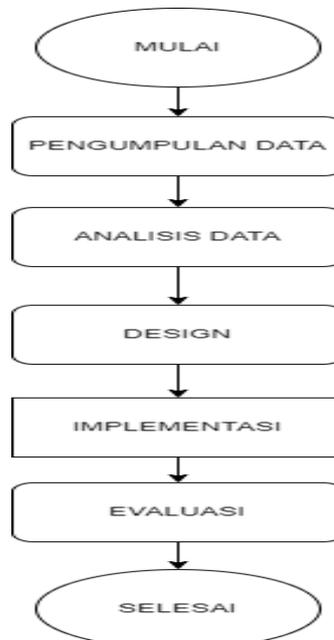


## BAB III METODOLOGI PENELITIAN

### 3.1 Tahapan Penelitian

Dalam penelitian ini metode yang digunakan untuk melakukan pengembangan perangkat lunak yang tepat dalam membuat “Aplikasi E-Katalog berbasis *Mobile Application*” adalah metode prototype. Penelitian dimulai dengan menganalisis kebutuhan pengguna dengan melakukan wawancara. Setelah mendapatkan informasi dan data untuk perangkat lunak, lanjut ketahapan selanjutnya membuat alur sistem aplikasi dengan rancangan *use case diagram* dan *activity diagram*. Kemudian melakukan pengumpulan bahan dengan melakukan pengambilan gambar produk alat tulis kantor dan juga mengumpulkan informasi dari produk tersebut. Berikutnya adalah tahapan membuat aplikasi dengan android studio setelah aplikasi jadi maka akan dilakukan tahapan pengujian yang akan dilakukan dengan pengujian black box dan melakukan beberapa pengujian terhadap pengguna untuk menguji apakah aplikasi berhasil berfungsi sesuai kebutuhan. Dan tahapan terakhir adalah tahapan pendistribusian dengan melakukan publikasi kepada pengguna.



Gambar 3.1 Tahap Penelitian

### 3.2 Metode Pengumpulan Data

Metode Pengumpulan Data dilakukan untuk mendapatkan informasi terkait dengan penelitian yang akan dilakukan. Metode Pengumpulan Data dilakukan dengan cara yang tepat untuk mengumpulkan data objektif yang relevan dengan pokok pembahasan terkait penelitian. Dalam hal ini, peneliti menggunakan beberapa langkah, sebagai berikut :

a. Observasi

Metode pengumpulan data terkait penelitian ini dilakukan dengan bertemu langsung dengan pihak *MP One Stationery* Kota Metro. Sekaligus mengamati toko dan produk apa saja yang tersedia disana.

b. Wawancara

Metode ini dilakukan dengan bertemu langsung dan melakukan proses tanya jawab atau wawancara kepada pihak terkait pada penelitian ini.

c. Studi Literatur

Metode ini dilakukan dengan cara mencari sumber data yang dapat mendukung dalam perancangan “E-Katalog Berbasis *Mobile Application* dengan Algoritma *Squeezer*”.

#### 3.2.1 Perencanaan Secara Cepat

Proses pengumpulan kebutuhan diperlukan untuk spesifikasi kebutuhan dari perangkat lunak agar sesuai dengan perangkat yang dibutuhkan. Pada tahap ini peneliti akan menjelaskan tahapan untuk membangun konsep dari pembuatan “E-Katalog berbasis *Mobile Application* Studi Kasus pada *MP One Stationery* Kota Metro” berdasarkan apa yang dibutuhkan oleh sistem.

Maka dengan itu dapat dibagi menjadi dua jenis kebutuhan, yakni Analisis Kebutuhan Fungsional dan Analisis Kebutuhan Non-Fungsional.

### **a. Analisis Kebutuhan Fungsional**

Pada analisis kebutuhan fungsional kali ini adalah bagian paparan mengenai fitur-fitur yang akan dimasukkan kedalam pembuatan “E-Katalog pada MP One Stationery” yang akan dibuat. Dalam fitur-fitur ini saling berhubungan.

### **b. Analisis Kebutuhan Non-Fungsional**

Pada Analisis kebutuhan Non-Fungsional ini terdapat dua komponen yakni :

#### **1. Analisis Perangkat Keras (*Hardware*)**

Perangkat keras yang akan digunakan dalam melakukan pengolahan data dan penyajian pada laporan tersebut adalah sebagai berikut :

- a) *Laptop Acer SF314-41*
- b) *RAM 4GB DDR4*
- c) *AMD Ryzen 5 3500U with Radeon Vega Mobile Gfx (8CPUs), ~2.1GHz*
- d) *Redmi 9 4/64*

#### **2. Analisis Perangkat Lunak (*Software*)**

Perangkat lunak yang akan digunakan untuk melakukan proses pembuatan “Perangkat lunak E-Katalog berbasis *Mobile Application* Studi Kasus pada MP One Stationery Kota Metro” dengan menggunakan dua *software* diantaranya :

##### **a. *Software* untuk pembuatan**

Berikut ini *software* yang akan digunakan pada saat melakukan pembuatan aplikasi adalah

1. *Operating Sistem*, menggunakan *Windows 10 Home Single Language 64-bit (10.0, Build 19041)*
2. *Microsoft Word 2010* adalah aplikasi yang digunakan pada proses pembuatan naskah
3. *Android Studio*, digunakan dalam membangun atau mengembangkan aplikasi android

4. *StarUML*, digunakan untuk mendesain rancangan *UML*
5. *AdobeXd*, yaitu untuk membuat rancangan awal tampilan pada aplikasi yang akan dibuat.
6. *Microsoft Excel* untuk sebelumnya mendata produk yang akan di masukkan kedalam *database*

b. *Software* untuk Penerapan

Perangkat lunak yang dapat dipergunakan untuk melakukan penerapan pada aplikasi *mobile* adalah menggunakan sistem operasi *AndroidVersiKitkat 4.4+*.

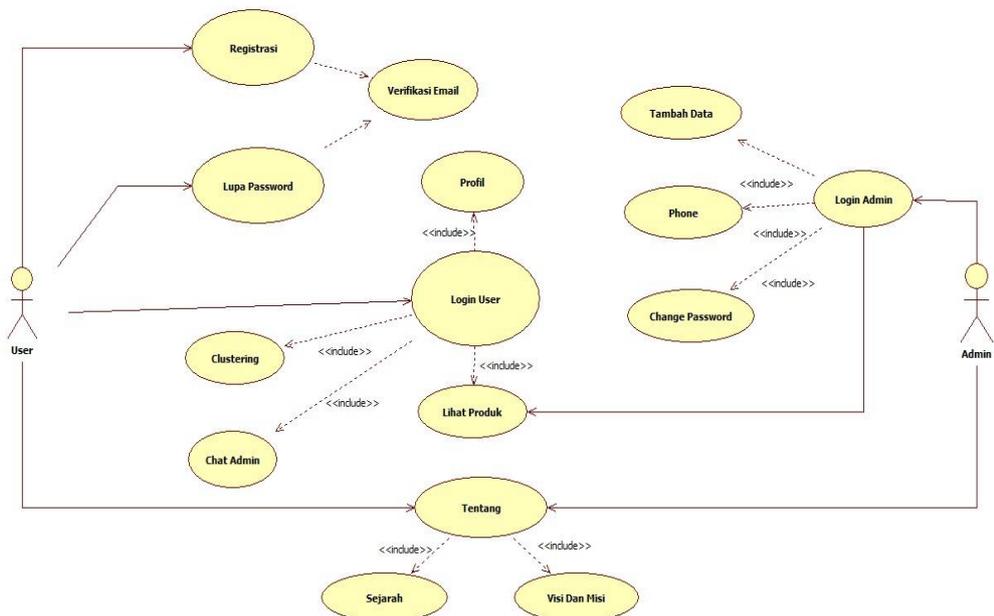
### 3.2.2 Pemodelan Perancangan Secara Cepat

Tahap ini merupakan tahapan untuk perancangan alur program aplikasi e-katalog berbasis *mobile application*. Perancangan ini terdiri dari dua aktor, yaitu aktor pengguna dan aktor admin, spesifikasi Perancangan Sistem dilakukan dengan UML (Unified Modelling Language) yang menerapkan *Use Case Diagram*, *Activity Diagram* dan *Design Interface*.

a. *Use Case Diagram*

*Use Case Diagram* digunakan untuk pemodelan kegiatan pada sistem yang akan dibuat.

Gambar 3.1 Berikut adalah rancangan *Use Case Diagram* dan Admin dari aplikasi E-Katalog yang akan dibuat :



Gambar 3.1 *Use Case Diagram*

Berdasarkan gambar *Use Case* di atas dapat kita lihat bahwa pada saat *user* mengakses aplikasi akan langsung menuju ke halaman awal aplikasi dengan registrasi terlebih dahulu, dan untuk *admin* ketika mengakses aplikasi harus *login* terlebih dahulu supaya dapat mengedit atau mengubah data yang tertera pada aplikasi E-Katalog.

Tabel 3.1 Identifikasi Aktor

<b>Nama Aktor</b>	<b>Definisi</b>
Pengguna ( <i>User</i> )	Pengguna sebagai orang yang dapat melihat informasi produk dan menghubungi admin.
Admin	Admin sebagai orang yang dapat melakukan kegiatan pengolahan produk toko seperti menambah, mengubah dan menghapus data tersebut.

Tabel 3.2 Identifikasi *Use Case*

<b>Nama Use Case</b>	<b>Deskripsi</b>	<b>Aktor</b>
Registrasi	Proses mendaftar sebagai pengguna pada aplikasi MP One Stationery.	User
Login	Proses masuk ke dalam aplikasi MP One Stationery sesuai hak aksesnya.	User dan Admin
Forget Password (Lupa Sandi)	Proses untuk mengubah kata sandi ketika lupa sandi ketika ingin masuk ke aplikasi.	User
Verifikasi Email	Proses memverifikasi email setelah selesai registrasi.	User
Tentang	Proses untuk melihat informasi seputar toko MP One Stationery.	User dan Admin

Profil	Proses untuk melihat profil pengguna setelah berhasil registrasi.	User
Klustering	Proses untuk mengkluster data berdasarkan merk produk yang akan ditampilkan.	User
Chat Admin	Proses untuk menghubungi admin terkait keperluan pengguna.	User
Tambah Data	Proses untuk menambahkan data produk ke dalam aplikasi.	Admin
Phone	Proses untuk menyimpan nomor admin yang dapat dihubungi user.	Admin
Change Password	Proses untuk mengubah kata sandi admin pada aplikasi.	Admin
Lihat Produk	Proses untuk melihat produk yang tertera pada aplikasi.	User dan Admin

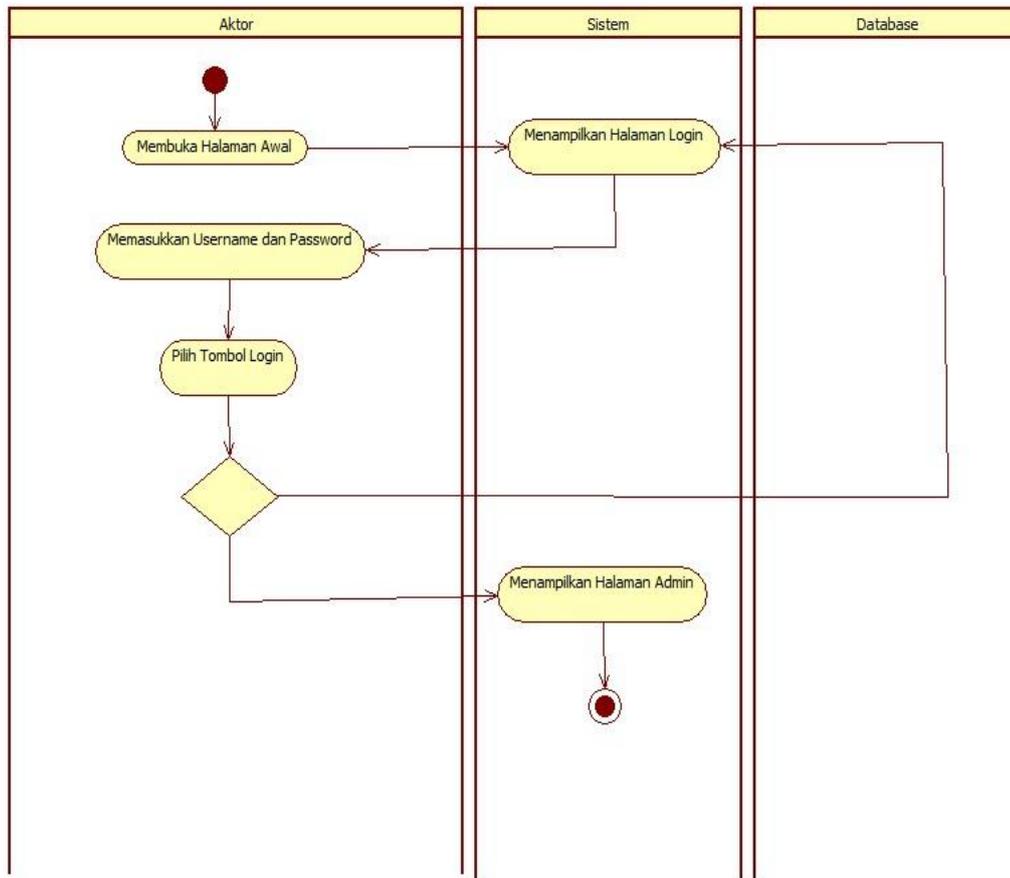
*b. Activity Diagram*

Digunakan untuk menggambarkan alur dari awal sebuah sistem, melakukan, dan mengakhiri sebuah proses. *Activity Diagram* yang akan diterapkan pada sistem ini adalah sebagai berikut :

1. *Activity Diagram Admin Login*

Pada *Activity Diagram Admin* merupakan urutan langkah-langkah yang dilakukan admin dalam melakukan input produk yang akan ditampilkan pada aplikasi mulai dari merubah dan menghapus informasi data produk yang tertera pada aplikasi.

Gambar 3.2 Pada gambar berikut adalah *Activity Diagram Admin* dari aplikasi E-Katalog yang akan dibuat :

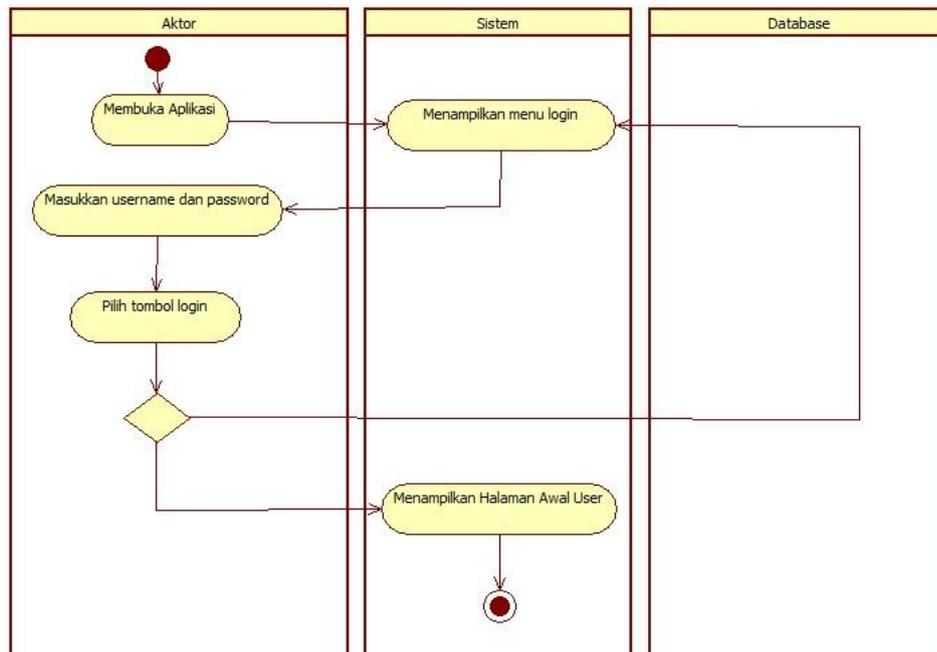


Gambar 3.2 Activity Diagram Admin Login

## 2. Activity Diagram User

*Activity Diagram User* adalah langkah-langkah yang dilakukan *user* untuk menggunakan Katalog Elektronik dalam perangkat lunak. Mulai dari membuka aplikasi ke menu utama, mengecek informasi produk dan tampil informasi produk yang di pilih.

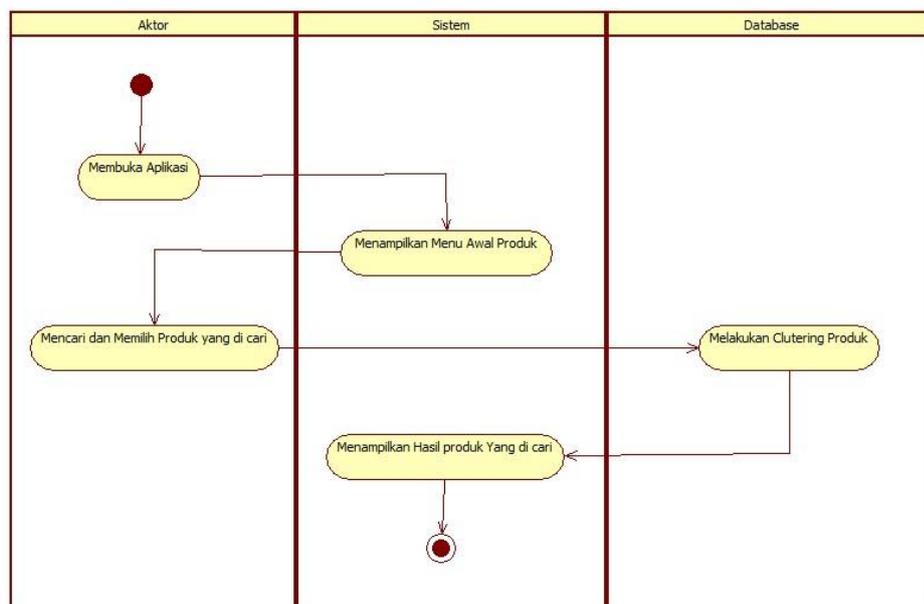
Gambar 3.3 Pada gambar berikut adalah *Activity Diagram User* dari aplikasi E-Katalog yang akan dibuat :



Gambar 3.3 Activity Diagram User

### 3. Activity Diagram Menu Pencarian Produk

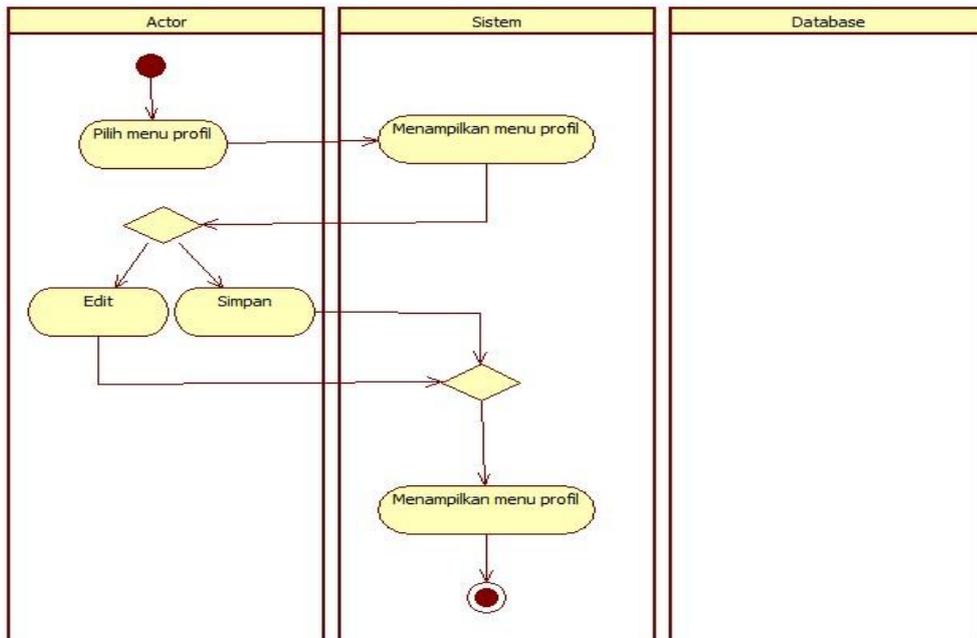
Pada menu ini user dapat memilih dan mencari produk sesuai jenis, merk dan nama produk yang ingin dicari. Selanjutnya aplikasi akan menampilkan hasil produk yang dicari dengan melakukan pengclusteringan produk pada database menggunakan *Algoritma Squeezer*.



Gambar 3.4 Activity Diagram Menu Pencarian Produk

#### 4. Activity Diagram Profil User

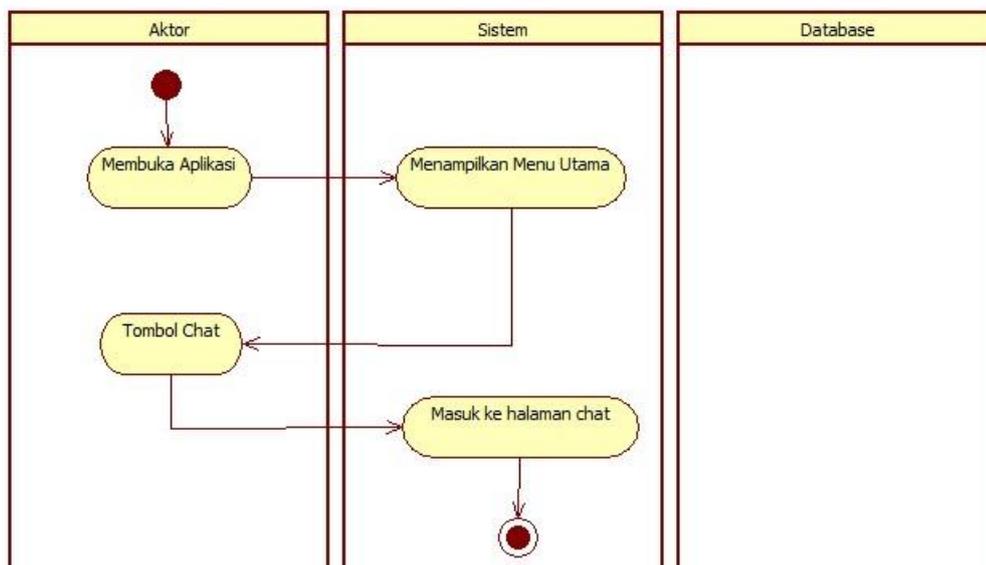
Pada menu ini user dapat mengubah dari informasi profil pengguna.



Gambar 3.5 Activity Diagram Profil User

#### 5. Activity Diagram Chat

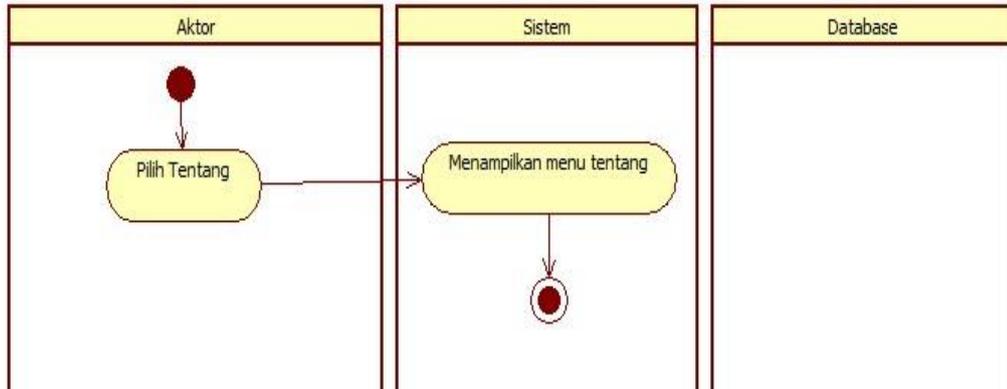
Pada menu tombol *chat* pengguna dapat melakukan komunikasi melalui fitur tersebut yang tersambung pada aplikasi *Whatsapp*.



Gambar 3.6 *Activity Diagram Chat*

6. *Activity Diagram* Tentang

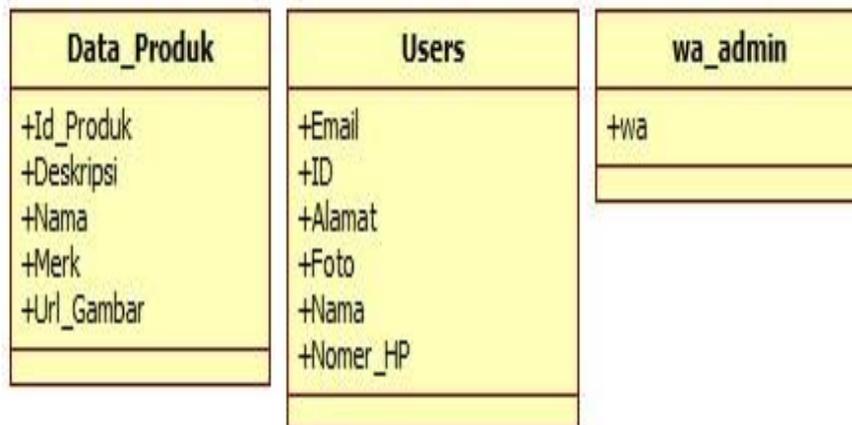
Pada menu tentang Admin dan User dapat melihat informasi sejarah dan visi misi dari MP One Stationery.



Gambar 3.7 *Activity Diagram* Tentang

7. *Class Diagram*

Kelas diagram adalah suatu diagram UML yang menggambarkan kelas-kelas dalam sebuah sistem dan hubungannya antara kelas diagram satu dengan yang lain, juga dimasukkan atribut dan operasinya.



Gambar 3.8 *Class Diagram*

## 8. Struktur *Database*

Berikut struktur database dari aplikasi MP One Stationery terdiri dari :

Tabel 3.3 Tabel Data Produk

<b>Key</b>	<b>Value</b>
Id_produk	Berisi kode acak dari firebase untuk menyimpan data
Deskripsi	Berisi deskripsi dari produk yang disimpan
Nama	Berisi nama produk yang disimpan
Merk	Berupa merk dari produk
Url_Gambar	Berisi url gambar produk

Tabel 3.4 Tabel Users

<b>Key</b>	<b>Value</b>
Email	Berisi alamat email pengguna aplikasi
Id	Berisi id pengguna aplikasi
Alamat	Berisi alamat pengguna aplikasi
Foto	Berisi foto dari pengguna
Nama	Berisi nama pengguna aplikasi
Nomer_HP	Berisi nomer HP pengguna aplikasi

Tabel 3.5 Tabel Wa\_Admin

<b>Key</b>	<b>Value</b>
Wa	Berisi nomer whatsapp admin

## 9. *Design Interface*

Desain adalah tahap merancang tampilan (*Interface*) aplikasi dan kebutuhan atau bahan yang dibutuhkan untuk membuat aplikasi tersebut. Adapun *Design Interface* pada aplikasi E-Katalog MP One Stationery di Kota Metro sebagai berikut :

a. Rancangan Tampilan Aplikasi *User*

Langkah selanjutnya setelah membuat *Use Case Diagram* dan *Activity Diagram* adalah membuat desain *Interface* aplikasi. Desain ini berguna untuk menentukan detail desain aplikasi yang akan digunakan untuk tampilan aplikasi tersebut.

1. Rancangan Desain *Interface Splashscreen*

Pada halaman ini akan muncul tampilan splash sebelum user memulai aksi dari aplikasinya.

Gambar 3.9 Pada halaman berikut merupakan rancangan tampilan *Splash* aplikasi E-Katalog pada *MP One Stationery* Kota Metro

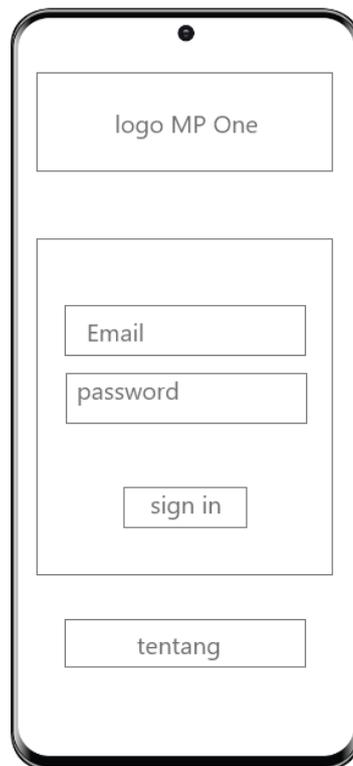


Gambar 3.9 Rancangan Desain *Interface Splashscreen*

## 2. Rancangan Desain *Interface Menu Login User*

Halaman menu login merupakan halaman untuk *User* dan *Admin Login*. Yang akan di tampilkan pada halaman ini kolom *Username* dan *Password*.

Gambar 3.10 Pada halaman berikut merupakan rancangan tampilan *Menu Login* aplikasi E-Katalog pada MP One Stationery Kota Metro.

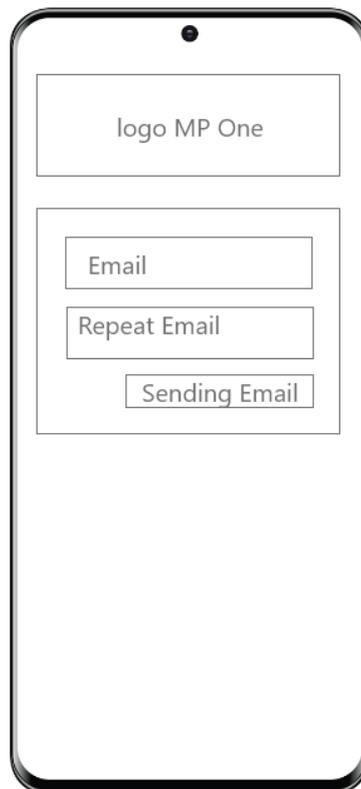


Gambar 3.10 Rancangan Desain *Interface Menu Login User*

## 3. Rancangan Desain *Interface Forget Password*

Pada halaman ini akan menampilkan form kolom email yang harus diisi untuk merubah kata sandi pengguna dengan mengirimkan notifikasi ke alamat email tersebut.

Gambar 3.11 Pada halaman berikut merupakan rancangan tampilan *Forget Password User* aplikasi E-Katalog pada MP One Stationery Kota Metro.

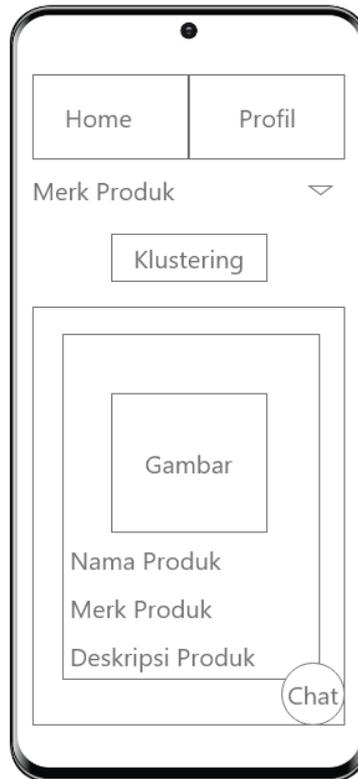


Gambar 3.11 Rancangan Desain *Interface Forget Password*

#### 4. Rancangan Desain *Interface Menu Utama User*

Halaman menu utama merupakan halaman yang pertama kali tampil saat pengguna membuka aplikasi. Yang akan di tampilkan pada halaman ini terdiri dari Menu Profil, Menu Produk, dan *Brand* Produk yang di jual.

Gambar 3.12 Pada halaman berikut merupakan rancangan tampilan Menu Utama aplikasi E-Katalog pada MP One Stationery Kota Metro.



Gambar 3.12 Rancangan Desain *Interface Menu Utama User*

#### 5. Rancangan Desain *Interface Menu Profil User*

Pada halaman ini yang akan di tampilkan adalah profil dari *User* yang terdiri dari Nama Produk, Merk Produk dan Deskripsi Produk.

Gambar 3.13 Pada halaman ini merupakan rancangan tampilan *Menu Profil User* aplikasi E-Katalog pada MP One Stationery Kota Metro

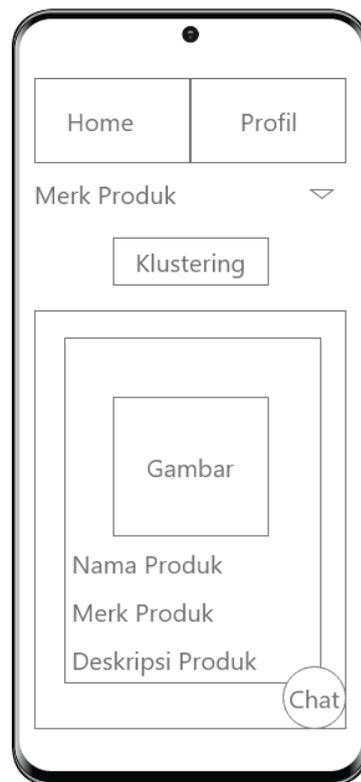


Gambar 3.13 Rancangan Desain *Interface Menu Profil User*

#### 6. Rancangan Desain *Interface Menu Produk*

Halaman menu produk akan menampilkan produk beserta informasi yang dijual oleh toko MP One Stationery Kota Metro serta mesin pencari produk agar memudahkan pengguna dalam mencari barang juga menu yang akan menampilkan kategori produk dengan melakukan *clustering* produk yang di cari menggunakan *Algoritma Squeezer*. Dilengkapi tombol chat untuk interaksi dengan admin yang di *link* ke via *whatsapp*.

Gambar 3.14 Halaman ini merupakan rancangan tampilan *Menu Produk* yang akan di tampilkan pada aplikasi E-Katalog



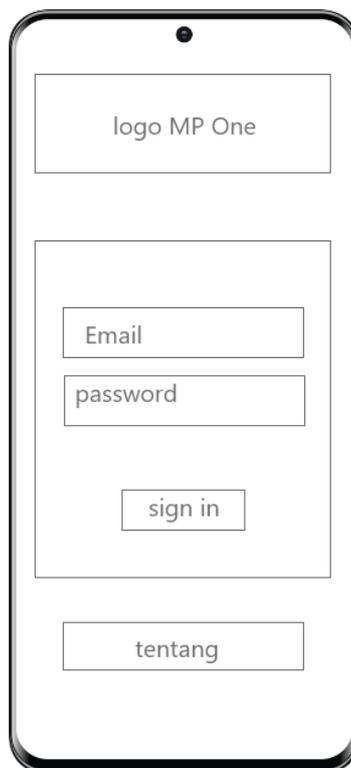
Gambar 3.14 Rancangan Desain Interface Menu Produk

b. Rancangan Tampilan Aplikasi *Admin*

1. Rancangan Desain *Interface Menu Login*

Untuk bisa mengakses aplikasi E-Katalog *admin* harus *login* terlebih dahulu. Dengan mengisi kolom *username* dan *password*.

Gambar 3.15 Pada halaman berikut merupakan rancangan tampilan *Menu Login Admin* aplikasi E-Katalog pada MP One Stationery Kota Metro.

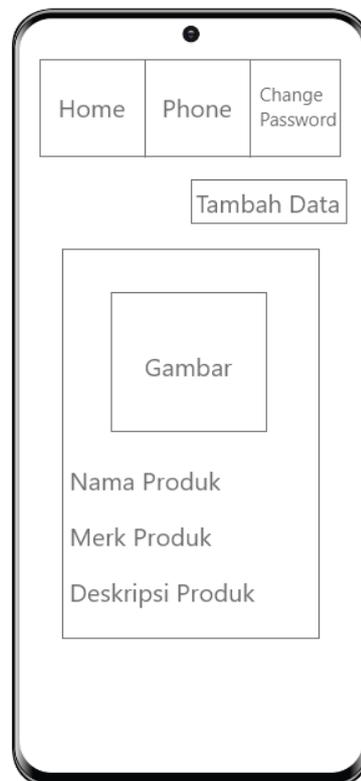


Gambar 3.15 Rancangan Desain *Interface Menu Login Admin*

## 2. Rancangan Desain *Interface* Halaman Utama Admin

Berikut tampilan rancangan halaman utama untuk *admin* yang menampilkan data produk, *menu change password* dan *menu* menambahkan data.

Gambar 3.16 Pada halaman berikut merupakan rancangan tampilan Halaman Utama *Admin* aplikasi E-Katalog pada MP One Stationery Kota Metro.

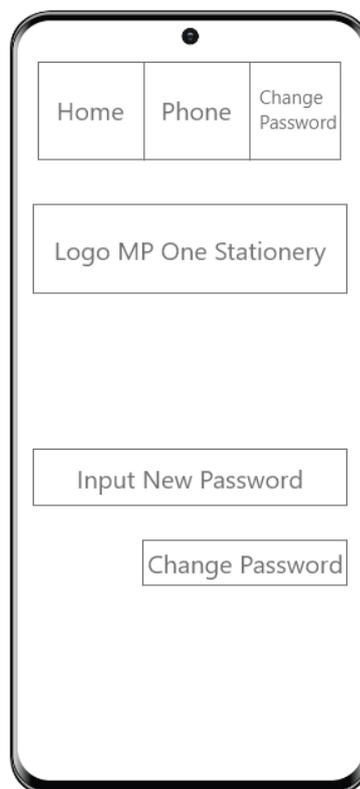


Gambar 3.16 Rancangan Desain *Interface* Halaman Utama Admin

### 3. Rancangan Desain *Interface Change Password*

Berikut tampilan rancangan halaman *Change Password* dimana *admin* bisa merubah kata sandi yang baru untuk akunnya tersebut.

Gambar 3.17 Pada halaman berikut merupakan rancangan tampilan *Change Password Admin* aplikasi E-Katalog pada MP One Stationery Kota Metro.

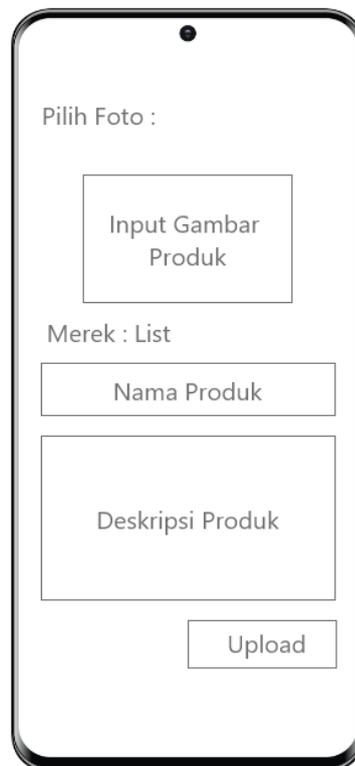


Gambar 3.17 Rancangan Desain *Interface Change Password Admin*

#### 4. Rancangan Desain *Interface Menu* Tambah Data

Berikut tampilan rancangan *menu* tambah data yang mana *admin* dapat menambahkan dan mengunggah data produk melalui aplikasi.

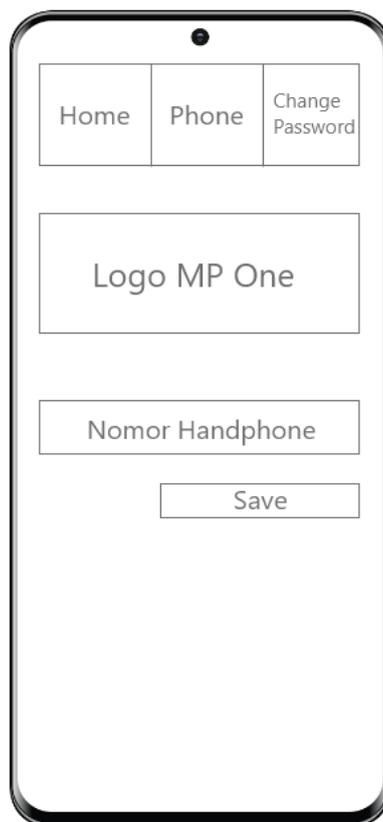
Gambar 3.18 Pada halaman berikut merupakan rancangan tampilan *Menu* Tambah Data aplikasi E-Katalog pada MP One Stationery Kota Metro.



Gambar 3.18 Rancangan Desain *Interface Menu* Tambah Data

5. Rancangan Desain *Interface Menu Phone Admin*  
Berikut tampilan rancangan *menu phone* yakni menu untuk mengubah nomor admin yang terhubung ke dalam via *whatsapp*.

Gambar 3.19 Pada halaman berikut merupakan rancangan tampilan *Menu Phone Admin* aplikasi E-Katalog pada MP One Stationery Kota Metro.



Gambar 3.19 Rancangan Desain *Interface Menu Phone*

### 3.2.3 Clustering produk dengan Algoritma Squeezer

Pada tahap pengklusteran produk, Algoritma Squeezer melakukan proses perbandingan nama produk yang dicari dengan nama produk yang ada di *data base*. Saat nama produk yang di cari sesuai dengan data pada database, maka sistem akan menampilkan hasil produk.

```
public class algoritmanya extends Fragment {  
  
    FirebaseAuth firebaseAuth;  
    FirebaseUser User;  
    DatabaseReference dataRef,dataRef2;  
    RecyclerView list;  
  
    Button cluster;  
    FloatingActionButton chat;  
  
    EditText opsinya;  
  
    ArrayList<String> namalist;  
    ArrayList<String> merklist;  
    ArrayList<String> deskripsilist;  
    ArrayList<String> keylist;  
    ArrayList<String> imageURL;  
  
    com.ian.aplikasiku.model.list_produk list_produk;  
  
    @Override  
    public View onCreateView(LayoutInflater inflater, ViewGroup container, Bundle savedInstanceState) {  
        View view = inflater.inflate(R.layout.algoritma, container, attachToRoot false);  
  
        firebaseAuth = FirebaseAuth.getInstance() ;  
        User = firebaseAuth.getCurrentUser();  
    }  
}
```

Gambar 3.20 Penerapan Algoritma Squeezer

Penerapan Algoritma Squeezer pada aplikasi MP One Stationary akan melakukan pengecekan produk yang dicari apakah terdapat data yang dapat dibaca secara berkelanjutan pada *index* produk atau urutan produk yang di tampilkan pada hasil pencarian. Jika *input* produk yang dimasukkan memiliki kemiripan dengan beberapa produk yang ada di *database*, maka akan di ambil hasil yang paling besar kemiripannya dan *index cluster* dengan kemiripan yang paling besar. Jika input barang yang

dicari tidak memiliki kemiripan dengan *database*, maka data yang ditampilkan kosong atau null. Tetapi, saat nama produk yang diinputkan pada kolom pencarian memiliki sedikit kemiripan, algoritma squeezer akan mengkluster produk dengan beberapa hasil sesuai keyword yang dimasukkan.

```
cluster = view.findViewById(R.id.cluster);

cluster.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        String pilihannya = opsinya.getText().toString();
        clusterring(pilihannya);
    }
});

chat = view.findViewById(R.id.chat);

chat.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        dataRef2.addListenerForSingleValueEvent(new ValueEventListener() {
            @Override
            public void onDataChange(@NonNull DataSnapshot snapshot) {
                String nomernya = snapshot.child("wa").getValue().toString();
                String url = "https://api.whatsapp.com/send?phone="+nomernya;
                Intent i = new Intent(Intent.ACTION_VIEW);
                i.setData(Uri.parse(url));
                startActivity(i);
            }

            @Override
            public void onCancelled(@NonNull DatabaseError error) {
            }
        });
    }
});

Load_data();
```

Gambar 3.21 Proses pengklusteran *Algoritma Squeezer*

### 3.2.4 Pembentukan *Prototype*

Pada tahap ini pembentukan prototype, yaitu pembuatan perangkat prototype termasuk dalam pengujian dan penyempurnaan. Pengujian dilakukan untuk menguji kelayakan fungsi tombol, *interface* dan respon aplikasi saat digunakan oleh *user*. Pada tahap ini dalam melakukan pembuatan aplikasi menggunakan android studio.

### 3.2.5 Penyerahan Sistem pada Pengguna

Pada tahap ini *user* akan menguji coba aplikasi. Apabila program yang diciptakan belum sesuai dengan metode prototype, maka program akan diperbaiki kembali, begitu seterusnya hingga program benar-benar sesuai dengan keinginan user.