Kingdom of Saudi Arabia
Majmaah University
Ministry of Higher Education
College of Science Al Zulfi

المملكة العربية السعودية
جامعة المجمعة
وزارة التعليم العالي
كلية العلوم بالزلفي

Secret Pass Application

Student Affairs System
For College of science Al Zulfi
Department of Computer Science and Information

Graduation Project
Submitted in partial fulfillment of the requirements for the award of
Bachelor's degree of the Majmaah University
(Semester 1, 2018-19)

Submitted by:

Hana Sultan Alturki	351205189

Under the supervision of:
Dr. Sarah M. Eljack

Dedication

I would like to present this research to those who have supported me in all stages of my life from childhood until this moment I waited for, those who were a source of encouragement during the years of my study who love me more than themselves and those who see me as a source of their happiness, To my great parents, to all my family and to myself after years to see the beginning.

Acknowledgments

I give a lot of thanks and appreciation to everyone who helped me during this research and to all those who supported me, and to the teachers who have done their job in the best way that helped me to implement this project.

Abstract

This research develops an application employing a modern technology in the field of data protection. It aims to protect the personal data from unauthorized access by categorizing the data and save it by using password and email.

in this project, a questionnaire was conducted to test the importance of this idea and then compared the idea with some similar ideas that have been implemented by other people.

The project designed by using the android studio and it runs on Android devices.

After implementing this Application successfully, we achieve the objectives that save personal data.

Key words

Android Application, Data protection.

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

(CERTIFICATE BY STUDENT)

This is to certify that the project titled "**Secret pass application**" submitted by me (**Hana Sultan Alturki ,351205189**) under the supervision of **Dr. Sarah M. Eljack** for award of bachelor's degree of the

Majmaah University carried out during the Semester 1, 2018-19 embodies my original work.

Signature in full: -------------------------------

Name in block letters: **Hana Sultan Alturki**

Student ID: **351205189**

Date: 21 Nov 2018

Contents

Contents **Page**

List of tables

Table **Page**

List of figures

Table **Page**

Chapter One: Introduction.

1-1 Overview:

Applications are the new wave in this modern time, which has developed to become available in all different fields. Recently, users prefer to provide applications to their works or to communicate with them instead of using traditional methods.

One of the most important technologies is smartphone applications that have the advantages of saving time and effort and facilitating access. And the most important requirements for users is the proportion of security in those applications there is relationship between the number of users using an application with the proportion of security in applications, if the application is developed by a reliable and secure resources, the proportion of users increases compared with applications or sites that pose a threat to users such as anonymous applications.

Applications help users save their effort by saving their data in various applications that have been converted and developed from a traditional way to the technical.

Applications are evolving day by day. The user has one device that stores all his data and needs instead of carrying many personal papers and cards. However, despite all the advantages provided by data-saving applications, they also carry significant security risks. User data can be stolen and exploited by others. One of the main purpose's users can use any application is the confidentiality of the application and security to increase reliability in use.

From this main objective, the idea of this application came, which provides confidentiality and security in the field of private and confidential data stored in one application.

1-1-1 Abstract system description:

Create an application that prevents user's data usage by others. As the application contributes to the preservation of private data and protected by a password to enter and use the app. The application increases the confidentiality and data privacy, so it allows the access only by using the password and email. The Application saves the data entered by the user directly or access the user device to bring the data from the mobile. As an example, it can save photos, Notes, Identity, Bank accounts with the password, websites (username with password), social media accounts …etc. user can access his data any time because this app using an online database.

1-2 Problem definition:

from the research review of the writer, people face many challenges regarding how to protect their private data especially their personal data, from loss, modification, and unauthorized access. Such as use of password of bank cards to withdraw the bank balance and Exploitation of private photos … etc. This project aims to solve this problem by developing an application that can protect the private data.

1-2-1 Goals:

The user's data must be protected to prevent use or exploitation of the user's data by others because such data usually contain private and confidential user information. The main goal of this project is restrictions limitations on this information to prevent unauthorized users from using the data and allow the user to access the data any time from any device.

1-2-2 Objectives:

The application aims to employ technology in the protection of user data in several aspects:
o to save data rather than using manual objects to save them.
o to protect data in a protected application can be accessed by an authorized user.
o an easy way to access the data.
o to save time and effort.
o to obtain Privacy.
o to keep personal data safe and secure.
o to schedule the important and personal events.
o to access and recover saved data any time.

1-2-3 Critical success factor:

To use the application, you need to:
o Download it on Android system.
o Needs device with 20 API and later.

1-2-4 Organization chart and responsibilities:

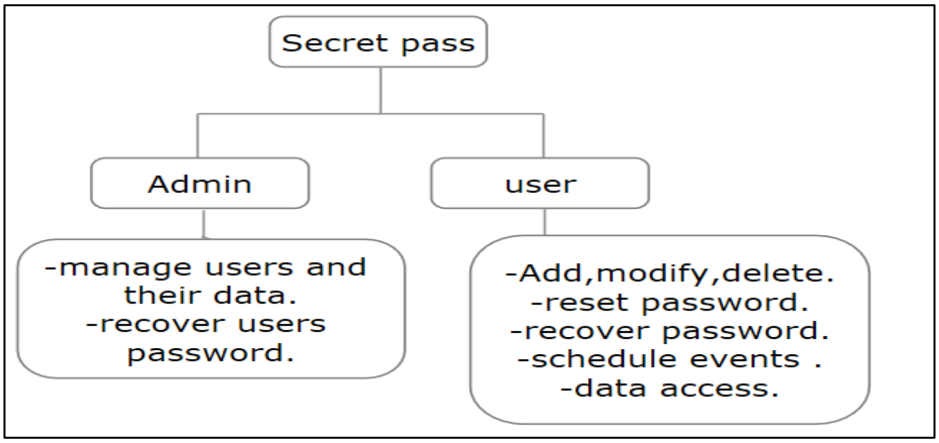


Figure (1-1) Organization chart.

1-2-4-1 Functional requirements:

Functional requirements considered as a set of inputs, outputs, processes, services that the system must provide and how it interacts with certain inputs.

- **Input Methods:**

  Access to data stored on the phone.

  Touchpad.

- **Cancellation:**

  A selected date can be deleted.

- **Display:**

  The simplicity of graphical interfaces.

- **Limitations:**

  Select a strong password with email to access the app.

  Internet connection.

- **Target group:**

  Suitable for all that have private data.

1-2-4-2 Non-Functional requirements:

- **Availability:**

Available for use on the Android system.

Free download.

- **Protection:**

As an application to protect data from others and it from any alteration, the most important criteria must be being the high level of security to prevent unauthorized users from accessing the application and choosing a strong password.

- **Performance:**

It should be simple and Flexible.

Easy to use and responsive.

- **Maintenance ability:**

Must be maintainable to improve performance.

1-3 General rules:

- Password and email to access the app.
- Android operating system.
- The existence of private and important data to use the application.
- Internet connection.

1-4 Feasibility study:

The feasibility study is one of the basics of building any project to know the importance of this project and the percentage of need and know all the details related to this project. I applied a questionnaire to measure the importance of the project and the percentage of people who need this application through Google drive, I got 24 responses based on these responses concluded that protecting the personal data of users is one of the most important goals needs by users with different type of data (for more details see appendix A). aspects of the Feasibility study are mentioned as following:

1-4-1 The technical study:

to use the application of secret pass on the Android operating system we need an Android mobile phone.

Android mobile phone specifications (minimum):

Operating system: Android with 20 API and later.

internet: available.

1-4-1 Operational Study:

The presence of devices capable of accommodating the size of the application does not require external competencies because the design of the application enables the user to identify the application tools and how to use.

1-4-2 Economic study:

Benefits should be greater than costs, so the secret pass application is free for users and Android devices are available with most people, so I expect the cost to be low for these reasons.

Chapter two: literature survey.

2-1 Introduction:

Smartphone applications are one of the most important technologies that have made a difference in the world of technology the applications of mobile phones have invaded many fields, including the field of protection of data, which vary in size and types. Methods of protection have appeared on various operating systems in different ways to protect.
This chapter deals with the related works that were searched and compared with the secret pass project to fill the missing gaps of these works.

2-2 Related works:

2-2-1 Private Notes and Secret Diary:

An application that Provides functionality for creating simple notes and keeping these notes secure by requiring a password to be entered before any note can be viewed or edited. Allows you to share your note via email with others and keep notes secure. Great for storing all notes without the risk that someone might get access. (App store,2018)

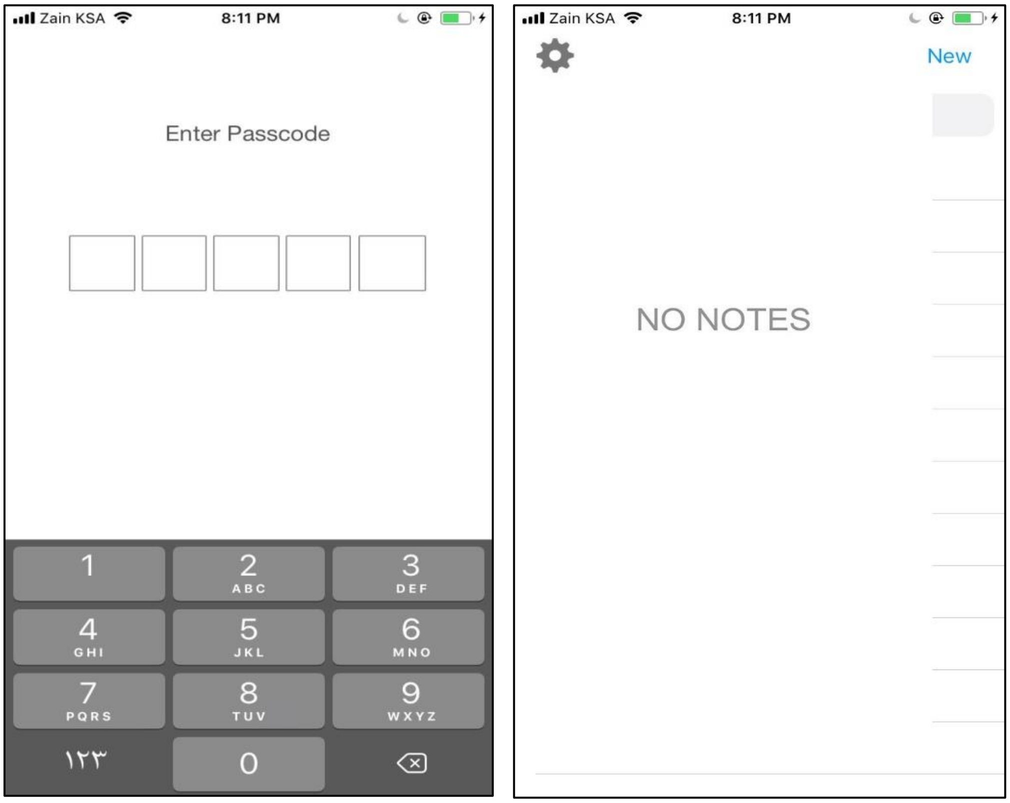
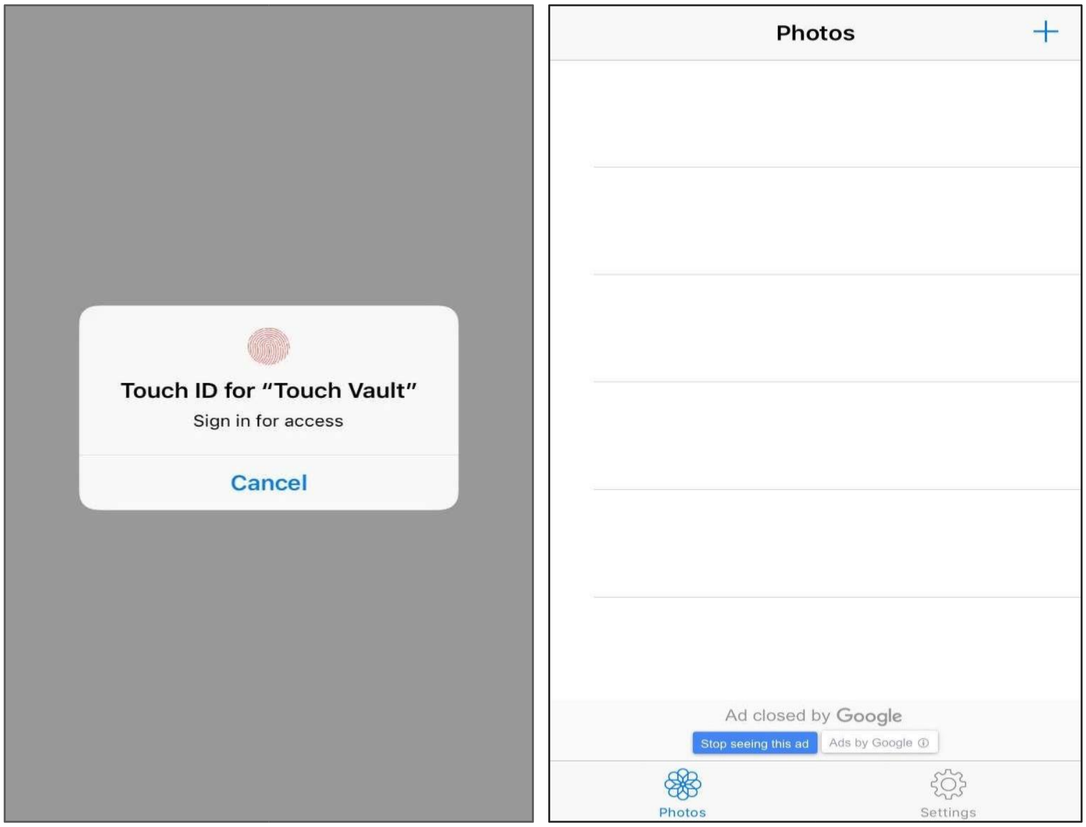


Figure (2-1) Private Notes and Secret Diary interface.

2-2-2 Touch Vault.

Application that Protect photos and videos with Touch ID and optional password to more secure. Photo management: Create Albums, Import / Export photos, Image Gallery and video management: Create Albums, Import / Export videos and Watch Videos.
photos and videos are 100% private and stored only on user device. (App Store, 2018)



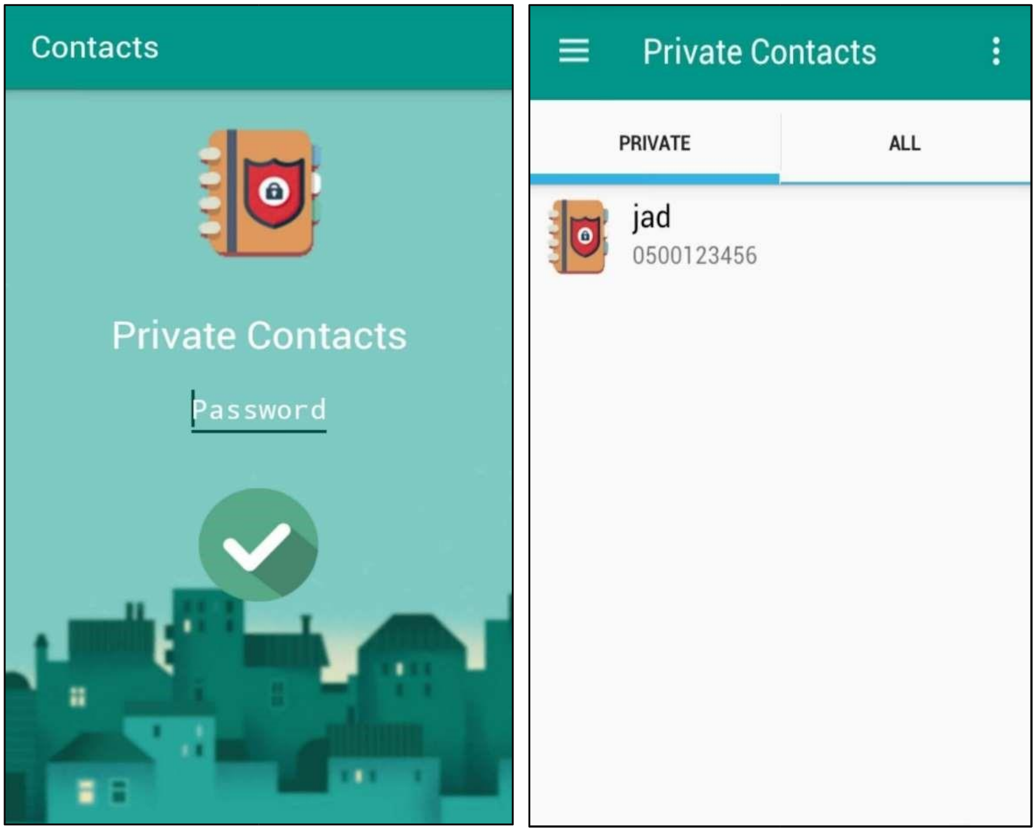
Touch ID for "Touch Vault"

Sign in for access

Figure (2-2) Touch Vault Interface.

2-2-3 Private Contacts and Dairy:

is a simple Application To hide your contacts and chat with those contacts easily, it’s not just Hide Contacts You can write Dairy and Save secretly, it is enabled with a secured offline login system. (Play.google.com, 2018)



jad
0500123456

Figure (2-3) Private Contacts and Dairy Interface.

2-3 Comparison:

Application name	programming language	operating system	Advantages	Disadvantages
Private Notes and Secret Diary.	Swift	iOS	- Allows to write several notes. - Saves the date of each note written in the app. - Allows All notes sent directly to e-mail.	- Do not save voice notes. - Available on one operating system. - There is no data recover when they lost.
Touch Vault	Swift	iOS	- Allows access to the application by fingerprint. - To increase protection, it is possible to put a passcode after the fingerprint. - Allows the user to access the camera. - A lot of photo albums can be created.	- When the password is forgotten, they cannot be restored. - Selecting photos does not work. - There is no data recover when they lost.

Application name	programming language	operating system	Advantages	Disadvantages
Private Contacts and Dairy.	Java	Android	- Save contacts by accessing user's contacts and sharing it or write new contact in the app. - Notes can be written	- Slow as well as technical problems. - available on one operating system. - There is no data recover when they lost.
Secret pass. (my app)	java	Android	- Save photo, notes, social media accounts (username and password) and bank cards (accounts and secret numbers), ID and others. - Protects data with a password to access and use it. - Recover password in case user forgot password. - Access saved data any time . - schedule important event.	- Available on one operating system.

Table (2-1) comparison.

Chapter Three: System Analysis.

3-1 Introduction:

The stage of analysing the system is one of the most important steps to build or develop a project, for reasons such as solving a problem or increasing improvements to the pre-prepared project. It separates the large projects into smaller parts to enable the development in a more accurate and better way. Then, the parts that represent an improved project are collected. (En.wikipedia.org, 2018)

to complete the implementation this project Selected the prototyping model that fits the size of the proposed project and acts as a sample to test the process. From this sample we learn and try to build a better final product also the system implements quickly, easy to test and solve problems in the system.

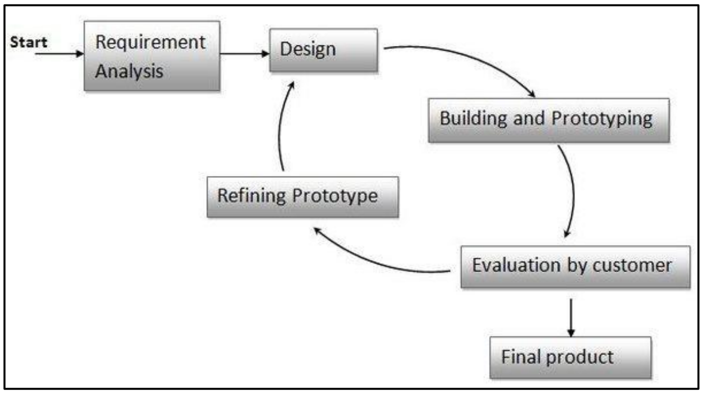


Figure (3-1) prototype model.

Retrieved from https://www.quora.com/What-is-the-prototype-model-When-is-prototype-modelling-used

3-2 Description of data flow diagram (DFD):

“A data flow diagram (DFD) is a graphical representation of the flow of data through an information system, modelling its process aspects. A DFD is often used as a preliminary step to create an overview of the system without going into detail, which can later be elaborated…A DFD shows what kind of information will be input to an output from the system, how the data will advance through the system, and where the data will be stored. It does not show information about process timing or whether processes will operate in sequence or in parallel.” (En.wikipedia.org, 2018)

3-2-1 Context diagram:

the context diagram is a general idea of the system which shows the limits of the system with its surroundings and its external environment. also, it contains only one process representing the whole system, data flows from and to the system and does not contain data stores.

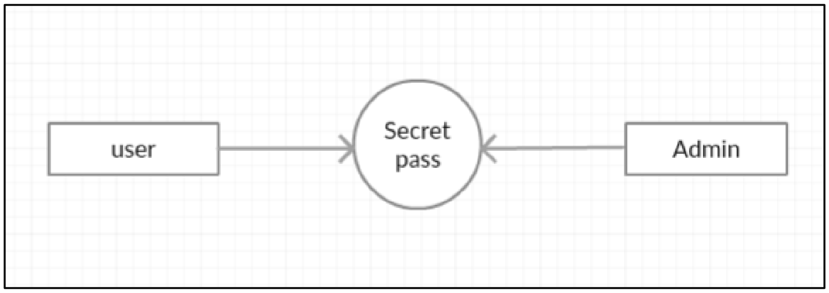


Figure (3-2) Context Diagram.

User, Admin are the entities of the Secret pass. (see figure 3-2)

3-2-2 Overview diagram (level 0):

It shows the main processes and represents the steps that the system performs from the first step to insert the data to the last step of the data output, including the data stores.

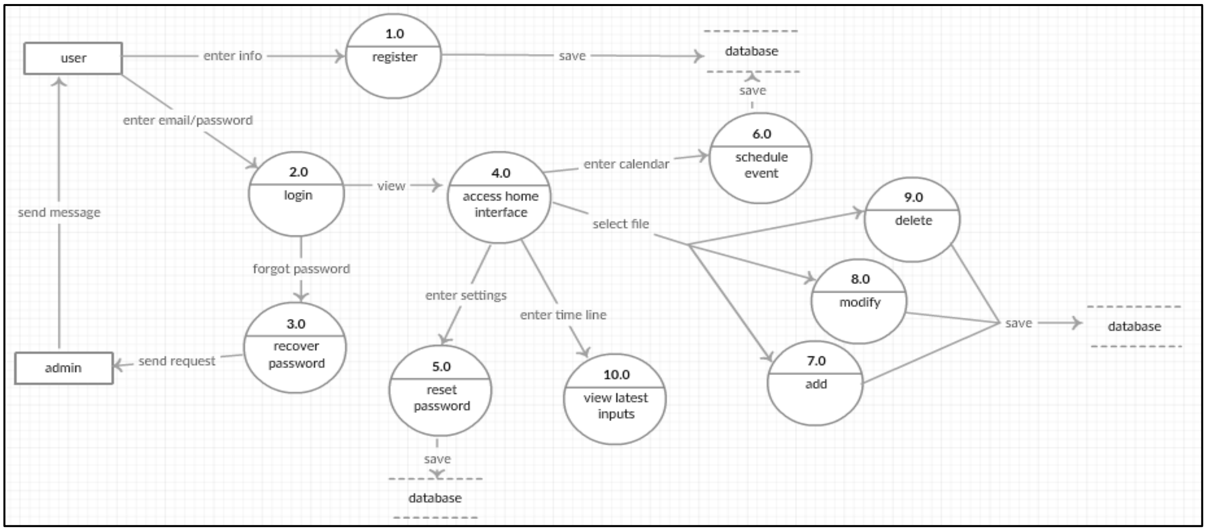


Figure (3-3) Level 0 diagram.

user's register in the app by entering required information then the user's info will be saved in the app database.to log in user enter the password and email to access the app and view all files then should select a file to add data and save it, the user can modify or delete the data. to reset the password the user enters to setting on the app and reset it. if the user forgot the password it allows to recover by request to the app admin, also user allowed to schedule his event after entering to the calendar, the timeline shows the user latest inputs. (see figure 3-3)

3-2-3 Detailed DFDs:

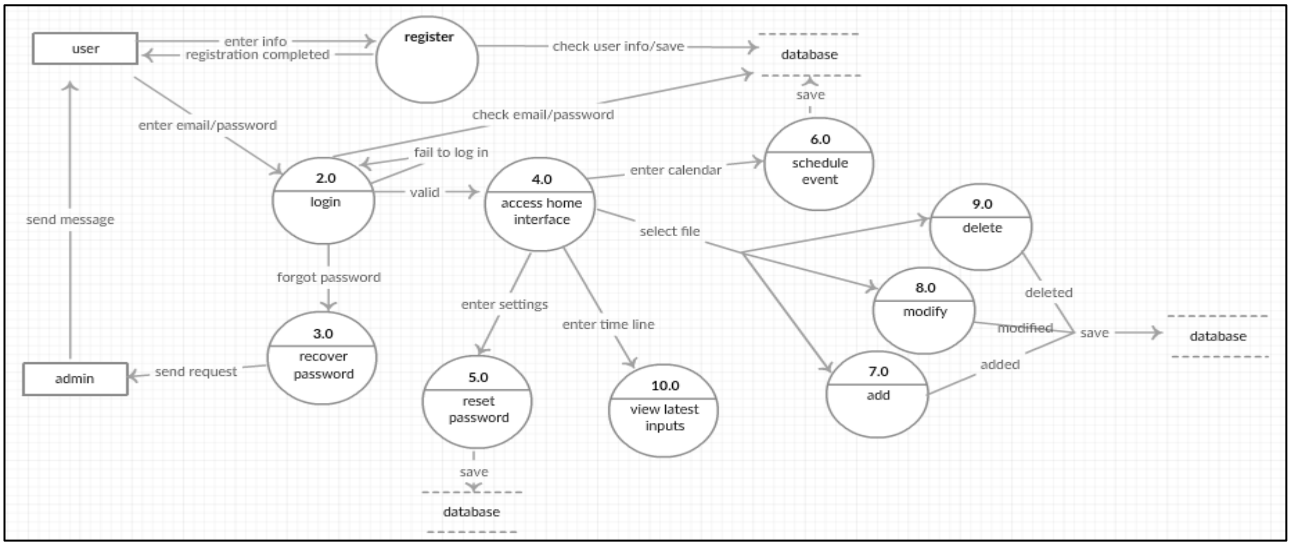


Figure (3-4) Detailed DFD diagram

3-2-4 Use case Diagram:

Use case Diagram is a description of the system which is consists of actors, system, use case and relationships such as extend and include.

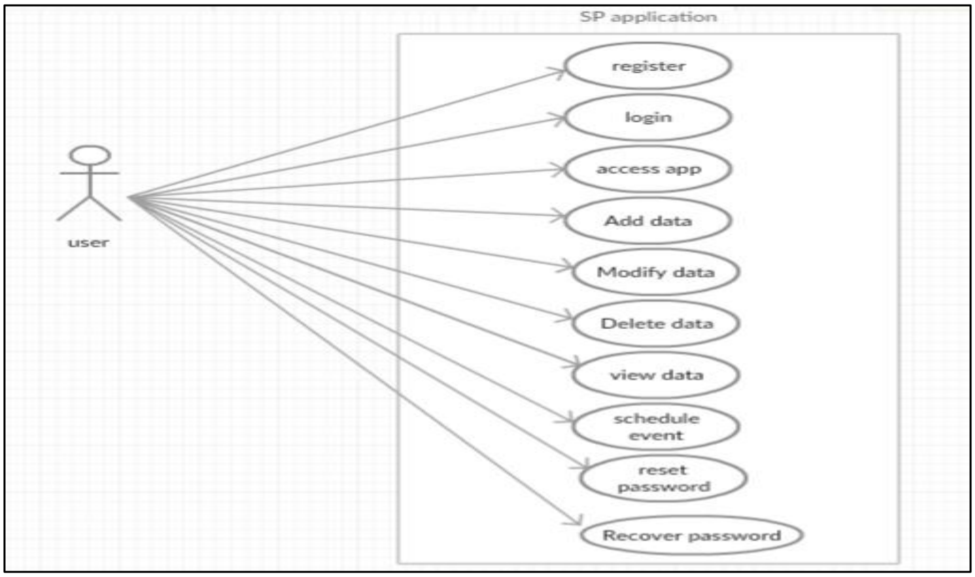


Figure (3-5) Use case Diagram for user.

Registration and log in: When the user start registering for the application, the new registration screen will appear, which contains several data, including e-mail, phone number, password, location, security questions. This data will be checked if it is entered correctly then registration is completed, after registering, the user can use the application by entering the password and email. **Access app:** After logging in, if the password and email correct user will access the app and home interface will appear.

Insert and delete: After selecting a specific file allows to the user now to add data into the file and save it or select saved data to delete it from this file.
Modify: the user can modify his/her data saved in the file.
View data: user see the latest inputs also he can access his data any time from another device.
Schedule event: user can Schedule his coming event through calendar.
Reset password: modify the password from setting.
Recover: In case user forgot the password, it allows to recover the password by send email from the admin to user's email and then user will enter to the link to reset password and log in again. (see figure 3-5)

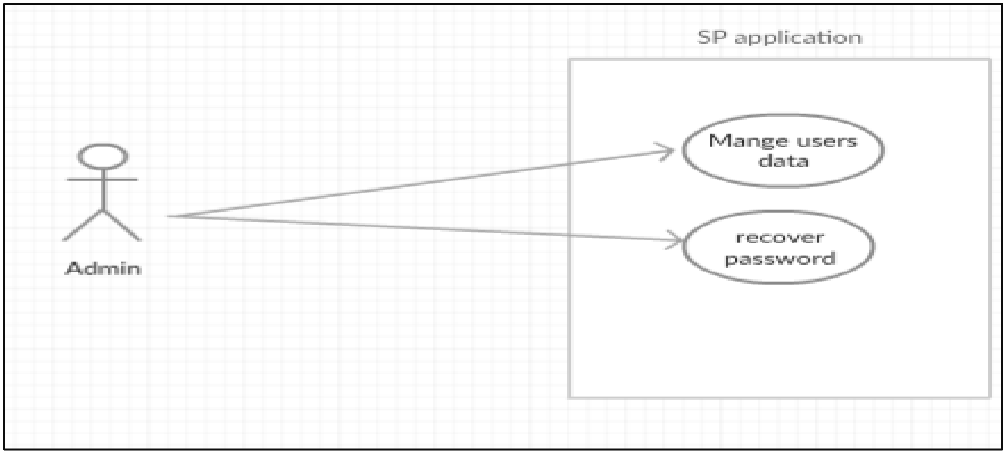


Figure (3-6) Use case Diagram for admin.

Mange user data: Admin can manage all user data (add, delete, modify).
Recover password: Admin help the user to recover his password through email.

3-2-5 Activity diagram:

Activity diagram shows the system operations and it contains a starting point, endpoint, decision node, and control flow, joint symbol, and fork symbol.

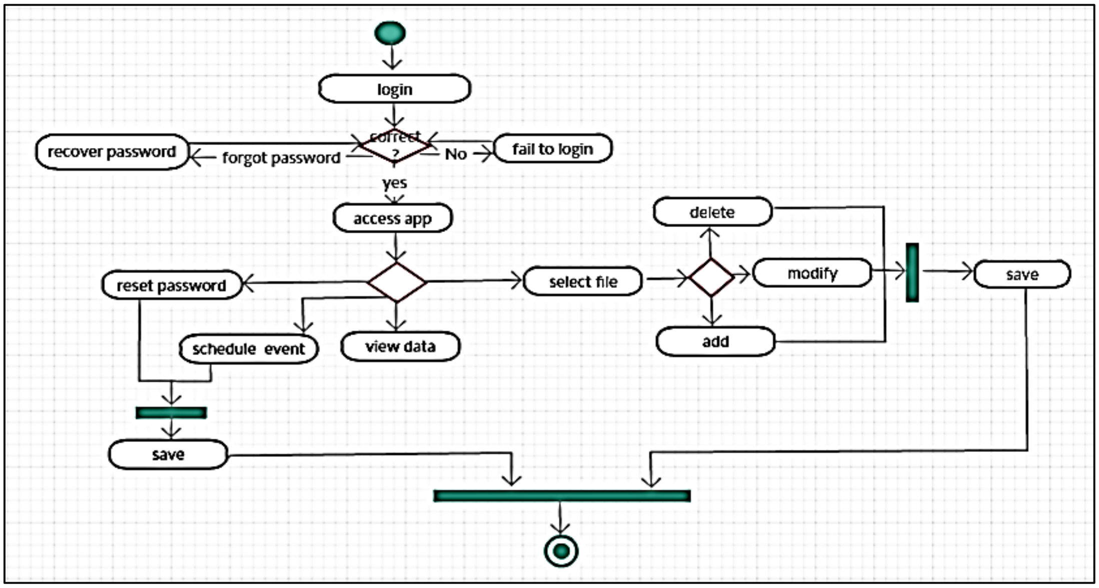


Figure (3-7) Activity Diagram.

After entering to the login interface there is a decision after entering the email and password if the login info invalid there is a message will appear indicating fail to log in if the login info valid user will access his data, user has a decision to enter to settings to reset password, calendar to schedule coming event, timeline to view latest inputs, and files to add/delete/modify data in case the user forgot his password he will recover the password and enter the app again. See figure (3-7)

3-2-6 Sequence Diagram:

Sequence diagram shows the time of the interaction between system interties.

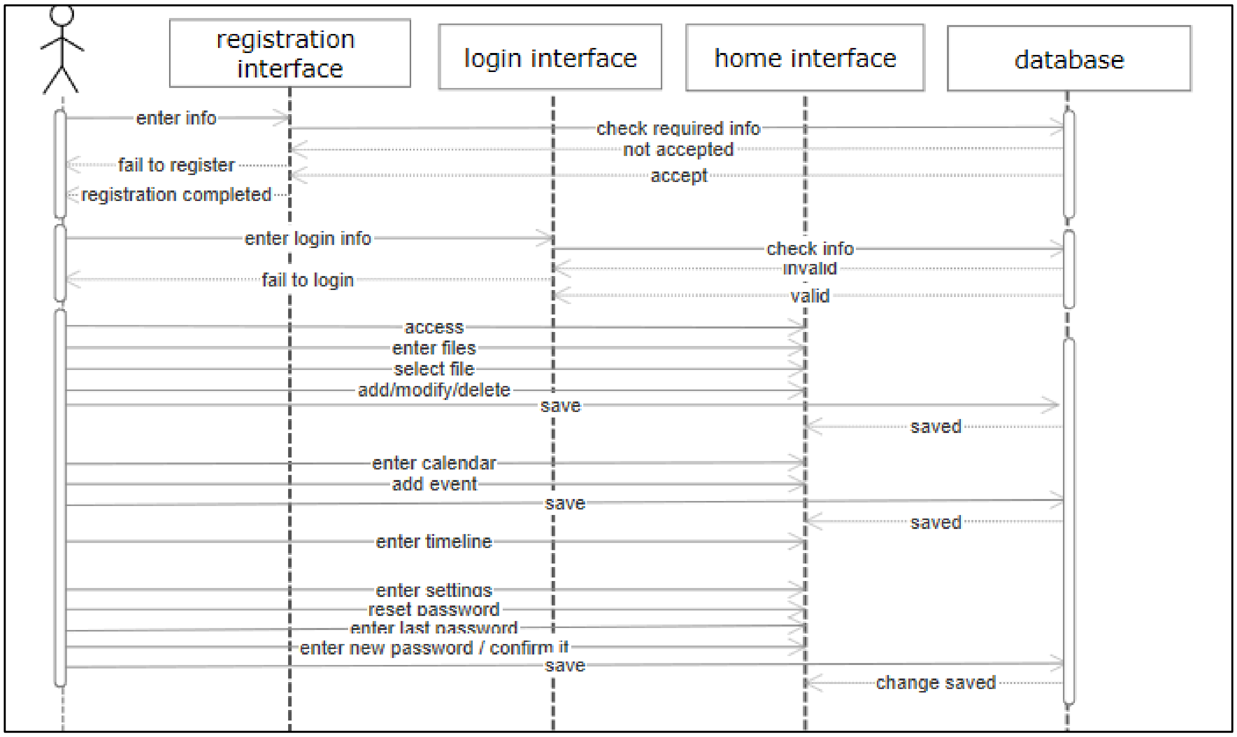


Figure (3-8) Sequence Diagram

First, user register in the app then the app checks user info if all entered as required, registration completed, and the user can use the app but after entering the login info if it valid allows to user to access otherwise not.

After accessing, the user should enter files a select specific file to add or modify entered data or delete it from the file.

the user can schedule an event through the calendar also, the user can view the latest inputs through the timeline.

The user can reset the password by entering the setting of the app and enter the previous password then entering new password then save the change. (see figure 3-8)

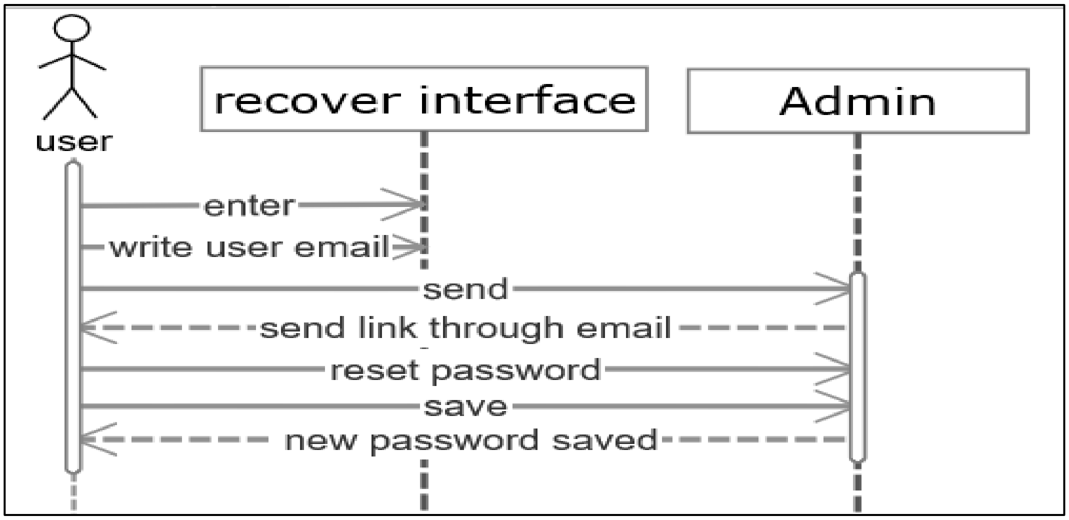


Figure (3-9) Sequence Diagram(recover)

If the user forgot the password the process to recover the password start when the user click (forgot password) the recover password interface will open then user write his email then click send, the admin will send email to user with link ,user will enter to link to reset the password then save change.(see figure 3-9)

3-2-7 Class diagram:

Class diagram shows the structure of a system with the entities and their attributes and operations. It includes many types of relationship between entities such as association, aggregation, generalization…etc.

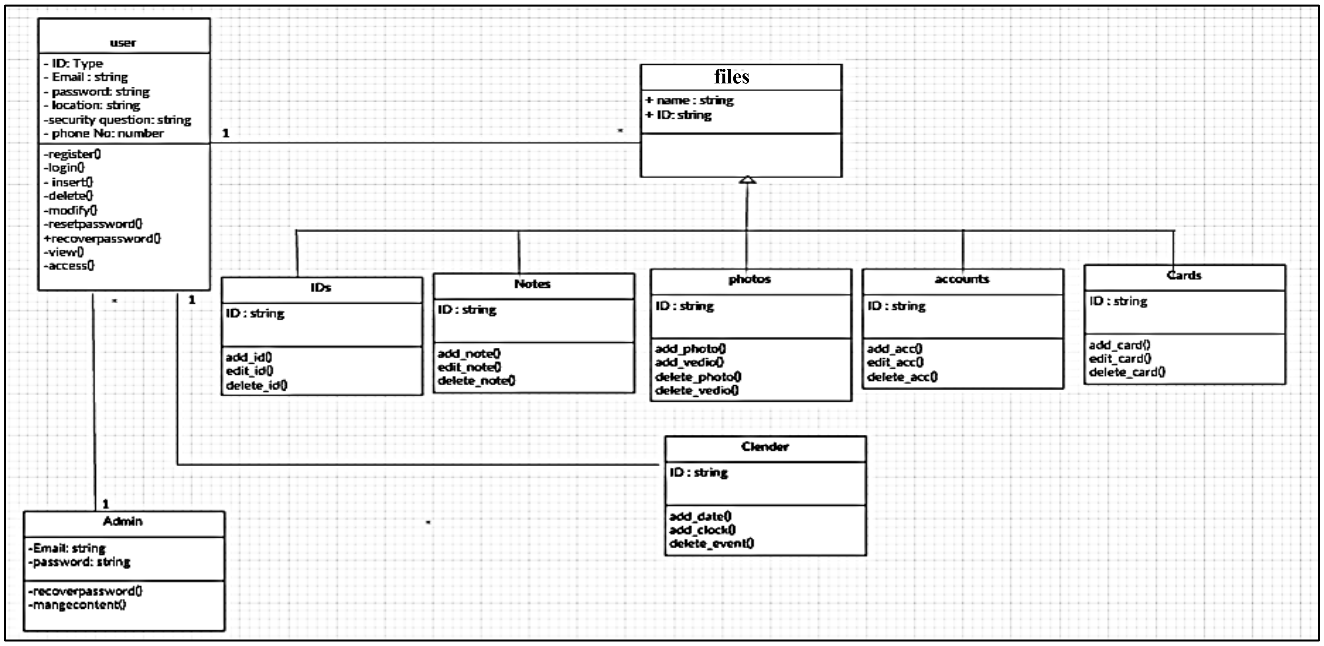


Figure (3-10) class Diagram.

3-2-8 Object diagram:

it is a describe a part of the class diagram with real entities, operation, and attribute.

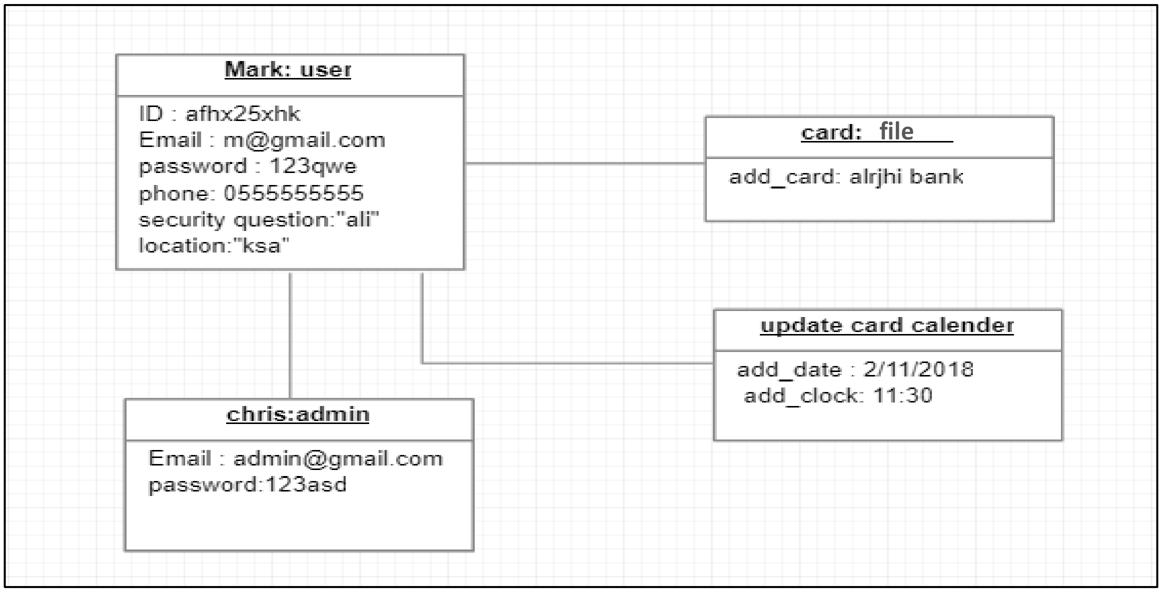


Figure (3-11) object Diagram.

3-2-9 State diagram:

It is a describe states of the system with the conditions of each state to be performed.

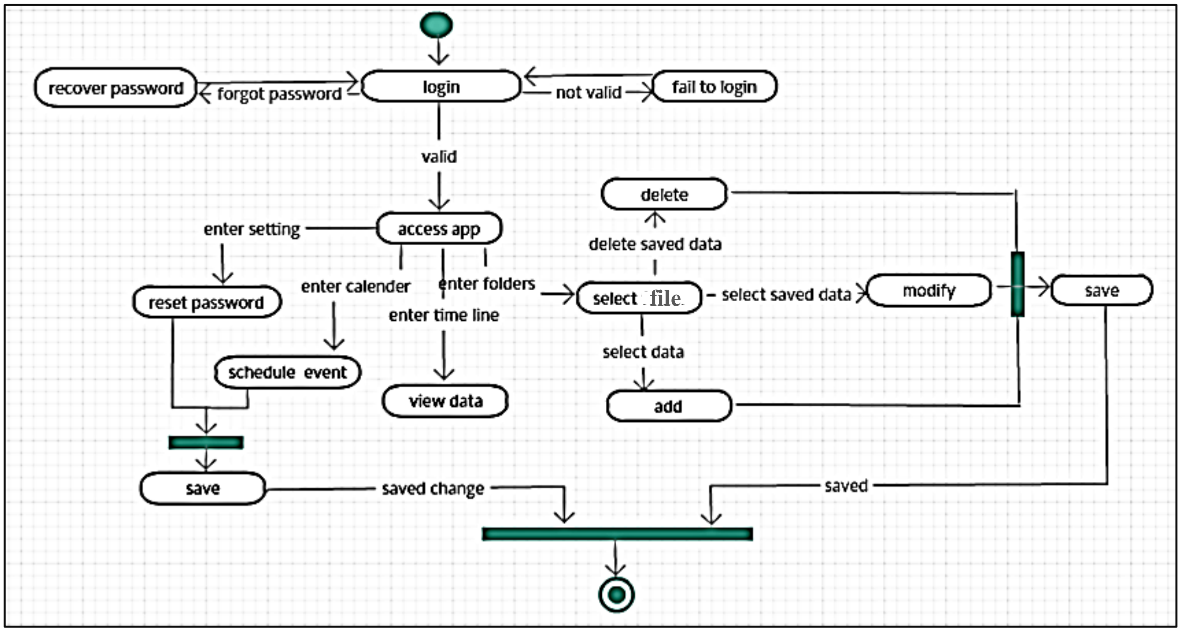


Figure (3-12) state diagram

The idea of state diagram is before performing any state is a condition to perform that state so as an example to access the app the user should do the condition enters the login info and if it is valid the state will be performed otherwise the user will stay in the same state and thus the process for all the states .(see figure 3-12)

3-3 Entity Relationship Diagram (ERD):

It describes the relationship in the system between entities of the system, it is drawn an entity in a rectangle box and relationship in polygon and connect between entities with a line hold the type of the relation between two entities.

There are different types of relation M:M, M:1 and 1:1.

3-3-1 Description of Entity:

1-Name: User.
Attributes: ID: user ID.
Password: user password to login to the application.
Email: user Email in case he forgot his password.
Location: location of the user.
Phone Number: number of user phone. Security Questions: user answer two questions.

2-Name: Admin.
Attributes: Email: email to access the database
Password: password of admin to manage the app.

3- **Name: calendar.**
Attributes: key: id of calendar.
Description: Description of date.
Clock: clock of notification.
Date: date of event.

4- **Name: Accounts.**
Attributes: key: id of account.
Password: password of the account.
Type: type of account.
Username: user name or email of the account.

5- **Name: card.**
Attributes: key: key of card.
Name: name of the bank.
Date: exp date of the card.
Password: password of the card.
Number: account number.

6- **Name: IDs.**
Attributes: key: key of ID.
Type: type of the id.
Id number.

7- **Name: notes.**
Attributes: key: key of the note.
Details: details of the note.

8- **Name: photos.**
Attributes: key: key of the video or photo.
Type: type of the inputs.
url: URL of photo or video.

Description of Relations:

- The relationship between user and calendar is one to many so one user has many events to schedule.
- The relationship between admin and user is one to many so one admin can manage many user’s data.
- The relationship between user and cards, notes, IDs and accounts is one to many so one user have many cards, accounts to add or modify or delete …etc.
- The relationship between user and photos is one to many so one user has many photos or videos to add or delete.

3-3-3 Drawing ERD:

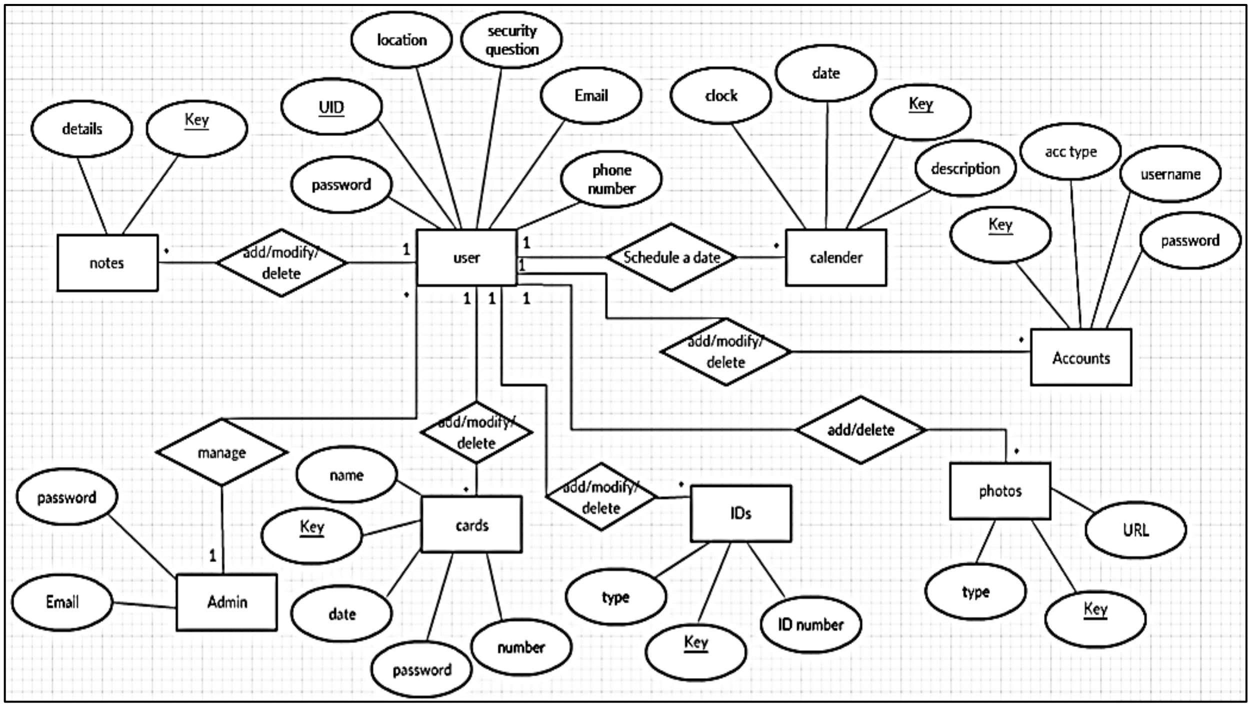


Figure (3-13) ERD diagram.

Chapter Four: System design

4-1Description of Procedures and Function:

The main procedures and function are mentioned in the following table:

Function	Description
add	The registered user of the application can add his data into a file for each type of data.
Delete	The registered user of the application can delete his saved data from any file.
Modify	The user can modify the saved data after selecting a specific file.
Reset password	The user can reset the password by entering the last password and then reset a new password.
Recover password	in case user forgot the password, it allows to recover the password and the admin will help to recover password.
View Data	User can see the latest inputs also he can access his data any time from any device.
Schedule event	User can Schedule his coming event through calendar.
Manage	This function is special for the admin of the app he can manage the app in term of updates, recover passwords, etc.

Table (4-1) Description of procedures and function

4-2 Relation Database Schema:

“A relational database schema is the tables, columns and relationships that make up a relational database. A relational database schema helps to organize and understand the structure of a database. This is particularly useful when designing a new database, modifying an existing database to support more functionality, or building integration between databases.” (Techwalla, 2018)

4-2-1 Tables:

In this project several tables required to implement our system.

1-User table.

2-Calender table.

3-Accounts table.

4-Cards.

5-IDs table.

6-Note table.

7-Photos table.

4-2-2 Attributes& Relations:

1- **User table**

Field name	Data type	Limitations
U_ID	string	Primary key
Password	string	
Address	string	
Email	string	
Question	string	
Phone Number	string	

Table (4-2) User table.

2-Calender table:

Field name	Data type	Limitations
U_ID	string	Foreign key
Et_desc	string	
Txt_clock	string	
Txt_desc	string	
keyAccount	string	Primary key

Table (4-3) calendar table.

3-Account table:

Field name	Data type	Limitations
U_ID	string	Foreign key
keyAccount	string	Primary key
passAccount	string	
typeAccount	string	
usernameAccount	string	

Table (4-4) Account table.

4- Cards table:

Field name	Data type	Limitations
U_ID	string	Foreign key
keycard	string	Primary key
bankname	string	
carddate	string	
cardpassword	string	
cardnumber	string	

Table (4-5) card table.

5- IDs table:

Field name	Data type	Limitations
U_ID	string	Foreign key
keyID	string	Primary key
typeid	string	
Ids	string	

Table (4-6) IDs table.

6-Notes table:

Field name	Data type	Limitations
U_ID	string	Foreign key
keynote	string	Primary key
notedetails	string	

Table (4-7) Notes table.

7-Photos table:

Field name	Data type	Limitations
U_ID	string	Foreign key
key	string	Primary key
type	string	
url	string	

Table (4-8) photos table.

1-3 Hardware and Software requirements:

1-3-1 software requirements:

In this project, we will use some complementary software to develop the secret pass application:

4-3-1-1 Firebase:

“Firebase provides a realtime database and backend as a service. The service provides application developers an API that allows application data to be synchronized across clients and stored on Firebase's cloud. The company provides client libraries that enable integration with Android, iOS, JavaScript, Java, Objective-C, Swift, and Node.” (En.wikipedia.org, 2018)

4-3-1-2 Android studio:

We will use the android studio to implement the application by write the code.

“Android Studio is the official integrated development environment (IDE) for Google's Android operating system, built on JetBrains' IntelliJ IDEA software and designed specifically for Android development. It is available for download on Windows, macOS and Linux based operating systems. It is a replacement for the Eclipse Android Development Tools (ADT) as primary IDE for native Android application development.” (En.wikipedia.org, 2018)

4-3-1-3 Adobe XD:

We will use the Adobe XD to design the application interfaces.

“Adobe XD is a user experience design software application developed and published by Adobe Systems. It supports vector design and wireframing.

“(En.wikipedia.org, 2018)

4-3-1-4 Creatly:

This software supports models, graphs, and charts. In this project, we used this software to draw some of the required UML diagrams such as (use case diagram, activity diagram, DFD, etc.

4-3-1-5 Google form:

This project applied questioner by using google form. Google form allows user to apply a questionnaire for free and share it with others.

4-3-1-6 Smart draw:

This software used in this project to draw diagrams such as sequence and object, smart draw support lots of diagrams.

4-3-1-7 Genymotion:

Simulator for Android phones.

4-3-2 Hardware requirements:

to implement this app, we need PC or laptop to download all required software.
internet router and to use this app we need a mobile device with android
OS with 20 API and more.

4-4 prototype of Interfaces:

4-4-1 Design of Android Application interfaces:

4-4-1-1 main interface

Through this screen, the user can log in if he is registered or registering in the application if he is a new user. (see figure 4-1)

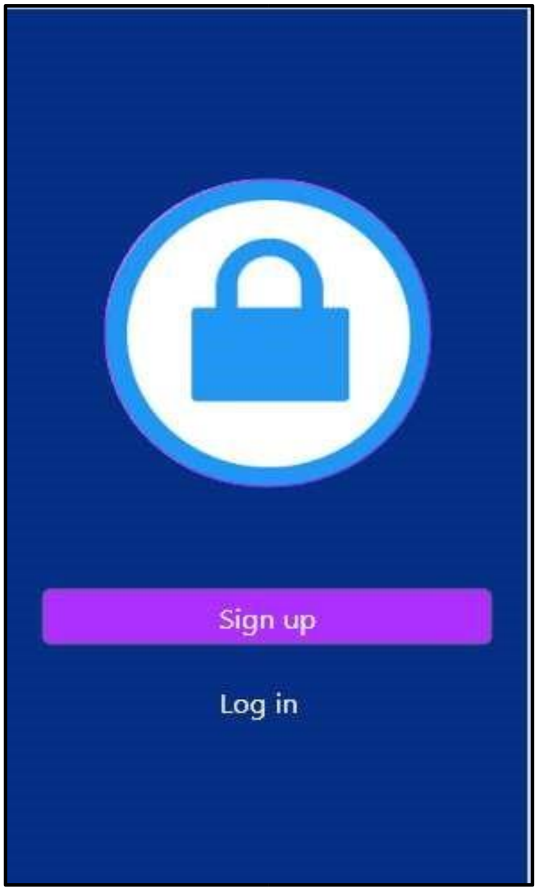


Figure (4-1) home interface of secret pass.

4-4-1-2 New registration interface:

The user enters all the required information and stores this information in the database.(see figure 4-2)

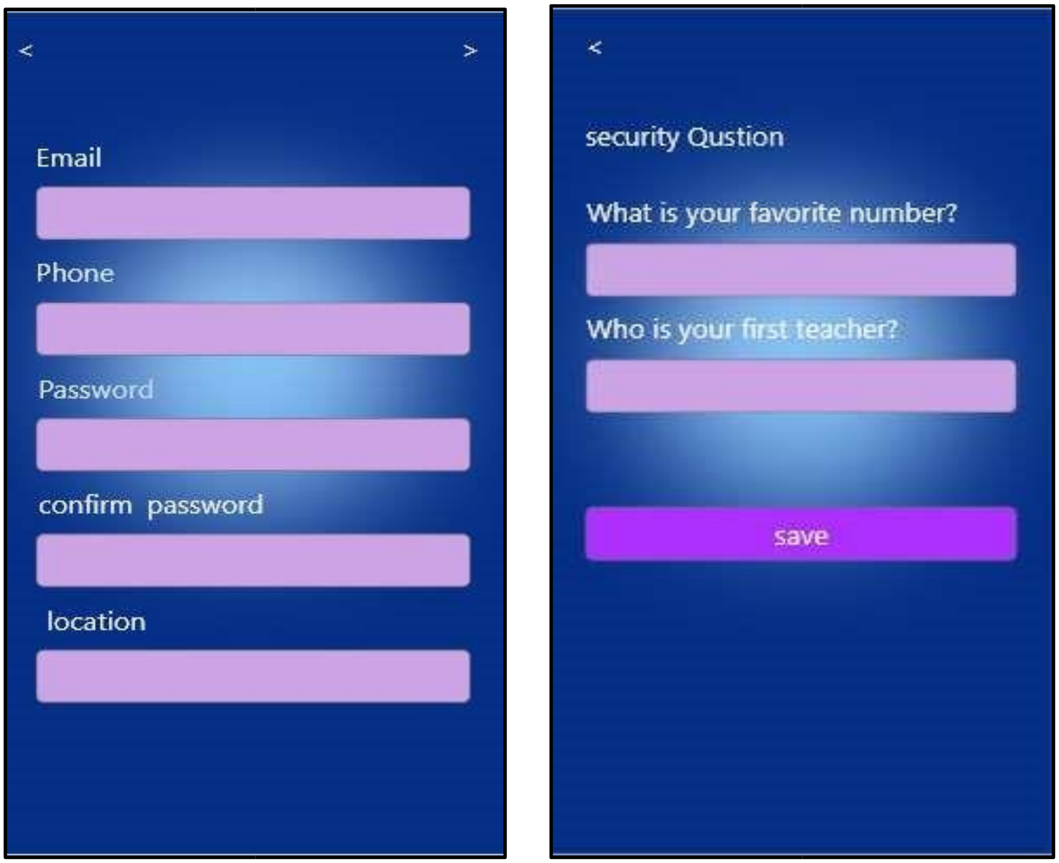


Figure (4-2) New registration interface.

4-4-1-3 Log in interface:

the user can log in if he is registered by entering the password shown in Figure (4-3).

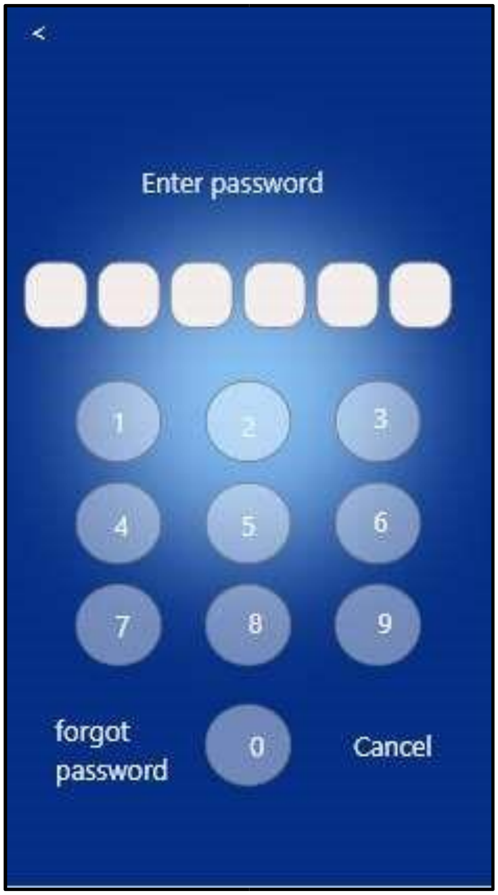


Figure (4-3) log in interface.

4-4-1-4 Application Home Interface:

The user can insert a file or select a file to save or delete the contents of that file. (see figure 4-4)

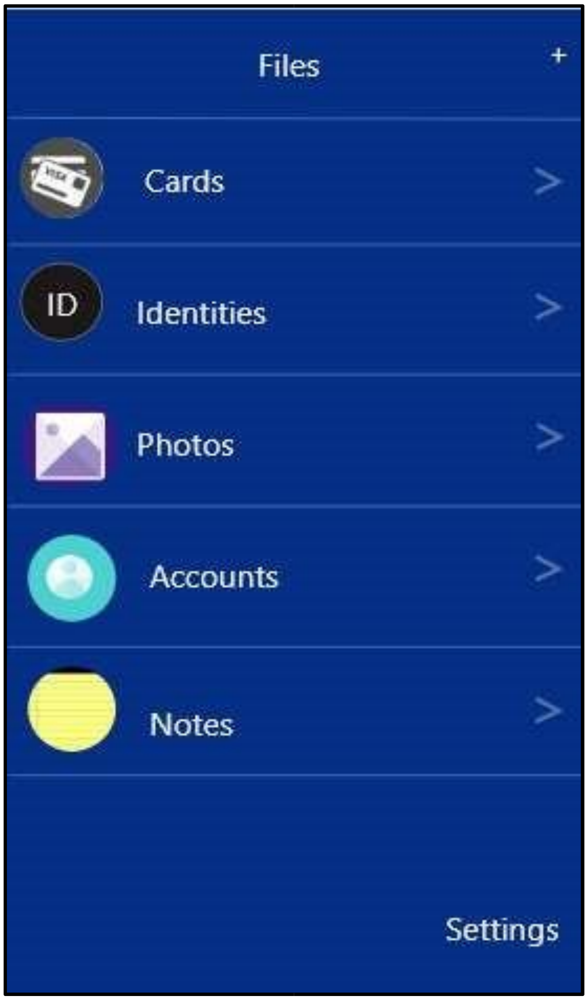


Figure (4-4) home interface.

4-4-1-5 select file interface (include all files with different process to add) :

The user can insert data after selecting a specific file. (see figure 4-5)

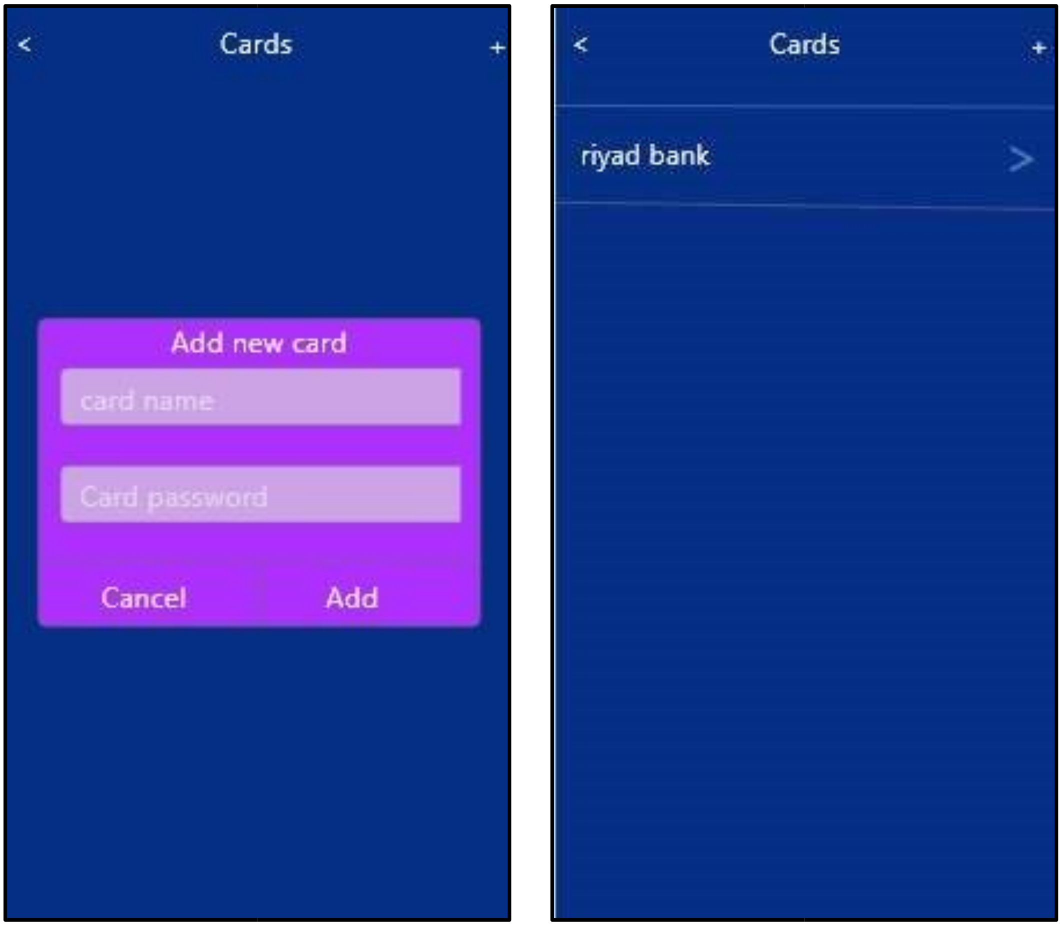
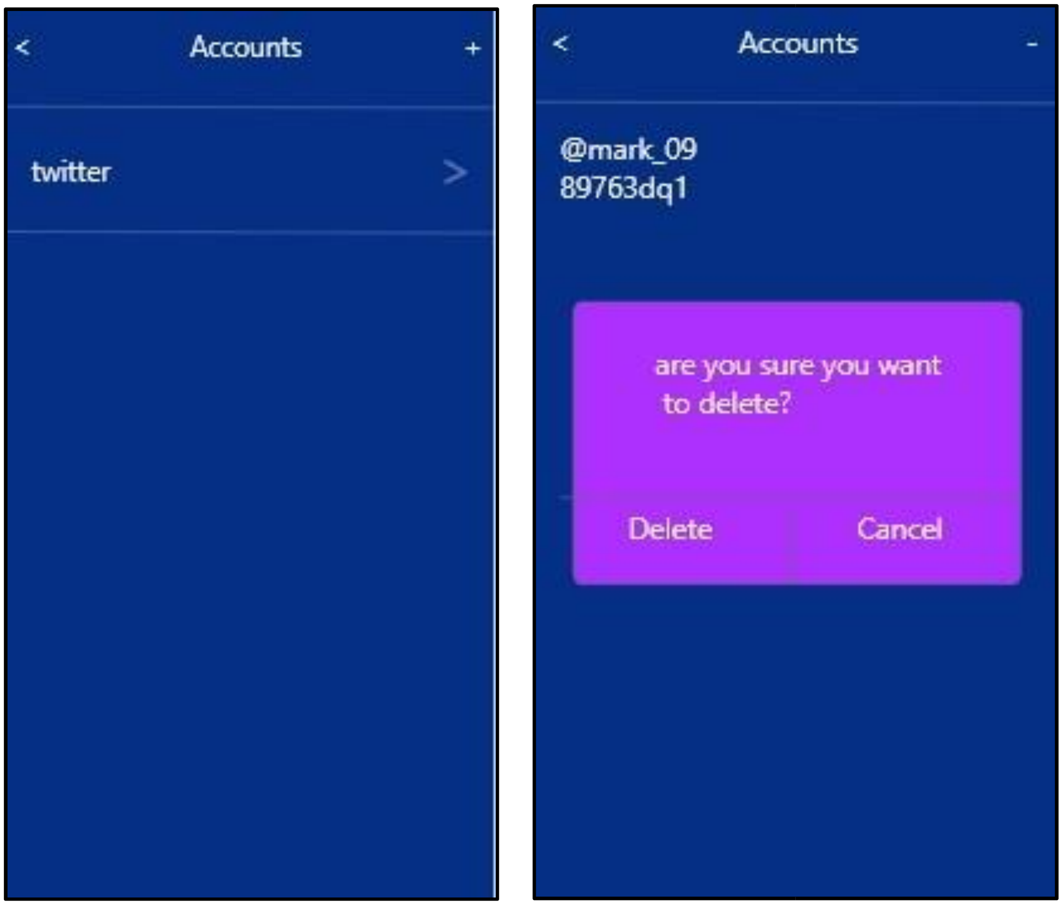


Figure (4-5) select file interface(add).

4-4-1-6 select file interface (include all files to delete) :

The user can delete data after selecting a specific file following this process. (see figure 4-6)



@mark_09
89763dq1

are you sure you want
to delete?

Figure (4-6) select file interface(delete).

4-4-1-7 reset Password interface:

From setting the user can reset the password by enter last password and new password. (see figure 4-7)

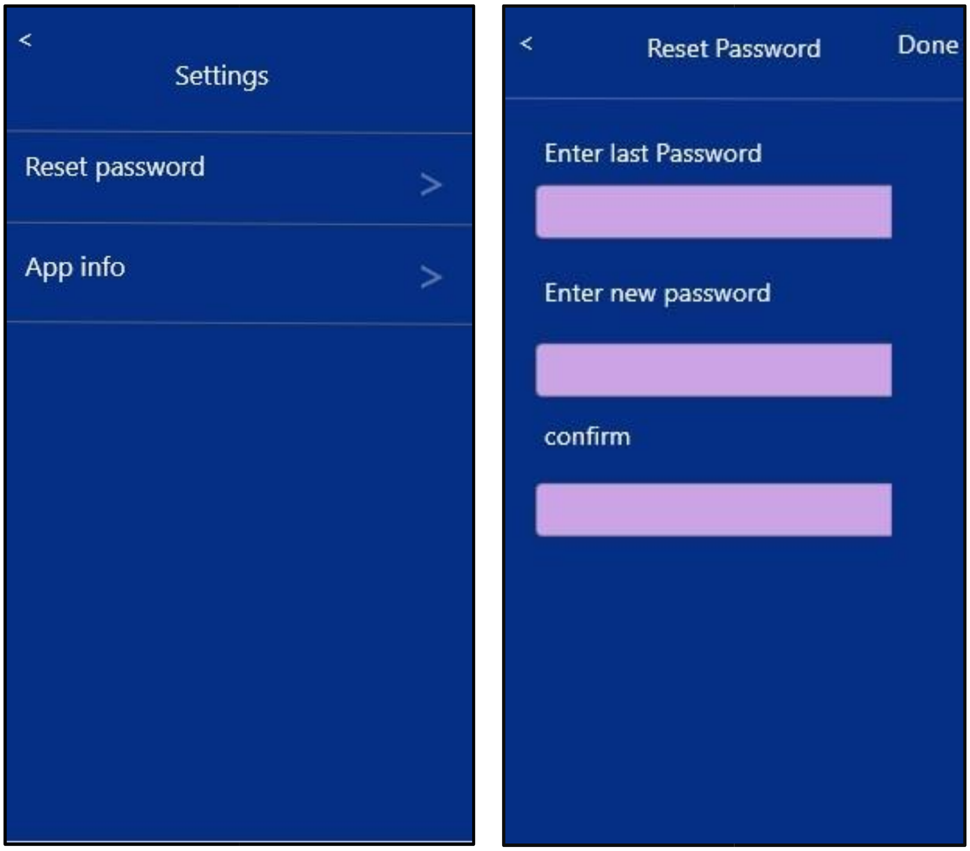
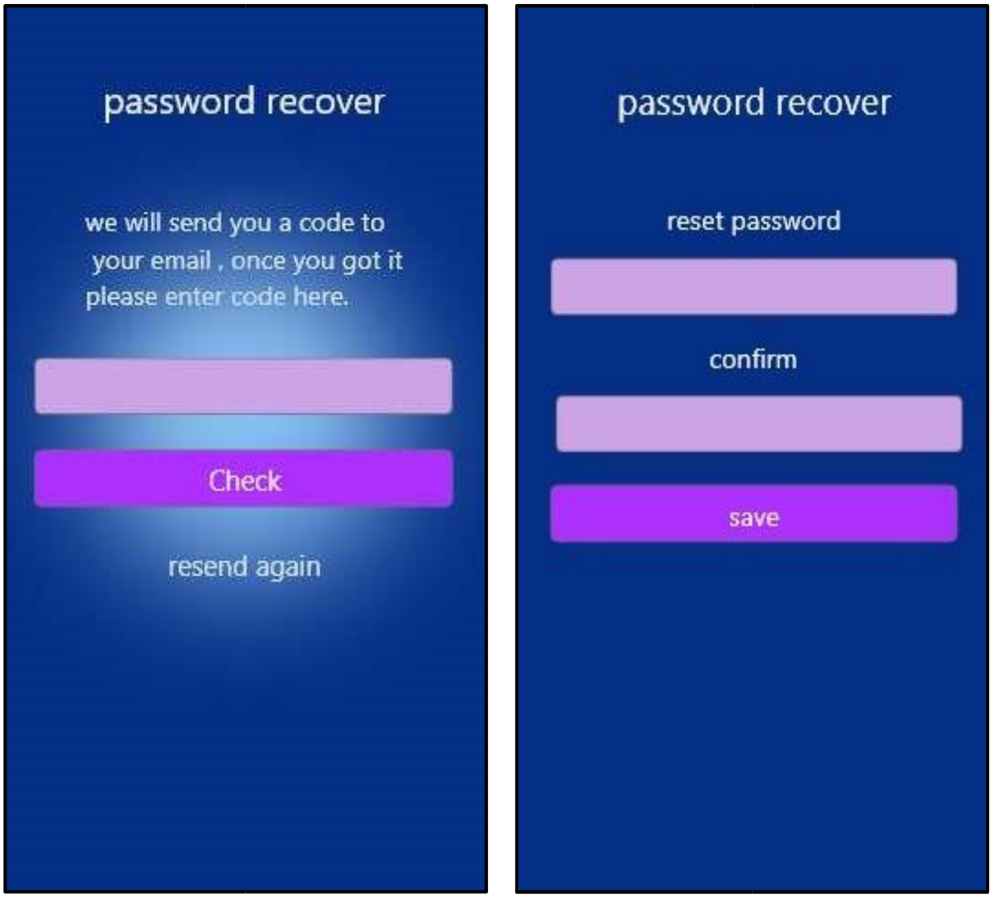


Figure (4-7) modify password interface.

3-4-1-8 Recover Password interface:

In case the user forgot the password, from this page will recover it.

The process of recover password the admin of application will send code to user email if the code corrects the use can enter new password and use the app if the code not correct the admin will resend again. (see figure 4-8)



we will send you a code to your email , once you got it please enter code here.

Figure (4-8) recover password interface.

Chapter Five: Implementation.

5-1 Introduction:

In order to complete this project, Java has been chosen for programming the application because it works on the Android system in addition to the link to the Firebase as a database supported by Google, which is an Online Database.

5-2 Procedures:

Basic functions used in this project as follow

Add:

```
findViewById(R.id.btn_add_ids).setOnClickListener(new View.OnClickListener(){
@Override
public void onClick(View v) {
final Dialog dialog = new Dialog(IDsActivity.this);
dialog.setContentView(R.layout.dialog_id_add);
WindowManager.LayoutParams lp = new WindowManager.LayoutParams();
lp.copyFrom(dialog.getWindow().getAttributes());
lp.width = WindowManager.LayoutParams.MATCH_PARENT;
lp.height = WindowManager.LayoutParams.WRAP_CONTENT;
dialog.findViewById(R.id.send_add).setOnClickListener(new
View.OnClickListener() {
@Override
public void onClick(View v) {
String id_ = ((AnyEditTextView)
dialog.findViewById(R.id.et_bankName)).getText().toString();
String et_type = ((AnyEditTextView)
dialog.findViewById(R.id.et_type)).getText().toString();
if (id_.isEmpty()||et_type.isEmpty()) {
Toast.makeText(IDsActivity.this, &quot;All fields are
required!&quot;, Toast.LENGTH_SHORT).show();
} else {
DatabaseReference mChildDatabase =
mDatabaseRef.child(&quot;IDs&quot;).push();
mChildDatabase.child(&quot;TypeID&quot;).setValue(et_type);
mChildDatabase.child(&quot;IDs&quot;).setValue(id_);
mChildDatabase.child(&quot;KeyID&quot;).setValue(mChildDatabase.getKey());
mChildDatabase.child(&quot;UIDUser&quot;).setValue(settings.getString(&quot;UID&quot;,
null));

JSONObject jsonObject=new JSONObject();
try {
jsonObject.put(&quot;TypeID&quot;,et_type);
jsonObject.put(&quot;IDs&quot;,id_);
jsonObject.put(&quot;KeyID&quot;,mChildDatabase.getKey());
jsonObject.put(&quot;UIDUser&quot;,settings.getString(&quot;UID&quot;,
null));}
```

Delete:

```
public void delete(final int position){
final Cards users1 = camList.get(position);
mDatabase =
FirebaseDatabase.getInstance().getReference().child(&quot;Cards&quot;);
Query applesQuery =
mDatabase.orderByChild(&quot;KeyCard&quot;).equalTo(users1.KeyCard);
applesQuery.addListenerForSingleValueEvent(new ValueEventListener() {
@Override
public void onDataChange(DataSnapshot dataSnapshot) {
for (DataSnapshot appleSnapshot: dataSnapshot.getChildren()) {
appleSnapshot.getRef().removeValue();
Toast.makeText(context, &quot;Delete successfully&quot;,
Toast.LENGTH_SHORT).show();
camList.remove(position);
bo();
}
```

Edit:

```
send_edit.setOnClickListener(new View.OnClickListener() {
@Override
public void onClick(View view) {
DatabaseReference mDatabase =
FirebaseDatabase.getInstance().getReference().child(&quot;Cards&quot;);
SharedPreferences settings = context.getSharedPreferences(&quot;USERS&quot;,
context.MODE_PRIVATE);
DatabaseReference mChildDatabase =mDatabase.child(users1.KeyCard);
mChildDatabase.child(&quot;BankName&quot;).setValue(et_bankName.getText().toString());
mChildDatabase.child(&quot;CardNumber&quot;).setValue(et_cardNum.getText().toString());
mChildDatabase.child(&quot;CardPassword&quot;).setValue(et_cvv.getText().toString());
mChildDatabase.child(&quot;CardDate&quot;).setValue(users1.CardDate);
mChildDatabase.child(&quot;UIDUser&quot;).setValue(settings.getString(&quot;UID&quot;,
null));
mChildDatabase.child(&quot;KeyCard&quot;).setValue(mChildDatabase.getKey());
Toast.makeText(context, &quot;Edit Successfully&quot;,
Toast.LENGTH_SHORT).show();
alertDialog.dismiss();
}});
```

Recover:

```
findViewById(R.id.btn_login).setOnClickListener(new View.OnClickListener() {
@Override
public void onClick(View view) {
email = editTextEmail.getText().toString();
if (TextUtils.isEmpty(email)) {
editTextEmail.setError(&quot;Fill Email&quot;);
error
} else { hud.show();
firebaseAuth.sendPasswordResetEmail(email).addOnCompleteListener(new
OnCompleteListener&lt;Void&gt;() {
@Override
public void onComplete(@NonNull Task&lt;Void&gt; task) {
hud.dismiss();
if (task.isSuccessful()) {
Toast.makeText(ForgetPasswordActivity.this, &quot;Your
password has been sent to your mail &quot;, Toast.LENGTH_SHORT).show();
finish();
} else {
Toast.makeText(ForgetPasswordActivity.this, &quot;Failed
to Send&quot;, Toast.LENGTH_SHORT).show();}}});}}});
```

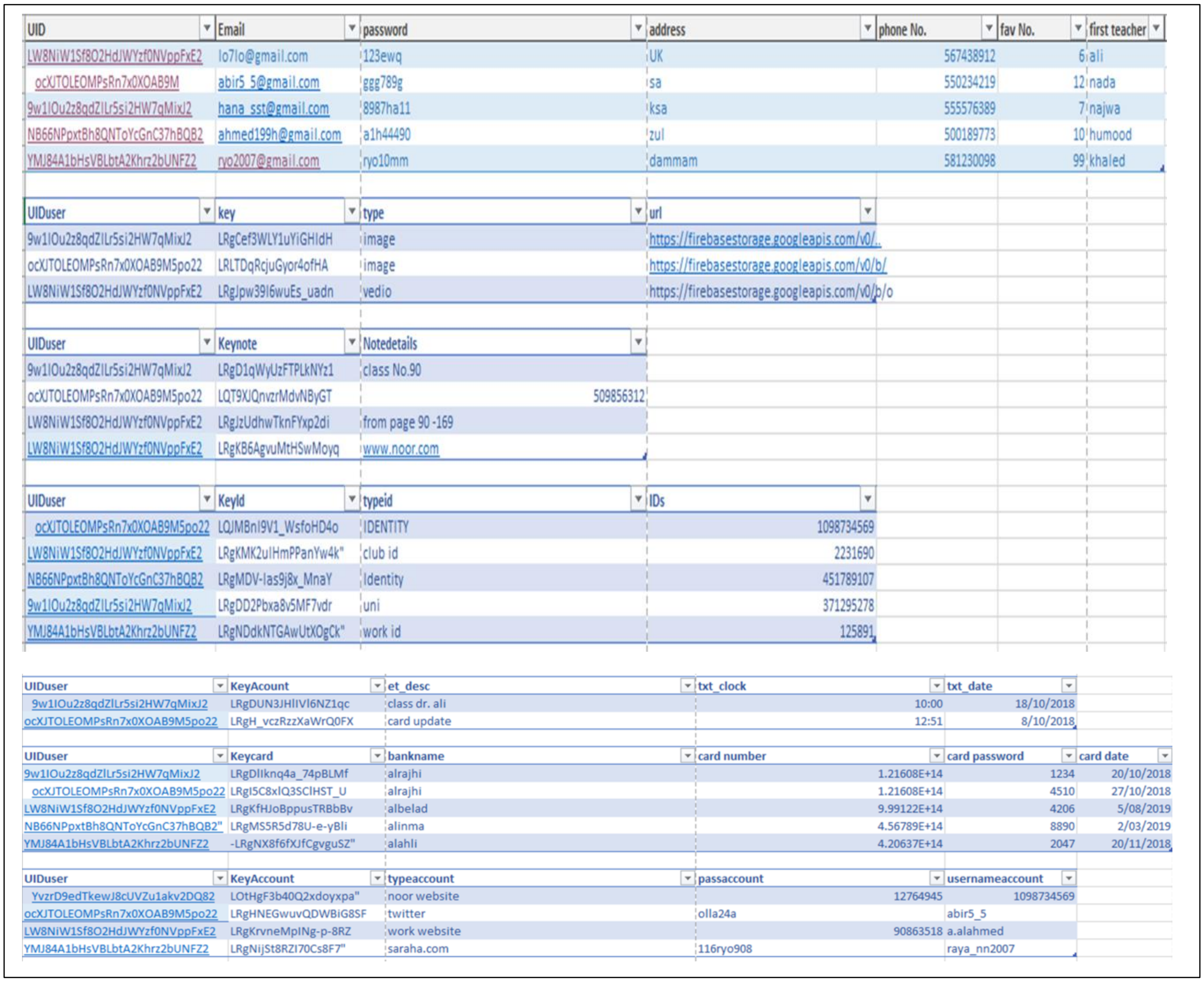
Reset:

```
findViewById(R.id.btn_register).setOnClickListener(new
View.OnClickListener() {
@Override
public void onClick(View v) {
final String oldpassword = ((AnyEditTextView)
findViewById(R.id.et_old_pass)).getText().toString();
final String password = ((AnyEditTextView)
findViewById(R.id.et_password)).getText().toString();
String cpassword = ((AnyEditTextView)
findViewById(R.id.et_c_password)).getText().toString();
if (oldpassword.isEmpty() || password.isEmpty() ||
cpassword.isEmpty()) {
Toast.makeText(ChangePassActivity.this, &quot;All fields are
required!&quot;, Toast.LENGTH_SHORT).show();
} else if (!password.equals(cpassword)) {
Toast.makeText(ChangePassActivity.this, &quot;Password and
password confirmation do not match&quot;, Toast.LENGTH_SHORT).show();
}
else {
hud.show();
```

```
final String email = Hawk.get(&quot;Email&quot;).toString();
mAuth.signInWithEmailAndPassword(email,oldpassword).addOnCompleteListener(ne
w OnCompleteListener&lt;AuthResult&gt;() {
@Override
public void onComplete(@NonNull Task&lt;AuthResult&gt;
task) {
user = mAuth.getInstance().getCurrentUser();
AuthCredential credential =
EmailAuthProvider.getCredential(email, oldpassword);
user.reauthenticate(credential).addOnCompleteListener(new
OnCompleteListener&lt;Void&gt;() {
@Override
public void onComplete(@NonNull Task&lt;Void&gt;
task) {
if (task.isSuccessful()) {
user.updatePassword(password).addOnCompleteListener(new
OnCompleteListener&lt;Void&gt;() {
@Override
public void onComplete(@NonNull
Task&lt;Void&gt; task) {
if (!task.isSuccessful()) {
hud.dismiss();
Toast.makeText(ChangePassActivity.this, &quot;Something went wrong. Please try
again later&quot;, Toast.LENGTH_SHORT).show();
} else {
Toast.makeText(ChangePassActivity.this, &quot;Change Password DONE&quot;,
Toast.LENGTH_SHORT).show();
hud.dismiss();
((AnyEditTextView)
findViewById(R.id.et_old_pass)).setText(&quot;&quot;);
((AnyEditTextView)
findViewById(R.id.et_password)).setText(&quot;&quot;);
((AnyEditTextView)
findViewById(R.id.et_c_password)).setText(&quot;&quot;);
}

}
});
} else {
hud.dismiss();
Toast.makeText(ChangePassActivity.this, &quot;Authentication Failed&quot;,
Toast.LENGTH_SHORT).show();
}
}});}});
```

5-3 Reports:



UID	Email	password	address	phone No.	fav No.	first teacher
LW8NiW1Sf8O2HdJWYzf0NVppFxE2	lo7lo@gmail.com	123ewq	UK	567438912	6	ali
ocXJTOLEOMPsRn7x0XOAB9M	abir5_5@gmail.com	ggg789g	sa	550234219	12	nada
9w1IOu2z8qdZILr5si2HW7qMixJ2	hana_sst@gmail.com	8987ha11	ksa	555576389	7	najwa
NB66NPpxtBh8QNToYcGnC37hBQB2	ahmed199h@gmail.com	a1h44490	zul	500189773	10	humood
YMJ84A1bHsVBLbtA2Khrz2bUNFZ2	ryo2007@gmail.com	ryo10mm	dammam	581230098	99	khaled

UIDuser	key	type	url
9w1IOu2z8qdZILr5si2HW7qMixJ2	LRgCef3WLY1uYiGHIdH	image	https://firebasestorage.googleapis.com/v0/..
ocXJTOLEOMPsRn7x0XOAB9M5po22	LRLTDqRcjuGyor4ofHA	image	https://firebasestorage.googleapis.com/v0/b/
LW8NiW1Sf8O2HdJWYzf0NVppFxE2	LRgJpw39I6wuEs_uadn	vedio	https://firebasestorage.googleapis.com/v0/b/o

UIDuser	Keynote	Notedetails
9w1IOu2z8qdZILr5si2HW7qMixJ2	LRgD1qWyUzFTPLkNYz1	class No.90
ocXJTOLEOMPsRn7x0XOAB9M5po22	LQT9XJQnvzrMdvNByGT	509856312
LW8NiW1Sf8O2HdJWYzf0NVppFxE2	LRgJzUdhwTknFYxp2di	from page 90 -169
LW8NiW1Sf8O2HdJWYzf0NVppFxE2	LRgKB6AgvuMtHSwMoyq	www.noor.com

UIDuser	KeyId	typeid	IDs
ocXJTOLEOMPsRn7x0XOAB9M5po22	LQJMBnI9V1_WsfoHD4o	IDENTITY	1098734569
LW8NiW1Sf8O2HdJWYzf0NVppFxE2	LRgKMK2uIHmPPanYw4k"	club id	2231690
NB66NPpxtBh8QNToYcGnC37hBQB2	LRgMDV-Ias9j8x_MnaY	Identity	451789107
9w1IOu2z8qdZILr5si2HW7qMixJ2	LRgDD2Pbxa8v5MF7vdr	uni	371295278
YMJ84A1bHsVBLbtA2Khrz2bUNFZ2	LRgNDdkNTGAwUtXOgCk"	work id	125891

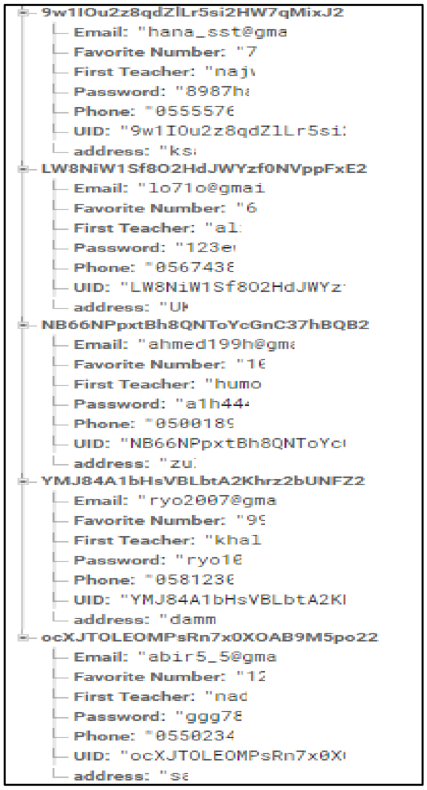
UIDuser	KeyAcount	et_desc	txt_clock	txt_date
9w1IOu2z8qdZlLr5si2HW7qMixJ2	LRgDUN3JHlIVl6NZ1qc	class dr. ali	10:00	18/10/2018
ocXJTOLEOMPsRn7x0XOAB9M5po22	LRgH_vczRzzXaWrQ0FX	card update	12:51	8/10/2018

UIDuser	Keycard	bankname	card number	card password	card date
9w1IOu2z8qdZlLr5si2HW7qMixJ2	LRgDlIknq4a_74pBLMf	alrajhi	1.21608E+14	1234	20/10/2018
ocXJTOLEOMPsRn7x0XOAB9M5po22	LRgI5C8xlQ3SClHST_U	alrajhi	1.21608E+14	4510	27/10/2018
LW8NiW1Sf8O2HdJWYzf0NVppFxE2	LRgKfHJoBppusTRBbBv	albelad	9.99122E+14	4206	5/08/2019
NB66NPpxtBh8QNToYcGnC37hBQB2"	LRgMS5R5d78U-e-yBli	alinma	4.56789E+14	8890	2/03/2019
YMJ84A1bHsVBLbtA2Khrz2bUNFZ2	-LRgNX8f6fXJfCgvguSZ"	alahli	4.20637E+14	2047	20/11/2018

UIDuser	KeyAccount	typeaccount	passaccount	usernameaccount
YvzrD9edTkewJ8cUVZu1akv2DQ82	LOtHgF3b40Q2xdoyxpa"	noor website	12764945	1098734569
ocXJTOLEOMPsRn7x0XOAB9M5po22	LRgHNEGwuvQDWBiG8SF	twitter	olla24a	abir5_5
LW8NiW1Sf8O2HdJWYzf0NVppFxE2	LRgKrvneMpINg-p-8RZ	work website	90863518	a.alahmed
YMJ84A1bHsVBLbtA2Khrz2bUNFZ2	LRgNijSt8RZI70Cs8F7"	saraha.com	116ryo908	raya_nn2007

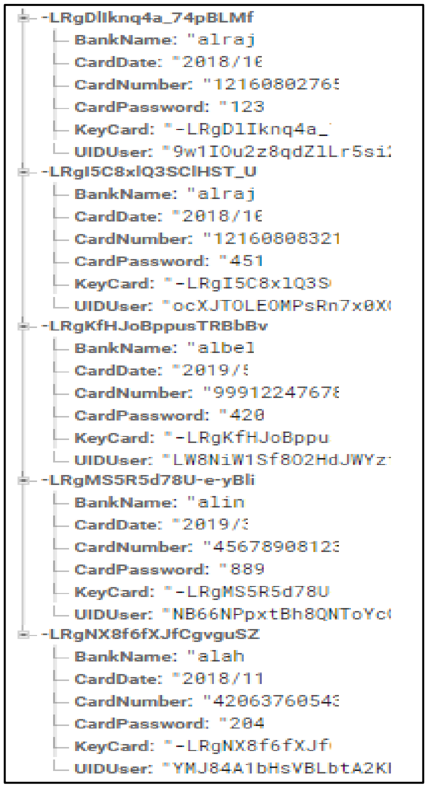
Figure (5-1) database table.

previous tables in firebase as following:



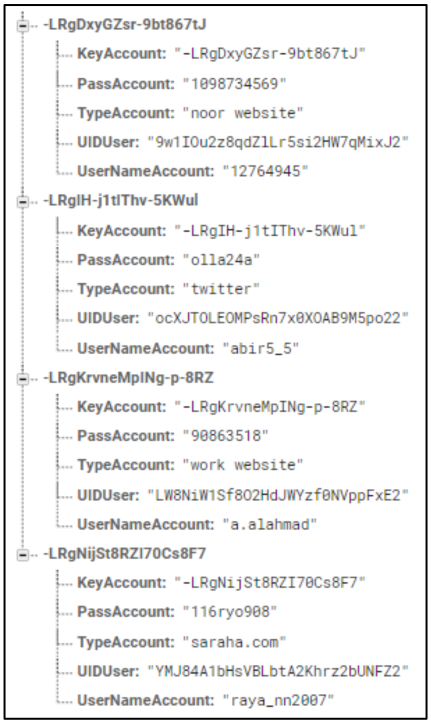
```
9w1IOu2z8qdZlLr5si2HW7qMixJ2
  Email: "hana_sst@gma
  Favorite Number: "7
  First Teacher: "naj
  Password: "8987h
  Phone: "055557
  UID: "9w1IOu2z8qdZ1Lr5si
  address: "ks
LW8NiW1Sf8O2HdJWYzf0NVppFxE2
  Email: "lo71o@gmai
  Favorite Number: "6
  First Teacher: "al
  Password: "123e
  Phone: "056743
  UID: "LW8NiW1Sf8O2HdJWYz
  address: "U
NB66NPpxtBh8QNToYcGnC37hBQB2
  Email: "ahmed199h@gm
  Favorite Number: "1
  First Teacher: "humo
  Password: "a1h44
  Phone: "050018
  UID: "NB66NPpxtBh8QNToYc
  address: "zu
YMJ84A1bHsVBLbtA2Khrz2bUNFZ2
  Email: "ryo2007@gma
  Favorite Number: "9
  First Teacher: "khal
  Password: "ryo1
  Phone: "058123
  UID: "YMJ84A1bHsVBLbtA2K
  address: "damm
ocXJTOLEOMPsRn7x0XOAB9M5po22
  Email: "abir5_5@gma
  Favorite Number: "1
  First Teacher: "nad
  Password: "ggg7
  Phone: "0550234
  UID: "ocXJTOLEOMPsRn7x0X
  address: "s
```

Figure (5-2) users in firebase.



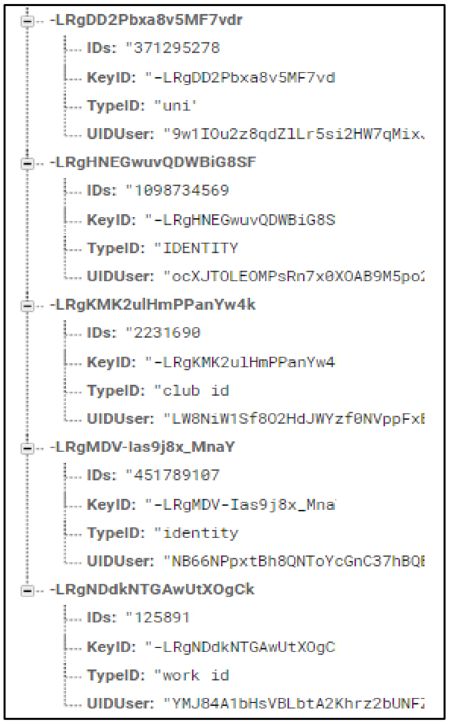
```
-LRgDlIknq4a_74pBLMf
  BankName: "alraj
  CardDate: "2018/1
  CardNumber: "1216080276
  CardPassword: "123
  KeyCard: "-LRgDlIknq4a_
  UIDUser: "9w1IOu2z8qdZ1Lr5si
-LRgI5C8xlQ3SClHST_U
  BankName: "alraj
  CardDate: "2018/1
  CardNumber: "12160808321
  CardPassword: "451
  KeyCard: "-LRgI5C8x1Q3S
  UIDUser: "ocXJTOLEOMPsRn7x0X
-LRgKfHJoBppusTRBbBv
  BankName: "albel
  CardDate: "2019/
  CardNumber: "9991224767
  CardPassword: "420
  KeyCard: "-LRgKfHJoBppu
  UIDUser: "LW8NiW1Sf8O2HdJWYz
-LRgMS5R5d78U-e-yBli
  BankName: "alin
  CardDate: "2019/
  CardNumber: "4567890812
  CardPassword: "889
  KeyCard: "-LRgMS5R5d78U
  UIDUser: "NB66NPpxtBh8QNToYc
-LRgNX8f6fXJfCgvguSZ
  BankName: "alah
  CardDate: "2018/11
  CardNumber: "4206376054
  CardPassword: "204
  KeyCard: "-LRgNX8f6fXJf
  UIDUser: "YMJ84A1bHsVBLbtA2K
```

Figure (5-3) cards in firebase.



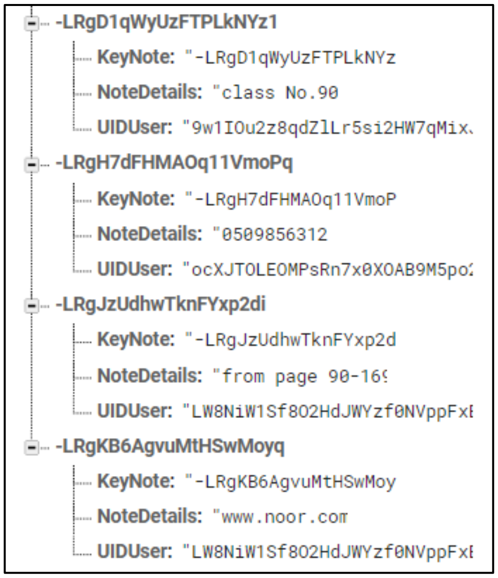
```
-LRgDxyGZsr-9bt867tJ
  KeyAccount: "-LRgDxyGZsr-9bt867tJ"
  PassAccount: "1098734569"
  TypeAccount: "noor website"
  UIDUser: "9w1IOu2z8qdZ1Lr5si2HW7qMixJ2"
  UserNameAccount: "12764945"
-LRgIH-j1tIThv-5KWul
  KeyAccount: "-LRgIH-j1tIThv-5KWul"
  PassAccount: "olla24a"
  TypeAccount: "twitter"
  UIDUser: "ocXJTOLEOMPsRn7x0XOAB9M5po22"
  UserNameAccount: "abir5_5"
-LRgKrvneMpINg-p-8RZ
  KeyAccount: "-LRgKrvneMpINg-p-8RZ"
  PassAccount: "90863518"
  TypeAccount: "work website"
  UIDUser: "LW8NiW1Sf8O2HdJWYzf0NVppFxE2"
  UserNameAccount: "a.alahmad"
-LRgNijSt8RZI70Cs8F7
  KeyAccount: "-LRgNijSt8RZI70Cs8F7"
  PassAccount: "116ryo908"
  TypeAccount: "saraha.com"
  UIDUser: "YMJ84A1bHsVBLbtA2Khrz2bUNFZ2"
  UserNameAccount: "raya_nn2007"
```

Figure (5-4) accounts in firebase.

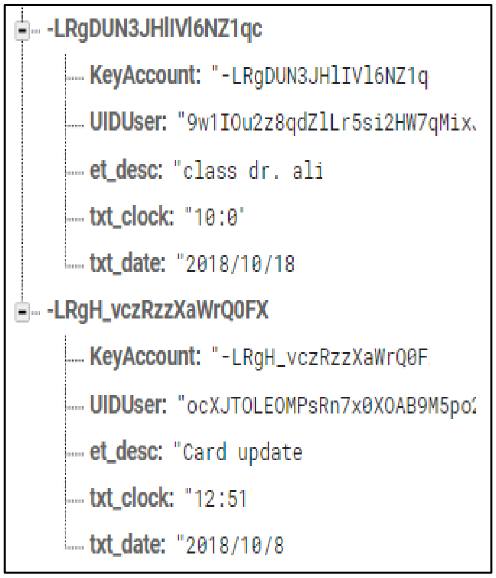


```
-LRgDD2Pbxa8v5MF7vdr
  IDs: "371295278
  KeyID: "-LRgDD2Pbxa8v5MF7vd
  TypeID: "uni'
  UIDUser: "9w1IOu2z8qdZ1Lr5si2HW7qMix
-LRgHNEGwuvQDWBiG8SF
  IDs: "1098734569
  KeyID: "-LRgHNEGwuvQDWBiG8S
  TypeID: "IDENTITY
  UIDUser: "ocXJTOLEOMPsRn7x0XOAB9M5po
-LRgKMK2ulHmPPanYw4k
  IDs: "2231690
  KeyID: "-LRgKMK2ulHmPPanYw4
  TypeID: "club id
  UIDUser: "LW8NiW1Sf8O2HdJWYzf0NVppFx
-LRgMDV-Ias9j8x_MnaY
  IDs: "451789107
  KeyID: "-LRgMDV-Ias9j8x_Mna
  TypeID: "identity
  UIDUser: "NB66NPpxtBh8QNToYcGnC37hBQ
-LRgNDdkNTGAwUtXOgCk
  IDs: "125891
  KeyID: "-LRgNDdkNTGAwUtXOgC
  TypeID: "work id
  UIDUser: "YMJ84A1bHsVBLbtA2Khrz2bUNF
```

Figure (5-5) IDs in firebase.



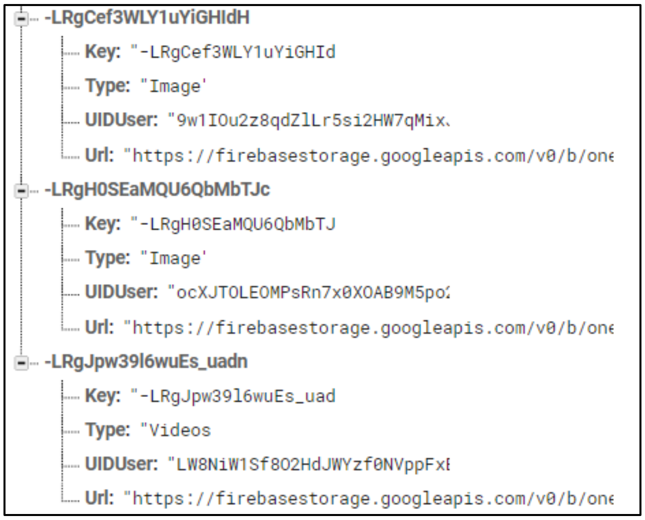
```
-LRgD1qWyUzFTPLkNYz1
    KeyNote: "-LRgD1qWyUzFTPLkNYz
    NoteDetails: "class No.90
    UIDUser: "9w1IOu2z8qdZlLr5si2HW7qMix
-LRgH7dFHMAOq11VmoPq
    KeyNote: "-LRgH7dFHMAOq11VmoP
    NoteDetails: "0509856312
    UIDUser: "ocXJTOLEOMPsRn7x0XOAB9M5po
-LRgJzUdhwTknFYxp2di
    KeyNote: "-LRgJzUdhwTknFYxp2d
    NoteDetails: "from page 90-16
    UIDUser: "LW8NiW1Sf8O2HdJWYzf0NVppFx
-LRgKB6AgvuMtHSwMoyq
    KeyNote: "-LRgKB6AgvuMtHSwMoy
    NoteDetails: "www.noor.com
    UIDUser: "LW8NiW1Sf8O2HdJWYzf0NVppFx
```



```
-LRgDUN3JHlIVl6NZ1qc
    KeyAccount: "-LRgDUN3JHlIVl6NZ1q
    UIDUser: "9w1IOu2z8qdZlLr5si2HW7qMix
    et_desc: "class dr. ali
    txt_clock: "10:0'
    txt_date: "2018/10/18
-LRgH_vczRzzXaWrQ0FX
    KeyAccount: "-LRgH_vczRzzXaWrQ0F
    UIDUser: "ocXJTOLEOMPsRn7x0XOAB9M5po
    et_desc: "Card update
    txt_clock: "12:51
    txt_date: "2018/10/8
```

Figure (5-6) Notes in firebase.

Figure (5-7) calendar in firebase



```
-LRgCef3WLY1uYiGHIdH
    Key: "-LRgCef3WLY1uYiGHId
    Type: "Image'
    UIDUser: "9w1IOu2z8qdZlLr5si2HW7qMix
    Url: "https://firebasestorage.googleapis.com/v0/b/one
-LRgH0SEaMQU6QbMbTJc
    Key: "-LRgH0SEaMQU6QbMbTJ
    Type: "Image'
    UIDUser: "ocXJTOLEOMPsRn7x0XOAB9M5po
    Url: "https://firebasestorage.googleapis.com/v0/b/one
-LRgJpw39l6wuEs_uadn
    Key: "-LRgJpw39l6wuEs_uad
    Type: "Videos
    UIDUser: "LW8NiW1Sf8O2HdJWYzf0NVppFx
    Url: "https://firebasestorage.googleapis.com/v0/b/one
```

Figure (5-8) photos in firebase

5-4 Layouts:

5-4-1 Main interface

The app's main interface after opening it, includes registration button for new users and login button for pre-registered users in the application.

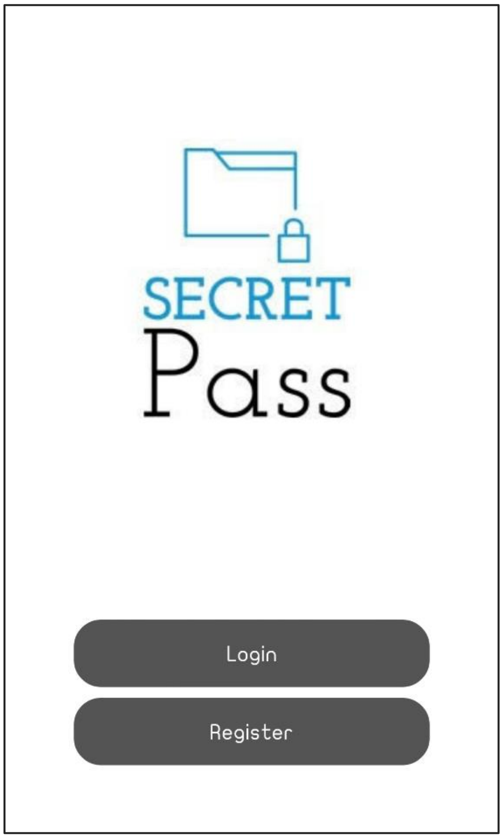


Figure (5-9) main interface.

5-4-2 Register interface

A new user registration interface includes entering the user's email, phone number and password to enter the application with confirmation of it.

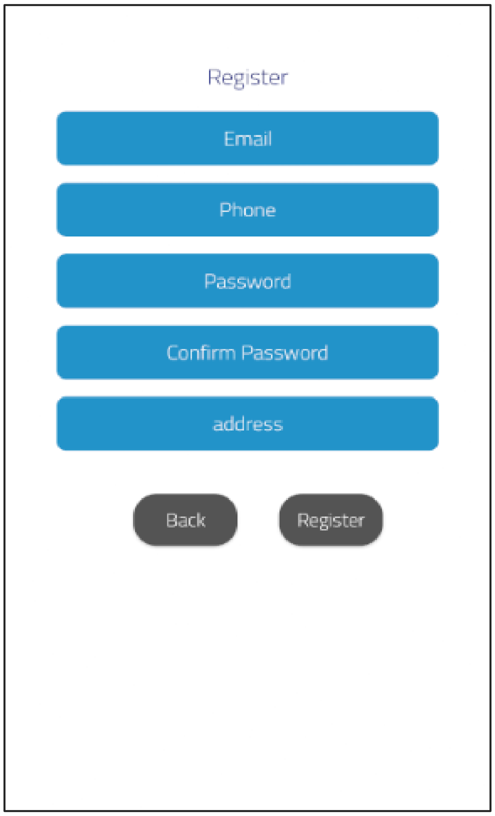


Figure (5-10) register interface

5-4-3 Log in interface

Once the user has finished registration, he can sign in to the application by entering his email with the password.

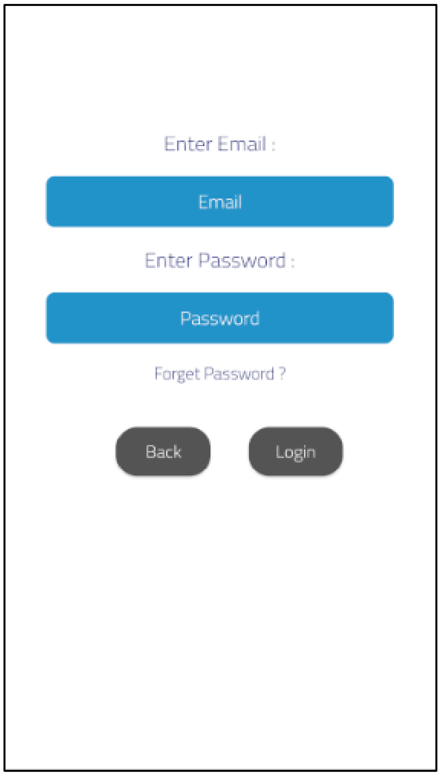


Figure (5-11) log in interface

5-4-4 Forgot Password Interface

If the user forgets the password, it can be retrieved via e-mail after sending an e-mail message to the user to reset.

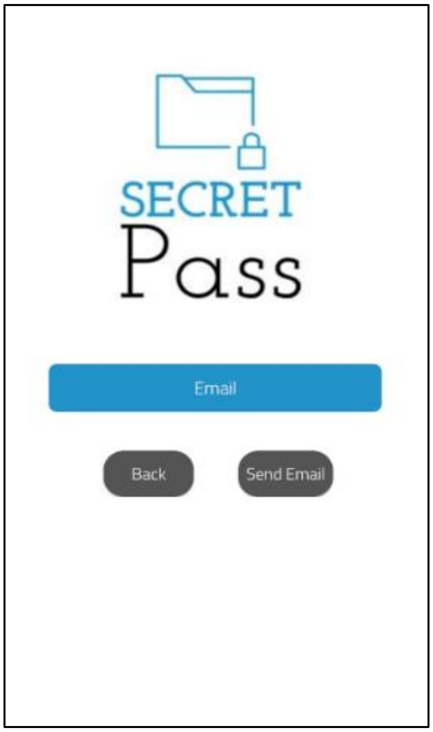


Figure (5-12) forgot password interface.

5-4-5 Home interface

If the user's password is correct, he will be able to access the applications home screen, which contains a list of files, calendar, timeline and settings.

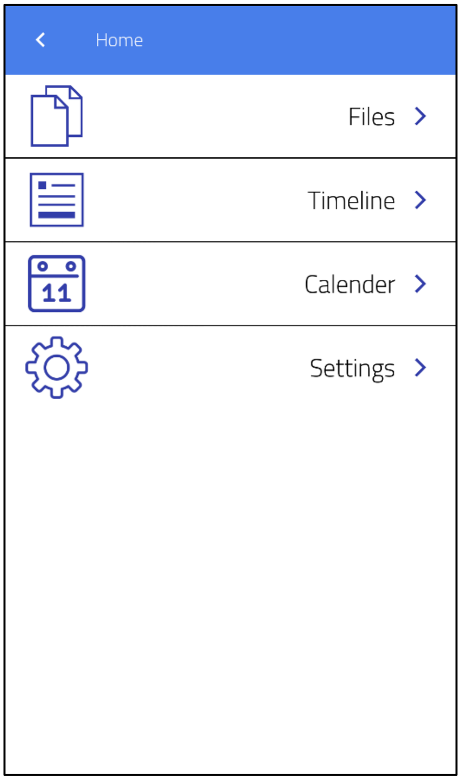


Figure (5-13) home interface

5-4-6 Files interface

When the user enters to the files, this interface will be displayed that contains a list of contents like cards file and file for the user ID, user photos, user accounts and notes.

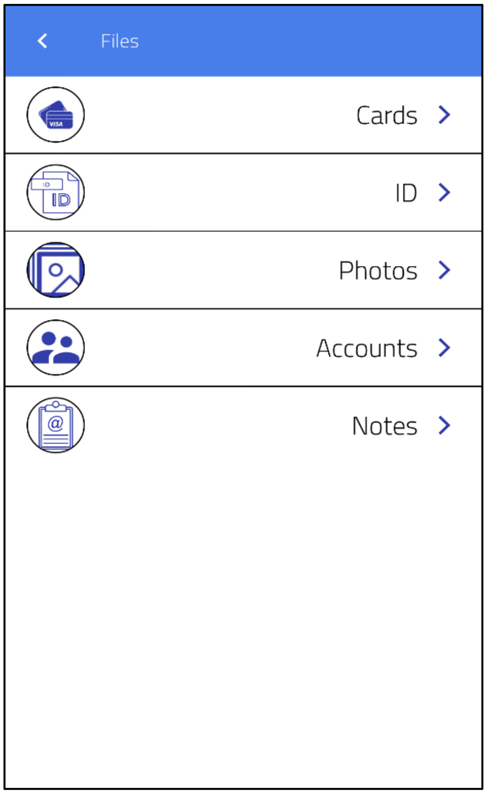


Figure (5-14) files interface.

5-4-7 Calendar Interface

The user can set the dates of expiration of his cards or set any special date or coming events by entering the calendar and insert the date and time with a description of the alert and this app gives an alarm to remind the user.

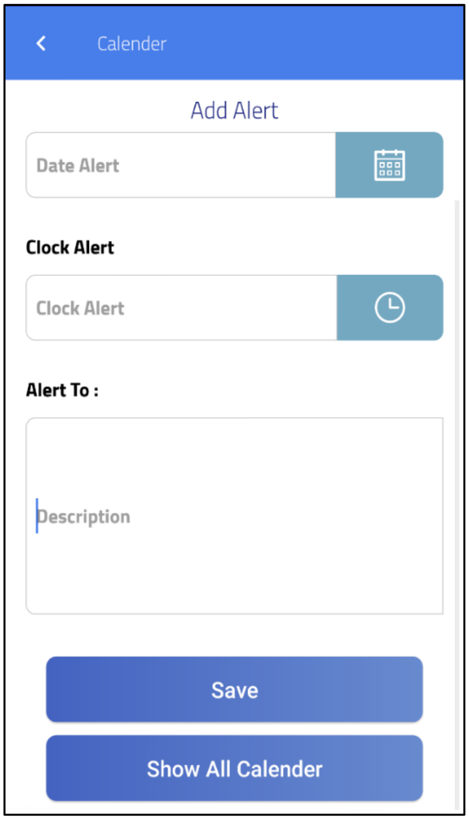


Figure (5-15) calendar interface.

5-4-8 Reset Password Interface

If the user wishes to change his password, he can reset it by entering the previous password then the new password entered and confirm it after that press save to save the changes.

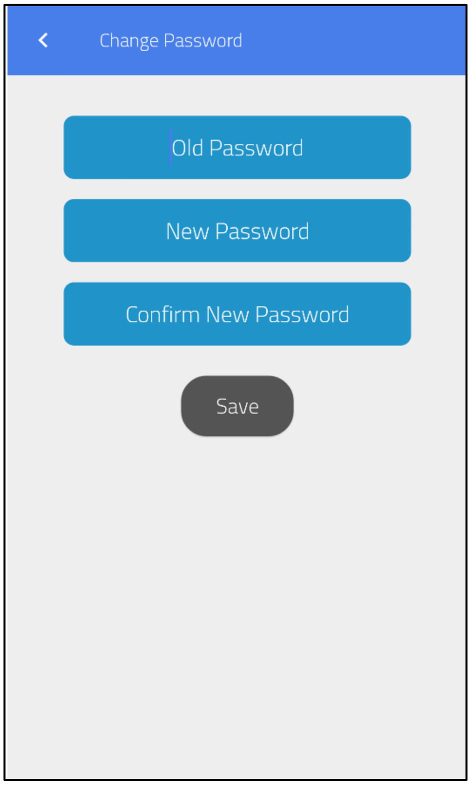
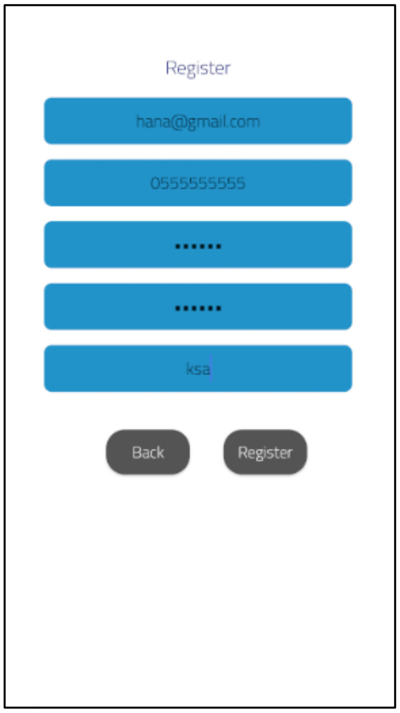


Figure (5-16) reset password interface

5-5 Reports layouts: some interface after running the app.



hana@gmail.com

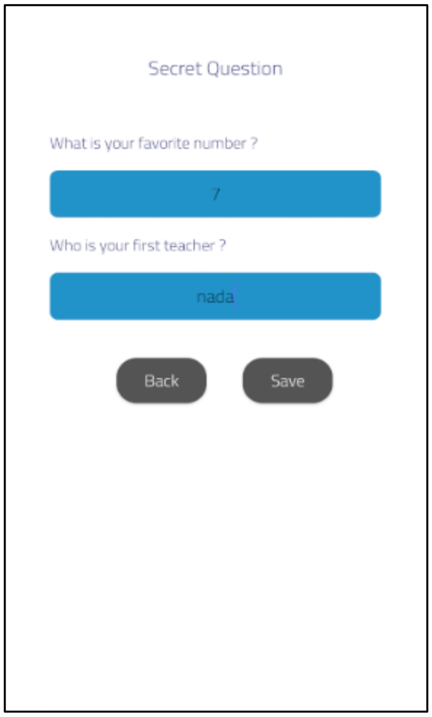
0555555555

••••••

••••••

ksa

Figure (5-17) Registration



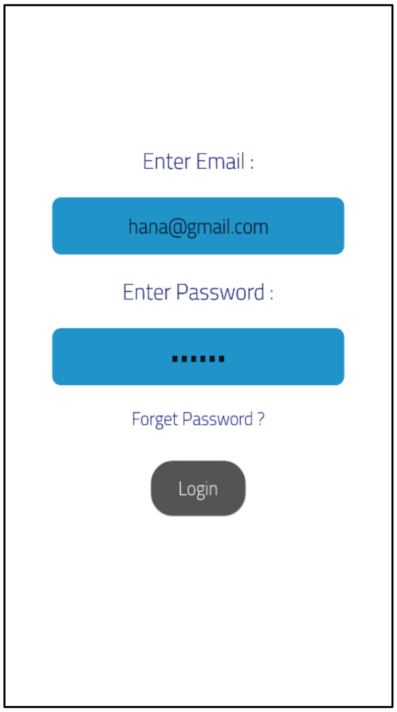
What is your favorite number ?

7

Who is your first teacher ?

nada

Figure (5-18) secret question



Enter Email :

hana@gmail.com

Enter Password :

••••••

Figure (5-19) Log in

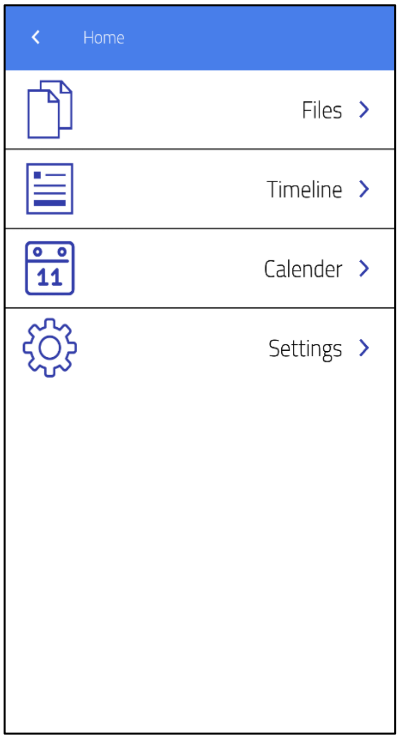


Figure (5-20) Files

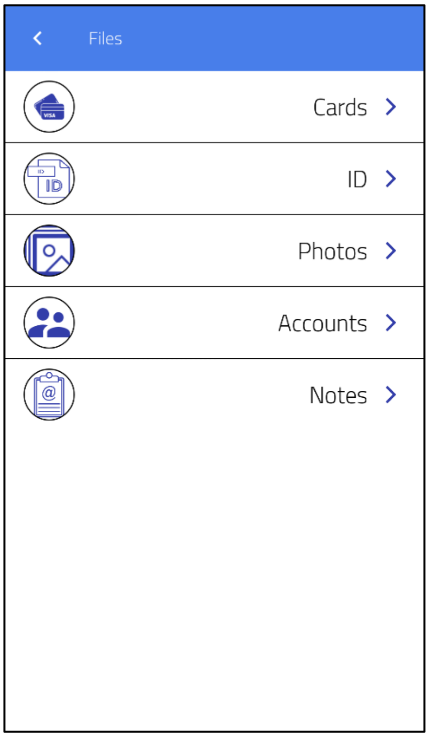
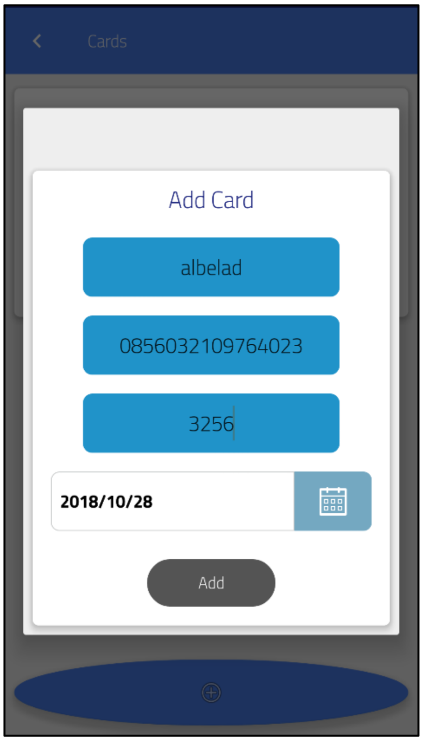


Figure (5-21) cards



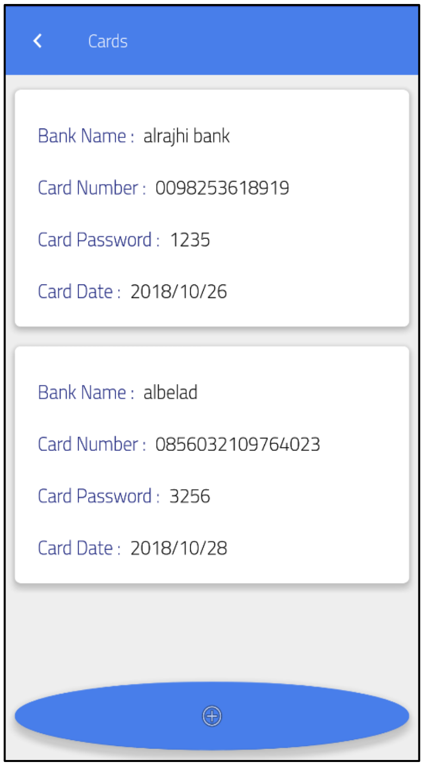
albelad

0856032109764023

3256

2018/10/28

Figure (5-22) add new card



Bank Name : alrajhi bank

Card Number : 0098253618919

Card Password : 1235

Card Date : 2018/10/26

Bank Name : albelad

Card Number : 0856032109764023

Card Password : 3256

Card Date : 2018/10/28

Figure (5-23) Card added

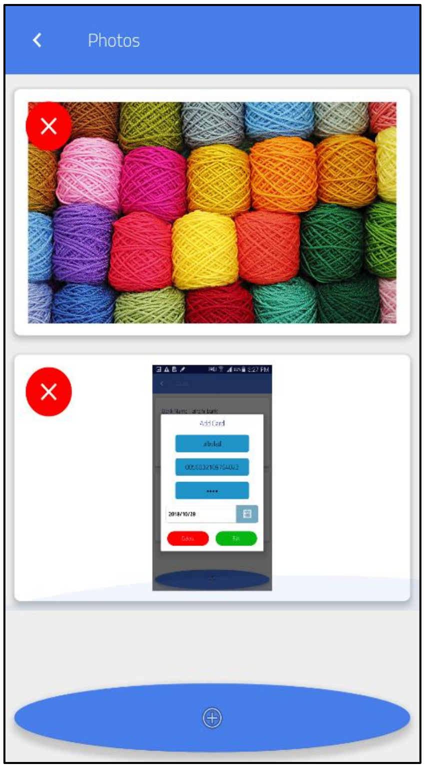
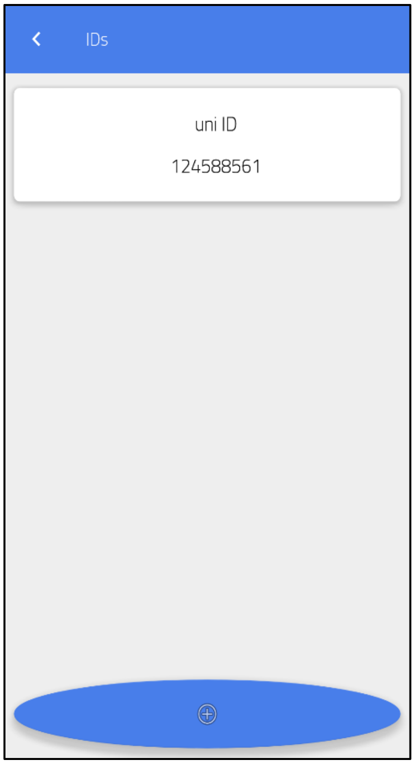


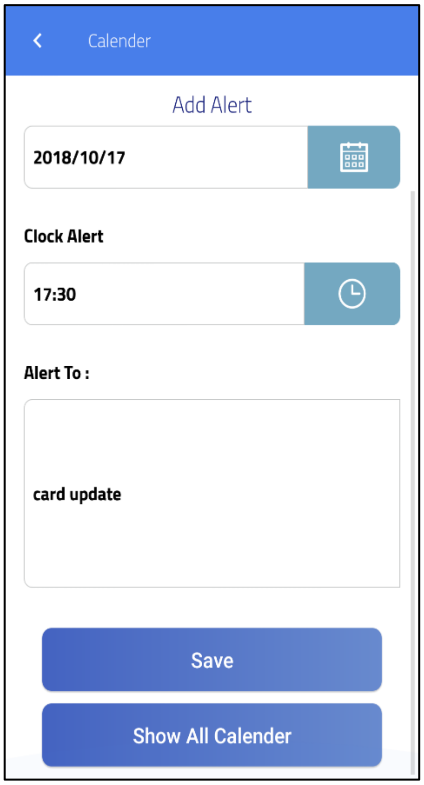
Figure (5-24) Photo added



uni ID

124588561

Figure (5-25) ID added



2018/10/17

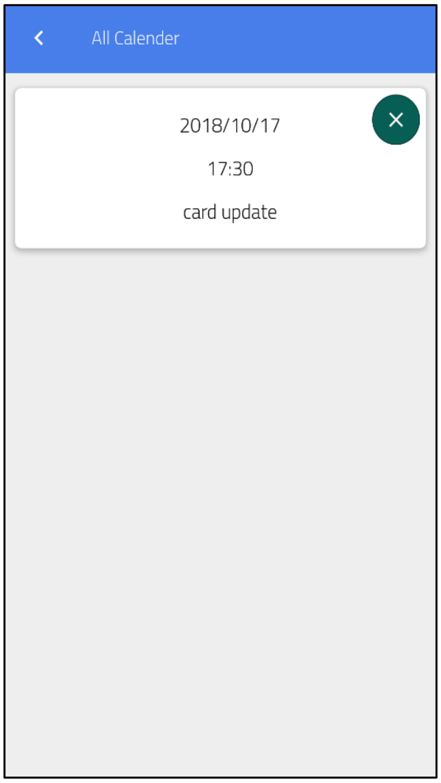
Clock Alert

17:30

Alert To :

card update

Figure (5-26) Add new alert

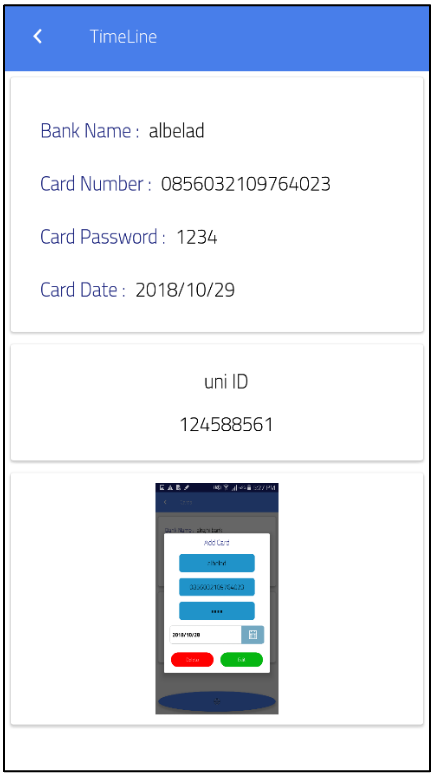


2018/10/17

17:30

card update

Figure (5-27) alert added



Bank Name : albelad

Card Number : 0856032109764023

Card Password : 1234

Card Date : 2018/10/29

uni ID

124588561

Figure (5-28) Timeline

Chapter Six: Conclusion and future works.

6-1 Conclusion:

in the conclusion of this work the result achieved is great and satisfy the project requirements, when user dealing with this application, they will be able to keep their data safe from loss, theft and other problems that they usually suffer, it supports several types of data also allows you to recover your data any time.

the application developed to give the user log in interface with email and password, the other layout give the chance to select upon different categorize after that the data will be kept and protected against unauthorized access, one more property of the application that it can help to arrange the user schedule for important events and accordingly give an alarm to remind the user.

6-2 Future works:

At the end, I recommend implementing this application on different platforms with periodic updates to achieve more efficient and useful features such as allowing the user to add a file to the list of existing files, restore the password by phone number and other features that meet needs of the users.

Chapter Seven: References.

- App Store. (2018). Private Notes and Secret Diary on the App Store. [online] Available at: https://itunes.apple.com/us/app/private-notes-andsecret-diary/id982348636?mt=8 [Accessed 26 feb. 2018].
- App Store. (2018). Touch Vault (photos and video) on the App Store. [online] Available at: https://itunes.apple.com/us/app/touch-vault-photosand-video/id916977998?mt=8 [Accessed 26 Feb. 2018].
- En.wikipedia.org. (2018). Adobe XD. [online] Available at: https://en.wikipedia.org/wiki/Adobe_XD [Accessed 23 Mar. 2018].
- En.wikipedia.org. (2018). Android Studio. [online] Available at: https://en.wikipedia.org/wiki/Android_Studio [Accessed 23 Mar. 2018].
- En.wikipedia.org. (2018). Data flow diagram. [online] Available at: https://en.wikipedia.org/wiki/Data_flow_diagram [Accessed 6 Mar. 2018].
- En.wikipedia.org. (2018). *Systems analysis*. [online] Available at: https://en.wikipedia.org/wiki/Systems_analysis [Accessed 4 Mar. 2018].
- En.wikipedia.org. (2018). *Firebase*. [online] Available at: https://en.wikipedia.org/wiki/Firebase#Realtime_Database [Accessed 23 Mar. 2018].
- Play.google.com. (2018). [online] Available at: https://play.google.com/store/apps/details?id=com.ek.privatecontacts&hl=ar [Accessed 27 Feb. 2018].
- Techwalla. (2018). What Is Relational Database Schema? | Techwalla.com. [online] Available at: https://www.techwalla.com/articles/what-is-relationaldatabase-schema [Accessed 23 Mar. 2018].

Chapter Eight: Appendixes.

Appendixes A (Questionnaire):

1-Do you have data you fear losing?

o Yes.

o No.

2-Have you ever lost your data before?

o Yes.

o No.

3- Are you worry that your data will be misused and that it will harm you?

o Yes.

o No.

4-do you trust giving others your passwords?

o Yes.

o No.

5-Do you save your data in public devices that are used by others?

o Yes.

o No.

6-Do you think your mobile phone is safe to keep your data private?

o Yes.

o No.

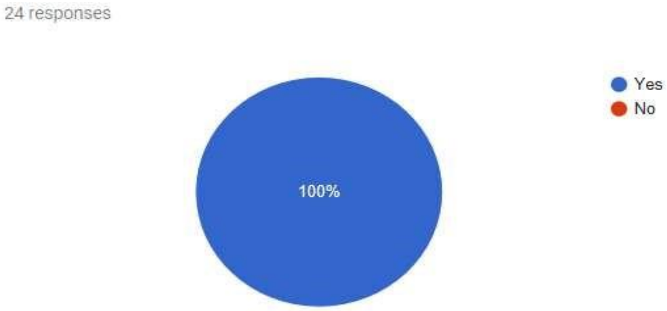
7- Do you prefer to use an application that protects your data of all types?

o Yes.

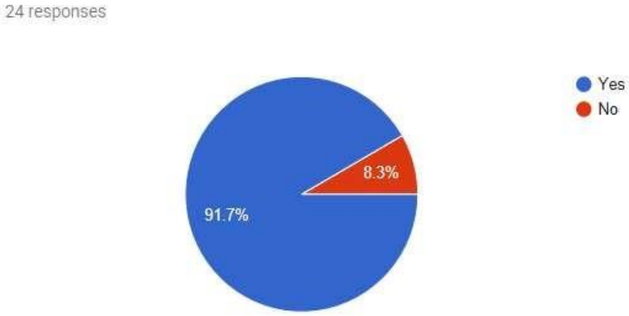
o No.

Questionnaire result:

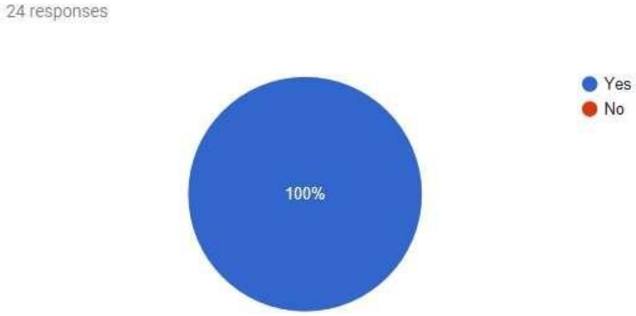
1-Do you have data you fear losing?



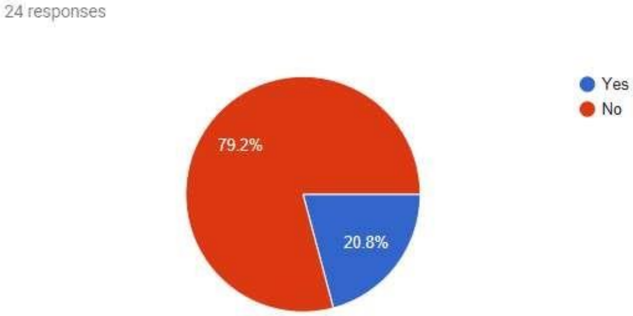
2-Have you ever lost your data before?



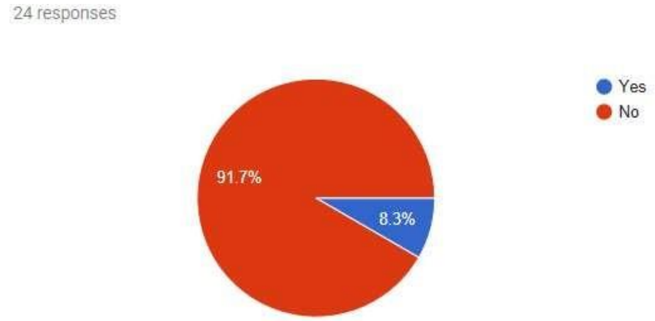
3- Are you worry that your data will be misused and that it will harm you?



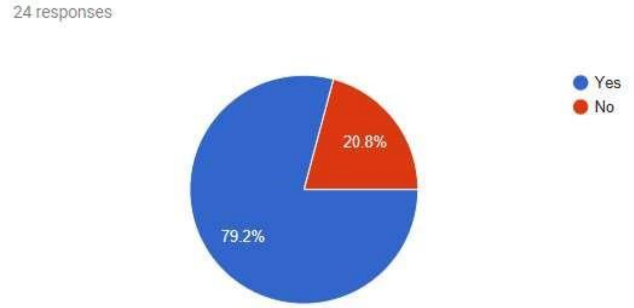
4-do you trust giving others your passwords?



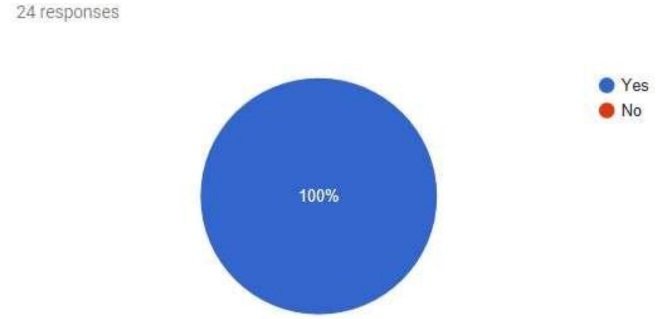
5-Do you save your data in public devices that are used by others?



6-Do you think your mobile phone is safe to keep your data private?



7- Do you prefer to use an application that protects your data of all types?



This Questionnaire applied and I get 24 response based on this responses the result is 100% from responses they fear to lose their data, 91.7% from responses they lost their data before and 8.3& they don’t lose any data, 100% from response they worry about misuse of their data after losing it, 79.2% do not trust giving others them password while 20.8% trust others such as friends, 91.7% from response they do not save private data in public devices but 8.3% do,79.2% think the mobile phone is safe to keep private data while 20.8% do not think that and 100% from responses prefer to use an app that protect their data.

Code of registration :

```
package com.universety.projectsecret;
import android.content.Intent;
import android.os.Handler;
import android.support.annotation.NonNull;
import android.os.Bundle;
import android.util.Log;
import android.view.View;
import android.widget.Toast;
import com.ctrlplusz.anytextview.AnyEditTextView;
import com.google.android.gms.common.GooglePlayServicesNotAvailableException;
import com.google.android.gms.common.GooglePlayServicesRepairableException;
import com.google.android.gms.tasks.OnCompleteListener;
import com.google.android.gms.tasks.Task;
import com.google.firebase.auth.AuthResult;
import com.google.firebase.auth.FirebaseAuth;
import com.google.firebase.auth.FirebaseUser;
import com.google.firebase.database.DataSnapshot;
import com.google.firebase.database.DatabaseError;
import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;
import com.google.firebase.database.ValueEventListener;
import com.kaopiz.kprogresshud.KProgressHUD;
import com.orhanobut.hawk.Hawk;
public class RegisterActivity extends YourBasicActivity {
private KProgressHUD hud;
FirebaseAuth mAuth;
FirebaseAuth.AuthStateListener mAuthListener;
String key_user = &quot;&quot;;
DatabaseReference mDatabaseRef, mUserCheckData;
@Override
protected void onCreate(Bundle savedInstanceState) {
super.onCreate(savedInstanceState);
setContentView(R.layout.activity_register);
hud = KProgressHUD.create(this)
.setStyle(KProgressHUD.Style.SPIN_INDETERMINATE);
simulateProgressUpdate();
mDatabaseRef = FirebaseDatabase.getInstance().getReference();
mUserCheckData =
FirebaseDatabase.getInstance().getReference().child(&quot;Users&quot;);
mAuth = FirebaseAuth.getInstance();
mAuthListener = new FirebaseAuth.AuthStateListener() {
@Override
public void onAuthStateChanged(FirebaseAuth firebaseAuth) {
FirebaseUser user = firebaseAuth.getCurrentUser();
if (user != null) {
final String emailForVer = user.getEmail();
mUserCheckData.addValueEventListener(new
ValueEventListener() {
@Override
public void onDataChange(DataSnapshot dataSnapshot)
```

```
{
}
@Override
public void onCancelled(DatabaseError databaseError)
{
}
});
} else {
}
}
};
findViewById(R.id.btn_register).setOnClickListener(new
View.OnClickListener() {
@Override
public void onClick(View v) {
final String email = ((AnyEditTextView)
findViewById(R.id.et_email)).getText().toString();
final String phone = ((AnyEditTextView)
findViewById(R.id.et_phone)).getText().toString();
final String address = ((AnyEditTextView)
findViewById(R.id.address)).getText().toString();
final String password = ((AnyEditTextView)
findViewById(R.id.et_password)).getText().toString();
String cpassword = ((AnyEditTextView)
findViewById(R.id.et_c_password)).getText().toString();
if (email.isEmpty() || password.isEmpty() ||
cpassword.isEmpty() || phone.isEmpty() || address.isEmpty()) {
Toast.makeText(RegisterActivity.this, &quot;All fields are
required!&quot;, Toast.LENGTH_SHORT).show();
} else if (!password.equals(cpassword)) {
Toast.makeText(RegisterActivity.this, &quot;Password and
password confirmation do not match&quot;, Toast.LENGTH_SHORT).show();
} else {
simulateProgressUpdate();
hud.show();
mAuth.createUserWithEmailAndPassword(email,
password).addOnCompleteListener(RegisterActivity.this,new
OnCompleteListener&lt;AuthResult&gt;() {
@Override
public void onComplete(@NonNull Task&lt;AuthResult&gt;
task) {
System.out.println(&quot;print
ol&quot;+task.isSuccessful());
if (task.isSuccessful()) {
hud.dismiss();
FirebaseUser current_user =
FirebaseAuth.getInstance().getCurrentUser();
String uid = current_user.getUid();
DatabaseReference mChildDatabase =
mDatabaseRef.child(&quot;Users&quot;).child(uid);
key_user = mChildDatabase.getKey();
Hawk.put(&quot;Email&quot;,email);
mChildDatabase.child(&quot;Email&quot;).setValue(email);
mChildDatabase.child(&quot;Phone&quot;).setValue(phone);
mChildDatabase.child(&quot;Password&quot;).setValue(password);
mChildDatabase.child(&quot;address&quot;).setValue(address);
mChildDatabase.child(&quot;UID&quot;).setValue(uid);
Intent intent = new
Intent(RegisterActivity.this, SecretQuesActivity.class);
intent.putExtra(&quot;uid&quot;, uid);
startActivity(intent);
} else {
hud.dismiss();
Toast.makeText(RegisterActivity.this,
&quot;Failed to create User Account&quot;, Toast.LENGTH_LONG).show();
}
}
});
}
}
});
findViewById(R.id.btn_back).setOnClickListener(new
View.OnClickListener() {
@Override
public void onClick(View v) {
Intent intent = new Intent(RegisterActivity.this,
WelcomeActivity.class);
startActivity(intent);
}
});
}
private void simulateProgressUpdate() {
hud.setMaxProgress(100);
final Handler handler = new Handler();
handler.postDelayed(new Runnable() {
int currentProgress;
@Override
public void run() {
currentProgress += 1;
hud.setProgress(currentProgress);
if (currentProgress == 80) {
}

if (currentProgress &lt; 100) {
handler.postDelayed(this, 50);
}
}
}, 100);
}
}
```

Code of main interface:

```
package com.universety.projectsecret;
import android.content.Intent;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
public class WelcomeActivity extends AppCompatActivity {
@Override
protected void onCreate(Bundle savedInstanceState) {
super.onCreate(savedInstanceState);
setContentView(R.layout.activity_weclome);
findViewById(R.id.btn_login).setOnClickListener(new
View.OnClickListener() {
@Override
public void onClick(View v) {
startActivity(new Intent(WelcomeActivity.this ,
LoginActivity.class));
}
});
findViewById(R.id.btn_reg).setOnClickListener(new
View.OnClickListener() {
@Override
public void onClick(View v) {
startActivity(new Intent(WelcomeActivity.this ,
RegisterActivity.class));
}
});
}
}
```