The second semester of 2020-2021

Computer science and technology in grade 2018

Android Development Technology

Final project examination
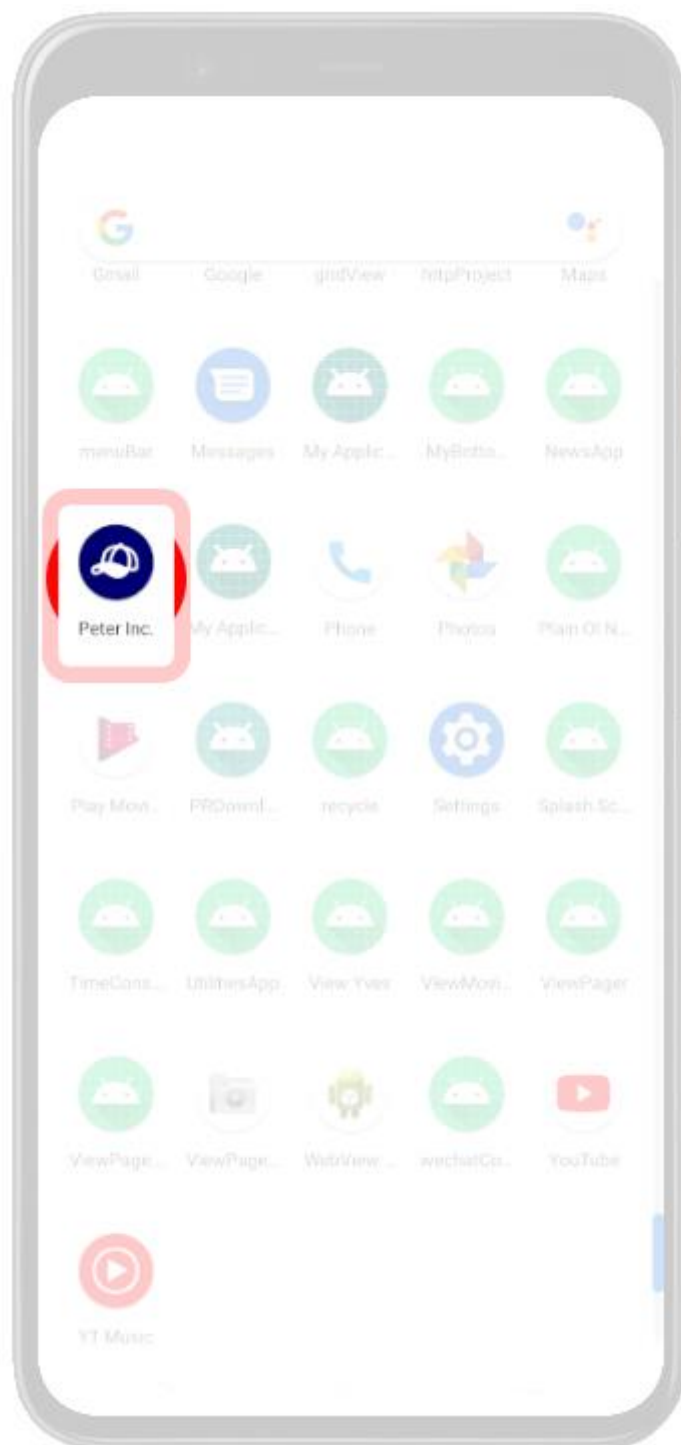
Passport number: EB264654

Full name: APEDO KOMI ETSE YVON

Chinese name: 皮特

1. Project description

1.1 Project purpose (why do you do this project)

My Android project is a "PETER inc." application that is purposed to link people all over the world. It's a SOCIAL MEDIA app. That allows users to create POSTS and invite new Friends to react to them.

First, you need to sign up(register) using your email and a password. Then you can login using your EMAIL and the PASSWORD you just created. If it's correct, Login will be successful.

Second, you can modify your detail in the About page by adding your profile picture, your name, your phone number and your gender. Your email cannot be modified because it's used as your identifier.

Third, you can also create new posts for other users to see it. You can see all the most popular posts all over the world.
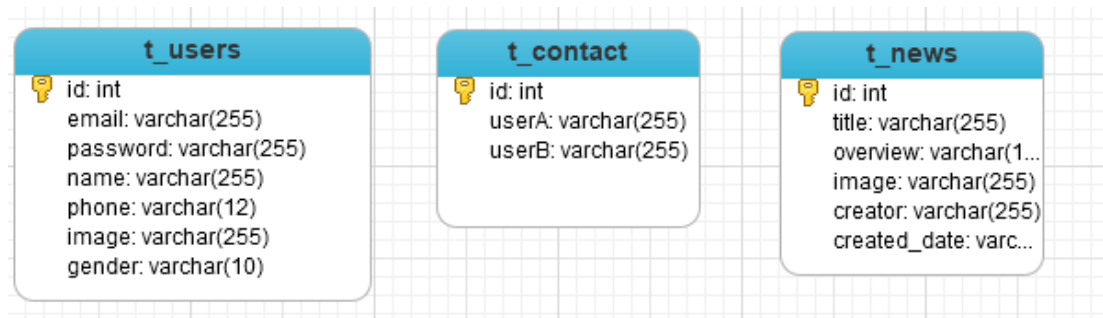
In addition, you can add your friends by their EMAIL address to your contacts. If the email is not correct nothing will be added. If the email correct then your friend's details will appear in your contacts.

1.2 Introduction to technology used

(1)MYSQL DATABASE:
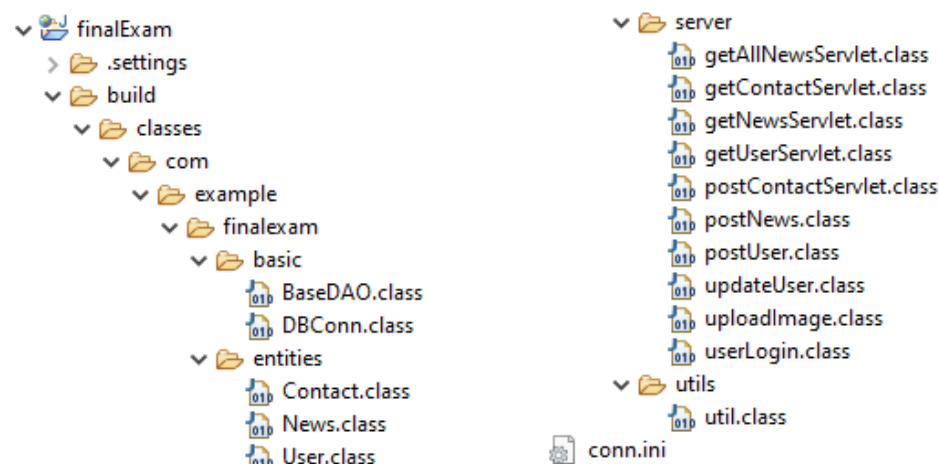
The database name is : **finalExamdb**.

It contains 3 tables: *t_users, t_contact, t_news*

t_users
- id: int
- email: varchar(255)
- password: varchar(255)
- name: varchar(255)
- phone: varchar(12)
- image: varchar(255)
- gender: varchar(10)

t_contact
- id: int
- userA: varchar(255)
- userB: varchar(255)

t_news
- id: int
- title: varchar(255)
- overview: varchar(1...
- image: varchar(255)
- creator: varchar(255)
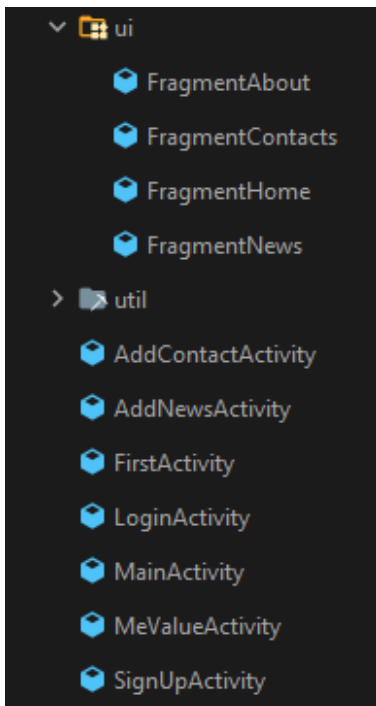- created_date: varc...

(2)JAVA EE(Java sever):

The name of the Java application is *finalExam*. It's a web app.

- It contains 3 ENTITIES each corresponding to the tables in the database.
  Contact, News, User.

- It contains 10 SERVLETS, each corresponding to the activities and fragments.
  *getAllNewsServlet, getContactServlet, getNewsServlet, getUserServlet
  userLogin, postNewst, postContactServlet, postUser, updateUser
  uploadImage*

```
v 🔧 finalExam
  > 📂 .settings
  v 📂 build
    v 📂 classes
      v 📂 com
        v 📂 example
          v 📂 finalexam
            v 📂 basic
                🔹 BaseDAO.class
                🔹 DBConn.class
            v 📂 entities
                🔹 Contact.class
                🔹 News.class
                🔹 User.class
```

```
v 📂 server
    🔹 getAllNewsServlet.class
    🔹 getContactServlet.class
    🔹 getNewsServlet.class
    🔹 getUserServlet.class
    🔹 postContactServlet.class
    🔹 postNews.class
    🔹 postUser.class
    🔹 updateUser.class
    🔹 uploadImage.class
    🔹 userLogin.class
  v 📂 utils
    🔹 util.class
  📄 conn.ini
```
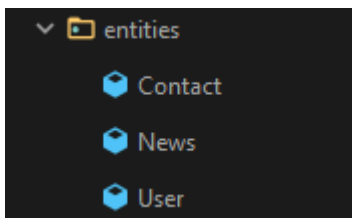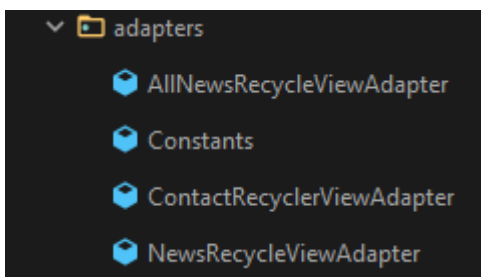
(3) ANDROID STUDIO

In Android studio, I created **7 ACTIVITIES** and **4 FRAGMENTS** making it **all 10 UIs** with the **MainActivity** hosting the 4 fragments.

I created **3 entities**, each corresponding to the tables in the database.

I also created **3 adapters**, each corresponding to the *RecyclerView* in the project.

NB: the Constant class contains the SERVER_LINK Url and the IP ADDRESS. Just edit that class and all the other settings will be affected.

CODING

-Usage of thread and processing technology

```java
private Handler handler=handleMessage(msg) → {
    if(msg.what==GET_JSON_SUCCESS){
        String json =msg.obj.toString();
        List<News> newsList= parsingJsonToList(json);
        AllNewsRecycleViewAdapter  adapter= new AllNewsRecycleViewAdapter(getContext(), newsList);
        Log.i( tag: "MSG", msg.obj.toString());
        LinearLayoutManager manager= new LinearLayoutManager(getContext(), RecyclerView.VERTICAL,  reverseLayout: false);
        rv_all.setLayoutManager(manager);
        rv_all.setAdapter(adapter);
    }
};
```

```java
private void GetNewsThread(){
    Runnable runnable=new Runnable() {
        @Override
        public void run() {
            try {
                String json= HttpClientUtils.HttpClientGet( getUrl: url+"/getAllNewsServlet",  map: null);
                if(!json.equals("")){
                    Message message=new Message();
                    message.what=GET_JSON_SUCCESS;
                    message.obj=json;
                    handler.sendMessage(message);
                }
            } catch (Exception e) {
                e.printStackTrace();
            } }
    };
    Thread thread=new Thread(runnable);
    thread.start();
}
```

-Usage of Http network communication technology

```java
private void GetNewsThread(){
    Runnable runnable=new Runnable() {
        @Override
        public void run() {
            try {
                String json= HttpClientUtils.HttpClientGet( getUrl: url+"/getNewsServlet",  map: null);
                if(!json.equals("")){
                    Message message=new Message();
                    message.what=GET_JSON_SUCCESS;
                    message.obj=json;
                    handler.sendMessage(message);
                }
            } catch (Exception e) {
                e.printStackTrace();
            }
        }
    };
    Thread thread=new Thread(runnable);
    thread.start();
}
```

-usage of GridLayoutManager

```
        NewsRecycleViewAdapter adapter= new NewsRecycleViewAdapter(getContext(), newsList);
        Log.i( tag: "MSG", msg.obj.toString());
//      LinearLayoutManager manager= new LinearLayoutManager(getContext(), RecyclerView.VERTICAL, false);
        GridLayoutManager manager1=new GridLayoutManager(getContext(),    spanCount: 2);
        rv_news.setLayoutManager(manager1);

        rv_news.setAdapter(adapter);
```
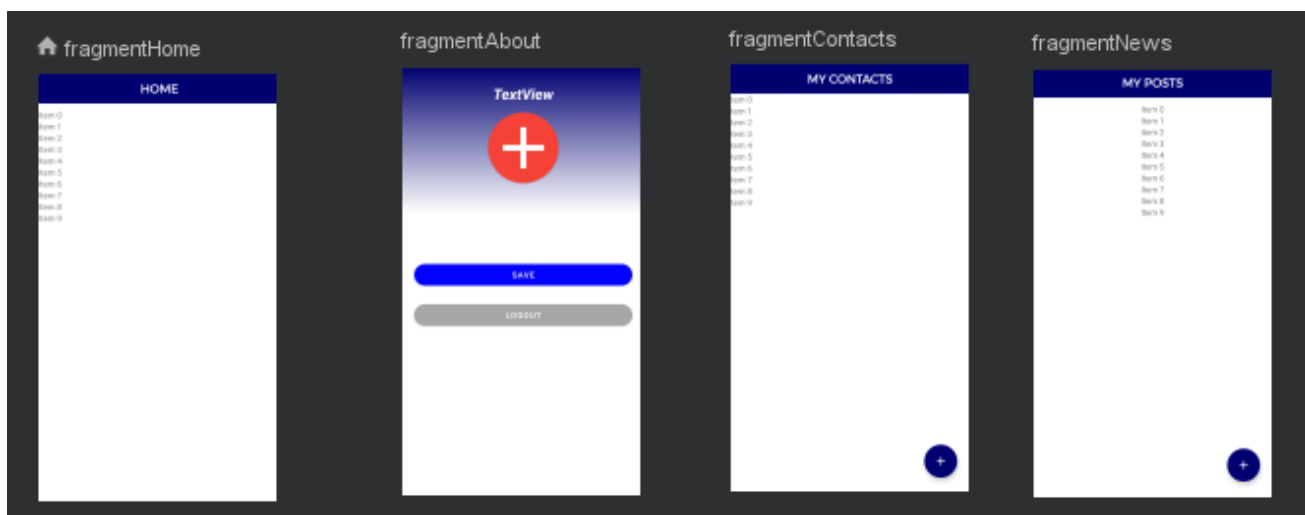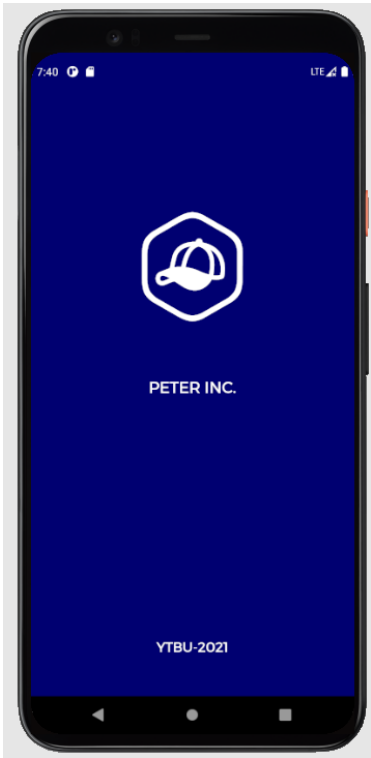
-usage of LinearLayoutManager

```
        ContactRecyclerViewAdapter adapter= new ContactRecyclerViewAdapter(getContext(), contactList);
        Log.i( tag: "MSG", msg.obj.toString());
        LinearLayoutManager manager= new LinearLayoutManager(getContext(), RecyclerView.VERTICAL,  reverseLayout: false);
        rv_contact.setLayoutManager(manager);
        rv_contact.setAdapter(adapter);
```
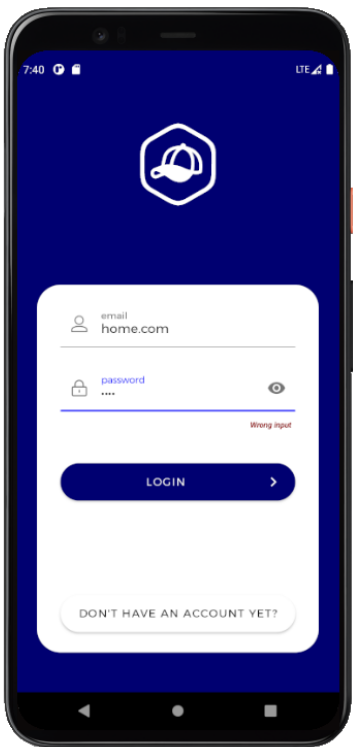
-navigation for the fragment
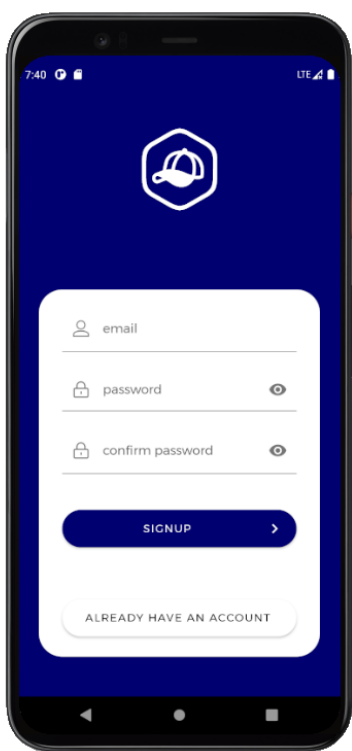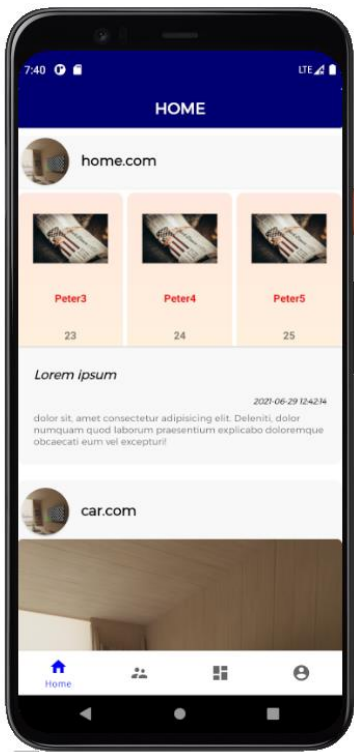
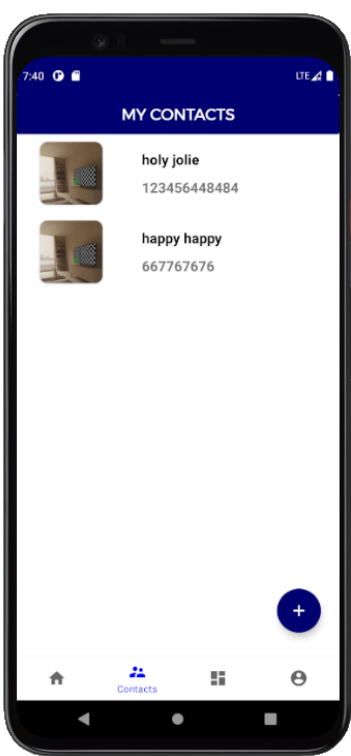**2. Screenshot of project running UI:**

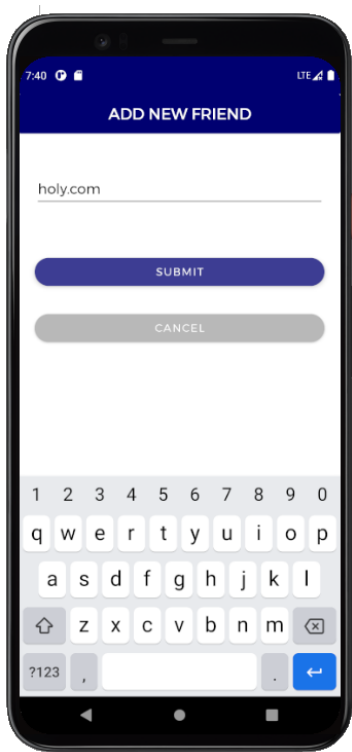| WELCOME UI	LOGIN UI	REGISTER UI

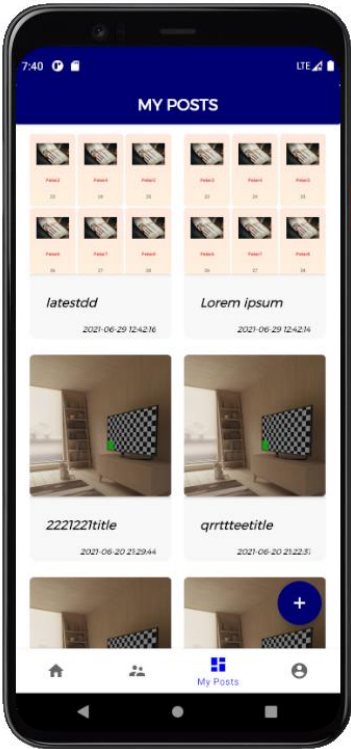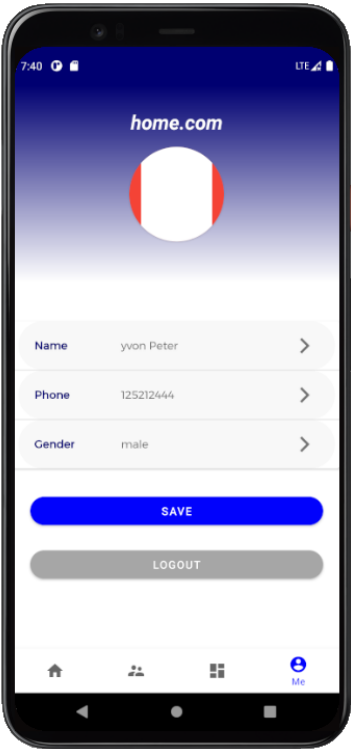| HOMEPAGE UI : ALL POSTS	CONTACTS UI :ALL CONTACTS	ADD NEW FRIEND UI

MY POSTS UI :

ADD NEW POST UI

UPDATE DETAILS UI

INPUT DETAULS UI