FINAL PROJECT

CS426 – Mobile Device Application Development

Group members :

- Do Le Duan , 18125127
- Bui Xuan Vinh , 18125119
- Cao Ngoc Son , 18125111

Description :

For people who have demand in finding house chores helper in a short amount of time , we make a mobile application “Find My Helper” . This platform will allow everyone to search helper with necessary information nearby through a google maps by using an account which can be easily created . Concurrently , the customers could assess the helpers by stars from 1 – 5 , those assessment will be stored and the others could rely on to pick up a suitable helper in the most convenient way

Novelty :

It ‘s inconvenient for people who are always busy to come to the job centers and find proper house chores helper . With this app , we simply choose right helper in anywhere at anytime based on real-time assessment compared with helpers from centers that we have no information about them at all .

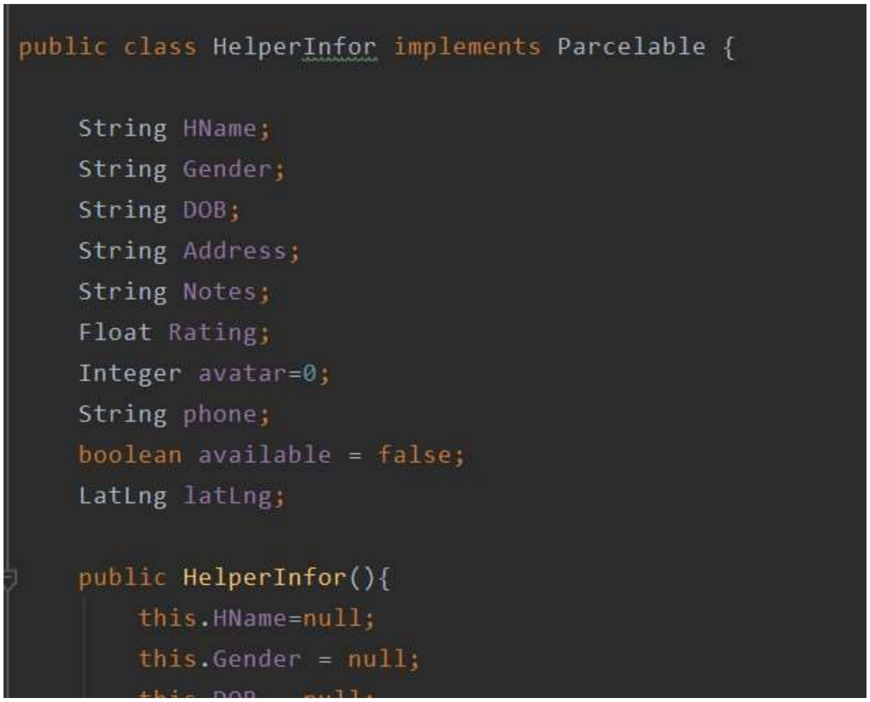
Moreover , this platform will not only create customers but also aim to people who want to have a helper job . With smart device , they could easily create profile for them online without any fees

Major functions , Techniques & Implementation

Our app consists of 11 classes in Java language

- CurrentHelperHired
- CustormerHire
- HelperInfor
- HelperProfile
- Login
- Maps
- Signin
- Favorite List
- Database
- Register
- Splash Screen

1. HelperInfo

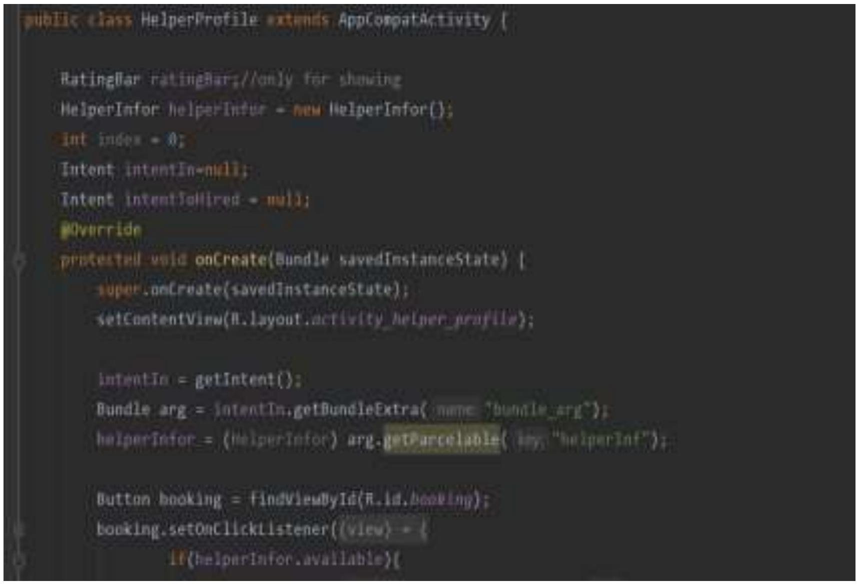


```
public class HelperInfor implements Parcelable {

    String HName;
    String Gender;
    String DOB;
    String Address;
    String Notes;
    Float Rating;
    Integer avatar=0;
    String phone;
    boolean available = false;
    LatLng latLng;

    public HelperInfor(){
        this.HName=null;
        this.Gender = null;
        this.DOB = null;
```

In this class, just declare attributes that a helper can have (as we assume). And have several getter and setter which can help during the project.

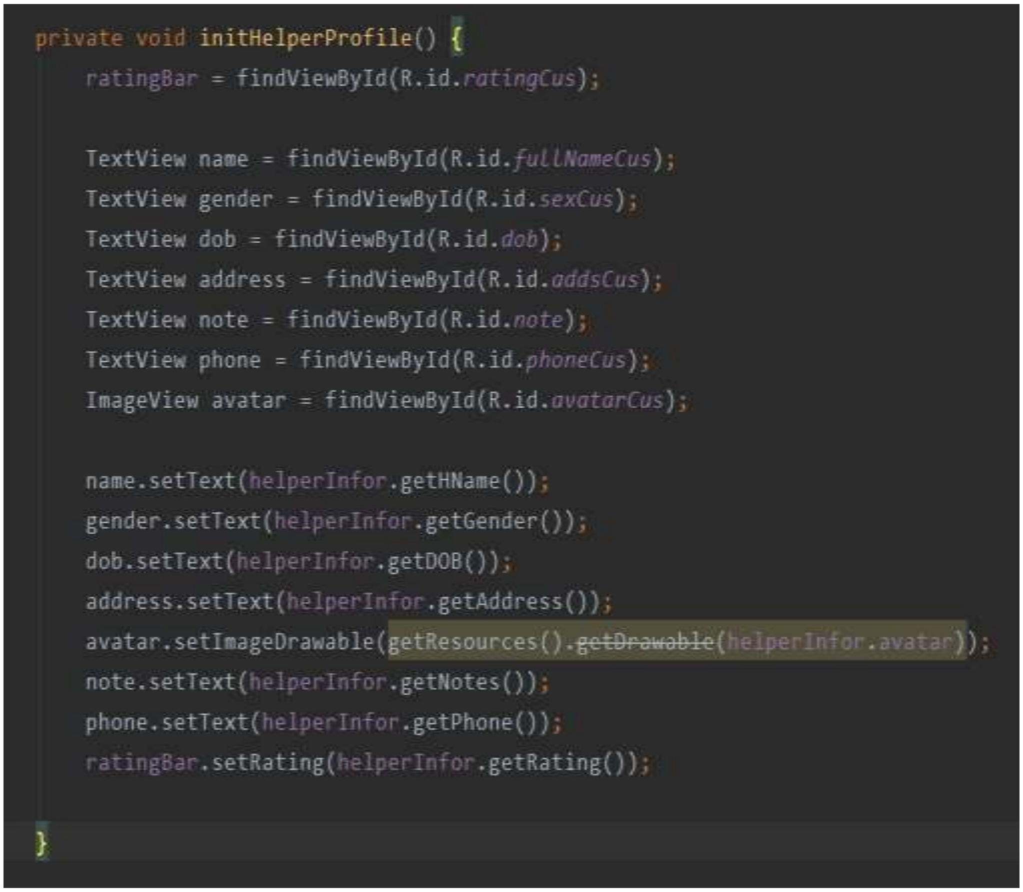
The helperinfor in this project we use the parcelable instead of the serilizable thanks to the faster and more easier to transmis the data in each activities of its. We write each information of helper into file to read later.

2. HelperProfile

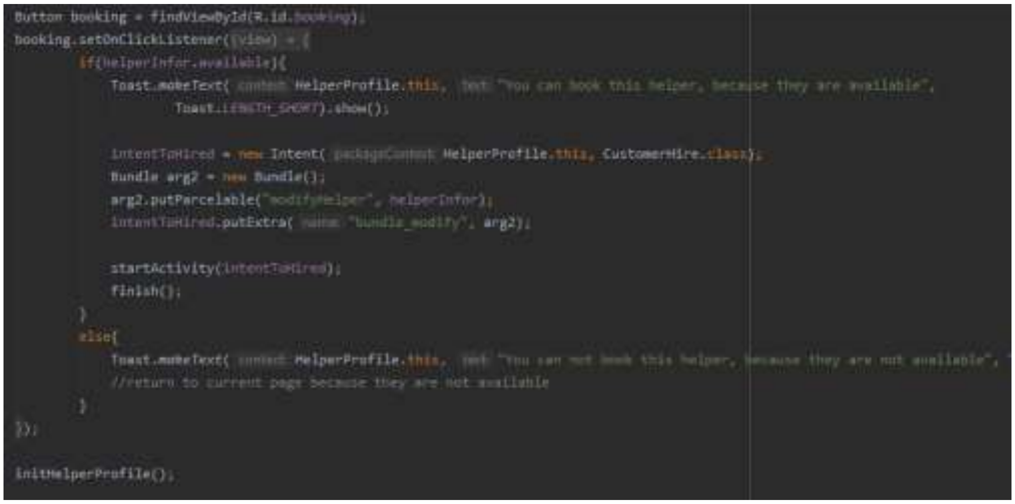


```
public class HelperProfile extends AppCompatActivity {

    RatingBar ratingBar;//only for showing
    HelperInfor helperInfor = new HelperInfor();
    int index = 0;
    Intent intentIn=null;
    Intent intentToHired = null;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_helper_profile);

        intentIn = getIntent();
        Bundle arg = intentIn.getBundleExtra( name: "bundle_arg");
        helperInfor = (HelperInfor) arg.getParcelable( key: "helperInf");

        Button booking = findViewById(R.id.booking);
        booking.setOnClickListener((view) -> {
                if(helperInfor.available){
```

This activity is used to represent profile of specific Helper when they are clicked on map.
Main function here is initHelperProfile();
Button Booking can be activated when this Helper is available for help other Customers. If not, you will get an announcement for that situation



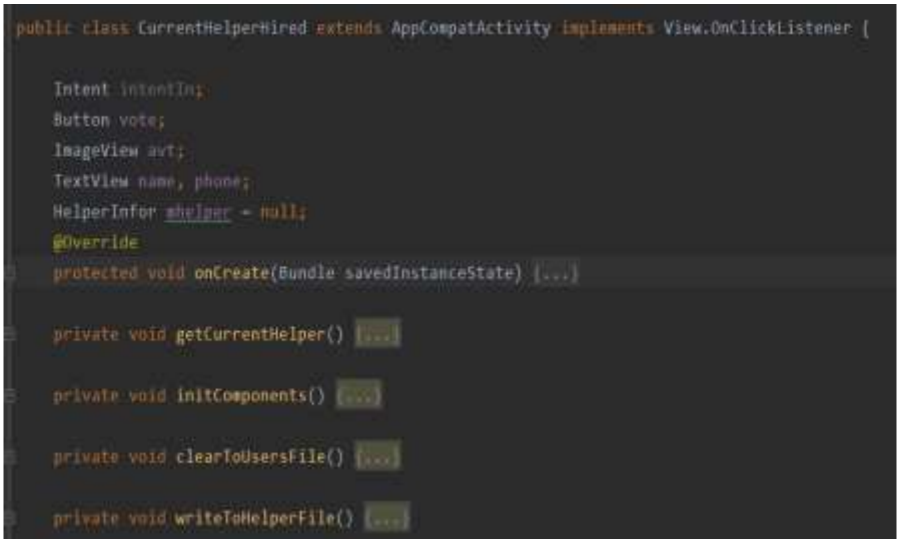
```
private void initHelperProfile() {
    ratingBar = findViewById(R.id.ratingCus);

    TextView name = findViewById(R.id.fullNameCus);
    TextView gender = findViewById(R.id.sexCus);
    TextView dob = findViewById(R.id.dob);
    TextView address = findViewById(R.id.addsCus);
    TextView note = findViewById(R.id.note);
    TextView phone = findViewById(R.id.phoneCus);
    ImageView avatar = findViewById(R.id.avatarCus);

    name.setText(helperInfor.getHName());
    gender.setText(helperInfor.getGender());
    dob.setText(helperInfor.getDOB());
    address.setText(helperInfor.getAddress());
    avatar.setImageDrawable(getResources().~~getDrawable~~(helperInfor.avatar));
    note.setText(helperInfor.getNotes());
    phone.setText(helperInfor.getPhone());
    ratingBar.setRating(helperInfor.getRating());

}
```



```
Button booking = findViewById(R.id.booking);
booking.setOnClickListener((view) -> {
        if(helperInfor.available){
            Toast.makeText( context: HelperProfile.this,  text: "You can book this helper, because they are available",
                    Toast.LENGTH_SHORT).show();

            intentToHired = new Intent( packageContext: HelperProfile.this, CustomerHire.class);
            Bundle arg2 = new Bundle();
            arg2.putParcelable("modifyHelper", helperInfor);
            intentToHired.putExtra( name: "bundle_modify", arg2);

            startActivity(intentToHired);
            finish();
        }
        else{
            Toast.makeText( context: HelperProfile.this,  text: "You can not book this helper, because they are not available",
            //return to current page because they are not available
        }
});

initHelperProfile();
```

In the HelperProfile, we init the profile for each helper, that most considerable is the rating bar, and the button book is put under the rating in order to the customer can book this helper. The major function in this is the booking onclicklistener. If someone has booked this helper, we will set the avaiable of him/her to be false, so that, others customer cannot book this helper.(Toast that the customer cannot book if someone has booked by the following code)

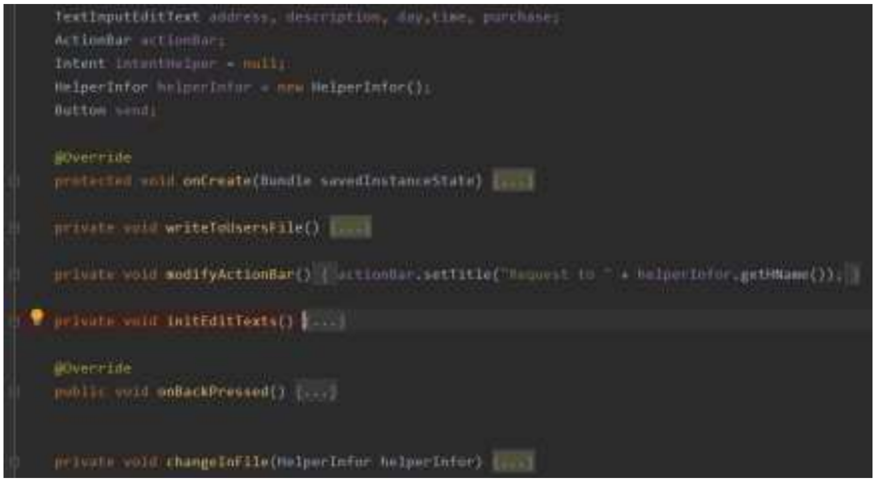
3. CurrentHelperHired:



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

    Intent intentIn;
    Button vote;
    ImageView avt;
    TextView name, phone;
    HelperInfor mhelper = null;
    @Override
    protected void onCreate(Bundle savedInstanceState) {...}

    private void getCurrentHelper() {...}

    private void initComponents() {...}

    private void clearToUsersFile() {...}

    private void writeToHelperFile() {...}
```

This activity to show the current helper that user hired.
Main functions is to write/read helper infor and write to users file.

4. CustomerHire



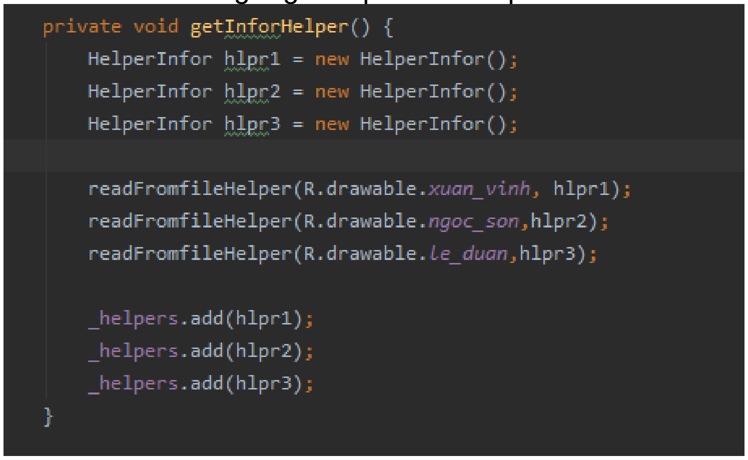
```
    TextInputEditText address, description, day,time, purchase;
    ActionBar actionBar;
    Intent intentHelper = null;
    HelperInfor helperInfor = new HelperInfor();
    Button send;

    @Override
    protected void onCreate(Bundle savedInstanceState) {...}

    private void writeToUsersFile() {...}

    private void modifyActionBar() { actionBar.setTitle("Request to " + helperInfor.getHName()); }

    private void initEditTexts() {...}

    @Override
    public void onBackPressed() {...}


    private void changeInFile(HelperInfor helperInfor) {...}
```

This is show the form that a customer must to fill all the necessary information which will be sent to the selected helper such as (customer’s address, time to come, day to come and salary that will be paid to helper)

Main functions : writeToUsersFile, changeInFile. Because, whenever the request is sent, that helper can not be hired by another customer.

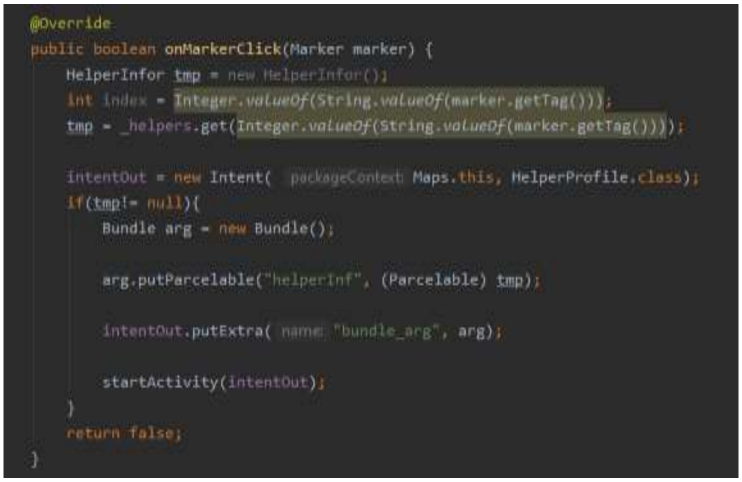
5. Maps

We use API from google to provide map for the customer



```java
private void getInforHelper() {
    HelperInfor hlpr1 = new HelperInfor();
    HelperInfor hlpr2 = new HelperInfor();
    HelperInfor hlpr3 = new HelperInfor();

    readFromfileHelper(R.drawable.xuan_vinh, hlpr1);
    readFromfileHelper(R.drawable.ngoc_son,hlpr2);
    readFromfileHelper(R.drawable.le_duan,hlpr3);

    _helpers.add(hlpr1);
    _helpers.add(hlpr2);
    _helpers.add(hlpr3);
}
```

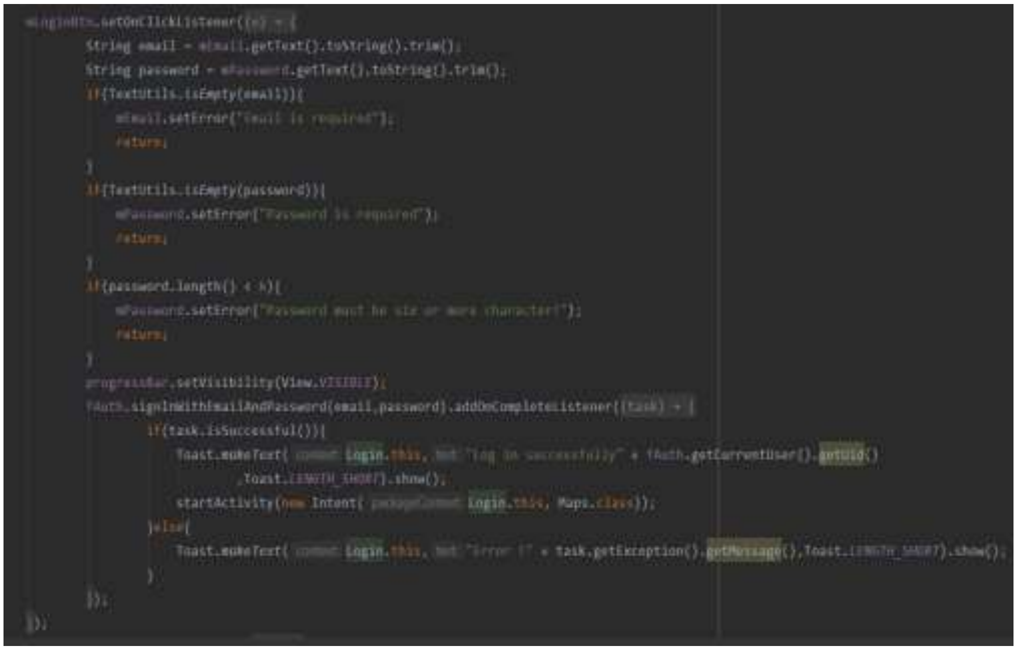
At first, this begins with login layout where customers can login by their account they already created, or they can create in the rest case. Continue, we go to the maps activity where we can choose helper by clicking on the marker by the functions:



```java
@Override
public boolean onMarkerClick(Marker marker) {
    HelperInfor tmp = new HelperInfor();
    int index = Integer.valueOf(String.valueOf(marker.getTag()));
    tmp = _helpers.get(Integer.valueOf(String.valueOf(marker.getTag())));

    intentOut = new Intent( packageContext: Maps.this, HelperProfile.class);
    if(tmp!= null){
        Bundle arg = new Bundle();

        arg.putParcelable("helperInf", (Parcelable) tmp);

        intentOut.putExtra( name: "bundle_arg", arg);

        startActivity(intentOut);
    }
    return false;
}
```

The helper we create by the transmission data into the file_name.txt with the filename equal to the value of their avatarID thanks to function readFromfileHelper. Also by using the address of the helper provided, we convert it into Latlng by function ConvertAddtoLatlng to get position of marker on the maps. Furthermore, if we click on the marker (represent for the helper) the intent will be send to the HelperProfile to create the profile of helper which include the image, phone, address,gender of the helper.
We insert 1 more item in the menu is the Favorite list, which include : Log out, History, Favrite List.

6. Login



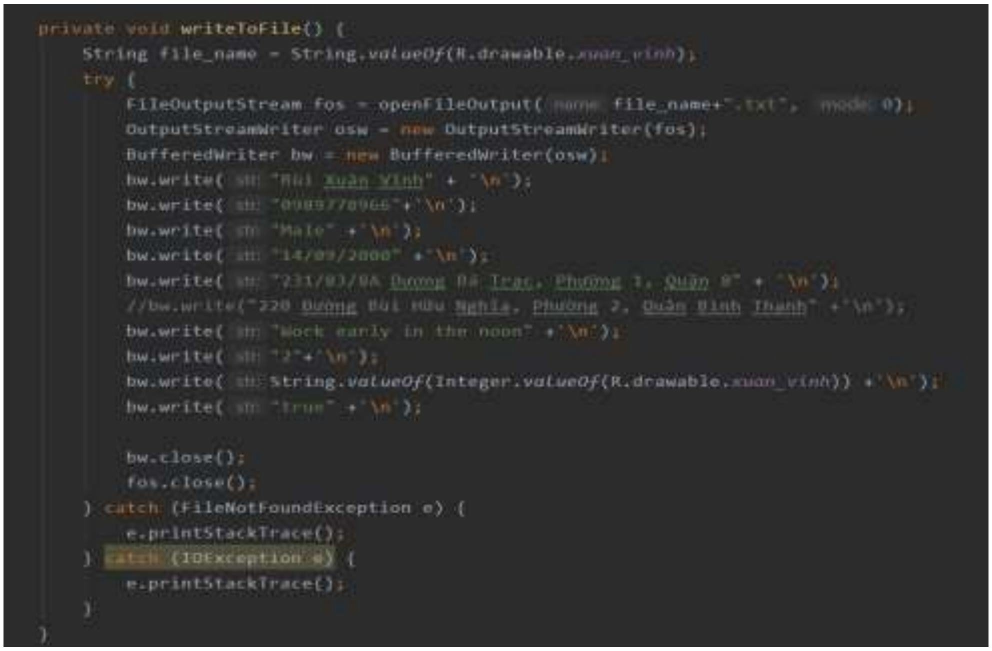
```java
mSignInBtn.setOnClickListener((v) -> {
        String email = mEmail.getText().toString().trim();
        String password = mPassword.getText().toString().trim();
        if(TextUtils.isEmpty(email)){
            mEmail.setError("Email is required");
            return;
        }
        if(TextUtils.isEmpty(password)){
            mPassword.setError("Password is required");
            return;
        }
        if(password.length() < 6){
            mPassword.setError("Password must be six or more character!");
            return;
        }
        progressBar.setVisibility(View.VISIBLE);
        fAuth.signInWithEmailAndPassword(email,password).addOnCompleteListener((task) -> {
                if(task.isSuccessful()){
                    Toast.makeText( context: Login.this, text: "Log in successfully" + fAuth.getCurrentUser().getUid()
                            ,Toast.LENGTH_SHORT).show();
                    startActivity(new Intent( packageContext: Login.this, Maps.class));
                }else{
                    Toast.makeText( context: Login.this, text: "Error !" + task.getException().getMessage(),Toast.LENGTH_SHORT).show();
                }
        });
});
```

We created Login interface to allow people to use account and password to login app

We connect Firebase to store the customer account and password

This function is to test the format of the account (we use email format) the length of password (must >6 characters) and the appearance of password

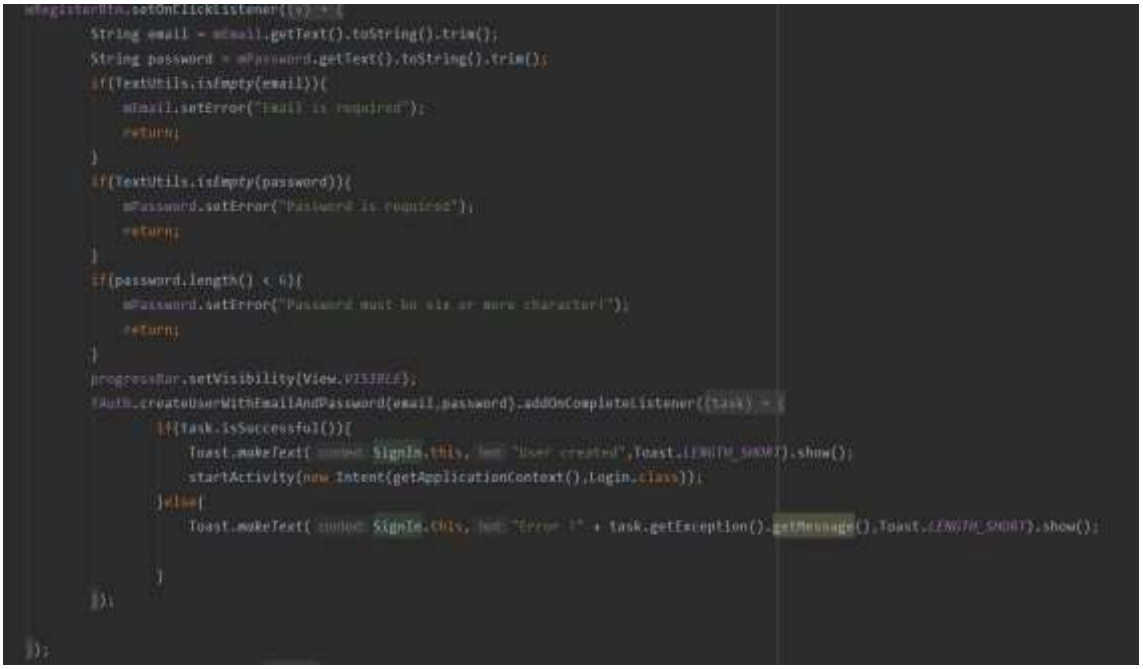
If login successfully , it will jump to Maps class else it will announce error



```java
private void writeToFile() {
    String file_name = String.valueOf(R.drawable.xuan_vinh);
    try {
        FileOutputStream fos = openFileOutput( name: file_name+".txt",  mode: 0);
        OutputStreamWriter osw = new OutputStreamWriter(fos);
        BufferedWriter bw = new BufferedWriter(osw);
        bw.write( str: "Bùi Xuân Vinh" + '\n');
        bw.write( str: "0989778966"+'\n');
        bw.write( str: "Male" +'\n');
        bw.write( str: "14/09/2000" +'\n');
        bw.write( str: "231/83/8A Đường Bà Trạc, Phường 1, Quận 8" + '\n');
        //bw.write("220 Đường Bùi Hữu Nghĩa, Phường 2, Quận Bình Thạnh" +'\n');
        bw.write( str: "Wock early in the noon" +'\n');
        bw.write( str: "2"+'\n');
        bw.write( str: String.valueOf(Integer.valueOf(R.drawable.xuan_vinh)) +'\n');
        bw.write( str: "true" +'\n');

        bw.close();
        fos.close();
    } catch (FileNotFoundException e) {
        e.printStackTrace();
    } catch (IOException e) {
        e.printStackTrace();
    }
}
```

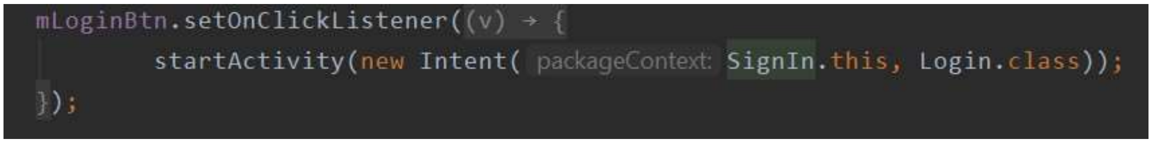
This function will provide information for the helper who will appear on the maps

7. Signin (Register account)



```
mRegisterBtn.setOnClickListener((v) → {
        String email = mEmail.getText().toString().trim();
        String password = mPassword.getText().toString().trim();
        if(TextUtils.isEmpty(email)){
            mEmail.setError("Email is required");
            return;
        }
        if(TextUtils.isEmpty(password)){
            mPassword.setError("Password is required");
            return;
        }
        if(password.length() < 6){
            mPassword.setError("Password must be six or more character!");
            return;
        }
        progressBar.setVisibility(View.VISIBLE);
        fAuth.createUserWithEmailAndPassword(email,password).addOnCompleteListener((task) → {
                if(task.isSuccessful()){
                    Toast.makeText( context: SignIn.this, text: "User created",Toast.LENGTH_SHORT).show();
                    startActivity(new Intent(getApplicationContext(),Login.class));
                }else{
                    Toast.makeText( context: SignIn.this, text: "Error !" + task.getException().getMessage(),Toast.LENGTH_SHORT).show();

                }
        });

});
```

Like Login class , Register will check the format of Email , Password . If account is registered successfully , it will go back to Login to enter in the Maps



```
mLoginBtn.setOnClickListener((v) → {
        startActivity(new Intent( packageContext: SignIn.this, Login.class));
});
```

If you have already have an account , this will immediately help you go back to login

Note : With Login and Register , we connect with FireBase via google Account

Username : duandole9199@gmail.com
Password : 0131914567

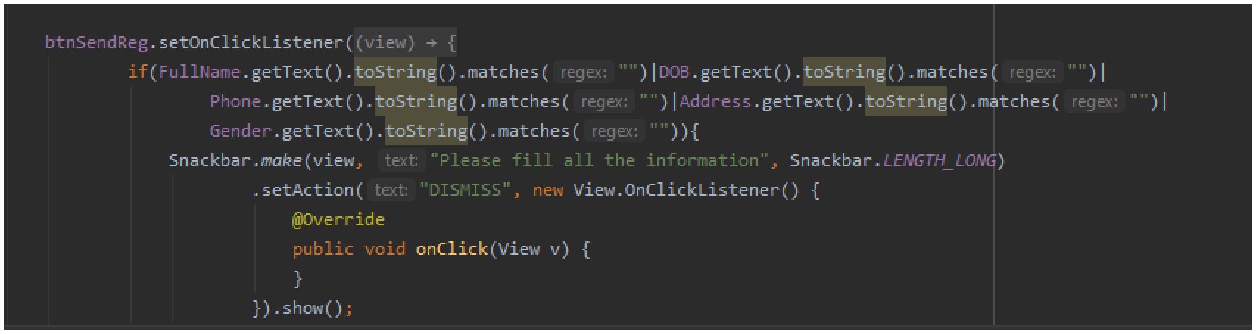
You can use this account to check

8. Register

The user can join to be a Helper by registering through a form, which is needed the basic information of those such as : Avatar, Full Name, Gender, Date of Birth, his/her Address and the last one is the Phone Number.

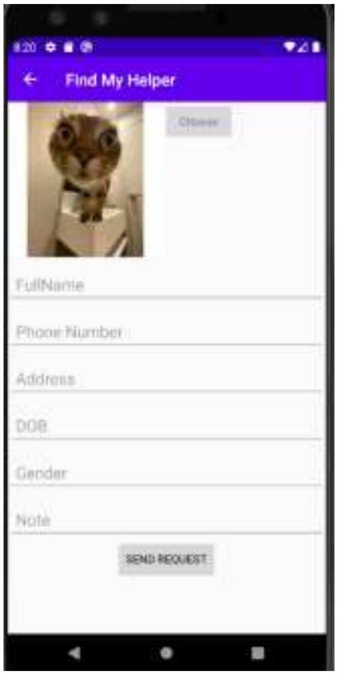
We already create a button for users to choose their avatar to appear in the profile through access their photo gallery.

If the user clicks in the send request button to finish, they must guarantee all information are filled, if not the notification will appear.



```
btnSendReg.setOnClickListener((view) → {
        if(FullName.getText().toString().matches( regex: "")|DOB.getText().toString().matches( regex: "")|
                Phone.getText().toString().matches( regex: "")|Address.getText().toString().matches( regex: "")|
                Gender.getText().toString().matches( regex: "")){
            Snackbar.make(view,  text: "Please fill all the information", Snackbar.LENGTH_LONG)
                    .setAction( text: "DISMISS", new View.OnClickListener() {
                        @Override
                        public void onClick(View v) {
                        }
                    }).show();
```

Then we insert it into database of Helper with elements we have inserted before.

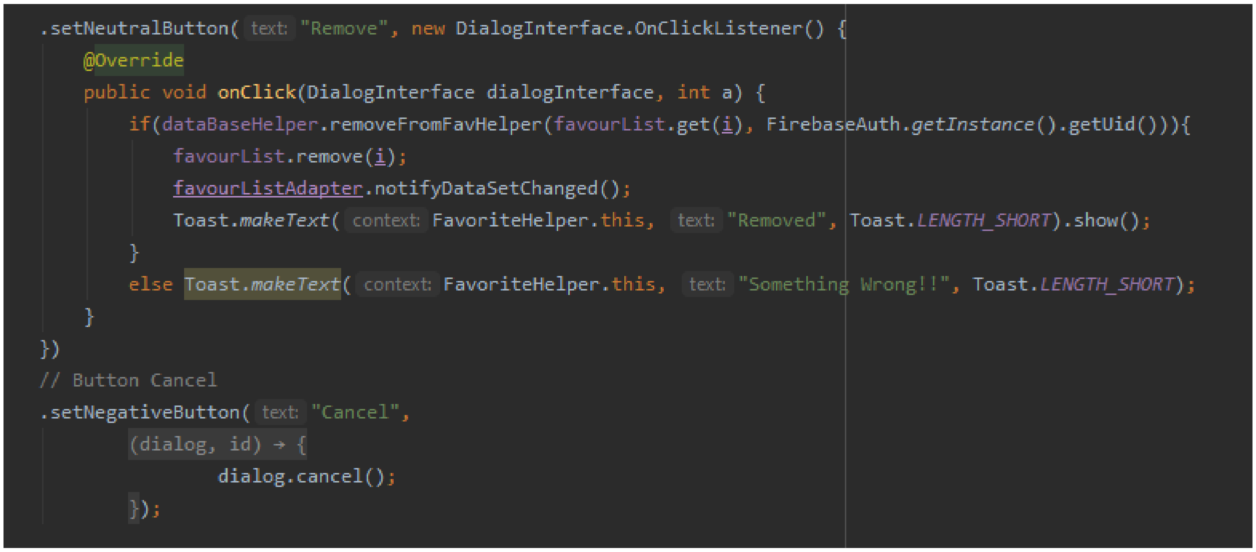


Here is the form for user to register to join in Helper Family.(The avatar is selected in the photo gallery of user)

9. Favorite List

The user can select someone to his/her favor list by clicking add. The information of this person will be saved into list include: his/her Avatar, FullName, Phone number, which is looked like the collapse list of Helper Infor by new adapter.

Beside that, If the user clicks on the item of the list, they can remove these helpers out of list instead of clicking at remove.



```
.setNeutralButton( text: "Remove", new DialogInterface.OnClickListener() {
    @Override
    public void onClick(DialogInterface dialogInterface, int a) {
        if(dataBaseHelper.removeFromFavHelper(favourList.get(i), FirebaseAuth.getInstance().getUid())){
            favourList.remove(i);
            favourListAdapter.notifyDataSetChanged();
            Toast.makeText( context: FavoriteHelper.this,  text: "Removed", Toast.LENGTH_SHORT).show();
        }
        else Toast.makeText( context: FavoriteHelper.this,  text: "Something Wrong!!", Toast.LENGTH_SHORT);
    }
})
// Button Cancel
.setNegativeButton( text: "Cancel",
        (dialog, id) → {
                dialog.cancel();
        });
```

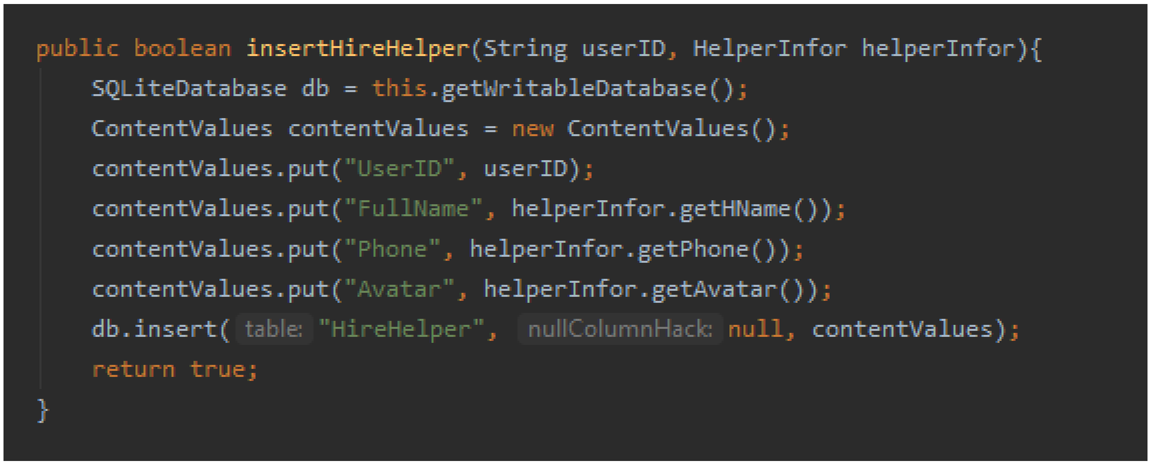
If the favor does not have anything yet, then the notification is: “Don’t have anything to show”

Finally, the data will be updated to the database include: Name, Gender, Phone, Note, Rating, Address and the available for someone who want to book his/her(if not available means someone booked them first, you cannot book them in this period), which is all of information of this helper in order to the user can be easily to manage.

10. Database

At midterm, we use firebase like a database to manage, but it has an disadvantage in real time database, so we consider to use both SQLite and Firebase to manage all of information of helper, and the update in the future.

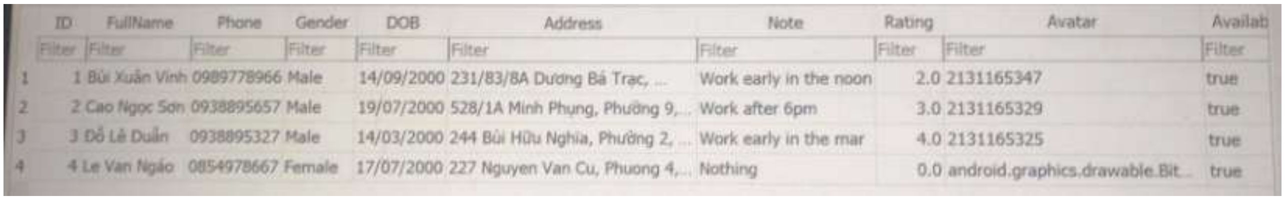
In Database Class, we create 3 tables (HelperInfor, HireHelper, FavHelper) can maintain all information of the user. In example, we can insert Hire Helper through the userID and the information of the helper.



```
public boolean insertHireHelper(String userID, HelperInfor helperInfor){
    SQLiteDatabase db = this.getWritableDatabase();
    ContentValues contentValues = new ContentValues();
    contentValues.put("UserID", userID);
    contentValues.put("FullName", helperInfor.getHName());
    contentValues.put("Phone", helperInfor.getPhone());
    contentValues.put("Avatar", helperInfor.getAvatar());
    db.insert( table: "HireHelper",  nullColumnHack: null, contentValues);
    return true;
}
```

-

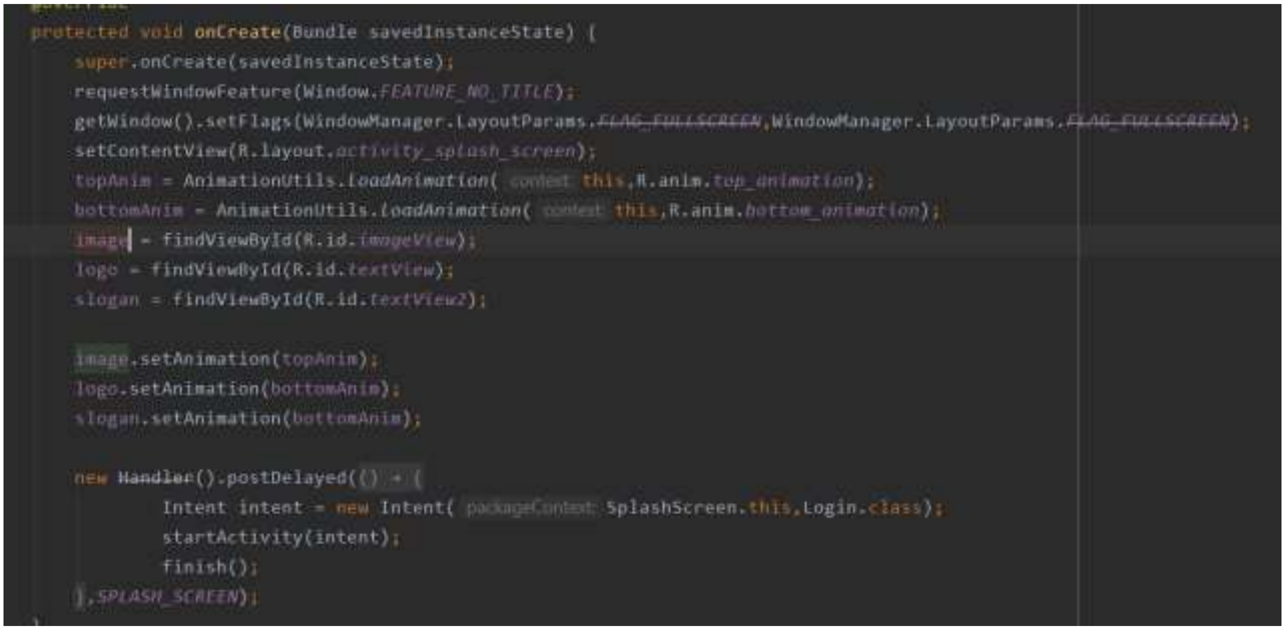
At first, we use the avatarID to distinguish people but later, we decide to use their phone number to manage and easily to control.



	ID	FullName	Phone	Gender	DOB	Address	Note	Rating	Avatar	Availab
	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter
1	1	Bùi Xuân Vinh	0989778966	Male	14/09/2000	231/83/8A Dương Bá Trạc, ...	Work early in the noon	2.0	2131165347	true
2	2	Cao Ngọc Sơn	0938895657	Male	19/07/2000	528/1A Minh Phụng, Phường 9,...	Work after 6pm	3.0	2131165329	true
3	3	Đỗ Lê Duẩn	0938895327	Male	14/03/2000	244 Bùi Hữu Nghĩa, Phường 2, ...	Work early in the mar	4.0	2131165325	true
4	4	Le Van Ngáo	0854978667	Female	17/07/2000	227 Nguyen Van Cu, Phuong 4,...	Nothing	0.0	android.graphics.drawable.Bit...	true

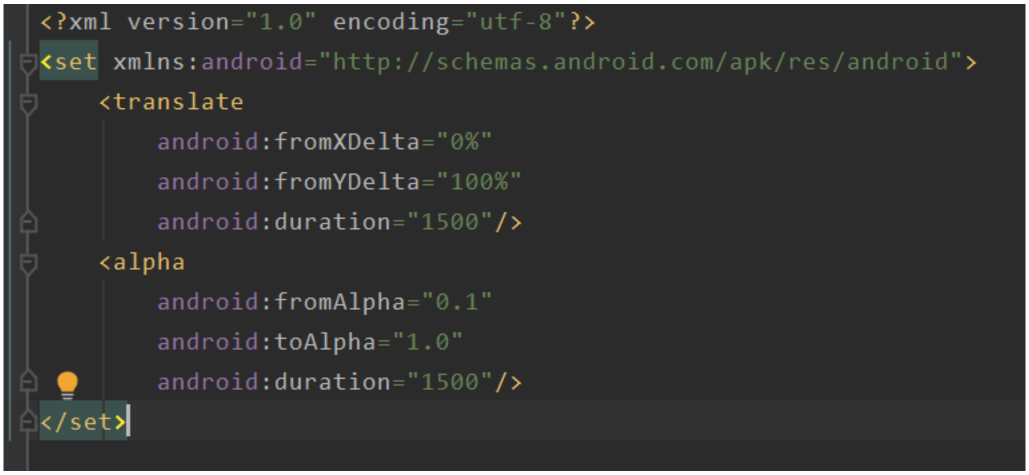
The helper 4th is registered, then the database will add this member into data of helper. Because they just register so the rating default here is 0.

11. Splash Screen



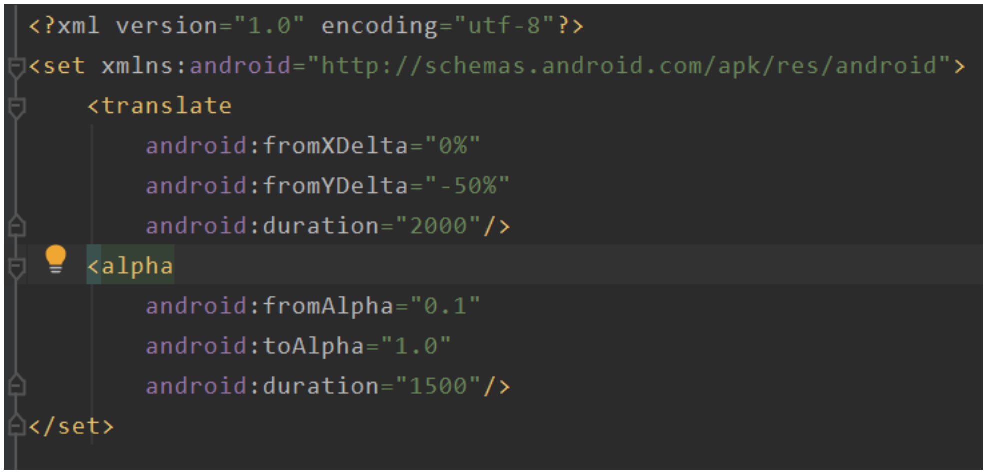
```
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    requestWindowFeature(Window.FEATURE_NO_TITLE);
    getWindow().setFlags(WindowManager.LayoutParams.FLAG_FULLSCREEN,WindowManager.LayoutParams.FLAG_FULLSCREEN);
    setContentView(R.layout.activity_splash_screen);
    topAnim = AnimationUtils.loadAnimation( context: this,R.anim.top_animation);
    bottomAnim = AnimationUtils.loadAnimation( context: this,R.anim.bottom_animation);
    image = findViewById(R.id.imageView);
    logo = findViewById(R.id.textView);
    slogan = findViewById(R.id.textView2);

    image.setAnimation(topAnim);
    logo.setAnimation(bottomAnim);
    slogan.setAnimation(bottomAnim);

    new Handler().postDelayed(() -> {
            Intent intent = new Intent( packageContext: SplashScreen.this,Login.class);
            startActivity(intent);
            finish();
    },SPLASH_SCREEN);
```

We create the splash screen with the idea that 1 image will go down from top to the center of the screen . By constrast , the name of app and the slogan will go up from the bottom



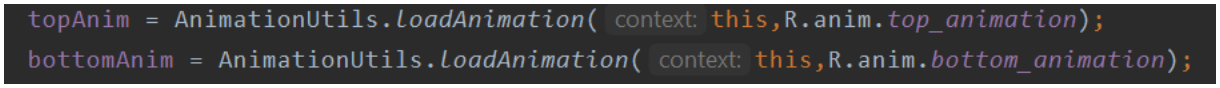
```
<?xml version="1.0" encoding="utf-8"?>
<set xmlns:android="http://schemas.android.com/apk/res/android">
    <translate
        android:fromXDelta="0%"
        android:fromYDelta="100%"
        android:duration="1500"/>
    <alpha
        android:fromAlpha="0.1"
        android:toAlpha="1.0"
        android:duration="1500"/>
</set>
```

This is how sentences will move , we call this is the bottom animation , fromAlpha and toAlpha to make it appear gradually and duration is the time for the animation to happen



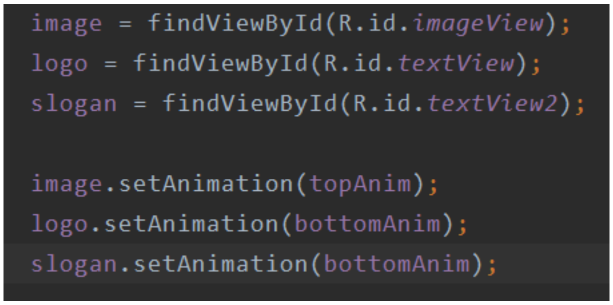
```xml
<?xml version="1.0" encoding="utf-8"?>
<set xmlns:android="http://schemas.android.com/apk/res/android">
    <translate
        android:fromXDelta="0%"
        android:fromYDelta="-50%"
        android:duration="2000"/>
    <alpha
        android:fromAlpha="0.1"
        android:toAlpha="1.0"
        android:duration="1500"/>
</set>
```

This is top animation , XDelta and Ydelta to locate the position that the figure will slide down



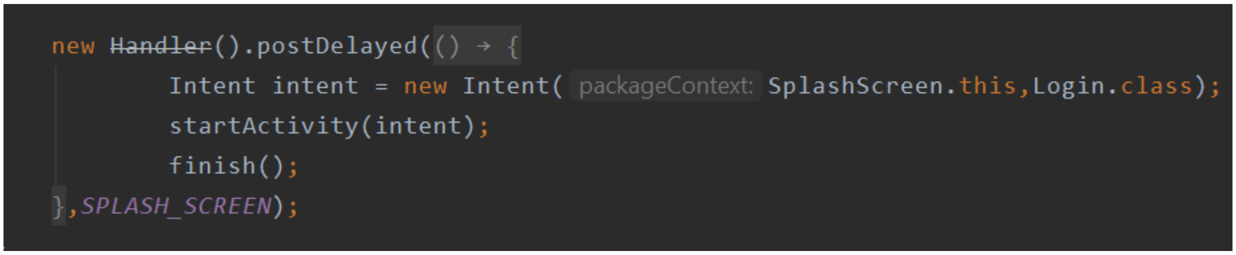
```java
topAnim = AnimationUtils.loadAnimation( context: this,R.anim.top_animation);
bottomAnim = AnimationUtils.loadAnimation( context: this,R.anim.bottom_animation);
```

We use AnimationUtils.LoadAnimation to load animation event



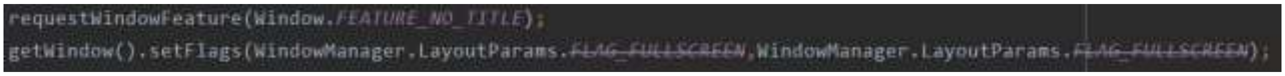
```java
image = findViewById(R.id.imageView);
logo = findViewById(R.id.textView);
slogan = findViewById(R.id.textView2);

image.setAnimation(topAnim);
logo.setAnimation(bottomAnim);
slogan.setAnimation(bottomAnim);
```

Set the animation for pictures and slogan as intention



```java
new Handler().postDelayed(() -> {
        Intent intent = new Intent( packageContext: SplashScreen.this,Login.class);
        startActivity(intent);
        finish();
},SPLASH_SCREEN);
```

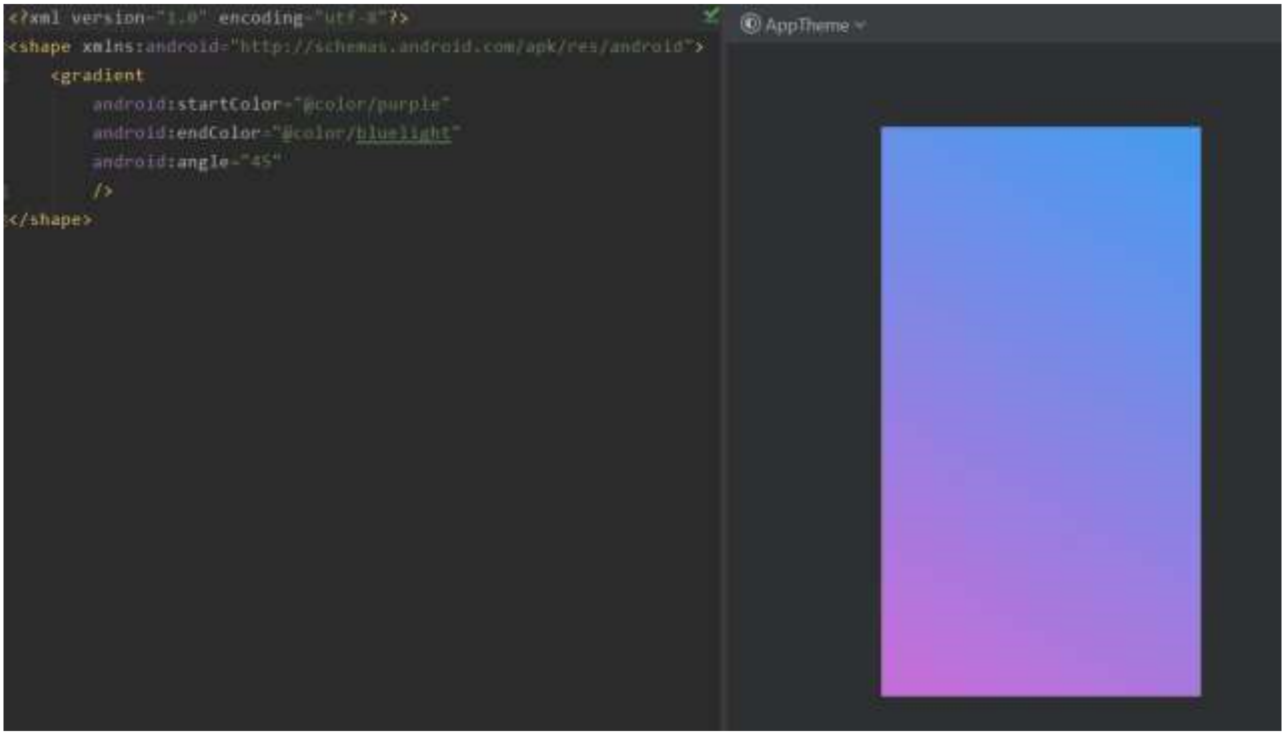
After doing the splash screen , we will go to login side



```java
requestWindowFeature(Window.FEATURE_NO_TITLE);
getWindow().setFlags(WindowManager.LayoutParams.FLAG_FULLSCREEN,WindowManager.LayoutParams.FLAG_FULLSCREEN);
```

We use this 2 sentences to hide the status bar of the mobile when running the app

12. Layout



```xml
<?xml version="1.0" encoding="utf-8"?>
<shape xmlns:android="http://schemas.android.com/apk/res/android">
    <gradient
        android:startColor="@color/purple"
        android:endColor="@color/bluelight"
        android:angle="45"
        />
</shape>
```

We create the background color of our app by creating gradient color with purple and bluelight which are declared in colors file

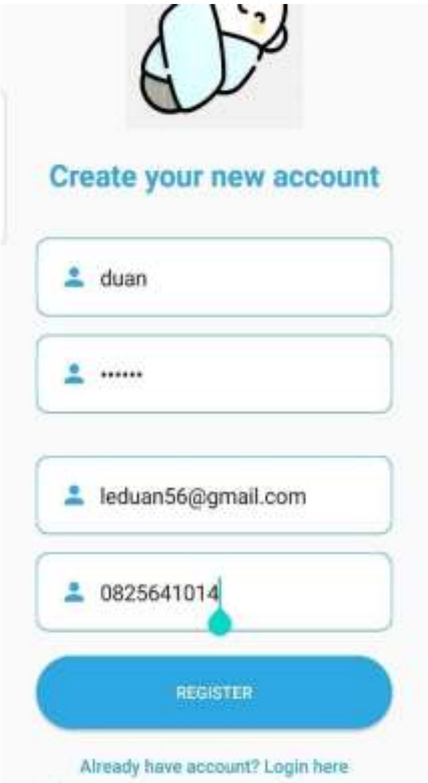
Android:angle to choose the mixture part of the gradient that is suitable for your design

Limitation of app

- We already create Register class for the user apply to be a helper. The whole data is uploaded to the database. Then we load it into the Helper Infor to manage, all data is loaded except the avatar. In this case, we use the Uri to store the content, but we cannot load the content despite using 3rd library and many methods. In the future, we will fix it soon. Thank you and have a great trial.

Demo app using sreenshots

Midterm Version



Create your new account

duan

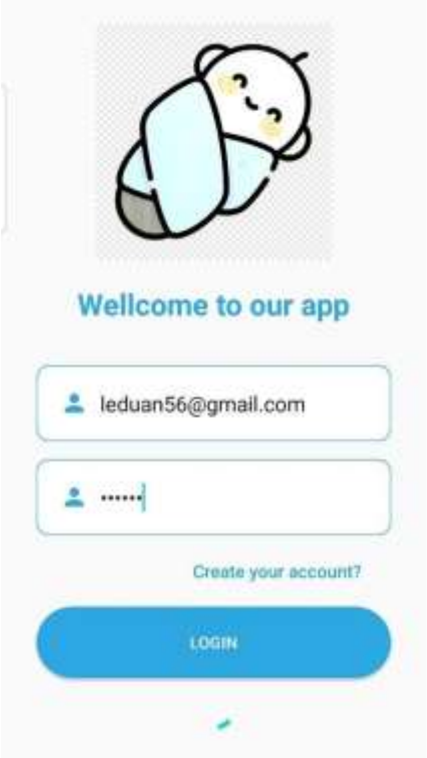
••••••

leduan56@gmail.com

0825641014

REGISTER

Already have account? Login here



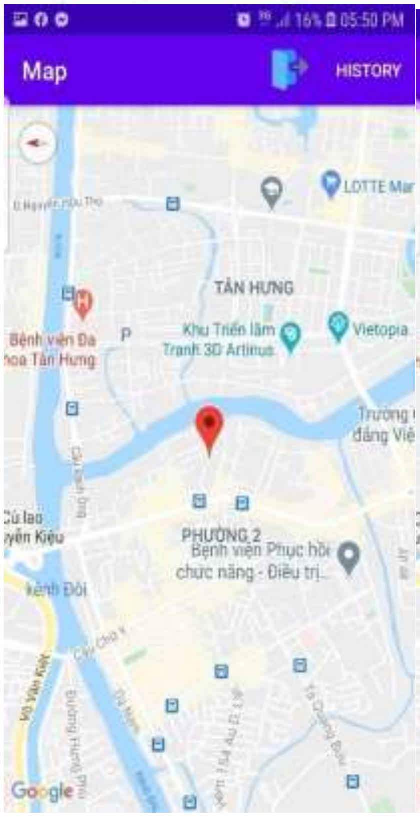
Wellcome to our app

leduan56@gmail.com

••••••

Create your account?

LOGIN

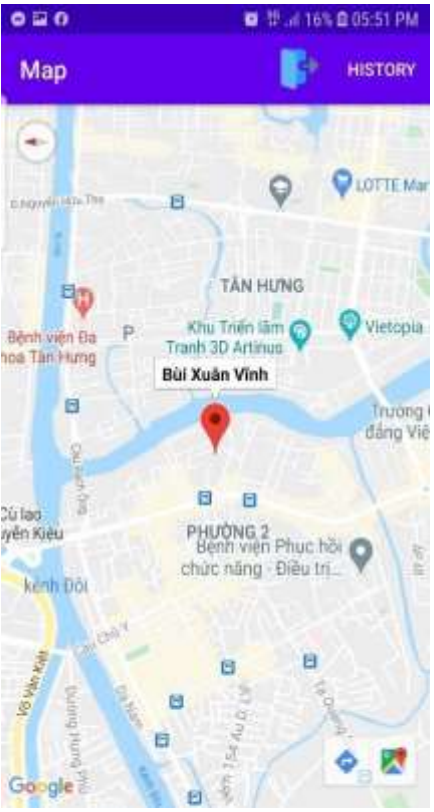


05:50 PM

Map

HISTORY

LOTTE Mar
TÂN HƯNG
Khu Triển lãm Tranh 3D Artinus
Vietopia
Bệnh viện Đa khoa Tân Hưng
Trường Đại đăng Việ
PHƯỜNG 2
Bệnh viện Phục hồi chức năng - Điều trị...
Cù lao Nguyễn Kiệu
kênh Đôi
Google



05:51 PM

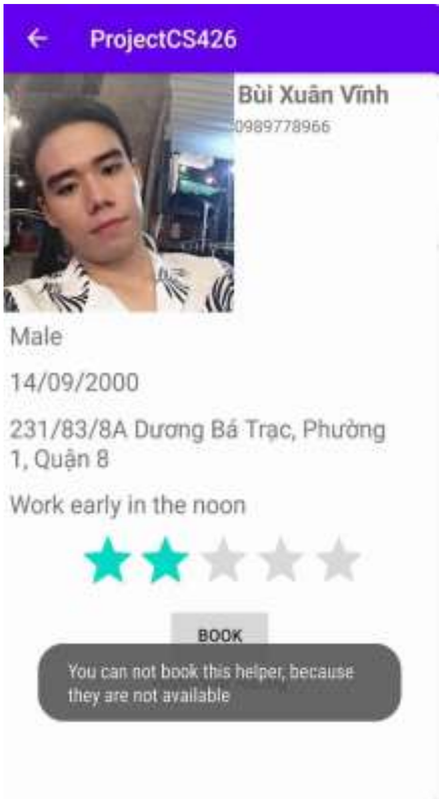
Map

HISTORY

Bùi Xuân Vĩnh

LOTTE Mar
TÂN HƯNG
Khu Triển lãm Tranh 3D Artinus
Vietopia
Bệnh viện Đa khoa Tân Hưng
Trường đăng Việ
PHƯỜNG 2
Bệnh viện Phục hồi chức năng - Điều trị...
Cù lao Nguyễn Kiệu
kênh Đôi
Google

First we create an account and login successfully , It will show the maps and we can see the red marker which means there is one helper in that place . Click on the marker we can see the name of the helper



Bùi Xuân Vĩnh

0989778966

Male

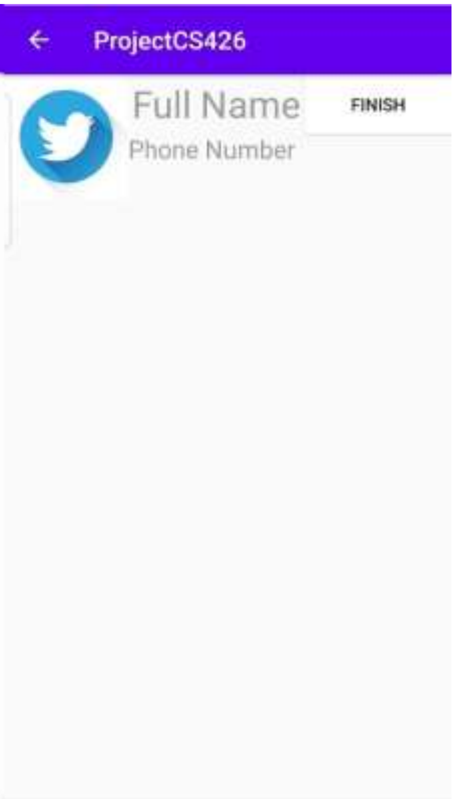
14/09/2000

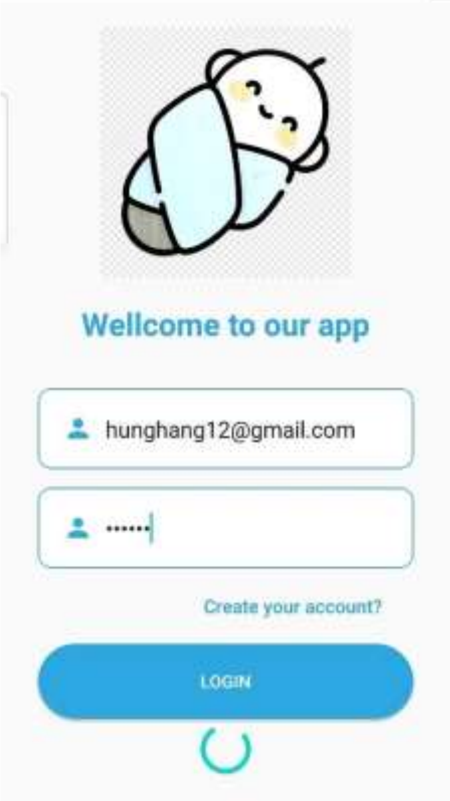
231/83/8A Dương Bá Trạc, Phường 1, Quận 8

Work early in the noon

★★☆☆☆

You can not book this helper, because they are not available

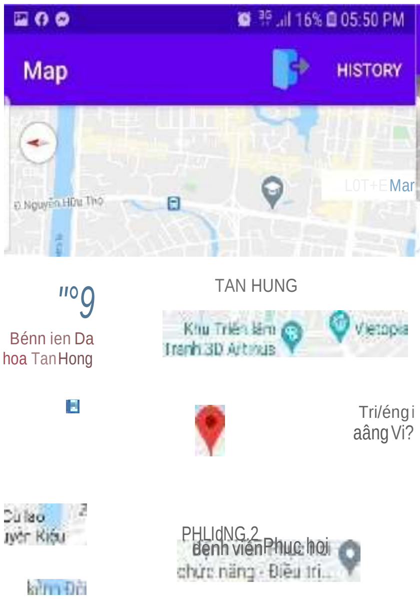


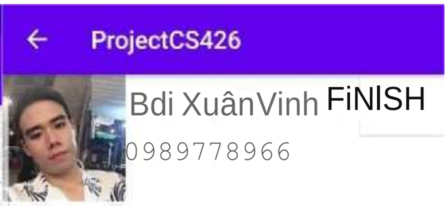


hunghang12@gmail.com

••••••

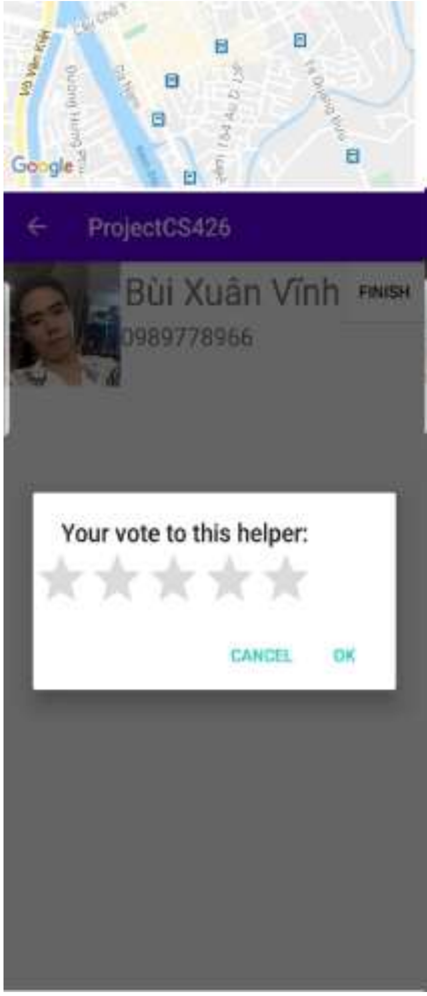
We cannot book this helper because the hunghang12@gmail.com which was created by us before , booked this helper . So that we go into hunghang12@gmail.com and the history of leduan56@gmail.com are still nothing





Bdi XuânVinh

0989778966

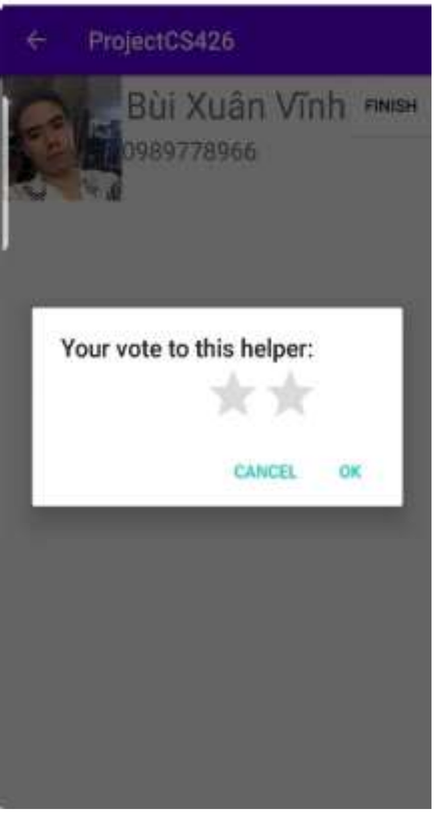


ProjectCS426

Bùi Xuân Vĩnh

0989778966

Your vote to this helper:



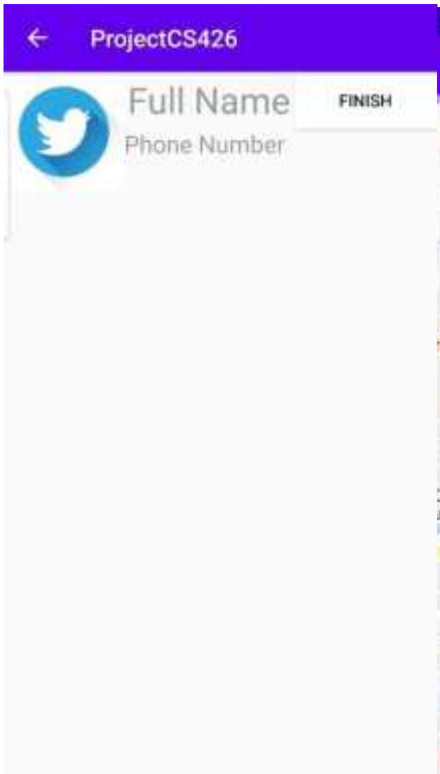
ProjectCS426

Bùi Xuân Vĩnh

0989778966

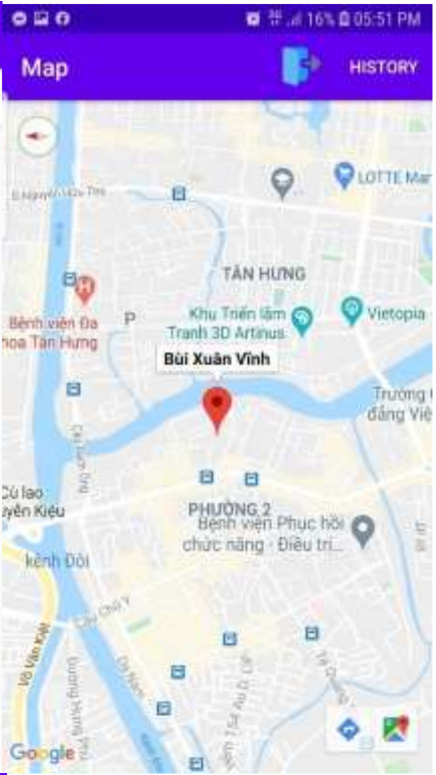
Your vote to this helper:

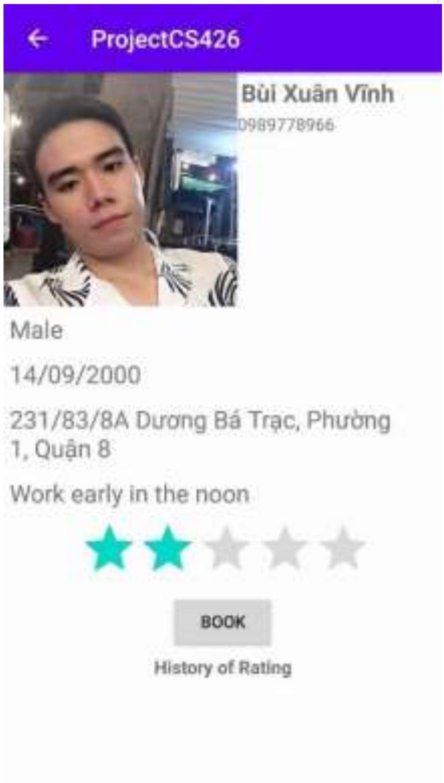
We click on the history and see the helper had been booker by account hanghung12@gmail.com . To end this we must rate 1 – 5 star for this helper to let them free because when finish hiring , customer often rate for the helper and click finish



Full Name

Phone Number





Bùi Xuân Vĩnh

0989778966

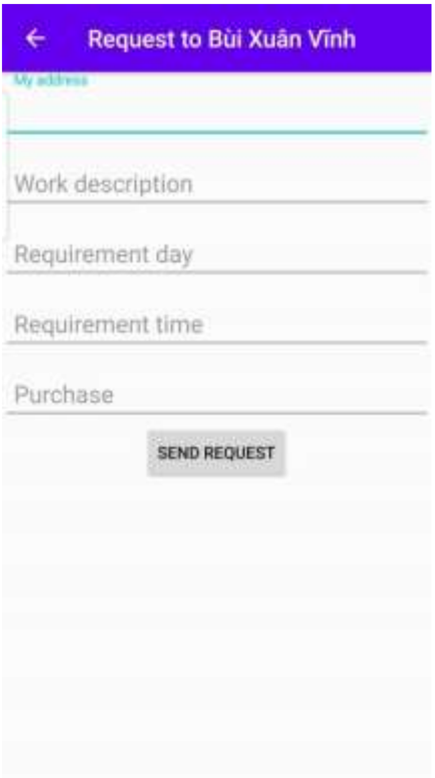
Male

14/09/2000

231/83/8A Dương Bá Trạc, Phường 1, Quận 8

Work early in the noon

★★☆☆☆



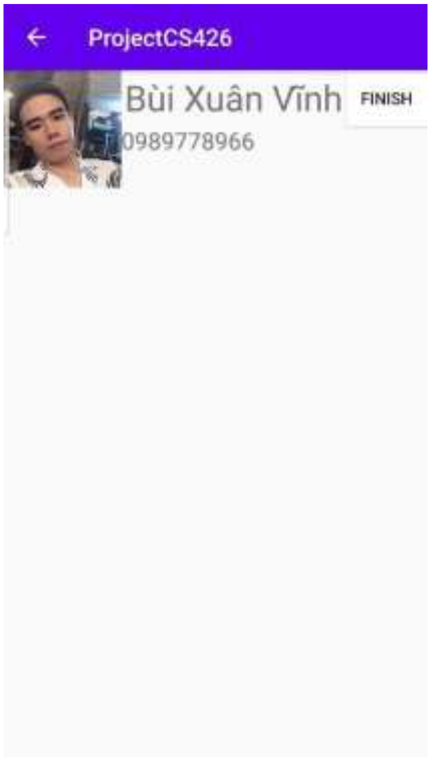
My address

Work description

Requirement day

Requirement time

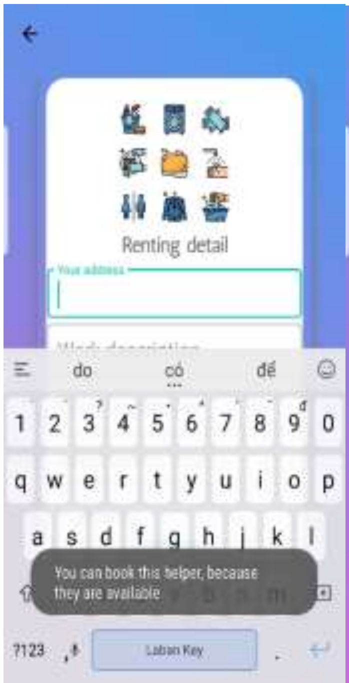
Purchase



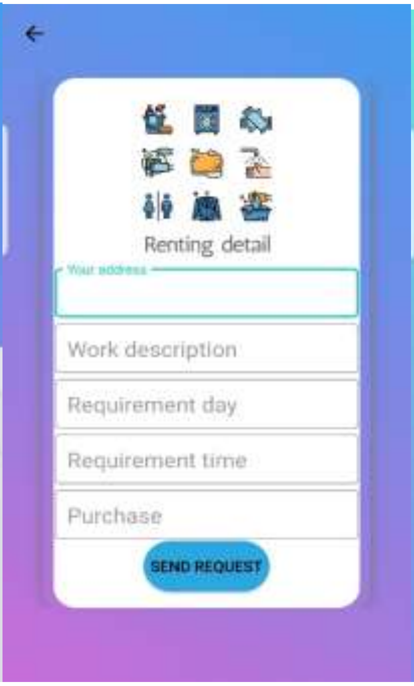
Bùi Xuân Vĩnh
0989778966

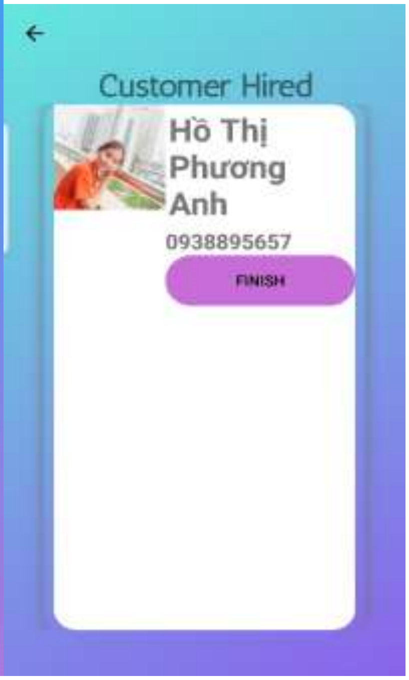
Now we go to the leduan56@gmail.com account and book this helper as normal and it will go into the history

Final Version

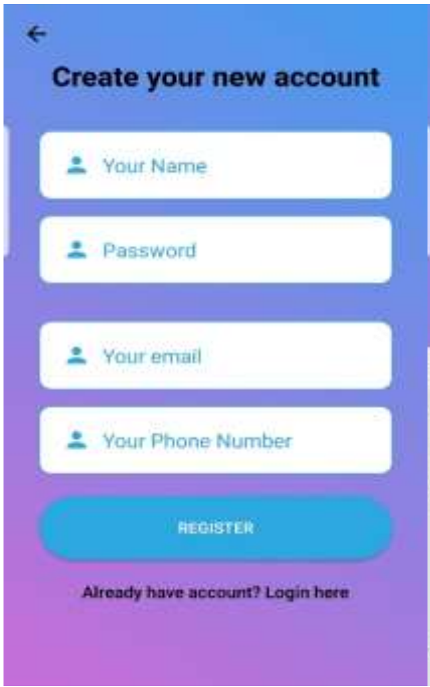


You can book this helper, because they are available





Hồ Thị Phương Anh
0938895657


Create your new account

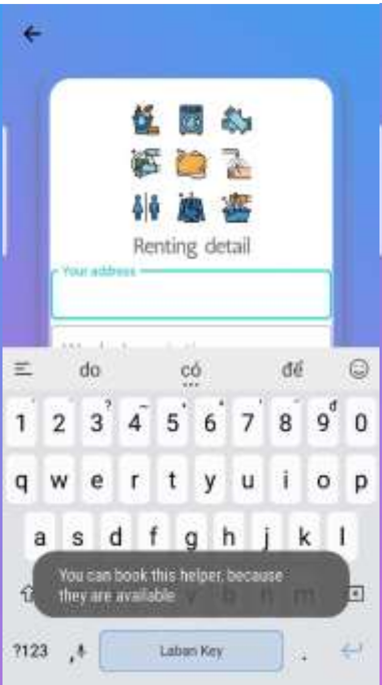
Your Name

Password

Your email

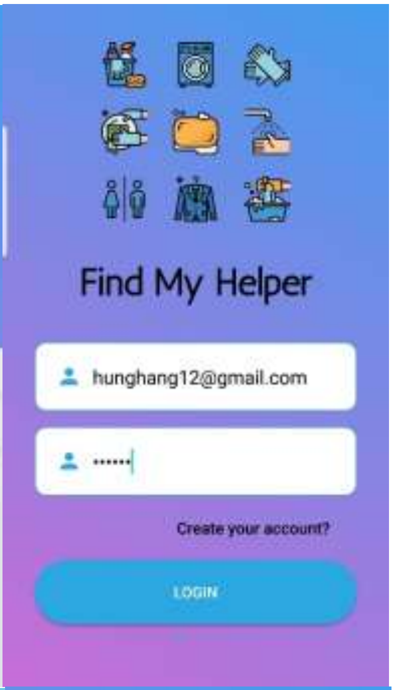
Your Phone Number

Already have account? Login here


Renting detail

Your address

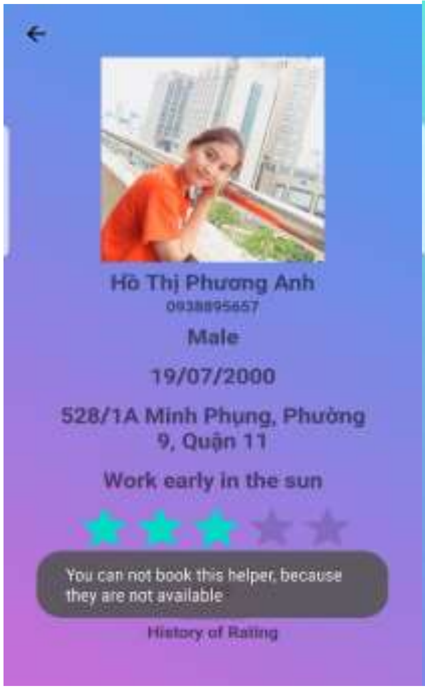
You can book this helper, because they are available


Find My Helper

hunghang12@gmail.com

••••••

Create your account?


Hồ Thị Phương Anh

0938895657

Male

19/07/2000

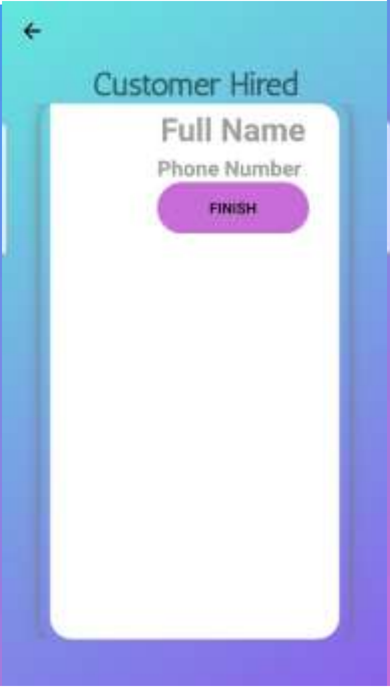
528/1A Minh Phụng, Phường 9, Quận 11

Work early in the sun

★★★☆☆

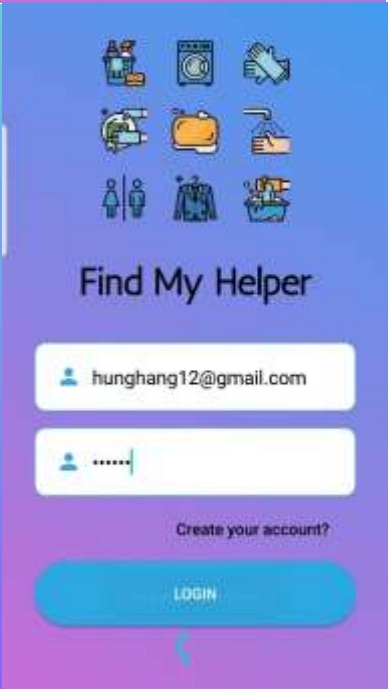
You can not book this helper, because they are not available

History of Rating


Customer Hired

Full Name

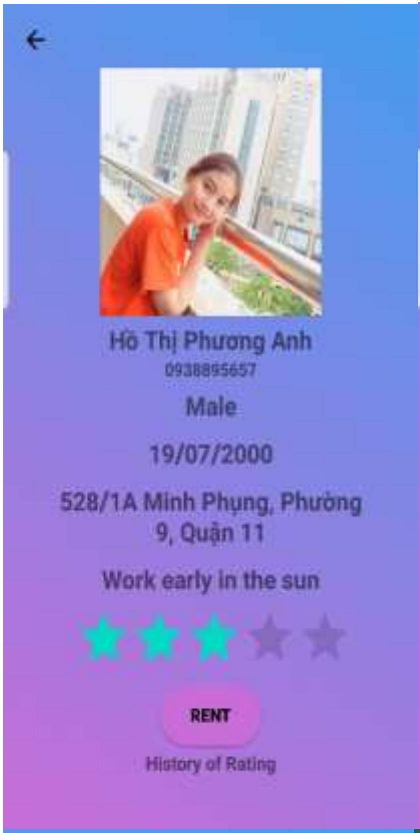
Phone Number


Find My Helper

hunghang12@gmail.com

••••••

Create your account?



Hồ Thị Phương Anh

0938895657

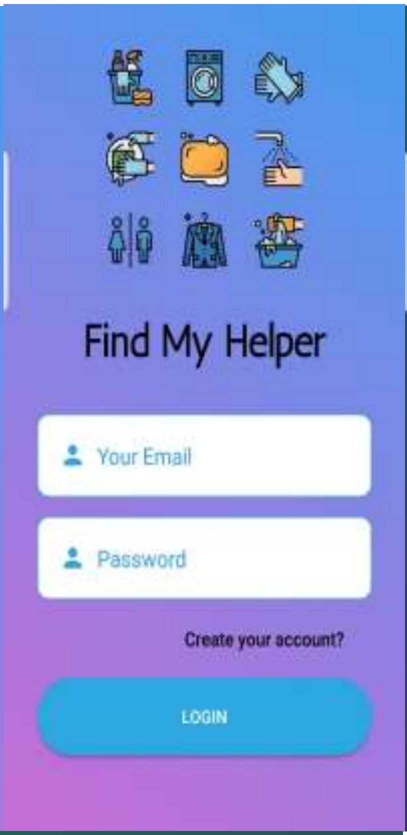
Male

19/07/2000

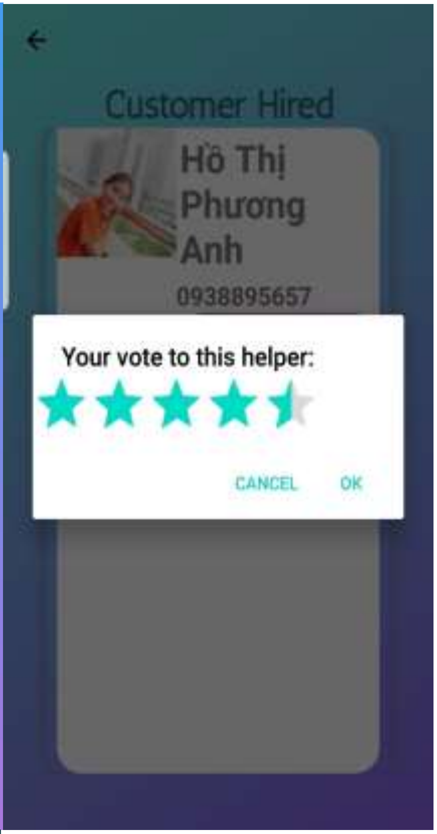
528/1A Minh Phụng, Phường 9, Quận 11

Work early in the sun

★★★☆☆



Find My Helper

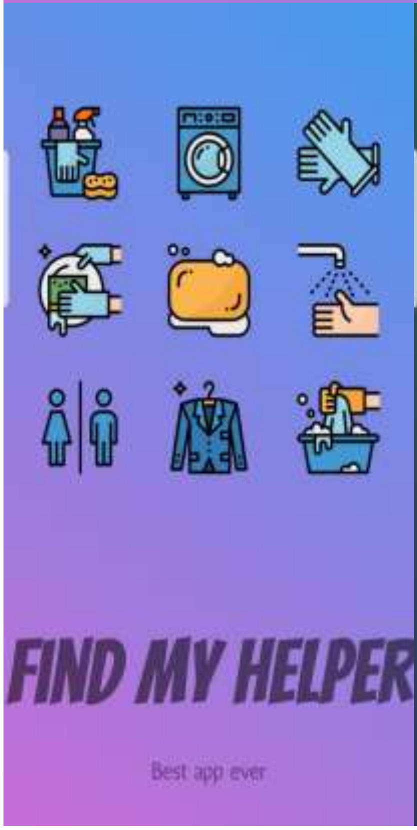


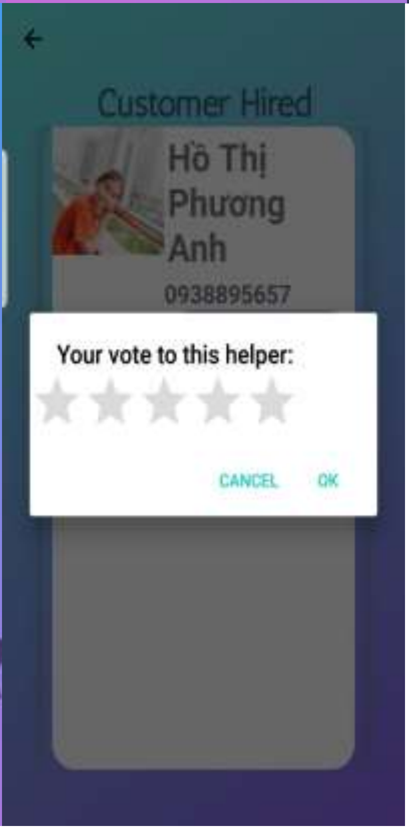
Customer Hired

Hồ Thị Phương Anh

0938895657

Your vote to this helper:





Customer Hired

Hồ Thị Phương Anh

0938895657

Your vote to this helper:

Material

- Firebase via https://firebase.google.com/
- Tourist App Assignment
- Stackoverflow