

## **FINAL PROJECT**

### **Aplikasi Job Finder - Jobizz**

Dibuat untuk memenuhi Ujian Akhir Semester

Mata Kuliah IS534 – Mobile Application Development



#### **Disusun oleh:**

- |                      |   |                    |
|----------------------|---|--------------------|
| Marcello Roy         | - | (NIM: 00000072593) |
| Victor Georgius W.   | - | (NIM: 00000073402) |
| Andika Nur Fitrianto | - | (NIM: 00000073646) |
| Febianus Felix W.    | - | (NIM: 00000072737) |

**PROGRAM STUDI SISTEM INFORMASI  
FAKULTAS TEKNIK DAN INFORMATIKA  
UNIVERSITAS MULTIMEDIA NUSANTARA  
TANGERANG  
2022**

## KATA PENGANTAR

Puji syukur kami panjatkan ke Tuhan Yang Maha Esa yang telah memberikan rahmat dan karunia-Nya sehingga kami dapat menyelesaikan tugas akhir dengan judul Aplikasi Job Finder - Jobizz. Tujuan dari tugas akhir ini adalah untuk menganalisis dan merancang sebuah sistem informasi yang dapat membantu pengguna dalam mencari pekerjaan secara online. Sistem ini diharapkan dapat meningkatkan efisiensi dan kenyamanan pengguna dalam mencari pekerjaan yang sesuai dengan kemampuan mereka.

Tidak lupa, kami juga ingin mengucapkan terimakasih kepada:

1. Ibu Ririn Ikana Desanti, S.Kom., M.Kom. selaku Ketua Program Studi Universitas Multimedia Nusantara.
2. Bapak Agus Sulaiman selaku Dosen Pengampu Mata Kuliah Mobile Application Development sebagai pembimbing yang telah memberikan bimbingan, arahan, dan motivasi atas terselesaiya laporan ini.
3. Semua Anggota Kelompok yang sudah berpartisipasi dalam memberikan kontribusi pada proses penyelesaian laporan ini.
4. Teman–teman jurusan Sistem Informasi yang telah membantu dan memberikan semangat dalam penyusunan laporan ini,
5. Keluarga yang telah memberikan bantuan dukungan material dan moral, sehingga laporan ini dapat terselesaikan.
6. Serta kepada semua pihak yang telah memberikan motivasi serta semangat yang tidak dapat penulis sebutkan satu per satu.

Kami menyadari bahwa tugas akhir ini masih sangat jauh dari sempurna dan masih banyak kekurangan yang perlu diperbaiki. Oleh karena itu, kami sangat mengharapkan kritik dan saran yang membangun dari pembaca untuk perbaikan di masa depan.

Akhir kata, semoga tugas akhir ini dapat bermanfaat bagi kami, juga bagi pembaca pada umumnya. Kami juga berharap semoga tulisan ini tidak hanya menjadi karya akademis, melainkan juga dapat menjadi kontribusi positif dalam mengatasi tantangan dan permasalahan yang dihadapi.

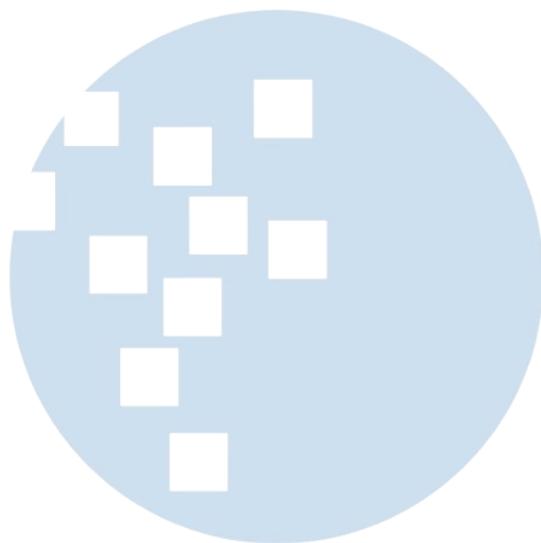
Tangerang, 05 Desember 2022

Kelompok ...

## DAFTAR ISI

<b>BAB I.....</b>	<b>2</b>
1.1 Background.....	3
1.2 Problem Boundaries.....	3
1.3 Research Objectives.....	3
<b>BAB II.....</b>	<b>4</b>
2.1 Business Procces.....	4
2.2 Flowchart.....	4
2.3 ERD Diagram.....	5
2.4 Application Layout.....	6
2.4.1 Splash Screen Layout.....	6
2.4.2 Splash Screen Layout.....	7
2.4.3 Home Page Layout.....	8
2.4.4 Home Page Sidebar.....	9
2.4.5 Personal Information.....	10
2.4.6 Application.....	11
2.4.7 Proposal.....	12
2.4.8 Resume.....	13
2.4.9 Portofolio.....	14
2.4.10 Cover Letter.....	15
2.4.11 Setting.....	16
2.4.12 Message.....	17
2.4.13 Bookmark.....	18
2.4.14 Category.....	19
2.5 Application Program.....	20
2.5.1 Splash Screen.....	20
2.5.2 Home Page.....	21
2.5.3 Personal Informaton.....	21
2.5.4 AppsFragment.....	22
2.5.5 Proposal.....	22
2.5.6 Resume.....	22
2.5.6 Portofolio.....	23
2.5.7 Cover Letter.....	23
2.5.8 Setting.....	24
2.5.9 Message.....	25
2.5.10 Bookmark.....	25
2.5.11 Category.....	26
2.5.12 SQLite.....	26
2.5.13 DescFragment.....	27
2.5.14 DescriptionFragment.....	28
2.5.15 Detail Jobs.....	29
2.5.16 Login.....	30
2.5.17 Lupa pass.....	31
2.5.18 MapsFragment.....	32

2.5.19 PersinfoFragment.....	32
2.5.20 Register.....	33
2.5.21 RequirementFragment.....	34
2.5.22 ReviewFragment.....	35
<b>BAB III.....</b>	<b>36</b>



**UMN**  
UNIVERSITAS  
MULTIMEDIA  
NUSANTARA

## **BAB I**

### **INTRODUCTION**

#### **1.1 Background**

In the rapidly evolving digital era, job search has become increasingly important and complex. Currently, many platforms such as LinkedIn have assisted individuals in connecting with other professionals, but there are still challenges in the job application process. Therefore, we introduce the "Jobizz" app - a new innovation that combines unique features to help users search for jobs more easily and efficiently.

One of the key features that sets "Jobizz" apart is the ability for users to easily apply for jobs at the companies they desire. We provide an intuitive interface that allows users to search for jobs based on specific criteria such as industry, location, or skills they possess. With a single click, users can submit their applications directly to the desired companies.

Furthermore, we also introduce a unique feature that allows teams from companies to assess whether applicants meet their criteria. This is a cutting-edge step in the recruitment process that enables companies to quickly and efficiently screen applicants who are most suitable for the positions they offer. This evaluation may include skill tests, short interviews through the platform, or even sophisticated algorithm-based assessments.

The "Jobizz" app creates a beneficial environment for both job seekers and companies. Job seekers gain easy access to various job opportunities, while companies can quickly find and assess potential candidates. We are confident that "Jobizz" will elevate the job search and recruitment experience to a higher level, bridging the gap between job seekers and companies, and enabling better career growth for all parties involved.

#### **1.2 Problem Boundaries**

1. How to improve efficiency and convenience in the job search process for job seekers in the midst of the continually evolving digital era?
2. What challenges do job seekers still face in applying for jobs, especially through digital platforms like LinkedIn, and how can "Jobizz" address these challenges?
3. How to enhance efficiency in the company recruitment process using "Jobizz's" unique features that enable rapid evaluation of the criteria sought by the company?

#### **1.3 Research Objectives**

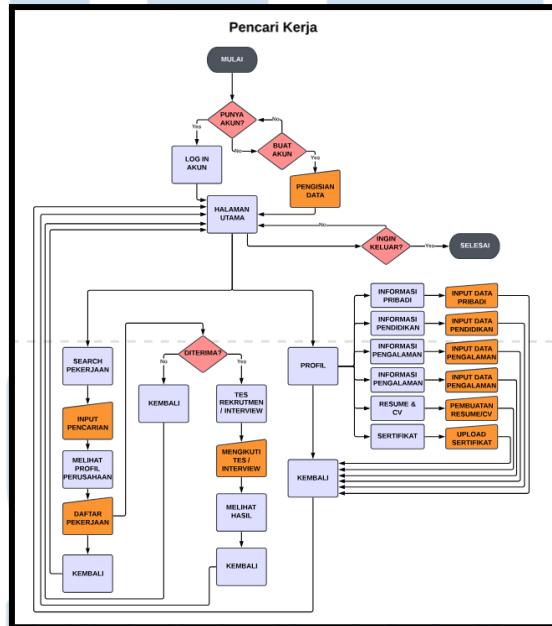
1. Improving the ease and speed of job searching for users by providing an intuitive interface and the ability to apply to desired companies with a single click.
2. Addressing challenges faced by job seekers by offering features that combine speed and accuracy in matching applicant qualifications with company requirements.
3. Enhancing efficiency in the company recruitment process by allowing recruitment teams to conduct rapid evaluations of applicants using various methods, including skill tests and brief interviews through the "Jobizz" platform.

## BAB II

### 2.1 Business Proces

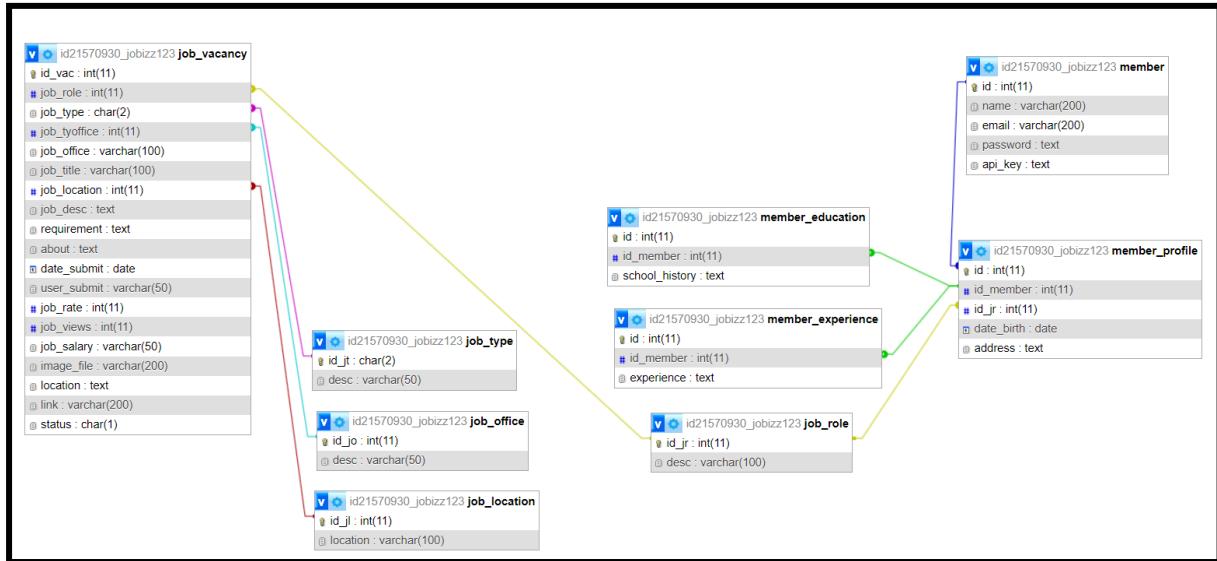
1. **Account Registration:** Job seekers must register an account first to use the application.
2. **Create Profile:** Job seekers need to create a profile containing personal information, education, work experience, and skills.
3. **Job Search:** Job seekers can search for jobs based on keywords, location, job type, and other criteria.
4. **View Company Profiles:** Job seekers can view company profiles to learn more about the respective companies.
5. **Job Application:** Job seekers can apply for desired positions by submitting their resume and cover letter.
6. **Take Tests:** Job seekers have the option to take tests to enhance their competencies.
7. **View Test Results:** Job seekers can view the results of the tests they have taken.

### 2.2 Flowchart



The picture above is the user flow chart of the application we created, at first, the user is required to log in if they already have an account, and register if they don't have an account, if they have logged in / registered, then the user will enter the main page of the application. On the main page, users can perform a number of interactions such as starting to search for jobs, opening the user's profile, and also logging out of the application. In the job search interaction, users can input the type of job they want or other inputs, then, users can view the job description, company description, and register for the job. After registering, the user will wait for a response from the company within a certain period of time about being accepted / not, when accepted, the user will take a test / interview according to the description given, after that, the user can see the test / interview results in the application. In the user profile, there is a number of information that is useful for applying to a job, such as education history, work experience that has been done, CV, certificates / achievements that have been obtained, and other personal data.

## 2.3 ERD Diagram



In the above relational diagram, there are 9 types of data present in the database used for this application:

1. **member**: Contains user data for logging into the application, such as email, password, and others. This data has a one-to-one relationship with the second set of data, **member\_profile**.
2. **member\_profile**: In addition to user name and email, this data contains more detailed user information such as member ID, job ID, date of birth, and address. This data has a one-to-many relationship with **member\_education**, **member\_experiences**, and **job\_role**.
3. **member\_education**: Contains information about the education pursued by the user.
4. **member\_experiences**: Contains information about the user's work experiences.
5. **job\_role**: Contains information about the user's job roles. This data has a one-to-many relationship with **job\_vacancy**.
6. **job\_vacancy**: Contains information about the jobs provided by companies. Within this data, there is information about the offered job (job type, job location, offered salary, etc.). This data has a many-to-one relationship with **job\_type**, **job\_office**, and **job\_location**.
7. **job\_type**: Contains explanations about the types of jobs provided.
8. **job\_office**: Contains the main office location of the job provider.
9. **job\_location**: Contains the location of the provided jobs.

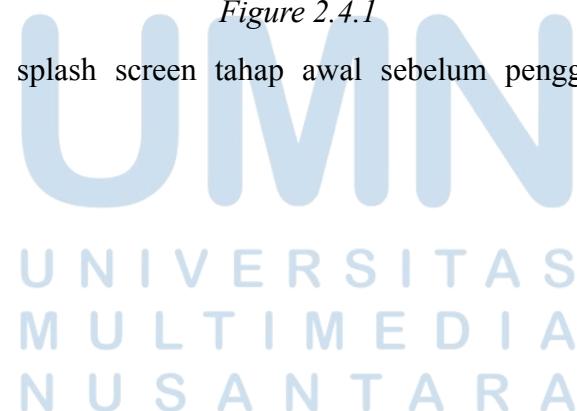
## 2.4 Application Layout

### 2.4.1 Splash Screen Layout



*Figure 2.4.1*

Layout di atas adalah splash screen tahap awal sebelum pengguna masuk ke halaman login/register.



## 2.4.2 Splash Screen Layout

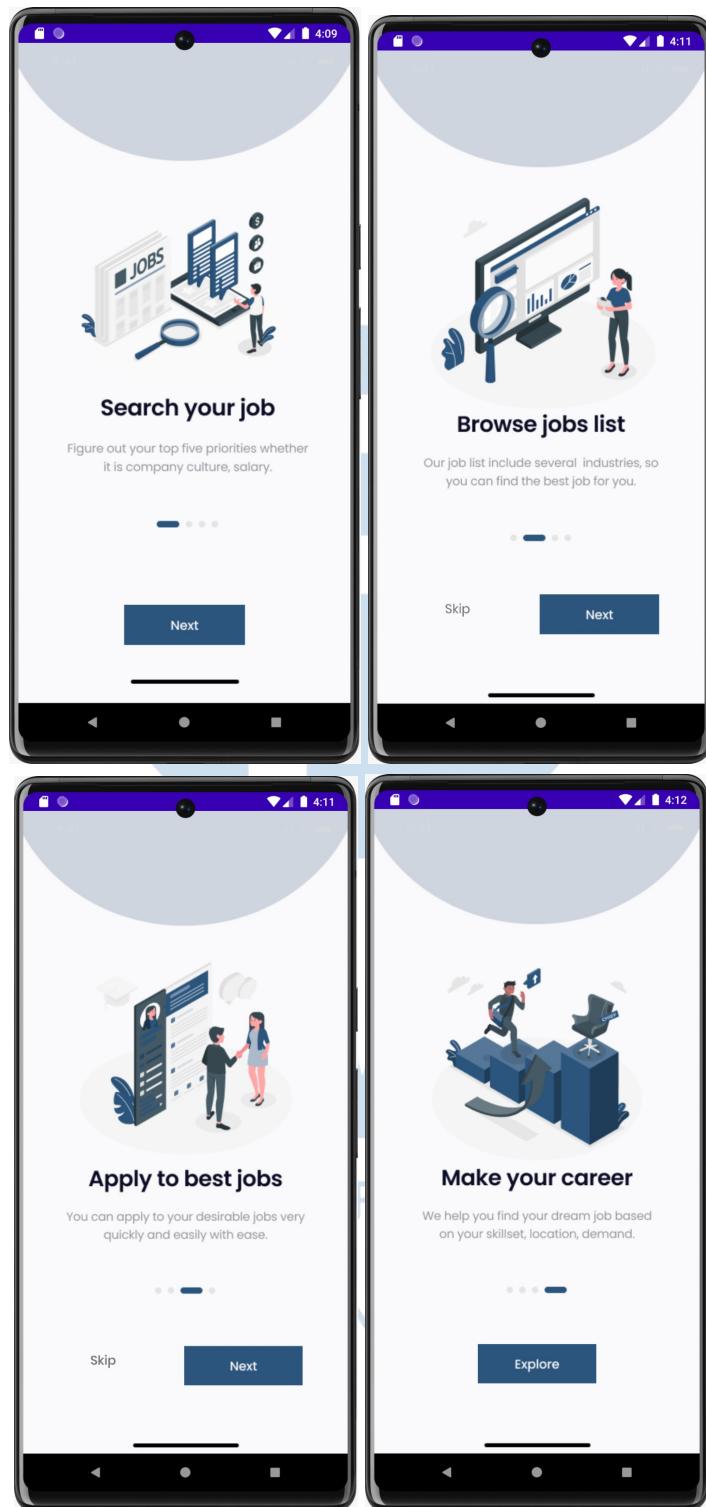


Figure 2.4.2

Layout di atas adalah 4 halaman pertama dari aplikasi Jobizz yang digunakan untuk splash screen. Jika tidak ingin melewati 4 splash screen tersebut, pengguna bisa langsung mengklik tombol “Skip”.

### 2.4.3 Home Page Layout

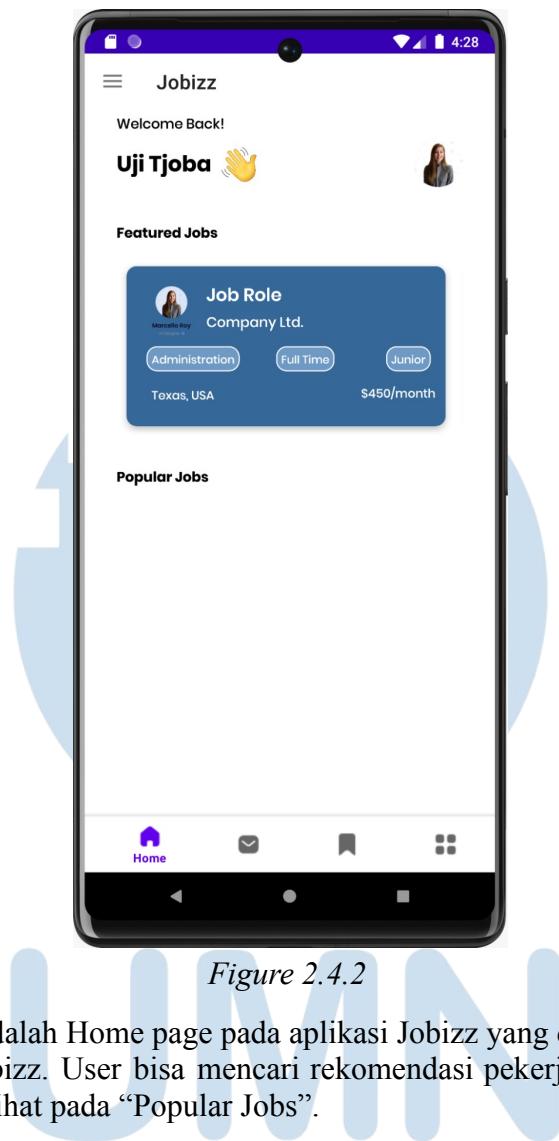


Figure 2.4.2

Layout di atas adalah Home page pada aplikasi Jobizz yang digunakan untuk halaman utama pada aplikasi Jobizz. User bisa mencari rekomendasi pekerjaan yang sedang populer saat ini dengan cara melihat pada “Popular Jobs”.

UNIVERSITAS  
MULTIMEDIA  
NUSANTARA

#### 2.4.4 Home Page Sidebar

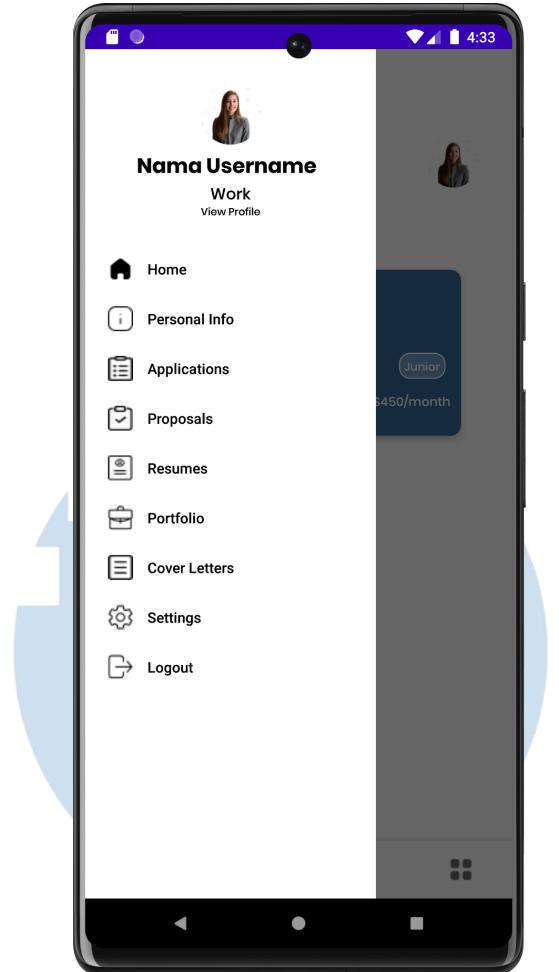


Figure 2.4.4

Layout di atas adalah Home page Sidebar pada aplikasi Jobizz yang digunakan untuk melihat fitur fitur pada aplikasi Jobizz. User bisa mengklik Home jika ingin kembali ke menu Halaman awal pada aplikasi, Jika user ingin melihat informasi tentang pribadi bisa ke menu Personal Info, ada Applications, Jika user ingin mengirim Proposal bisa klik menu Proposals , ada menu Resumes, Portfolio, Cover Letters, Settings dan yang terakhir yaitu menu Logout. Semua fitur itu bisa digunakan jika user membutuhkan fitur tersebut.

MULTIMEDIA  
NUSANTARA

## 2.4.5 Personal Information

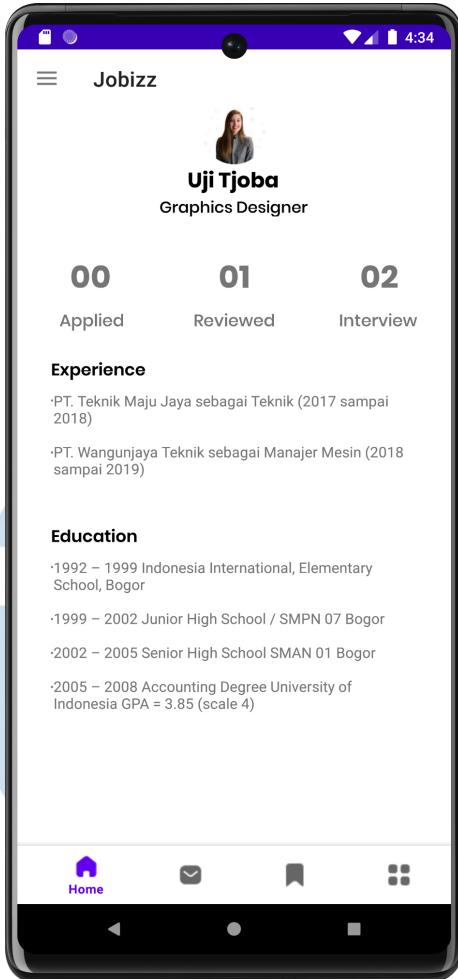
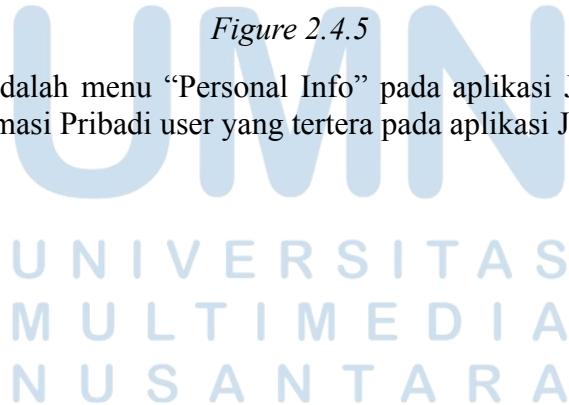
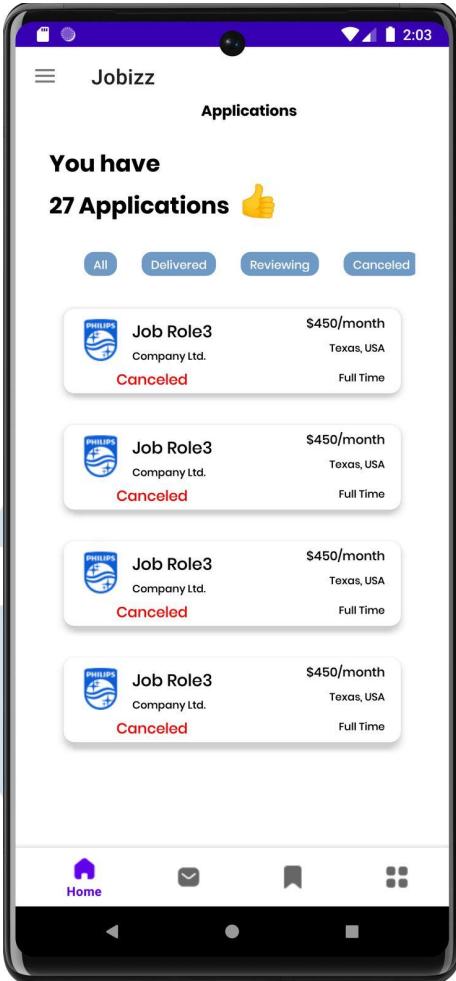


Figure 2.4.5

Layout di atas adalah menu “Personal Info” pada aplikasi Jobizz yang berguna jika user ingin melihat Informasi Pribadi user yang tertera pada aplikasi Jobizz



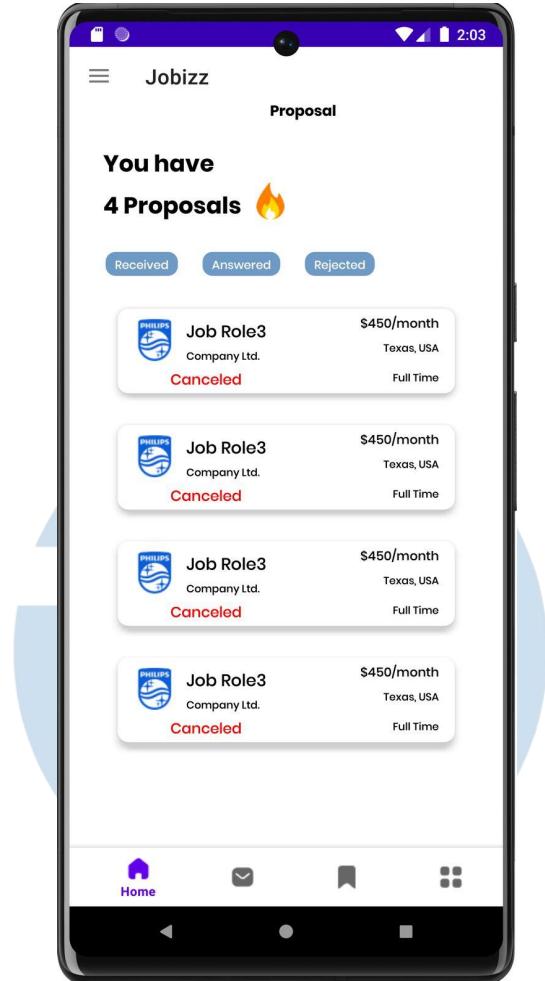
## 2.4.6 Application



Halaman ini merupakan halaman yang berisi tentang pekerjaan yang di daftar oleh pengguna aplikasi, pada halaman ini lamaran pekerjaan yang dilakukan memiliki 3 jenis status, yaitu delivered, yang berarti lamaran pekerjaan telah terkirim ke perusahaan yang dituju, reviewing yang berarti perusahaan tersebut sudah menerima lamaran yang dikirim dan sedang dalam proses peninjauan, dan canceled yang berarti lamaran tersebut dibatalkan / ditolak oleh perusahaan.

UNIVERSITAS  
MULTIMEDIA  
NUSANTARA

## 2.4.7 Proposal



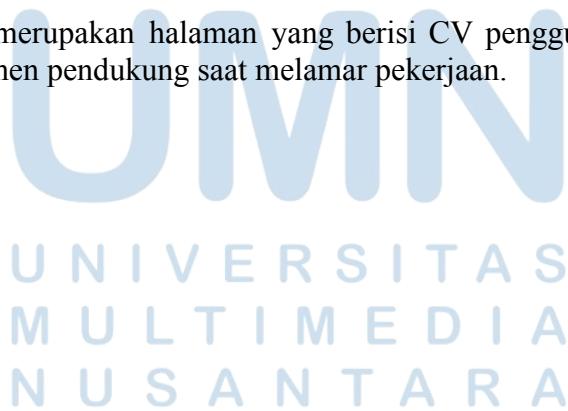
Halaman ini berisi daftar pekerjaan yang ditawarkan oleh perusahaan kepada para pengguna aplikasi.



## 2.4.8 Resume



Halaman resume ini, merupakan halaman yang berisi CV pengguna aplikasi Jobizz yang berfungsi sebagai dokumen pendukung saat melamar pekerjaan.



## 2.4.9 Portofolio



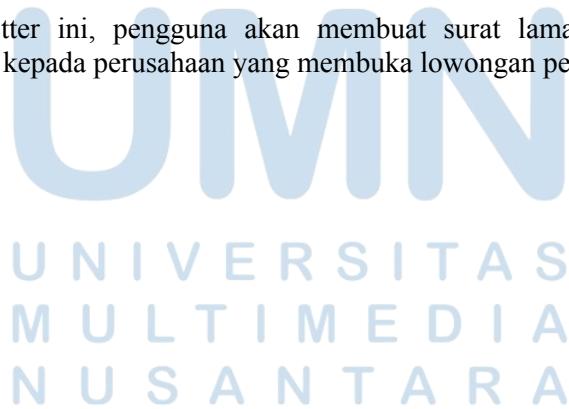
Halaman ini merupakan halaman yang ditujukan untuk pengguna yang akan mengisi semua pengalaman pekerjaan yang pernah dilakukan.



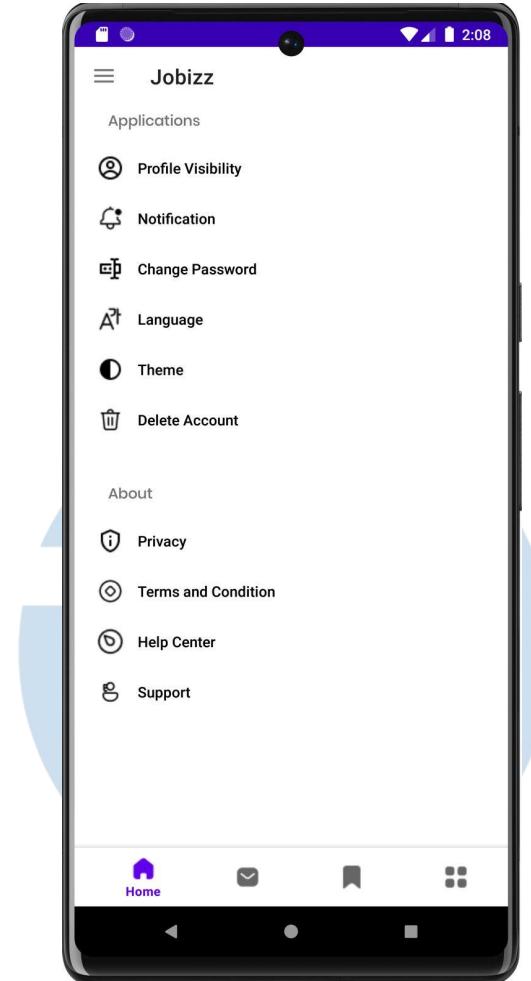
## 2.4.10 Cover Letter



Pada halaman Cover Letter ini, pengguna akan membuat surat lamaran pekerjaan yang akan digunakan dan dikirimkan kepada perusahaan yang membuka lowongan pekerjaan..



## 2.4.11 Setting



Layout di atas adalah menu “Settings” pada aplikasi Jobizz yang berguna jika user ingin melihat profile visibility, mengatur notification,, mengganti password, mengatur bahasa(Language), mensetting tampilan (Theme) dan bisa delete account juga. Jika user ada masalah bisa juga mengklik “Help Center” untuk menghubungi pegawai yang ada di Jobizz.

UNIVERSITAS  
MULTIMEDIA  
NUSANTARA

#### 2.4.12 Message

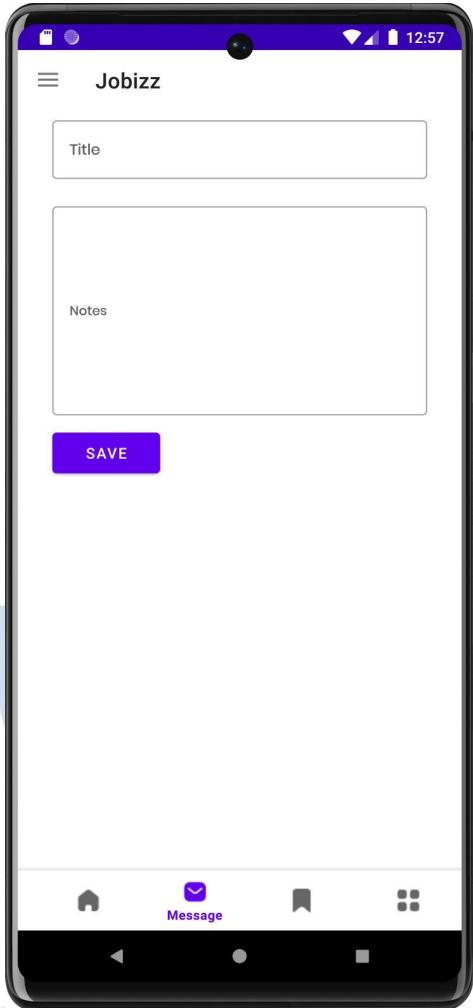
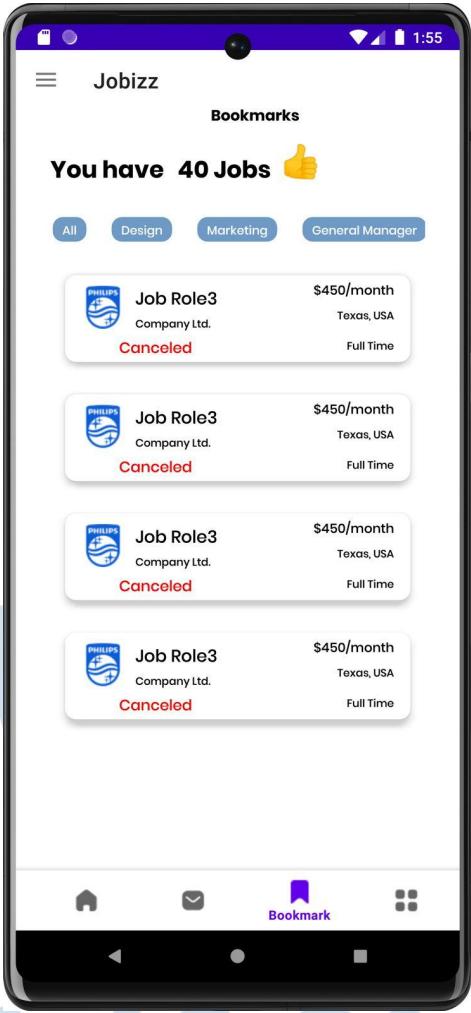


Figure 2.4.6

Layout di atas adalah menu “Message” pada aplikasi Jobizz yang berguna jika user ingin membuat sebuah catatan atau pengingat sebuah kegiatan yang berhubungan dengan pencarian kerja di dalam aplikasi Jobizz.

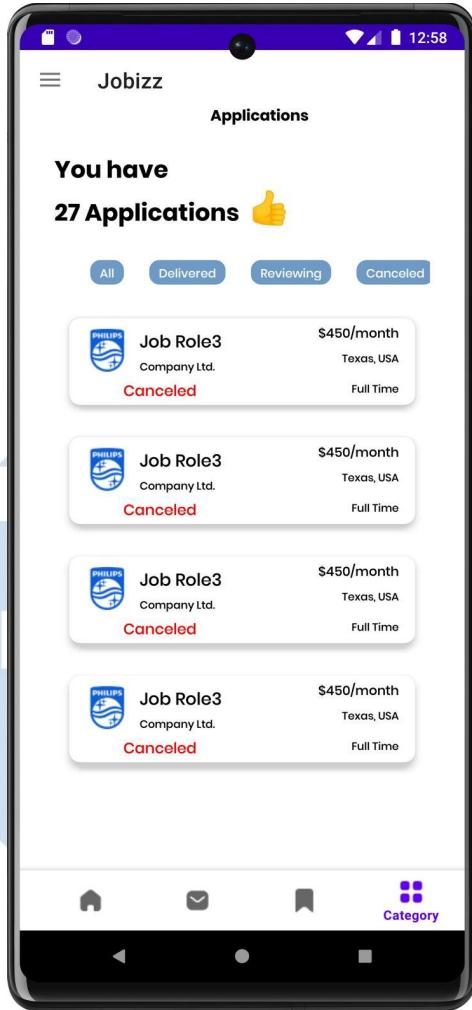
UNIVERSITAS  
MULTIMEDIA  
NUSANTARA

#### 2.4.13 Bookmark

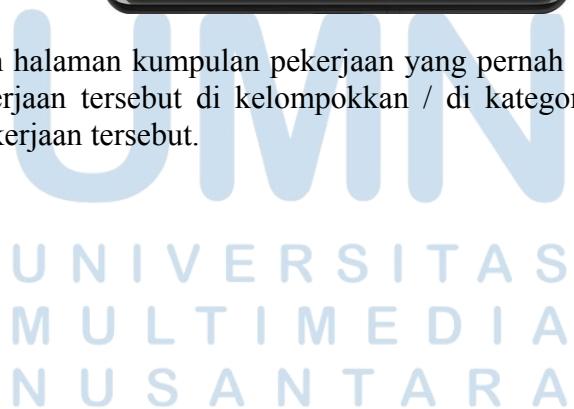


Layout di atas merupakan menu “bookmark” dimana segala jenis informasi / pekerjaan yang user simpan akan ditampilkan pada halaman ini.

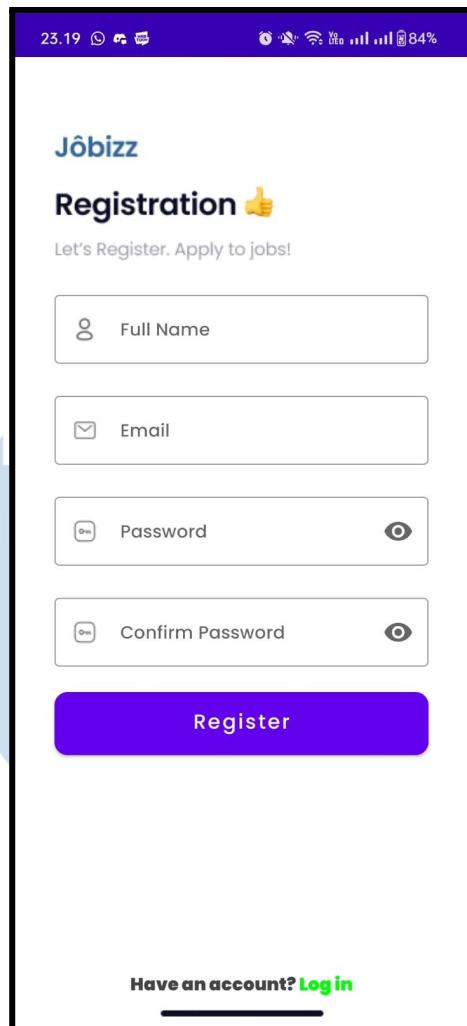
#### 2.4.14 Category



Halaman ini merupakan halaman kumpulan pekerjaan yang pernah di daftar oleh pengguna, dimana kumpulan pekerjaan tersebut di kelompokkan / di kategorikan berdasarkan status pemrosesan lamaran pekerjaan tersebut.



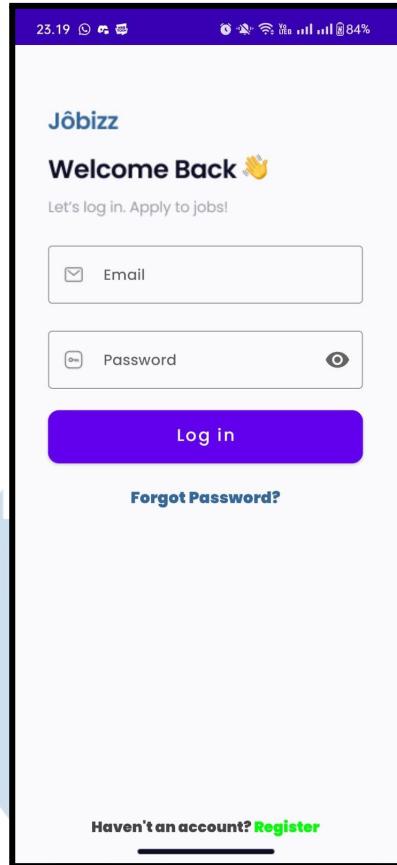
## 2.4.15 Register



Layout diatas merupakan desain dari menu Register, yang berguna jika user belum punya akun, maka menu ini akan memberikan form register untuk user yang belum mempunyai akun, tapi jika sudah mempunyai akun, maka user bisa langsung klik tombol Log In.

UNIVERSITAS  
MULTIMEDIA  
NUSANTARA

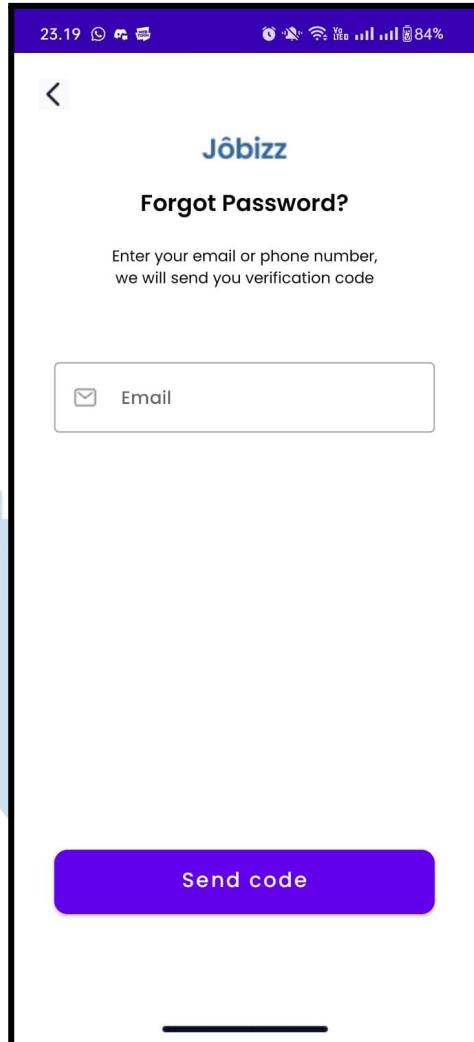
## 2.4.16 Log in



Program diatas memungkinkan pengguna masuk ke aplikasi, dengan memasukan email dan password, jika mereka belum terdaftar, maka mereka bisa menekan tombol register



#### 2.4.17 Forgot password



Layout ini berguna jika user lupa password, maka user bisa mengklik tombol forgot password, dan otomatis akan diarahkan ke menu tersebut

UNIVERSITAS  
MULTIMEDIA  
NUSANTARA

## 2.5 Application Program

### 2.5.1 Splash Screen

```
package com.example.jobizz;

import android.content.Intent;
import android.os.Bundle;
import android.os.Handler;

import androidx.appcompat.app.AppCompatActivity;

2 usages

public class splash extends AppCompatActivity {

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

        final Handler handler = new Handler();
        handler.postDelayed(new Runnable() {
            @Override
            public void run() {
                startActivity(new Intent(getApplicationContext(), sp1.class));
                finish();
            }
        }, delayMillis: 3000L);
    }
}
```

Secara umum, kode di atas digunakan untuk memberikan jeda waktu pada bagian yang kita inginkan, pada kode tersebut, bagian yang akan diberikan jeda yaitu halaman splash screen pertama (yang berisi logo aplikasi).



```
3 usages
public class Sp2 extends AppCompatActivity {

    2 usages
    ImageButton imageView;
    2 usages
    TextView textView;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_sp2);

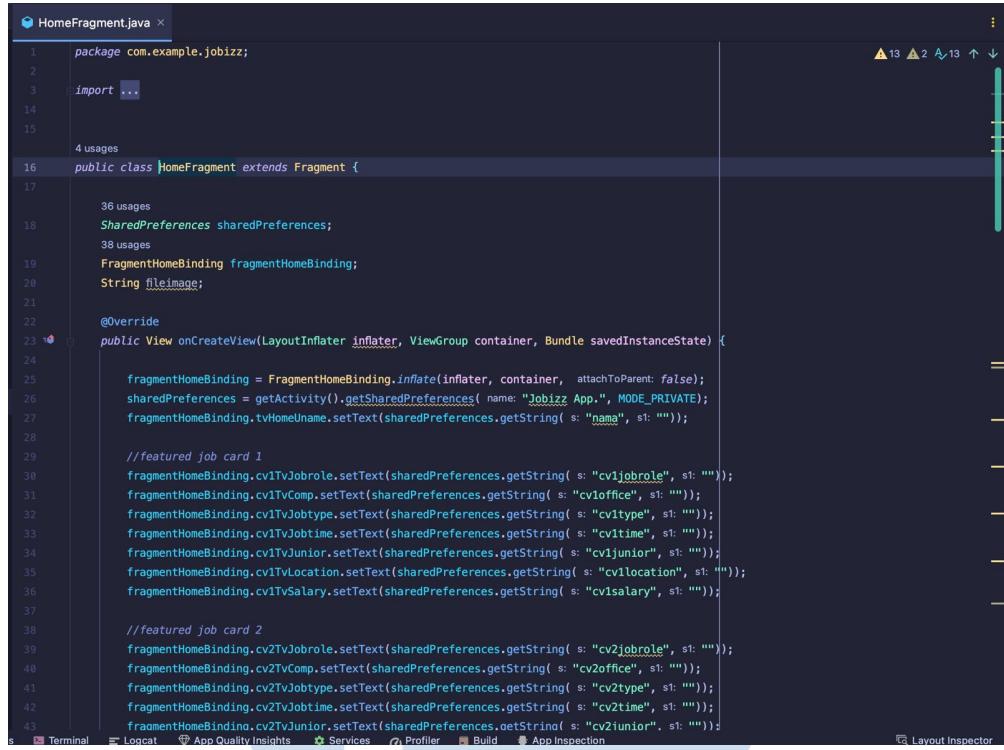
        imageView = (ImageButton) findViewById(R.id.sp2_next_btn);
        imageView.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                startActivity(new Intent(getApplicationContext(), sp3.class));
                finish();
            }
        });

        textView = (TextView) findViewById(R.id.tx_skip_sp2);
        textView.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                startActivity(new Intent(getApplicationContext(), login.class));
                finish();
            }
        });
    }
}
```

Pada kode di atas, 2 jenis deklarasi tombol, yaitu imagebutton dan textView, dimana kedua tombol tersebut memberikan pengguna kontrol untuk melanjutkan proses aplikasi ke halaman berikutnya dengan menggunakan fungsi onClickListener. Tombol imagebutton berfungsi untuk melanjutkan proses ke halaman selanjutnya (sp3) dengan menggunakan fungsi startActivity. Tombol textView memiliki fungsi yang sama, namun proses yang dilanjutkan yaitu langsung ke halaman login dengan menggunakan fungsi yang sama seperti tombol sebelumnya.

UNIVERSITAS  
MULTIMEDIA  
NUSANTARA

## 2.5.2 Home Page

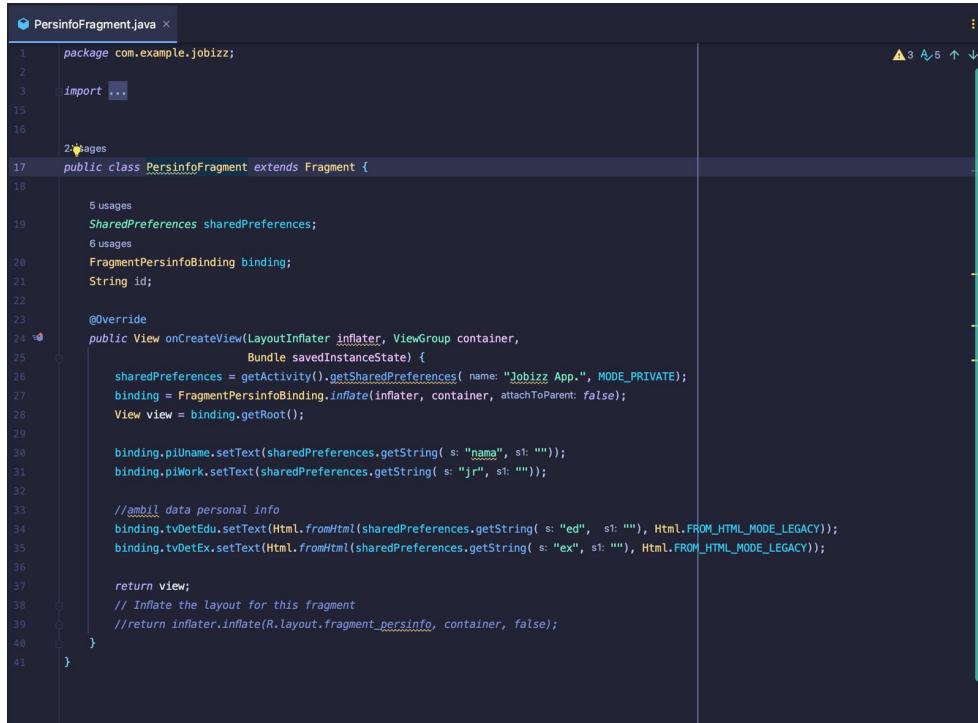


```
1 package com.example.jobizz;
2
3 import ...
4
5
6 4 usages
7 public class HomeFragment extends Fragment {
8
9     36 usages
10    SharedPreferences sharedPreferences;
11    38 usages
12    FragmentHomeBinding fragmentHomeBinding;
13    String fileImage;
14
15    @Override
16    public View onCreateView(LayoutInflater inflater, ViewGroup container, Bundle savedInstanceState) {
17
18        fragmentHomeBinding = FragmentHomeBinding.inflate(inflater, container, attachToParent: false);
19        sharedPreferences = getActivity().getSharedPreferences( name: "Jobizz App.", MODE_PRIVATE);
20        fragmentHomeBinding.tvHomeName.setText(sharedPreferences.getString( s: "name", s: ""));
21
22        //featured job card 1
23        fragmentHomeBinding.cv1TvJobrole.setText(sharedPreferences.getString( s: "cv1jobrole", s: ""));
24        fragmentHomeBinding.cv1TvComp.setText(sharedPreferences.getString( s: "cv1office", s: ""));
25        fragmentHomeBinding.cv1TvJobtype.setText(sharedPreferences.getString( s: "cv1type", s: ""));
26        fragmentHomeBinding.cv1TvJobtime.setText(sharedPreferences.getString( s: "cv1time", s: ""));
27        fragmentHomeBinding.cv1TvJunior.setText(sharedPreferences.getString( s: "cv1junior", s: ""));
28        fragmentHomeBinding.cv1TvLocation.setText(sharedPreferences.getString( s: "cv1location", s: ""));
29        fragmentHomeBinding.cv1TvSalary.setText(sharedPreferences.getString( s: "cv1salary", s: ""));
30
31        //featured job card 2
32        fragmentHomeBinding.cv2TvJobrole.setText(sharedPreferences.getString( s: "cv2jobrole", s: ""));
33        fragmentHomeBinding.cv2TvComp.setText(sharedPreferences.getString( s: "cv2office", s: ""));
34        fragmentHomeBinding.cv2TvJobtype.setText(sharedPreferences.getString( s: "cv2type", s: ""));
35        fragmentHomeBinding.cv2TvJobtime.setText(sharedPreferences.getString( s: "cv2time", s: ""));
36        fragmentHomeBinding.cv2TvJunior.setText(sharedPreferences.getString( s: "cv2junior", s: ""));
37
38
39
40
41
42
43 }
```

Cuplikan kode diatas bertujuan untuk mengisi card view dengan informasi-informasi yang sesuai, di bawah cuplikan tersebut terdapat kode yang digunakan untuk memberikan interaksi klik kepada cardView yang terdapat di dalam halaman tersebut



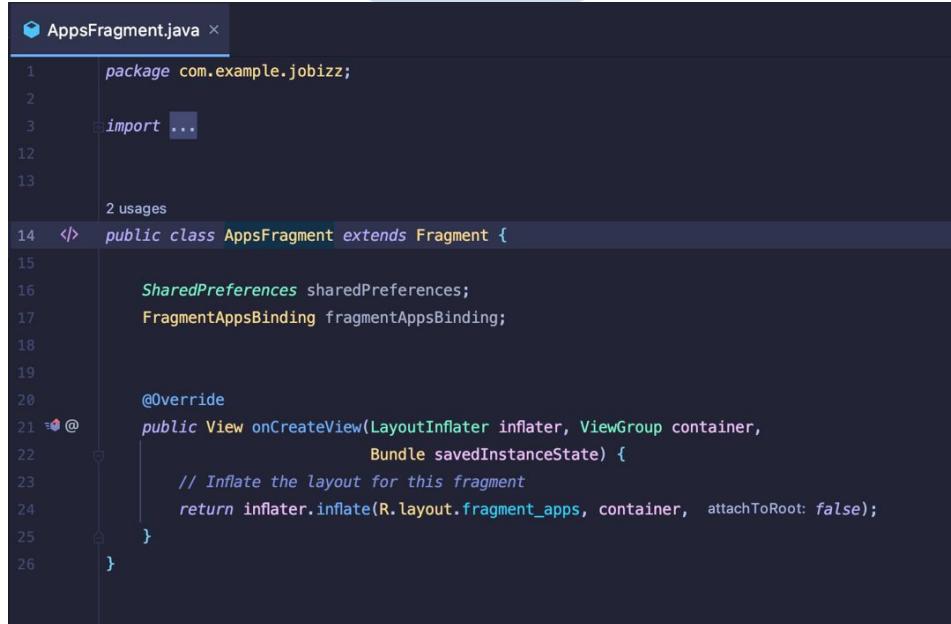
### 2.5.3 Personal Information



```
1 package com.example.jobizz;
2
3 import ...
4
5
6
7 public class PersinfoFragment extends Fragment {
8
9     5 usages
10    SharedPreferences sharedPreferences;
11
12    FragmentPersinfoBinding binding;
13    String id;
14
15    @Override
16    public View onCreateView(LayoutInflater inflater, ViewGroup container,
17                             Bundle savedInstanceState) {
18        sharedPreferences = getActivity().getSharedPreferences("Jobizz App.", MODE_PRIVATE);
19        binding = FragmentPersinfoBinding.inflate(inflater, container, attachToParent: false);
20        View view = binding.getRoot();
21
22        binding.pvName.setText(sharedPreferences.getString("nama", ""));
23        binding.pvWork.setText(sharedPreferences.getString("jr", ""));
24
25        //ambil data personal info
26        binding.tvDptEdu.setText(Html.fromHtml(sharedPreferences.getString("ed", "", Html.FROM_HTML_MODE_LEGACY)));
27        binding.tvDptEx.setText(Html.fromHtml(sharedPreferences.getString("ex", "", Html.FROM_HTML_MODE_LEGACY)));
28
29        return view;
30        // Inflate the layout for this fragment
31        //return inflater.inflate(R.layout.fragment_persinfo, container, false);
32    }
33
34    }
35
36
37    }
38
39    }
40
41 }
```

Kode di atas bertujuan untuk memberikan informasi ke dalam halaman personal information seperti informasi tentang job role pengguna, nama, info tentang edukasi dan lain-lain.

### 2.5.4 AppsFragment



```
1 package com.example.jobizz;
2
3 import ...
4
5
6
7 public class AppsFragment extends Fragment {
8
9     2 usages
10    SharedPreferences sharedPreferences;
11    FragmentAppsBinding fragmentAppsBinding;
12
13
14    @Override
15    public View onCreateView(LayoutInflater inflater, ViewGroup container,
16                            Bundle savedInstanceState) {
17        // Inflate the layout for this fragment
18        return inflater.inflate(R.layout.fragment_apps, container, attachToRoot: false);
19    }
20
21    }
22
23    }
24
25    }
26
27    }
```

Kode diatas memiliki tujuan untuk mengambil data dan mengisi informasi tersebut ke halaman application.

## 2.5.5 Proposal

```
package com.example.jobizz;

import ...

2 usages
public class ProposalFragment extends Fragment {

    @Override
    public View onCreateView(LayoutInflater inflater, ViewGroup container,
                           Bundle savedInstanceState) {
        // Inflate the layout for this fragment
        return inflater.inflate(R.layout.fragment_proposal, container, attachToRoot: false);
    }
}
```

Sama seperti kode sebelumnya, tujuan nya yaitu mengambil data yang nanti nya akan digunakan untuk mengisi halaman proposal

## 2.5.6 Resume

```
package com.example.jobizz;

import ...

2 usages
public class ResumeFragment extends Fragment {

    @Override
    public View onCreateView(LayoutInflater inflater, ViewGroup container,
                           Bundle savedInstanceState) {
        // Inflate the layout for this fragment
        return inflater.inflate(R.layout.fragment_resume, container, attachToRoot: false);
    }
}
```

Begitu juga dengan kode berikut, tujuannya untuk mengambil data dan menampilkannya di halaman resume.

UNIVERSITAS  
MULTIMEDIA  
NUSANTARA

## 2.5.6 Portofolio

The screenshot shows the `PortfolioFragment.java` file in the Android Studio code editor. The code defines a fragment that inflates a layout with a RecyclerView and populates it with 10 items. Several inspection annotations are present:

- Line 17: `2 usages`
- Line 17: `public class PortfolioFragment extends Fragment {` has a yellow exclamation mark icon.
- Line 26: `10` is highlighted in red, indicating a potential issue.
- Line 31: A yellow double underline is under `// Inflate the layout for this fragment`.
- Line 38: A yellow double underline is under `linearLayoutManager = new LinearLayoutManager(this, LinearLayoutManager.VERTICAL, false);`.
- Line 39: A yellow double underline is under `recyclerView.setLayoutManager(linearLayoutManager);`.
- Line 41: A yellow double underline is under `viewAdapter = new ViewAdapter(this, listData);`.
- Line 42: A yellow double underline is under `recyclerView.setAdapter(viewAdapter);`.
- Line 43: A yellow double underline is under `viewAdapter.notifyDataSetChanged();`.

Memiliki fungsi yang sama dengan kode lainnya, kode tersebut digunakan untuk mengisi halaman portofolio dengan informasi-informasi yang sesuai.

## **2.5.7 Cover Letter**

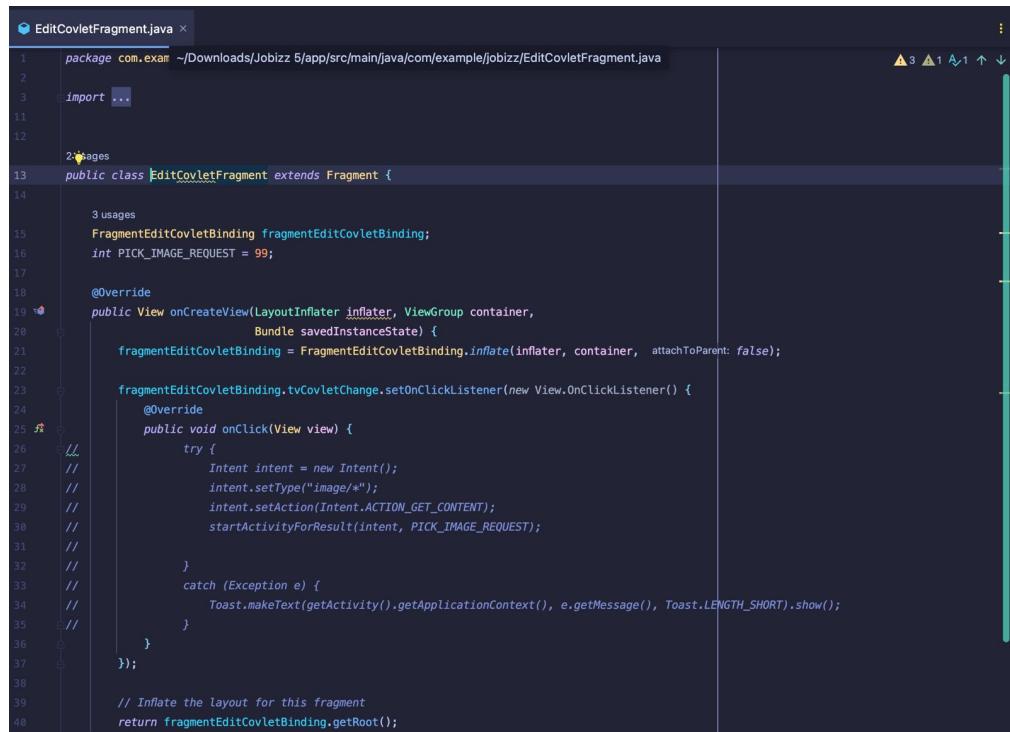
```
❶ CovleettFragment.java x
  package com.example -/Downloads/Jobizz 5/app/src/main/java/com/example/jobizz/CovleettFragment.java
  import ...
  11
  12
  2 usages
  13 public class CovleettFragment extends Fragment {
  14
  15     3 usages
  16     FragmentCovleettBinding fragmentCovleettBinding;
  17
  18     @Override
  19     public View onCreateView(LayoutInflater inflater, ViewGroup container,
  20                             Bundle savedInstanceState) {
  21
  22         fragmentCovleettBinding = FragmentCovleettBinding.inflate(inflater, container, attachToParent: false);
  23
  24         fragmentCovleettBinding.tvCovleettEdit.setOnClickListener(new View.OnClickListener() {
  25             @Override
  26             public void onClick(View view) {
  27                 getActivity().getSupportFragmentManager().beginTransaction().replace(R.id.frame_home, new EditCovleettFragment()).commit();
  28                 //fragmentCovleettBinding.tv
  29                 //Toast.makeText(getActivity(), "response", Toast.LENGTH_SHORT).show();
  30             }
  31         });
  32
  33         // Inflate the layout for this fragment
  34         return fragmentCovleettBinding.getRoot();
  35     }
  36 }
```

Code diatas adalah Cover Letter Fragment di atas berguna untuk menampilkan halaman yang memungkinkan pengguna untuk melihat dan mengedit informasi tentang Cover letter. Kode dimulai dengan mendeklarasikan kelas CovlettFragment yang ngasih dari kelas Fragment. Kelas Fragment adalah kelas dasar untuk semua fragment di Android.

Kemudian, kode mendeklarasikan variabel fragmentCovlettBinding dari tipe FragmentCovlettBinding. FragmentCovlettBinding adalah kelas yang mengikat kode Java dengan layout XML untuk fragment.

Fungsi onCreateView() bertanggung jawab untuk membuat tampilan fragment. Fungsi ini pertama-tama menggunakan metode inflate() untuk membuat objek FragmentCovlettBinding dari layout XML. Kemudian, fungsi ini menetapkan tampilan fragmentCovlettBinding ke tampilan fragment.

Fungsi onViewCreated() dipanggil setelah tampilan fragment dibuat. Fungsi ini biasanya digunakan untuk mengatur tampilan fragment.



```
1 package com.example.jobizz_5/app/src/main/java/com/example/jobizz/EditCovlettFragment.java
2
3 import ...
4
5
6 public class EditCovlettFragment extends Fragment {
7
8     3 usages
9     FragmentEditCovlettBinding fragmentEditCovlettBinding;
10    int PICK_IMAGE_REQUEST = 99;
11
12    @Override
13    public View onCreateView(LayoutInflater inflater, ViewGroup container,
14                            Bundle savedInstanceState) {
15        fragmentEditCovlettBinding = FragmentEditCovlettBinding.inflate(inflater, container, attachToParent: false);
16
17        fragmentEditCovlettBinding.tvCovlettChange.setOnClickListener(new View.OnClickListener() {
18            @Override
19            public void onClick(View view) {
20                try {
21                    Intent intent = new Intent();
22                    intent.setType("image/*");
23                    intent.setAction(Intent.ACTION_GET_CONTENT);
24                    startActivityForResult(intent, PICK_IMAGE_REQUEST);
25                }
26                catch (Exception e) {
27                    Toast.makeText(getActivity().getApplicationContext(), e.getMessage(), Toast.LENGTH_SHORT).show();
28                }
29            }
30        });
31
32    }
33
34    // Inflate the layout for this fragment
35    return fragmentEditCovlettBinding.getRoot();
36
37
38
39
40}
```

Bagian kode diatas masih termasuk ke bagian Cover Letter, lebih spesifik, kode diatas merupakan bagian kode untuk memberikan fungsi untuk mengedit cover letter yang dibuat, pengeditan dapat dilakukan karena adanya penggunaan fungsi click listener.

UNIVERSITAS  
MULTIMEDIA  
NUSANTARA

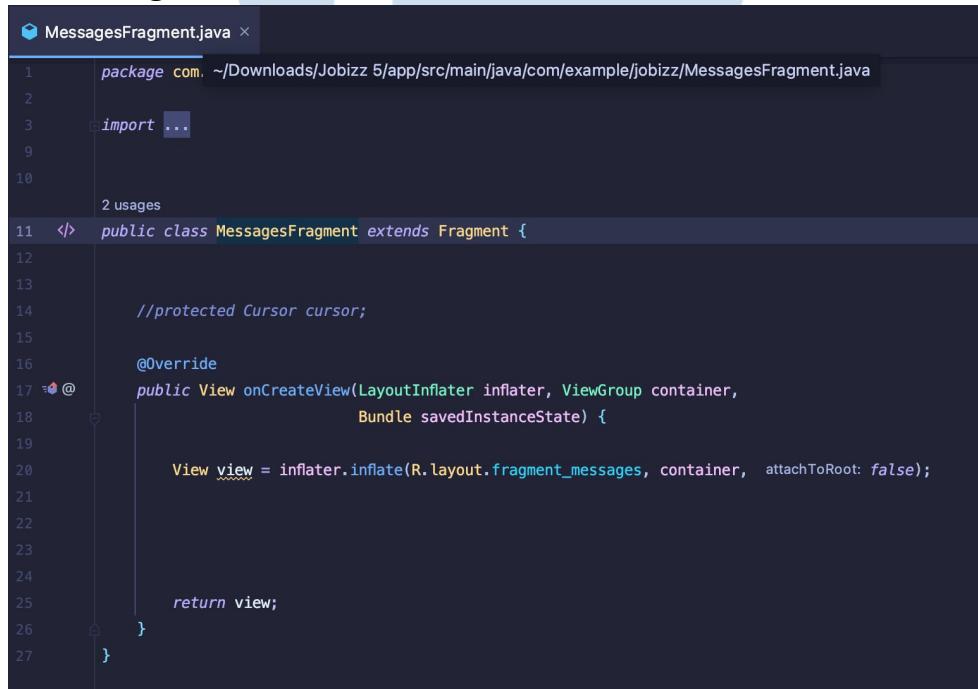
## 2.5.8 Setting



```
SettingsFragment.java
1 package com.example.jobizz;
2
3 import ...
4
5
6
7
8
9
10
11 </> public class SettingsFragment extends Fragment {
12
13
14     @Override
15     public View onCreateView(LayoutInflater inflater, ViewGroup container,
16                             Bundle savedInstanceState) {
17         // Inflate the layout for this fragment
18         return inflater.inflate(R.layout.fragment_settings, container, attachToRoot: false);
19     }
20 }
```

Code diatas adalah code yang memberikan fungsi untuk bagian menu setting, yang jika user ingin mengubah settingan dalam aplikasi Jobizz, code inilah yang membantu dalam process user

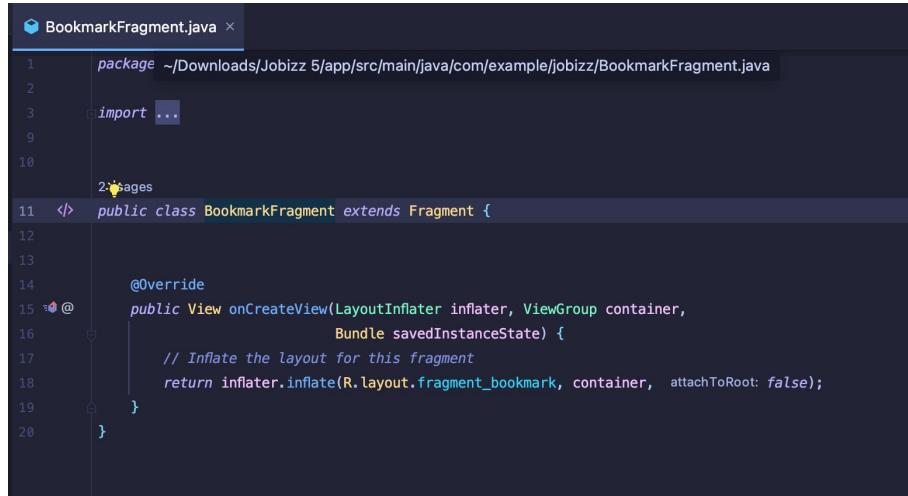
## 2.5.9 Message



```
MessagesFragment.java
1 package com.~/Downloads/Jobizz 5/app/src/main/java/com/example/jobizz/MessagesFragment.java
2
3 import ...
4
5
6
7
8
9
10
11 </> public class MessagesFragment extends Fragment {
12
13
14     //protected Cursor cursor;
15
16     @Override
17     public View onCreateView(LayoutInflater inflater, ViewGroup container,
18                             Bundle savedInstanceState) {
19
20         View view = inflater.inflate(R.layout.fragment_messages, container, attachToRoot: false);
21
22
23
24
25         return view;
26     }
27 }
```

Code diatas adalah code untuk Message fragment, yang berguna untuk menampilkan daftar notes pada menu message.

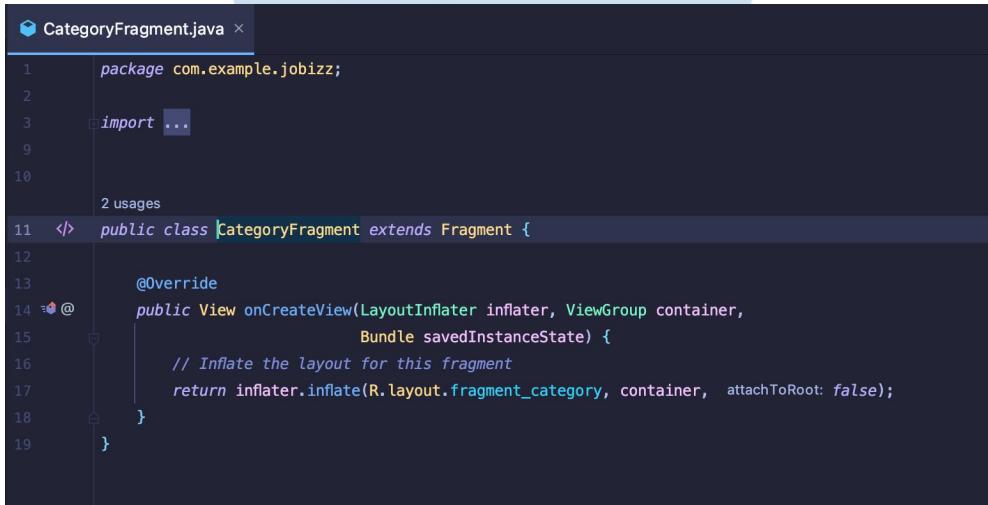
## 2.5.10 Bookmark



```
BookmarkFragment.java
1 package ~/Downloads/Jobizz 5/app/src/main/java/com/example/jobizz/BookmarkFragment.java
2
3 import ...
4
5
6
7
8
9
10
11 <|-- 2 usages
12
13
14 @Override
15 * @*
16 public View onCreateView(LayoutInflater inflater, ViewGroup container,
17                         Bundle savedInstanceState) {
18     // Inflate the layout for this fragment
19     return inflater.inflate(R.layout.fragment_bookmark, container, attachToRoot: false);
20 }
```

Kode diatas seharusnya berfungsi utnuk menyimpan pekerjaan yang di tandai oleh pengguna, dimana nanti nya akan masuk ke dalam database dan akan dibuat pengelompokan jenis pekerjaan juga.

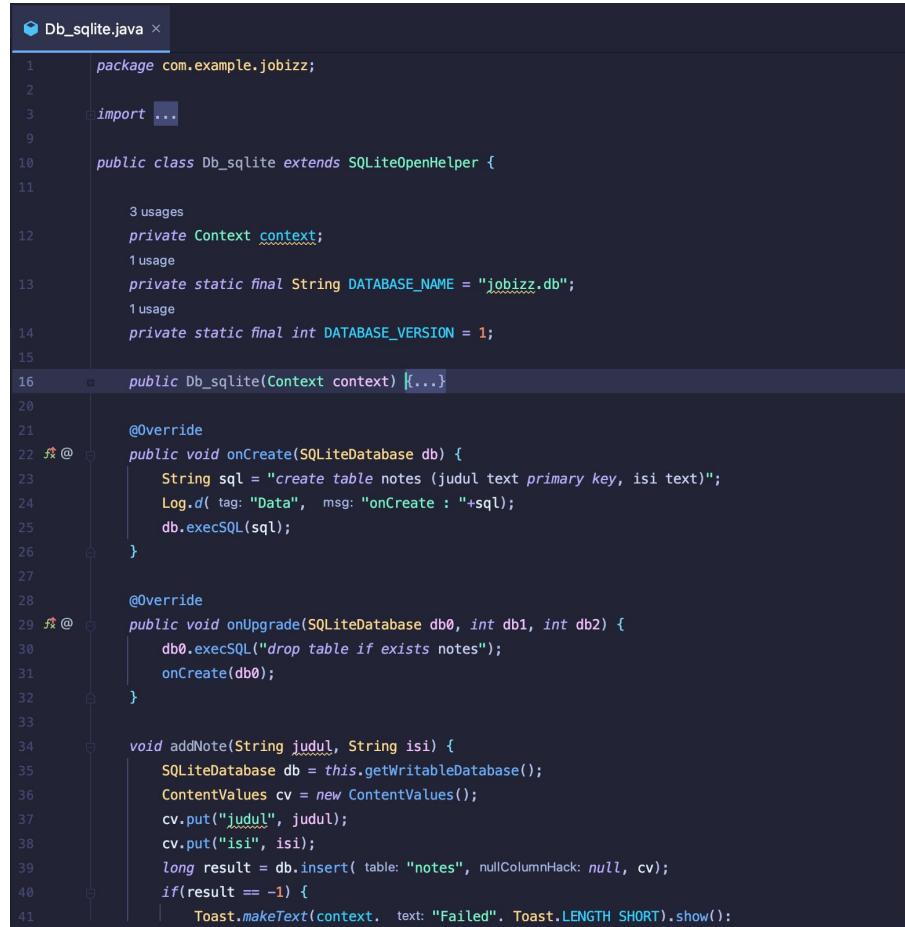
## 2.5.11 Category



```
CategoryFragment.java
1 package com.example.jobizz;
2
3 import ...
4
5
6
7
8
9
10
11 <|-- 2 usages
12
13
14 @Override
15 * @*
16 public View onCreateView(LayoutInflater inflater, ViewGroup container,
17                         Bundle savedInstanceState) {
18     // Inflate the layout for this fragment
19     return inflater.inflate(R.layout.fragment_category, container, attachToRoot: false);
20 }
```

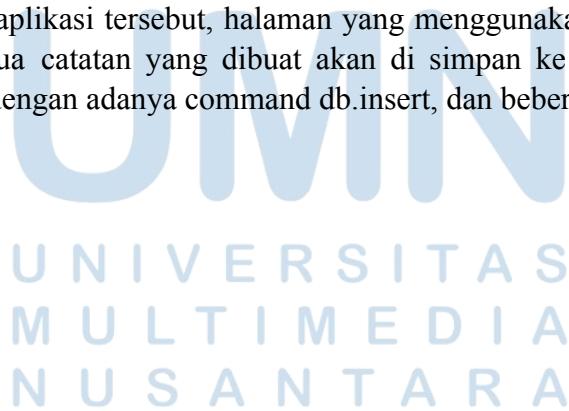
Kode di atas berfungsi untuk membuat sebuah halaman yang berisi kumpulan pekerjaan berdasarkan jenis pekerjaan nya.

## 2.5.12 SQLite

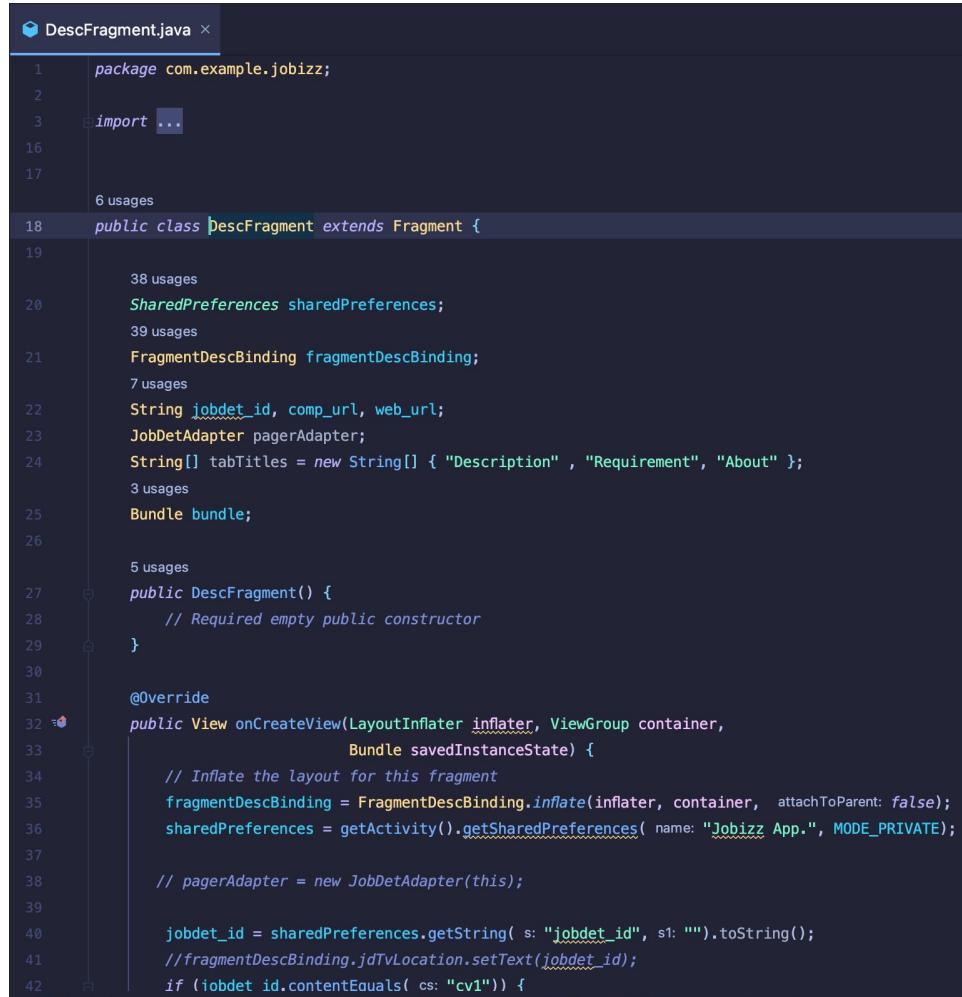


```
1 package com.example.jobizz;
2
3 import ...
4
5
6 public class Db_sqlite extends SQLiteOpenHelper {
7
8     private Context context;
9
10    private static final String DATABASE_NAME = "jobizz.db";
11
12    private static final int DATABASE_VERSION = 1;
13
14    public Db_sqlite(Context context) {...}
15
16    @Override
17    public void onCreate(SQLiteDatabase db) {
18        String sql = "create table notes (judul text primary key, isi text)";
19        Log.d( tag: "Data", msg: "onCreate : "+sql);
20        db.execSQL(sql);
21    }
22
23    @Override
24    public void onUpgrade(SQLiteDatabase db0, int db1, int db2) {
25        db0.execSQL("drop table if exists notes");
26        onCreate(db0);
27    }
28
29    void addNote(String judul, String isi) {
30        SQLiteDatabase db = this.getWritableDatabase();
31        ContentValues cv = new ContentValues();
32        cv.put("judul", judul);
33        cv.put("isi", isi);
34        long result = db.insert( table: "notes", nullColumnHack: null, cv);
35        if(result == -1) {
36            Toast.makeText(context, text: "Failed", Toast.LENGTH_SHORT).show();
37        }
38    }
39
40
41 }
```

Kode diatas merupakan kode SQLite, fungsi tersebut digunakan untuk menyimpan catatan yang pengguna buat di aplikasi tersebut, halaman yang menggunakan SQLite adalah halaman messages, dimana semua catatan yang dibuat akan disimpan ke dalam database. Fungsi tersebut dapat berjalan dengan adanya command db.insert, dan beberapa lainnya.



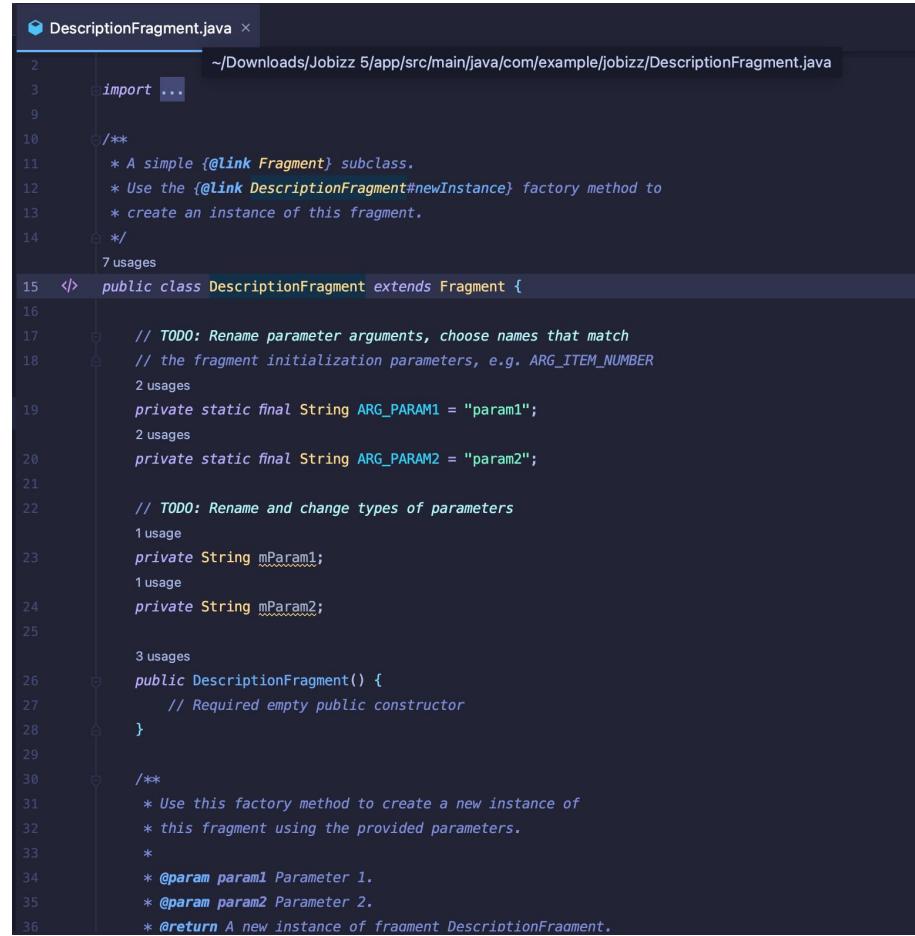
### 2.5.13 DescFragment



```
1 package com.example.jobizz;
2
3 import ...
4
5
6 usages
7
8 public class DescFragment extends Fragment {
9
10
11     38 usages
12     SharedPreferences sharedPreferences;
13
14     39 usages
15     FragmentDescBinding fragmentDescBinding;
16
17     7 usages
18     String jobdet_id, comp_url, web_url;
19     JobDetAdapter pagerAdapter;
20     String[] tabTitles = new String[] { "Description" , "Requirement" , "About" };
21
22     3 usages
23     Bundle bundle;
24
25
26     5 usages
27     public DescFragment() {
28         // Required empty public constructor
29     }
30
31     @Override
32     public View onCreateView(LayoutInflater inflater, ViewGroup container,
33                             Bundle savedInstanceState) {
34         // Inflate the layout for this fragment
35         fragmentDescBinding = FragmentDescBinding.inflate(inflater, container, attachToParent: false);
36         sharedPreferences = getActivity().getSharedPreferences( name: "Jobizz App.", MODE_PRIVATE);
37
38         // pagerAdapter = new JobDetAdapter(this);
39
40         jobdet_id = sharedPreferences.getString( s: "jobdet_id", s1: "").toString();
41         //fragmentDescBinding.jdtvLocation.setText(jobdet_id);
42         if (jobdet_id.contentEquals( cs: "cv1")) {
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
59
60
61
62
63
64
65
66
67
68
69
69
70
71
72
73
74
75
76
77
78
79
79
80
81
82
83
84
85
86
87
88
89
89
90
91
92
93
94
95
96
97
98
99
99
100
101
102
103
104
105
106
107
108
109
109
110
111
112
113
114
115
116
117
118
119
119
120
121
122
123
124
125
126
127
127
128
129
129
130
131
132
133
134
135
136
137
138
139
139
140
141
142
143
144
145
146
147
148
149
149
150
151
152
153
154
155
156
157
158
159
159
160
161
162
163
164
165
166
167
168
169
169
170
171
172
173
174
175
176
177
178
179
179
180
181
182
183
184
185
186
187
188
189
189
190
191
192
193
194
195
196
197
198
199
199
200
201
202
203
204
205
206
207
208
209
209
210
211
212
213
214
215
216
217
218
219
219
220
221
222
223
224
225
226
227
227
228
229
229
230
231
232
233
234
235
236
237
237
238
239
239
240
241
242
243
244
245
246
247
247
248
249
249
250
251
252
253
254
255
256
257
257
258
259
259
260
261
262
263
264
265
266
267
267
268
269
269
270
271
272
273
274
275
276
277
277
278
279
279
280
281
282
283
284
285
286
287
287
288
289
289
290
291
292
293
294
295
296
297
297
298
299
299
300
301
302
303
304
305
306
307
307
308
309
309
310
311
312
313
314
315
316
317
317
318
319
319
320
321
322
323
324
325
326
327
327
328
329
329
330
331
332
333
334
335
336
337
337
338
339
339
340
341
342
343
344
345
346
347
347
348
349
349
350
351
352
353
354
355
356
357
357
358
359
359
360
361
362
363
364
365
366
367
367
368
369
369
370
371
372
373
374
375
376
377
377
378
379
379
380
381
382
383
384
385
386
387
387
388
389
389
390
391
392
393
394
395
396
397
397
398
399
399
400
401
402
403
404
405
406
407
407
408
409
409
410
411
412
413
414
415
416
416
417
418
418
419
419
420
421
422
423
424
425
426
426
427
428
428
429
429
430
431
432
433
434
435
436
437
437
438
439
439
440
441
442
443
444
445
446
447
447
448
449
449
450
451
452
453
454
455
456
457
457
458
459
459
460
461
462
463
464
465
466
467
467
468
469
469
470
471
472
473
474
475
476
477
477
478
479
479
480
481
482
483
484
485
486
487
487
488
489
489
490
491
492
493
494
495
496
497
497
498
499
499
500
501
502
503
504
505
506
507
507
508
509
509
510
511
512
513
514
515
516
517
517
518
519
519
520
521
522
523
524
525
526
527
527
528
529
529
530
531
532
533
534
535
536
537
537
538
539
539
540
541
542
543
544
545
546
547
547
548
549
549
550
551
552
553
554
555
556
557
557
558
559
559
560
561
562
563
564
565
566
567
567
568
569
569
570
571
572
573
574
575
576
577
577
578
579
579
580
581
582
583
584
585
586
587
587
588
589
589
590
591
592
593
594
595
596
597
597
598
599
599
600
601
602
603
604
605
606
607
607
608
609
609
610
611
612
613
614
615
616
616
617
618
618
619
619
620
621
622
623
624
625
626
627
627
628
629
629
630
631
632
633
634
635
636
637
637
638
639
639
640
641
642
643
644
645
646
646
647
648
648
649
649
650
651
652
653
654
655
656
657
657
658
659
659
660
661
662
663
664
665
666
666
667
668
668
669
669
670
671
672
673
674
675
676
676
677
678
678
679
679
680
681
682
683
684
685
686
687
687
688
689
689
690
691
692
693
694
695
696
696
697
698
698
699
699
700
701
702
703
704
705
706
706
707
708
708
709
709
710
711
712
713
714
715
716
716
717
718
718
719
719
720
721
722
723
724
725
726
726
727
728
728
729
729
730
731
732
733
734
735
736
736
737
738
738
739
739
740
741
742
743
744
745
746
746
747
748
748
749
749
750
751
752
753
754
755
756
756
757
758
758
759
759
760
761
762
763
764
765
766
766
767
768
768
769
769
770
771
772
773
774
775
776
776
777
778
778
779
779
780
781
782
783
784
785
786
786
787
788
788
789
789
790
791
792
793
794
795
795
796
796
797
797
798
798
799
799
800
801
802
803
804
805
805
806
807
807
808
808
809
809
810
811
812
813
814
815
815
816
817
817
818
818
819
819
820
821
822
823
824
825
825
826
827
827
828
828
829
829
830
831
832
833
834
835
835
836
837
837
838
838
839
839
840
841
842
843
844
845
845
846
847
847
848
848
849
849
850
851
852
853
854
855
855
856
857
857
858
858
859
859
860
861
862
863
864
865
865
866
867
867
868
868
869
869
870
871
872
873
874
875
875
876
877
877
878
878
879
879
880
881
882
883
884
885
885
886
887
887
888
888
889
889
890
891
892
893
894
894
895
895
896
896
897
897
898
898
899
899
900
901
902
903
903
904
904
905
905
906
906
907
907
908
908
909
909
910
911
912
913
913
914
914
915
915
916
916
917
917
918
918
919
919
920
921
922
923
923
924
924
925
925
926
926
927
927
928
928
929
929
930
931
932
933
933
934
934
935
935
936
936
937
937
938
938
939
939
940
941
942
943
943
944
944
945
945
946
946
947
947
948
948
949
949
950
951
952
953
953
954
954
955
955
956
956
957
957
958
958
959
959
960
961
962
963
963
964
964
965
965
966
966
967
967
968
968
969
969
970
971
972
973
973
974
974
975
975
976
976
977
977
978
978
979
979
980
981
982
983
983
984
984
985
985
986
986
987
987
988
988
989
989
990
991
992
992
993
993
994
994
995
995
996
996
997
997
998
998
999
999
1000
1000
1001
1001
1002
1002
1003
1003
1004
1004
1005
1005
1006
1006
1007
1007
1008
1008
1009
1009
1010
1010
1011
1011
1012
1012
1013
1013
1014
1014
1015
1015
1016
1016
1017
1017
1018
1018
1019
1019
1020
1020
1021
1021
1022
1022
1023
1023
1024
1024
1025
1025
1026
1026
1027
1027
1028
1028
1029
1029
1030
1030
1031
1031
1032
1032
1033
1033
1034
1034
1035
1035
1036
1036
1037
1037
1038
1038
1039
1039
1040
1040
1041
1041
1042
1042
1043
1043
1044
1044
1045
1045
1046
1046
1047
1047
1048
1048
1049
1049
1050
1050
1051
1051
1052
1052
1053
1053
1054
1054
1055
1055
1056
1056
1057
1057
1058
1058
1059
1059
1060
1060
1061
1061
1062
1062
1063
1063
1064
1064
1065
1065
1066
1066
1067
1067
1068
1068
1069
1069
1070
1070
1071
1071
1072
1072
1073
1073
1074
1074
1075
1075
1076
1076
1077
1077
1078
1078
1079
1079
1080
1080
1081
1081
1082
1082
1083
1083
1084
1084
1085
1085
1086
1086
1087
1087
1088
1088
1089
1089
1090
1090
1091
1091
1092
1092
1093
1093
1094
1094
1095
1095
1096
1096
1097
1097
1098
1098
1099
1099
1100
1100
1101
1101
1102
1102
1103
1103
1104
1104
1105
1105
1106
1106
1107
1107
1108
1108
1109
1109
1110
1110
1111
1111
1112
1112
1113
1113
1114
1114
1115
1115
1116
1116
1117
1117
1118
1118
1119
1119
1120
1120
1121
1121
1122
1122
1123
1123
1124
1124
1125
1125
1126
1126
1127
1127
1128
1128
1129
1129
1130
1130
1131
1131
1132
1132
1133
1133
1134
1134
1135
1135
1136
1136
1137
1137
1138
1138
1139
1139
1140
1140
1141
1141
1142
1142
1143
1143
1144
1144
1145
1145
1146
1146
1147
1147
1148
1148
1149
1149
1150
1150
1151
1151
1152
1152
1153
1153
1154
1154
1155
1155
1156
1156
1157
1157
1158
1158
1159
1159
1160
1160
1161
1161
1162
1162
1163
1163
1164
1164
1165
1165
1166
1166
1167
1167
1168
1168
1169
1169
1170
1170
1171
1171
1172
1172
1173
1173
1174
1174
1175
1175
1176
1176
1177
1177
1178
1178
1179
1179
1180
1180
1181
1181
1182
1182
1183
1183
1184
1184
1185
1185
1186
1186
1187
1187
1188
1188
1189
1189
1190
1190
1191
1191
1192
1192
1193
1193
1194
1194
1195
1195
1196
1196
1197
1197
1198
1198
1199
1199
1200
1200
1201
1201
1202
1202
1203
1203
1204
1204
1205
1205
1206
1206
1207
1207
1208
1208
1209
1209
1210
1210
1211
1211
1212
1212
1213
1213
1214
1214
1215
1215
1216
1216
1217
1217
1218
1218
1219
1219
1220
1220
1221
1221
1222
1222
1223
1223
1224
1224
1225
1225
1226
1226
1227
1227
1228
1228
1229
1229
1230
1230
1231
1231
1232
1232
1233
1233
1234
1234
1235
1235
1236
1236
1237
1237
1238
1238
1239
1239
1240
1240
1241
1241
1242
1242
1243
1243
1244
1244
1245
1245
1246
1246
1247
1247
1248
1248
1249
1249
1250
1250
1251
1251
1252
1252
1253
1253
1254
1254
1255
1255
1256
1256
1257
1257
1258
1258
1259
1259
1260
1260
1261
1261
1262
1262
1263
1263
1264
1264
1265
1265
1266
1266
1267
1267
1268
1268
1269
1269
1270
1270
1271
1271
1272
1272
1273
1273
1274
1274
1275
1275
1276
1276
1277
1277
1278
1278
1279
1279
1280
1280
1281
1281
1282
1282
1283
1283
1284
1284
1285
1285
1286
1286
1287
1287
1288
1288
1289
1289
1290
1290
1291
1291
1292
1292
1293
1293
1294
1294
1295
1295
1296
1296
1297
1297
1298
1298
1299
1299
1300
1300
1301
1301
1302
1302
1303
1303
1304
1304
1305
1305
1306
1306
1307
1307
1308
1308
1309
1309
1310
1310
1311
1311
1312
1312
1313
1313
1314
1314
1315
1315
1316
1316
1317
1317
1318
1318
1319
1319
1320
1320
1321
1321
1322
1322
1323
1323
1324
1324
1325
1325
1326
1326
1327
1327
1328
1328
1329
1329
1330
1330
1331
1331
1332
1332
1333
1333
1334
1334
1335
1335
1336
1336
1337
1337
1338
1338
1339
1339
1340
1340
1341
1341
1342
1342
1343
1343
1344
1344
1345
1345
1346
1346
1347
1347
1348
1348
1349
1349
1350
1350
1351
1351
1352
1352
1353
1353
1354
1354
1355
1355
1356
1356
1357
1357
1358
1358
1359
1359
1360
1360
1361
1361
1362
1362
1363
1363
1364
1364
1365
1365
1366
1366
1367
1367
1368
1368
1369
1369
1370
1370
1371
1371
1372
1372
1373
1373
1374
1374
1375
1375
1376
1376
1377
1377
1378
1378
1379
1379
1380
1380
1381
1381
1382
1382
1383
1383
1384
1384
1385
1385
1386
1386
1387
1387
1388
1388
1389
1389
1390
1390
1391
1391
1392
1392
1393
1393
1394
1394
1395
1395
1396
1396
1397
1397
1398
1398
1399
1399
1400
1400
1401
1401
1402
1402
1403
1403
1404
1404
1405
1405
1406
1406
1407
1407
1408
1408
1409
1409
1410
1410
1411
1411
1412
1412
1413
1413
1414
1414
1415
1415
1416
1416
1417
1417
1418
1418
1419
1419
1420
1420
1421
1421
1422
1422
1423
1423
1424
1424
1425
1425
1426
1426
1427
1427
1428
1428
1429
1429
1430
1430
1431
1431
1432
1432
1433
1433
1434
1434
1435
1435
1436
1436
1437
1437
1438
1438
1439
1439
1440
1440
1441
1441
1442
1442
1443
1443
1444
1444
1445
1445
1446
1446
1447
1447
1448
1448
1449
1449
1450
1450
1451
1451
1452
1452
1453
1453
1454
1454
1455
1455
1456
1456
1457
1457
1458
1458
1459
1459
1460
1460
1461
1461
1462
1462
1463
1463
1464
1464
1465
1465
1466
1466
1467
1467
1468
1468
1469
1469
1470
1470
1471
1471
1472
1472
1473
1473
1474
1474
1475
1475
1476
1476
1477
1477
1478
1478
1479
1479
1480
1480
1481
1481
1482
1482
1483
1483
1484
1484
1485
1485
1486
1486
1487
1487
1488
1488
1489
1489
1490
1490
1491
1491
1492
1492
1493
1493
1494
1494
1495
1495
1496
1496
1497
1497
1498
1498
1499
1499
1500
1500
1501
1501
1502
1502
1503
1503
1504
1504
1505
1505
1506
1506
1507
1507
1508
1508
1509
1509
1510
1510
1511
1511
1512
1512
1513
1513
1514
1514
1515
1515
1516
1516
1517
1517
1518
1518
1519
1519
1520
1520
1521
1521
1522
1522
1523
1523
1524
1524
1525
1525
1526
1526
1527
1527
1528
1528
1529
1529
1530
1530
1531
1531
1532
1532
1533
1533
1534
1534
1535
1535
1536
1536
1537
1537
1538
1538
1539
1539
1540
1540
1541
1541
1542
1542
1543
1543
1544
1544
1545
1545
1546
1546
1547
1547
1548
1548
1549
1549
1550
1550
1551
1551
1552
1552
1553
1553
1554
1554
1555
1555
1556
1556
1557
1557
1558
1558
1559
1559
1560
1560
1561
1561
1562
1562
1563
1563
1564
1564
1565
1565
1566
1566
1567
1567
1568
1568
1569
1569
1570
1570
1571
1571
1572
1572
1573
1573
1574
1574
1575
1575
1576
1576
1577
1577
1578
1578
1579
1579
1580
1580
1581
1581
1582
1582
1583
1583
1584
1584
1585
1585

```

## 2.5.14 DescriptionFragment



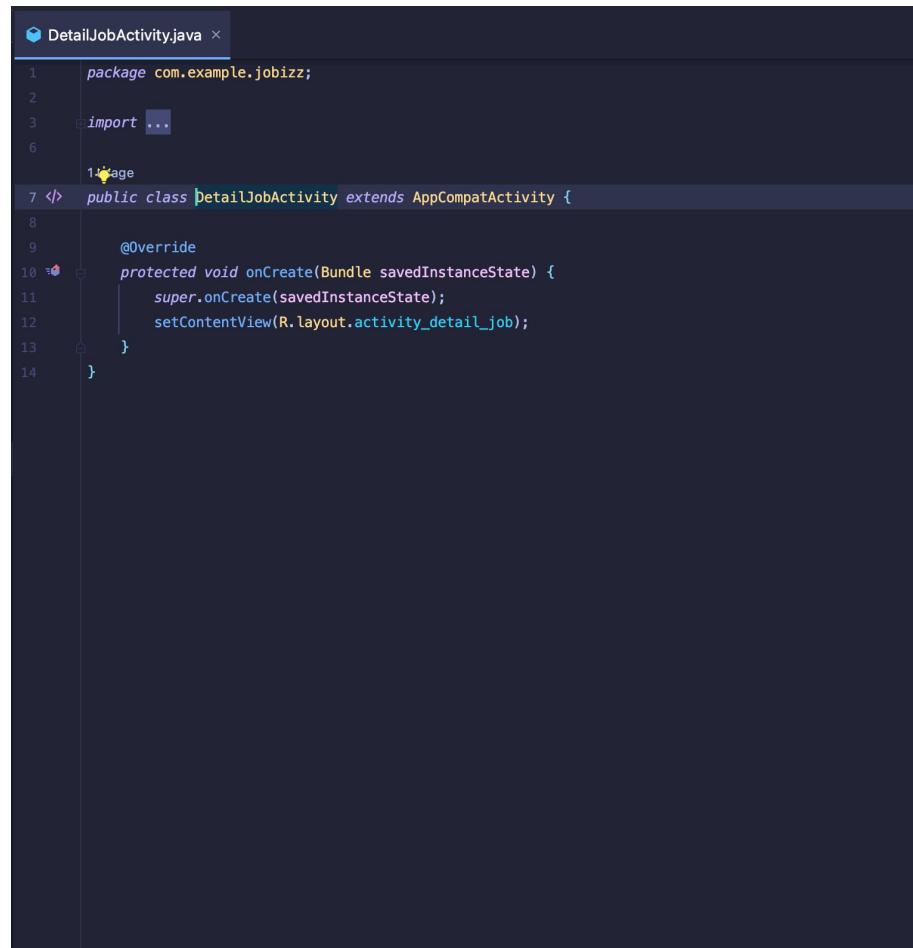
The screenshot shows a Java code editor with the file `DescriptionFragment.java` open. The code defines a fragment class with two parameters: `ARG_PARAM1` and `ARG_PARAM2`. It includes a constructor and a factory method `#newInstance`.

```
2 import ...
9
10 /**
11 * A simple {@link Fragment} subclass.
12 * Use the {@link DescriptionFragment#newInstance} factory method to
13 * create an instance of this fragment.
14 */
15 public class DescriptionFragment extends Fragment {
16
17     // TODO: Rename parameter arguments, choose names that match
18     // the fragment initialization parameters, e.g. ARG_ITEM_NUMBER
19     private static final String ARG_PARAM1 = "param1";
20     private static final String ARG_PARAM2 = "param2";
21
22     // TODO: Rename and change types of parameters
23     private String mParam1;
24     private String mParam2;
25
26     public DescriptionFragment() {
27         // Required empty public constructor
28     }
29
30     /**
31      * Use this factory method to create a new instance of
32      * this fragment using the provided parameters.
33      *
34      * @param param1 Parameter 1.
35      * @param param2 Parameter 2.
36      * @return A new instance of fragment DescriptionFragment.
37     */
38 }
```

Kode ini adalah kode yang berada di Description Fragment yang berguna untuk mengubah description profile user pada aplikasi Jobizz, di code ini ada parameter nya juga yang berguna untuk me rename atau mengganti type parameter nya



## 2.5.15 Detail Jobs

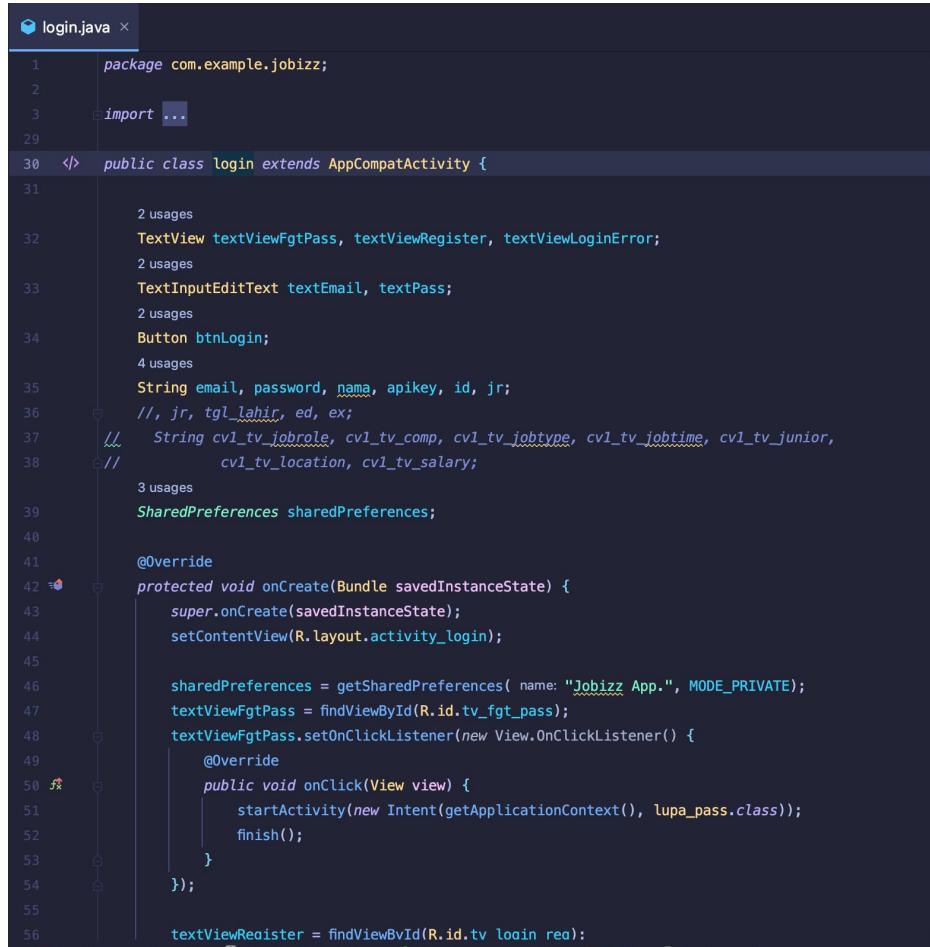


```
DetailJobActivity.java
1 package com.example.jobizz;
2
3 import ...
4
5
6
7 public class DetailJobActivity extends AppCompatActivity {
8
9     @Override
10    protected void onCreate(Bundle savedInstanceState) {
11        super.onCreate(savedInstanceState);
12        setContentView(R.layout.activity_detail_job);
13    }
14}
```

Code ini adalah code DetailJobActivity yang berguna jika dilihat dari aplikasi Jobizz, ia berguna untuk melihat detail detail pekerjaan pada sebuah perusahaan yang dituju oleh user.



## 2.5.16 Login

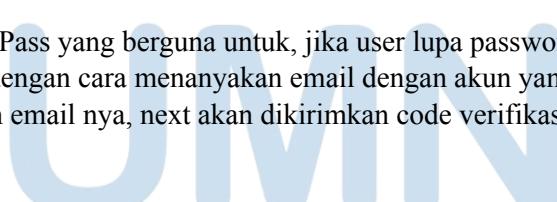


```
login.java ×
1 package com.example.jobizz;
2
3 import ...
29
30 </> public class login extends AppCompatActivity {
31
32     2 usages
33     TextView textViewFgtPass, textViewRegister, textViewLoginError;
34     2 usages
35     TextInputEditText textEmail, textPass;
36     2 usages
37     Button btnLogin;
38     4 usages
39     String email, password, nama, apikey, id, jr;
40     //, jr, tgl_lahir, ed, ex;
41     //, cv1_tv_jobrole, cv1_tv_comp, cv1_tv_jobtype, cv1_tv_jobtime, cv1_tv_junior,
42     //, cv1_tv_location, cv1_tv_salary;
43     3 usages
44     SharedPreferences sharedPreferences;
45
46     @Override
47     protected void onCreate(Bundle savedInstanceState) {
48         super.onCreate(savedInstanceState);
49         setContentView(R.layout.activity_login);
50
51         sharedPreferences = getSharedPreferences( name: "Jobizz App.", MODE_PRIVATE);
52         textViewFgtPass = findViewById(R.id.tv_fgt_pass);
53         textViewFgtPass.setOnClickListener(new View.OnClickListener() {
54             @Override
55             public void onClick(View view) {
56                 startActivity(new Intent(getApplicationContext(), lupa_pass.class));
57                 finish();
58             }
59         });
60
61         textViewRegister = findViewById(R.id.tv_login_rea);
```

Kode diatas merupakan kode yang digunakan untuk login ke dalam aplikasi Jobizz, dimana username, password, dan email yang dimasukan akan di sesuaikan dengan data yang berada di dalam database.



## 2.5.17 Lupa pass



```
lupa_pass.java
1 package com.example.jobizz;
2
3 import ...
4
5
6 3 usages
7
8 public class lupa_pass extends AppCompatActivity {
9
10    2 usages
11    ImageButton imageButtonLeft;
12
13    @Override
14    protected void onCreate(Bundle savedInstanceState) {
15        super.onCreate(savedInstanceState);
16        setContentView(R.layout.activity_lupa_pass);
17
18        imageButtonLeft = findViewById(R.id.ib_fp_left);
19        imageButtonLeft.setOnClickListener(new View.OnClickListener() {
20            @Override
21            public void onClick(View view) {
22                startActivity(new Intent(getApplicationContext(), login.class));
23                finish();
24            }
25        });
26    }
27
28 }
29
30
31
32 }
```

Code ini adalah code LupaPass yang berguna untuk, jika user lupa password account nya, code ini akan memberikan respon dengan cara menanyakan email dengan akun yang lupa password nya, setelah sudah memasukkan email nya, next akan dikirimkan code verifikasi atau link untuk mengubah pass yang baru.

### 2.5.18 MapsFragment

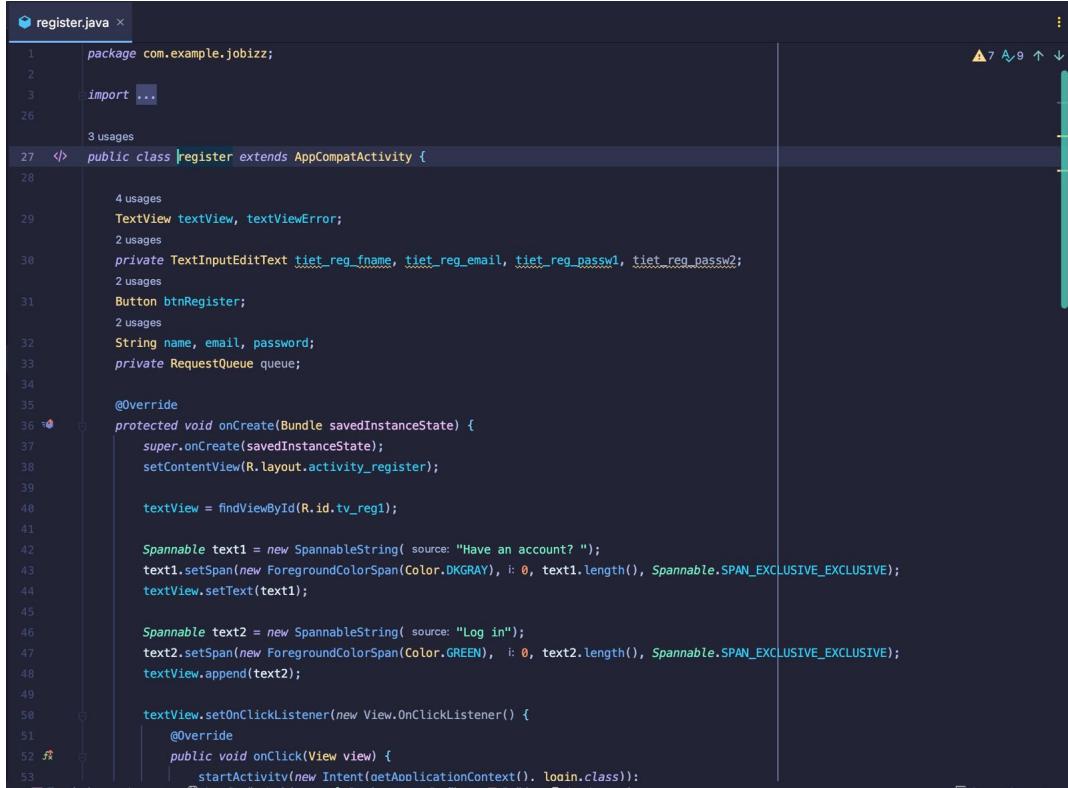
Kode berikut bertujuan untuk menggunakan fitur maps untuk menampilkan lokasi, pada aplikasi ini, fitur tersebut digunakan untuk menampilkan lokasi tempat kerja yang pengguna daftar.

### **2.5.19 PersinfoFragment**

```
❶ PersinfoFragment.java x
❷ package com.example.jobizz;
❸ import ...
❹
❺ 2 pages
❻ public class PersinfoFragment extends Fragment {
❼
❼     5 usages
❼     SharedPreferences sharedPreferences;
❼
❼     6 usages
❼     FragmentPersinfoBinding binding;
❼     String id;
❼
❼     @Override
❼     public View onCreateView(LayoutInflater inflater, ViewGroup container,
❼             Bundle savedInstanceState) {
❼         sharedPreferences = getActivity().getSharedPreferences("Jobizz App.", MODE_PRIVATE);
❼         binding = FragmentPersinfoBinding.inflate(inflater, container, attachToParent: false);
❼         View view = binding.getRoot();
❼
❼         binding.pvName.setText(sharedPreferences.getString("name", ""));
❼         binding.pvWork.setText(sharedPreferences.getString("jr", ""));
❼
❼         //ambil data personal info
❼         binding.tvDptEdu.setText(Html.fromHtml(sharedPreferences.getString("ed", "", Html.FROM_HTML_MODE_LEGACY)));
❼         binding.tvDptEx.setText(Html.fromHtml(sharedPreferences.getString("ex", "", Html.FROM_HTML_MODE_LEGACY)));
❼
❼         return view;
❼         // Inflate the layout for this fragment
❼         //return inflater.inflate(R.layout.fragment_persinfo, container, false);
❼     }
❼ }
```

Kode tersebut digunakan untuk mengambil data pengguna yang pernah dimasukan seperti nama pengguna, pekerjaan yang sedang di lakukan, dan pengalaman dari dalam database dan di tampilkan kembali ke dalam aplikasi.

## 2.5.20 Register



The screenshot shows the code editor for the `register.java` file. The code is written in Java and defines a `register` class that extends `AppCompatActivity`. The class contains several fields for text inputs and a button, and overrides the `onCreate` method to handle account creation logic. The code uses `SpannableString` to format text and `setOnClickListener` to handle button clicks.

```
1 package com.example.jobizz;
2
3 import ...
4
5 3 usages
6 <?> public class register extends AppCompatActivity {
7
8     4 usages
9     TextView textView, textViewError;
10    2 usages
11    private TextInputEditText tiet_reg_fname, tiet_reg_email, tiet_reg_passw1, tiet_reg_passw2;
12    2 usages
13    Button btnRegister;
14    2 usages
15    String name, email, password;
16    private RequestQueue queue;
17
18    @Override
19    protected void onCreate(Bundle savedInstanceState) {
20        super.onCreate(savedInstanceState);
21        setContentView(R.layout.activity_register);
22
23        textView = findViewById(R.id.tv_reg1);
24
25        Spannable text1 = new SpannableString( source: "Have an account? ");
26        text1.setSpan(new ForegroundColorSpan(Color.DKGRAY), 0, text1.length(), Spannable.SPAN_EXCLUSIVE_EXCLUSIVE);
27        textView.setText(text1);
28
29        Spannable text2 = new SpannableString( source: "Log in" );
30        text2.setSpan(new ForegroundColorSpan(Color.GREEN), 0, text2.length(), Spannable.SPAN_EXCLUSIVE_EXCLUSIVE);
31        textView.append(text2);
32
33        textView.setOnClickListener(new View.OnClickListener() {
34            @Override
35            public void onClick(View view) {
36                startActivity(new Intent(getApplicationContext(), login.class));
37            }
38        });
39    }
40
41    5 usages
42    2 usages
43    2 usages
44    2 usages
45    2 usages
46    2 usages
47    2 usages
48    2 usages
49    2 usages
50    2 usages
51    2 usages
52    2 usages
53    2 usages
```

Code ini adalah code untuk Register, sebelum code ini lanjut ke form register, dia akan menanyakan terlebih dahulu dengan pertanyaan “Have an account?” setelah itu jika user tidak mempunyai account, otomatis akan muncul Tombol register yang berguna jika user ingin register pada aplikasi jobizz, maka code ini akan menampilkan form register



## 2.5.21 RequirementFragment

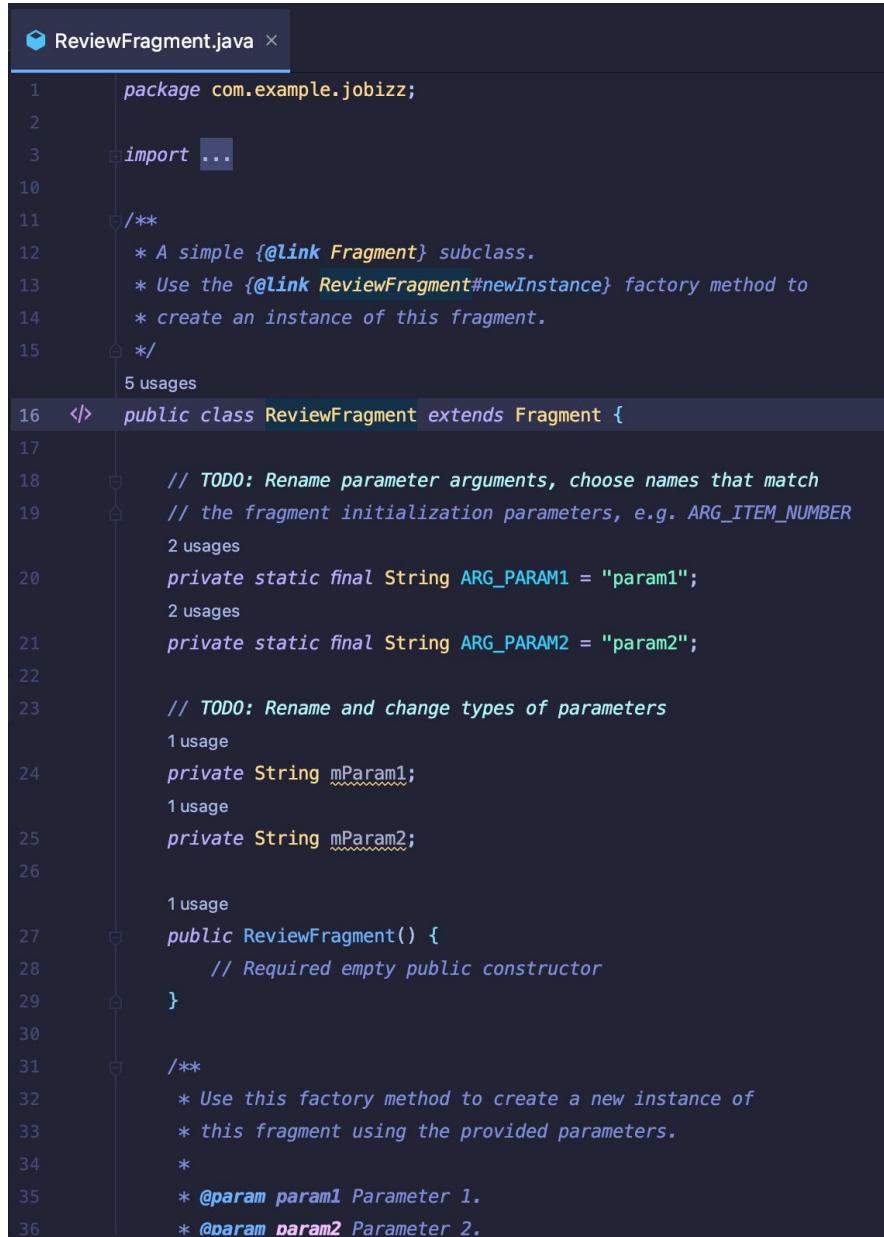


```
RequirementFragment.java ×

1 package com.example.jobizz;
2
3 import ...
4
5 /**
6  * A simple {@link Fragment} subclass.
7  * Use the {@link RequirementFragment#newInstance} factory method to
8  * create an instance of this fragment.
9  */
10
11 /**
12  * @param param1 Parameter 1.
13  * @param param2 Parameter 2.
14  */
15
16 public class RequirementFragment extends Fragment {
17
18     // TODO: Rename parameter arguments, choose names that match
19     // the fragment initialization parameters, e.g. ARG_ITEM_NUMBER
20     2 usages
21     private static final String ARG_PARAM1 = "param1";
22     2 usages
23     private static final String ARG_PARAM2 = "param2";
24
25     // TODO: Rename and change types of parameters
26     1 usage
27     private String mParam1;
28     1 usage
29     private String mParam2;
30
31     2 usages
32     /**
33      * Use this factory method to create a new instance of
34      * this fragment using the provided parameters.
35      *
36      * @param param1 Parameter 1.
37      * @param param2 Parameter 2.
```

UNIVERSITAS  
MULTIMEDIA  
NUSANTARA

## 2.5.22 ReviewFragment



The screenshot shows a code editor window with the file "ReviewFragment.java" open. The code is a Java class definition for a fragment. It includes package declarations, imports, class headers, field definitions, and constructor declarations. The code is annotated with TODO comments and usage statistics.

```
1 package com.example.jobizz;
2
3 import ...
4
5 /**
6  * A simple {@link Fragment} subclass.
7  * Use the {@link ReviewFragment#newInstance} factory method to
8  * create an instance of this fragment.
9  */
10
11 public class ReviewFragment extends Fragment {
12
13     // TODO: Rename parameter arguments, choose names that match
14     // the fragment initialization parameters, e.g. ARG_ITEM_NUMBER
15     2 usages
16     private static final String ARG_PARAM1 = "param1";
17     2 usages
18     private static final String ARG_PARAM2 = "param2";
19
20     // TODO: Rename and change types of parameters
21     1 usage
22     private String mParam1;
23     1 usage
24     private String mParam2;
25
26     1 usage
27     public ReviewFragment() {
28         // Required empty public constructor
29     }
30
31     /**
32      * Use this factory method to create a new instance of
33      * this fragment using the provided parameters.
34      *
35      * @param param1 Parameter 1.
36      * @param param2 Parameter 2.
```

Kode berikut bertujuan untuk memberikan pengguna kemampuan untuk melihat deskripsi dari perusahaan yang akan di daftar, data data yang diambil akan berasal dari databse yang telah dibuat dan ditampilkan ke aplikasi.

## BAB III

### Conclusion

#### 3.1 Conclusion

Aplikasi Jobizz hadir sebagai solusi yang mempermudah para pekerja dalam mencari peluang pekerjaan. Dengan fungsionalitas yang canggih dan antarmuka yang user-friendly, Jobizz menyediakan platform yang efektif dan efisien untuk menjembatani antara pencari kerja dengan beragam peluang pekerjaan yang sesuai dengan kualifikasi dan preferensi mereka. Aplikasi ini tidak hanya menawarkan akses yang lebih mudah ke berbagai lowongan kerja, tetapi juga menyediakan fitur-fitur tambahan, seperti notifikasi pekerjaan terbaru, informasi perusahaan, dan bahkan saran karir. Dengan Jobizz, para pekerja dapat mengoptimalkan proses pencarian pekerjaan mereka, meningkatkan keterlibatan dengan pasar kerja, dan pada akhirnya, meraih peluang karir yang lebih baik sesuai dengan keahlian dan aspirasi mereka. Sebagai mitra setia dalam perjalanan karir, Jobizz membantu mempercepat dan menyederhanakan langkah-langkah menuju pekerjaan yang diimpikan.

#### 3.2 Member Roles

Nama	Role
Andika (00000073646)	Home, Database, Forgot Password, Proposal
Marcello Roy (00000072593)	Homepage, Menu bar, Login, Proposal
Victor Georgius Wididyanto (00000073402)	Sql lite, Message, Register, Proposal
Febianus Felix Widisulistyo (00000072737)	Webview, Maps, Bookmark, Portofolio, Proposal

