

I.T.T. ENRICO FERMI PROGETTO APP ALTERNANZA S/L

Con la collaborazione di *formaScienza*

IDEATA DALLE RISPETTIVE CLASSI:

V G: BATTISTI, BRINATI, CAPPELLINI, GASPERINI, GIULIANI, OTTAVIANELLI

V I: BORRELLI, COLÒ, PINELLI, FILIPPI, CERRONI, BIANCHI, ANTONELLI,
SPLENDORI, CARDINI

INDICE

- 1 *APP CAMEROGGI (VG-I)*
- 2 *APP R.E.D.I. (VG)*
- 3 *APP VeraTour (VI)*

APP CAMEROGGI

Come ambiente di sviluppo per l'applicazione, si è ricorsi ad **Android Studio 3.0.**

Le applicazioni in Android sono scritte con il linguaggio di programmazione Java.

Il primo passo è stato realizzare la **pagina di login**, seguita a quella di registrazione generale, la quale poi si suddivide in registrazione per il datore e per il lavoratore, così come la pagina di accesso.

```
import android.content.Intent;[]

public class LoginActivity extends AppCompatActivity {
    private Button registerButton;
    private Button loginButton;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_login);

        final ServerRequest sr = new ServerRequest();

        registerButton = (Button) findViewById(R.id.registerButton);
        loginButton = (Button) findViewById(R.id.email_sign_in_button);

        View.OnClickListener register = new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent intentRegister = new Intent(LoginActivity.this, RegisterActivity.class);
                startActivity(intentRegister);
            }
        };

        registerButton.setOnClickListener(register);

        View.OnClickListener login = new View.OnClickListener() {
            @Override
            public void onClick(View v) {

                Intent intentLogin = new Intent(LoginActivity.this, homePageEmployer.class);
                startActivity(intentLogin);

                int risposta = sr.checkLogin("prova", "prova");

                if (risposta == ServerRequest.EMPLOYER) {

                }
                if (risposta == ServerRequest.WORKER) {

                }
                if (risposta == ServerRequest.LOGINERROR) {

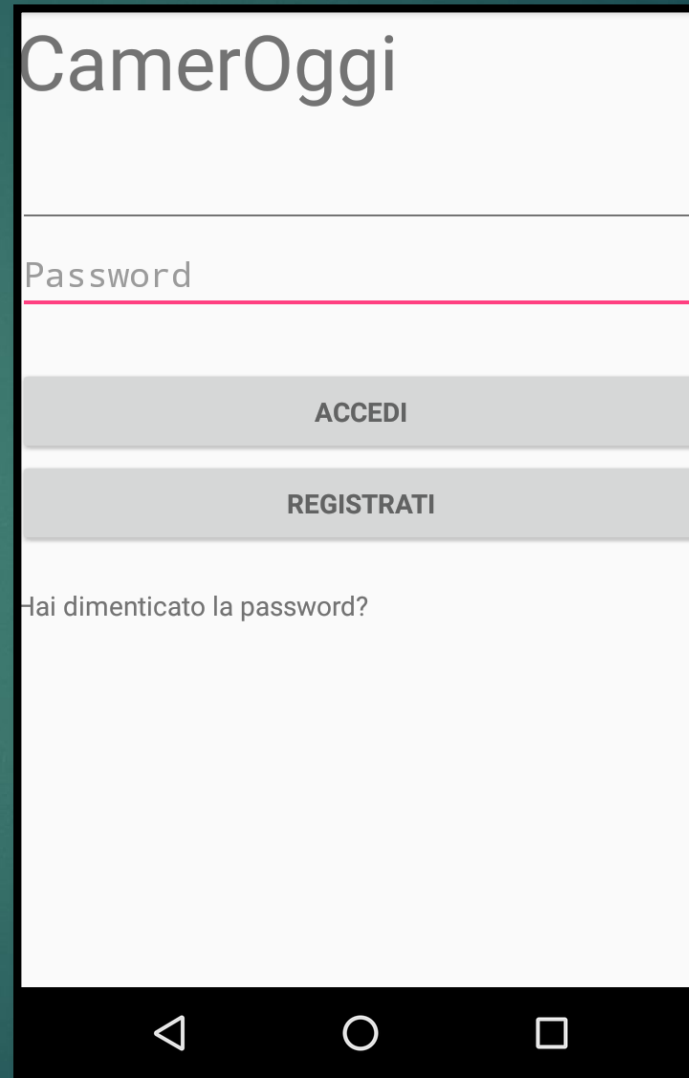
                }
            }
        };

        loginButton.setOnClickListener(login);

    }
}
```

APP CAMEROGGI

Il **risultato**, installandola su un dispositivo qualsiasi Android, è il seguente.



APP CAMEROGGI

Successivamente, si è passati all'activity (pagina) della **registrazione in generale**, con tre campi in comune per tutte e due i tipi; questo per facilitare le operazioni di immissione dati.

```
import android.content.Intent;

public class RegisterActivity extends AppCompatActivity {
    private Button registerEmployer;
    private Button registerWorker;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_register);

        registerEmployer = (Button) findViewById(R.id.registerEmployer);
        registerWorker = (Button) findViewById(R.id.registerWorker);
        View.OnClickListener registerEmp = new View.OnClickListener() {
            @Override
            public void onClick(View v) {

                Intent intentRegisterEmployer = new Intent(RegisterActivity.this, registerEmployerActivity.class);
                startActivity(intentRegisterEmployer);
            }
        };

        registerEmployer.setOnClickListener(registerEmp);

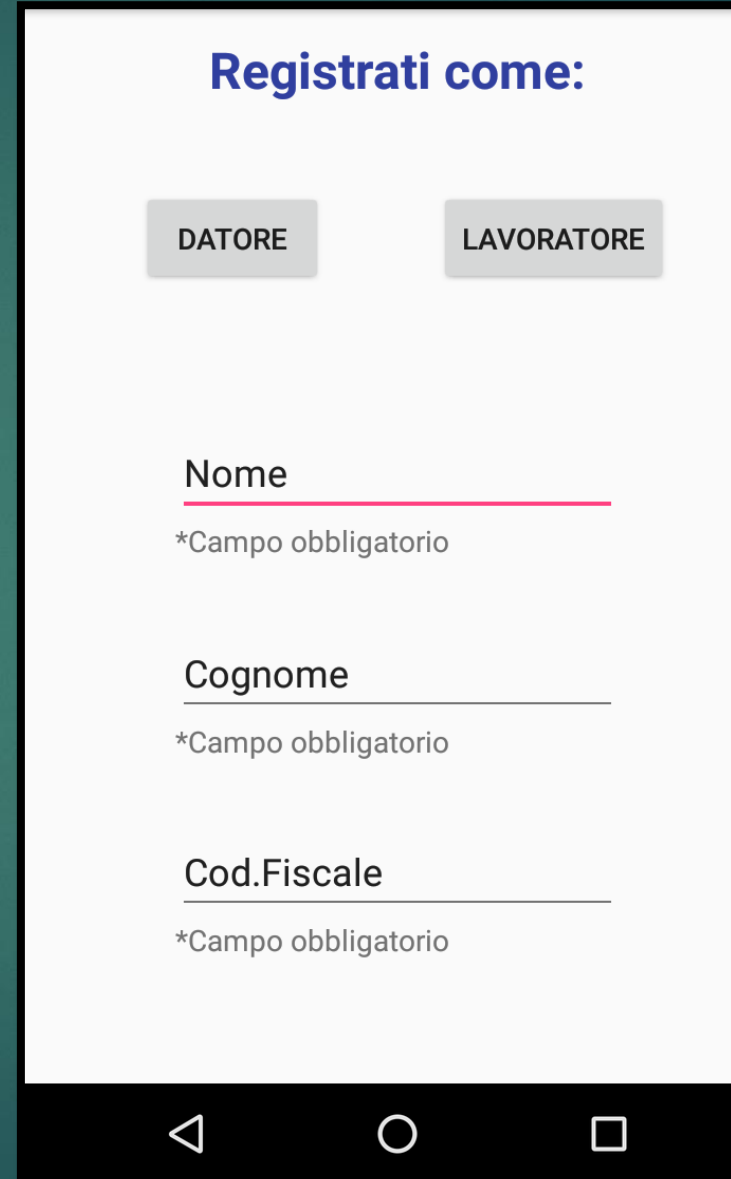
        View.OnClickListener registerWor = new View.OnClickListener() {
            @Override
            public void onClick(View v) {

                Intent intentRegisterWorker = new Intent(RegisterActivity.this, registerWorkerActivity.class);
                startActivity(intentRegisterWorker);
            }
        };

        registerWorker.setOnClickListener(registerWor);
    }
}
```

APP CAMEROGGI

Il **risultato**, installandola su un dispositivo qualsiasi Android, è il seguente.



Registrati come:

DATORE LAVORATORE

Nome
*Campo obbligatorio

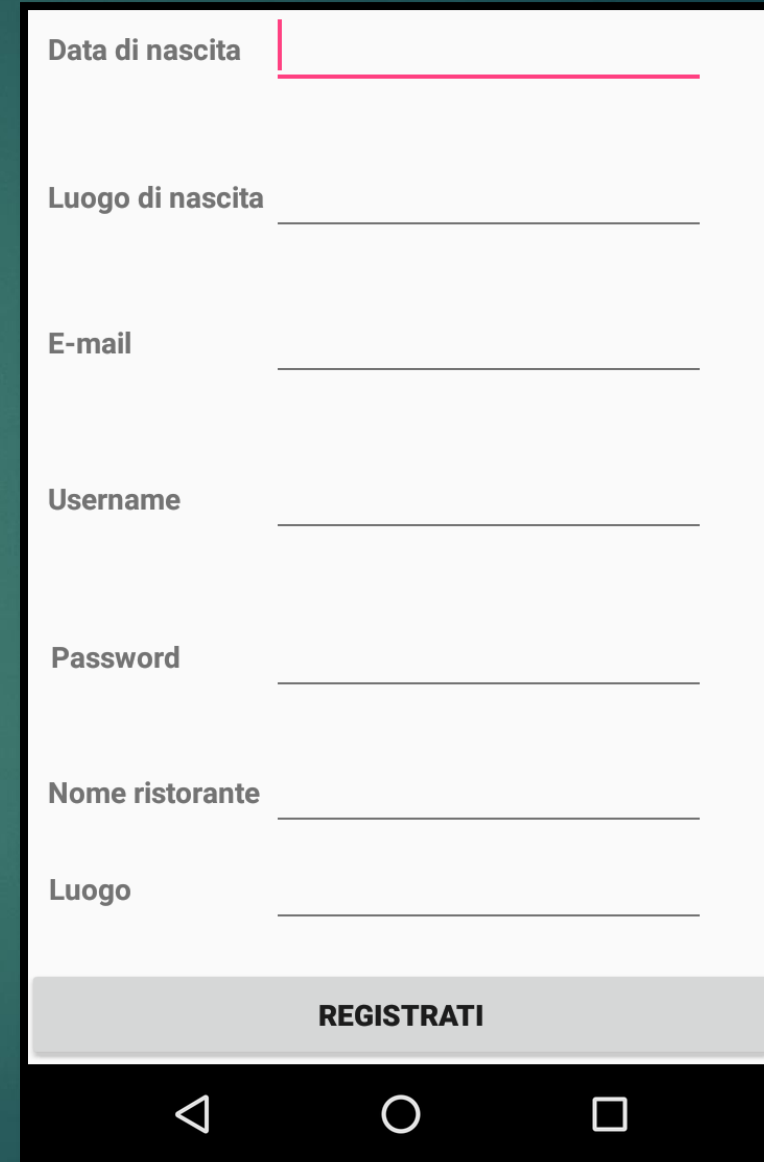
Cognome
*Campo obbligatorio

Cod.Fiscale
*Campo obbligatorio

APP CAMEROGGI

```
import android.support.v7.app.AppCompatActivity;
public class registerEmployerActivity extends AppCompatActivity {
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_register_employer);
    }
}
```

Si è poi passati alla **registrazione del datore di lavoro**, la quale apparirà nel seguente modo:



The screenshot shows a registration form for an employer. The form is displayed on a white background with a black border. It contains several input fields, each with a label to its left and a horizontal line for text entry. The labels are: "Data di nascita", "Luogo di nascita", "E-mail", "Username", "Password", "Nome ristorante", and "Luogo". At the bottom of the form, there is a grey button with the text "REGISTRATI" in black. Below the button, there is a black navigation bar with three white icons: a triangle pointing left, a circle, and a square.

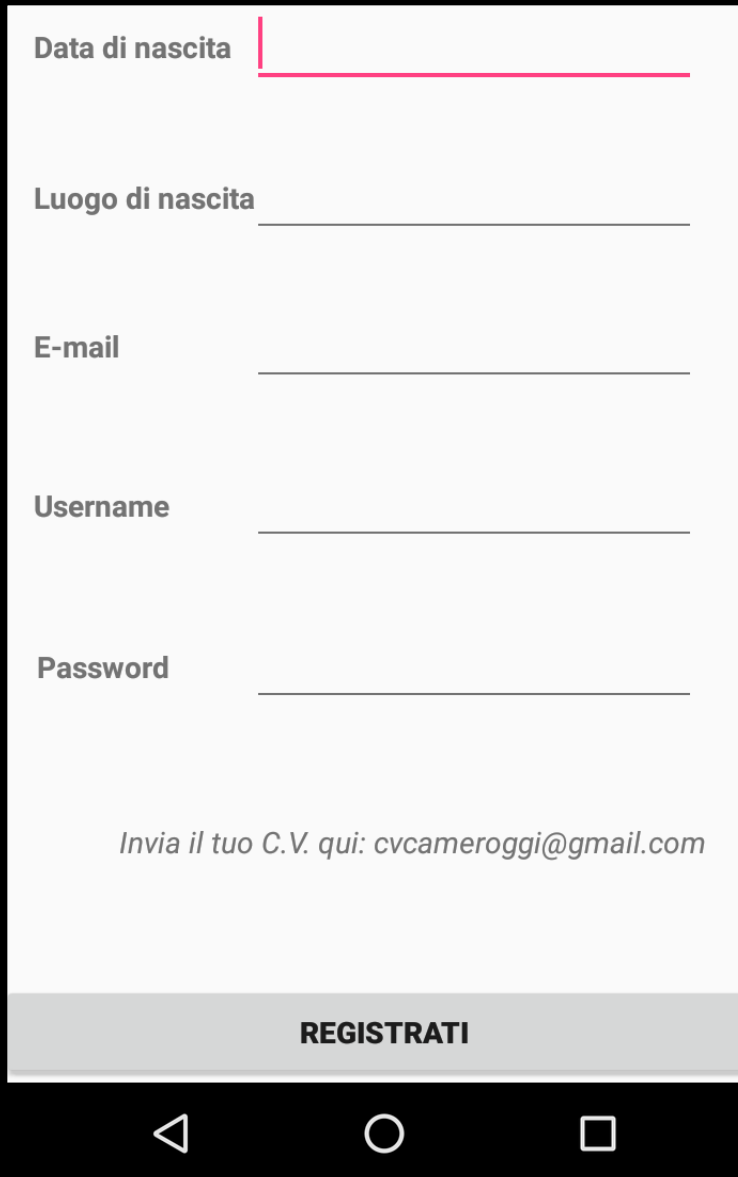
APP CAMEROGGI

```
import android.content.Intent;

public class registerWorkerActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_register_worker);
    }
}
```

Ovviamente, non
manca l'activity della
**registrazione del
lavoratore.**



The screenshot shows a registration form with the following fields:

- Data di nascita
- Luogo di nascita
- E-mail
- Username
- Password

Below the fields, there is a text prompt: *Invia il tuo C.V. qui: cvcameroggi@gmail.com*

At the bottom of the form, there is a button labeled **REGISTRATI**.

APP CAMEROGGI

Non avendo sufficiente tempo a disposizione, il gruppo non ha potuto ultimare l'applicazione. Infatti, ci sarebbero dovute essere anche le due **activity di Homepage** (sia lavoratore che datore).

Nonostante ciò, il gruppo si è reso molto partecipe ed interessato a questo progetto che va fuori dalla solita didattica del programma scolastico, visto che si interfaccia con quella che è una delle tante realtà degli ultimi anni: la creazione di applicazioni per dispositivi mobili.

COS'è REDI

R.E.D.I. (Ricerca eventi discoteche italiane) è un'app di ricerca di discoteche e locali che permette di trovare eventi serali in tutta Italia, attraverso una mappa o un calendario. Permette poi di acquistare prevendite per l'evento desiderato, o di accedere a un servizio di car sharing per raggiungere il luogo della serata. E' stata realizzata da Andrea Brinati, Giorgio Gasperini, Davide Giuliani e Marco Ottavianelli.

Inoltre, permette ai titolari di locali di creare nuovi eventi, completi di descrizione, orari e localizzazione.

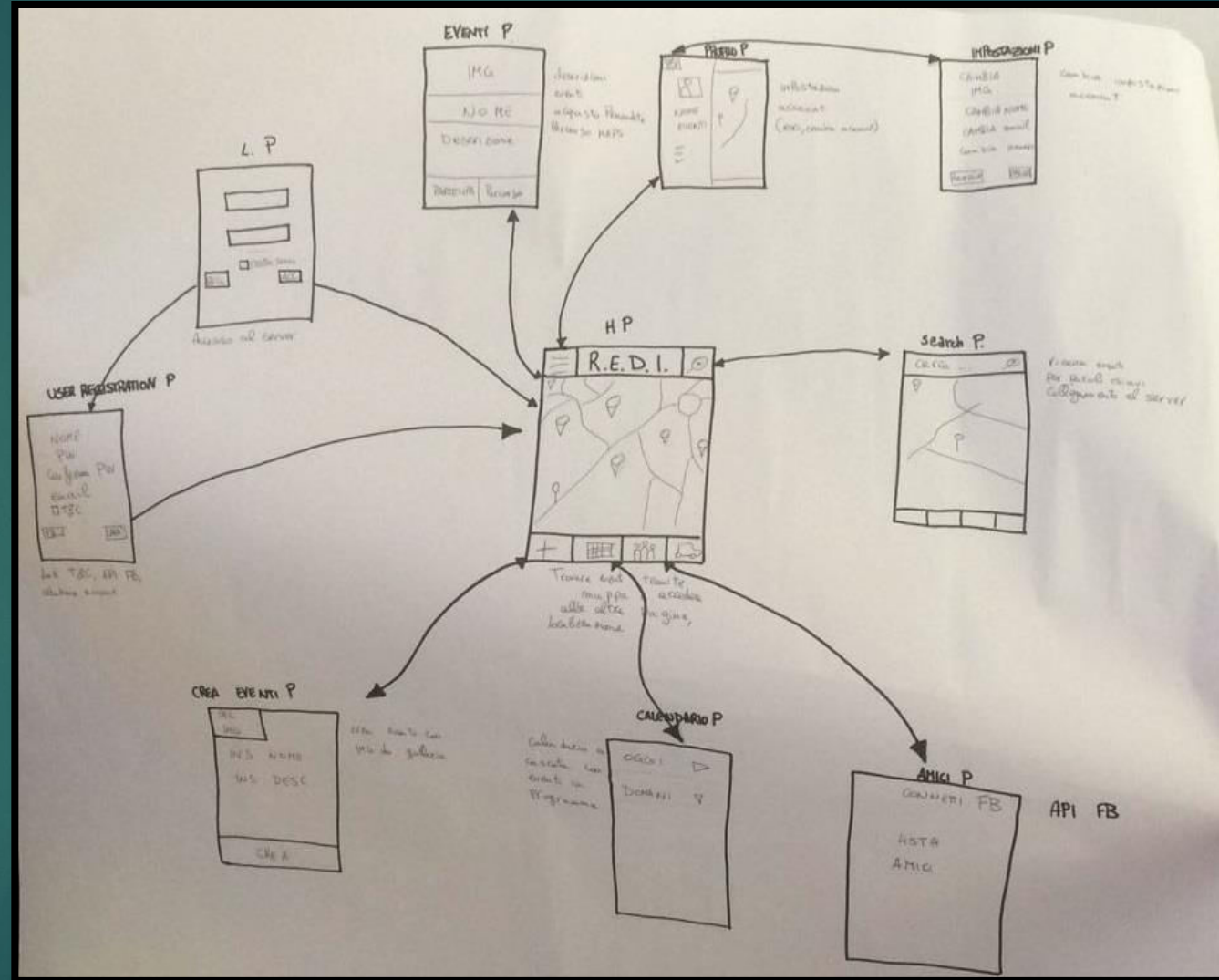
Android Studio

Al fine di realizzare la nostra app, REDI, abbiamo utilizzato come IDE Android Studio.

Basato sul software di JetBrains IntelliJ IDEA, Android Studio è stato progettato specificamente per lo sviluppo di applicazioni Android. E' disponibile il download su Windows, Mac OS X e Linux, e sostituisce gli Android Development Tools (ADT) di Eclipse.



Prima di passare alla fase di implementazione, abbiamo disegnato uno schema delle pagine da inserire nell'app, con relativi collegamenti e funzioni di ogni pagina.



Il Login

La pagina iniziale di REDI è la Login Page, che permette di accedere al proprio account personale attraverso username e password. In caso non si posseda un account, si ha la possibilità di crearne uno, attraverso la Registration Page, associandolo a un indirizzo email. E' anche possibile registrarsi con Facebook grazie alle API.

La Home Page

Una volta completato il login si accede alla Home page, che presenta una mappa (implementata grazie ai cookies di Google), che indica gli eventi disponibili attraverso degli indicatori.

Nella Home Page si può, inoltre, cercare un evento specifico, creare un evento, visualizzare il calendario, accedere alla propria lista amici, e approfittare del servizio di car sharing.

Eventi

Ogni evento è formato da:

- ▶Nome
- ▶Immagine
- ▶Descrizione

Questi sono gli elementi che il creatore dell'evento dovrà mettere a disposizione agli altri utenti.

Inoltre, ogni utente potrà acquistare prevendite, o aggiungere l'evento ai preferiti.

Account

Ogni utente ha la possibilità di gestire il proprio account, al quale si può anche aggiungere una foto profilo. Vi è inoltre una pagina contenente le impostazioni avanzate inerenti al proprio account, accessibile tramite un menù a tendina nella Home Page, dalla quale si possono modificare il proprio nome utente, la propria foto profilo, e la propria password.

Calendario

La app fornisce agli utenti un calendario, con cui si possono esplorare gli eventi in base alla data invece che alla locazione geografica.

Amici

E' possibile aggiungere altri utenti alla propria lista amici. Quest'ultima è implementata automaticamente se si è effettuato l'accesso con Facebook.

VeriTour

Applicazione sviluppata da:

*Simone Filippi, Francesco Cerroni, Federico Bianchi, Matteo Antonelli,
Andrea Splendori, Luca Cardini.*

Tutor: Giacomo Cappucci

VeriTour

Cosa permette di fare l'applicazione?

- ▶ *Creare percorsi turistici personalizzati per i vari utenti;*
- ▶ *Possibilità di condividere i propri percorsi con gli altri utenti;*
- ▶ *Possibilità di avere una guida turistica virtuale durante un percorso;*
- ▶ *Possibilità di ricercare il percorso ideale in base alla tipologia delle attrazioni (monumenti, gastronomia...);*
- ▶ *Gestione automatica del percorso in base al traffico e al tempo di percorrenza (orari ed eventuali ritardi dei mezzi);*