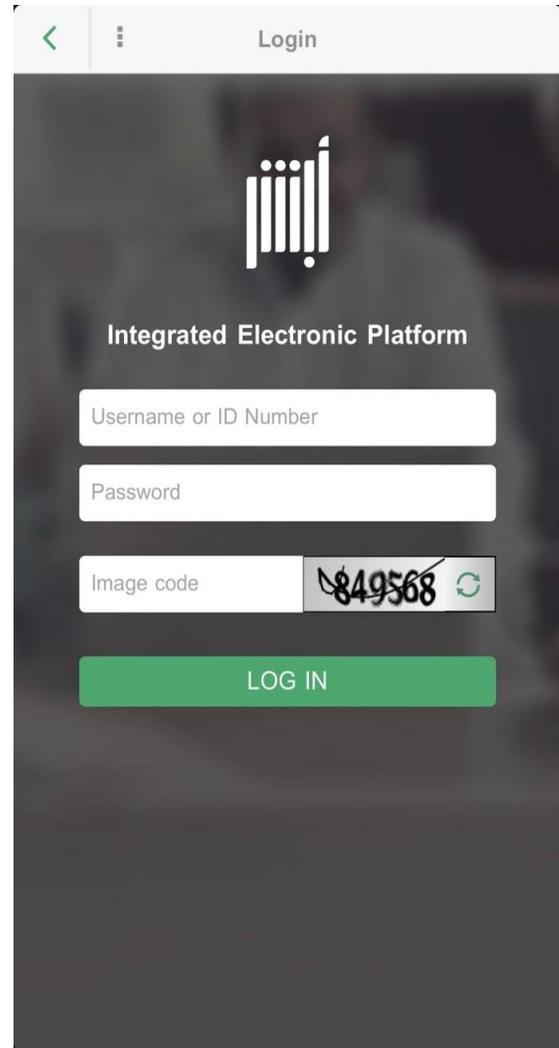


Figure 1-1 Absher app

## **Mawid**

The centralized appointments system (Mawid), is an electronic service provided by Saudi ministry of health, to enable patients to book their appointments across primary health care centers and manage them by cancelling or rescheduling as well as managing their referral appointments [2].

The main screen of Mawid app is shown by the following Figure 1-2



Figure 1-2 Mawid app

**Comparison between similar works and the current project**

	<u>Absher</u>	Mawid	Bank Customers
<b>SCREENS LAYOUT</b>	SIMPLE	Complicated	Simple
<b>Arabic Interface</b>	Arabic and English interface	Arabic and English interface	Arabic and English interface
<b>Showing Appointment</b>	No	Yes	Yes

Table 1-1 comparison

### **1.2.3. Collect of data**

The questionnaire method has been adopted to gather information. iT was prepared consisting of 7 questions focused on the benefit of having an application to book bank appointments and problems encountered by the customers with no such application, A sample of 25 persons was taken. In the appendix there is an analysis of the results of the questionnaire.

The results of the questionnaire are:

92% of participants face a problem with bank appointments

96% face a problem with bank appointments spend a long time waiting for appointments

92% would like to choose a suitable date to complete their banking operations

96% would like reservation appointments

96% believe that having an application to record appointments will be beneficial

96% believe that an application to record appointments will organize appointments

92% believe that an application for registration of appointments will complete the banking operations in a standard manner

### **1.2.4. Objectives**

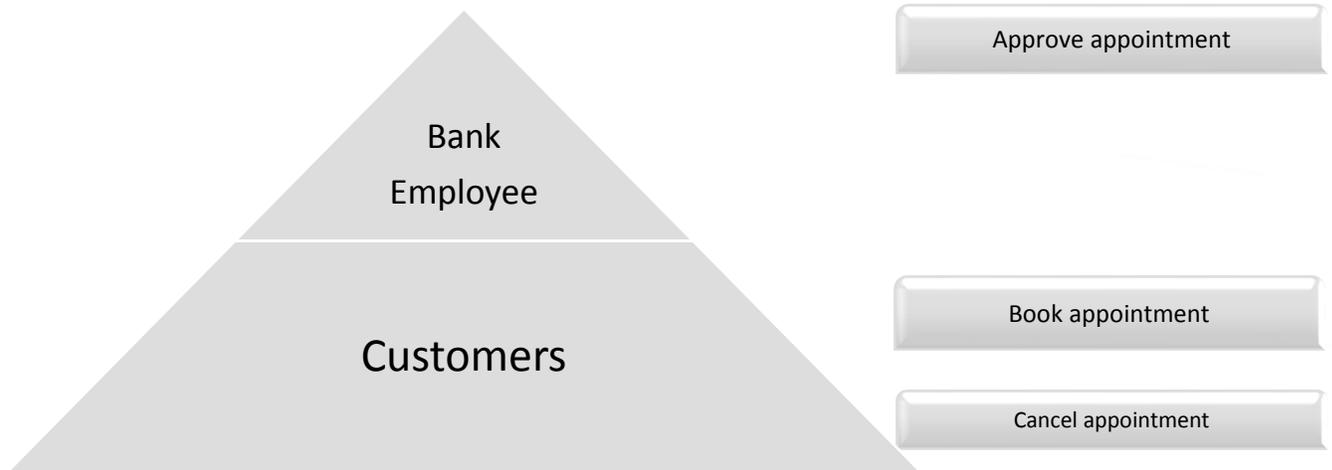
Bank appointment App will be able to achieve the following objectives:

- Automate the manual procedure of scheduling appointment
- Providing an easy to use unified scheduling platform for all customer.
- Increases customer satisfaction.
- Provide a mobile application to book customers' appointments at the bank.
- View some services that may not require customers to go to the bank and can be made through the ATM

### **1.2.5. Critical success factors**

The system will provide customers with the ability to book appointments in an easy and simple way

### **1.2.6. Organization chart and responsibilities**



### **1.3.General rules (assumptions)**

- The customers just need know the basic of mobile application

The application must achieves the following nonfunctional Requirements where is a requirement that specifies criteria that can be used to judge the operation of a system, rather than specific behaviors

- Usability :The application provides ease of handling through easy and flexible screens
- Reliability: The application provides reliability so that it responds appropriately.
- Availability: The application must be accessible seven days a week for 24 hours a day.

## **2. System analysis**

### **2.1.Introduction**

System analysis is defined as an understanding of its components, processes and problems, and therefore needs. Elements of the system and what is expected to be implemented by the system, some people call this stage the diagnosis stage of the system or smart phase in the context of system development processes.

Systems analysis examines the components of the systems, their modus and their association with the internal and external entities, and the type of relationship that links the system at each level with its main and subsidiary entities.

### **2.2.Description of Data Flow Diagram (DFD)**

Data Flow It is a two-dimensional diagram that explains how data is processed and transferred in a system. The graphical depiction identifies each source of data and how it interacts with other data sources to reach a common output.

Four simple notations are used to complete a DFD. These notations are given below: [3]

**Data Flow:** - The data flow is used to describe the movement of information from one part of the system to another part. Flows represent data in motion. It is a pipe line through which information flows. Data flow is represented by an arrow.

**Process:** - A circle or bubble represents a process that transforms incoming data to outgoing data. Process shows a part of the system that transform inputs to outputs.

**External Entity:** - A square defines a source or destination of system data. External entities represent any entity that supplies or receive information from the system but is not a part of the system.

**Data Store:** - The data store represents a logical file it is a repository of data that is to be stored for use by one or more processes. It is represented by open rectangle.

### 2.2.1. Context Diagram

Context is a brief structure that describes the environment in which a system exists and helps in communicating about what lies outside the system boundary. Context Diagram is summarizing all processing activity within the system in single process symbol, it describes highest level view of a system, the whole system is represented as one process and showing the data flows that pass between the external entities and the system. [4]

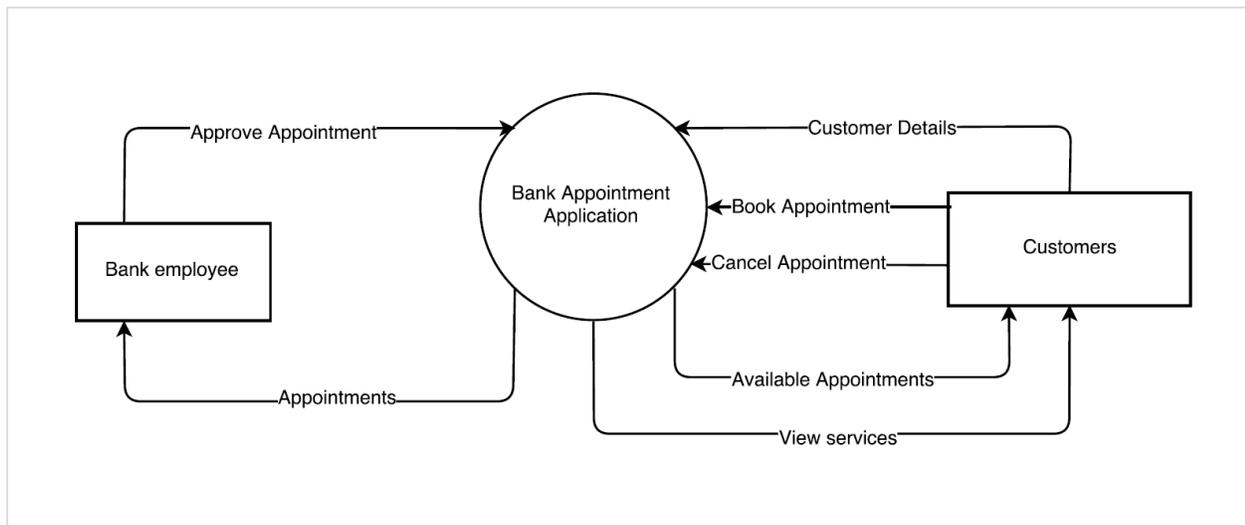


Figure 2-1 context Diagram

### 2.2.2. Overview diagram (level 0)

Level 1 DFD's aim to give an overview of the full system. They look at the system in more detail. Major processes are broken down into sub-processes. Level 1 DFD's also identifies data stores that are used by the major processes [5]

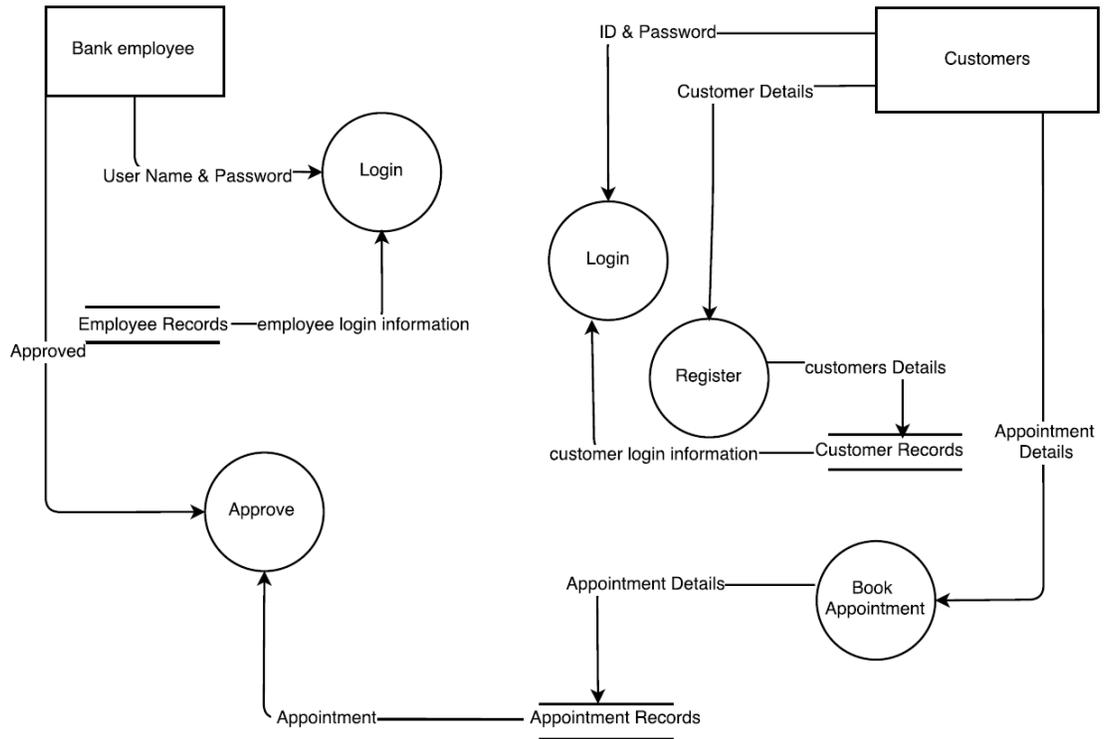


Figure 2-2 Overview diagram (level 0)

### 2.2.3. Detailed DFDs

The Detailed DFD provides a more detailed and comprehensive view of the interaction among the sub-processes within the system.

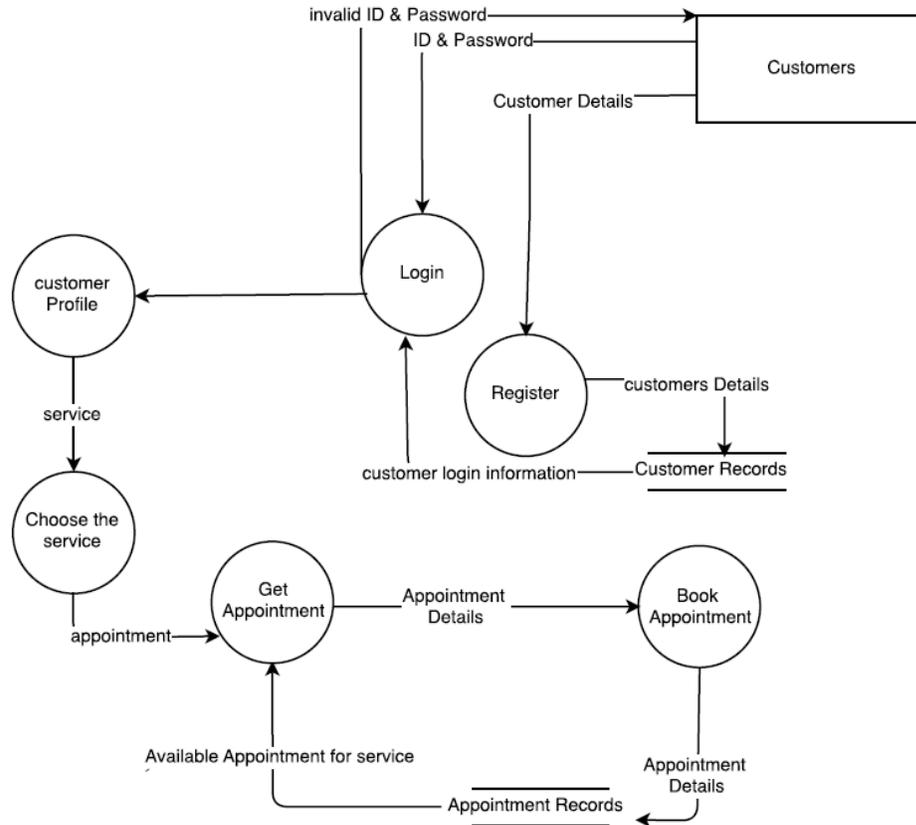


Figure 2-3 Detailed DFD

### 2.3. Entity Relationship Diagram (ERD)

An entity relationship diagram helps illustrate the conceptual design of the database showing high level abstractions, relationships and cardinalities between each relation. Each entity in the database should also have a unique identifier to identify specific instances of each entity[3].

An ER diagram also helps to model the relationship between each entity.

### **2.3.1. Description of Entities**

The application entities are customers, bank employee and appointments every entity has many attributes.

- customers entity is present customers information.
- bank employee is present bank employee information.
- Appointments is present customer appointments information.
- Services is present the services representing the services provided by the Bank

### **2.3.2. Description of relations**

- The customers can able to book one or many appointments in different time.
- The bank employee approved one or more appointments.
- The appointment is for specific service.

### **2.3.3. Drawing ERD.**

The ERD serves is a blue print from which a relational database my be deduced. Figure below shows the ERD for our application.

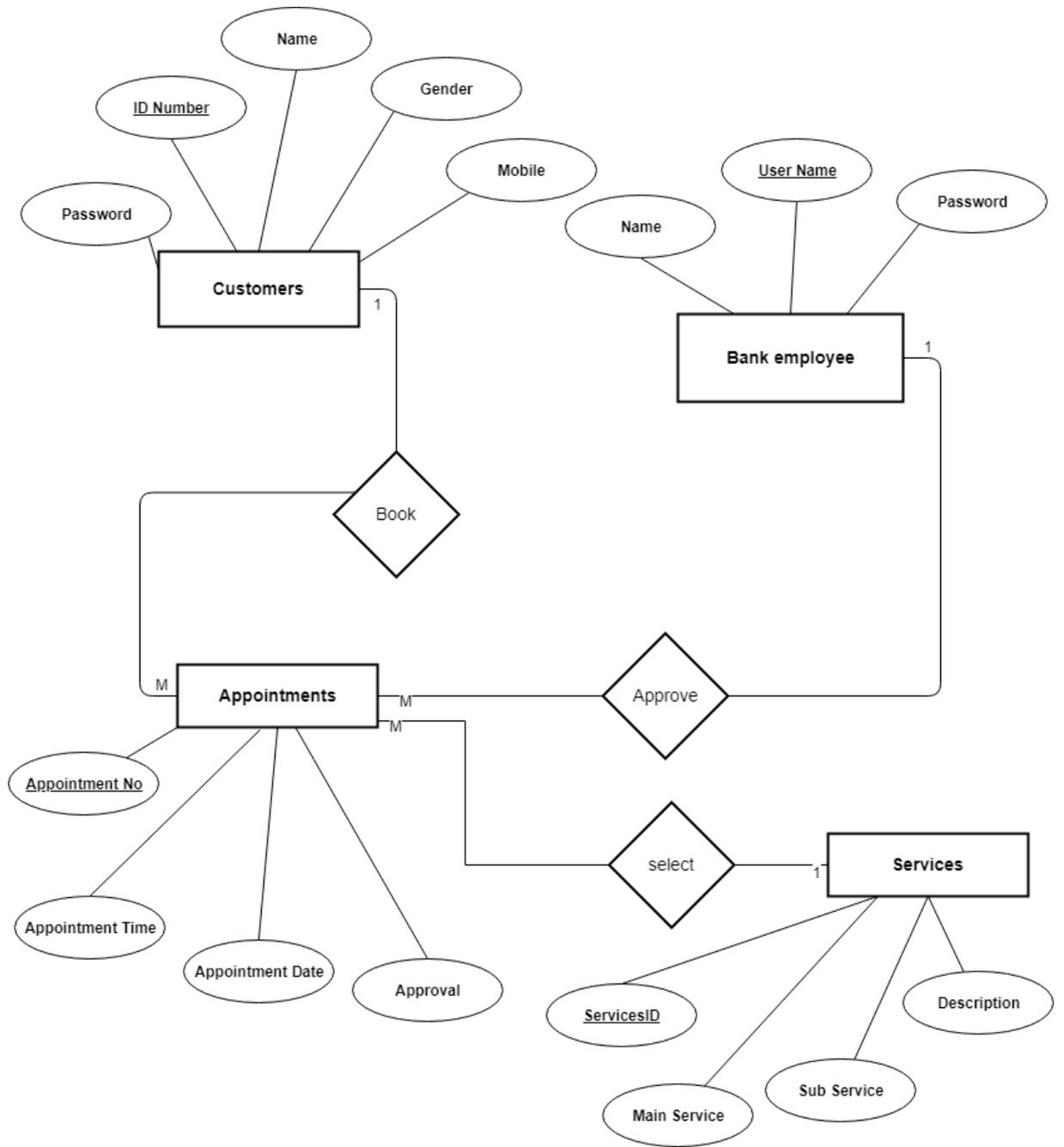


Figure 2-4 ERD diagram

## 2.4. Structure Diagram

### 2.4.1. Class Diagram

class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among objects[4].

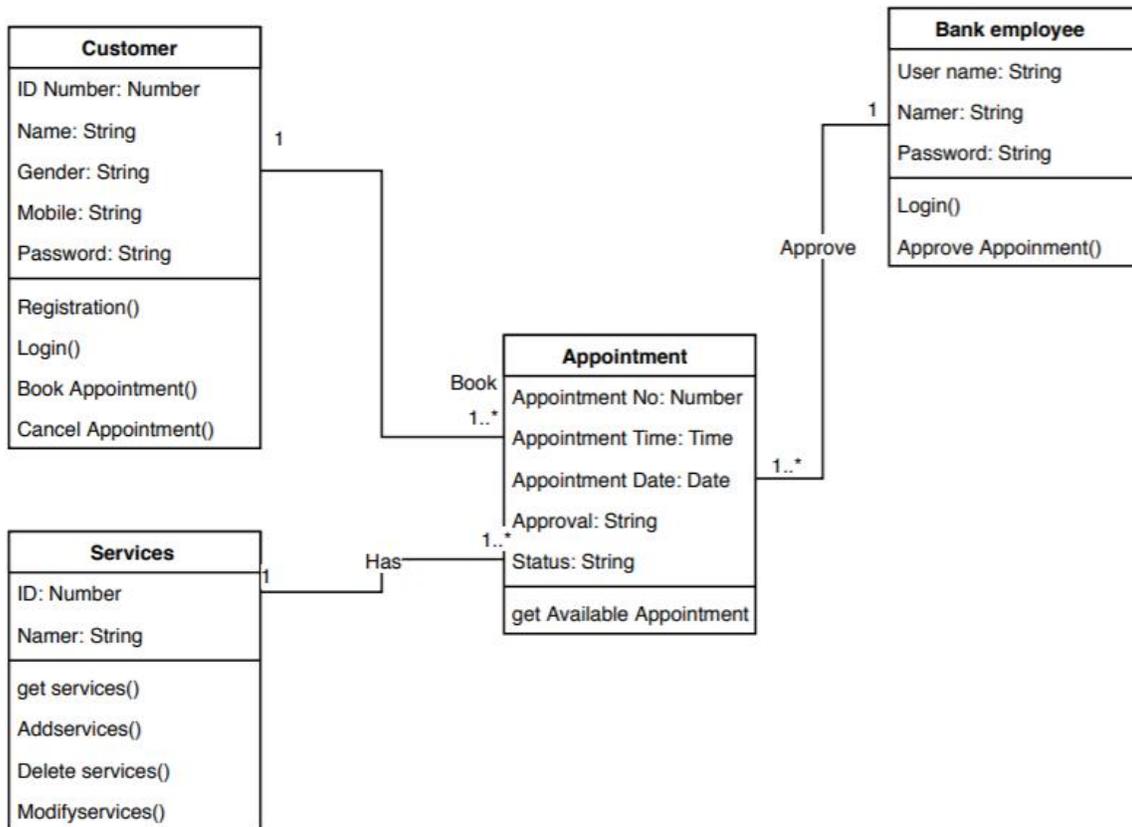


Figure 2-5 Class Diagram

## 2.5. Behavior diagram

### 2.5.1. use case Diagram

a use case diagram can summarize the details of your system's users (also known as actors) and their interactions with the system [6].

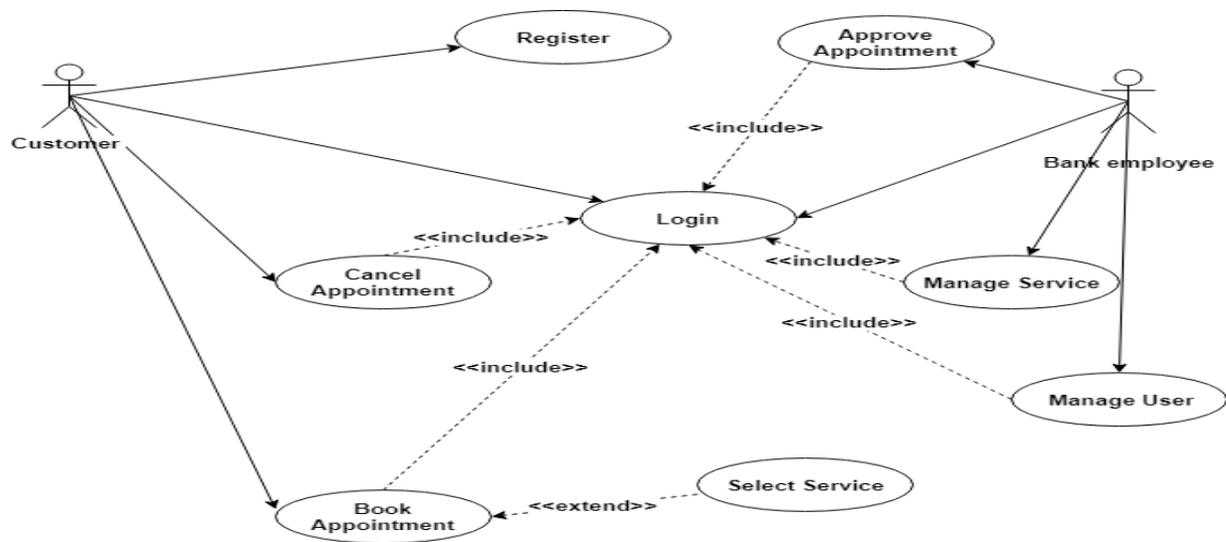


Figure 2-6 Use case diagram

a) Customer Use Case

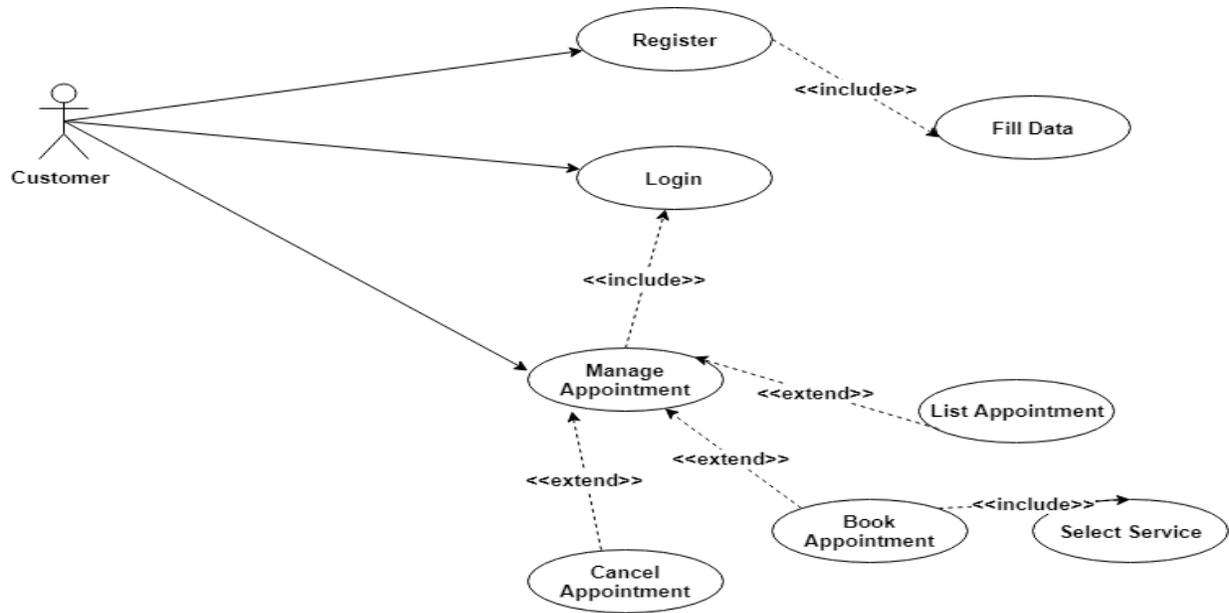


Figure 2-7 Customer Use Case

b) Bank employee Use Case

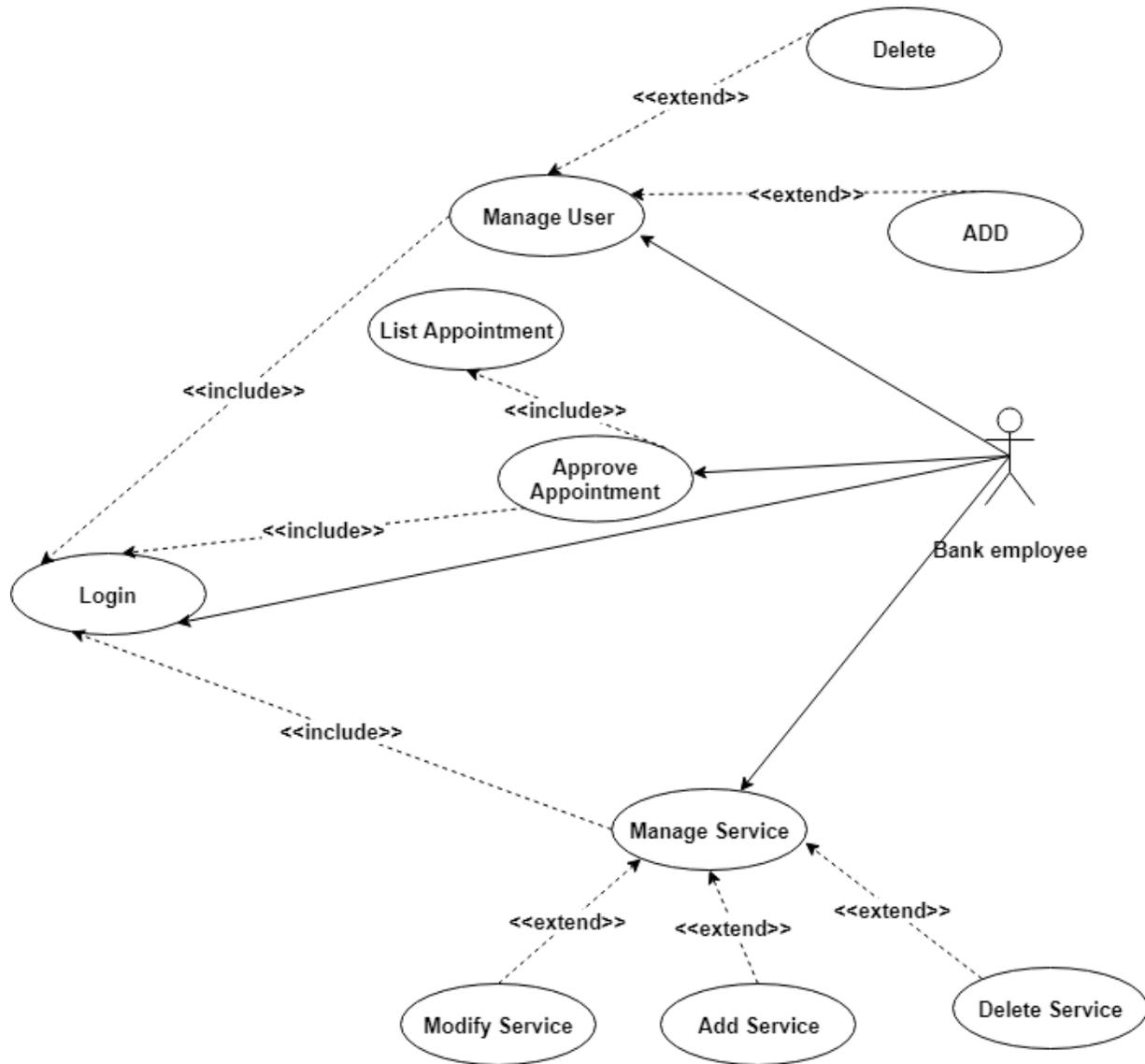


Figure 2-8 Bank Use Case

### 2.5.2. Activity Diagram

Activity diagram is basically a flowchart to represent the flow from one activity to another activity. The activity can be described as an operation of the system.

The control flow is drawn from one operation to another. This flow can be sequential, branched, or concurrent. Activity diagrams deal with all type of flow control by using different elements such as fork, join, etc[7].

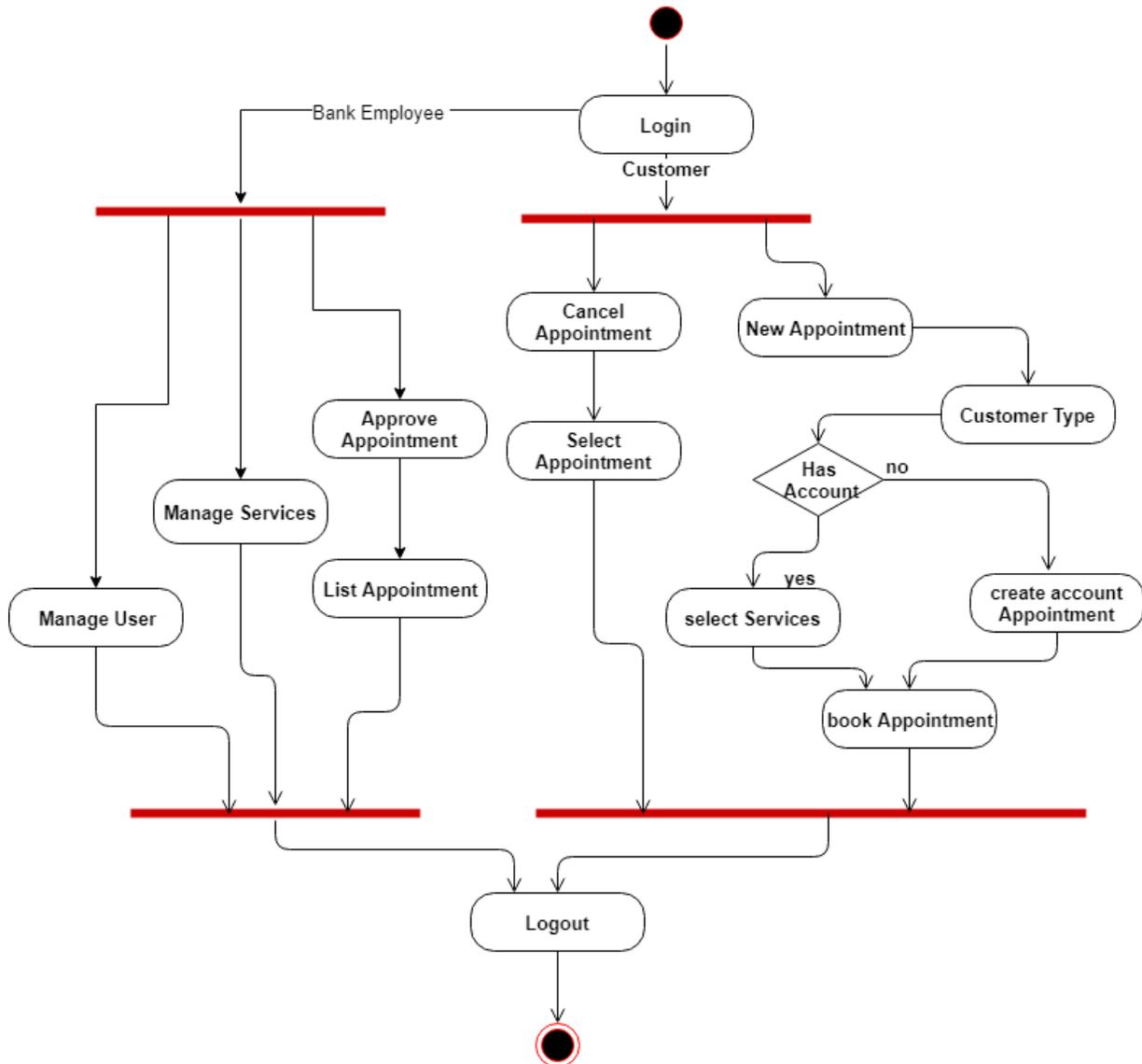


Figure 2-9 Activity Diagram

### 2.5.3. State Diagram

State machine diagram is a behavior diagram which shows discrete behavior of a part of designed system through finite state transitions. State machine diagrams can also be used to express the usage protocol of part of a system [8].

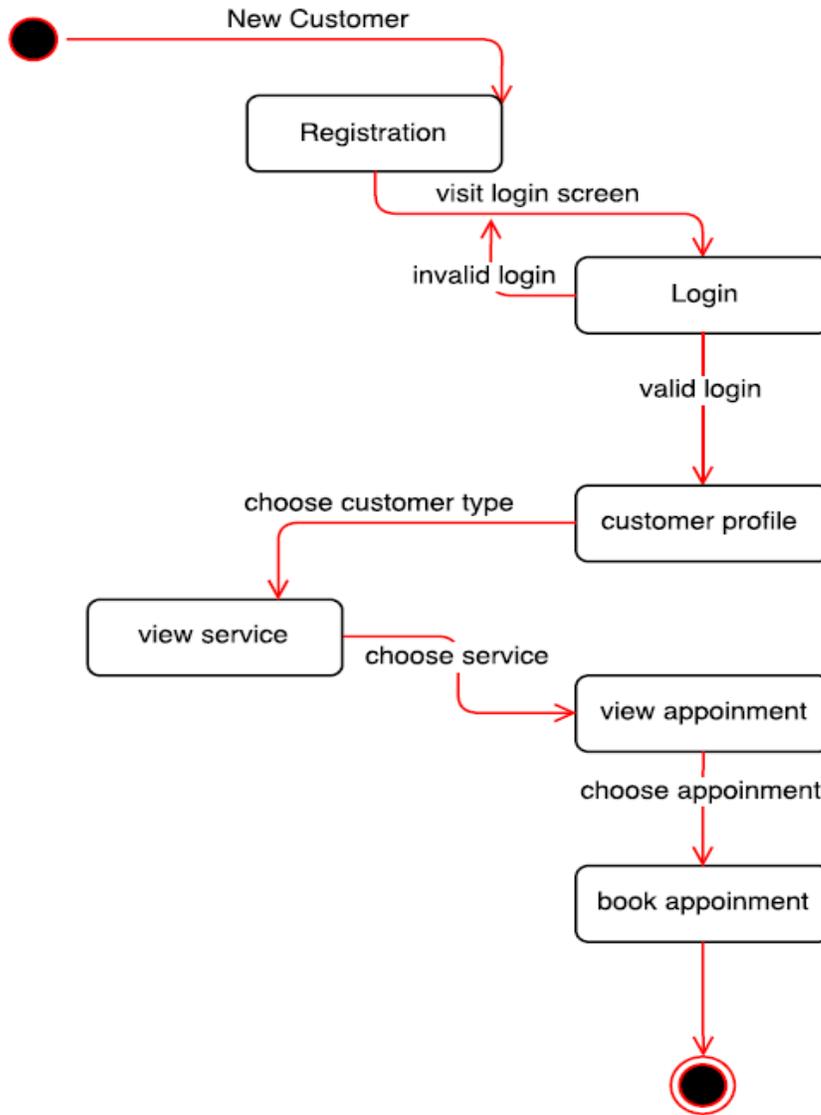


Figure 2-10 State Diagram

## 2.6. Interaction Diagram

### 2.6.1. Sequence Diagram

Sequence diagrams describe interactions among classes in terms of an exchange of messages over time. They're also called event diagrams. A sequence diagram is a good way to visualize and validate various runtime scenarios [9].

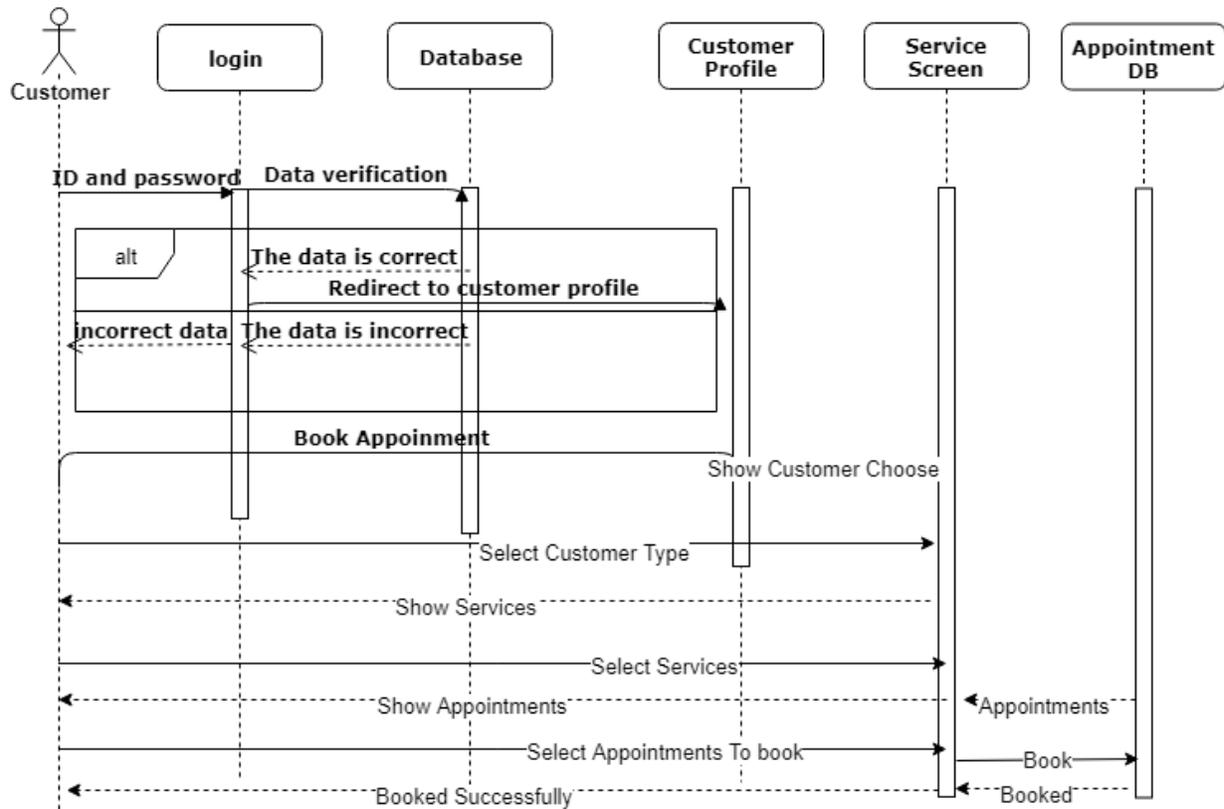


Figure 2-11 Sequence Diagram

### 3. System design

System design is the process of defining the elements of a system this phase lays a fundamental for actual programming and implementation to satisfy specific needs and requirements [11].

#### 3.1. Description of procedures and function

We choose the main function to describes Customer Login and Registration, Book

Appointment and bank employee.

- Customer Login and Registration:

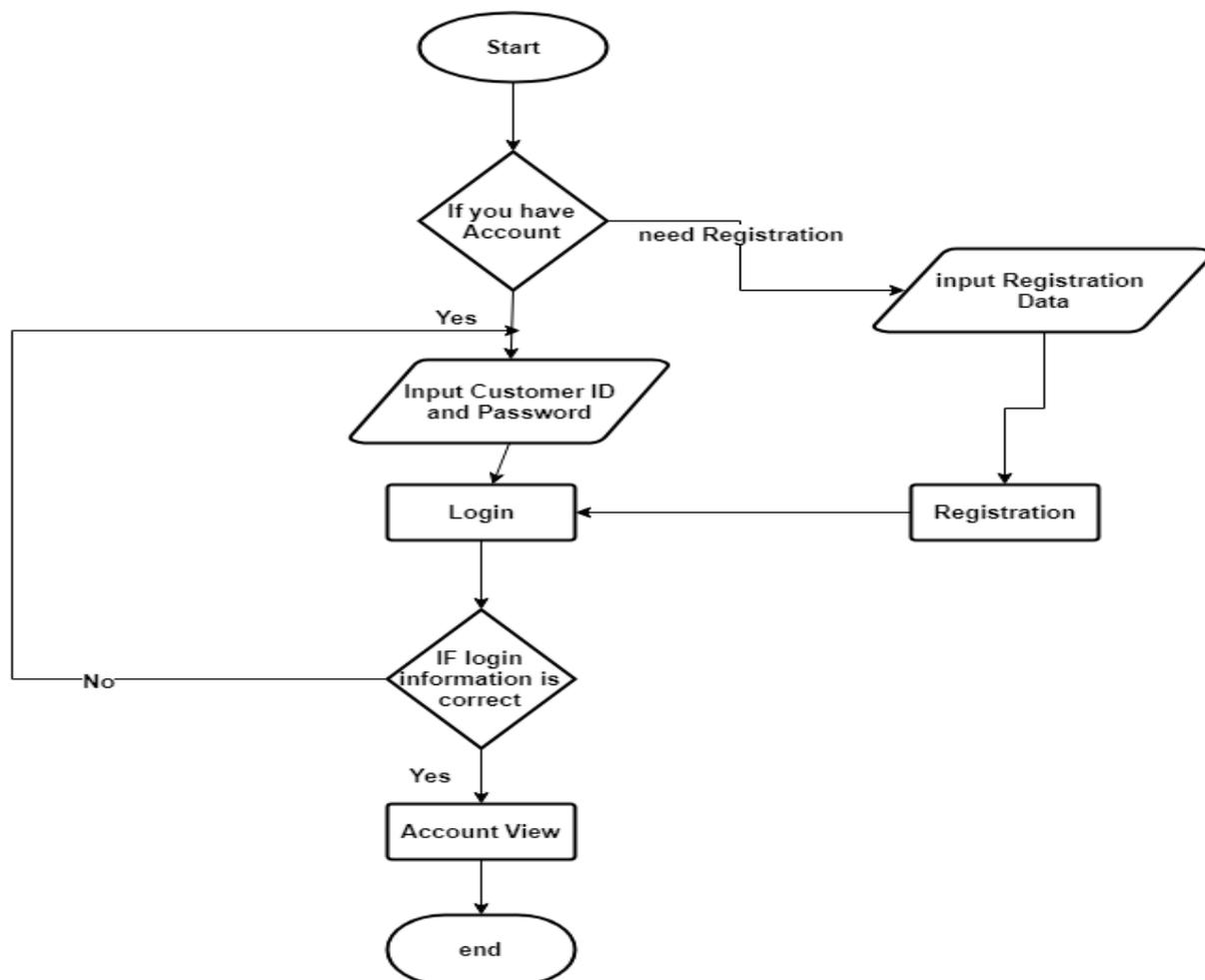


Figure 3-1 Customer Login and Registration

- Book Appointment

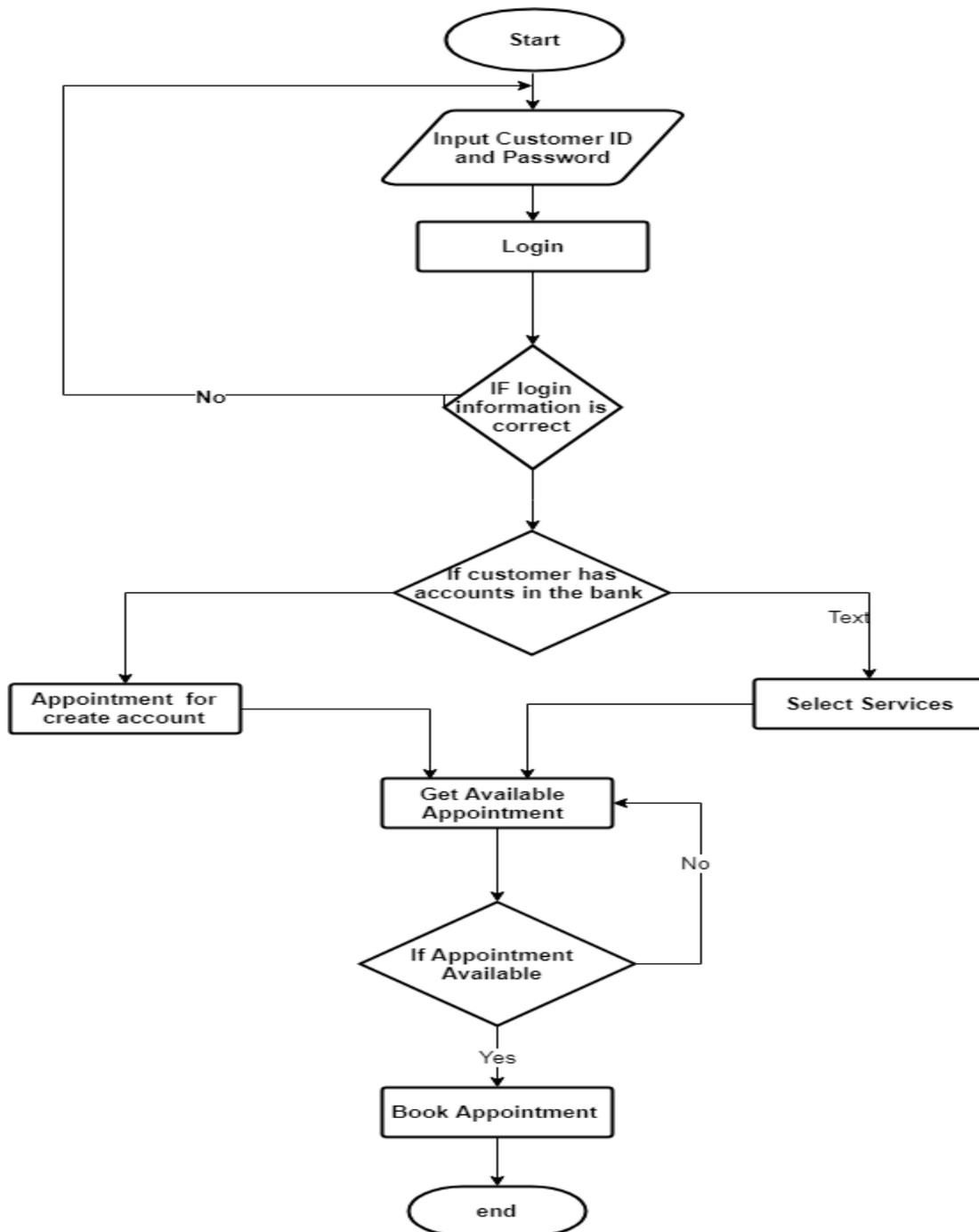


Figure 3-2 Book Appointment

- Flowchart bank employee

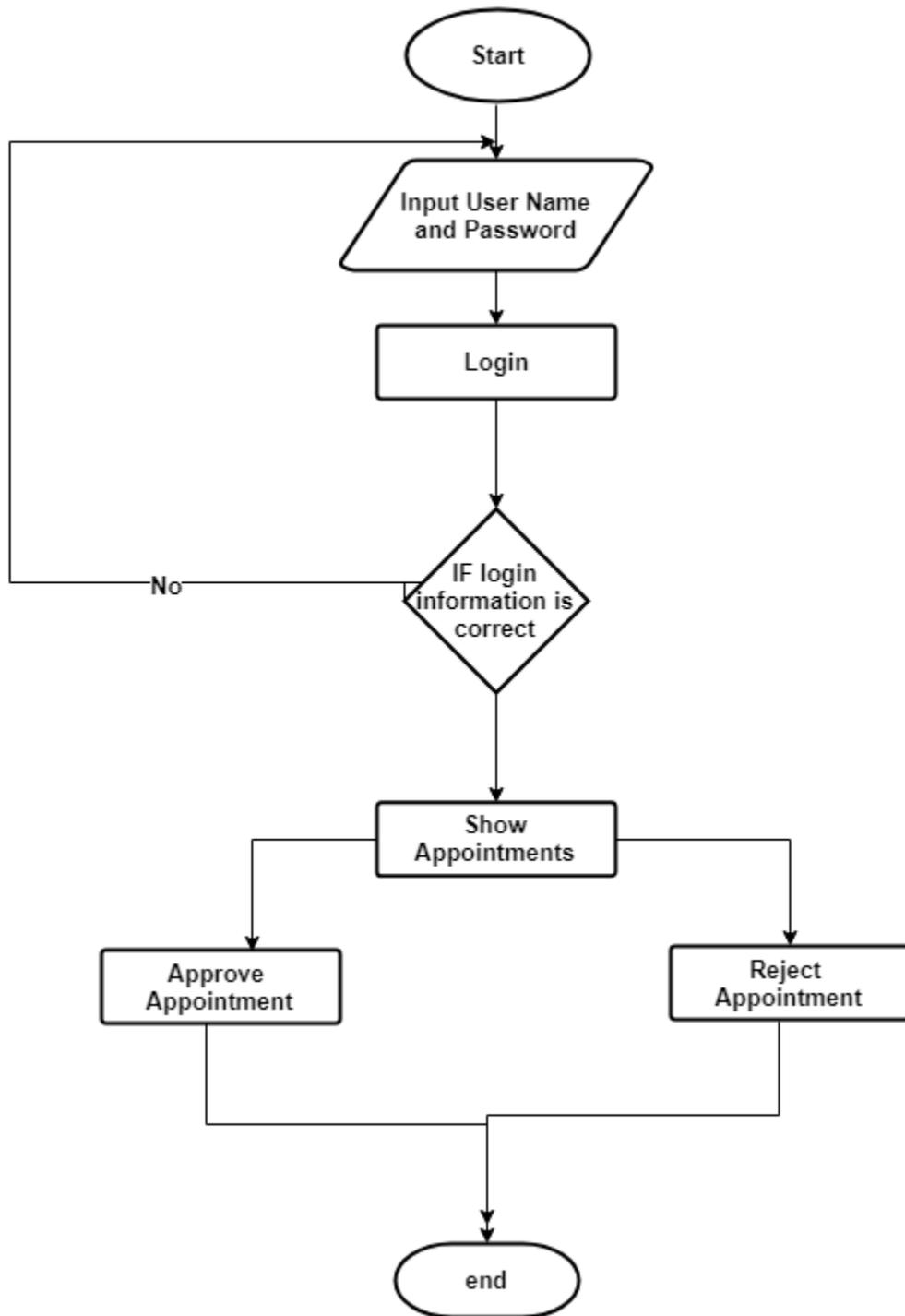


Figure 3-3 bank employee

## **3.2.Relation database schema**

### **3.2.1. Tables**

- Customer Table
- Bank employee Table
- Appointments Table
- Main Services Table
- Sub Services Table

### **3.2.2. Attributes**

- Customer Attribute

<b>Attribute Name</b>	<b>Data Type</b>	<b>Key</b>
ID Number	Number	PK
Name	String	
Gender	String	
Mobile	String	
Password	String	

Table 3-1 customer Table

#### Bank employee Attribute

<b>Attribute Name</b>	<b>Data Type</b>	<b>Key</b>
User name	String	PK
Name	String	
Password	String	

Table 3-2 Bank employee Table

## Appointment Attribute

<b>Attribute Name</b>	<b>Data Type</b>	<b>Key</b>
Appointment No	Number	PK
Service Id	Number	FK
Appointment Time	Time	
Appointment Date	Date	
Approval	String	
Status	String	
Customer ID	Number	FK
User name	String	FK

Table 3-3 Appointment Table

## - Main Services Attribute

<b>Attribute Name</b>	<b>Data Type</b>	<b>Key</b>
Service ID	Number	PK
Service Description	String	

Table 3-4 Main Services Table

## - Sub Services Attribute

<b>Attribute Name</b>	<b>Data Type</b>	<b>Key</b>
Service ID	Number	PK
Service Description	String	
Main Service	Number	FK

Table 3- 5 sub Services Table

### **3.2.3. Relations**

There are three types of relationship that can be modelled: one-to-one (1:1); one-to-many (1:M); or many-to-many (N:M).

- The relationship between customer and Appointment one-to-many (1:M) The customers can able to book one or many appointments in different time.
- The relationship between Appointment and Sub Services is one-to-many (1:M) the Appointment it specific for on service.
  - The bank employee approved one or more appointments.
  -
- The relationship between Main Service and Sub Service is one to many.

### **3.3. Hardware and software requirements**

To Implement the application, we needed the following hardware's and software's

#### **3.3.1. Hardware Requirements**

- Mobile Device to test app
- Pc that stand for server to database

#### **3.3.2. Software Requirements**

- Database Managing System: MySQL
- Tool to access the database: phpMyAdmin
- Tool to build application: Android Studio

### 3.4. Screen

The most important screens in bank appointment application are

The following screens or interfaces are for making registration of the customers as a first use of our services and the second one for making logging into our system after having an account in the system.

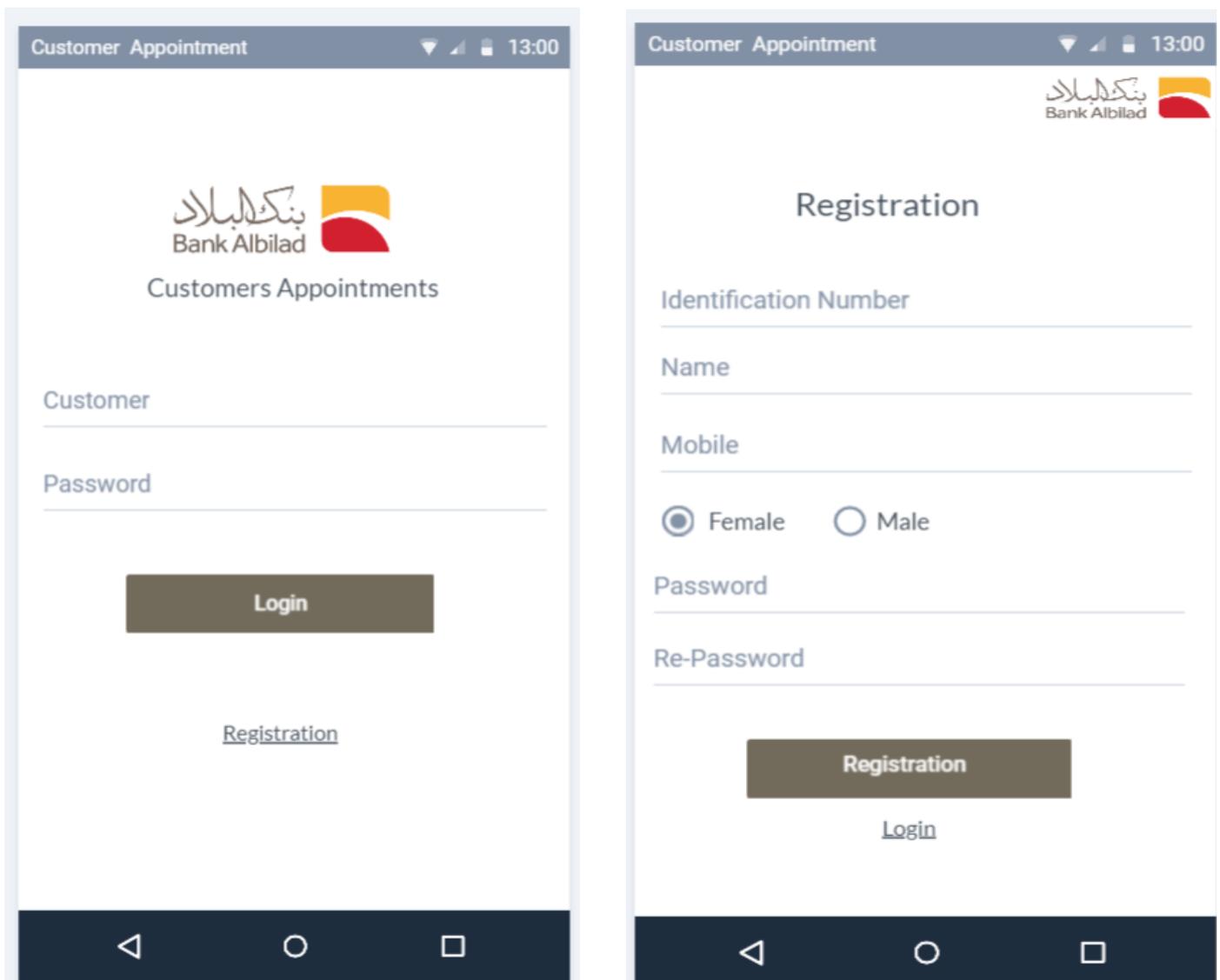


Figure 3-4 Login And Registration Screen

The following screens are one for asking the customer if he/she has account to move the user into the login screen or if he/she has not account then it will move the user to the registration screen to have new account.

The second screen will supply the user with the services of our system, which support the user through using the system.

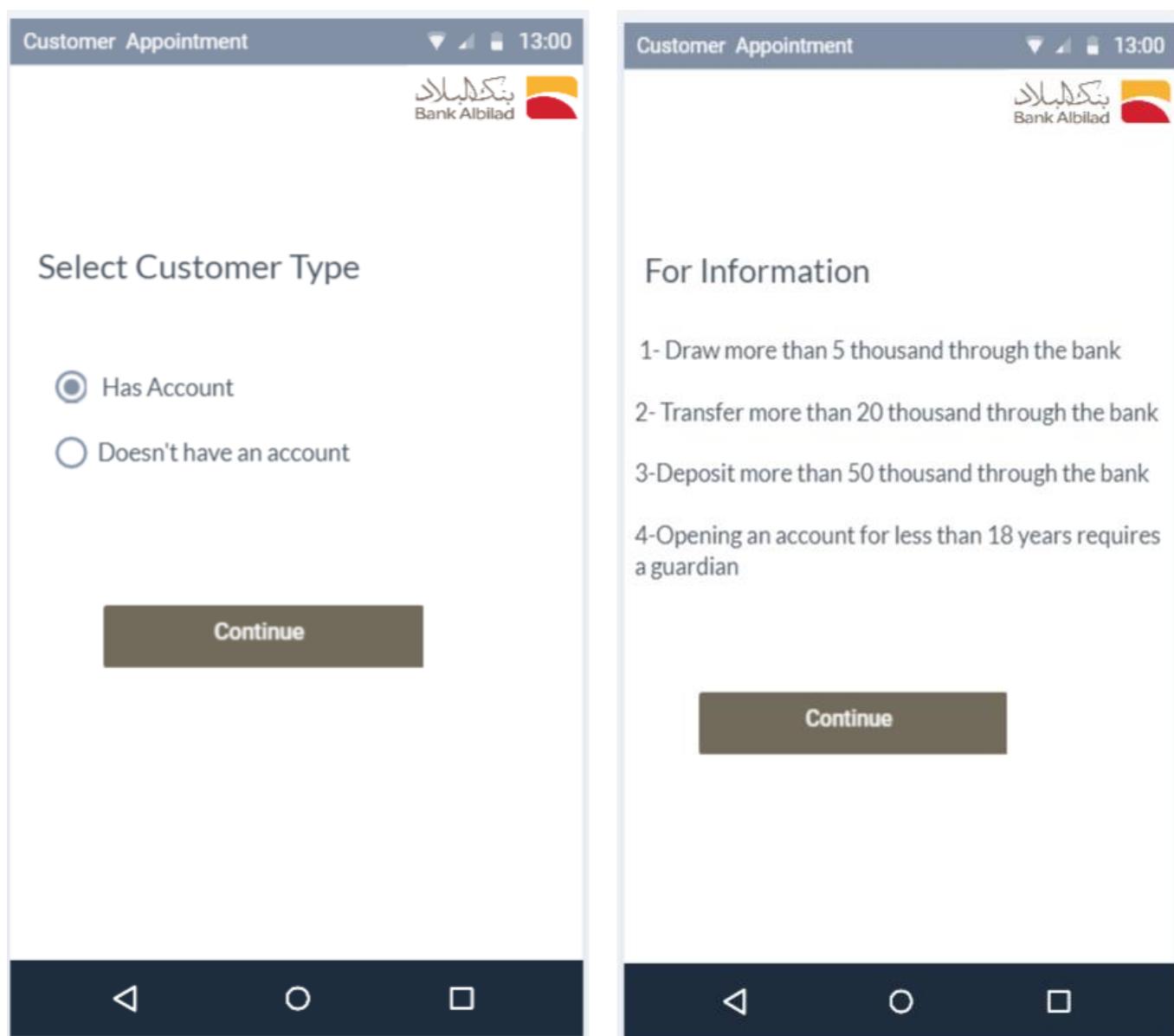


Figure 3-5 Customer Type and Service Information Screen

The Figure 3-6 book appointment screen

This interface will help the customer to determine the main service and the sub service to show him the appointment.

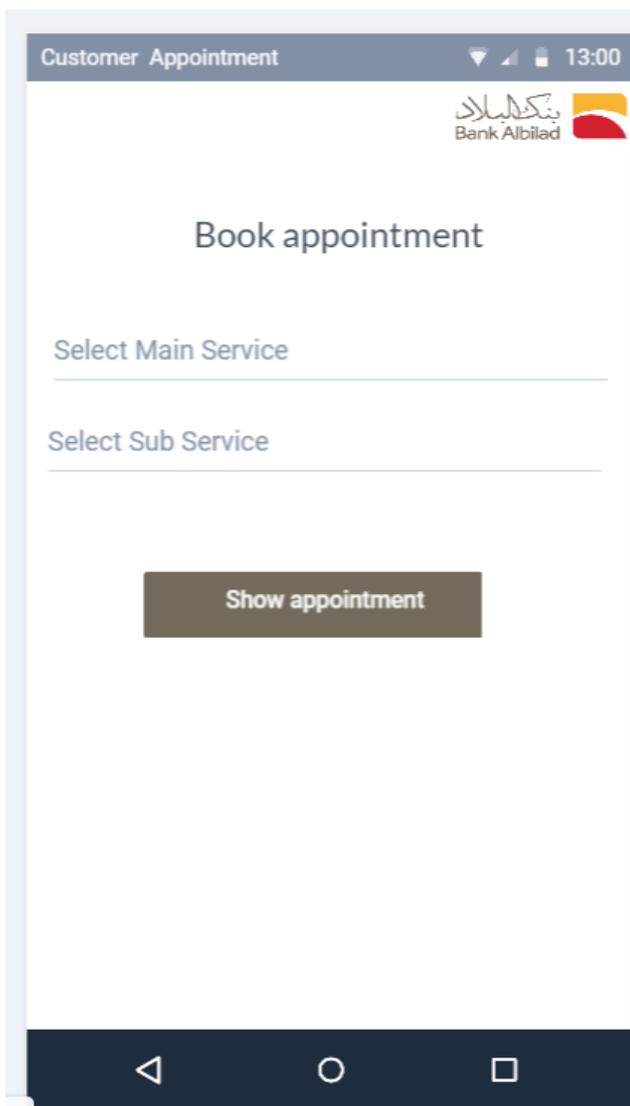


Figure 3-6 book appointment screen

The Figure 3-7 Shown The screen the show approval or reject customer .

This screen will show the administrator who is (the customer with date and time) asked for showing the appointment and then the administrator can making confirm of the demand or reject it.

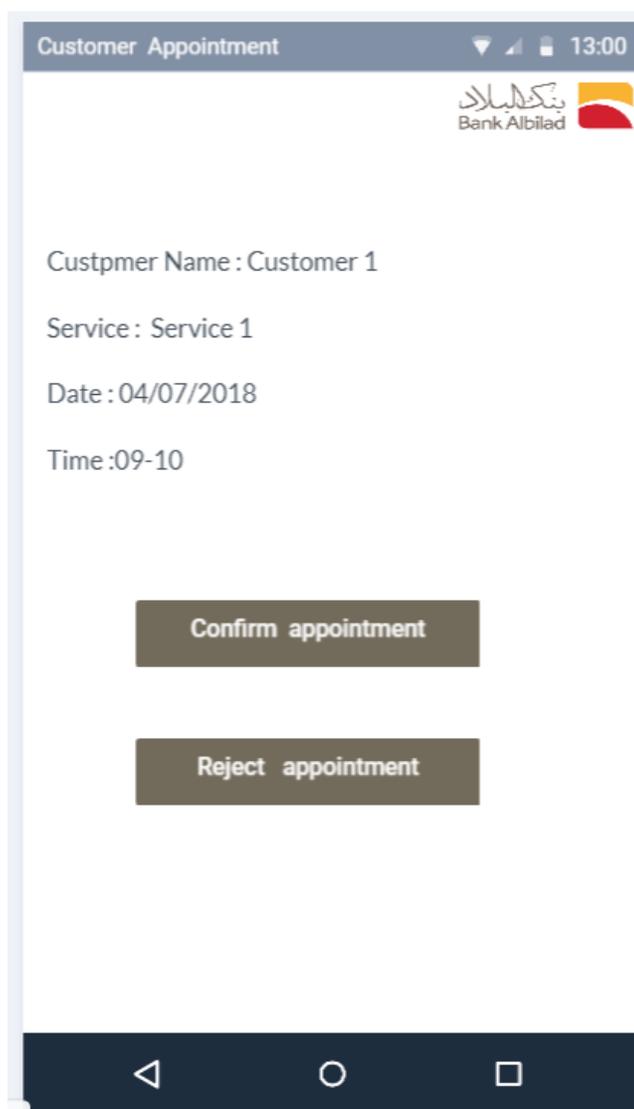


Figure 3-7 Screen confirm appointment

The Figure 3-8 Shown The screen the show customer appointments details.

This screen will represent all done appointments for a specified user in the system.

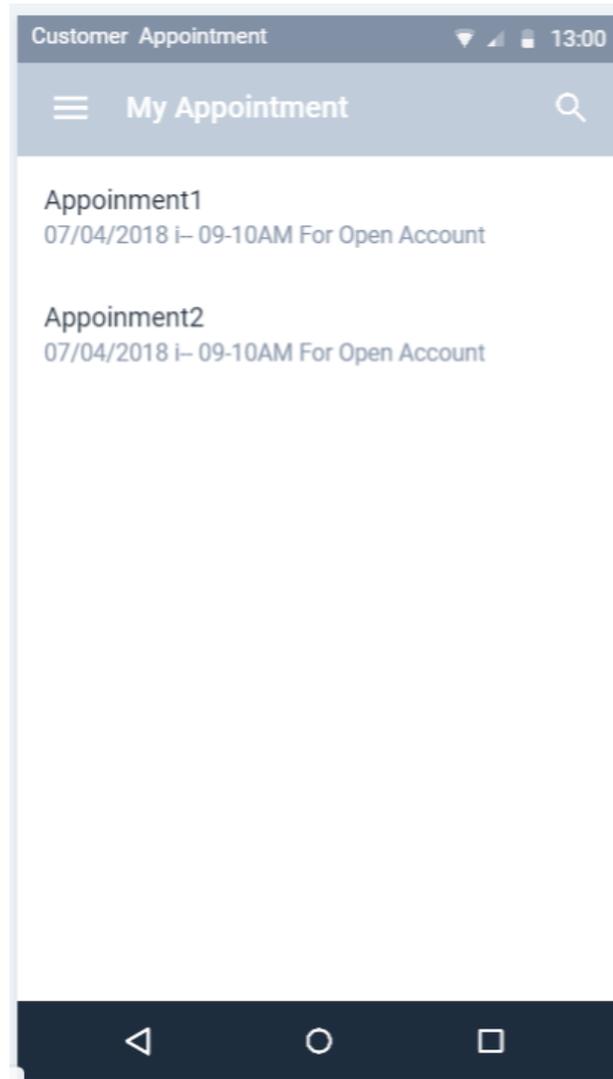


Figure 3-8 Screen about all appointment for each customer

## 4. Implementation

### 4.1 Introduction

This project is a mobile application for the bank, it provides many services for clients like register , reserve appointment, cancel the appointment, and the client can reserve more than one appointment; the employee in the bank can accept or reject the appointments and manage user. We use the MySQL to build the database and android studio for writing the java code.

### 4.2 Procedures

We have many procedures in the project as like:

1- Login

```
2- public login(String aa,String bb,SharedPreferences s)
    {
        a=aa; b=bb;spre=s;
    }
loginbut.setOnClickListener(
    new View.OnClickListener()
    {
        public void onClick(View view) {

            isEmpty(Password);
            isEmpty(Customerid);
            if (cancel) {

                focusView.requestFocus();

            } else {

                new login(Customerid.getText().toString(),
                Password.getText().toString(), spre).execute();
            }

        }
    });
protected void doInBackground(Void... voids) {

    try {

        Class.forName("com.mysql.jdbc.Driver");
        Connection con =connectionstirng.getConnection();
        PreparedStatement st=con.prepareStatement("SELECT
IDNumber, Name, Gender, Mobile, Password from customers where
IDNumber=? and Password=?");
        st.setString(1, a);
        st.setString(2, b);
```

```

        ResultSet rs = st.executeQuery();
        if (rs.next()) {

            SharedPreferences.Editor editor = spre.edit();
            editor.putString("CustomerId", rs.getString(1));
            editor.putString("Password", rs.getString(5));
            editor.putString("Name", rs.getString(2));
            editor.putString("Type", "Customer");
            editor.apply();
            Intent i = new Intent(loginActivity.this,
MainCustomerActivity.class);
            i.addFlags(Intent.FLAG_ACTIVITY_CLEAR_TOP |
Intent.FLAG_ACTIVITY_NEW_TASK);
            startActivity(i);

        } else
        {
            runOnUiThread(new Runnable() {
                @Override
                public void run() {

Password.setError(getString(R.string.error_incorrect_password));
                Password.requestFocus();
            }
        });
    }
}

} catch (Exception e) {
    e.printStackTrace();
}

return null;
}

```

### 3- Register

```

public Registrater( String Customerid ,String Password ,String
Name ,
                    String Mobile ,String Gender)
{
    this.Customerid=Customerid;
    this.Passwordt=Password ;
    this.Name=Name ;
    this.Mobile=Mobile ;
    this.Gender=Gender;
    ProgressDialog progressDialog;
}
@Override
protected Void doInBackground(Void... voids) {

    try {

        Class.forName("com.mysql.jdbc.Driver");
        Connection con =connectionstirng.getConnection();

```

```

PreparedStatement st=con.prepareStatement("INSERT INTO
customers (IDNumber,Name,Gender,Mobile,Password)
VALUES"+"(?,?,?,?,?)");
st.setString(1, Customerid);
st.setString(2, Name);
st.setString(3, Gender);
st.setString(4, Mobile);
st.setString(5, Password);
int rs = st.executeUpdate();
if (rs>0) {

    int time = Integer.parseInt("2")*1000;

    try {
        Thread.sleep(time);
    } catch (InterruptedException e) {
        e.printStackTrace();
    }
    new Handler(Looper.getMainLooper()).post(new
Runnable() {

                                                                    @Override
                                                                    public

void run() {
        Toast.makeText(getApplicationContext(),
R.string.SucussfullyRegistrtion, Toast.LENGTH_LONG).show();

                                                                    }
                                                                    });
        Intent i = new Intent(RegistrationActivity.this,
MainInformationActivity.class);
        i.addFlags(Intent.FLAG_ACTIVITY_CLEAR_TOP |
Intent.FLAG_ACTIVITY_NEW_TASK);
        startActivity(i);

    } else
    {
        runOnUiThread(new Runnable() {
            @Override
            public void run() {
                Password.setError("Incorrect Password");
                Password.requestFocus();
            }
        });
    }
}
catch (Exception e)
{
    if (e.getMessage().toString().contains("Duplicate entry"))
    {
        runOnUiThread(new Runnable() {
            @Override
            public void run() {
                Customerid.setError("The Customer Already
exist");
                Customerid.requestFocus();
            }
        });
    }
}
}

```

```

        });

    }
    System.out.println("*****");
    System.out.println(e.getMessage());
    System.out.println("*****");
}

return null;
}

```

#### 4- Bank login

```

protected void doInBackground(Void... voids) {

    try {

        Class.forName("com.mysql.jdbc.Driver");
        Connection con =connectionstirng.getConnection();
        PreparedStatement st=con.prepareStatement("SELECT Username,
Name, Password from Bankemployee where Username=? and Password=?");
        st.setString(1,a);
        st.setString(2, b);
        ResultSet rs = st.executeQuery();
        if (rs.next()) {

            SharedPreferences.Editor editor = spre.edit();
            editor.putString("Username", rs.getString(1));
            editor.putString("Password", rs.getString(3));
            editor.putString("Name", rs.getString(2));
            editor.putString("Type", "Bank");
            editor.apply();
            Intent i = new Intent(BankLoginActivity.this,
MainBankerActivity.class);
            i.addFlags(Intent.FLAG_ACTIVITY_CLEAR_TOP |
Intent.FLAG_ACTIVITY_NEW_TASK);
            startActivity(i);

        } else
        {
            runOnUiThread(new Runnable() {
                @Override
                public void run() {

                    Password.setError(getString(R.string.error_incorrect_password));
                    Password.requestFocus();

                }
            });
        }
    } catch (Exception e) {
        System.out.println("*****");
        System.out.println(e.getMessage().toString());
    }
}

```

```

        System.out.println("*****");
        e.printStackTrace();
    }

    return null;
}

```

## 5- Booking

```

But_Select.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {

        Fragment fragment = new NewAppFragment();
        Bundle bundle = new Bundle();
        bundle.putString("AppDate", map.get(FIRST_COLUMN));
        bundle.putInt("AppService", ServicesSelec);
        bundle.putInt("AppMainServices", MainServicesSelc);
        bundle.putInt("PerHoureServices", PerServicesHours);

        fragment.setArguments(bundle);
        FragmentTransaction ft =
        getActivity().getSupportFragmentManager().beginTransaction();
        ft.replace(R.id.content_frame, fragment);
        ft.commit();

    }
});

```

## 6- Cancel booking

```

public void cancel_app(int app_no)
{
    final int v_appno=app_no;
    Thread t1;
    t1=new Thread(new Runnable() {

        @Override
        public void run() {

            try {

                Class.forName("com.mysql.jdbc.Driver");
                Connection con =connectionstirng.getConnection();
                PreparedStatement st;

                st=con.prepareStatement(" update appointment
set Status='Cancelled' where AppointmentNo=?");

                st.setInt(1,v_appno);
                int rs = st.executeUpdate();
            }
        }
    });
}

```

```

        if (rs>0)
        {
            new Handler(Looper.getMainLooper()).post(new
Runnable() {
                @Override
                public void run() {
                    Toast.makeText(getActivity().getApplicationContext(),
getString(R.string.succ_change_data), Toast.LENGTH_LONG).show();
                    Myappointments_Data();
                    adapter = new
AppoinmentsAdapter(getActivity(), R.layout.cancelapplayout,
AppList);
                    listView.setAdapter(adapter);
                }
            });
        }
    } catch (Exception s) {
        s.printStackTrace();
    }
}
});
try {
    t1.start();
    t1.join();
} catch (InterruptedException e) {
    e.printStackTrace();
}
}
}

```

### 4.3 Reports

We have many tables in database, our DB called "bankcusappdb" and has this tables:

#### **Appointment**

This table for saving appointment data.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	AppointmentNo	int(11)			No	None		AUTO_INCREMENT	Change Drop Primary Unique Index Spatial Fulltext Distinct values More
2	ServiceId	int(11)			No	None			Change Drop Primary Unique Index Spatial Fulltext Distinct values More
3	AppointmentTime	time			No	None			Change Drop Primary Unique Index Spatial Fulltext Distinct values More
4	AppointmentDate	date			No	None			Change Drop Primary Unique Index Spatial Fulltext Distinct values More
5	Approval	varchar(20)	latin1_swedish_ci		No	None			Change Drop Primary Unique Index Spatial Fulltext Distinct values More
6	Status	varchar(50)	latin1_swedish_ci		No	None			Change Drop Primary Unique Index Spatial Fulltext Distinct values More
7	CustomerID	varchar(10)	latin1_swedish_ci		No	None			Change Drop Primary Unique Index Spatial Fulltext Distinct values More
8	Username	varchar(50)	latin1_swedish_ci		Yes	NULL			Change Drop Primary Unique Index Spatial Fulltext Distinct values More

Figure 4-1 appointment Table

## Bank employee

This table for saving bank employee data.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	Username	varchar(50)	latin1_swedish_ci		No	None			Change Drop Primary Unique Index Spatial Fulltext Distinct values Add to central columns
2	Name	varchar(200)	latin1_swedish_ci		No	None			Change Drop Primary Unique Index Spatial Fulltext Distinct values Add to central columns
3	Password	varchar(50)	latin1_swedish_ci		No	None			Change Drop Primary Unique Index Spatial Fulltext Distinct values Add to central columns

Check all    With selected:  Browse    Change    Drop    Primary    Unique    Index    Add to central columns    Remove from central columns

Figure 4-2 bank employee Table

## Customers

This table for saving customer data.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	IDNumber	varchar(10)	latin1_swedish_ci		No	None			Change Drop Primary Unique Index Spatial Fulltext Distinct values Add to central columns
2	Name	varchar(200)	latin1_swedish_ci		No	None			Change Drop Primary Unique Index Spatial Fulltext Distinct values Add to central columns
3	Gender	varchar(1)	latin1_swedish_ci		No	None			Change Drop Primary Unique Index Spatial Fulltext Distinct values Add to central columns
4	Mobile	varchar(20)	latin1_swedish_ci		No	None			Change Drop Primary Unique Index Spatial Fulltext Distinct values Add to central columns
5	Password	varchar(50)	latin1_swedish_ci		No	None			Change Drop Primary Unique Index Spatial Fulltext Distinct values Add to central columns

Figure 4-3 Customer Table

## Main services

This table for saving main services data.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	ServiceID	int(11)			No	None		AUTO_INCREMENT	Change Drop Primary Unique Index Spatial Fulltext Distinct values More
2	ServiceDescription	varchar(200)	utf8_general_ci		No	None			Change Drop Primary Unique Index Spatial Fulltext Distinct values More
3	PerHour	int(11)			No	None			Change Drop Primary Unique Index Spatial Fulltext Distinct values More

Check all    With selected:  Browse    Change    Drop    Primary    Unique    Index    Add to central columns    Remove from central columns

Figure 4-4 main services Table

## Sub services

This table for saving sub services data.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	SubServiceID	int(11)			No	None		AUTO_INCREMENT	Change Drop Primary Unique Index Spatial Fulltext Distinct values More
2	SubDescription	varchar(200)	utf8_general_ci		No	None			Change Drop Primary Unique Index Spatial Fulltext Distinct values More
3	MainService	int(11)			No	None			Change Drop Primary Unique Index Spatial Fulltext Distinct values More

Check all    With selected:  Browse     Change     Drop     Primary     Unique     Index     Add to central columns     Remove from central columns

Figure 4-5 sub services Table

## 4.4 Layouts

We have interfaces for mobile application as like:

### ❖ Welcome Screen

This screen, we call it welcome screen for the user who visit our system in addition to get information from the visitor if he/she normal customer or banker.

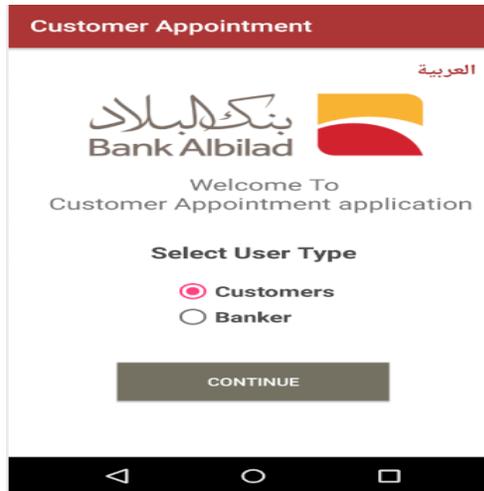


Figure 4-6 Welcome Screen

### ❖ Customer login Screen

This screen is for logging the user into the system by using username and password and it has link for making registration if the user has not an account yet.

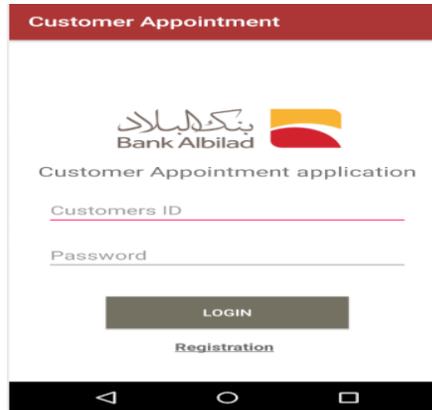


Figure 4-7 Customer Screen

❖ Register Screen

This is the registration screen, which enable the new users to have new account in the system of the application.

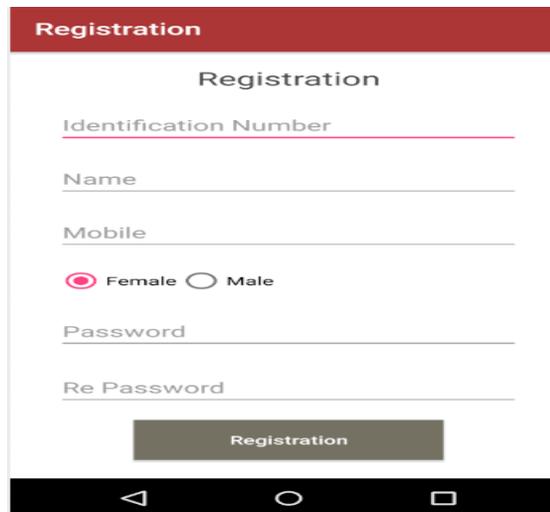


Figure 4-8 Register Screen

❖ Employee login Screen

This screen is specified for logging the employees users in the bank application.

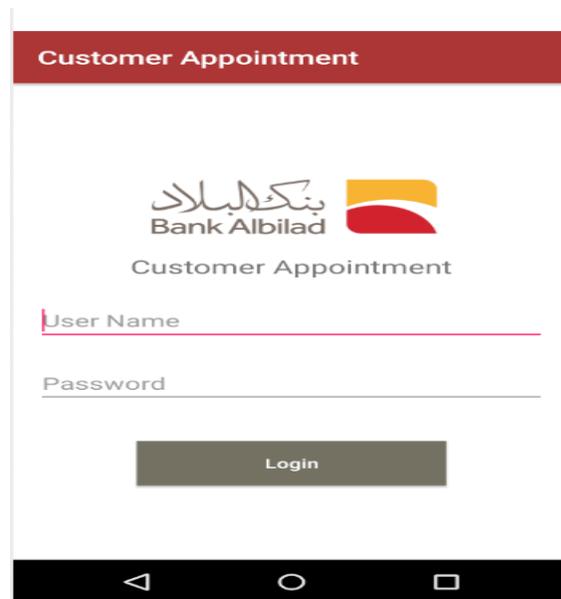


Figure 4-9 Employee login Screen

❖ Appointment for customer Screen

This screen will explore all appointments for a special user in the system with all details of the service name, date and time, and the status.

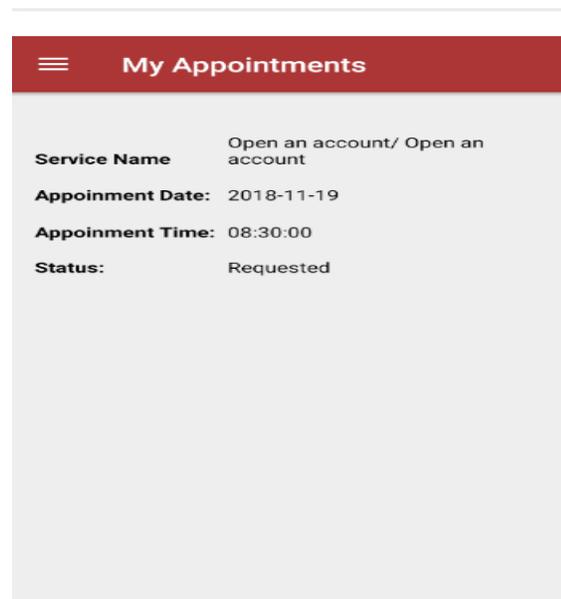


Figure 4-10 Appointment for customer Screen

❖ New appointment Screen

This screen enable the user to select the customer type, and press continue

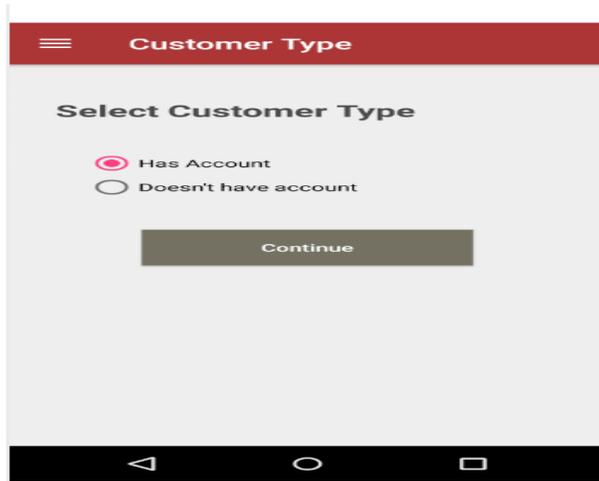


Figure 4-11 New appointment Screen

❖ Select service:

This screen enable the user to select the main services and service name then press show appointment.

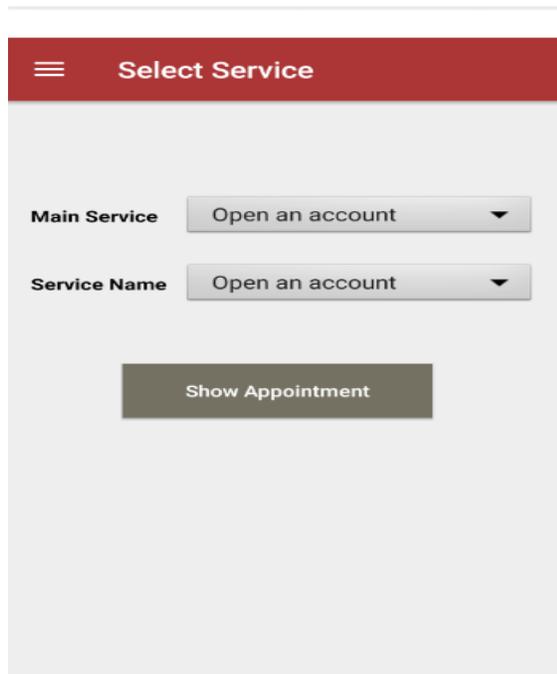


Figure 4-12 Select service Screen

❖ Appointment screen

This screen show us the time and date available of all appointments.

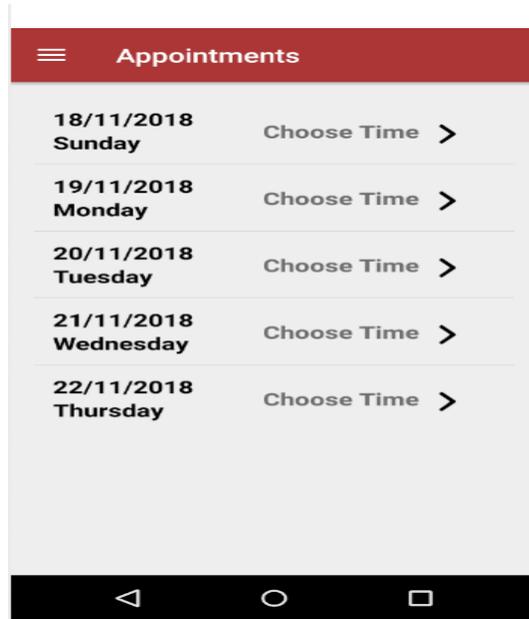


Figure 4-13 Appointment screen

❖ Time of the appointment:

This screen enable the user from choosing the time of appointment.

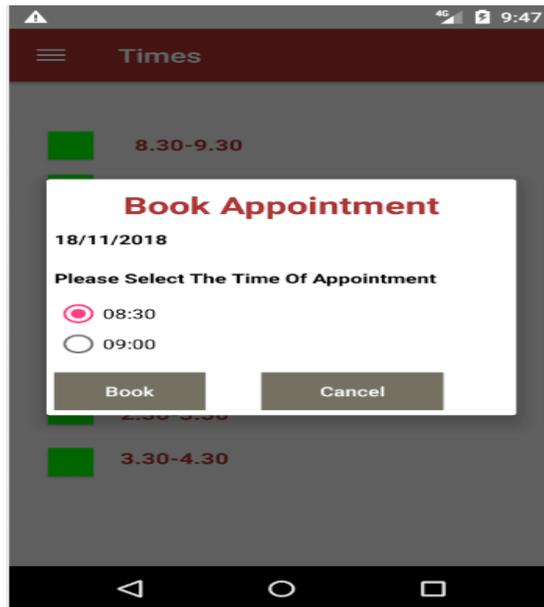


Figure 4-14 Time of the appointment Screen

❖ Cancel appointment Screen

This screen inform the user that he/she can cancel the appointment.

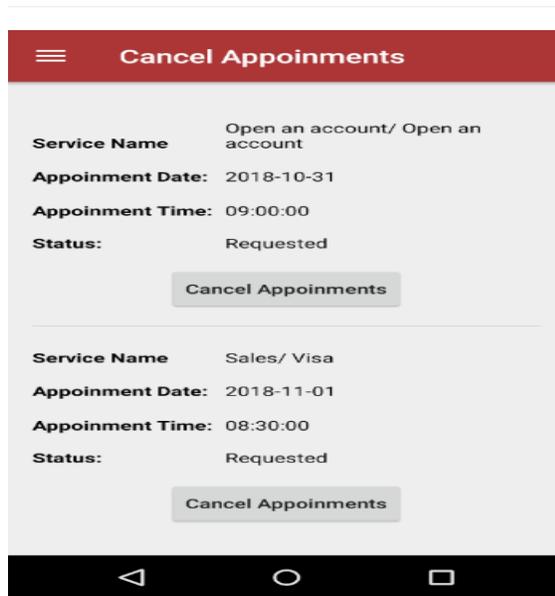


Figure 4-15 Cancel appointment Screen

❖ Booking procedures

This screen offer all available services in the application to support the user.

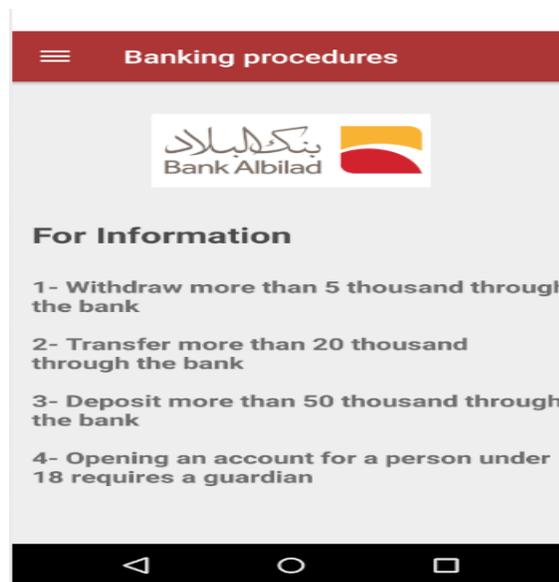


Figure 4-16 Booking procedures Screen

❖ change password

This screen enable the user from changing the password of his account.

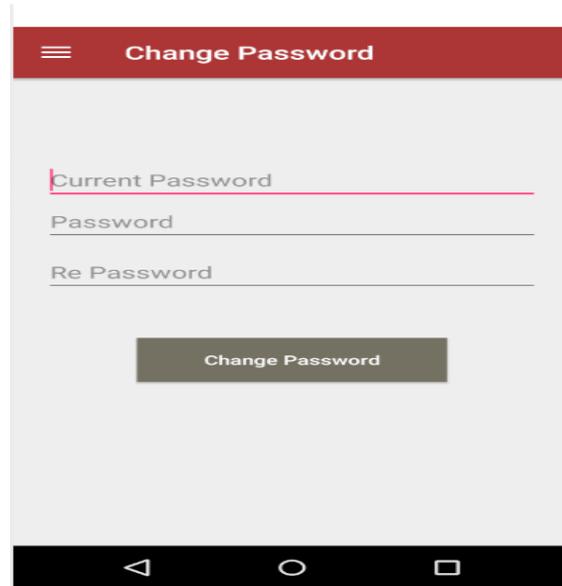


Figure 4-17 change password Screen

❖ User management

This screen enable the administrator to make managements on the accounts of users in the system.

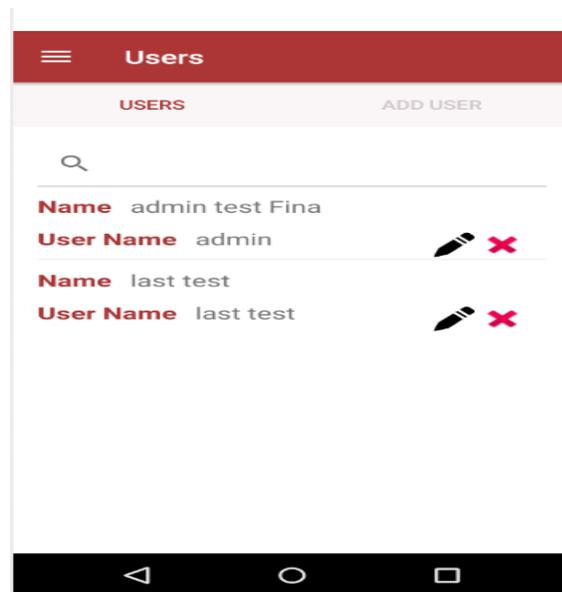


Figure 4-18 User management Screen

❖ service management

This screen enable the user from choosing the needed service to make management on it.

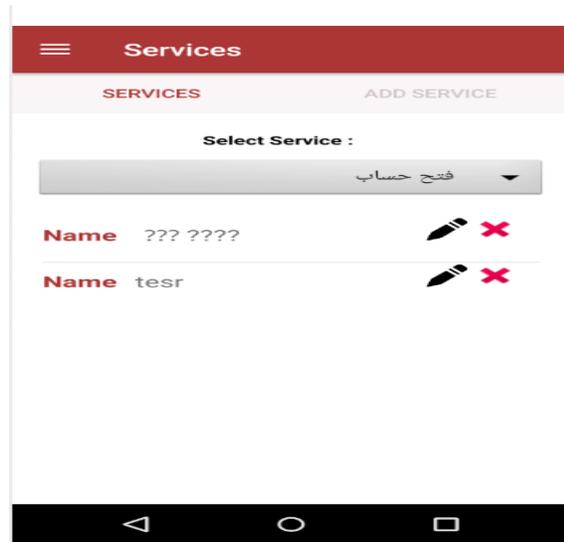


Figure 4-19 service management Screen

❖ appointment management

This screen for making management on the appointments that he/she can approve the appointment or reject it.

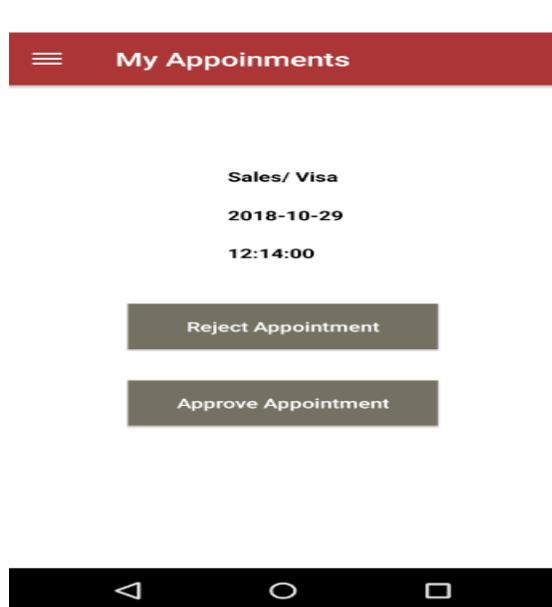
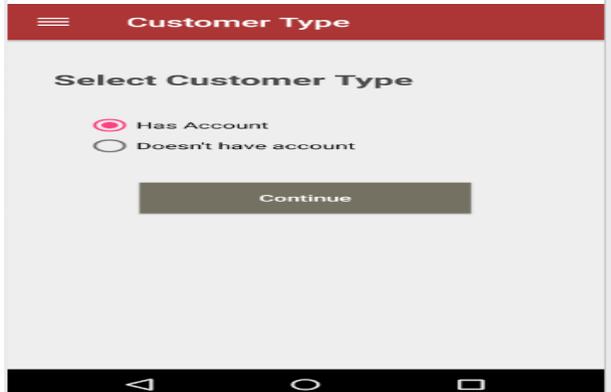
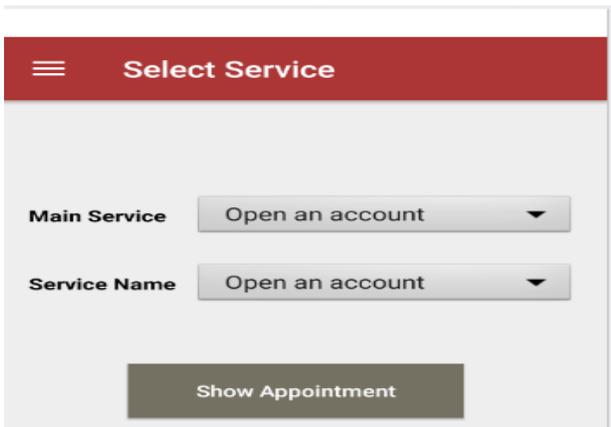
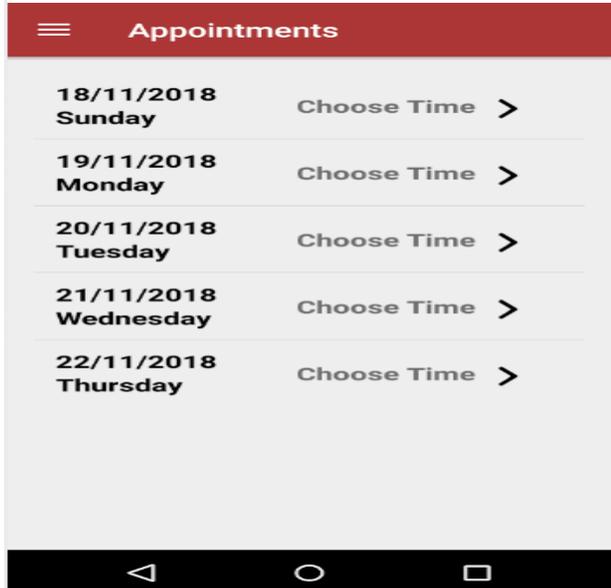


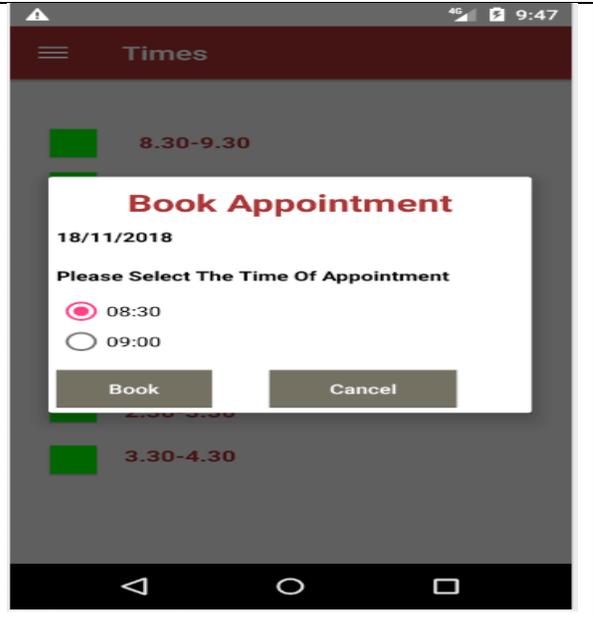
Figure 4-20 appointment management Screen

## 4.5 Reports layouts

### ❖ Steps of Booking

<p>New appointment Screen</p> <p>This screen enable the user to select the customer type and press continue</p>	
<p>Select service:</p> <p>This screen enable the user to select the main services he want to use it.</p>	
<p>Appointment screen</p> <p>This screen show us the time and date available of all appointments.</p>	

Time of the appointment:  
This screen enable the user from  
choosing the time of appointment.



## **5. Conclusion and future work**

Bank Customers Appointments app will specially design for banks to enable both banks employs and customers to do the work in optimal and easy way and accomplish their daily work efficiently, it much more efficient it makes the operations of to run more smoothly with is time consuming it consider a platform for improving the quality of booking appointment.

The analysis and design phase of the application has been completed. And the application implementation .

## **6. References**

[1] <https://play.google.com/store/apps/details?id=sa.gov.moi>

[2] <https://www.lucidchart.com/pages/ER-diagram-symbols-and-meaning>

[3] <http://www.businessdictionary.com/definition/data-flow-diagram.html>

[3] <https://play.google.com/store/apps/details?id=moh.gov.sa.mawid>

[4] [https://www.ccri.edu/faculty\\_staff/comp/kelly/.../Exam1\\_Terms.pdf](https://www.ccri.edu/faculty_staff/comp/kelly/.../Exam1_Terms.pdf)

[5] <https://eternalsunshineofthemind.wordpress.com/2013/03/05/context-level-dfds-level-1-dfds/comment-page-1/>

[6] [https://en.wikipedia.org/wiki/Class\\_diagram](https://en.wikipedia.org/wiki/Class_diagram)

[7] <https://www.lucidchart.com/pages/uml-use-case-diagram>

[8] [https://www.tutorialspoint.com/uml/uml\\_activity\\_diagram.htm](https://www.tutorialspoint.com/uml/uml_activity_diagram.htm)

[9] <https://www.uml-diagrams.org/state-machine-diagrams.html>

[10] <https://www.smartdraw.com/sequence-diagram/>

[11] <https://economictimes.indiatimes.com/definition/systems-design>

<https://www.draw.io/>

## 7. Appendixes

هل ترغب ان يكون هنالك حجز مسبق؟؟

نعم

لا

هل تعتقد ان وجود تطبيق تسجيل مواعيد سوف يكون ذا عون لك؟؟

نعم

لا

هل يساعد التطبيق على تنظيم المواعيد؟؟

نعم

لا

هل يساعد التطبيق على انجاز العمليات البنكية بوقت قياسي؟؟

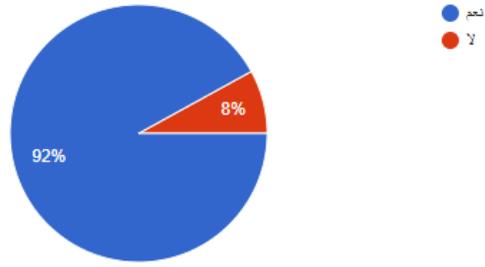
نعم

لا

تذا  
اتة

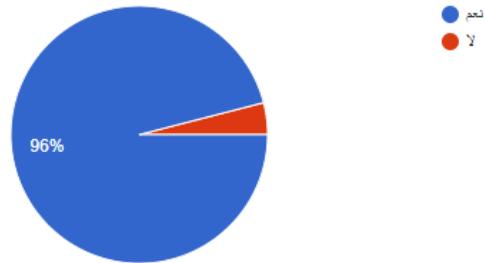
هل تواجه مشكلة بمواعيد البنك؟؟

٢٥ رداً



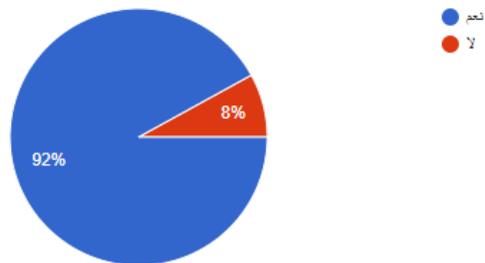
هل تقضي وقت طويل بانتظار موعدك؟؟

٢٥ رداً



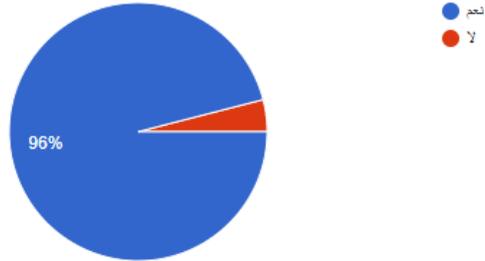
هل تود اختيار موعد مناسب لك لإتمام عملياتك البنكية؟؟

٢٥ رداً



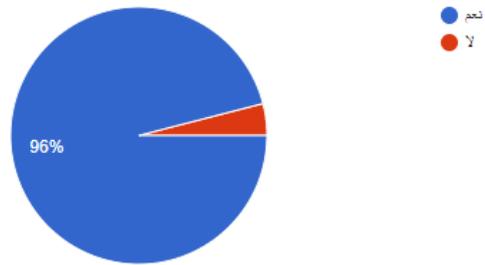
هل ترغب ان يكون هنالك حجز مسبق؟؟

ردًا ٢٥



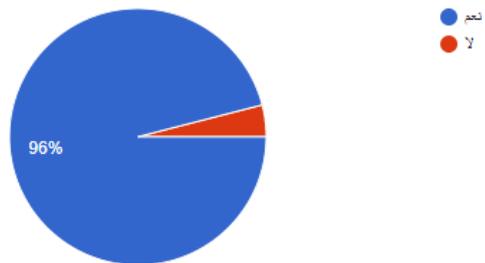
هل تعتقد ان وجود تطبيق تسجيل مواعيد سوف يكون ذا عون لك؟؟

ردًا ٢٥



هل يساعد التطبيق على تنظيم المواعيد؟؟

ردًا ٢٥



هل يساعد التطبيق على انجاز العمليات البنكية بوقت فياسي؟؟

٢٥ ردًا

