

*Corso di Informatica e Telecomunicazioni
articolazione Informatica*

**Istituto Tecnico Industriale
Liceo Scientifico
“Francesco Giordani”
Caserta**

**Progetto Android
Connessione dell'app ad un database remoto**

Candidato
Emanuele d'Albenzio
Antonio Finale
classe V B informatica

Tutor
Prof.
Ennio Ranucci

Anno Scolastico
2020/2021

DESCRIZIONE DEL PROGETTO

L'obiettivo è quello di realizzare un'app eseguibile su device con sistema operativo Android che permetterà all'utente di collegarsi al database remoto.

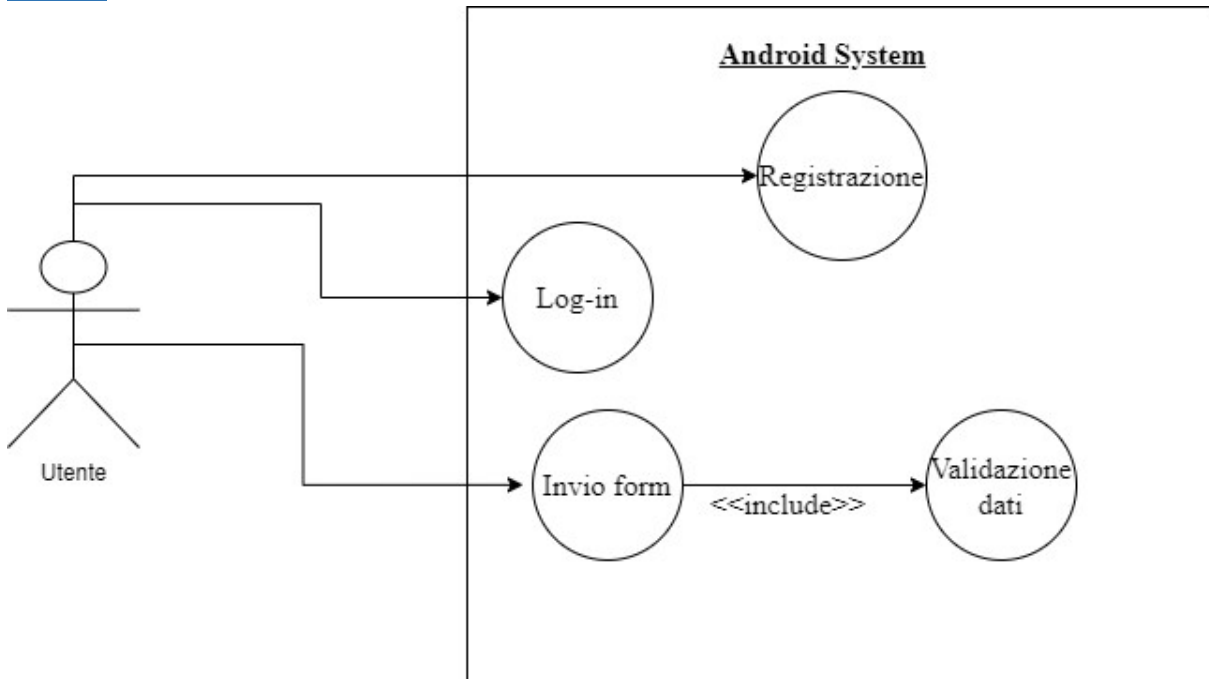
DESCRIZIONE DEL PRODOTTO

L'applicazione avrà un'interfaccia molto intuitiva ed amichevole. All'avvio vengono richiesti i dati da trasferire al database remoto con un pulsante di invio, subito dopo restituisce se l'operazione è avvenuta con successo o no.

STUDIO DI FATTIBILITÀ

L'obiettivo dell'app è dunque quello di acquisire i dati inseriti dall'utente e di inviarli al database memorizzato su di un server. Per realizzare il Progetto, l'azienda utilizzerà l'ambiente di sviluppo Android Studio, per l'implementazione del database saranno implementati degli script in linguaggio Php (linguaggio lato server). Il server è già presente in azienda e quindi il database MySql verrà caricato su di esso.

UML



SCHEMA DI DEFINIZIONE DEI DATI

Tabella	Attributo	Chiave	Formato	Dimensione	Indicizzato	Note
Clienti	idUtente	Primary Key	alfanumerico (varchar)	16	NO	autoincrementante
	username		carattere (varchar)	50	NO	
	e-mail		carattere (varchar)	50	NO	
	cellulare		carattere (varchar)	10	NO	
	password		carattere (varchar)	16	NO	
	sex		carattere (varchar)	10	NO	

IMPLEMENTAZIONE IN AMBIENTE ANDROID STUDIO

In android studio è possibile programmare sia in linguaggio Kotlin sia in linguaggio Java, noi abbiamo scelto di programmare in linguaggio Java.

Una volta scaricato ed installato il software Android studio possiamo passare all'implementazione dell'app iniziando dalla home. Creiamo il progetto e nella finestra con estensione XML impostiamo la parte grafica dell'app. Possiamo quindi trascinare all'interno del codice xml un pulsante di Start successivamente apriamo il file .xml del log-in e trasciniamo due textbox per l'inserimento di e-mail e password e un pulsante. In modo analogo nella finestra xml collegata all'interfaccia di registrazione trasciniamo un pulsante e quattro textbox: username, e-mail, cellulare e psw. Abbiamo usato due radiobutton per gestire il sesso dell'utente. Di seguito riportiamo il codice:

Login.xml

```
<?xml version="1.0" encoding="utf-8"?>

<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity"
    >

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="vertical"
        android:layout_centerVertical="true"
        >

        <TextView
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:gravity="center"
            android:text="Login"
            android:textSize="30sp"
            android:textStyle="bold"
            android:layout_marginTop="10dp"
            android:textColor="@color/colorPrimary"
```

```

/>

<com.rengwuxian.materialEditText.MaterialEditText
    android:id="@+id/email"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_gravity="center_horizontal"
    android:layout_marginLeft="20dp"
    android:layout_marginRight="20dp"
    android:layout_marginTop="10dp"
    app:met_floatingLabel="normal"
    android:background="@android:color/white"
    android:hint="Email "
    android:inputType="textEmailAddress"/>
<com.rengwuxian.materialEditText.MaterialEditText
    android:id="@+id/password"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_gravity="center_horizontal"
    android:layout_marginLeft="20dp"
    android:layout_marginRight="20dp"
    app:met_floatingLabel="normal"
    android:layout_marginTop="10dp"
    android:background="@android:color/white"
    android:hint="Password"
    android:inputType="textPassword"/>

<androidx.appcompat.widget.AppCompatCheckBox
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Ricordami"
    android:buttonTint="@color/colorPrimaryDark"
    android:textColor="@color/colorPrimaryDark"
    android:layout_gravity="center"
    android:id="@+id/checkbox"
    android:layout_marginTop="10dp"/>

<Button
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginLeft="20dp"
    android:layout_marginRight="20dp"
    android:layout_marginTop="10dp"
    android:text="Login"
    android:textColor="@android:color/white"
    android:textAllCaps="false"
    android:id="@+id/loginBtn"
    android:background="@color/colorPrimaryDark"/>

<Button
    android:layout_width="match_parent"
    android:layout_height="wrap_content"

```

```
    android:layout_marginLeft="20dp"
    android:layout_marginRight="20dp"
    android:layout_marginTop="10dp"
    android:text="Registrati"
    android:textColor="@android:color/white"
    android:textAllCaps="false"
    android:id="@+id/buttonOne"
    android:background="@color/colorPrimaryDark"/>
```

```
</LinearLayout>
```

```
</RelativeLayout>
```

Registrazione.xml

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".Register">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:orientation="vertical"
        android:layout_below="@+id/toolbar"
        tools:ignore="UnknownId">
        <TextView
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_marginTop="10dp"
            android:text="Crea nuovo account"
            android:gravity="center"
            android:textColor="@color/colorPrimaryDark"
            android:textSize="20dp"
            android:textStyle="bold"/>

        <com.rengwuxian.materialEditText.MaterialEditText
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:hint="Username"
            android:layout_marginRight="20dp"
            android:layout_marginLeft="20dp"
            android:layout_marginTop="10dp"
            android:id="@+id/username"
            android:padding="5dp">
```

```

/>
<com.rengwuxian.materialEditText.MaterialEditText
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:hint="Email"
    android:layout_marginRight="20dp"
    android:layout_marginLeft="20dp"
    android:layout_marginTop="10dp"
    android:id="@+id/email"
    android:padding="5dp"
/>
<com.rengwuxian.materialEditText.MaterialEditText
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:hint="Password"
    android:layout_marginRight="20dp"
    android:layout_marginLeft="20dp"
    android:layout_marginTop="10dp"
    android:inputType="textPassword"
    android:id="@+id/password"
    android:padding="5dp"
/>
<com.rengwuxian.materialEditText.MaterialEditText
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:hint="Cellulare"
    android:layout_marginRight="20dp"
    android:layout_marginLeft="20dp"
    android:layout_marginTop="10dp"
    android:id="@+id/mobile"
    android:padding="5dp"
/>

<RadioGroup
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="vertical"
    android:layout_marginTop="10dp"
    android:padding="10dp"
    android:id="@+id/radioButton">
    <RadioButton
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:textStyle="bold"
        android:textSize="14sp"
        android:textColor="@color/colorPrimaryDark"
        android:text="Maschio"
        android:buttonTint="@color/colorPrimaryDark"
        android:id="@+id/male"/>
    <RadioButton
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"

```

```
        android:textStyle="bold"
        android:textSize="14sp"
        android:buttonTint="@color/colorPrimaryDark"
        android:textColor="@color/colorPrimaryDark"
        android:text="Femmina"
        android:layout_marginTop="10dp"
        android:id="@+id/female"/>
</RadioGroup>

<Button
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginTop="20dp"
    android:background="@color/colorPrimaryDark"
    android:layout_marginLeft="30dp"
    android:layout_marginRight="30dp"
    android:text="Registrati"
    android:textColor="@android:color/white"
    android:textAllCaps="false"
    android:id="@+id/register"/>
</LinearLayout>

</RelativeLayout>
```

Una volta impostata la parte grafica dell'applicazione possiamo andare nella finestra della log-in con estensione .java in modo da iniziare a programmare l'interfaccia. Utilizzando il metodo findViewById colleghiamo ad ogni oggetto inserito nell'xml una variabile e controlliamo il corretto inserimento dei dati. In fine andremo a gestire il pulsante per l'invio del form ad uno script php usando il metodo POST. Nello script php, editabile con un qualsiasi editor di testo, basta salvare con estensione .php, effettuiamo il collegamento al database ed eseguiamo una query per il controllo delle credenziali, ritornando un messaggio di fine operazione.

Login.java

```
package com.example.login;

import android.app.ProgressDialog;
import android.content.Context;
import android.content.Intent;
import android.content.SharedPreferences;
import android.os.Bundle;
import android.text.TextUtils;
import android.view.View;
import android.widget.Button;
import android.widget.CheckBox;
import android.widget.ImageButton;
import android.widget.Toast;

import androidx.appcompat.app.AppCompatActivity;

import com.android.volley.AuthFailureError;
import com.android.volley.DefaultRetryPolicy;
import com.android.volley.Request;
import com.android.volley.Response;
import com.android.volley.VolleyError;
import com.android.volley.toolbox.StringRequest;
import com.rengwuxian.materialedittext.MaterialEditText;

import java.util.HashMap;
import java.util.Map;

public class MainActivity extends AppCompatActivity {

    MaterialEditText email,password;

    CheckBox checkedStatus;

    SharedPreferences sharedPreferences;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        Button login = findViewById(R.id.loginBtn);
        Button buttonOne = findViewById(R.id.buttonOne);
        buttonOne.setOnClickListener(new View.OnClickListener() {
            public void onClick(View v) {

                Intent activity2Intent = new Intent(getApplicationContext(), Register.class);
                startActivity(activity2Intent);
            }
        });
    }
}
```

```

        email = findViewById(R.id.email);
        password = findViewById(R.id.password);
        checkedStatus = findViewById(R.id.checkbox);
        sharedPreferences = getSharedPreferences("UserInfo", Context.MODE_PRIVATE);
        String loginStatus = sharedPreferences.getString(getResources().getString(R.string.prefStatus), "");
        if (loginStatus.equals("loggedin")){
            startActivity(new Intent(MainActivity.this, AppStartActivity.class));
        }
        finish();
    }

    login = findViewById(R.id.loginBtn);
    login.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            String tex_email = email.getText().toString();
            String tex_password = password.getText().toString();
            if (TextUtils.isEmpty(tex_email) || TextUtils.isEmpty(tex_password)){
                Toast.makeText(MainActivity.this, "Sono richiesti tutti i campi", Toast.LENGTH_SHORT).show();
            }
            else{
                login(tex_email, tex_password);
            }
        }
    });
}

private void login(final String email, final String password){
    final ProgressDialog progressDialog = new ProgressDialog(MainActivity.this);
    progressDialog.setTitle("Login in corso");
    progressDialog.setCancelable(false);
    progressDialog.setCanceledOnTouchOutside(false);
    progressDialog.setIndeterminate(false);
    progressDialog.show();
    String url = "http://192.168.1.7/loginregister/login.php";
    StringRequest request = new StringRequest(Request.Method.POST, url, new Response.Listener<String>() {
        @Override
        public void onResponse(String response) {

            if (response.equals("Login Success")){
                Toast.makeText(MainActivity.this, response, Toast.LENGTH_SHORT).show();
            }
        }
    });
}

```

```

        SharedPreferences.Editor editor = sharedPreferences.edit()
;
        if (checkedStatus.isChecked()){
            editor.putString(getResources().getString(R.string.pre
fStatus),"loggedin");
        }
        else{
            editor.putString(getResources().getString(R.string.pre
fStatus),"loggedout");
        }
        editor.apply();
        startActivity(new Intent(MainActivity.this,AppStartActivit
y.class));

        progressDialog.dismiss();
        finish();
    }
    else {
        Toast.makeText(MainActivity.this, response, Toast.LENGTH_S
HORT).show();

        progressDialog.dismiss();
    }
}
}, new Response.ErrorListener() {
    @Override
    public void onErrorResponse(VolleyError error) {
        Toast.makeText(MainActivity.this, error.toString(), Toast.LENG
TH_SHORT).show();
        progressDialog.dismiss();
    }
}){
    @Override
    protected Map<String, String> getParams() throws AuthFailureError
{
        HashMap<String,String> param = new HashMap<>();
        param.put("email",email);
        param.put("psw",password);
        return param;
    }
};

    request.setRetryPolicy(new DefaultRetryPolicy(30000,DefaultRetryPolicy
.DEFAULT_MAX_RETRIES,DefaultRetryPolicy.DEFAULT_BACKOFF_MULT));
    MySingleton.getInstance(MainActivity.this).addToRequestQueue(request)
;
}
}
}

```

Login.php

```
<?php
```

```
require "conn.php";
```

```
$email = $_POST["email"];
```

```
$password = $_POST["psw"];
```

```
$isValidEmail = filter_var($email, FILTER_VALIDATE_EMAIL);
```

```
if($conn){
```

```
if( $isValidEmail === false){
```

```
echo "This Email is not valid";
```

```
}else{
```

```
$sqlCheckEmail = "SELECT * FROM users_table WHERE email LIKE  
'$email'";
```

```
$usernameQuery = mysqli_query($conn,$sqlCheckEmail);
```

```
if(mysqli_num_rows($usernameQuery) > 0){
```

```
$sqlLogin = "SELECT * FROM users_table WHERE email LIKE  
'$email' AND password LIKE '$password'";
```

```
$loginQuery = mysqli_query($conn,$sqlLogin);
```

```
if(mysqli_num_rows($loginQuery) > 0){
```

```
echo "Login Success";

}

else{

echo "Wrong Password";

}

}else{

echo "This Email is not registered";

}

}

}

else{

echo "Connection Error";

} ?>
```

Finita la parte di accesso passiamo alla parte di registrazione dell'utente al database. Allo stesso modo del form di login, una volta acquisiti i dati li inviamo tramite metodo POST ad uno script php il quale si occuperà di trasferire questi dati nel database.

Registrazione.java

```
package com.example.login;

import android.app.ProgressDialog;
import android.content.Context;
import android.content.Intent;
import android.content.SharedPreferences;
import android.os.Build;
import android.os.Bundle;
import android.text.TextUtils;
import android.view.View;
import android.widget.Button;
import android.widget.CheckBox;
import android.widget.RadioButton;
import android.widget.RadioGroup;
import android.widget.Toast;
import android.widget.Toolbar;

import androidx.annotation.RequiresApi;
import androidx.appcompat.app.AppCompatActivity;

import com.android.volley.AuthFailureError;
import com.android.volley.DefaultRetryPolicy;
import com.android.volley.Request;
import com.android.volley.Response;
import com.android.volley.VolleyError;
import com.android.volley.toolbox.StringRequest;
import com.rengwuxian.materialedittext.MaterialEditText;

import java.util.HashMap;
import java.util.Map;
import java.util.Objects;

public class Register extends AppCompatActivity {

    MaterialEditText userName, emailAddress, password, mobile;
    RadioGroup radioGroup;
    Button register;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_register);
        userName = findViewById(R.id.username);
        emailAddress = findViewById(R.id.email);
        password = findViewById(R.id.password);
        mobile = findViewById(R.id.mobile);
        radioGroup = findViewById(R.id.radioButton);

        register = findViewById(R.id.register);
        register.setOnClickListener(new View.OnClickListener() {
```

```

@Override
public void onClick(View v) {

    final String user_name = userName.getText().toString();
    final String email = emailAddress.getText().toString();
    final String txt_password = password.getText().toString();
    final String txt_mobile = mobile.getText().toString();
    int checkedId = radioGroup.getCheckedRadioButtonId();
    RadioButton selected_gender = radioGroup.findViewById(checkedI
d);

    if (selected_gender == null){
        Toast.makeText(Register.this, "Seleziona il sesso,grazie."
, Toast.LENGTH_SHORT).show();
    }
    else {
        final String gender = selected_gender.getText().toString()
;

        if(TextUtils.isEmpty(user_name) || TextUtils.isEmpty(email
) || TextUtils.isEmpty(txt_password) ||
        TextUtils.isEmpty(txt_mobile)){
            Toast.makeText(Register.this, "Tutti i campi sono rich
iesti", Toast.LENGTH_SHORT).show();
        }

        else{
            register(user_name,email,txt_password,txt_mobile,gende
r);
        }
    }
});
}

private void register(final String user_name, final String email, final St
ring txt_password, final String txt_mobile, final String gender) {
    final ProgressDialog progressDialog = new ProgressDialog(Register.this
);

    progressDialog.setTitle("Registerando il tuo account");
    progressDialog.setCancelable(false);
    progressDialog.setCanceledOnTouchOutside(false);
    progressDialog.setIndeterminate(false);
    progressDialog.show();
    String url = "http://192.168.1.7/loginregister/register.php";
    StringRequest request = new StringRequest(Request.Method.POST, url, ne
w Response.Listener<String>() {
        @Override
        public void onResponse(String response) {

```

```

        if (response.equals("You are registered successfully")){
            Toast.makeText(Register.this, response, Toast.LENGTH_SHORT
).show();
            startActivity(new Intent(Register.this,MainActivity.class)
);
            progressDialog.dismiss();
            finish();
        }else {
            Toast.makeText(Register.this, response, Toast.LENGTH_SHORT
).show();
            progressDialog.dismiss();
        }
    }
}, new Response.ErrorListener() {
    @Override
    public void onErrorResponse(VolleyError error) {
        Toast.makeText(Register.this, error.toString(), Toast.LENGTH_S
HORT).show();
        progressDialog.dismiss();
    }
}){
    @Override
    protected Map<String, String> getParams() throws AuthFailureError
{
        HashMap<String,String> param = new HashMap<>();
        param.put("username", user_name);
        param.put("email",email);
        param.put("psw", txt_password);
        param.put("mobile", txt_mobile);
        param.put("gender",gender);
        return param;
    }
};

    request.setRetryPolicy(new DefaultRetryPolicy(30000,DefaultRetryPolicy
.DEFAULT_MAX_RETRIES,DefaultRetryPolicy.DEFAULT_BACKOFF_MULT));
    MySingleton.getInstance(Register.this).addToRequestQueue(request);
}
}
}

```

MySingleton.java (Questa classe è la classe predefinita,della libreria volley, e serve per gestire la coda del traffic internet.)


```
package com.example.quizdb;

import android.content.Context;

import com.android.volley.Cache;

import com.android.volley.Network;

import com.android.volley.Request;

import com.android.volley.RequestQueue;

import com.android.volley.toolbox.BasicNetwork;

import com.android.volley.toolbox.DiskBasedCache;

import com.android.volley.toolbox.HurlStack;

import com.android.volley.toolbox.Volley;

public class MySingleton {

    private static MySingleton mInstance;

    private RequestQueue mRequestQueue;

    private Context mCtx;

    public MySingleton(Context mCtx) {

        this.mCtx = mCtx;
```

```
mRequestQueue = getmRequestQueue();
```

```
}
```

```
public RequestQueue getmRequestQueue(){
```

```
    if(mRequestQueue == null){
```

```
        Cache cache = new
```

```
DiskBasedCache(mCtx.getCacheDir(),1024*1024);
```

```
        Network network = new BasicNetwork(new HurlStack());
```

```
        mRequestQueue = new RequestQueue(cache,network);
```

```
        mRequestQueue =
```

```
Volley.newRequestQueue(mCtx.getApplicationContext());
```

```
    }
```

```
    return mRequestQueue;
```

```
}
```

```
public static synchronized MySingleton getInstance(Context  
context){
```

```
    if (mInstance == null){
```

```
        mInstance = new MySingleton(context);

    }

    return mInstance;

}

public<T> void addToRequestQueue(Request<T> request) {

    mRequestQueue.add(request);

}

}
```

Registrazione.php

```
<?php
```

```
require "conn.php";
```

```
$username = $_POST["username"];
```

```
$email = $_POST["email"];
```

```
$password = $_POST["psw"];
```

```

$mobile = $_POST["mobile"];

$gender = $_POST["gender"];

$isValidEMail = filter_var($email , FILTER_VALIDATE_EMAIL);

if($conn){

if(strlen($password ) > 40 || strlen($password ) < 6){

echo "Password length must be more than 6 and less than 40";

}

else if($isValidEMail === false){

echo "This Email is not valid";

}else{

$sqlCheckUname = "SELECT * FROM users_table WHERE username

LIKE '$username'";

$u_name_query = mysqli_query($conn, $sqlCheckUname);

$sqlCheckEmail = "SELECT * FROM users_table WHERE email LIKE

'$email'";

$email_query = mysqli_query($conn, $sqlCheckEmail);

if(mysqli_num_rows($u_name_query) > 0){

echo "User name allready used type another one";

}else if(mysqli_num_rows($email_query) > 0){

```

```

echo "This Email is allready registered";

}else{

$sql_register      =      "INSERT      INTO      users_table
(username,email,password,phone,gender)      VALUES
('$username','$email','$password','$mobile','$gender)";

if(mysql_query($conn,$sql_register)){

echo "You are registered successfully";

}else{

echo "Failed to register you account";

}

}

}

}

}

else{

echo "Connection Error";

}??>

```

Implementazione librerie (build gradle module:app)

Obbligatorie:

```
-Volley (implementation 'com.android.volley:volley:1.1.1')
```

La libreria volley è una libreria http,la quale, permette il networking per le app android.

“Facoltative”:

in quest’app abbiamo implementazioni “facoltative” in quanto senza di queste ultime il programma funzionerebbe ma con piccole variazioni della grafica e sono le seguenti:

```
-implementation 'androidx.appcompat:appcompat:1.1.0'  
-implementation 'androidx.constraintlayout:constraintlayout:1.1.3'  
-testImplementation 'junit:junit:4.13'  
-androidTestImplementation 'androidx.test:runner:1.2.0'  
-androidTestImplementation 'androidx.test.espresso:espresso-core:3.2.0'
```

AndroidManifest.xml

Nel manifest.xml importiamo, successivamente, i permessi per far sì che

l’app abbia la possibilità di connettersi ad internet ed utilizzare sia la wifi

che la rete dati e sono le seguenti linee di codice da aggiungere prima

dell’application:

```
<uses-permission android:name="android.permission.INTERNET" />  
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />  
<uses-permission android:name="android.permission.ACCESS_WIFI_STATE" />
```

Dichiarazione IP per poter arrivare ai file php:

Per far sì che l’app possa reperire i file di connessione, login e registrazione l’utente deve inserire il percorso anteceduto dall’ip. Quest’ultimo può variare in base al dispositivo sul quale utilizziamo l’app:

-Dispositivo connesso sulla rete locale: viene preso l’ip del pc. (eseguire cmd come amministratore -> digitare ipconfig -> prendere l’ipv4. Esempio: 192.168.x.x)

-Dispositivo virtuale fornitoci da android studio: 10.0.0.2

-Altri tipi di dispositivi virtuali: 10.0.0.3