

## BAB 3

### METODOLOGI PENELITIAN

#### 3.1 Metodologi Penelitian

Pada penelitian ini, metodologi penelitian yang digunakan ialah Design Science Research Methodology (DSRM) yang terdiri dari 6 tahapan yaitu identifikasi masalah dan motivasi, menentukan tujuan dari solusi, perancangan dan pengembangan, implementasi, evaluasi dan komunikasi (Fernando, Surjandy, Meyliana, & Siagian, 2020). Berikut merupakan tahapan dari metodologi yang digunakan dalam penelitian ini.

1. Identifikasi Masalah dan Motivasi

Tahap awal pada DSRM ialah mengidentifikasi masalah yang menjadi latar belakang dilakukannya penelitian ini.

2. Menentukan Tujuan

Tahap ini ialah menentukan tujuan dari permasalahan yang didapat setelah melakukan identifikasi masalah dari hasil penelitian ini. Tujuan tersebut diharapkan dapat mendukung solusi untuk masalah yang tidak ditangani.

3. Perancangan dan pengembangan

Tahapan ini didasarkan dari hasil solusi yang didapat dari tahapan sebelumnya yaitu melakukan perancangan *system* dengan membuat *mockup user interface* dan *flowchart* diagram serta melakukan pengembangan sistem dengan menggunakan metode waterfall agar didapatkan solusi yang sesuai.

4. Implementasi

Implementasi yang dilakukan yaitu dengan melakukan pemrograman sistem

yang menggunakan Bahasa pemrograman Java pada bagian backend dan XML digunakan pada bagian frontend dalam pembuatan layout.

## 5. Evaluasi Sistem

Pada tahap ini dilakukan pengujian dan analisis dari perancangan dan pembangunan pada implementasi yang sudah dilakukan. Pengujian ini dilakukan dengan menggunakan testing sistem berdasarkan pada metode Black Block Testing serta untuk mendapatkan hasil dari Analisa evaluasi sistem akan menggunakan teori Technology Acceptance Model (TAM) untuk melakukan kuisisioner dan hasil kuisisioner akan dianalisa menggunakan skala Likert.

## 6. Komunikasi

Pada tahap ini, melakukan komunikasi untuk merancang struktur yang benar dalam penulisan yang akan dipublikasikan kedepannya.

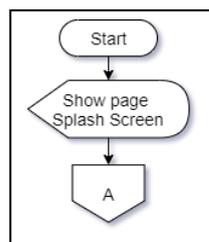
### 3.2 Perancangan Aplikasi

Perancangan aplikasi dibuat untuk menjadi dasar arsitektur sistem secara keseluruhan dan sebagai acuan dalam pengerjaan aplikasi yang didasari dari metode Waterfall.

#### 3.2.1 Flowchart

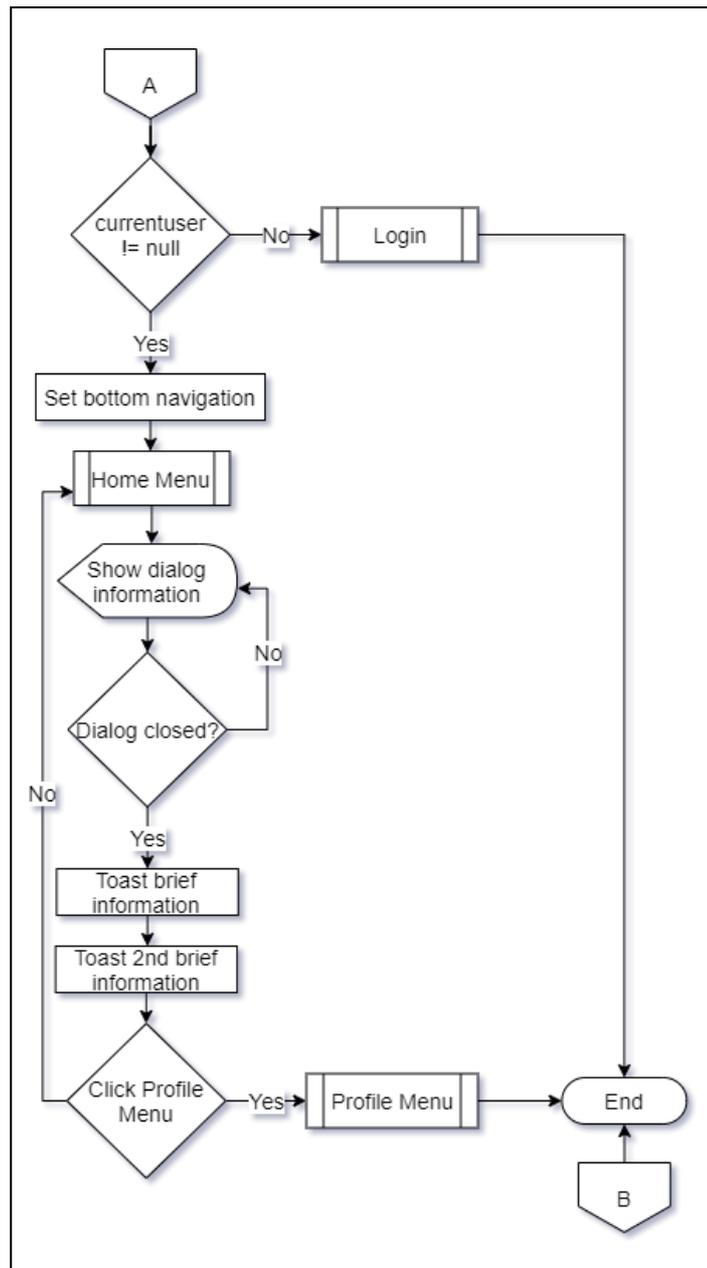
Berikut ini adalah flowchart diagram yang menunjukkan alur proses kerja dari setiap modul yang dibuat.

##### A. Splashscreen



Gambar 3.1 Flowchart Splashscreen

Gambar 3.1 merupakan proses *flowchart* dari Splashscreen. *Flowchart* ini merupakan diagram yang menunjukkan proses ketika *user* membuka aplikasi. Pada splash screen ini, *user* akan ditampilkan logo aplikasi dan nama aplikasi pada bagian tengah halaman. Halaman splashscreen ini akan mengarah ke-main activity (Connector symbol A).

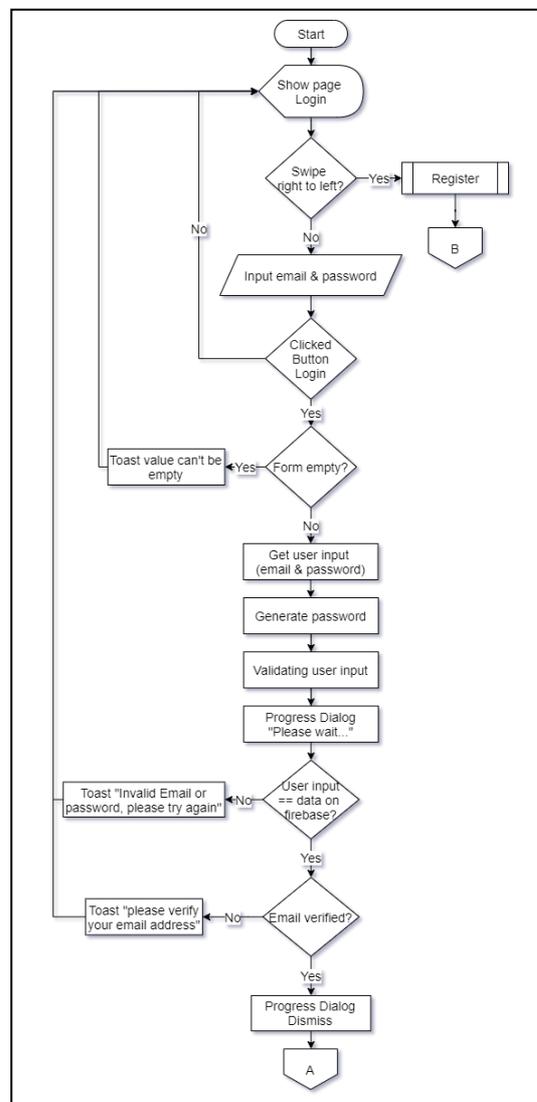


Gambar 3.2 Flowchart Splashscreen (Lanjutan\_ConnectorA)

Gambar 3.2 merupakan proses *flowchart* dari Splashscreen (Lanjutan).

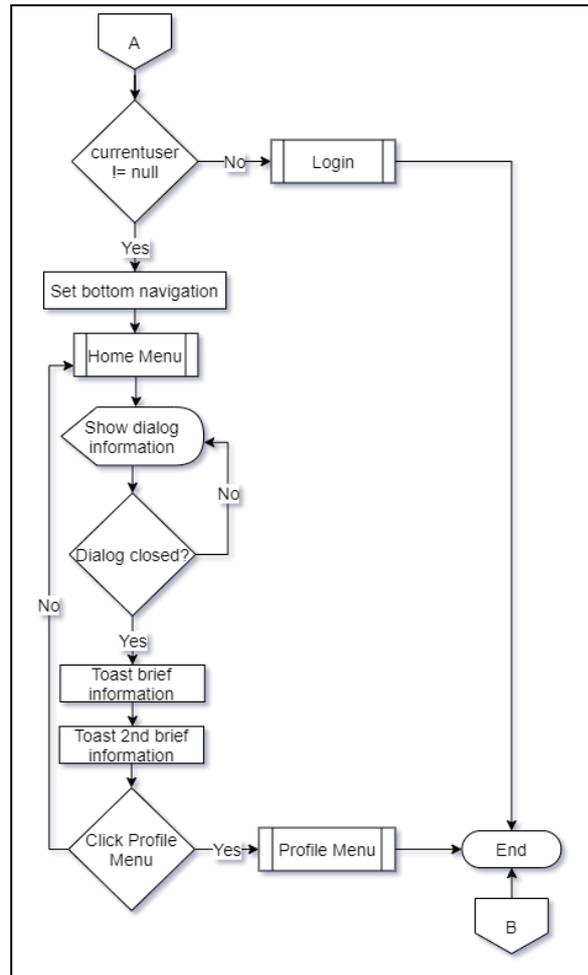
Dimana main activity ini akan mengecek data *user*, jika *user* sudah melakukan *login* sebelumnya maka *user* akan diarahkan ke halaman home namun jika *user* belum melakukan *login* sebelumnya maka *user* akan diarahkan ke halaman *login*. Pada *flowchart* ini juga mengacu pada proses setelah *home menu* terbuka, yaitu dengan memunculkan *bottom navigation*, *dialog how to use*, *close dialog how to use*, terdapat 2 *toast brieft information*, dan proses *clicked button menu* untuk *navigasi* ke halaman *profile* menu atau *home* menu.

## B. Login



Gambar 3.3 Flowchart Login

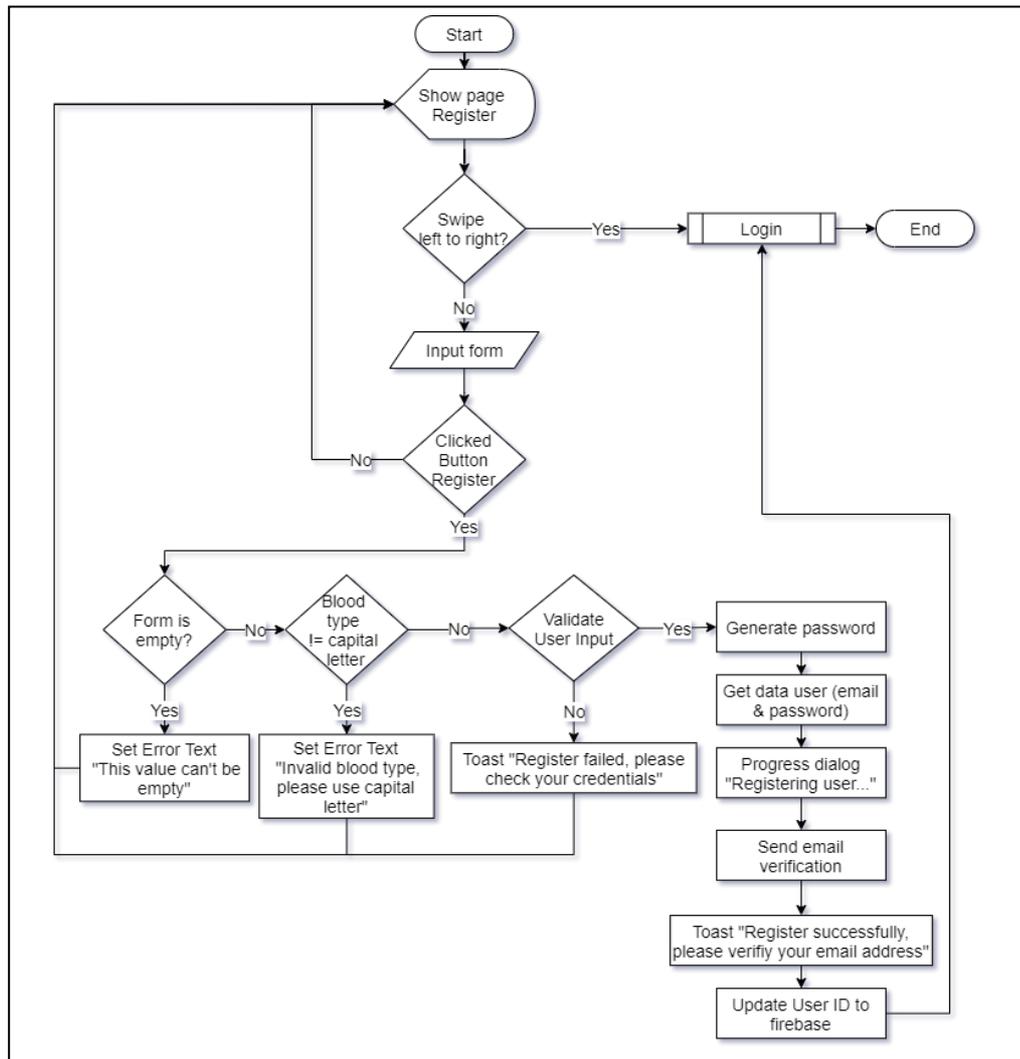
Gambar 3.3 merupakan proses *flowchart* dari *Login*. *Flowchart* ini merupakan diagram yang menunjukkan proses ketika *user* ingin memasuki main menu aplikasi. Pada *login* ini, *user* akan ditampilkan nama aplikasi, *icon alert user input*, *button login* dan informasi untuk mengarah ke halaman *register*. *User* dapat melakukan proses *login* dihalaman ini atau melakukan *register* terlebih dahulu dihalaman *register* dengan menggeser dari halaman *login* ke arah kanan ke kiri. Namun jika *user* telah memiliki akun, maka *user* dapat mengisi *form login*-nya secara langsung dan *user* dapat menekan tombol *login* jika data yang diisi sudah sesuai dengan data akun yang telah diregister sebelumnya. Disini sistem akan mengambil data inputan *user* dan mengecek ke-firebase untuk melakukan *validating user input*. Selama nunggu pengecekan data *user input*, sistem akan menampilkan *progress dialog* yang isinya *loading* dan *text* "Please wait". Setelah pengecekan sudah selesai, jika data *user* sudah benar dan sudah terverifikasi emailnya maka *user* akan diarahkan ke *flowchart* main activity (connector symbol A) lalu *progress dialog* akan ter-*dismiss*, namun jika data *user* tidak sesuai atau belum terverifikasi emailnya maka *user* akan dimunculkan *toast message*.



Gambar 3.4 Flowchart Login (Lanjutan\_ConnectorA)

Gambar 3.4 merupakan proses *flowchart* dari Login (Lanjutan). *Flowchart* ini merupakan diagram yang menunjukkan proses main activity sedang melakukan pengecekan data *user* yang berada di-firebase, Jika data *user* terdapat di-firebase dan sudah melewati tahap *validate* dihalaman *login* maka sistem akan mengarahkan *user* kehalaman *home*, namun jika data *user* tidak sesuai dengan yang ada di-firebase maka sistem akan mengarahkan kembali ke halaman login. Pada *flowchart* ini juga mengacu pada proses setelah *home* menu dibuka, yaitu dengan memunculkan *bottom navigation*, *dialog how to use*, *close dialog how to use*, terdapat 2 *toast brieft information*, dan proses *clicked button menu* untuk navigasi ke halaman *profile* menu atau *home* menu.

### C. Register

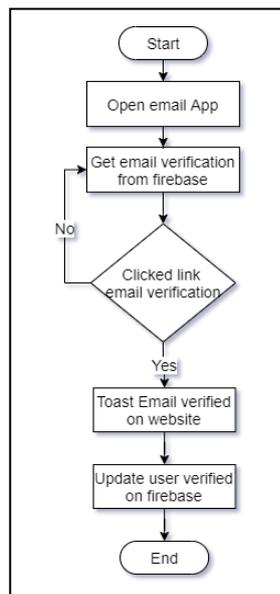


Gambar 3.5 Flowchart Register

Gambar 3.5 merupakan proses *flowchart* dari Register. *Flowchart* ini merupakan diagram yang menunjukkan proses ketika *user* ingin mendaftarkan akunnya pada aplikasi. Pada *register* ini, *user* akan ditampilkan *form register*, *button register* dan informasi untuk mengarah ke halaman *login*. *User* dapat melakukan proses *register* dihalaman ini atau kalau sudah punya akun dapat mengisi ke halaman *login* dengan cara mengeser halaman *register* dari arah kiri kekanan. *User* dapat mengisi *form register* dan setelah data *register* sudah ter-input dengan benar maka *user* dapat menekan tombol *register*. Dalam bagian ini, sistem

akan mengecek *user* input, jika data yang diisi kosong atau tidak sesuai dengan ketentuan yang ada maka sistem akan memunculkan toast *error message* namun jika sistem sudah berhasil mengecek data *user* dengan benar maka sistem akan meng-*generate password* terlebih dahulu dan mengirimkannya data barunya ke firebase. Selama nunggu pengecekan data input *user*, sistem akan menampilkan *progress dialog* yang isinya *loading* dan text “Registering user . . .”. Jika data *user* sudah masuk di-firebase maka *user* akan dikirimkan email *verification* untuk verifikasi akun *user* serta *user* akan dimunculkan toast “Register successfully, please verify your email address” yang artinya pada toast ini *user* diminta untuk melakukan verifikasi pada email *user*.

#### D. Email Verification

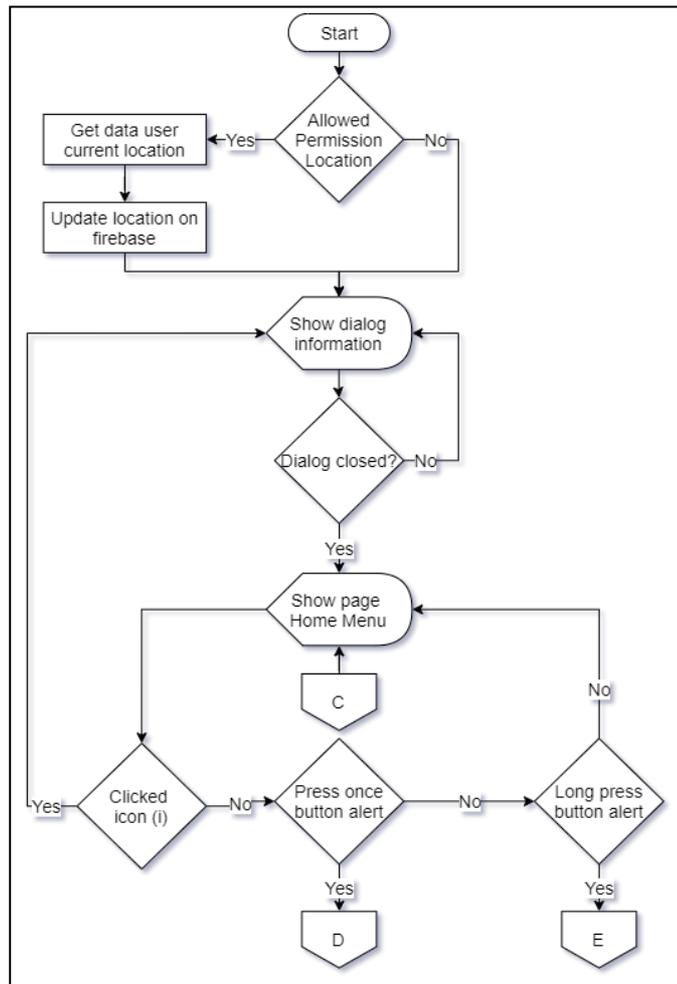


Gambar 3.6 Flowchart Email Verification

Gambar 3.6 merupakan proses *flowchart* dari Email Verification. *Flowchart* ini merupakan diagram yang menunjukkan proses ketika *user* ingin melakukan verifikasi email pada app email. Verifikasi email ini dilakukan jika user sudah berhasil melakukan pendaftaran akun di halaman register pada aplikasi emergency

alert. Proses ini dapat dilakukan dengan cara *click link* yang sudah dikirimkan oleh sistem (firebase) secara otomatis ke-email *user* yang sudah didaftarkan sebelumnya, ketika *user* sudah menekan link verifikasi tersebut maka *user* akan dimunculkan message “Email verified” lalu sistem akan mengupdate data *user* di-firebase agar *user* dapat melakukan *login* pada halaman *login*.

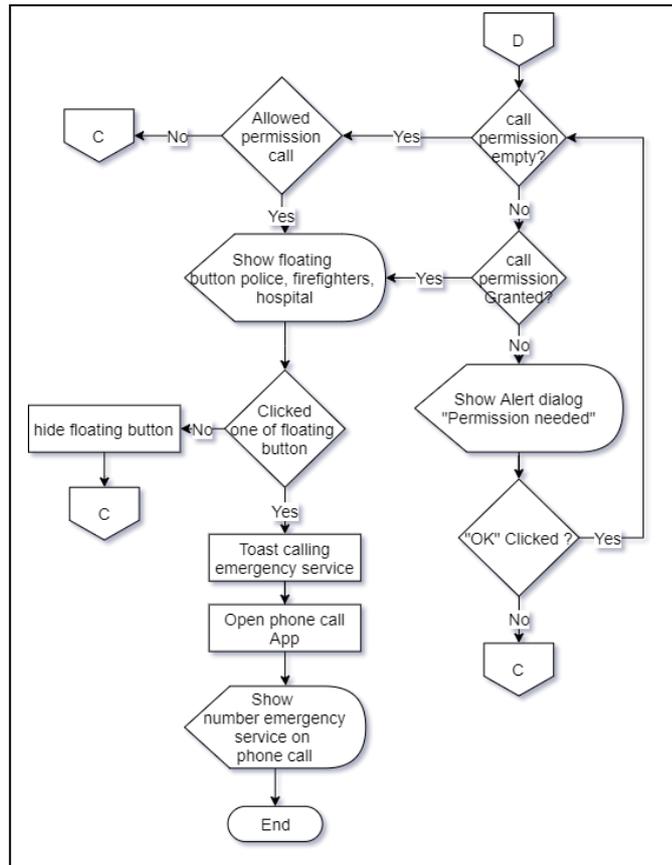
### E. Home Menu



Gambar 3.7 Flowchart Home Menu

Gambar 3.7 merupakan proses *flowchart* dari home menu. *Flowchart* ini merupakan diagram yang menunjukkan proses ketika *user* sudah melakukan login sebelumnya. Pada *home* menu ini, *user* akan ditampilkan terlebih dahulu untuk melakukan persetujuan pada *permission location*. *Permission location* ini bertujuan

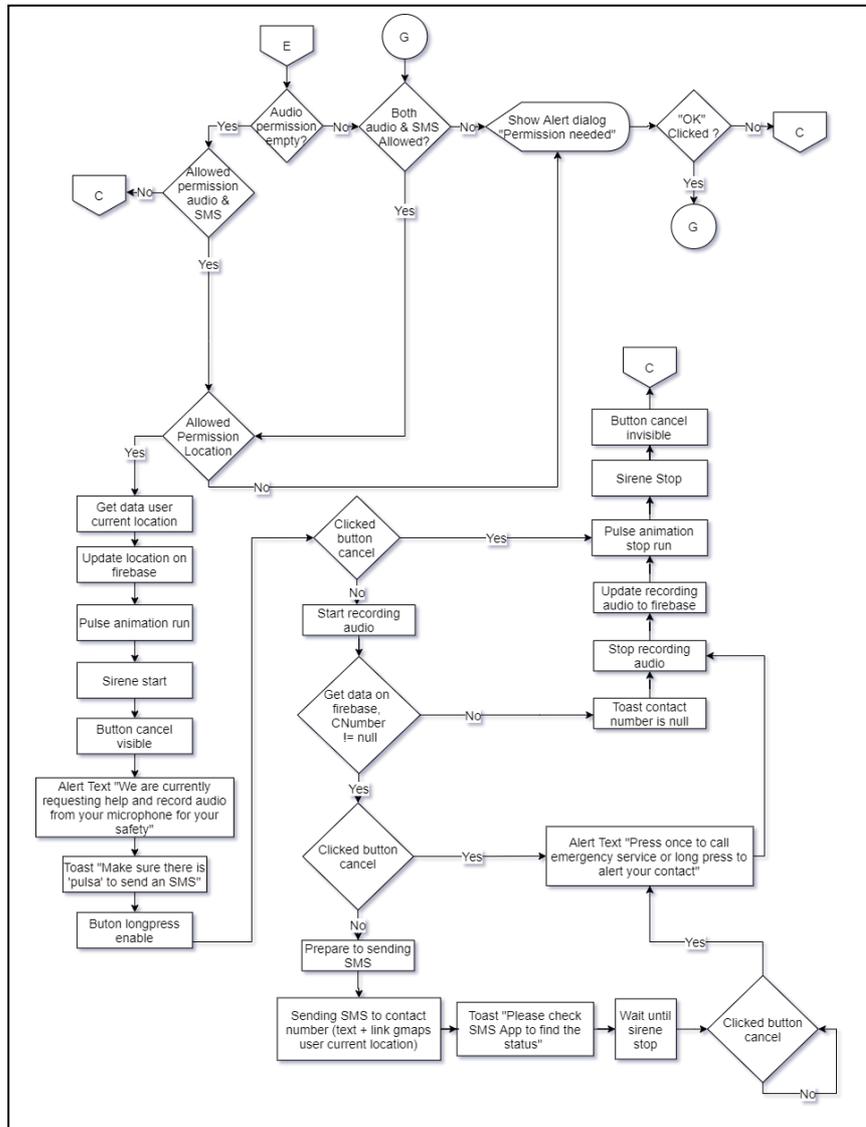
untuk mengakses data lokasi *user*, namun jika *user* tidak memberikan ijin pada *permission* ini *user* akan ditampilkan kembali *permission* ini pada saat melakukan *longpress button* pada bagian *body home* menu. Selanjutnya setelah melakukan proses *permission location*, *user* akan ditampilkan *dialog how to use*. Pada *dialog how to use* bertujuan untuk menginformasikan cara penggunaan aplikasi *emergency alert* ini kepada *user*. Ketika *user* sudah selesai membaca informasi cara penggunaan aplikasi ini maka *user* dapat mengklik icon(x) pada bagian kanan atas *dialog*. Setelah *dialog how to use* sudah tertutup, maka *user* akan ditampilkan isi dari halaman *home* menu. Pada bagian ini, *user* dibebaskan untuk memilih dari ketiga fungsi utama dihalaman *home* menu ini yaitu seperti mengklik icon (i) untuk melihat *dialog how to use* kembali, melakukan *press once button alert* untuk menghubungi *emergency service* (Connector symbol D), dan *longpress* untuk memberitahukan keadaan *user* ke orang-orang terdekat yang berada dikontak bahwa *user* sedang dalam bahaya dengan cara mengirimkan pesan darurat seperti *text* dan link *gmaps* (*user current location*) lewat *SMS Alert* (Connector symbol E). Pada pelaksanaan diproses *press once button alert* atau *long press button alert* ini, sistem akan berjalan dengan lancar jika *user* sudah mengikuti *step-step* yang ada dari aplikasi seperti menyetujui *permission* dari *location*, *audios*, *SMS*, dan *call*. Jika *user* menolak ijin pada *permission* tersebut maka sistem akan terus berulang melakukan *alert dialog* sampai *user* dapat menyetujui pada *permission* tersebut.



Gambar 3.8 Flowchart Home Menu (Lanjutan\_ConnectorD)

Gambar 3.8 merupakan proses *flowchart* dari home menu. *Flowchart* ini merupakan diagram yang menunjukkan proses ketika *user* ingin melakukan *press once button alert*. Pada proses ini menunjukkan jika *user* sudah melakukan persetujuan di *permission call* ini, maka *user* akan ditampilkan *floating button emergency service*. Namun jika *user* tidak menyetujui pada *permission* ini, maka *user* akan ditampilkan *alert dialog* yang isinya “*Permission needed*” dengan tujuan *user* dapat melakukan persetujuan pada *permission* ini agar *user* dapat melanjutkan proses berikutnya. Tujuan *permission* yang ditampilkan pada aplikasi ini, agar *user* dapat menggunakan fitur *device* yang akan digunakan pada aplikasi ini. Ketika *user* sudah melakukan persetujuan di-*permission call*, maka *user* akan ditampilkan *floating button emergency service* dan *user* dapat menekan salah satu *floating button* tersebut lalu *user* akan diarahkan ke *phone call* sambil dimunculkan *toast*

message “calling emergency service”, setelah app *phone call* sudah terbuka maka *user* akan diberikan nomor *emergency service* yang dituju dan *user* dapat menelepon nomor dari *emergency service* tersebut.



Gambar 3.9 Flowchart Home Menu (Lanjutan\_ConnectorE)

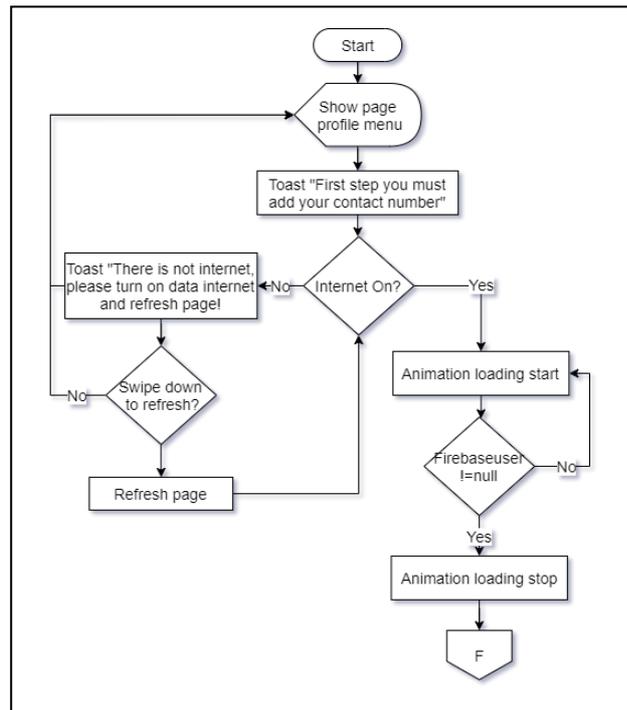
Gambar 3.9 merupakan proses *flowchart* dari home menu. *Flowchart* ini merupakan diagram yang menunjukkan proses ketika *user* ingin melakukan *long press button alert*. Pada proses ini *user* akan ditampilkan terlebih dahulu pada permission-permission yang ada seperti Audio, SMS, dan location. Tujuan *permission* yang ditampilkan pada aplikasi ini, agar *user* dapat menggunakan fitur

*device* yang akan digunakan pada aplikasi ini. Jika *user* tidak menyetujui pada *permission* yang ada, maka user akan ditampilkan alert dialog yang bertujuan agar *user* dapat melakukan persetujuannya di-*long press button alert* yang berikutnya.

Ketika user sudah melakukan persetujuan pada *permission* yang ada, sistem akan melakukan pengecekan nomor kontak terlebih dahulu. Jika *user* belum menyimpan kontak orang terdekatnya dihalaman profile maka sistem tidak bisa melanjutkan proses *longpress button alert* namun jika *user* sudah memasukan kontak tersebut dihalaman profile maka *user* akan ditampilkan animasi *pulse* dengan tulisan “we are currently requesting help and record audio from your microphone for your safety”. Pada bagian ini sistem akan memunculkan tombol *cancel*, jika *user* menekan tombol cancel maka sistem akan memberhentikan segala proses di-*longpress button* ini dan sistem akan mengupdate file audios yang telah direkam pada firebase storage. Namun jika user tidak menekan tombol *cancel* maka sistem akan melanjutkan proses di-*longpress button* ini seperti memunculkan suara *sirene*, *recording audios*, memunculkan toast “Make sure there is ‘pulsa’ to send an sms, dan pengiriman pesan kepada orang-orang yang berada dikontak aplikasi.

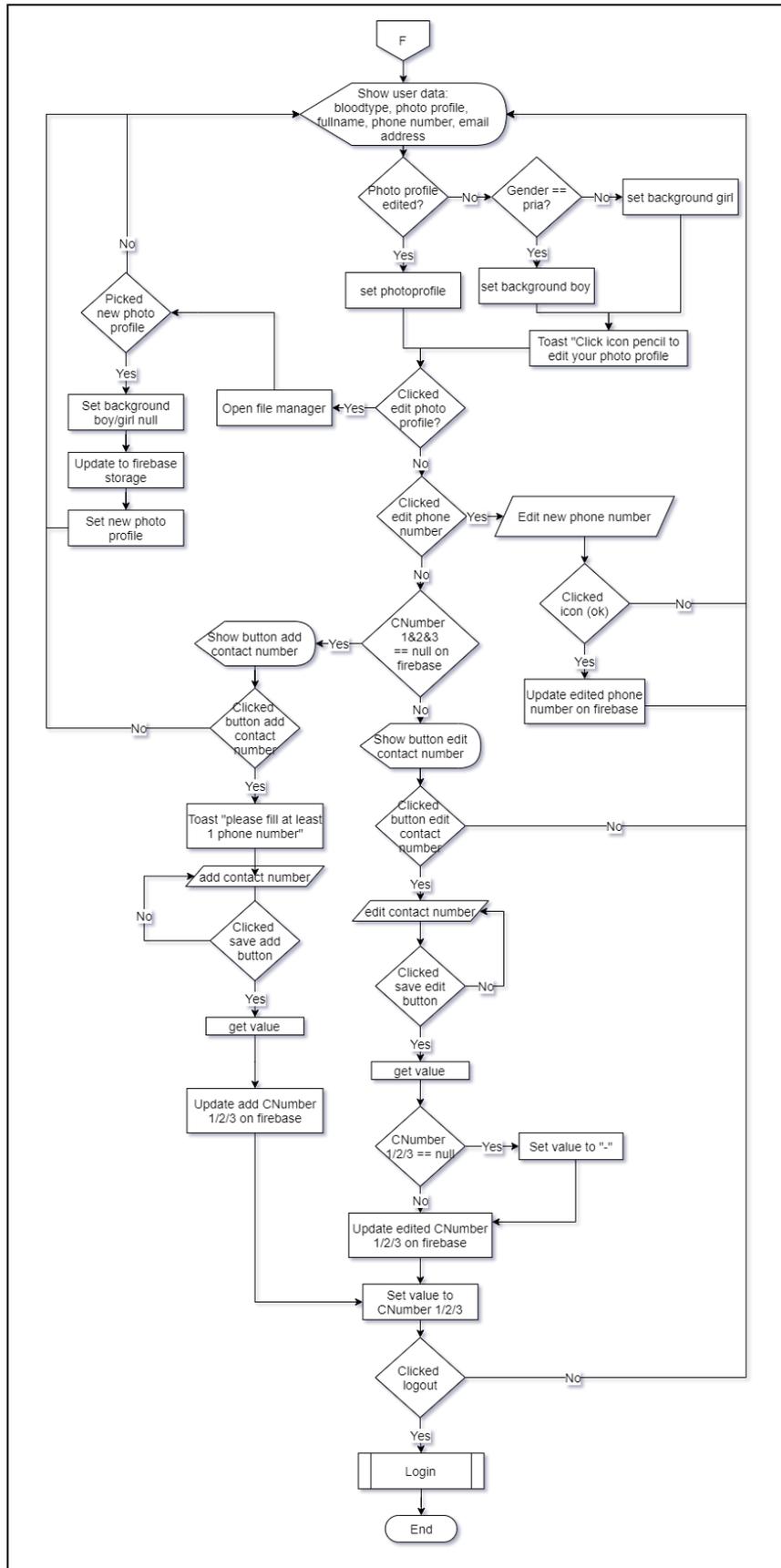
Ketika sistem berhasil melakukan pengiriman pesan darurat kepada orang-orang yang berada di dalam kontak user, sistem akan memunculkan toast message “Please check SMS App to find the status”. Pesan darurat yang dikirimkan yaitu *text* dan link gmaps (user current location).

## F. Profile Menu



Gambar 3.10 Flowchart Profile Menu

Gambar 3.10 merupakan proses *flowchart* dari *profile* menu. *Flowchart* ini merupakan diagram yang menunjukkan proses ketika *user* ingin melihat data diri diaplikasi ini. Pada saat *user* membuka halaman *profile* menu ini *user* akan dicek oleh sistem bahwa *user* telah menyalakan internet, namun jika internet *user* mati maka sistem akan memunculkan *toast* serta animasi *loading* untuk meng-*load* data *user* dari *firebase*. Ketika *user* sudah menyalakan internetnya kembali dan ingin melihat data diri dihalaman *profile* maka *user* dapat meng-*refresh* halaman dengan cara *swipe down* dan sistem akan memunculkan animasi *loading* serta langsung mengambil data *user* dari *firebase* lalu animasi *loading* akan berhenti dengan sendirinya.



Gambar 3.11 Flowchart Profile Menu (Lanjutan\_ConnectorF)

Gambar 3.11 merupakan proses *flowchart* lanjutan dari *profile* menu. *Flowchart* ini merupakan diagram yang menunjukkan proses ketika *user* memasuki halaman *profile*. Data *user* yang akan ditampilkan pada halaman *profile* ini yaitu terdapat *bloodtype*, *photoprofile*, *fullname*, *phone number* dan *email address*. Disini sistem akan mengecek terlebih dahulu, apakah *user* sudah meng-*edit* data dirinya seperti *photoprofile*, *phone number*, dan *contact number*. Jika *user* belum mengubahnya sama sekali maka sistem akan menampilkan data terakhir yang *user input* pada saat *user* melakukan *register*. Dikarenakan pada bagian *photoprofile* *user* tidak menginput pada bagian *register*, maka sistem akan memunculkan gambar sesuai dari *gender user*. Dalam *page* ini *user* dapat meng-*edit* ketiga fitur tersebut dengan cara mengklik *icon pencil* pada bagian samping *photo profile*, yang nanti akan diarahkan ke file manager untuk memilih *image* yang akan dijadikan *photo profile* dan jika sudah terpilih maka sistem akan meng-*update photoprofile* tersebut di-*firebase storage*. Selanjutnya untuk meng-*edit phone number*, *user* juga dapat mengklik *icon pencil* pada bagian kanan *phone number* lalu *user* akan ditampilkan *phonenumber* yang siap di-*edit* dan *icon checklist* untuk menyimpan data *phone number* baru ke *firebase*. Disini *user* juga dapat menambahkan atau meng-*edit* dari *contact number* dengan menekan tombol *add/edit* di bawah form *contact number* dan *save* dengan mengklik *button save*. *Contact number* yang ada akan di-*update* ke dalam *firebase* dan *contact number* ini bertujuan untuk menyimpan kontak-kontak orang terdekat jika ada terjadi sesuatu yang darurat maka orang-orang inilah yang akan menjadi penolong *user*. Lalu setelah sudah selesai melakukan seluruh aktivitas di aplikasi ini, *user* dapat keluar dari aplikasi dengan menekan tulisan *logout* pada bagian kanan atas di halaman *profile page*.

### 3.2.2 Struktur Table

Tabel 3.1 Firebase Realtime Database

<b>Nama Kolom</b>	<b>Tipe Data</b>	<b>Keterangan</b>
User Id	String	<i>User Id</i>
CNumber1	String	No kontak orang terdekat pertama
CNumber2	String	No kontak orang terdekat kedua
CNumber3	String	No kontak orang terdekat ketiga
address	String	Alamat <i>user</i>
bloodType	String	Golongan darah <i>user</i>
email	String	Alamat email yang aktif untuk melakukan proses verifikasi <i>register</i> dan <i>login</i>
fullname	String	Nama lengkap <i>user</i>
gender	String	Jenis kelamin <i>user</i>
location	String	<i>Current location user</i>
Password	String	Kata sandi yang digunakan <i>user</i> dalam melakukan proses <i>login</i>
phone	String	Nomor telepon <i>user</i> yang digunakan untuk mengirimkan sms darurat kepada orang-orang yang ada berada di dalam kontak aplikasi
profilePicture	String	ImageUrl yang mengarah ke firebase Storage

Tabel 3.1 merupakan data *realtime* yang digunakan pada aplikasi *mobile* yaitu menggunakan firebase “Realtime Database”. Firebase “Realtime Database” ini digunakan sebagai komunikasi antar aplikasi *mobile* untuk mendapatkan data secara *realtime*.

Tabel 3.2 Firebase Storage

<b>Nama Kolom</b>	<b>Tipe Data</b>	<b>Keterangan</b>
User Id	String	Tempat <i>user</i> menyimpan file-file di-firebase storage
audios	Mp3	Tempat menyimpan <i>recorder audio</i> setelah <i>user</i> melakukan <i>longpress button alert</i>
profilePicture	Image/JPG	Tempat menyimpan <i>photo profile</i> setelah <i>user</i> melakukan <i>edit photo profile</i>

Tabel 3.2 merupakan data file yang digunakan pada aplikasi *mobile* yaitu menggunakan Firebase Storage. Firebase storage ini digunakan dalam melakukan penyimpanan file-file pada aplikasi mobile.

### 3.2.3 Mockup Aplikasi

Berikut ini adalah mockup aplikasi *emergency alert* yang akan dijadikan sebagai acuan dalam pembuatan setiap halamannya.

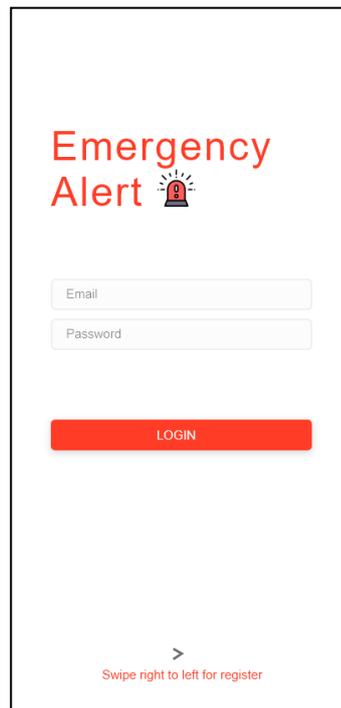
#### A. Splash Screen



Gambar 3.12 Mockup Splash Screen

Gambar 3.12 merupakan tampilan mockup pada halaman *splash screen* yang digunakan sebagai acuan tampilan awal pada aplikasi *emergency alert*. Halaman *splash screen* ini hanya menampilkan sebuah logo aplikasi yaitu *Emergency Alert*.

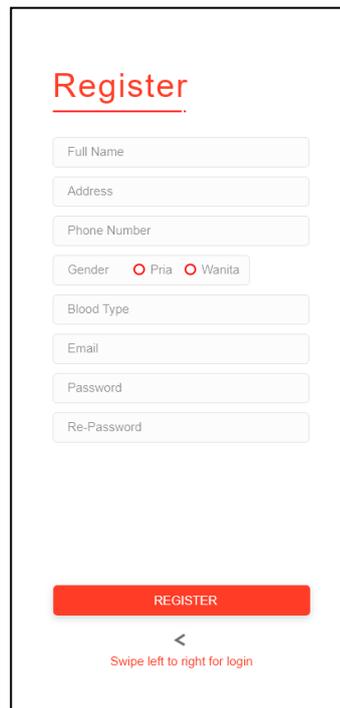
## B. Login



Gambar 3.13 Mockup Login

Gambar 3.13 merupakan tampilan *mockup* pada halaman *login* yang akan ditampilkan setelah halaman *splashscreen*. Halaman *login* ini, terdapat tulisan “Emergency Alert” dan icon *alert*. Kemudian diikuti dengan 2 kolom teks untuk mengisi *email* dan *password*. Pada di bawah kolom *password* terdapat tombol *login* untuk menjalankan fungsi *login* yang akan mengarah ke halaman *home* jika sudah sesuai dalam pengisian data *loginnya*. Namun jika *user* belum melakukan pengisian data, *user* dapat melakukan *registrasi* terlebih dahulu dengan menggeser halaman dari arah kanan ke kiri. *Text* intruksi untuk menuju ke halaman *registrasi* terdapat pada bawah tombol *login*.

### C. Register



The image shows a mobile application register form. At the top, the word "Register" is written in red and underlined. Below it are several input fields: "Full Name", "Address", "Phone Number", "Gender" (with radio buttons for "Pria" and "Wanita"), "Blood Type", "Email", "Password", and "Re-Password". At the bottom of the form is a red button labeled "REGISTER". Below the button is a left-pointing arrow and the text "Swipe left to right for login".

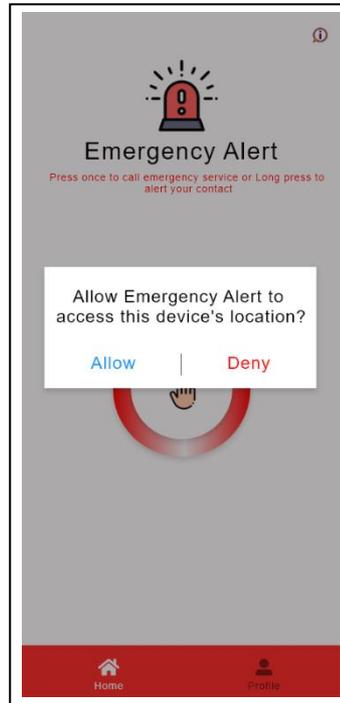
Gambar 3.14 Mockup Register

Gambar 3.14 merupakan tampilan *mockup* pada halaman *register* yang akan ditampilkan setelah *user* sudah menggeser halaman *login* dari arah kanan ke kiri. Halaman *register* ini, digunakan untuk mendaftarkan akun *user* pada aplikasi. Kemudian data yang akan di-*input* oleh *user* terdapat pilihan *gender* yaitu pria dan wanita dan 7 kolom teks untuk mengisi *fullname*, *address*, *phonenumber*, *bloodtype*, *email*, *password* dan *re-password*. Pada di bawah kolom *re-password* terdapat tombol *register* untuk menjalankan fungsi *register*. Lalu, untuk kembali ke halaman *login*, *user* dapat mengeser halaman *register* dari arah kiri ke kanan.

### D. Home Menu

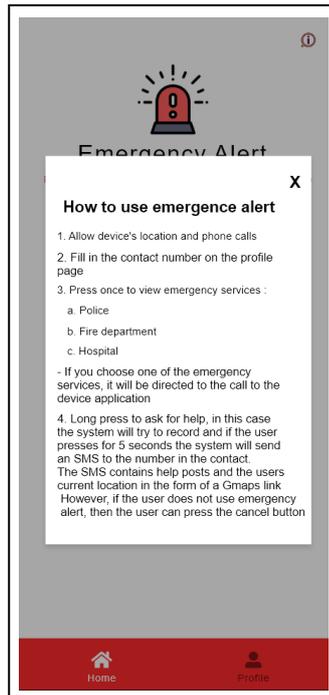
Mockup pada halaman *home* menu digunakan sebagai acuan awal dalam pembuatan menu utama pada aplikasi. Halaman *home* menu ini terdapat 2 bagian yaitu bagian *body* yang merupakan isi dari halaman dan bagian *bottom* yang

merupakan menu *navigasi*. Namun sebelum memasuki halaman *home* terdapat hal-hal yang harus dilakukan oleh user yaitu sebagai berikut.



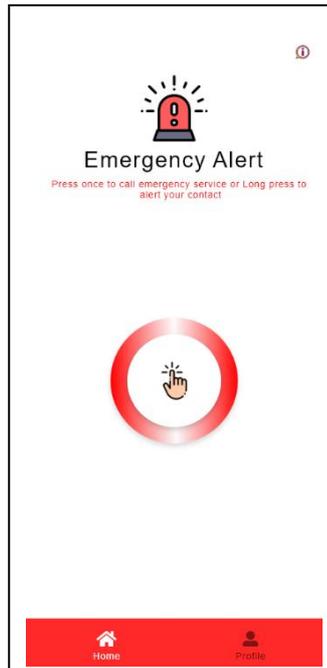
Gambar 3.15 Mockup Home Menu (Permission Location)

Gambar 3.15 merupakan mockup halaman *home* menu bagian *permission location*. Sebelum melakukan aktivitas di halaman *home*, *user* akan ditampilkan *popup permission location* yang isinya merupakan izin untuk mengakses lokasi user pada aplikasi ini. Di dalam *permission* ini terdapat tombol *deny* dan *allow*. Jika *user* tidak menyetujui *permission* tersebut maka *user* dapat menekan tombol *deny* namun jika *user* menyetujui *permission* tersebut maka *user* dapat menekan tombol *allow*. Tampilan *permission location* ini akan diminta ulang pada saat *user* melakukan *longpress* pada tombol *alert* jika *user* menekan tombol *deny*.



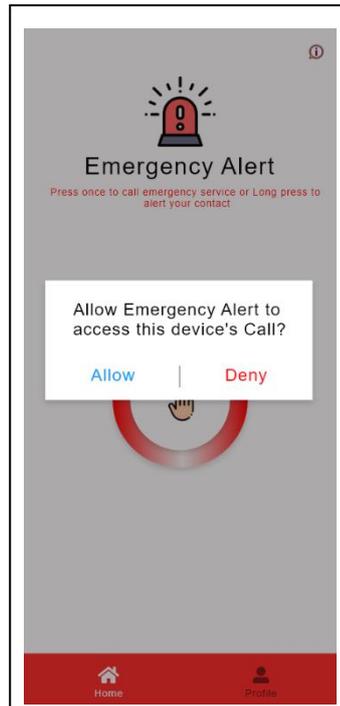
Gambar 3.16 Mockup Home Menu (Dialog How to Use)

Gambar 3.16 merupakan mockup halaman home menu bagian *dialog how to use*. Setelah melakukan pilihan pada *permission location*, *user* akan ditampilkan *popup dialog how to use* yang isinya merupakan informasi-informasi untuk menggunakan aplikasi. Selain itu di dalam dialog tersebut *user* dapat melakukan *scroll* jika informasinya terlalu panjang dan *user* dapat menekan icon (x) pada kanan atas untuk menutup dialog informasi. Dialog ini juga dapat diakses kembali ketika *user* menekan icon (i) pada kanan atas halaman home menu.



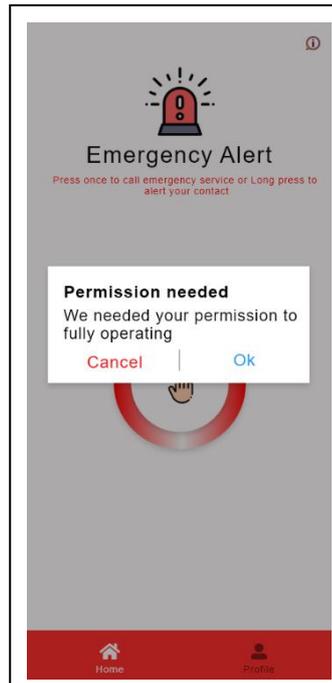
Gambar 3.17 Mockup Home Menu

Gambar 3.17 merupakan mockup halaman home menu. Pada bagian body ini akan ditampilkan icon (i), icon alert, tulisan “Emergency Alert”, *Alert text* dan tombol *alert*. Lalu pada bagian *bottom* akan ditampilkan 2 menu *navigation* yaitu *home* menu dan *profile* menu.



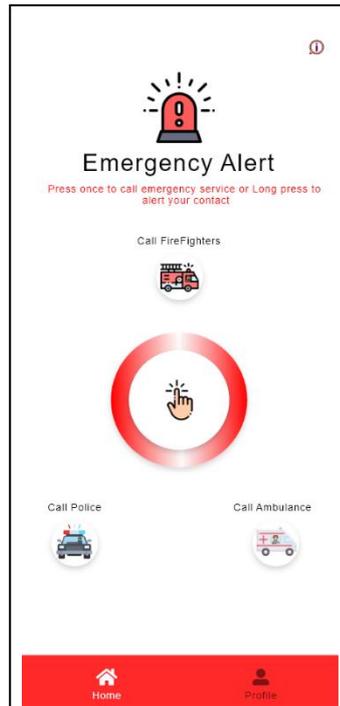
Gambar 3.18 Mockup Home Menu (Permission Call)

Gambar 3.18 merupakan *mockup* halaman *home* menu bagian *permission call*. Pada halaman *home*, ketika *user* menekan sekali pada *button alert* maka *user* akan ditampilkan *permission call*. *Permission* ini digunakan untuk mengakses *phone call* pada *device user*. Jika *user* tidak memberikan izin pada *permission call* ini maka *user* dapat menekan tombol *deny* namun jika *user* memberikan izin pada *permission call* ini maka *user* dapat menekan tombol *allow*. *User* yang tidak menyetujui *permission call* ini maka *user* akan ditampilkan *dialog alert* "permission needed" sampai *user* menyetujui *permission* ini, baru *user* akan ditampilkan *floating button emergency service*.



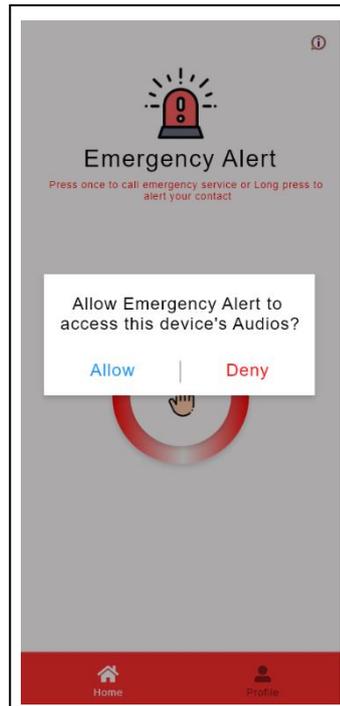
Gambar 3.19 Mockup Home Menu (Dialog Alert Permission Needed)

Gambar 3.19 merupakan mockup halaman *home* menu bagian *dialog alert permission needed*. Pada bagian ini, *user* yang telah melakukan penolakan pada *permission* diaplikasi ini maka akan ditampilkan *dialog alert* yang isinya “we need your permission to fully operating”. Jika *user* menekan tombol *cancel* maka *user* akan ditampilkan terus menerus dialog ini, namun jika *user* menekan tombol *ok* maka *user* akan diarahkan ke *permission* yang dituju untuk melakukan persetujuan pada *permission* tersebut.



Gambar 3.20 Mockup Home Menu (Press Once Button Alert)

Gambar 3.20 merupakan mockup halaman *home* menu bagian *press once button alert*. Pada bagian ini, jika *user* telah melakukan persetujuan pada *permission call* maka *user* dapat menekan sekali pada *button alert* ini untuk melihat *floating button* yang merupakan *service emergency* yaitu *firefighters*, *police* dan *hospital*. Jika *user* menekan salah satu dari ketiga *floating button* tersebut, maka *user* akan diarahkan kearah *phone call* dan sistem akan menampilkan nomor dari *service emergency* yang telah dipilih sebelumnya.



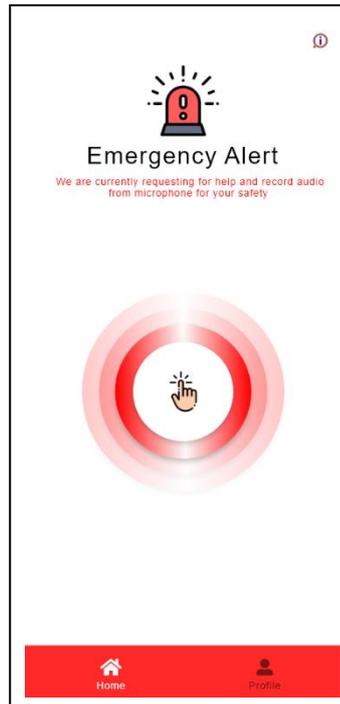
Gambar 3.21 Mockup Home Menu (Permission Audios)

Gambar 3.21 merupakan *mockup* halaman *home* menu bagian *permission audios*. Pada bagian ini, ketika *user* menekan lama pada *button alert* maka *user* akan ditampilkan *permission audios*. *Permission* ini bertujuan untuk mengakses *audio* yang berada dalam *device user* yang akan digunakan untuk merekam suara sekitar ketika *user* melakukan *longpress button alert*. Jika *user* tidak memberikan izin pada *permission audios* ini maka *user* dapat menekan tombol *deny* namun jika *user* memberikan izin pada *permission audios* ini maka *user* dapat menekan tombol *allow*. *User* yang tidak menyetujui *permission audios* ini maka *user* akan ditampilkan *dialog alert* “permission needed” sampai *user* menyetujui *permission* ini, baru *user* akan ditampilkan proses selanjutnya.



Gambar 3.22 Mockup Home Menu (Permission SMS)

Gambar 3.22 merupakan mockup halaman *home* menu bagian permission SMS. Pada bagian ini, ketika *user* menekan lama pada *button alert* maka *user* akan ditampilkan *permission SMS*. *Permission* ini bertujuan untuk mengakses SMS yang berada dalam device *user* yang akan digunakan untuk mengirimkan pesan darurat ketika *user* melakukan *longpress button alert*. Jika *user* tidak memberikan izin pada *permission SMS* ini maka *user* dapat menekan tombol *deny* namun jika *user* memberikan izin pada *permission call* ini maka *user* dapat menekan tombol *allow*. *User* yang tidak menyetujui *permission SMS* ini maka *user* akan ditampilkan *dialog alert* “permission needed” sampai *user* menyetujui *permission* ini, baru *user* akan ditampilkan proses selanjutnya.

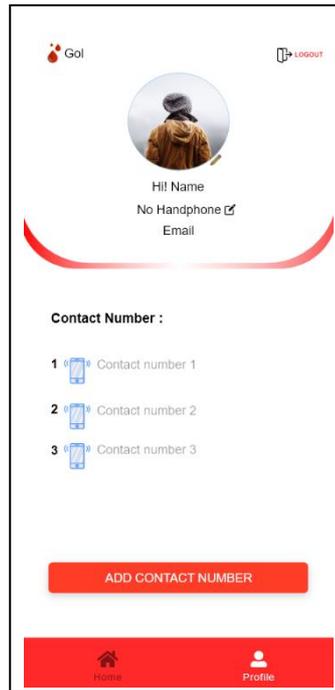


Gambar 3.23 Mockup Home Menu (Long Press Button Alert)

Gambar 3.23 merupakan mockup halaman *home* menu bagian *long press button alert*. Pada bagian ini, ketika semua *permission* sudah diterima oleh *user* maka *user* dapat menekan lama pada *button alert* untuk meng-*record* suara sekitar dan mengirimkan pesan darurat via *SMS Alert*. Pesan darurat ini akan dikirimkan ke orang-orang yang nomornya sudah disimpan pada kontak aplikasi di halaman *profile*. Isi pesan darurat tersebut berupa *text* pesan otomatis dan *link current location* yang mengarah ke aplikasi google maps jika link tersebut ditekan.

#### **E. Profile**

*Mockup* pada halaman *profile* menu digunakan sebagai informasi data diri. Halaman *profile* menu ini terdapat 2 bagian yaitu bagian *body* yang merupakan isi dari halaman dan bagian *bottom* yang merupakan menu *navigasi*.



Gambar 3.24 Mockup Profile Menu

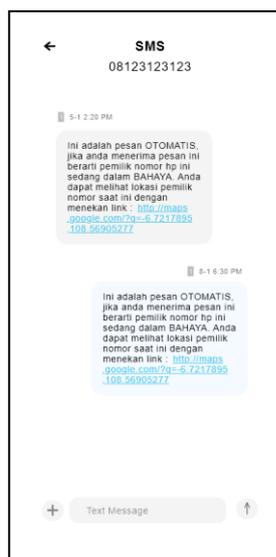
Gambar 3.24 merupakan *mockup* halaman *profile* menu pada bagian *body*. Pada halaman *profile* ini, *user* dapat melihat *informasi* data diri seperti golongan darah, *photo profile*, *username*, nomor *handphone*, *email*, dan *contact number*. Selain informasi data diri, *user* juga ditampilkan fitur *edit* untuk mengubah *photo profile*, *no handphone* dan *edit contact number*. Lalu untuk melakukan keluar dari aplikasi *user* dapat menekan tulisan *logout* dipojok kanan atas.



Gambar 3.25 Mockup Profile Menu (Add or Edit Contact Number)

Gambar 3.25 merupakan *mockup* halaman *profile* menu pada bagian *add or edit contact number*. Pada bagian ini, *user* dapat menambahkan atau meng-*edit contact number* dengan menekan tombol *add* atau *edit* lalu *user* dapat meng-*input* nomor telephone orang terdekat pada kolom *text*. Setelah *user* sudah selesai meng-*input phone number* dari orang terdekat maka *user* dapat menekan tombol *save*.

## F. Luaran Aplikasi (SMS Alert)



Gambar 3.26 Mockup Luaran Aplikasi (SMS Alert)

Gambar 3.26 merupakan *mockup* luaran aplikasi pada bagian SMS Alert. Pada bagian ini, setelah user sudah berhasil melakukan proses long press button alert maka sistem akan mengirimkan pesan darurat berupa SMS Alert kepada nomor yang berada didalam kontak aplikasi. SMS Alert yang sudah dikirimkan oleh sistem dapat dilihat oleh user pengirim atau user penerima dengan membuka app SMS. SMS Alert ini berisikan sebuah pesan darurat beserta link current location dari user pengirim yang sedang membutuhkan pertolongan.

### 3.2.4 Pemilihan Aset

Tabel 3.3 Daftar Aset (Gambar)

No	Gambar	Penjelasan	Sumber Gambar
1.		<i>Image</i> lonceng alert ini merupakan logo dari aplikasi 'Emergency Alert'	Image penggabungan dari 2 gambar yang diambil dari <a href="http://www.flaticon.com">www.flaticon.com</a> dengan menggunakan Adobe XD
2.		<i>Image</i> lampu merah ini dipakai sebagai hiasan didalam halaman login dan <i>home</i> menu.	<a href="http://www.flaticon.com">www.flaticon.com</a>
3.		<i>Image</i> (i) ini dipakai sebagai penanda untuk melihat informasi aplikasi di dalam halaman <i>home</i> menu.	<a href="http://www.flaticon.com">www.flaticon.com</a>
4.		<i>Image</i> tangan seperti menekan ini dipakai sebagai penanda untuk menekan pada tombol yang ada di dalam halaman <i>home</i> menu.	<a href="http://www.flaticon.com">www.flaticon.com</a>
5.		<i>Image</i> darah ini dipakai sebagai penanda untuk menginformasikan golongan darah user yang dipakai dalam halaman <i>profile</i> menu.	<a href="http://www.flaticon.com">www.flaticon.com</a>
6.		<i>Image</i> pintu dan panah keluar ini dipakai sebagai penanda untuk menginformasikan <i>logout</i> dari aplikasi yang dipakai dalam halaman <i>profile</i> menu.	<a href="http://www.flaticon.com">www.flaticon.com</a>
7.		<i>Image</i> pensil ini dipakai sebagai penanda untuk menginformasikan user dapat meng- <i>edit</i> photo <i>profile</i> yang dipakai dalam halaman <i>profile</i> menu.	<a href="http://www.flaticon.com">www.flaticon.com</a>

Tabel 3.3 Daftar Aset (Gambar)

No	Gambar	Penjelasan	Sumber Gambar
8.		<i>Image</i> pensil dengan box ini dipakai sebagai penanda untuk menginformasikan <i>user</i> dapat meng- <i>edit phone number</i> yang dipakai dalam halaman <i>profile</i> menu.	<a href="http://www.flaticon.com">www.flaticon.com</a>
9.		<i>Image</i> handphone beserta sinyal pemancar ini dipakai sebagai penanda untuk menginformasikan <i>contact number user</i> yang dipakai dalam halaman <i>profile</i> menu.	<a href="http://www.flaticon.com">www.flaticon.com</a>
10.		<i>Image</i> mobil <i>ambulance</i> beserta dokter yang berada di dalam mobil <i>ambulance</i> tersebut dipakai sebagai penanda untuk menginformasikan dapat mengecek atau menelepon ambulance dari <i>floating button</i> yang dipakai dalam halaman <i>home</i> menu.	<i>Image</i> penggabungan dari 2 gambar yang diambil dari <a href="http://www.flaticon.com">www.flaticon.com</a> dengan menggunakan Adobe XD
11.		<i>Image</i> mobil polisi beserta polisi yang berada didalam mobil polisi tersebut dipakai sebagai penanda untuk menginformasikan dapat mengecek atau menelepon polisi dari <i>floating button</i> yang dipakai dalam halaman <i>home</i> menu.	<i>Image</i> penggabungan dari 2 gambar yang diambil dari <a href="http://www.flaticon.com">www.flaticon.com</a> dengan menggunakan Adobe XD
12.		<i>Image</i> mobil pemadam kebakaran beserta petugas pemadam kebakaran yang berada didalam mobil pemadam kebakaran tersebut dipakai sebagai penanda untuk menginformasikan dapat mengecek atau menelepon pemadam kebakaran dari <i>floating button</i> yang dipakai dalam halaman <i>home</i> menu.	<i>Image</i> penggabungan dari 2 gambar yang diambil dari <a href="http://www.flaticon.com">www.flaticon.com</a> dengan menggunakan Adobe XD

Tabel 3.3 merupakan daftar gambar yang digunakan pada aplikasi mobile yaitu dalam penggunaan di-*layout-layout* aplikasi. Daftar gambar ini, diambil dari flaticon sebagai sumber yang berada diinternet serta memiliki izin dalam memakai gambar tersebut.

Tabel 3.4 Daftar Aset (Icon)

No	Icon	Penjelasan	Sumber Icon
1.		Icon <i>home</i> digunakan dalam pemakaian menu navigasi ke halaman <i>home</i> menu	Vector Asset (xml) Android Studio
2.		Icon <i>people</i> digunakan dalam pemakaian menu navigasi ke halaman <i>profile</i> menu	Vector Asset (xml) Android Studio
3.		Icon x digunakan dalam pemakaian <i>close dialog how to use</i>	Vector Asset (xml) Android Studio
4.		Icon panah ke kanan ini dipakai sebagai penanda untuk menggeser halaman <i>login</i> dari arah kanan ke kiri.	Vector Asset (xml) Android Studio
5.		Icon panah kekiri ini dipakai sebagai penanda untuk menggeser halaman <i>register</i> dari arah kiri ke kanan.	Vector Asset (xml) Android Studio
6.		Icon centang ini dipakai sebagai penanda <i>save</i> untuk menyimpan nomor telephone yang sehabis di- <i>edit</i> pada halaman <i>profile</i> menu.	Vector Asset (xml) Android Studio
7.		Icon laki-laki ini dipakai sebagai penanda gender laki-laki di <i>photo profile</i> pada halaman <i>profile</i> menu.	<a href="http://www.flaticon.com">www.flaticon.com</a>
8.		Icon perempuan ini dipakai sebagai penanda <i>gender</i> perempuan di <i>photo profile</i> pada halaman <i>profile</i> menu.	<a href="http://www.flaticon.com">www.flaticon.com</a>

Tabel 3.4 merupakan daftar *icon* yang digunakan pada aplikasi *mobile* yaitu dalam penggunaan di *layout-layout* aplikasi. Daftar *icon* ini, diambil dari android studio dan flaticon sebagai sumber yang berada di internet serta memiliki izin dalam memakai icon tersebut.