

LAPORAN HASIL EVALUASI AKHIR SEMESTER
RANCANGAN APLIKASI RESTORAN BERBASIS ANDROID

*Laporan ini disusun untuk memenuhi salah satu tugas mata kuliah Pemrograman
Perangkat Bergerak*



Disusun oleh:

201511045	Irfan Pertrio Nugroho
201511057	Restu Triwahyuni

D3 Teknik Informatika
Jurusan Teknik Komputer dan Informatika
Politeknik Negeri Bandung
2022

Daftar Isi

KATA PENGANTAR	3
BAB I PENDAHULUAN	4
1.1. Deskripsi Aplikasi	4
1.2. Spesifikasi Umum Aplikasi	4
1.2.1. Analisa Kebutuhan Aplikasi	4
1.2.2. Analisa Perangkat Lunak	4
1.2.3. Analisa Perangkat Keras	5
1.3. Batasan Aplikasi	5
1.4. Tujuan	6
1.5. Manfaat	6
BAB II ANALISIS DAN PERANCANGAN APLIKASI	7
2.1. Fitur-fitur Aplikasi	7
2.2. Konsep yang Diterapkan	7
2.3. Desain Aplikasi	8
2.3.1. Use Case	8
2.3.2. Activity Diagram	9
2.3.3. Mock Up	11
2.3.3.1. Halaman Login	11
2.3.3.3. Halaman List Menu Makanan	12
2.3.3.4. Halaman Detail Info Menu	12
2.3.3.5. Halaman Search Menu	13
2.3.3.6. Halaman Hasil Searching Menu	13
2.3.4. Mekanisme Konsumsi API	14
BAB III PENUTUP	14

KATA PENGANTAR

Puji syukur kehadiran Tuhan YME yang telah memberikan rahmat dan inayah-Nya sehingga kami dapat menyelesaikan laporan EAS Pemrograman Perangkat Bergerak Teori.

Terima kasih penyusun ucapkan kepada para Dosen pengampu mata kuliah Pemrograman Perangkat Lunak yang telah membantu baik secara moral maupun materi. Terima kasih juga penyusun ucapkan kepada teman-teman seperjuangan yang telah mendukung, sehingga penyusun bisa menyelesaikan tugas ini tepat waktu.

Penyusun menyadari, bahwa laporan yang dibuat ini masih jauh dari kata sempurna baik segi penyusunan, bahasa, maupun penulisannya. Oleh karena itu, penyusun sangat mengharapkan kritik dan saran yang membangun dari semua pembaca guna menjadi acuan agar penyusun bisa menjadi lebih baik lagi di masa mendatang.

Semoga laporan ini bisa menambah wawasan para pembaca dan bisa bermanfaat untuk perkembangan dan peningkatan ilmu pengetahuan.

Bandung, 30 Mei 2022

Penyusun,

BAB I

PENDAHULUAN

1.1. Deskripsi Aplikasi

“**Itadakimasu**” merupakan sebuah aplikasi berbasis android yang menyediakan fitur memilih makanan dengan tema makanan Jepang. Dengan adanya aplikasi ini, nantinya pengguna diharapkan dapat dengan mudah melihat menu-menu makanan Jepang lengkap dengan penjelasan makanannya tanpa perlu lagi *browsing* di internet.

1.2. Spesifikasi Umum Aplikasi

Dalam hal ini dilakukan analisa kebutuhan sesuai dengan kebutuhan aplikasi yang dibangun, analisa kebutuhan aplikasi terkait apa saja yang dibutuhkan agar sesuai dengan tujuan yang diharapkan. Selain menganalisa kebutuhan aplikasi, pada analisa ini juga akan menganalisa kebutuhan perangkat lunak dan perangkat keras dalam pembuatan aplikasi Itadakimasu ini.

Berikut adalah analisa kebutuhan aplikasi, perangkat lunak dan perangkat keras dalam pembuatan aplikasi Itadakimasu ini, adapun analisa sebagai berikut:

1.2.1. Analisa Kebutuhan Aplikasi

Proses untuk mendapatkan pengetahuan kebutuhan dapat dilakukan dengan berbagai macam cara, yakni pengetahuan dari internet dan berdiskusi bersama rekan satu kelompok. Dari sumber pengetahuan tersebut dikumpulkan dan kemudian direpresentasikan kedalam program aplikasi.

1.2.2. Analisa Perangkat Lunak

a. Android Studio

Software yang digunakan untuk membuat aplikasi android, membuat user interface dengan menggunakan xml sebagai layout kemudian memasukan perintah untuk memanggil user interface yang telah dibuat dengan program dalam bahasa pemrograman java.

b. Figma

Software yang digunakan untuk menyediakan asset-asset yang diperlukan untuk kebutuhan pembuatan aplikasi. Beberapa asset yang dibuat yaitu seperti desain gambar setiap menu makanan yang ditampilkan pada saat halaman Detail Menu Makanan ditampilkan.

c. Android 7.1 Nougat

Android 7.1 Nougat digunakan sebagai sistem operasi dari handphone android yang digunakan untuk pembuatan aplikasi Itadakimasu ini.

1.2.3. Analisa Perangkat Keras

a. Laptop

Laptop yang digunakan untuk membuat aplikasi android, spesifikasi laptop yang di gunakan pada pembuatan aplikasi android ini adalah :

Processor : Intel® Core™ i5-10110U Processor 4M Cache, up to 4.10 GHz.

RAM : 4GB DDR4.

HDD : 1TB.

b. Handphone Android

Pada pembuatan aplikasi ini, handphone android digunakan langsung untuk uji coba aplikasi, spesifikasi handphone android yang digunakan adalah sebagai berikut :

OS : Android 7.1 (Nougat), ColorOS 3.1

Screen Size : 5.2 inches, 74.5 cm² (~68.2% screen-to-body ratio)

Screen Resolution : 720 x 1280 pixels, 16:9 ratio (~282 ppi density)

1.3. Batasan Aplikasi

Batasan dari Aplikasi Itadakimasu ini diantaranya:

1. Aplikasi ini hanya berjalan pada perangkat mobile android dengan sistem operasi android 7 keatas.
2. Pada fitur Login belum terdapat reset password jika User lupa password yang didaftarkan pada proses Register.

1.4. Tujuan

Tujuan dari pembuatan Aplikasi Itadakimasu ini adalah:

1. Memenuhi EAS mata kuliah Pemrograman Perangkat Bergerak (Mobile).
2. Melatih serta meningkatkan pengetahuan dan pemahaman dalam pembuatan aplikasi berbasis mobile beserta dokumentasinya.

1.5. Manfaat

Manfaat dari aplikasi Itadakimasu ini antara lain:

1. Mempermudah pengguna dalam mencari menu makanan Jepang yang sedang *booming* di kalangan masyarakat saat ini.
2. Pengguna dapat menambahkan *knowledge*-nya terhadap menu-menu makanan yang berasal dari negara Jepang.
3. Pengguna dapat dengan mudah mendapatkan informasi terkait menu makanan Jepang tanpa perlu lagi bersusah payah menggunakan *browser*.

BAB II

ANALISIS DAN PERANCANGAN APLIKASI

2.1. Fitur-fitur Aplikasi

Aplikasi ini terdiri beberapa fitur untuk menunjang kebutuhan pengguna dalam mencari menu makanan Jepang secara online, fitur tersebut diantaranya:

a. Dapat melakukan Register dan Login

Fitur Register digunakan ketika User belum mempunyai akun untuk masuk ke dalam aplikasi dengan menginputkan email, phone, username, password, dan confirm password. Sedangkan apabila User telah memiliki akun, maka dapat langsung melakukan Login dengan menginputkan username dan password yang didaftarkan pada proses Register yang dihubungkan oleh API.

b. Dapat menampilkan daftar menu makanan dari negara Jepang

Fitur ini diterapkan pada halaman utama dengan menerapkan informasi meliputi nama menu makanan beserta foto dari makanan tersebut.

c. Dapat menampilkan detail/deskripsi dari makanan yang dipilih

Detail dari makanan yang dipilih akan menampilkan informasi meliputi nama lengkap makanan, deskripsi makanan, serta beberapa foto makanan yang dipilih oleh pengguna.

d. Dapat melakukan pencarian makanan berdasarkan kata kunci nama makanan

Pencarian makanan dapat dilakukan berdasarkan kata kunci dari nama makanan tersebut yang sudah disimpan dalam ArrayList.

2.2. Konsep yang Diterapkan

Aplikasi ini dibangun dengan menerapkan beberapa konsep yang telah diajarkan sebelumnya pada mata kuliah Pemrograman Perangkat Bergerak. Berikut adalah konsep yang digunakan pada aplikasi ini, diantaranya:

a. Activity

b. Intent

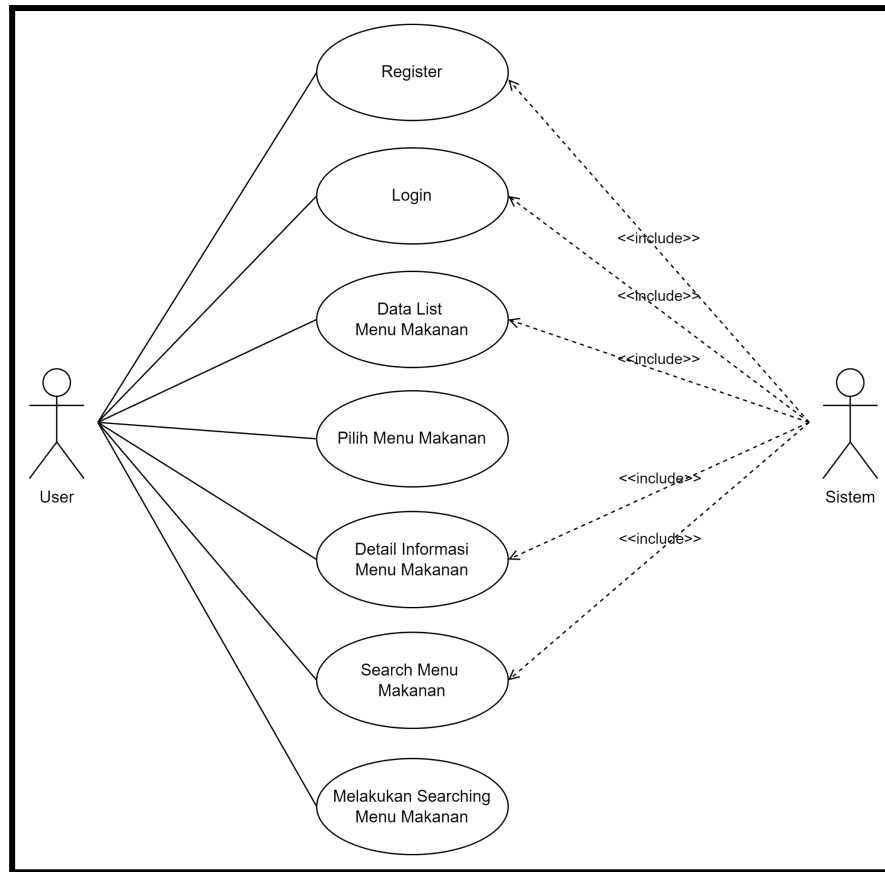
c. Fragment

- d. RecyclerView/ Card View
- e. SQLite Database
- f. Array List
- g. Live Data
- h. View Model
- i. Layout Inflater
- j. Search View
- k. Restful API

2.3. Desain Aplikasi

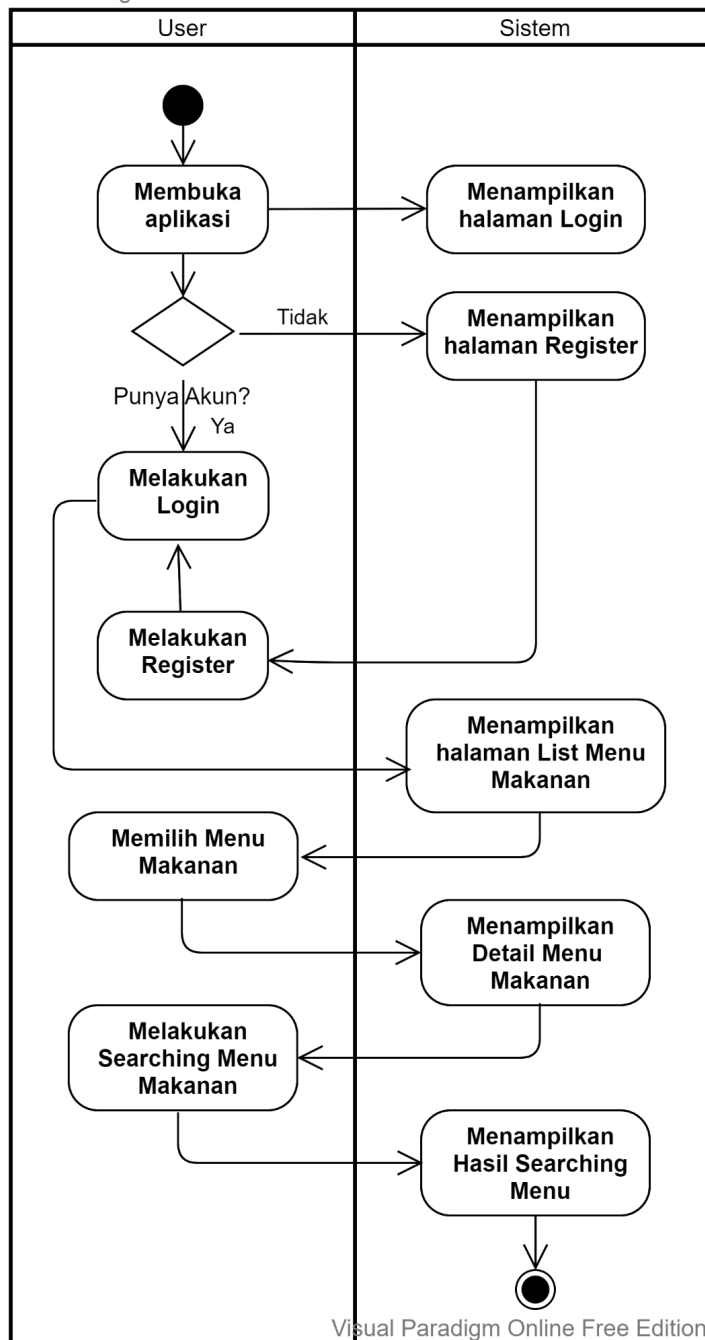
2.3.1. Use Case

Usecase diagram merupakan suatu aktivitas yang menggambarkan urutan interaksi antar satu atau lebih aktor dan sistem. Usecase yang akan dirancang yaitu usecase diagram untuk pengaksesan melalui perangkat Mobile. Gambar dibawah ini menjelaskan aliran usecase diagram pengaksesan melalui perangkat Mobile. Interaksi antara use case dengan aktor terhadap sistem digambarkan menggunakan use case diagram. Terdapat dua aktor yang terlibat dalam proses penggunaan aplikasi, yaitu User dan Sistem.



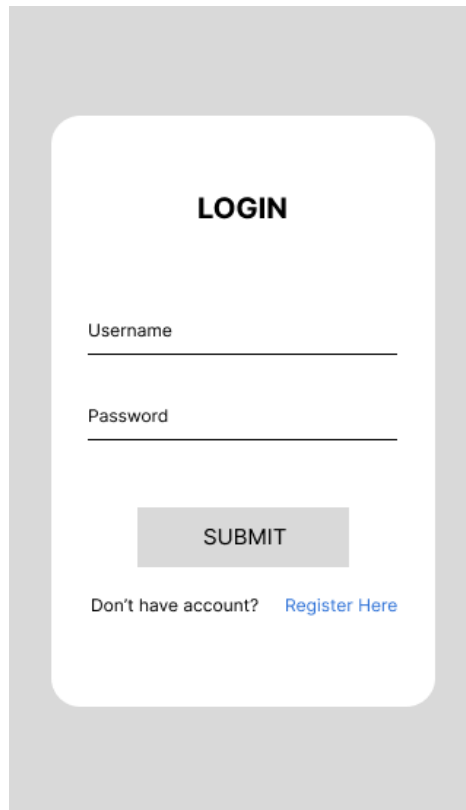
2.3.2. Activity Diagram

Activity diagram merupakan alur kerja pada setiap usecase. Activity diagram pada analisa ini mencakup activity diagram setiap usecase. Berikut ini adalah gambar activity diagram dari proses aplikasi Itadakimasu.



2.3.3. Mock Up

2.3.3.1. Halaman Login



A mockup of a login page. It features a white rounded rectangle centered on a gray background. Inside the rectangle, the word "LOGIN" is centered at the top. Below it are two input fields: "Username" and "Password", each with a horizontal line underneath. A gray "SUBMIT" button is centered below the password field. At the bottom, the text "Don't have account?" is followed by a blue link "Register Here".

LOGIN

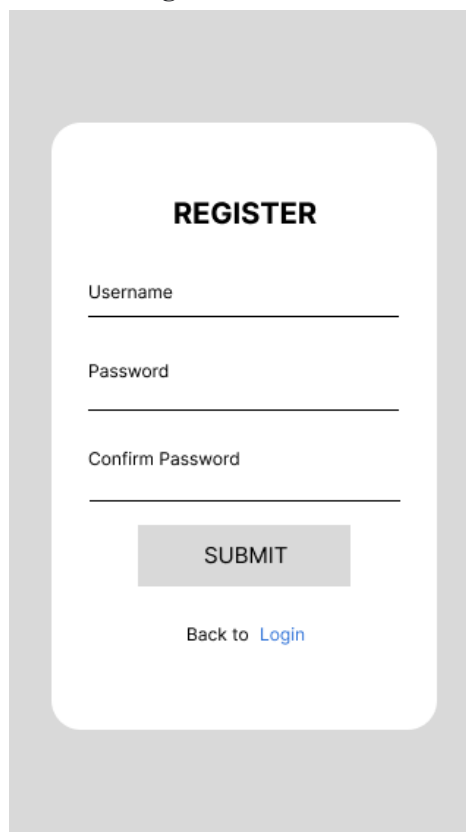
Username

Password

SUBMIT

Don't have account? [Register Here](#)

2.3.3.2. Halaman Register



A mockup of a register page. It features a white rounded rectangle centered on a gray background. Inside the rectangle, the word "REGISTER" is centered at the top. Below it are three input fields: "Username", "Password", and "Confirm Password", each with a horizontal line underneath. A gray "SUBMIT" button is centered below the "Confirm Password" field. At the bottom, the text "Back to" is followed by a blue link "Login".

REGISTER

Username

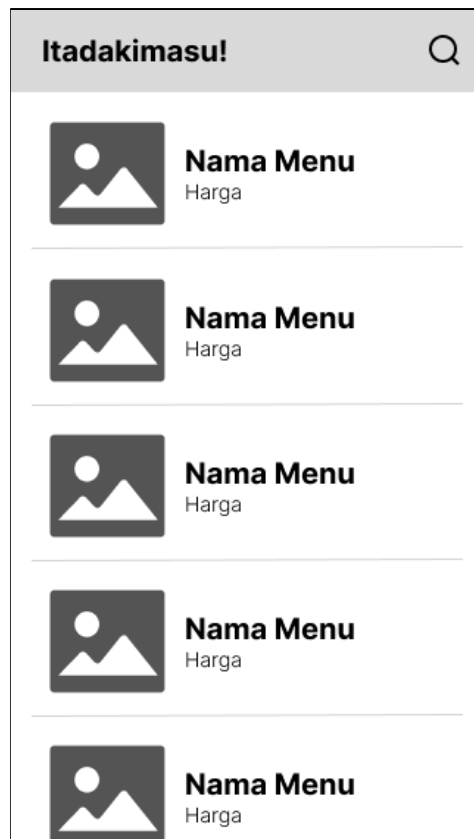
Password

Confirm Password

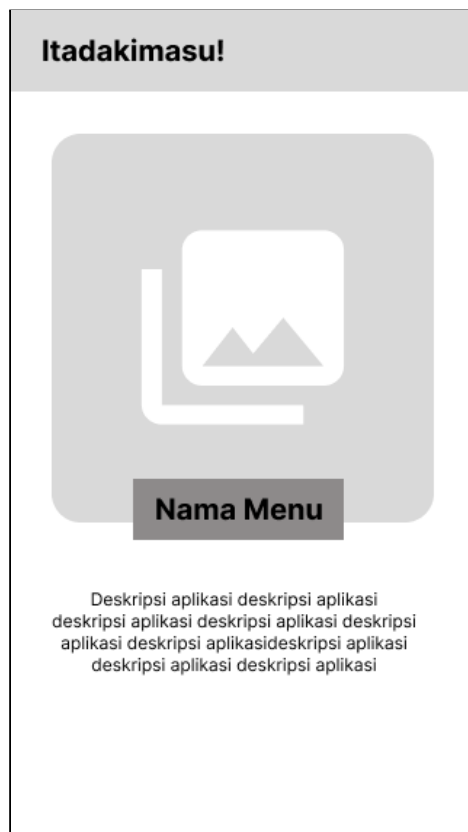
SUBMIT

Back to [Login](#)

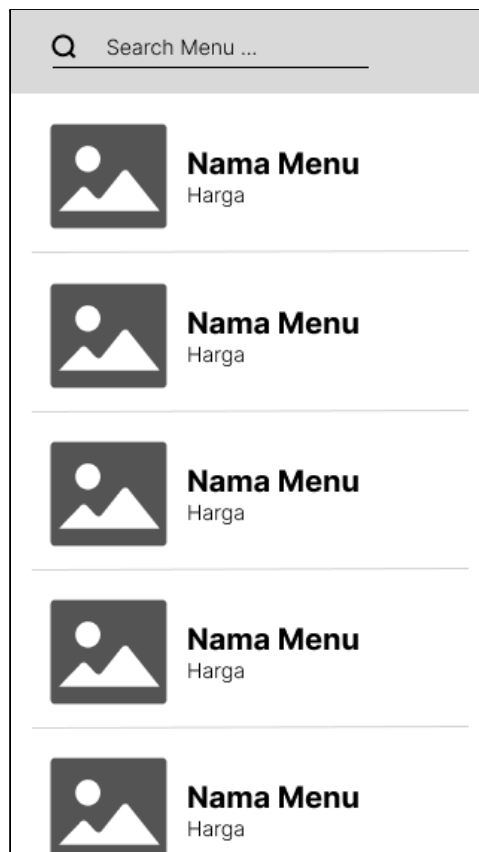
2.3.3.3. Halaman List Menu Makanan



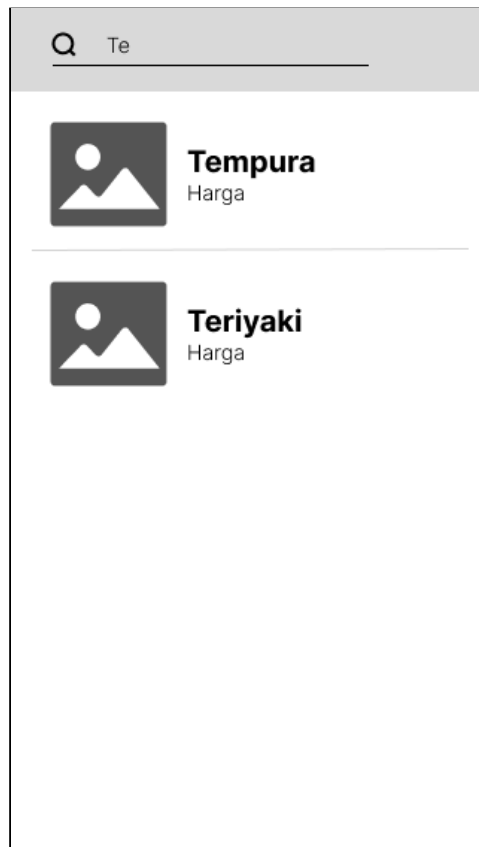
2.3.3.4. Halaman Detail Info Menu



2.3.3.5. Halaman Search Menu



2.3.3.6. Halaman Hasil Searching Menu



2.3.4. Mekanisme Konsumsi API

Rest Client yang kami gunakan adalah retrofit, dimana penggunaannya dipakai pada saat login dan register. Untuk pengimplementasiannya cukup sederhana yaitu dengan menambahkan module pada build.gradle. seperti berikut :

```
public class RestClient {

    private static Interface service;
    public static String token = "";
    public RestClient() {}

    public static Interface getService() {
        if (service == null) {
            String BASE_URL =
                "https://foodbukka.herokuapp.com/api/v1/";

            OkHttpClient.Builder httpClient = new
                OkHttpClient.Builder();
            httpClient.addInterceptor(new Interceptor() {
                @Override
                public Response intercept(Chain chain) throws
                    IOException {
                    Request request = chain.request().newBuilder()
                        .addHeader("Authorization", "Bearer " +
                            token)
                        .addHeader("Cookie", "token="+token)
                        .build();
                    Log.d("Authorization", "Bearer " + token);
                    return chain.proceed(request);
                }
            });

            Retrofit.Builder builder = new Retrofit.Builder()
                .baseUrl(BASE_URL)

            .addConverterFactory(GsonConverterFactory.create());

            Retrofit retrofit = builder.client(httpClient.build()).build();
            service = retrofit.create(Interface.class);
        }
        return service;
    }
}
```

Selanjutnya kami membuat class RegisterResponse.java yang berisi constructor dari atribut-atribut yang diperlukan pada REST, berikut source codenya :

```
public class RegisterResponse{

    @SerializedName("status")
    private String status;

    @SerializedName("token")
    private String token;

    public void setStatus(String status){
        this.status = status;
    }

    public String getStatus(){
        return status;
    }

    public void setToken(String token){
        this.token = token;
    }

    public String getToken(){
        return token;
    }

    @Override
    public String toString(){
        return
            "RegisterResponse{" +
                "status = '" + status + '\'' +
                ", token = '" + token + '\'' +
                "}";
    }
}
```

Lalu yang terakhir, kami membuat LoginResponse.java untuk mengambil data dari Username serta Password yang sudah didaftarkan sebelumnya agar pengguna dapat masuk ke laman utama aplikasi, untuk source codenya sebagai berikut :

```
public class LoginResponse{

    private String status;

    private String token;

    public void setStatus(String status){
        this.status = status;
    }

    public String getStatus(){
```

```

        return status;
    }

    public void setToken(String token){
        this.token = token;
    }

    public String getToken(){
        return token;
    }

    @Override
    public String toString(){
        return
            "LoginResponse{" +
                "status = '" + status + '\'' +
                ", token = '" + token + '\'' +
                "}";
    }
}

```

BAB III

PENUTUP

DAFTAR PUSTAKA

<https://blog.cakap.com/arti-itadakimasu/>

<https://badoystudio.com/membuat-login-android-dengan-sqlite-database/>

<https://www.geeksforgeeks.org/mvvm-model-view-viewmodel-architecture-pattern-in-android/>

<https://documenter.getpostman.com/view/12204297/TVKJwEWL#c5ec7244-02af-4dc5-8e5e-f76f64e1aecd>

<https://betterprogramming.pub/create-an-app-that-uses-livedata-and-viewmodel-in-java-f8086ca94229>

