

## **BAB IV. ANALISIS DAN PERANCANGAN SISTEM**

### **4.1 Analisis**

Pada tahap analisis berisikan uraian pokok pemikiran untuk menelaah suatu persoalan dengan tujuan sebagai pemahaman yang tepat terhadap suatu objek kajian, analisis pada tahap ini berisikan pengumpulan data, analisa pengguna, analisis kebutuhan sistem serta desain arsitektur sistem.

#### **4.1.1 Pengumpulan Data**

Metode pengumpulan data yang penulis gunakan yaitu dengan mencari data-data, serta informasi apa saja yang nantinya dibutuhkan dalam perancangan dan pengembangan Aplikasi Pemesanan Ambulans. Pengumpulan data pada penelitian ini melalui 2 tahapan, yakni studi literatur dan observasi. Studi literature dilakukan dengan mencari informasi terkait dengan pokok pembahasan melalui beberapa referensi seperti buku, jurnal ilmiah, artikel laporan penelitian, laporan akhir tahun-tahun sebelumnya, serta bacaan yang dapat mendukung dalam tinjauan pustaka. Selanjutnya, tahapan observasi dilakukan dengan menjalankan studi terhadap data ambulans yang diperlukan pada aplikasi, termasuk data lokasi ambulans. Informasi diperoleh melalui proses wawancara terhadap petugas ambulans dalam hal ini adalah Wiu-Wiu Ambulans Malang, Ambulans Malang Strudel, dan Ambulans Yuekin 66.

#### **4.1.2 Analisa Pengguna**

Analisisa Pengguna merupakan analisis terhadap siapa saja aktor yang terlibat dalam Pengembangan Aplikasi Pemesanan Ambulans, sebagai berikut:

Tabel 4.1 : Analisis Pengguna

No.	Pengguna	Keterangan
1.	Admin	Melakukan <i>create, read, update, delete</i> terhadap data ambulans, data petugas, data <i>Customer</i> , data pesanan, dan memvalidasi data pengguna.
2.	Petugas	Melakukan pengelolaan data ambulans, dan data petugas, dan memverifikasi pesanan.
3.	<i>Customer</i>	Melakukan pemesanan

#### 4.1.3 Analisis Kebutuhan Fungsional

Semua aktor seperti masyarakat, pengelola ambulans, dan administrator adalah pengguna yang dapat menjalankan aplikasi. Aktor masyarakat berfungsi sebagai target utama pengguna aplikasi yang memiliki dapat menjalankan fitur mulai dari pendaftaran aplikasi dengan mengisi data berupa NIK, nama lengkap, alamat, *Username, password*, dll. Selanjutnya adalah *Login* aplikasi dengan memasukkan *Username* dan *password* yang telah didaftarkan sebelumnya. Pemesanan ambulans, pembatalan ambulans fitur ini tidak begitu saja dapat dilakukan, terdapat beberapa persyaratan untuk melakukan fitur ini, di antaranya mengisi form pembatalan dan memberikan alasan mengapa dilakukan pembatalan serta melalui validasi dari administrator agar masyarakat tidak semena-mena dalam melakukan pemesanan ambulans, melihat informasi ambulans, melihat lokasi ambulans, mencari ambulans terdekat, melihat estimasi waktu, melihat estimasi biaya, melihat rekomendasi ambulans dimana masyarakat diberikan kemudahan dalam pemesanan ambulans tanpa memilih ambulans terlebih dahulu, masyarakat hanya perlu memasukkan kondisi korban lalu aplikasi akan memberikan rekomendasi ambulans yang harus dipesan, dan fitur melihat kontak ambulans tanpa melakukan registrasi atau *login*.

Selanjutnya adalah pengelola ambulans yang dapat melakukan fitur pendaftaran aplikasi dengan mengisi beberapa data, *login* aplikasi dengan memasukkan *username* dan *password* yang telah didaftarkan sebelumnya, melihat lokasi ambulans, serta menerima pesanan. Aktor selanjutnya adalah administrator

sebagai pengelola data pengguna dan melakukan verifikasi data pengguna yang mendaftar untuk menghindari pengguna yang tidak bertanggung jawab atas aplikasi ini.



Gambar 4.1 Analisis Kebutuhan Fungsional

#### 4.1.4 Analisis Kebutuhan Sistem

Analisis kebutuhan sistem digunakan untuk menerapkan pada sistem yang akan dibuat agar dapat digunakan apabila perlu untuk dilakukan penambahan dalam proses analisa tersebut. Analisis kebutuhan sistem terdiri dari spesifikasi kebutuhan perangkat keras (*hardware*) dan spesifikasi kebutuhan perangkat lunak (*software*). Berikut penjelasannya:

1. Kebutuhan Perangkat Keras (*Hardware*)

Untuk menjalankan Aplikasi Pemesanan Ambulans ini dibutuhkan sistem perangkat keras yang mampu mendukung dalam pengembangan dan pengoperasian aplikasi. Sistem perangkat keras tersebut harus memenuhi spesifikasi minimal dari

kebutuhan *hardware* dari sistem yang akan diterapkan. Adapun pada tabel dibawah ini menunjukkan spesifikasi minimal *hardware* yang dibutuhkan:

Tabel 4.2 : Analisis Kebutuhan Perangkat Keras (*Hardware*)

Perangkat Keras	Keterangan
CPU	Intel® Core™ i5-7200U 2.50GHz
<i>Memory</i>	RAM 8192MB
<i>Harddisk</i>	1 TB HDD 240GB SSD
VGA	NVIDIA GEFORCE 950MX

2. Kebutuhan Perangkat Lunak (*Software*)

Untuk membangun sistem pemesanan ambulans ini dibutuhkan perangkat lunak (*Software*) seperti pada tabel dibawah ini:

Tabel 4.3 : Analisis Kebutuhan Perangkat Lunak (*Software*)

Perangkat Lunak	Keterangan
Windows 10	Sistem operasi yang digunakan untuk membuat aplikasi dan mengolah data.
Android Studio	Digunakan untuk mengembangkan aplikasi <i>mobile</i> .
Visual Studio Code	Sebagai tools untuk menulis kode program.
XAMPP	Sebagai <i>Software</i> yang membundling fasilitas Apache dan MySQL didalamnya.
Apache	Sebagai <i>web server</i> .
MySQL	Sebagai aplikasi <i>Database server</i> untuk menyimpan data.
<i>Google Maps API</i>	Sebagai peta untuk memetakan lokasi.
Mozilla Firefox / Google Chrome	<i>Web browser</i> yang digunakan untuk sisi admin Aplikasi Pemesanan Ambulans.

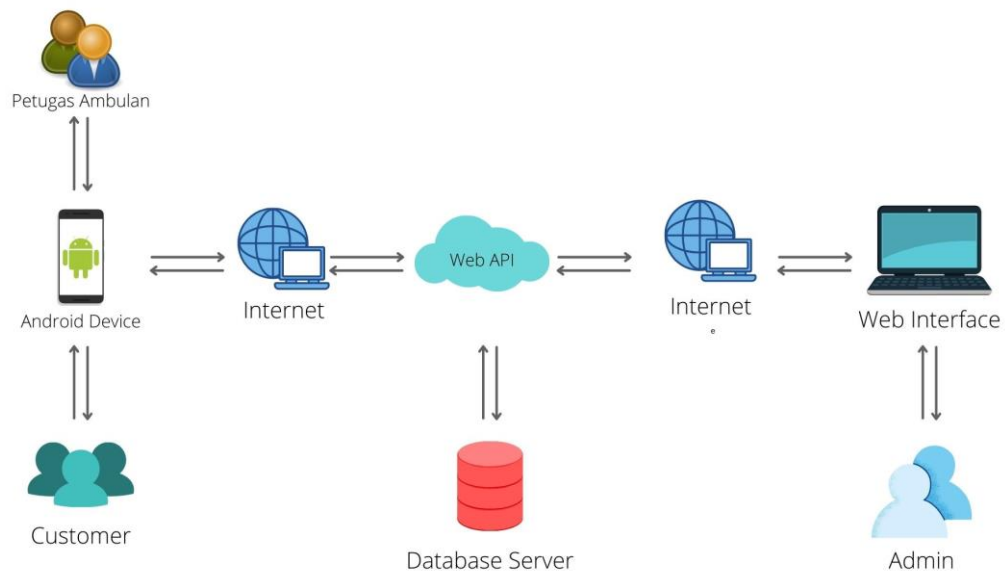
Sedangkan spesifikasi minimal yang diperlukan oleh sisi *user* adalah sebagai berikut ini:

- a. Perangkat *mobile / smartphone* berbasis Android.
- b. Spesifikasi Android minimum versi 4.4 (*Kitkat*).

- c. Memiliki fitur *GPS (Global Positioning System)* dan koneksi internet.

#### 4.1.5 Desain Arsitektur Sistem

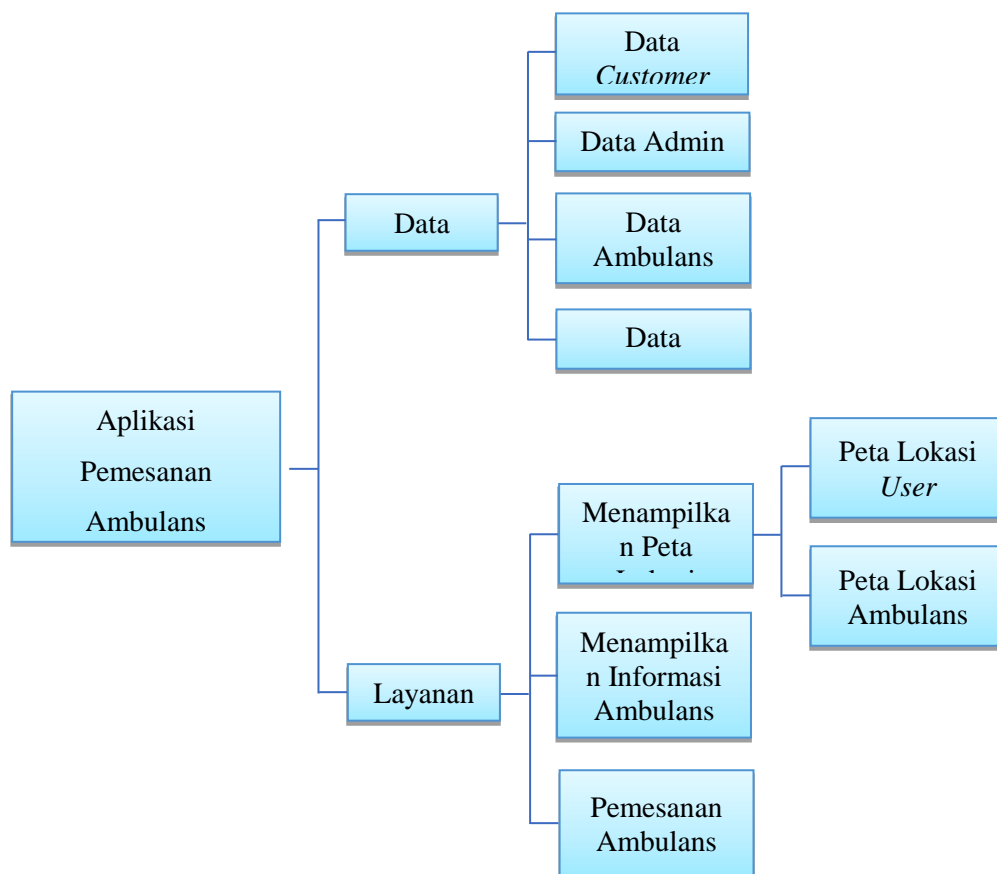
Pengembangan aplikasi pemesanan ambulans berbasis android ini merupakan sebuah aplikasi yang dibuat dengan tujuan untuk mempermudah masyarakat untuk melakukan pemesanan ambulans dan sebagai sarana edukasi kepada masyarakat mengenai jenis-jenis ambulans dan lainnya. Dalam aplikasi ini terdapat 3 hak akses yaitu admin, *user*, dan pengelola ambulans. Aplikasi akan melibatkan *user* dan pengelola ambulans yang memanfaatkan perangkat *smartphone* berbasis Android. Untuk menggunakan aplikasi, *user* dan pengelola ambulans harus terlebih dahulu tergabung dengan *GPS (Global Positioning System)* untuk mendeteksi koordinat posisi terkini. Kemudian *user* dan pengelola ambulans dapat mengakses aplikasi melalui jaringan internet. Aplikasi akan membaca *database* dan membaca peta dari *Google Maps* dengan memanfaatkan *API (Application Programming Interface)*. Admin melakukan pengelolaan terhadap data yang ada melalui website dengan perangkat komputer dengan mengakses *server* aplikasi. Perancangan sistem tersebut telah dibuat dengan desain arsitektur sistem yang dapat dilihat pada Gambar 4.2 dibawah ini:



Gambar 4.2 Desain Arsitektur Sistem

## 4.2 Perancangan

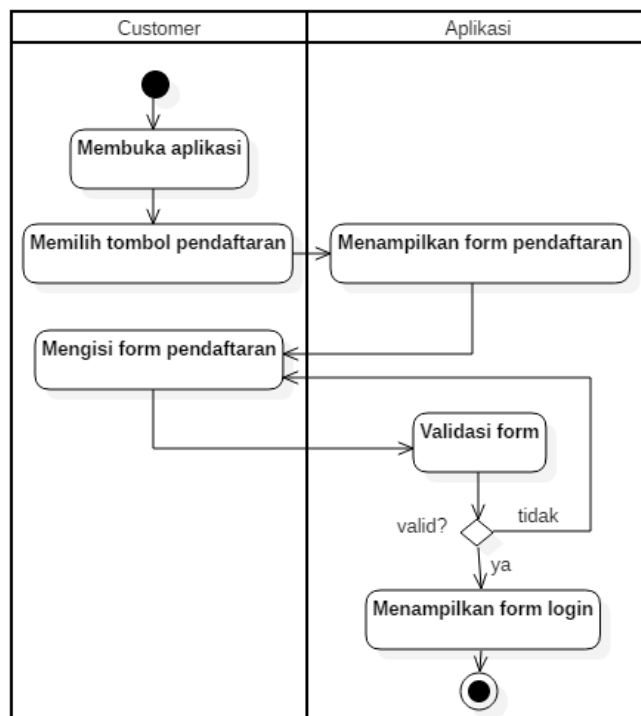
Tahap perancangan adalah tahap penggambaran, perencanaan, dan pembuatan sketsa atau pengaturan dari beberapa elemen yang terpisah ke dalam satu kesatuan yang utuh. Perancangan pada tahap ini berisikan perancangan arsitektural dan perancangan prosedural. Perancangan Arsitektural pada Aplikasi Pemesanan Ambulans Berbasis Android ini memiliki layanan WBS (*Work Breakdown Structure*), yaitu metode pengorganisasian proyek menjadi suatu pelaporan yang hirarkis. WBS digunakan untuk melakukan *breakdown* atau pemecahan tiap proses pekerjaan menjadi lebih detail. Hal ini dimaksudkan agar proses perencanaan proyek memiliki tingkat yang lebih baik. Pada Aplikasi Pemesanan Ambulans ini data yang digunakan antara lain adalah data *user*, data ambulans, data pengelola ambulans, dan data admin. Sedangkan layanan yang terdapat pada Aplikasi Pemesanan Ambulans ini antara lain adalah, menampilkan peta lokasi, menampilkan informasi ambulans, pemesanan ambulans, menampilkan informasi edukasi. WBS (*Work Breakdown Structure*) Aplikasi Pemesanan Ambulans dapat dilihat pada gambar 4.3 dibawah ini:



Gambar 4.3 *Work Breakdown Structure*

Adapun perancangan prosedural untuk pembuatan aplikasi pemesanan ambulans ini dapat dilihat pada *Activity Diagram* dan *Flowchart* dibawah ini. *Activity Diagram* dan *Flowchart Diagram* adalah sebuah gambaran bagan yang memperlihatkan urutan dan hubungan dari sebuah proses beserta instruksinya. Gambaran ini dinyatakan dengan simbol, dengan demikian setiap simbol berisi sebuah proses dan hubungan antar proses tersebut dihubungkan dengan garis penghubung. Berikut *Activity Diagram* dan *Flowchart Diagram* pada Pengembangan Aplikasi Pemesanan Ambulans Berbasis Android:

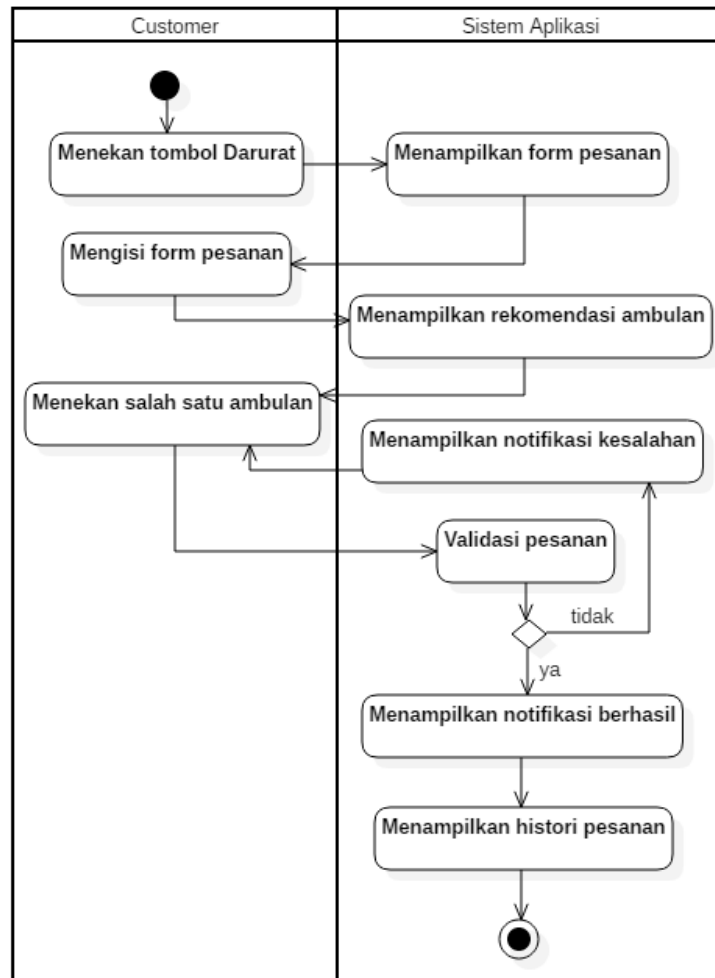
1. *Activity Diagram* Fitur Registrasi *Customer*



Gambar 4.5 *Activity Diagram* Fitur *Login* Pengguna

Jika customer belum memiliki akun, bisa melakukan pendaftaran dengan menuju ke menu registrasi dan mengisi form registrasi dengan memasukkan *username* dan *password*, setelah itu sistem akan memvalidasi, jika berhasil maka akan menuju halaman *login*. *Activity diagram* dijelaskan pada gambar 4.5

2. *Activity Diagram* Fitur Pemesanan Ambulan

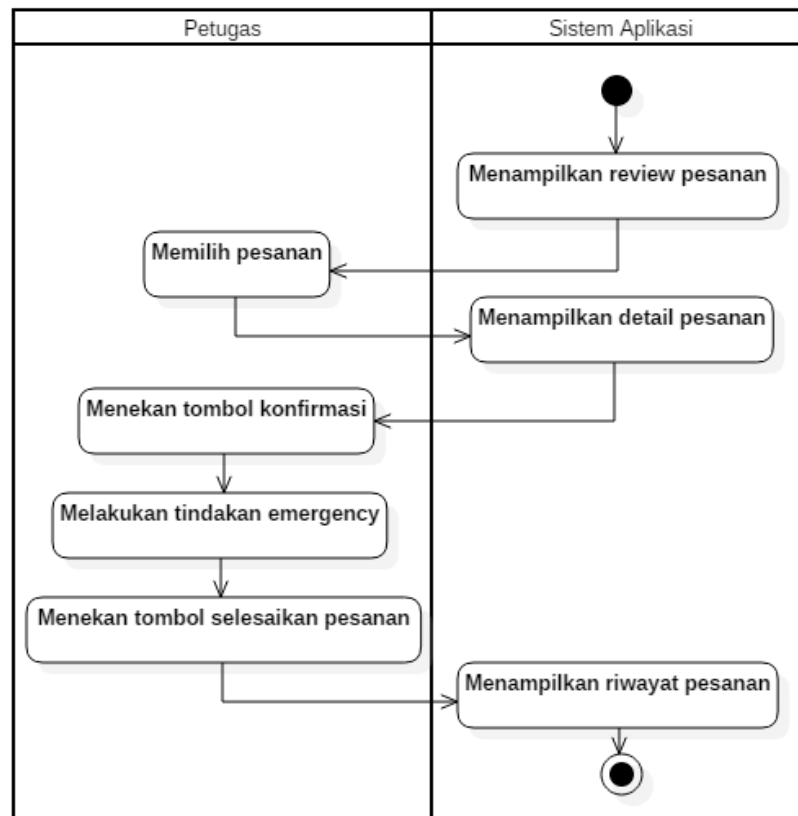


Gambar 4.6 *Activity Diagram* Fitur Pemesanan Ambulan

Pada gambar 4.6 digambarkan customer melakukan pemesanan dengan menekan tombol darurat, setelah itu akan ditampilkan form pesanan, lalu customer melakukan pengisian form pesanan dan menekan tombol pesan, setelah itu aplikasi akan menampilkan rekomendasi ambulan sesuai dengan jenis kejadian dan jarak terdekat dari lokasi customer, setelah itu memilih ambulan yang diinginkan dengan menekan salah satu ambulan, sistem akan memvalidasi diantaranya, apakah customer telah melakukan pesanan sebelumnya dengan status pesanan belum selesai, jika validasi gagal maka akan muncul notifikasi kesalahan dan mengulahi pemilihan ambulan, jika verifikasi berhasil maka akan muncul notifikasi berhasil dan ambulan dapat terpesan.



### 3. Activity Diagram Fitur Penerimaan Pesanan Petugas



Gambar 4.7 Activity Diagram Fitur Penerimaan Pesanan Petugas

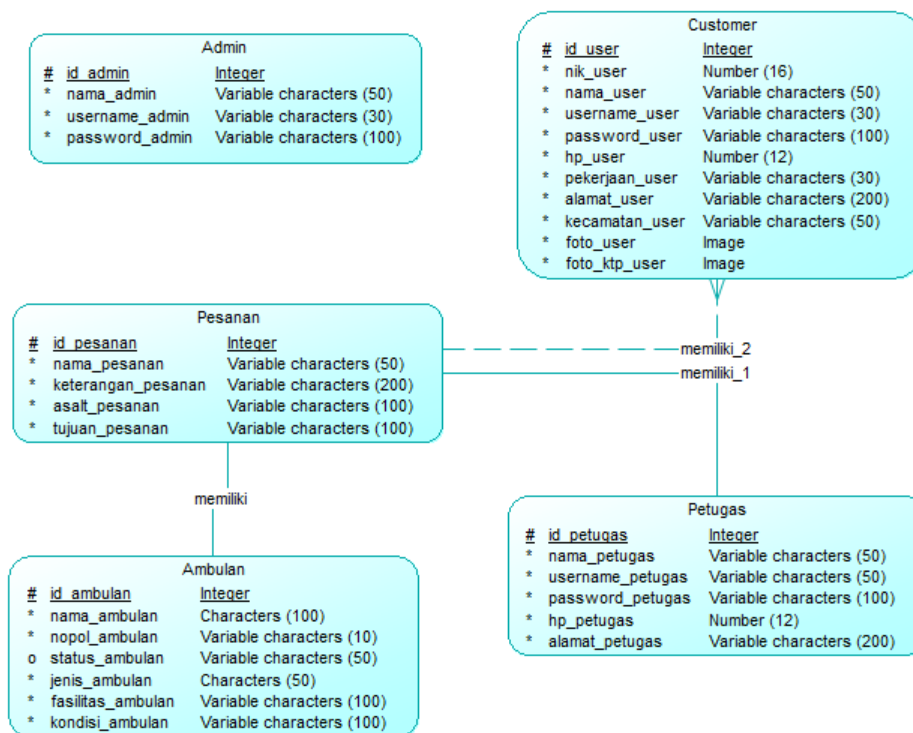
Pada gambar 4.7 digambarkan ketika petugas hendak melakukan penerimaan pesanan dapat dilakukan dengan menekan pesanan yang ada di halaman review pesanan, lalu aplikasi akan menampilkan halaman detail pesanan dan petugas menekan tombol konfirmasi pesanan, setelah itu petugas dapat melakukan tindakan *emergency* terhadap pasien, setelah selesai petugas harus menekan tombol selesaikan pesanan, dan pesanan tersebut akan berpindah ke halaman riwayat pesanan.

### 4.3 Perancangan Database

Perancangan *database* aplikasi pemesanan ambulans berbasis android terdiri dari *Conceptual Data Model* (CDM) dan *Physical Data Model* (PDM) yang dapat dilihat dibawah ini:

### 4.3.1 Conceptual Data Model (CDM)

*Conceptual Data Model* (CDM) merupakan sebuah struktur model yang berfungsi sebagai permodelan logis dari keseluruhan *database* aplikasi. CDM memberikan representasi formal dari data yang diperlukan untuk kebutuhan aplikasi meliputi objek data atau entitas serta relasi dalam *database login* atau konseptual. Dalam perancangan *database* proses desain dimulai pada tingkat konseptual, sehingga pengguna tidak perlu untuk mempertimbangkan rincian implementasi fisik yang sebenarnya. CDM Pengembangan Aplikasi Pemesanan Ambulans berbasis Android dapat dilihat pada gambar 4.8 dibawah ini .

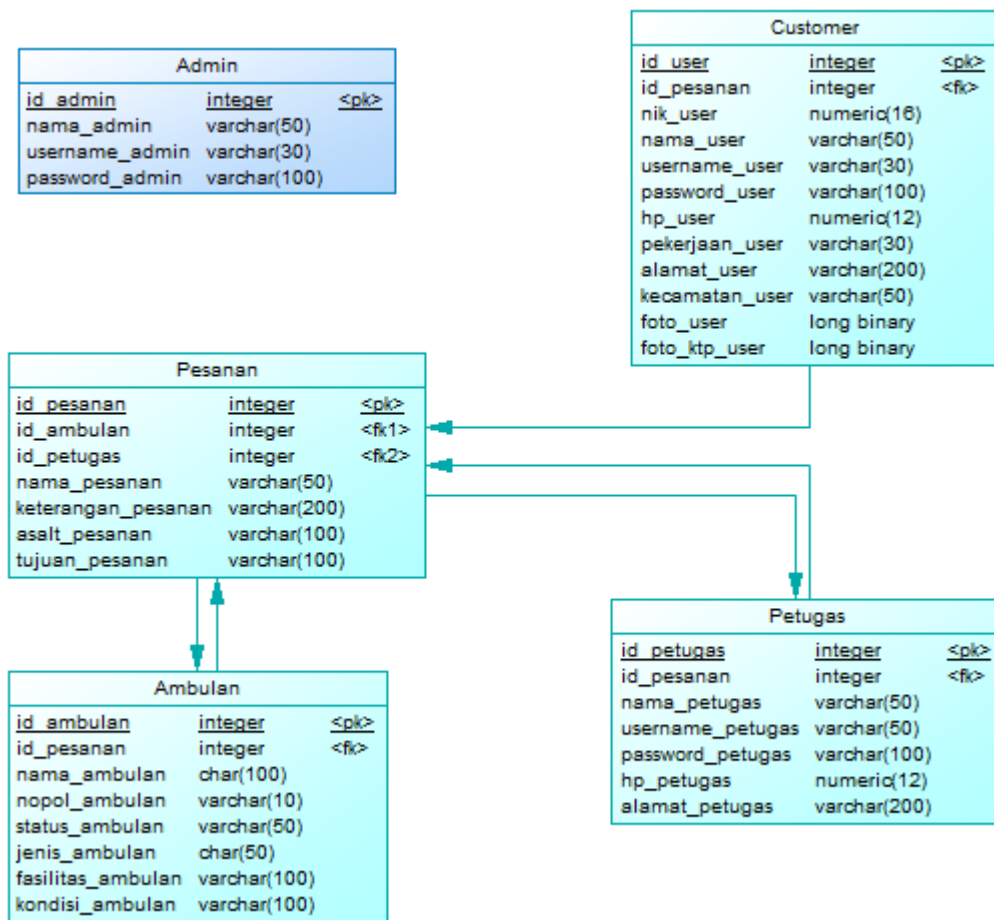


Gambar 4.9 *Conceptual Data Model* (CDM)

### 4.3.2 Physical Data Model (PDM)

*Physical Data Model* (PDM) merupakan gambaran fisik dari *database* yang akan dibuat dengan mempertimbangkan DBMS (*Database Management System*) yang akan digunakan. PDM merupakan hasil dari permodelan logis atau konseptual yaitu CDM, didalam PDM hubungan antar entitas yang digunakan adalah hubungan secara kardinalitas, dimana hubungan *one to one* akan ditandai dengan sebuah garis penghubung dan panah dua arah yang sejajar, hubungan *one to many* juga ditandai

dengan garis hubung dan tanpa panah yang mengarah pada tabel *many* dan PK (*Primary Key*) pada tabel *one* akan menjadi FK (*Foreign Key*) pada tabel *many*, dan hubungan *many to many* akan membuat sebuah tabel baru yang berisikan PK dari masing-masing tabel pembentuk hubungan *many to many*. PDM pengembangan aplikasi pemesanan ambulans berbasis android dapat dilihat pada gambar dibawah ini:

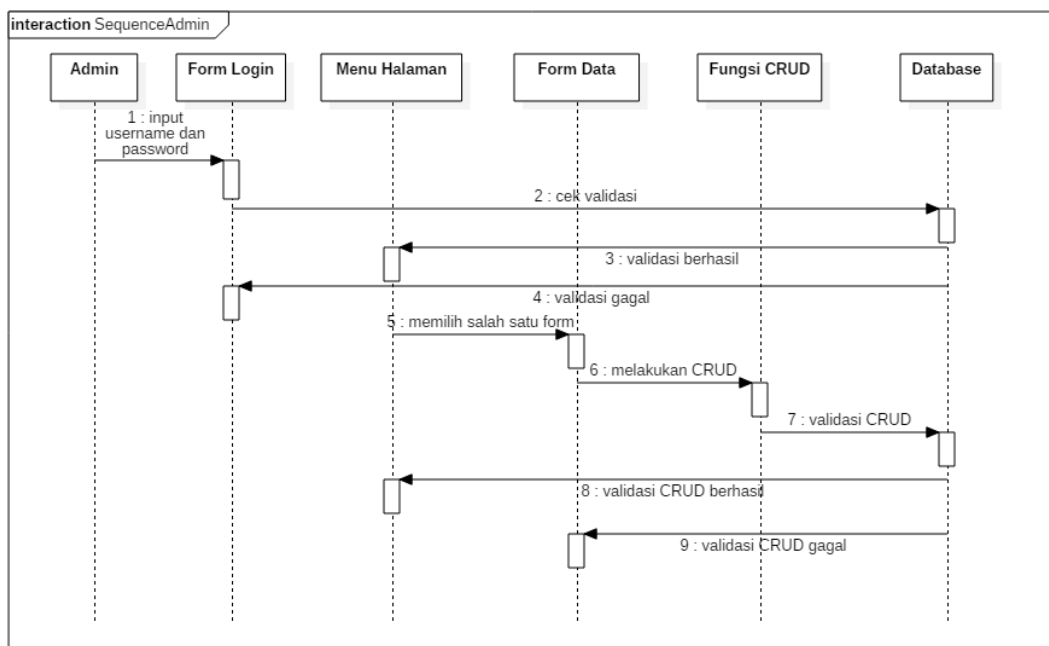


Gambar 4.10 Physical Data Model (PDM)

#### 4.4 Perancangan Sistem Menggunakan Sequence Diagram

*Sequence* diagram merupakan suatu diagram yang menjelaskan interaksi objek dan menunjukkan komunikasi diantara objek-objek tersebut dengan notasi petunjuk atau tanda panah. *Sequence* diagram digunakan untuk menjelaskan perilaku pada sebuah skenario dan menggambarkan bagaimana entitas dan sistem berinteraksi dalam sebuah urutan atau rangkaian waktu. *Sequence* diagram admin

pada gambar 4.10 menjelaskan alur kerja admin terhadap website admin yang berfungsi sebagai pengolahan data. Pertama admin harus melakukan *login* terlebih dahulu dengan memasukkan *username* dan *password*, kemudian sistem akan melakukan validasi terhadap data yang dimasukkan, jika berhasil maka admin akan masuk kedalam sistem dan menampilkan halaman utama. Kemudian admin dapat memilih 4 menu, antara lain menu admin, menu *customer*, menu petugas, dan menu ambulan. Didalam sistem ini admin dapat melakukan CRUD (*Create Read Update Delete*) terhadap data yang tersedia. *Sequence* diagram admin dapat dilihat pada Gambar 4.11 berikut ini:

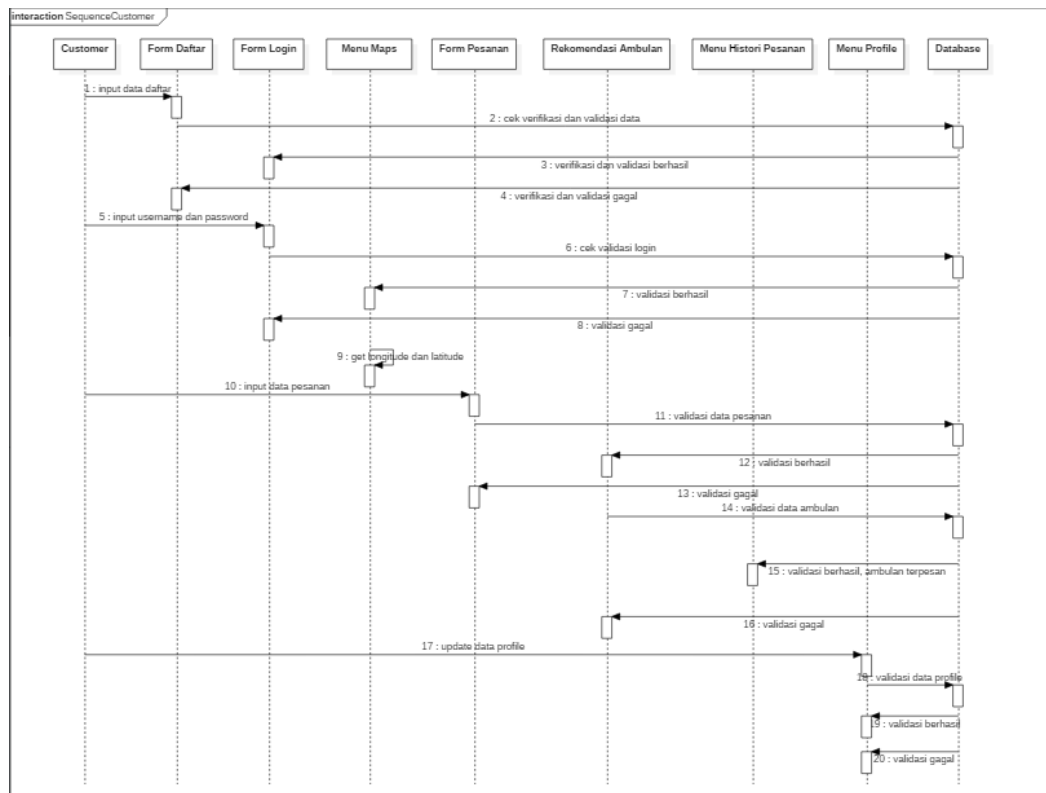


Gambar 4.11 *Sequence* Diagram Admin

Pada gambar 4.12 menjelaskan alur kerja *customer* terhadap aplikasi android yang berfungsi sebagai fitur utama didalam aplikasi ini adalah melakukan pemesanan terhadap ambulans. Pertama jika *customer* belum memiliki akun, harus melakukan registrasi terlebih dahulu dengan memasukkan *username*, *password*, dan data diri sesuai dengan ktp, seperti NIK, nama lengkap, no hp, pekerjaan, alamat lengkap, kecamatan, dan foto ktp. Setelah itu *customer* harus mendapatkan persetujuan terlebih dahulu oleh admin untuk dilakukan verifikasi terhadap akun yang didaftarkan, setelah akun telah diverifikasi oleh admin, *customer* dapat melakukan login dengan memasukkan *username* dan *password* yang telah

didaftarkan sebelumnya, lalu aplikasi akan melakukan validasi terhadap akun tersebut.

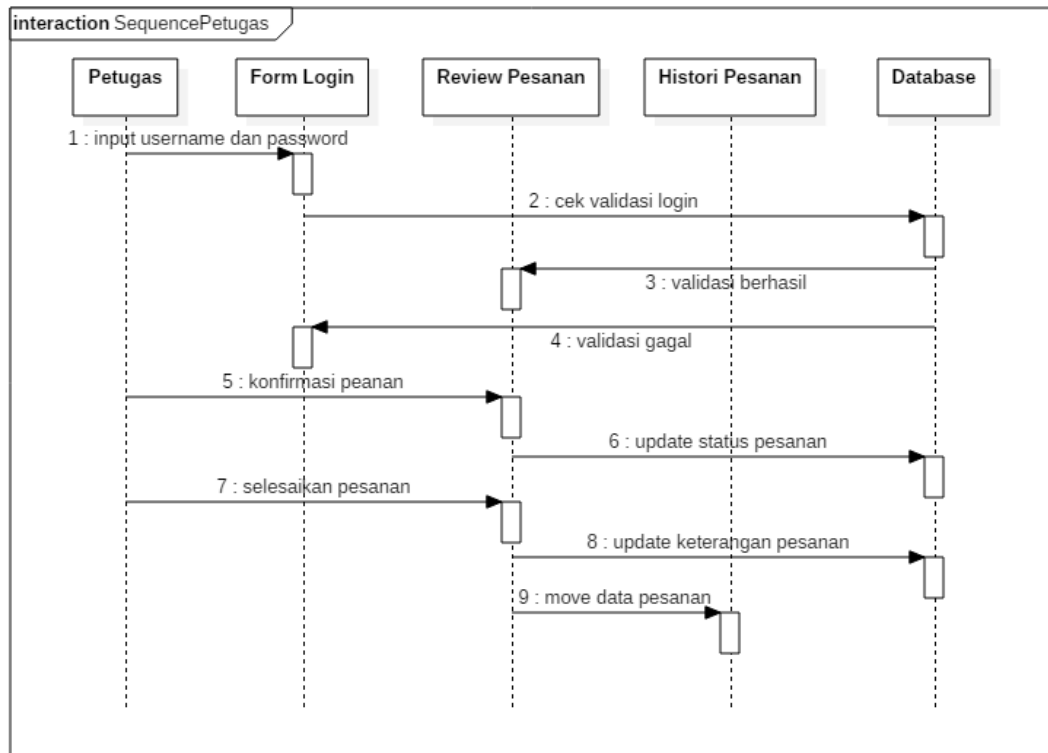
*Customer* dapat melakukan update data pada menu profile, selanjutnya *customer* dapat melakukan pemesanan terhadap ambulan dengan menekan tombol darurat pada menu *maps* dan mengisikan form pemesanan yang tersedia, lalu sistem akan melakukan validasi terhadap form pesanan, jika validasi berhasil maka akan menuju halaman rekomendasi ambulan, pada halaman ini *customer* diberikan informasi mengenai ambulan yang direkomendasikan berdasarkan jarak terdekat dan sesuai dengan jenis kejadian yang telah dimasukkan pada form pemesanan, selanjutnya *customer* memilih salah satu dari ambulan yang direkomendasikan, dan ambulan dapat terpesan, *customer* dapat melihat status pemesanan pada halaman riwayat pemesanan. *Sequence* diagram *customer* dapat dilihat pada gambar 4.12 berikut ini:



Gambar 4.12 *Sequence* Diagram *Customer*

Pada gambar 4.13 menjelaskan alur kerja petugas terhadap aplikasi android yang berfungsi untuk menerima pesanan dari *customer*. Pertama petugas melakukan *login* didalam aplikasi android dengan memasukkan *username* dan

*password*, jika proses validasi gagal maka akan menampilkan pesan *error*, namun jika berhasil maka akan masuk ke dalam halaman utama aplikasi, jika petugas mendapatkan pesanan akan ditampilkan pada halaman pesanan, dan petugas melakukan penerimaan pesanan dengan memberikan konfirmasi, lalu petugas melakukan tindakan *emergency* terhadap pesanan tersebut. *Sequence* diagram petugas dapat dilihat pada gambar 4.13 berikut ini:



Gambar 4.13 *Sequence* Diagram Petugas

#### 4.5 Perancangan *Interface*

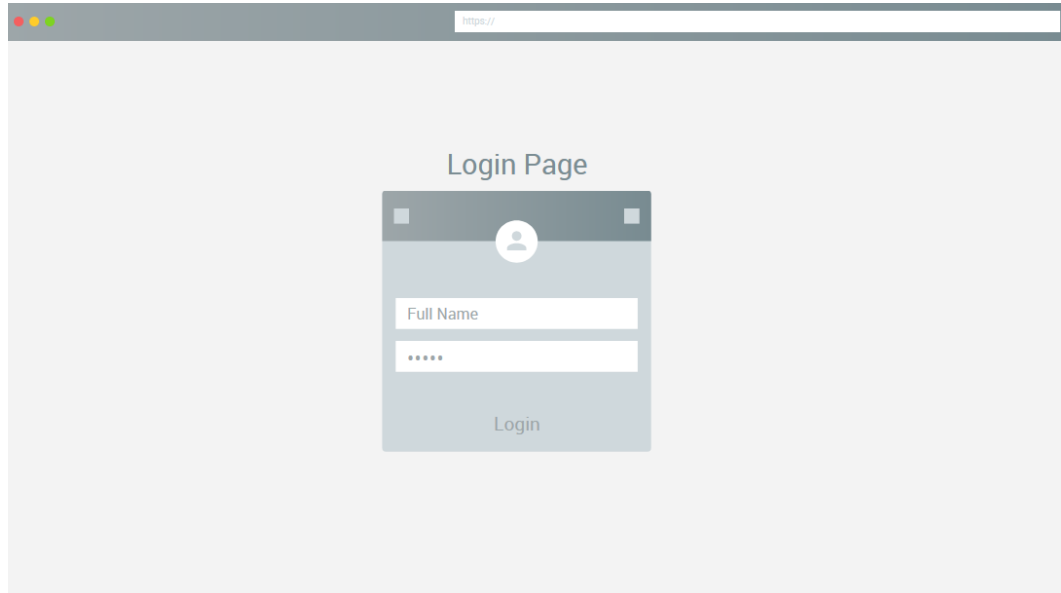
Desain *interface* aplikasi pemesanan ambulans berbasis android ini dibagi menjadi 2 macam, yaitu desain *interface website* admin dan desain *interface* aplikasi Android *customer*. Berikut penjelasannya:

##### 4.5.1 Desain *Interface Website Admin*

Desain *interface website* admin dibagi menjadi desain halaman *login*, desain halaman utama, dan desain halaman masing-masing menu CRUD.

1. Desain Halaman *Login*

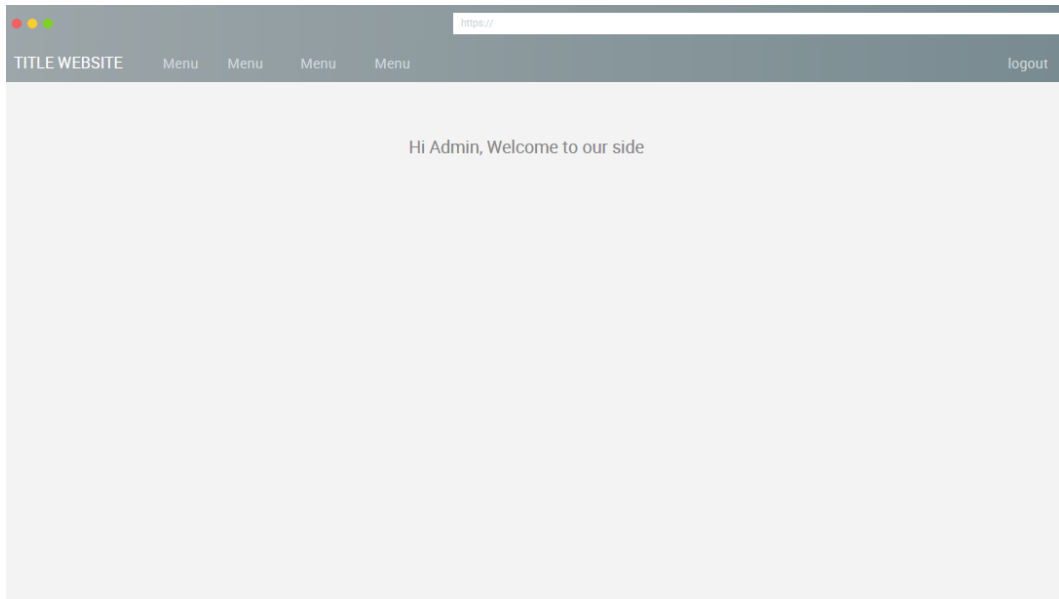
Halaman *login* merupakan halaman yang digunakan oleh admin untuk masuk ke dalam sistem. Pada halaman tersebut terdapat *form login* yang terdiri atas *username* dan *password*. Adapun desain halaman ini dapat dilihat pada Gambar 4.14 berikut :



Gambar 4.14 Desain Halaman *Login*

## 2. Desain Halaman Utama

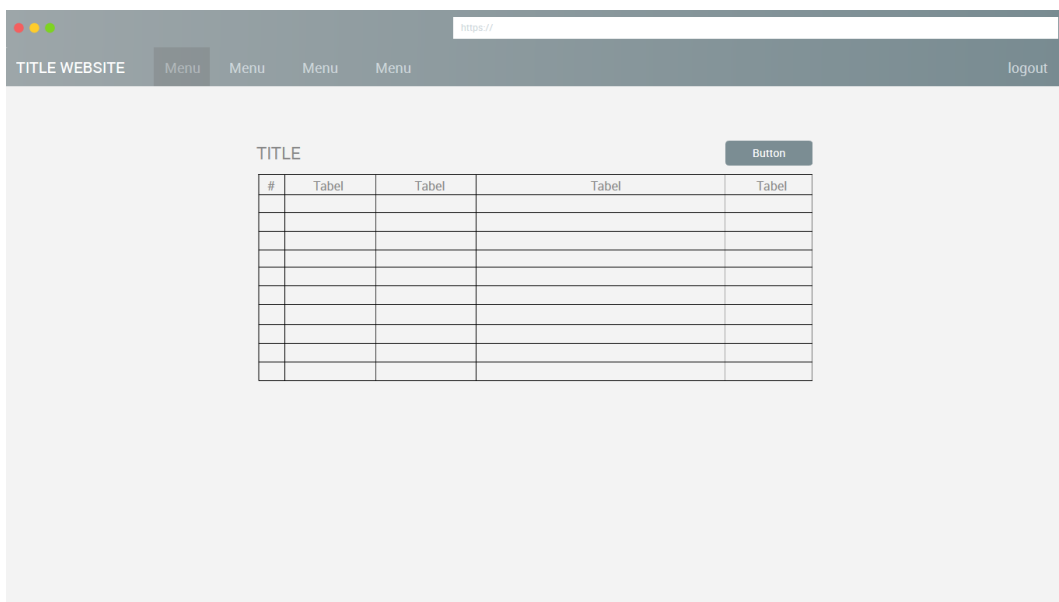
Halaman utama merupakan halaman yang berisi tampilan informasi mengenai data admin, data *customer*, data petugas, dan data ambulans. Desain halaman ini terdiri dari nama *website*, menu-menu, *header*, dan tombol *logout*. Admin dapat menuju ke menu yang diinginkan untuk berpindah ke salah satu halaman untuk melakukan proses CRUD (*Create Read Update Delete*).



Gambar 4.15 Desain Halaman Utama

### 3. Desain Halaman Masing-Masing Menu CRUD

Halaman CRUD (*Create Read Update Delete*) merupakan halaman yang berisi menu-menu untuk admin dalam mengolah data. Menu tersebut antara lain menu admin, menu *customer*, menu petugas dan menu ambulans. Di dalam halaman ini terdapat proses data yang dilakukan oleh admin yaitu berupa tambah data, lihat data, perbarui data dan hapus data. Desain dari halaman ini terdiri dari judul, *header*, *content* yang berisi tabel, tombol, dan *form* data. Adapun desain halaman ini dapat dilihat pada Gambar 4.16.



Gambar 4.16 Desain Halaman Masing-Masing Menu CRUD

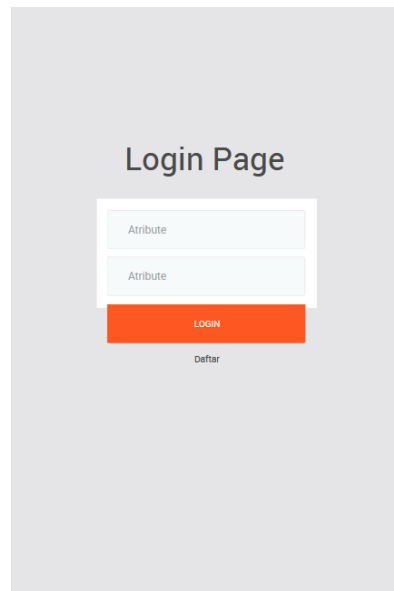


#### 4.5.2 Desain *Interface* Aplikasi Android *Customer*

Desain *interface* aplikasi android *customer* dibagi menjadi desain halaman *login*, desain halaman utama, desain halaman pemesanan, dan desain halaman *profile* yang dapat dijelaskan sebagai berikut:

##### 1. Desain Halaman *Login*

Halaman *login* merupakan halaman pertama yang akan dijumpai oleh *customer* didalam aplikasi android. Pada halaman ini, *customer* harus melakukan *login* terlebih dahulu untuk masuk ke dalam aplikasi dengan memasukkan *username* dan *password*. Jika *customer* belum memiliki akun dapat memilih menu daftar dan akan berpindah ke halaman pendaftaran. Berikut adalah desain halamam *login* aplikasi:



Gambar 4.17 Desain Halaman *Login*

##### 2. Desain Halaman Utama

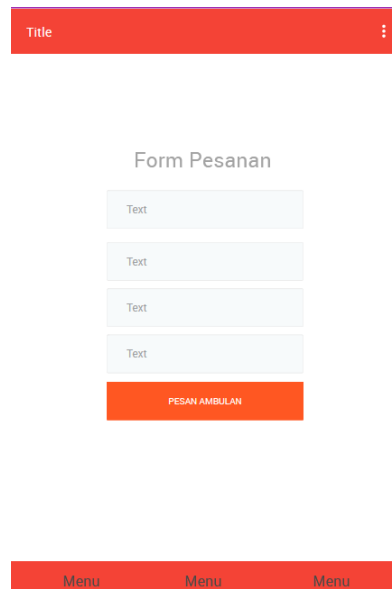
Setelah *customer* berhasil melakukan *login* maka akan menuju halaman utama, halaman ini merupakan halaman yang memperlihatkan peta lokasi *customer* terkini dan lokasi ambulans. Halaman ini terdiri dari, *title*, menu *logout*, tombol darurat, dan menu-menu. Adapun desain dari halaman utama dapat dilihat pada gambar 4.18 berikut ini:



Gambar 4.18 Desain Halaman Utama

### 3. Desain Halaman Pemesanan

Ketika *customer* menekan tombol darurat pada halaman utama maka akan menuju halaman pemesanan. Pada halaman ini, *ccustomer* memasukkan data yang diperlukan untuk melakukan pemesanan ambulans, lalu klik tombol pesan ambulans untuk melakukan pemesanan. Adapun desain halaman pemesanan dapat dilihat pada gambar 4.19 berikut ini:



Gambar 4.19 Desain Halaman Pemesanan

#### 4. Desain Halaman *Profile*

Setelah *customer* melakukan *login*, *customer* belum bisa melakukan pemesanan terhadap ambulans, *customer* diharuskan untuk melakukan pendaftaran akun terlebih dahulu pada halaman *profile*. *Customer* diharuskan mengisi data-data yang diperlukan untuk pendaftaran lalu klik daftar jika data sudah merasa benar. Lalu admin akan melakukan proses verifikasi terhadap data tersebut. Jika data sudah sesuai maka admin akan mengaktifkan data *customer* dan kemudian dapat melakukan pemesanan ambulans. Adapun desain dari halaman *profile* dapat dilihat pada gambar 4.20 dibawah ini:

The image shows a user registration form titled "DAFTAR AKUN". At the top, there is a red header bar with the word "Title" on the left and three vertical dots on the right. Below the header, the text "DAFTAR AKUN" is centered. Underneath, there are four light blue text input fields, each containing the placeholder text "Text". At the bottom of the form is a red button with the word "DAFTAR" in white capital letters. Below the form, there is a red footer bar containing three "Menu" items.

Gambar 4.20 Desain Halaman *Profile*