

## BAB III

### METODE PENELITIAN

#### 3.1 Analisa Permasalahan Sistem

Dalam bab ini akan dijelaskan mengenai arsitektur dari menentukan jarak terdekat lokasi/tempat perawatan bayi dengan *Haversine Formula* dan juga mengenai cara kerja tiap bagian dari aplikasi tersebut. Dalam sistem yang ada saat ini, aplikasi hanya bersifat pemesanan sehingga hal ini menimbulkan kebingungan bagi para pengguna jika mereka ingin mencari tempat perawatan bayi terdekat dengan posisi mereka. Pada aplikasi ini akan diberikan rekomendasi tempat perawatan bayi terdekat dengan posisi penggunanya, aplikasi ini juga memberikan informasi mengenai tempat perawatan bayi. Dengan ada rekomendasi tempat perawatan bayi terdekat ini bertujuan agar pengguna tidak mendapatkan biaya serta rute untuk menuju lokasi/tempat perawatan bayi lebih cepat, dan sesuai yang diterapkan. Metode yang digunakan adalah Algoritma *Haversine Formula*, algoritma ini dipakai untuk menentukan jarak terdekat antara posisi pengguna dengan lokasi/tempat perawatan bayi. Algoritma ini akan menghitung jarak antara dua titik tersebut dengan memanfaatkan bentuk permukaan bumi yang bulat. Selanjutnya sistem akan menampilkan hasil berupa daftar tempat perawatan bayi yang terdekat dengan posisi pengguna.

##### 3.1.1 Penerapan Rumus Haversine

Untuk mengetahui bahwa rumus *Haversine* dapat menghitung jarak antara dua buah titik dipermukaan bumi maka perlu dilakukan uji coba berikut ini merupakan penjabaran rumus *Haversine* di dalam *query MySQL*.

R = 6371 km.

lats = latitude perawatan bayi

lngs = longitude perawatan bayi

\$lat = latitude perawatan bayi

\$lngs = longitude user

1 derajat = 0.0174532925 radian.

Dari penjabaran di atas maka didapatkan rumus dalam query MySQL di bawah ini.

**(6371\*ACOS(SIN(RADIANS(lats))\*SIN(RADIANS(\$lat))+COS(RADIANS(ln  
gs-\$lng))\*COS(RADIANS(lats))\*COS(RADIANS(\$lat))))**

### 1. Hanan Holistic Care

#### Lokasi User

lat 1 \*  $\pi$  = -5.377409 \* 0.0174532925

lat 1  $r$  = -0,0938534921691325

lon 1 \*  $\pi$  = 105.247456 \* 0.0174532925

lon 1  $r$  = 1.83691463444888

#### Lokasi Hanan Holitic Care

lat 2 \*  $\pi$  = -5.3743503 \* 0.0174532925

lat 2  $r$  = -0.09380010778336275

lon 2 \*  $\pi$  = 105.2249151 \* 0.0174532925

lon 2  $r$  = 1.83652122152796675

#### Menentukan x dan y

$x = (\text{lon } 2 - \text{lon } 1) * \cos((\text{lat } 1 + \text{lat } 2)/2)$

$x = (1.83652122152796675 - 1.83691463444888) *$

$\cos((-0.0938534921691325 + (-0.09380010778336275))/2)$

$x = -0.00039341292091325 * 0.99999865915538600017823099608205$

$x = -3.9341239340765391553089464126592e-4$

$y = (\text{lat } 2 - \text{lat } 1)$

$y = (-0.09380010778336275 - (-0.0938534921691325))$

$y = 0.00005338438576975$

hasil  $x^2 = 1.5477331128673865401583996735027e-7$

hasil  $y^2 = 2.8498926440134863000150625e-9$

$$\text{hasil } x^2 + y^2 = 1.5762320393075214031585502985027e-7$$

$$\text{hasil } \sqrt{x^2 + y^2} = 3.9701788867852307432487962287859e-4 * 6371$$

$$\text{hasil akhir} = 2.5294009687708705065238080773595$$

$$\text{hasil pembulatan} = 2.5 \text{ km}$$

## 2. Ratu Baby Spa Lampung

### Lokasi User

$$\text{lat 1} * \pi = -5.377409 * 0.0174532925$$

$$\text{lat 1 } r = -0.0938534921691325$$

$$\text{lon 1} * \pi = 105.247456 * 0.0174532925$$

$$\text{lon 1 } r = 1.83691463444888$$

### Lokasi Ratu Baby Spa Lampung

$$\text{lat 2} * \pi = -5.39000617 * 0.0174532925$$

$$\text{lat 2 } r = -0.094073354261814725$$

$$\text{lon 2} * \pi = 105.2193577 * 0.0174532925$$

$$\text{lon 2 } r = 1.83642422660022725$$

### Menentukan x dan y

$$x = (\text{lon 2} - \text{lon 1}) * \cos((\text{lat 1} + \text{lat 2})/2)$$

$$x = (1.83642422660022725 - 1.83691463444888) *$$

$$\cos((-0.0938534921691325 + (-0.094073354261814725))/2)$$

$$x = -0.00049040784865275 * 0.99999865524767802594893109504269$$

$$x = -4.9040718917565680991542528765833e-4$$

$$y = (\text{lat 2} - \text{lat 1})$$

$$y = (-0.094073354261814725 - (-0.0938534921691325))$$

$$y = -0.000219862092682225$$

$$\text{hasil } x^2 = 2.4049840380037620170418684939571e-7$$

$$\text{hasil } y^2 = 4.8339339798607295894830950625e-8$$

$$\text{hasil } x^2 + y^2 = 2.8883855099377574168434835161474e-7$$

$$\text{hasil } \sqrt{x^2 + y^2} = 5.3743702049056477486766221768067e-4 * 6371$$

$$\text{hasil akhir} = 3.4240112575453881806818759888436$$

$$\text{hasil pembulatan} = 3.4 \text{ km}$$

### 3. Asiyah Mom Baby Care

#### Lokasi User

$$\text{lat 1} * \pi = -5,377409 * 0,0174532925$$

$$\text{lat 1 } r = -0,0938534921691325$$

$$\text{lon 1} * \pi = 105,247456 * 0,0174532925$$

$$\text{lon 1 } r = 1,83691463444888$$

#### Lokasi Asiyah Mom Baby Care

$$\text{lat 2} * \pi = -5.371206 * 0.0174532925$$

$$\text{lat 2 } r = -0.093745229395755$$

$$\text{lon 2} * \pi = 105.2170884 * 0.0174532925$$

$$\text{lon 2 } r = 1.836384619843557$$

#### Menentukan x dan y

$$x = (\text{lon 2} - \text{lon 1}) * \cos((\text{lat 1} + \text{lat 2})/2)$$

$$x = (1.836384619843557 - 1.83691463444888) *$$

$$\cos((-0.0938534921691325 + -0.093745229395755)/2)$$

$$x = -0.000530014605323 * 0.99988438412531541854438007876786$$

$$x = -5.2995332722080997813268616260785e-4$$

$$y = (\text{lat 2} - \text{lat 1})$$

$$y = (-0.093745229395755 - (-0.0938534921691325))$$

$$y = 0.0001082627733775$$

$$\text{hasil } x^2 = 2.8085052903240689414118560508474e-7$$

$$\text{hasil } y^2 = 0.00000001172082809938792275750625$$

$$\text{hasil } x^2 + y^2 = 2.9257135713179481689869185508474e-7$$

$$\text{hasil } \sqrt{x^2 + y^2} = 5.4089865698834455222039496173292e-4 * 6371$$

$$\text{hasil akhir} = 3.4460653436727431421961363012004$$

$$\text{hasil pembulatan} = 3.4 \text{ km}$$

#### 4. Klinik Krakatau

##### Lokasi User

$$\text{lat 1} * \pi = -5.377409 * 0.0174532925$$

$$\text{lat 1 } r = -0.0938534921691325$$

$$\text{lon 1} * \pi = 105.247456 * 0,0174532925$$

$$\text{lon 1 } r = 1.83691463444888$$

.....

##### Lokasi Klinik Krakatau

$$\text{lat 2} * \pi = -5.393986 * 0.0174532925$$

$$\text{lat 2 } r = -0.094142815398905$$

$$\text{lon 2} * \pi = 105.219759 * 0.0174532925$$

$$\text{lon 2 } r = 1.8364312306065075$$

.....

##### Menentukan x dan y

$$x = (\text{lon 2} - \text{lon 1}) * \cos((\text{lat 1} + \text{lat 2})/2)$$

$$x = (1.8364312306065075 - 1.83691463444888) *$$

$$\cos((-0,0938534921691325 + (-0,094142815398905))/2)$$

$$x = -0.0004834038423725 * 0.99999865425340531983256252050309$$

$$x = -4.8340319183342527189969300075403e-4$$

$$y = (\text{lat 2} - \text{lat 1})$$

$$y = (-0.094142815398905 - (-0.0938534921691325))$$

$$y = -0.0002893232297725$$

.....

$$\text{hasil } x^2 = 2.336786458747433534873061408108e-7$$

$$\text{hasil } y^2 = 0.00000008370793128599083040175625$$

$$\text{hasil } x^2 + y^2 = 3.173865771607341838890623908108e-7$$

$$\text{hasil } \sqrt{x^2 + y^2} = 5.633707279942171293367490591539e-4 * 6371$$

$$\text{hasil akhir} = 3.5892349080511573310044282558695$$

$$\text{hasil pembulatan} = 3.5$$

## 5. Salon Bocah

### Lokasi User

$$\text{lat 1} * \pi = -5.377409 * 0.0174532925$$

$$\text{lat 1 } r = -0,0938534921691325$$

$$\text{lon 1} * \pi = 105.247456 * 0,0174532925$$

$$\text{lon 1 } r = 1,83691463444888$$

.....

### Lokasi Salon Bocah

$$\text{lat 2} * \pi = -5.4188452 * 0.0174532925$$

$$\text{lat 2 } r = -0.094576690287821$$

$$\text{lon 2} * \pi = 105.2510995 * 0.0174532925$$

$$\text{lon 2 } r = 1.83697822552010375$$

.....

### Menentukan x dan y

$$x = (\text{lon 2} - \text{lon 1}) * \cos((\text{lat 1} + \text{lat 2})/2)$$

$$x = (1.83697822552010375 - 1.83691463444888) *$$

$$\cos(-0.0938534921691325 + (-0.094576690287821))/2)$$

$$x = 0.00006359107122375 * 0.99999864803456727782391003826324$$

$$x = 6.359098525081987571611141715391e-5$$

$$y = (\text{lat 2} - \text{lat 1})$$

$$y = (-0.094576690287821 - (-0.0938534921691325))$$

$$y = -0.0007231981186885$$

.....

$$\text{hasil } x^2 = 4.0438134051699909716408208273717e-9$$

$$\text{hasil } y^2 = 0.00000052301551887458573296003225$$

$$\text{hasil } x^2 + y^2 = 5.2705933227975572393167307082737e-7$$

$$\text{hasil } \sqrt{x^2 + y^2} = 7.2598852076307358369068410110729e-4 * 6371$$

$$\text{hasil akhir} = 4.6252728657815418016933484081546$$

$$\text{hasil pembulatan} = 4.6 \text{ km}$$

**Tabel 3.1** Hasil Perhitungan Manual

No.	Lokasi User	Latitude User	Longitude User	Alamat Tempat Perawatan Bayi	Latitude Tempat Perawatan Bayi	Longitude Tempat Perawatan Bayi	Jarak (Km)
1.	Jl. ZA. Pagar Alam			Jalan Sagitarius, No 10, Rajabasah Indah, Pramuka.	-0.0938001 0778336275	1.83652122 152796675	2.5
2.	No.93, Gedong Meneng, Kec. Rajabasa,	-0.0938534 921691325	1.836914 63444888	Jalan. Pramuka, Sumber Rejo, Kec. Kemiling.	-0.09407335 4261814725	1.83642422 660022725	3.4
3.	Kota Bandar Lampung , Lampung 35141			Jalan Raden Gunawan 11 Rajabasa Blok c17, Perum Griya Kencana.	-0.09374522 9395755	1.83638461 9843557	3.4

**Tabel 3.1** (Lanjut)

4.				Jalan Imam Bonjol, Sumber Rejo, Langkapura.	-0.09414281 5398905	1.83643123 06065075	3.5
5.				Jalan Kartika No.40, Tanjung Karang.	-0,09457669 0287821	1.83697822 552010375	4.6

Dari tabel dapat di lihat hasil perhitungan pencarian lokasi perawatan bayi terdekat dengan posisi user yaitu pada hasil perhitungan nomor 2 dengan jarak 1.4 Km. Jarak lainnya digunakan untuk pembandingan dalam menentukan jarak terdekat agar *haversine formula* dapat menghitung dan memberi keputusan mana lokasi yang terdekat dengan *user*.

### 3.2 Metode Pengembangan Prangkat Lunak

Model yang digunakan dalam pengembangan sistem pada penelitian ini adalah dengan menerapkan model *prototype* untuk merancang aplikasi *Penerapan Metode Haversine Pada Layanan Perawatan Bayi di Bandar Lampung Berbasis Android*. Tahap-tahap yang dilakukan dalam pengembangan sistem pada penelitian ini adalah sebagai berikut :

#### 3.2.1.1 Metode Pengumpulan Data

Pengumpulan data diperlukan untuk mendapatkan data yang dibutuhkan dalam membangun aplikasi *Penerapan Metode Haversine Pada Layanan Perawatan Bayi di Bandar Lampung Berbasis Android*. Metode yang digunakan untuk mendapatkan data tersebut adalah sebagai berikut :

a. Observasi

Dalam metode observasi, penulis melakukan pengumpulan data dengan mengamati langsung pada objek penelitian di lokasi perawatan bayi wilayah Bandar Lampung.

b. Wawancara

Metode wawancara dilakukan dengan cara bertatap muka secara langsung dan melakukan proses tanya jawab atau wawancara kepada pemilik, karyawan, dan pelanggan.

c. Studi Literatur

Metode Studi Literatur dilakukan dengan mengumpulkan literatur yang bersumber dari jurnal, buku atau hasil penelitian orang lain yang berkaitan dengan obyek penelitian ini.

### **3.2.1.2 Perencanaan Cepat**

*Quick Plan* atau perencana cepat merupakan tahapan dengan melakukan analisis dan perencanaan setelah mendapatkan data-data dan informasi dari tahapan komunikasi atau interaksi dengan pemilik Perawatan Bayi.

1. Analisis Sistem yang berjalan

Proses sistem yang berjalan pada layanan Perawatan Bayi Bandar Lampung masih menggunakan cara konvensional.

2. Analisis Sistem yang diajukan

Dari analisa sistem yang berjalan maka penulis mengajukan sebuah sistem *Booking* online dimana terdapat aplikasi yang tertanam di handphone pelanggan agar dapat memudahkan pelanggan untuk melakukan komunikasi tentang *Booking* Perawatan Bayi *online*. Aplikasi ini dibuat menggunakan sistem berbasis Android.

### 3. Analisis Kebutuhan Perangkat Lunak

Dalam Analisis kebutuhan perangkat lunak yang digunakan untuk membangun sistem *Booking* Perawatan Bayi *online* di Bandar Lampung adalah sebagai berikut :

- a. Perangkat lunak sistem operasi pada Laptop adalah *Microsoft Windows 10 64 bit*.
- b. Perangkat lunak sistem operasi pada android minimal adalah Android versi 5.1.1 (Lollipop).
- c. Perangkat lunak untuk pembuatan program adalah Android Studio.
- d. *Software* pendukung lain.

#### 3.2.1.3 Analisis Kebutuhan Perangkat Lunak

Perangkat lunak yang dikembangkan dalam membangun aplikasi *Penerapan Metode Haversine Pada Layanan Perawatan Bayi di Bandar Lampung Berbasis Android* adalah berbasis *website* untuk pengelola sistem dan berbasis *Android* bagi pelanggan. Perangkat lunak yang disarankan untuk menjalankan aplikasi ini adalah sebagai berikut :

1. Sistem Operasi yang digunakan adalah *Windows 10*
2. *Web Server* menggunakan XAMPP
3. *Web Browser* Internet (*Mozilla Firefox/Google Chrome*)
4. *Editot Interface* menggunakan *Sublime Text*
5. Emulator Android
6. *Provider* yang digunakan adalah yang mempunyai koneksi stabil.

#### 3.2.1.4 Analisis Kebutuhan Perangkat Keras

Spesifikasi perangkat keras yang digunakan dalam perancangan sistem tersebut adalah satu unit Laptop Acer Aspire E14 Series dengan spesifikasi sebagai berikut:

1. *Processor Core I3*
2. *SSD 500 GB*
3. *RAM 12 GB*
4. *Keyboard dan Mouse standard*

### **3.3 Perancangan**

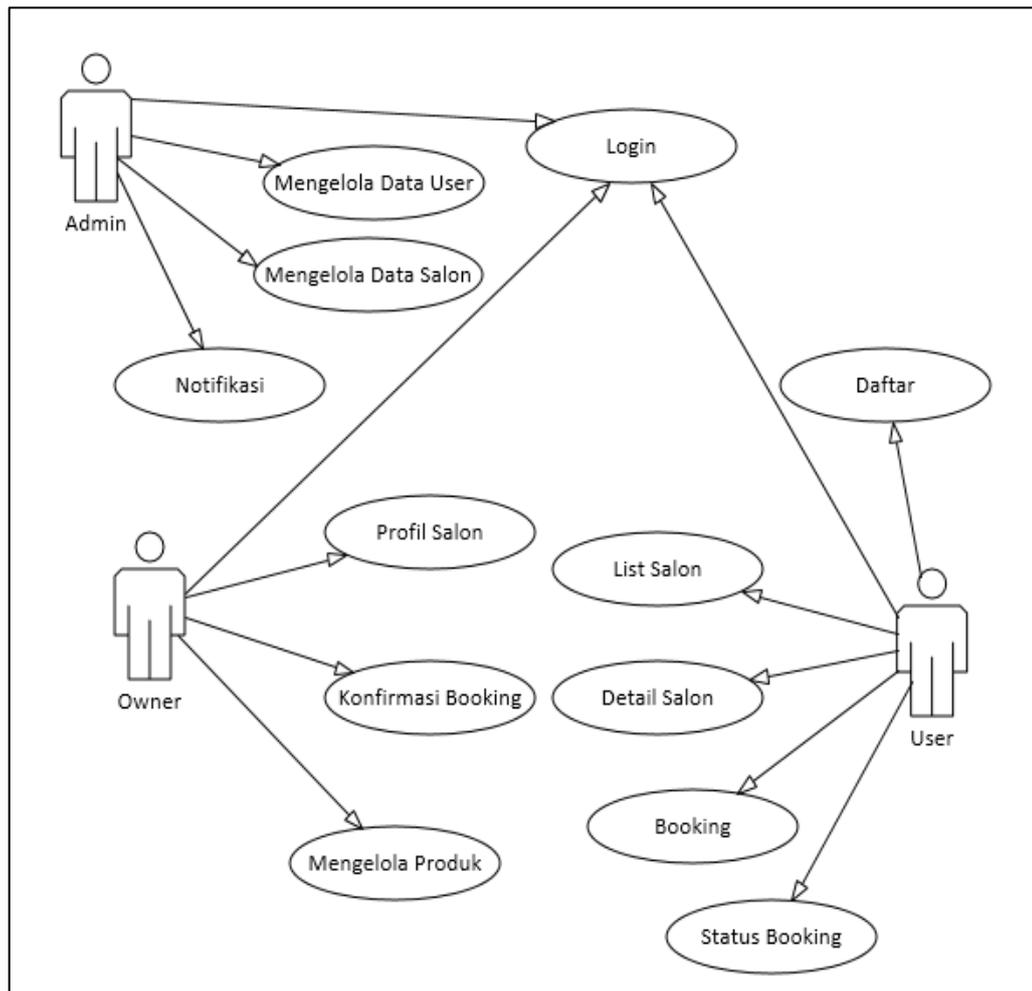
Perancangan merupakan tahapan yang dilakukan untuk memulai pembangunan sistem dimana disesuaikan dengan identifikasi pengumpulan kebutuhan yang telah dilakukan peneliti. Proses perancangan dimulai dari perancangan sistem yang telah disusulkan kemudian dilanjutkan dengan pembuatan perangkat lunak dimana berupa *Unified Modeling Language (UML)*, *Struktur Database*, dan *Perancangan Antarmuka (Interface)* sistem.

#### **3.3.1.1 Unified Modeling Language (UML)**

UML merupakan sebuah bahasa yang berdasarkan pada grafik atau gambar untuk memvisualisasikan dan mendokumentasikan informasi yang digunakan dalam proses pembuatan perangkat lunak. Informasi dalam pembuatan perangkat lunak berupa model dan atau deskripsi perangkat lunak. Terdapat berbagai macam jenis diagram yang digunakan untuk memvisualisasikan perangkat lunak, dalam membangun *Penerapan Metode Haversine Pada Layanan Perawatan Bayi di Bandar Lampung Berbasis Android* peneliti menggunakan diagram yaitu meliputi *Use Case Diagram*, *Class Diagram*, *Activity Diagram*, dan *Sequence Diagram*.

##### **a. Use Case Diagram**

*Use case Diagram* merupakan pemodelan sistem informasi yang digunakan untuk mendeskripsikan sebuah interaksi antara satu aktor atau lebih dengan aktor lainnya sesuai pada sistem yang ada. Dalam penelitian ini, aktor yang terlibat dalam aplikasi *Penerapan Metode Haversine Pada Layanan Perawatan Bayi di Bandar Lampung Berbasis Android* adalah *user* sebagai pihak pengguna layanan, *Admin* sebagai pengelola sistem dan *owner* sebagai pemilik perawatan bayi tersebut yang dapat dilihat pada Gambar 3.1.



Gambar 3.1 Use Case Diagram Aplikasi.

Berdasarkan Gambar 3.1 Use Case Diagram Sistem, skenario pendeskripsian dan pendefinisiannya dapat dilihat pada Tabel 3.2 dan Tabel 3.3.

**Tabel 3.2** Deskripsi Aktor.

No	Aktor	Deskripsi
1	User	Orang yang ingin melakukan pemesanan atau menggunakan jasa perawatan bayi.
2.	Admin	Orang yang mengelola seluruh prosedur yang ada pada sistem.
3.	Ower	Pengelola perawatan bayi atau pemilik.

Tabel 3.3 Skenario *Use Case Diagram* Aplikasi.

Aktor	No	Skenario	Sistem
User	1	Memasukkan <i>password</i> dan <i>username</i>	1. Jika belum memiliki <i>password</i> dan <i>username</i> diminta registrasi terlebih dahulu. 2. Masuk ke aplikasi perawatan bayi.
	2	Memulai <i>Booking</i> (pemesanan)	1. Memilih perawatan bayi terdekat dari lokasi pengguna 2. Memilih <i>booking</i> . 3. Mengisi <i>form booking</i>
	3.	Status <i>Booking</i>	1. Memilih menu status <i>booking</i> . 2. Klik batal jika ingin membatalkan.
Admin	1.	Memasukkan <i>username</i> dan <i>password</i>	1. Memeriksa valid tidaknya data masukkan. 2. Masuk ke dalam sistem.
	2.	Kelola Data <i>User</i>	Admin dapat menambah, menghapus, dan mengedit data <i>user</i> .
	3.	Kelola Data Perawatan Bayi	Admin dapat menambah, menghapus, dan mengedit data perawatan bayi
	4.	Notifikasi	Admin dapat mengirim notifikasi sesuai kebutuhan kepada pengguna aplikasi ( <i>user</i> ).
Ower	1.	Memasukkan <i>password</i> dan <i>username</i>	1. Jika belum memiliki <i>password</i> dan <i>username</i> diminta registrasi terlebih dahulu. 2. Masuk ke aplikasi perawatan bayi.
	2.	Konfirmasi <i>Booking</i>	1. <i>Owner</i> dapat mengkonfirmasi pemesanan jika ada pesanan. 2. <i>Owner</i> dapat membatalkan pemesanan jika tidak sesuai ketentuan.

Tabel 3.3 (lanjutan).

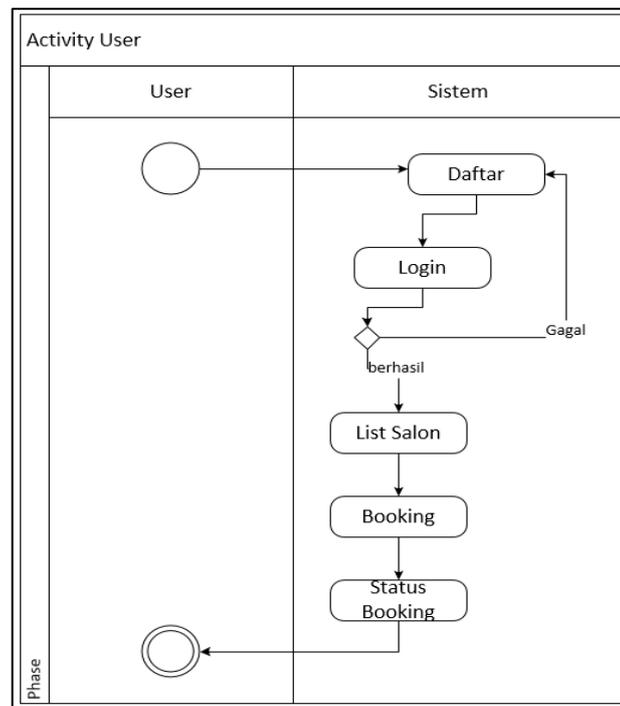
	3.	Kelola Data Produk	Admin dapat menambah, menghapus, dan mengedit data produk.
	4.	Profil Perawatan bayi	Owner dapat mengedit profil perawatan bayi sesuai dengan ketentuan.

### b. Activity Diagram

*Activity Diagram* merupakan aktivitas yang berfungsi untuk menggambarkan langkah-langkah aliran kerja dari sebuah sistem yang dirancang. Dalam membangun aplikasi perawatan bayi aktivitas yang terjadi dalam sistem terdapat 3 bagian yaitu :

#### 1. Activity Diagram User

Aliran kerja *User* yang terjadi dalam sistem dapat dilihat pada Gambar 3.2.



Gambar 3.2 Activity Diagram User Dengan Sistem.

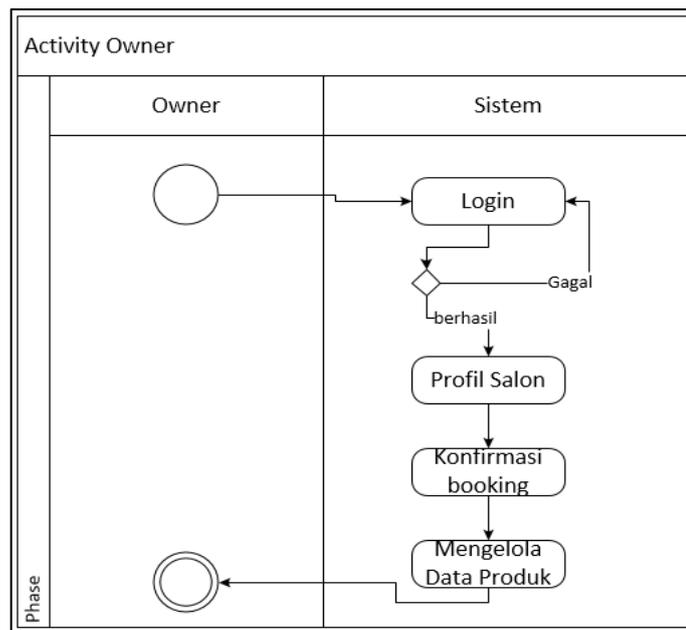
Berdasarkan Gambar 3.2 deskripsi *activity diagram User* dalam sistem dapat dilihat pada Tabel 3.4.

**Tabel 3.4** Skenario *Activity Diagram User*.

Aktor	Skenario
Customer	<p>Aliran kerja <i>user</i> dalam sistem yaitu :</p> <ol style="list-style-type: none"> <li>1. <i>User</i> melakukan registrasi untuk mendapatkan <i>username</i> dan <i>password</i>.</li> <li>2. <i>User</i> melakukan <i>login</i> untuk dapat mengakses fitur pada aplikasi perawatan bayi.</li> <li>3. <i>User</i> memilih menu perawatan bayi untuk melihat jasa perawatan bayi yang ingin <i>dibooking</i> (pesanan).</li> <li>4. Setelah memilih perawatan bayi, <i>User</i> diminta untuk mengisi <i>form booking</i>.</li> <li>5. Setelah melakukan proses <i>booking</i>, <i>User</i> dapat melihat status <i>booking</i> perawatan bayi pada menu status <i>booking</i>.</li> </ol>

## 2. Activity Diagram Owner

Aliran kerja *owner* dalam sistem perawatan bayi dapat dilihat pada Gambar 3.3.



Gambar 3.3 Activity Diagram Owner Dengan Sistem.

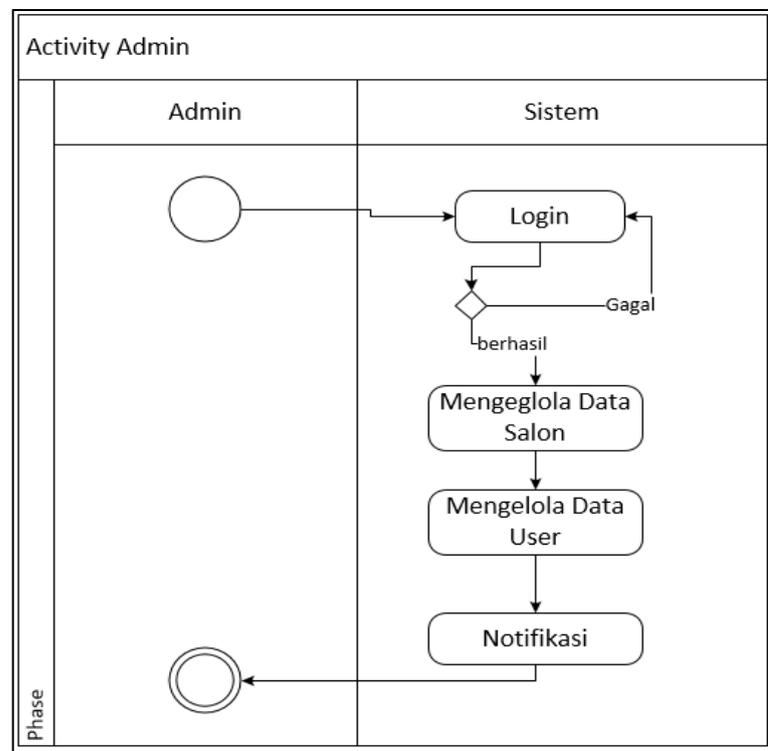
Berdasarkan Gambar 3.3 deskripsi *activity diagram owner* dalam sistem dapat dilihat pada Tabel 3.5.

**Tabel 3.5** Skenario *Activity Diagram Owner*.

Aktor	Deskripsi
Owner	<p>Aliran kerja admin pada sistem perawatan bayi yaitu :</p> <ol style="list-style-type: none"> <li>1. <i>Owner</i> melakukan <i>login</i> terlebih dahulu untuk mengakses seluruh fitur pada sistem.</li> <li>2. <i>Owner</i> dapat melakukan konfirmasi <i>booking</i> dan dapat membatalkannya jika tidak sesuai dengan ketentuan.</li> <li>3. <i>Owner</i> dapat mengubah profil perawatan bayi</li> <li>4. <i>Owner</i> dapat melakukan pengelolaan data produk.</li> </ol>

### 3. Activity Diagram Admin

Aliran kerja admin dalam sistem perawatan bayi dapat dilihat pada Gambar 3.4.



Gambar 3.4 *Activity Diagram Admin* Dengan Sistem.

Berdasarkan Gambar 3.4 deskripsi *activity diagram* admin dalam sistem dapat dilihat pada Tabel 3.6.

**Tabel 3.6** Skenario *Activity Diagram* Admin.

Aktor	Deskripsi
Admin	<p>Aliran kerja admin pada sistem perawatan bayi yaitu :</p> <ol style="list-style-type: none"> <li>1. Admin melakukan <i>login</i> terlebih dahulu untuk mengakses seluruh fitur pada sistem.</li> <li>2. Admin dapat melakukan pilihan untuk melakukan pengolahan data pada setiap menu yang terdapat pada sistem, diantaranya mengelola data <i>user</i>, mengelola data perawatan bayi, memberikan notifikasi.</li> <li>3. Admin dapat keluar dari sistem setelah melakukan prosedur pengelolaan data dalam sistem.</li> </ol>

### c. Sequence Diagram

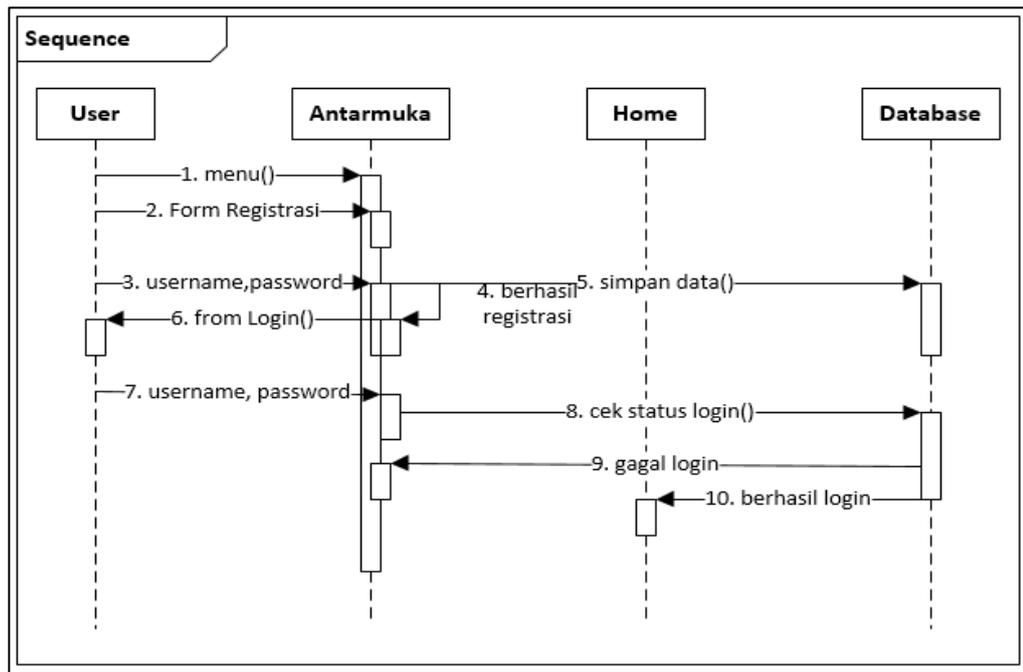
Diagram sekuen menggambarkan interaksi antar objek di dalam sistem, dimana interaksi tersebut berupa *message* yang digambarkan terhadap waktu. Dalam menggambarkan interaksi objek harus diketahui objek-objek yang terlibat dalam sebuah *use case*, sehingga cocok untuk menggambarkan model deskripsi *use case* menjadi spesifikasi desain. Diagram sekuen yang digunakan dalam merancang aplikasi *delivery order* memiliki 3 bagian Aktor, yaitu *User*, *Owner* dan *Admin*.

#### 1. Sequence Diagram (User)

Dalam aplikasi perawatan bayi, *user* harus melakukan registrasi untuk mendapatkan *username* dan *password* agar dapat *login* dan mengakses fitur *booking* dan status *booking*. Gambaran sequence diagram yang dilakukan *user* dalam aplikasi adalah sebagai berikut :

##### a. Sequence Diagram Proses Login

Proses *login* pada aplikasi diawali dengan melakukan registrasi seperti terlihat dalam diagram sekuen pada Gambar 3.5.



Gambar 3.5 *Sequence Diagram Login.*

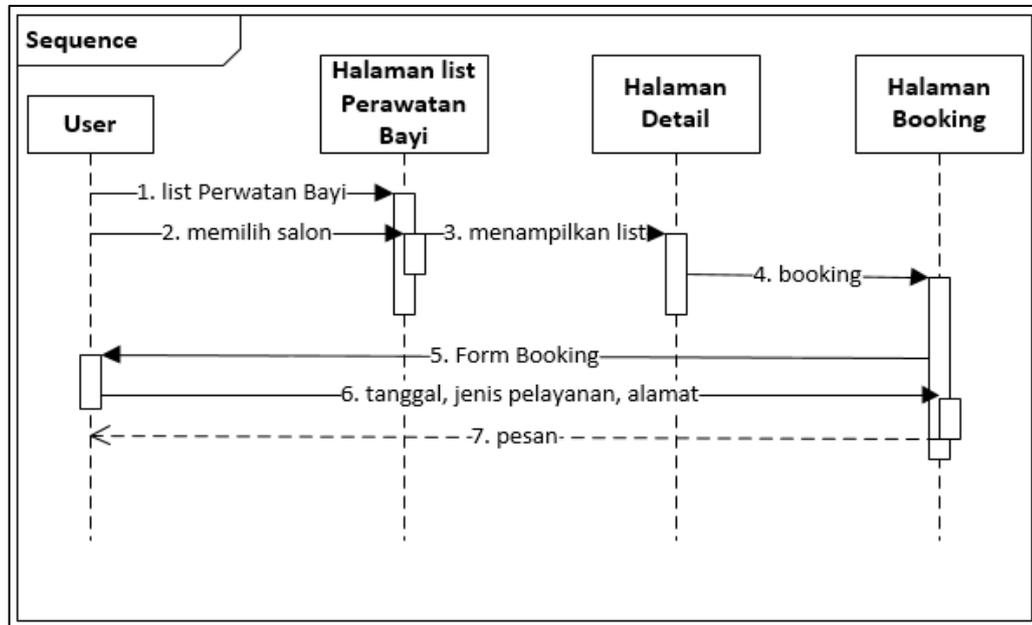
Berdasarkan gambar di atas, deskripsi dan skenario pada diagram sekuen proses *login* dapat dilihat pada Tabel 3.7.

Tabel 3.7 Skenario *Sequence Diagram Login.*

Objek	Deskripsi	Skenario
<i>User</i>	Pengguna dari aplikasi	Aplikasi dibuka, maka <i>user</i> diminta untuk registrasi, dengan mengisi <i>form</i> pada tampilan menu registrasi ( <i>username</i> dan <i>password</i> ). Data yang diisi akan tersimpan dalam <i>database users</i> dan pelanggan mendapat pesan telah berhasil melakukan registrasi, kemudian pelanggan secara otomatis akan diarahkan pada halaman <i>login</i> dengan memasukkan <i>username</i> dan <i>password</i> yang telah didaftarkan. Jika berhasil login maka akan terarah ke halaman <i>home</i> jika gagal kembali ke halaman <i>login</i> .
Antarmuka	Tampilan awal aplikasi	
<i>Home</i>	Tampilan berhasil <i>login</i>	
<i>Database</i>	Penyimpanan data <i>user</i>	

b. *Sequence Diagram Booking Perawatan bayi*

Proses sekuen pemesanan produk pada aplikasi dapat dilihat pada Gambar 3.6.



Gambar 3.6 *Sequence Diagram Booking Perawatan bayi*.

Berdasarkan gambar di atas, deskripsi dan skenario pada diagram sekuen pemesanan produk dapat dilihat pada Tabel 3.8.

**Tabel 3.8** Skenario *Sequence Diagram Booking Perawatan bayi*.

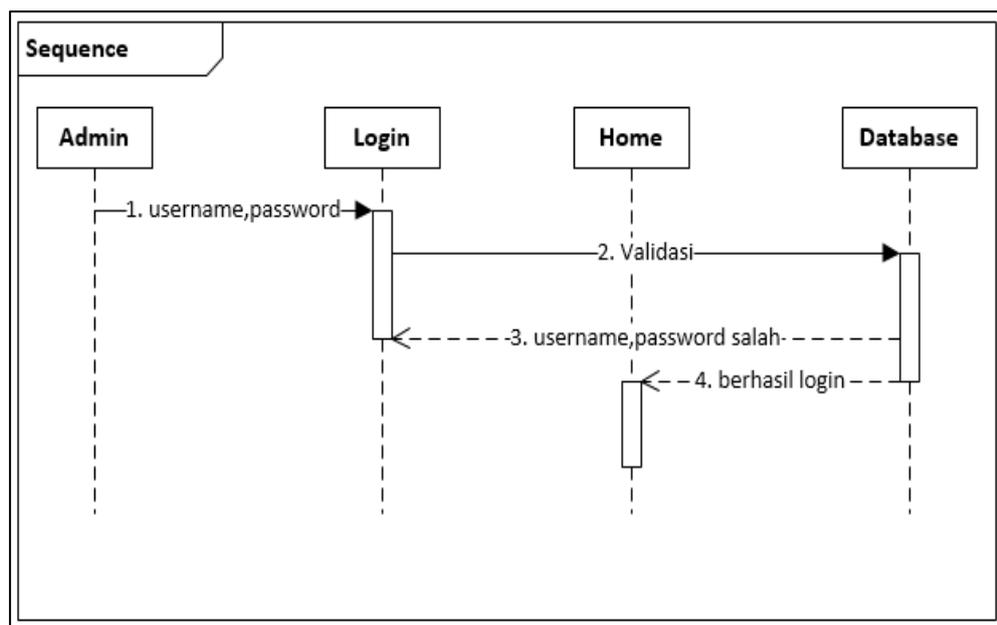
Objek	Deskripsi	Skenario
User	Pengguna dari aplikasi	Pelanggan yang berhasil melakukan prosedur <i>login</i> , dapat langsung memilih perawatan bayi sesuai dengan jarak terdekat dari pengguna dan dapat melakukan proses booking perawatan bayi dengan mengisi <i>form</i> yang telah disediakan.
Halaman List	Menampilkan list perawatan bayi	
Halaman Detail	Informasi detail perawatan bayi	
Halaman Booking	Menu <i>booking</i> perawatan bayi	

## 2. Sequence Diagram (Admin)

Dalam mengelola aplikasi perawatan bayi, Admin memiliki peran dalam mengelola data *user*, data perawatan bayi, dan notifikasi. Selain itu juga Admin harus mengelola proses-proses dari setiap objek-objek yang berkaitan dengan data-data yang masuk ke dalam sistem. Sehingga dalam proses tersebut, dapat dilihat gambarannya sebagai berikut :

### a. Sequence Diagram Login

Merupakan urutan/tahap awal dimana Admin harus melakukan *login* untuk dapat mengakses fitur-fitur yang terdapat pada sistem yang dapat dilihat pada Gambar 3.7.



Gambar 3.7 Sequence Diagram Login Admin.

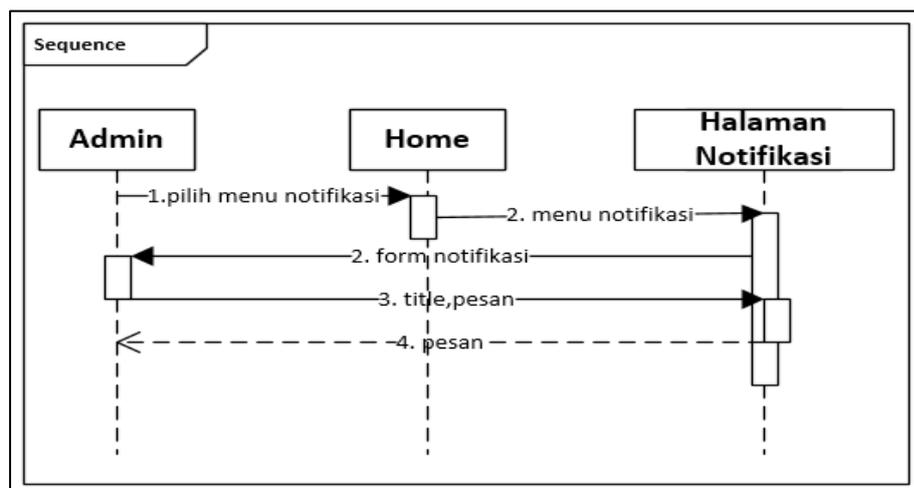
Berdasarkan gambar di atas, deskripsi dan skenario yang terjadi pada diagram sekuen dapat dilihat pada Tabel 3.9.

**Tabel 3.9** Skenario *Sequence Diagram Login Admin*.

Objek	Deskripsi	Skenario
Admin	Pengelola sistem perawatan bayi	Admin perlu melakukan <i>login</i> untuk dapat mengakses fitur-fitur sistem.
<i>Login</i>	Halaman validasi admin	Dengan memasukkan <i>username</i> dan <i>password</i> sistem akan melakukan operasi cek status <i>login</i> pada <i>database</i> . Jika berhasil <i>login</i> maka sistem akan menuju halaman <i>home</i> dan jika sebaliknya maka tetap di halaman <i>login</i> . Dan selesai.
<i>Home</i>	Halaman utama sistem	
<i>Database</i>	Penyimpanan data	

*b. Sequence Diagram Notifikasi*

Berdasarkan pendefinisian *use case* oleh peneliti, notifikasi merupakan pesan pemberitahuan kepada *user* (pengguna) yang telah di kirim oleh admin, Admin dapat mengisi *form title* dan pesan yang akan dikirim, dimana dapat dilihat pada Gambar 3.8.



Gambar 3.8 *Sequence Diagram Notifikasi*.

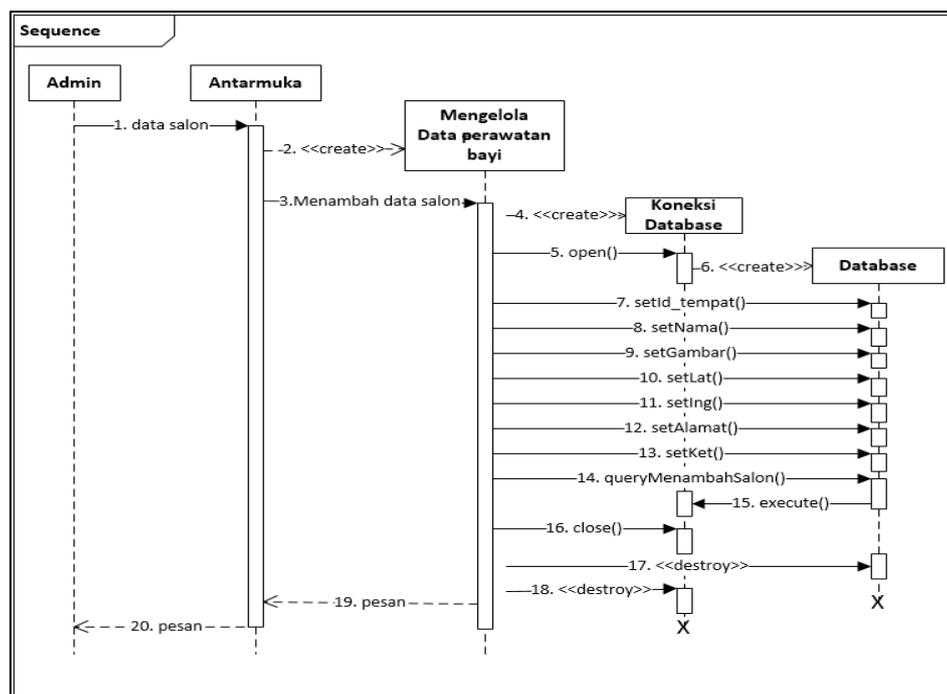
Berdasarkan gambar di atas, skenario dan deskripsi diagram sekuen yang terjadi pada proses hapus data *user* dapat dilihat pada Tabel 3.10.

**Tabel 3.10** Skenario *Sequence Diagram* Notifikasi.

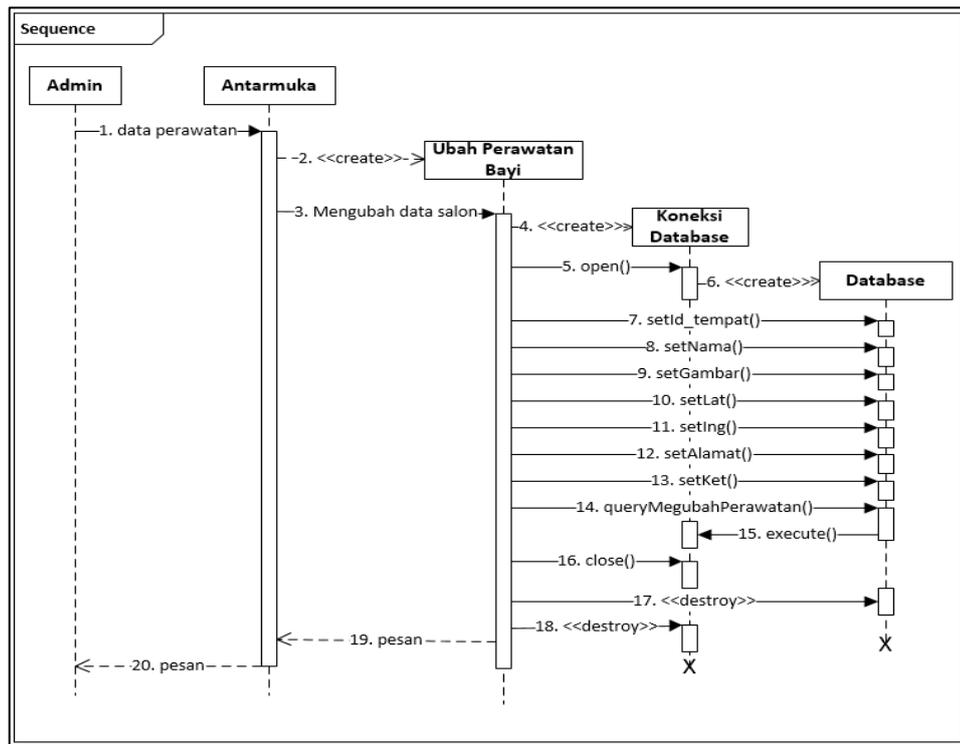
Objek	Deskripsi	Skenario
Admin	Pengelola sistem	Proses yang terjadi pada halaman notifikasi adalah Admin mengisi <i>form</i> , alurnya adalah : 1. Admin memilih menu notifikasi 2. Antarmuka akan memunculkan <i>form</i> pesan. 3. Admin mengisi <i>form</i> yang tersedia kemudian klik kirim.
<i>Home</i>	Halaman utama sistem	
Halaman Notifikasi	Halaman yang berisi Form pesan untuk memberikan notifikasi kepada <i>user</i>	

c. *Sequence Diagram* Kelola Perawatan bayi

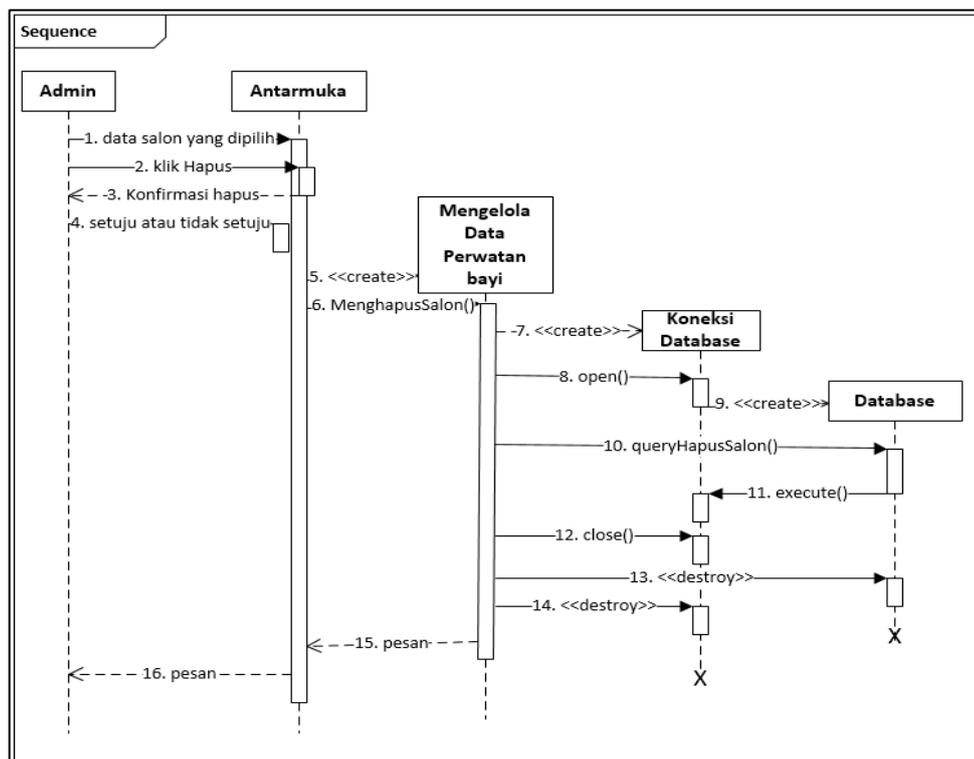
Dalam mengelola Perawatan bayi, Admin dapat melakukan eksekusi untuk menambahkan, mengubah, atau menghapus Perawatan bayi sesuai dengan data perusahaan. Alur diagram sekuen dalam proses ini dapat dilihat pada Gambar 3.9. Gambar 3.10, dan Gambar 3.11.



Gambar 3.9 *Sequence Diagram* Tambah Perawatan bayi.



Gambar 3.10 Sequence Diagram Ubah Perawatan bayi.



Gambar 3.11 Sequence Diagram Hapus Perawatan bayi.

Berdasarkan gambar di atas, skenario yang terjadi pada proses sekuen kelola perawatan bayi dapat dilihat pada Tabel 3.11.

**Tabel 3.11** Skenario *Sequence Diagram* Kelola Perawatan bayi.

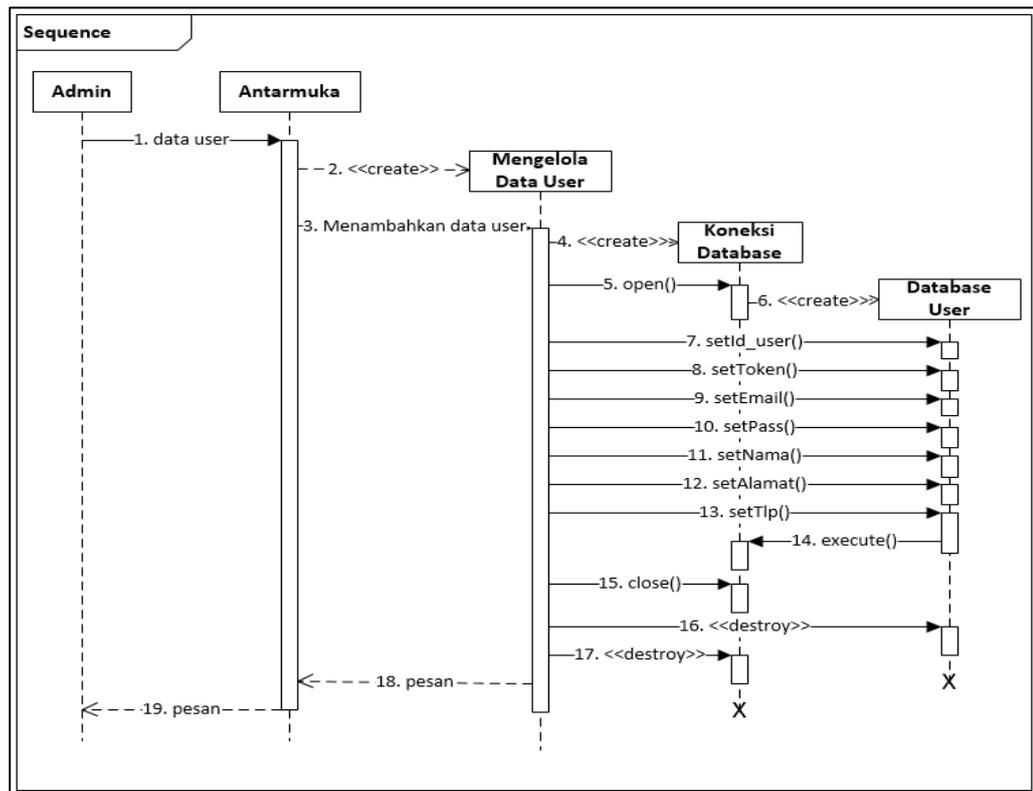
Proses	Skenario	Objek	Deskripsi
<b>Tambah Perawatan Bayi</b>	Menambahkan Perawatan bayi ke dalam sistem, alur prosesnya adalah :	Admin	Pengelola sistem.
	<ol style="list-style-type: none"> <li>1. Tampilan data perawatan bayi, pilih tombol tambah dan isi (<i>set</i>) setiap <i>field</i></li> <li>2. Dalam proses menyimpan data produk yang baru menggunakan <i>query</i> simpan dan dikoneksikan ke dalam <i>database</i>, tabel penyimpanan akan melakukan eksekusi hasil</li> <li>3. Data perawatan bayi yang baru otomatis tersimpan ke dalam tabel <i>database</i> dan sistem akan memberi pesan ke <i>admin</i> dengan tampilan data baru muncul dan selesai.</li> </ol>	Antarmuka	Tampilan sistem.
<b>Ubah Perawatan Bayi</b>	<p>Mengubah perawatan bayi dalam sistem, alur prosesnya adalah :</p> <ol style="list-style-type: none"> <li>1. Tampilan data perawatan bayi sistem, pilih produk yang akan diubah dengan mengklik tombol <i>edit</i> dan <i>set</i> data isi data yang akan diubah</li> </ol>	Kelola Perawatan bayi	Objek dalam prosedur pengelolaan perawatan bayi.

Tabel 3.11 (Lanjut)

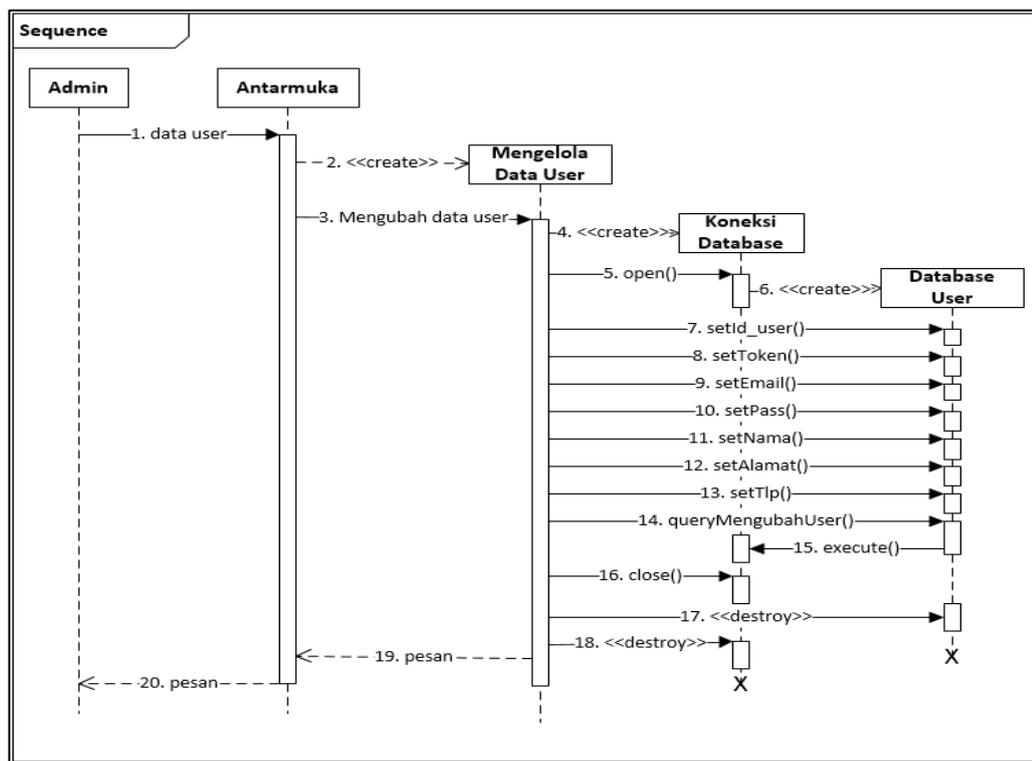
	2. Dalam proses operasi perubahan menggunakan <i>query</i> ubah dan dikoneksikan pada <i>database</i> sehingga tabel perawatan bayi akan melakukan eksekusi dan proses perubahan data perawatan bayi berhasil tersimpan dan selesai.	Koneksi <i>Database</i>	Operasi yang digunakan untuk terhubung dalam <i>database</i> .
<b>Hapus Perawatan Bayi</b>	Menghapus perawatan bayi dalam sistem, alur prosesnya adalah : 1. Tampil data perawatan bayi, pilih perawatan bayi yang akan dihapus dari <i>database</i> 2. Sistem akan memberikan pesan setuju/tidak setuju melakukan proses hapus 3. Prosedur untuk menghapus data dengan menggunakan <i>query</i> hapus, dan tabel perawatan bayi pada <i>database</i> akan melakukan eksekusi dan data berhasil dihapus dan selesai.	Perawatan Bayi	Tabel yang digunakan untuk menyimpan data perawatan bayi ke dalam <i>database</i> .

d. *Sequence Diagram* Kelola Data User

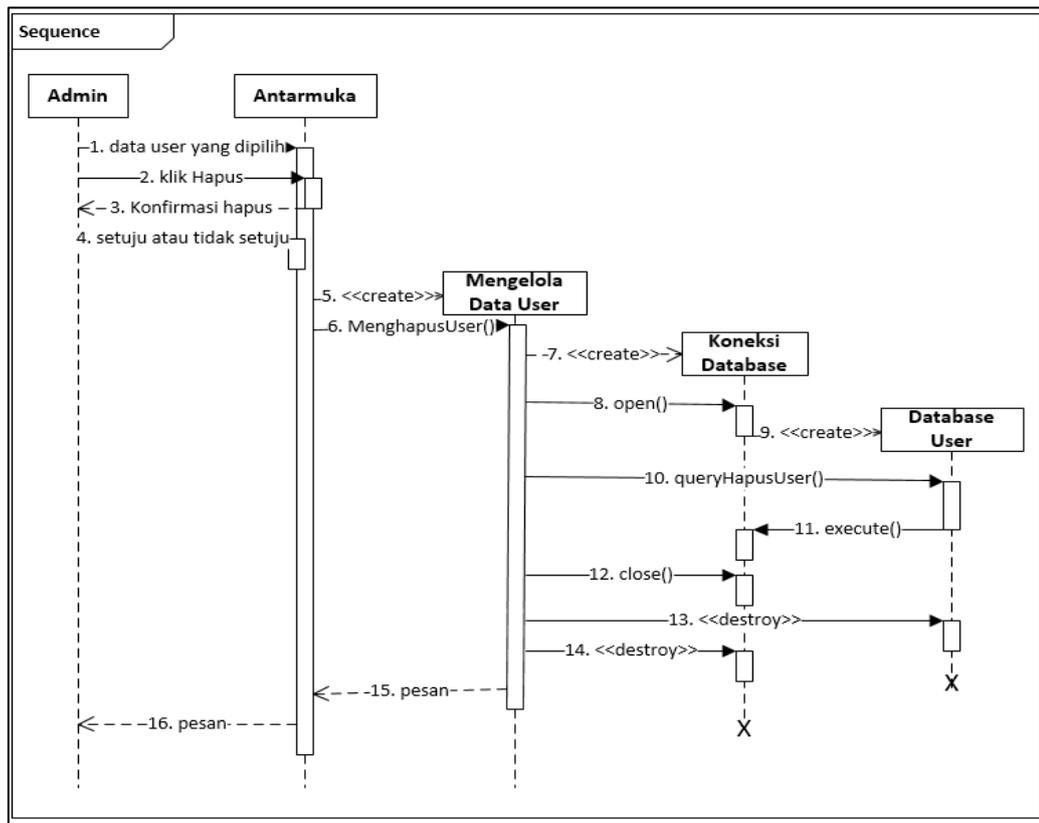
Data *user* berkaitan dengan data pengguna aplikasi yaitu *owner* dan *user*. Sehingga Admin dalam mengelola data *user* dapat melakukan penambahan *user*, mengubah *user*, atau menghapus *user* yang dalam diagram sekuen dapat dilihat pada Gambar 3.12, Gambar 3.13, dan Gambar 3.14.



Gambar 3.12 Sequence Diagram Tambah Data User.



Gambar 3.13 Sequence Diagram Ubah Data User.



Gambar 3.1 Sequence Diagram Hapus Data User.

Berdasarkan gambar di atas, skenario yang terjadi pada proses sekuen kelola data user dapat dilihat pada Tabel 3.14.

Tabel 3.12 Skenario Sequence Diagram Kelola Data User.

Proses	Skenario	Objek	Deskripsi
<b>Tambah Data User</b>	Menambahkan data <i>user</i> ke dalam sistem, alur prosesnya adalah:	Admin	Pengelola sistem.
	1. Tampilan data <i>user</i> , pilih tombol tambah dan isi ( <i>set</i> ) setiap <i>field</i> . 2. Dalam proses menyimpan data <i>user</i> yang baru, menggunakan <i>query</i> simpan dan dikoneksikan ke dalam <i>database</i> , tabel penyimpanan akan melakukan eksekusi hasil.	Antarmuka	Tampilan sistem.

Tabel 3.12 (Lajut)

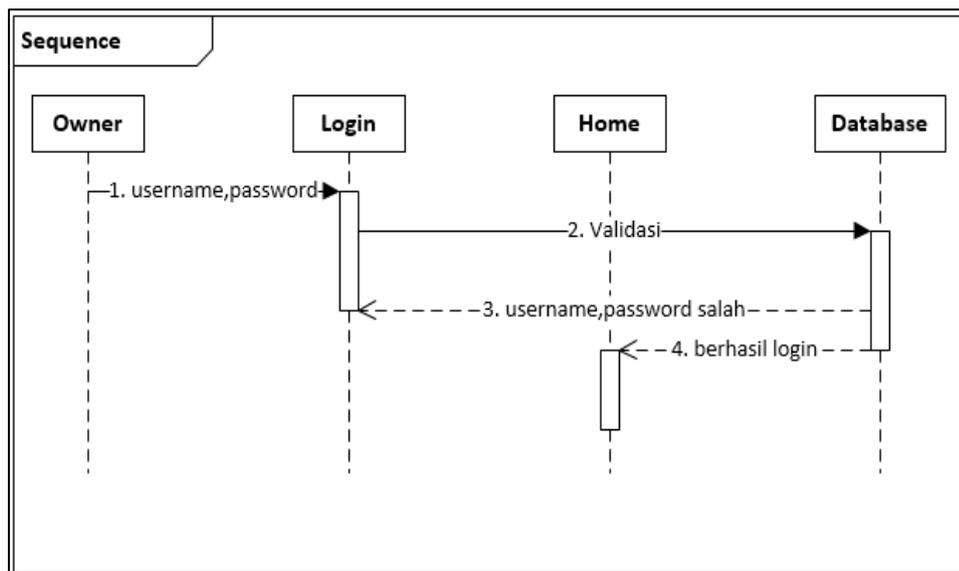
	3. Data <i>user</i> yang baru otomatis tersimpan ke dalam tabel <i>database</i> dan sistem akan memberi pesan ke <i>admin</i> dengan tampilan data baru muncul dan selesai.		.
<b>Ubah Data User</b>	Mengubah data <i>user</i> dalam sistem, alur prosesnya adalah : 1. Tampilan data <i>user</i> sistem, pilih <i>user</i> yang akan diubah dengan mengklik tombol <i>edit</i> dan <i>set field</i> data yang akan diubah.	Kelola Data <i>User</i>	Objek dalam prosedur pengelolaan data <i>user</i> .
	2. Dalam proses operasi pengubahan menggunakan <i>query</i> ubah dan dikoneksikan pada <i>database</i> sehingga tabel <i>user</i> akan melakukan eksekusi dan proses pengubahan data <i>user</i> berhasil tersimpan dan selesai.	Koneksi Database	Operasi yang digunakan untuk terhubung dalam <i>database</i> .
<b>Hapus Data User</b>	Menghapus <i>user</i> dalam sistem, alur prosesnya adalah : 1. Tampil data <i>user</i> , pilih jenis <i>user</i> yang akan dihapus dari <i>database</i> . 2. Sistem akan memberikan pesan setuju/tidak setuju melakukan proses hapus. 3. Prosedur untuk menghapus data dengan menggunakan <i>query</i> hapus dan dengan mengoneksikan dan eksekusi ke dalam <i>database</i> , maka data berhasil dihapus dan selesai.	Data <i>User</i>	Tabel yang digunakan untuk menyimpan data jenis cetakan dalam <i>database</i> .

### 3. Sequence Diagram (Owner)

*Owner* memiliki peran dalam mengelola data produk, konfirmasi *booking*, dan edit profil perawatan bayi. Selain itu juga *Owner* harus mengelola proses-proses dari setiap objek-objek yang berkaitan dengan data-data yang masuk ke dalam sistem. Sehingga dalam proses tersebut, dapat dilihat gambarannya sebagai berikut :

#### e. Sequence Diagram Login

Merupakan urutan/tahap awal dimana *Owner* harus melakukan *login* untuk dapat mengakses fitur-fitur yang terdapat pada sistem yang dapat dilihat pada Gambar 3.15.



Gambar 3.15 Sequence Diagram Login Owner.

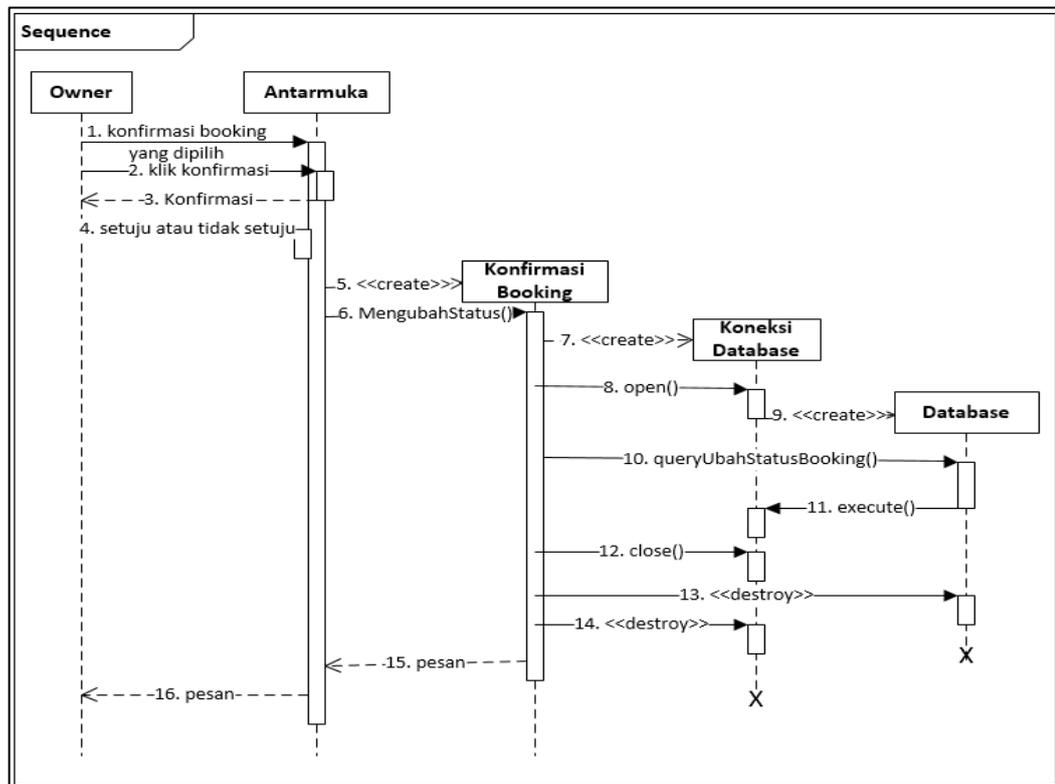
Berdasarkan gambar di atas, deskripsi dan skenario yang terjadi pada diagram sequence dapat dilihat pada Tabel 3.13.

**Tabel 3.13** Skenario *Sequence Diagram Login Owner*.

Objek	Deskripsi	Skenario
<i>Owner</i>	Pengguna sistem <i>perawatan bayi</i>	<i>Owner</i> perlu melakukan <i>login</i> untuk dapat mengakses fitur-fitur sistem. Dengan memasukkan <i>username</i> dan <i>password</i> sistem akan melakukan operasi cek status <i>login</i> pada <i>database</i> . Jika berhasil <i>login</i> maka sistem akan menuju halaman <i>home</i> dan jika sebaliknya maka tetap di halaman <i>login</i> . Dan selesai.
<i>Login</i>	Halaman validasi <i>owner</i>	
<i>Home</i>	Halaman utama sistem	
<i>Database</i>	Penyimpanan data	

*f. Sequence Diagram Konfirmasi Booking*

Berdasarkan pendefinisian *use case* oleh peneliti, notifikasi merupakan pesan pemberitahuan kepada *user* (pengguna) yang telah di kirim oleh admin, Admin dapat mengisi *form title* dan pesan yang akan dikirim, dimana dapat dilihat pada Gambar 3.16.



Gambar 3.16 *Sequence Diagram* Konfirmasi.

Berdasarkan gambar di atas, skenario dan deskripsi diagram sekuen yang terjadi pada proses hapus data *user* dapat dilihat pada Tabel 3.14.

**Tabel 3.14** Skenario *Sequence Diagram* Konfirmasi.

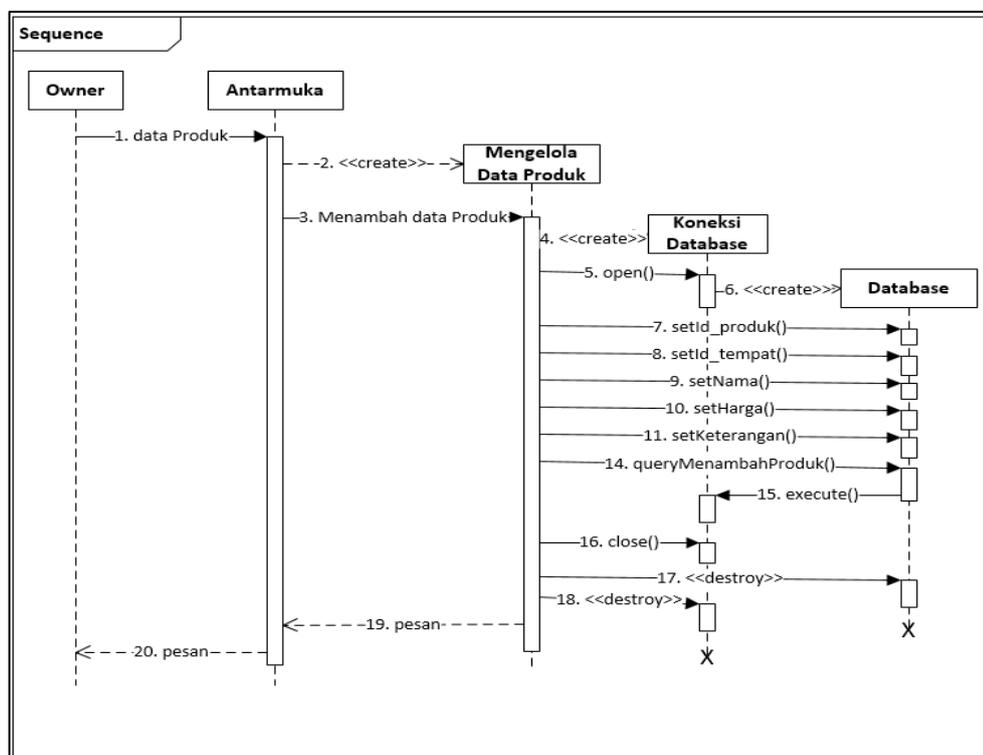
Objek	Deskripsi	Skenario
Admin	Pengelola sistem	Proses yang terjadi pada halaman Konfirmasi <i>booking</i> adalah :
Antarmuka	Halaman utama sistem	<ol style="list-style-type: none"> <li>1. <i>Owner</i> memilih data yang akan di konfirmasi.</li> <li>2. <i>Owner</i> mengklik <i>button</i> konfirmasi jika ingin mengkonfirmasi.</li> <li>3. <i>Owner</i> juga dapat mengklik <i>button</i> batal jika <i>booking</i> perawatan bayi tidak sesuai dengan ketentuan</li> </ol>

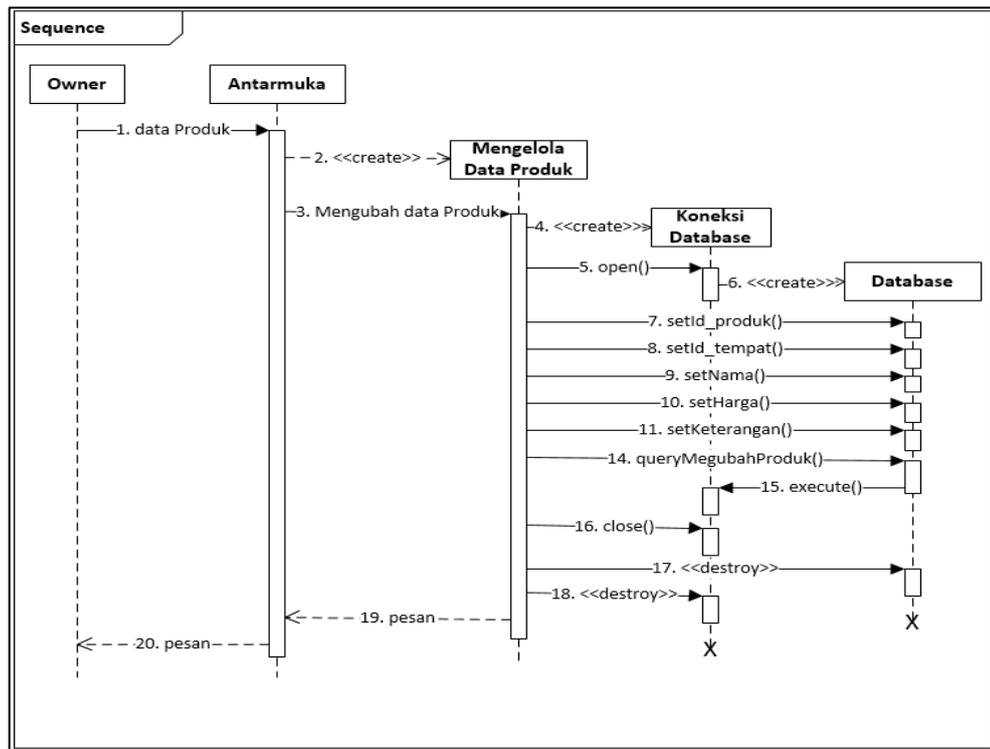
Tabel 3.14 (Lanjut)

Halaman Konfirmasi <i>Booking</i>	Halaman yang menampilkan data <i>booking</i> atau pemesanan perawatan bayi	
<i>Database</i>	Penyimpanan data	

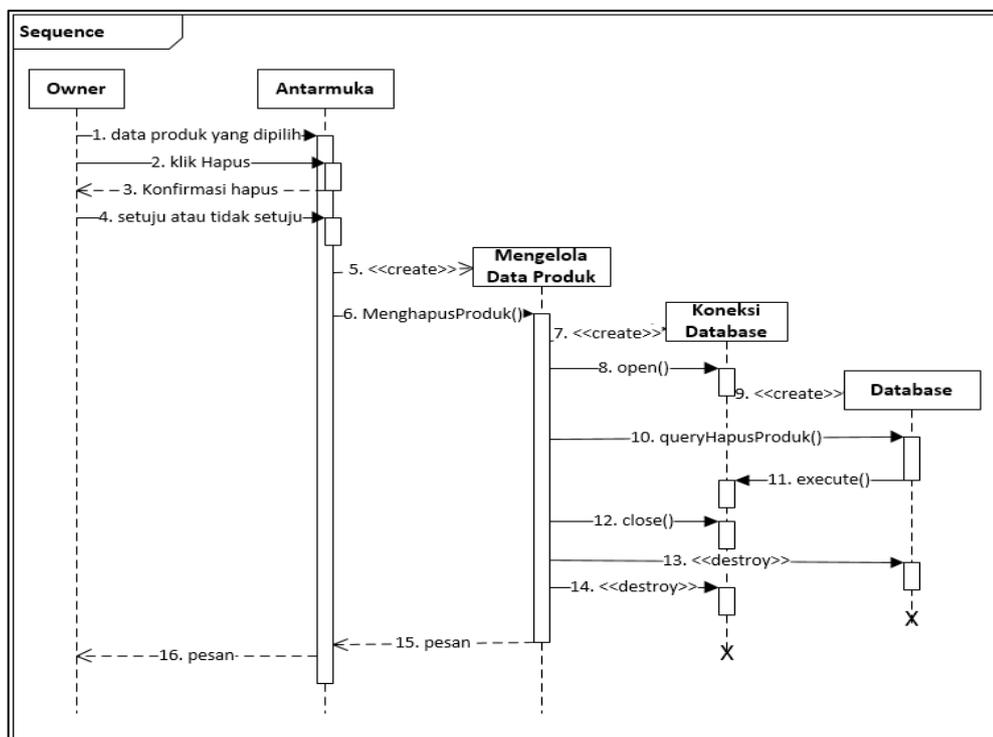
g. *Sequence Diagram* Kelola Produk

Dalam mengelola Produk, *Owner* dapat melakukan eksekusi untuk menambahkan, mengubah, atau menghapus Perawatan bayi sesuai dengan data perusahaan. Alur diagram sekuen dalam proses ini dapat dilihat pada Gambar 3.17, Gambar 3.18, dan Gambar 3.19.

Gambar 3.17 *Sequence Diagram* Tambah Produk.



Gambar 3.18 Sequence Diagram Ubah Produk.



Gambar 3.19 Sequence Diagram Hapus Produk.

Berdasarkan gambar di atas, skenario yang terjadi pada proses sekuen kelola produk dapat dilihat pada Tabel 3.15.

**Tabel 3.15** Skenario *Sequence Diagram* Kelola Produk.

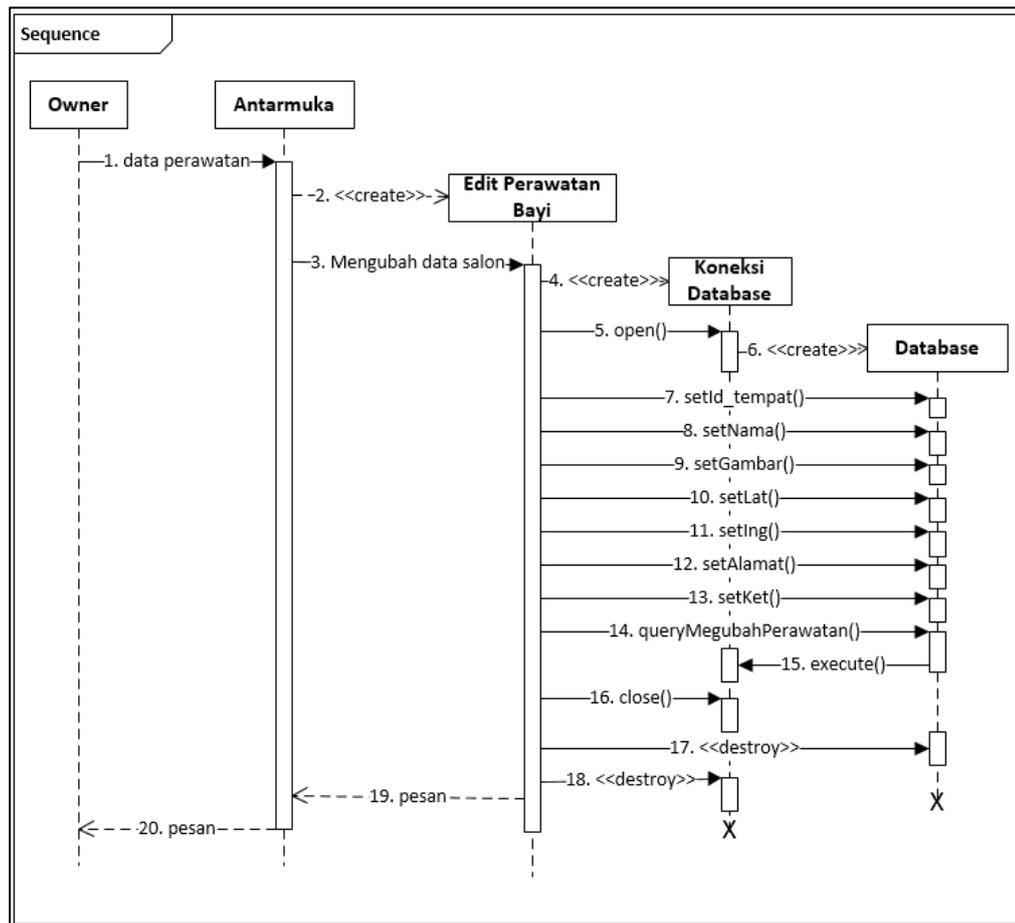
Proses	Skenario	Objek	Deskripsi
<b>Tambah Produk</b>	Menambahkan Produk ke dalam sistem, alur prosesnya adalah : 1. Tampilan data produk, pilih tombol tambah dan isi ( <i>set</i> ) setiap <i>field</i> . 2. Dalam proses menyimpan data produk yang baru menggunakan <i>query</i> simpan dan dikoneksikan ke dalam <i>database</i> , tabel penyimpanan akan melakukan eksekusi hasil. 3. Data produk yang baru otomatis tersimpan ke dalam tabel <i>database</i> dan sistem akan memberi pesan ke <i>admin</i> dengan tampilan data baru muncul dan selesai.	<i>Owner</i>	Pengelola sistem.
		Antarmuka	Tampilan sistem.
<b>Ubah Produk</b>	Mengubah produk dalam sistem, alur prosesnya adalah : 1. Tampilan data produk sistem, pilih produk yang akan diubah dengan mengklik tombol <i>edit</i> dan <i>set</i> data isi data yang akan diubah.	Kelola produk	Objek dalam prosedur pengelolaan produk.
		Koneksi <i>Database</i>	Operasi yang digunakan untuk terhubung

Tabel 3.15 (Lanjutan)

	2. Dalam proses operasi perubahan menggunakan <i>query</i> ubah dan dikoneksikan pada <i>database</i> sehingga tabel produk akan melakukan eksekusi dan proses perubahan data produk berhasil tersimpan dan selesai.		dalam <i>database</i> .
<b>Hapus Produk</b>	Menghapus produk dalam sistem, alur prosesnya adalah : 1. Tampil data produk, pilih produk yang akan dihapus dari <i>database</i> . 2. Sistem akan memberikan pesan setuju/tidak setuju melakukan proses hapus. 3. Prosedur untuk menghapus data dengan menggunakan <i>query</i> hapus, dan tabel produk pada <i>database</i> akan melakukan eksekusi dan data berhasil dihapus dan selesai.	Produk	Tabel yang digunakan untuk menyimpan data produk ke dalam <i>database</i> .

*h. Sequence Diagram* Kelola Edit Profil Perawatan bayi

Profil perawatan bayi berkaitan dengan data perawatan bayi. Sehingga *Owner* dapat melakukan mengubah data perawatan bayi yang dalam diagram sekuen dapat dilihat pada Gambar 3.20.



Gambar 3.20 *Sequence Diagram* Profil Perawatan bayi.

Berdasarkan gambar di atas, skenario dan deskripsi diagram sekuen yang terjadi pada proses hapus data *user* dapat dilihat pada Tabel 3.16.

**Tabel 3.16** Skenario *Sequence Diagram* Edit Profil Perawatan bayi.

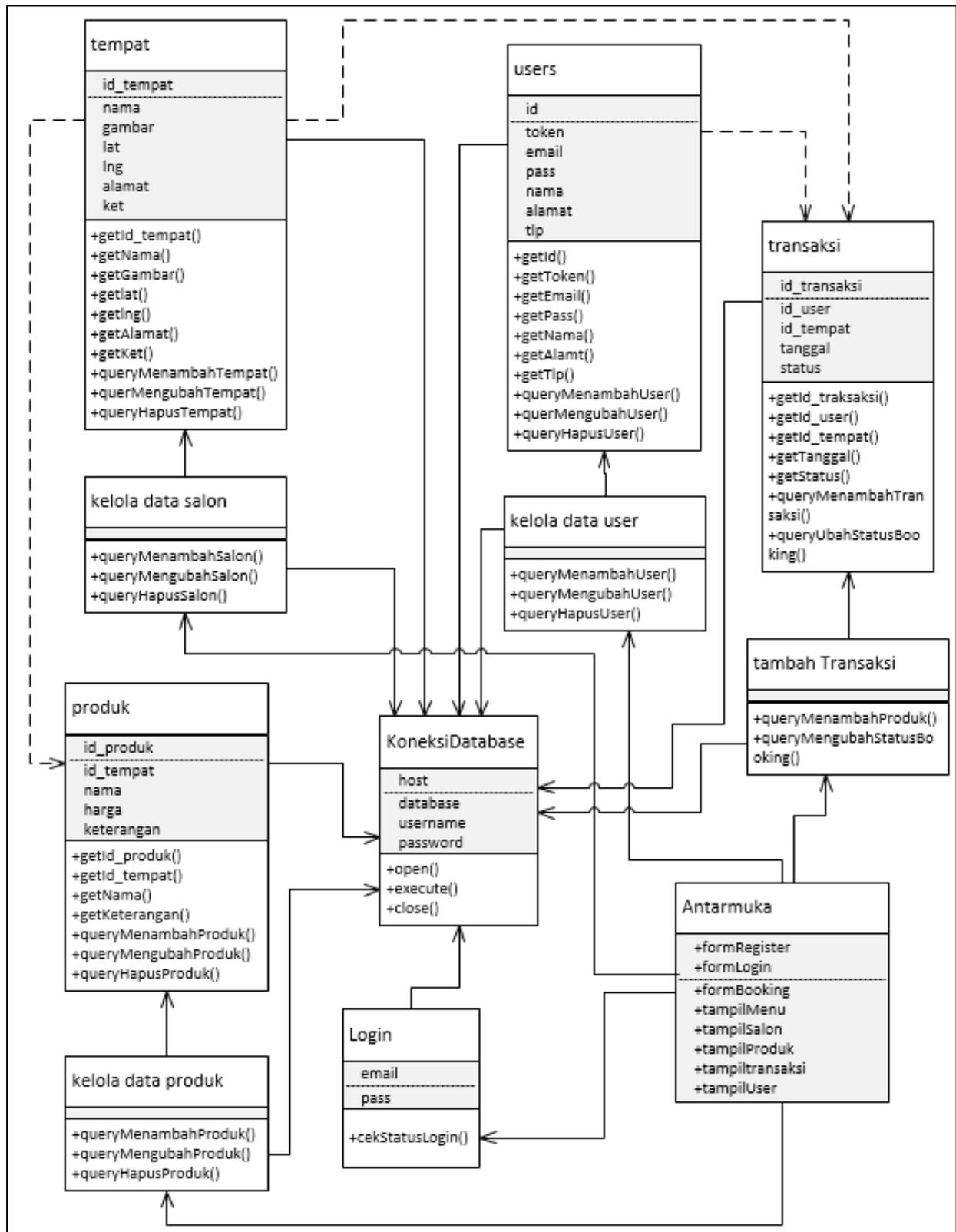
Objek	Deskripsi	Skenario
<i>Owner</i>	Pengelola sistem	Proses yang terjadi pada halaman edit profil perawatan bayi adalah : 1. <i>Owner</i> memilih edit profil. 2. Antarmuka akan menampilkan <i>form edit</i> profil. 3. <i>Owner</i> mengisi <i>form</i> yang telah di sediakan jika ingin merubahnya.
Antarmuka	Halaman utama sistem	

**Tabel 3.16 (Lanjut)**

Halaman Edit Profil Perawatan bayi	Halaman yang menampilkan data <i>booking</i> atau pemesanan perawatan bayi	4. Klik <i>button</i> simpan jika ingin menyimpan. 5. Klik <i>button</i> batal jika ingin membatalkanya
<i>Database</i>	Penyimpanan data	

**d. Class Diagram**

*Class diagram* merupakan diagram yang menggambarkan struktur dan komponen-komponen yang terkait pada sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Selain itu *class diagram* berfungsi menjelaskan tipe dari objek sistem dan hubungannya dengan objek lain. Objek merupakan nilai tertentu yang memiliki atribut dan metode atau operasi. *Class diagram* pada perancangan aplikasi *perawatan bayi* dapat dilihat pada Gambar 3.21.



Gambar 3.21 Class Diagram Aplikasi Perawatan bayi.

Deskripsi berdasarkan Gambar 3.21 *Class Diagram* Aplikasi *Perawatan bayi*, dapat dilihat pada Tabel 3.17.

**Tabel 3.17** Keterangan *Class Diagram* Aplikasi *Perawatan bayi*.

Nama Kelas	Keterangan
Antarmuka	Merupakan kelas yang menangani tampilan.
<i>Login</i>	Merupakan kelas proses yang diambil dari pendefinisian <i>use case login</i> .
Kelola data <i>User</i>	Merupakan kelas proses yang digunakan untuk mengelola data <i>user</i> yang dimiliki perusahaan, selain itu juga menangani proses menambah, mengubah, dan menghapus data <i>user</i> .
Kelola data Perawatan Bayi	Merupakan kelas proses yang digunakan untuk mengelola bahan perawatan bayi, sekaligus juga menangani proses menambah, mengubah, dan menghapus data perawatan bayi.
Kelola Data Produk	Merupakan kelas proses yang digunakan untuk mengelola bahan produk, sekaligus juga menangani proses menambah, mengubah, dan menghapus data produk.
Tambah Transaksi	Merupakan kelas proses yang digunakan untuk mengelola data pemesanan atau <i>booking</i> perawatan bayi yang masuk ke dalam sistem. Selain itu juga menangani proses <i>update</i> status <i>booking</i> perawatan bayi.
Data <i>User</i>	Merupakan kelas data yang digunakan untuk mengetahui jumlah <i>user</i> yang mengunjungi aplikasi melalui <i>form</i> registrasi.
Produk	Merupakan kelas data yang digunakan untuk memproses segala pengaksesan terhadap tabel produk.
<i>User</i>	Merupakan kelas data yang digunakan untuk memproses segala pengaksesan terhadap tabel <i>users</i> .

**Tabel 3.17** (Lanjut)

Transaksi	Merupakan kelas data yang digunakan untuk memproses segala pengaksesan terhadap tabel transaksi.
Koneksi <i>Database</i>	Merupakan kelas utilitas untuk koneksi ke <i>database</i> dan melakukan <i>query</i> .

**3.3.1.2 Struktur Database***a. Database haversine*

Nama Tabel : produk

Fungsi : Digunakan untuk menyimpan data produk

*Primary key* : id\_produk

**Tabel 3.18** Tabel Produk.

<i>Field Name</i>	<i>Type</i>	<i>Size</i>	<b>Deskripsi</b>
id_produk	Int	11	Id produk, primary key
Id_tempat	Int	11	Id tempat
Nama	Varchar	60	Nama produk
Harga	Double		Harga dari produk
keterangan	Text		Keterangan dari produk

*b. Database haversine*

Nama Tabel : tempat

Fungsi : Digunakan untuk menyimpan data tempat (perawatan bayi)

*Primary key* : id\_tempat

**Tabel 3.19** Tabel Tempat.

<i>Field Name</i>	<i>Type</i>	<i>Size</i>	<b>Deskripsi</b>
Id_tempat	Int	11	Id_tempat, primary key
Nama	Varchar	60	Nama tempat (perawatan bayi)
Gambar	Varchar	100	Gambar tempat
Lat	Double		Kordinat map lat dari tempat
Ling	Double		Kordinat map lng dari tempat
Alamat	Varchar	100	Alamat tempat
Ket	text		Keterangan dari tempat

c. *Database* haversine

Nama Tabel : *users*

Fungsi : Digunakan untuk menyimpan data pemesanan

*Primary key* : id

**Tabel 3.20** Tabel Users.

<i>Field Name</i>	<i>Type</i>	<i>Size</i>	<b>Deskripsi</b>
Id	Int	20	Id, primary key.
Token	Varchar	200	Token atau id aplikasi dari user
Email	Varchar	100	Email dari user, digunakan untuk login
Pass	Varchar	30	Password user, digunakan untuk login
Nama	Varchar	100	Nama dari user
Alamat	Text		Alamat dari user.
Tlp	Varchar	15	Nomer telephone dari user

d. Database harversine

Nama Tabel : transaksi

Fungsi : Digunakan untuk menyimpan data pemesan yang melakukan konfirmasi booking perawatan bayi

Primary key : id\_transaksi

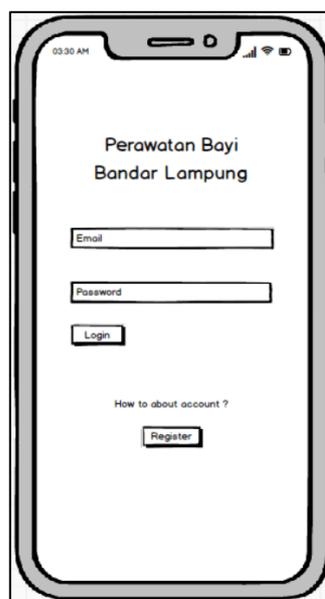
**Tabel 3.21** Tabel Transaksi.

<i>Field Name</i>	<i>Type</i>	<i>Size</i>	<b>Deskripsi</b>
id_transaksi	Int	30	Id transaksi, primary key,
Id_user	Int		Id user yang melakukan transaksi.
Id_tempat	Int	30	Id tempat yang di booking
Tanggal	Int	11	Tanggal dari transaksi atau booking
Status	Varchar	60	Status transaksi atau booking

### 3.3.1.3 Desain Rancangan Aplikasi

#### 1. Rancang Halaman *Login (User)*

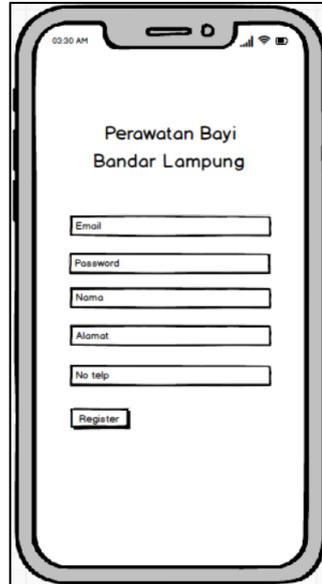
Rancangan halam ini dapat dilihat pada Gambar 3.22.



Gambar 3.22 Rancangan *Login*.

## 2. Rancangan Halaman Registrasi (*User*)

Rancangan halaman ini dapat dilihat pada Gambar 3.23.



The image shows a mobile application registration screen. At the top, the status bar displays '03:30 AM' and signal strength icons. The main heading is 'Perawatan Bayi' followed by 'Bandar Lampung'. Below the heading are five input fields: 'Email', 'Password', 'Nama', 'Alamat', and 'No telp'. At the bottom of the form is a 'Register' button.

Gambar 3.23 Rancangan Halaman Registrasi.

## 3. Rancangan Halaman *Home* (*User*)

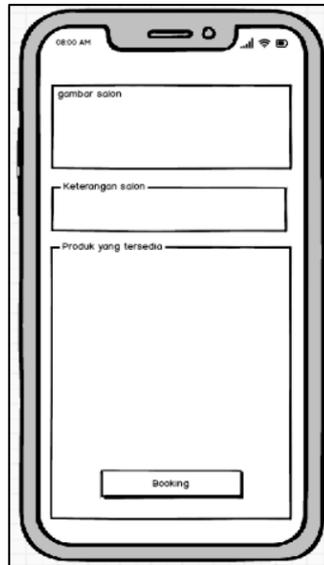
Rancangan halaman ini dapat dilihat pada Gambar 3.24.



Gambar 3.24 Rancangan Halaman *Home*.

#### 4. Rancangan Halaman *Detail* Perawatan Bayi Aplikasi (*User*)

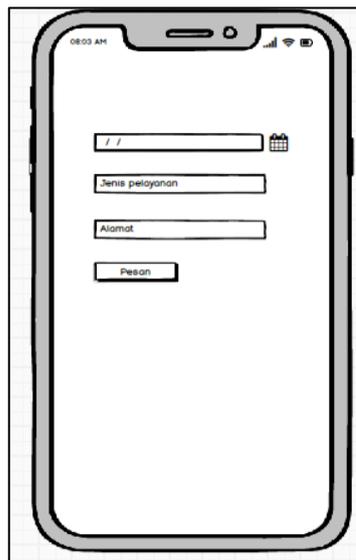
Rancangan halaman *Detail* Perawatan Bayi aplikasi dapat dilihat pada Gambar 3.25



Gambar 3.25 Rancangan Halaman *Detail* Perawatan bayi.

#### 5. Rancangan Halaman Pemesanan atau *Booking* (*User*)

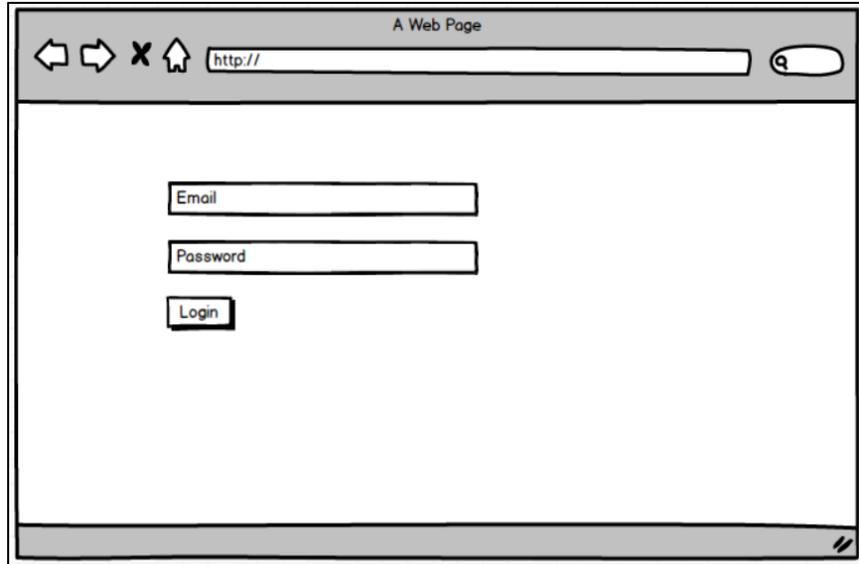
Rancangan halaman ini dapat dilihat pada Gambar 3.26.



Gambar 3.26 Rancangan Halaman *Booking* Perawatan bayi.

## 6. Rancangan Halaman *Login (Admin dan Owner)*

Rancangan halaman ini dapat dilihat pada Gambar 3.27.

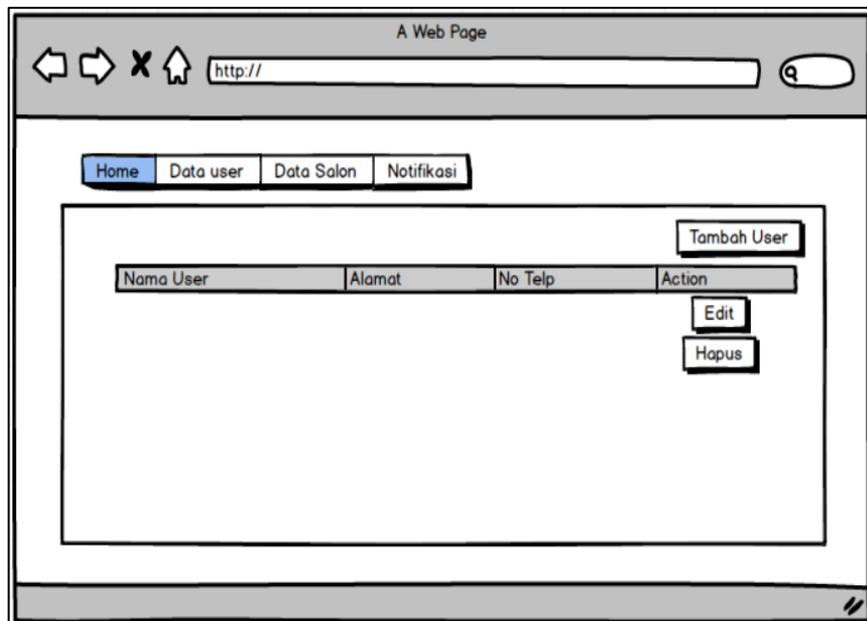


The image shows a browser window titled "A Web Page" with a search bar containing "http://". The main content area contains a login form with three input fields: "Email", "Password", and a "Login" button.

Gambar 3.27 Rancangan Halaman *Login*.

## 7. Rancangan Halaman *Data User (Admin)*

Rancangan halaman ini dapat dilihat pada Gambar 3.28.

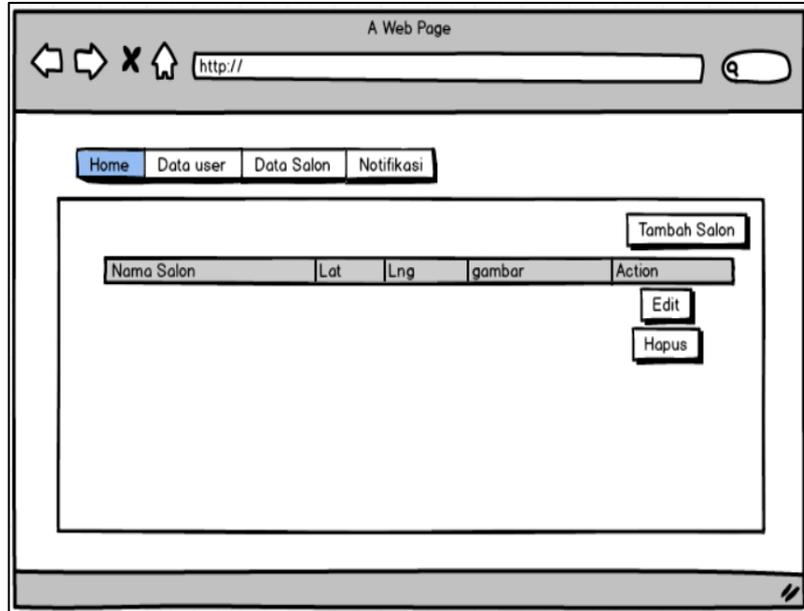


The image shows a browser window titled "A Web Page" with a search bar containing "http://". The main content area features a navigation menu with "Home", "Data user", "Data Salon", and "Notifikasi". Below the menu is a table with columns "Nama User", "Alamat", "No Telp", and "Action". The "Action" column contains "Tambah User", "Edit", and "Hapus" buttons.

Gambar 3.28 Rancangan Halaman *Data User*.

## 8. Rancangan Halaman Data Perawatan bayi (*Admin*)

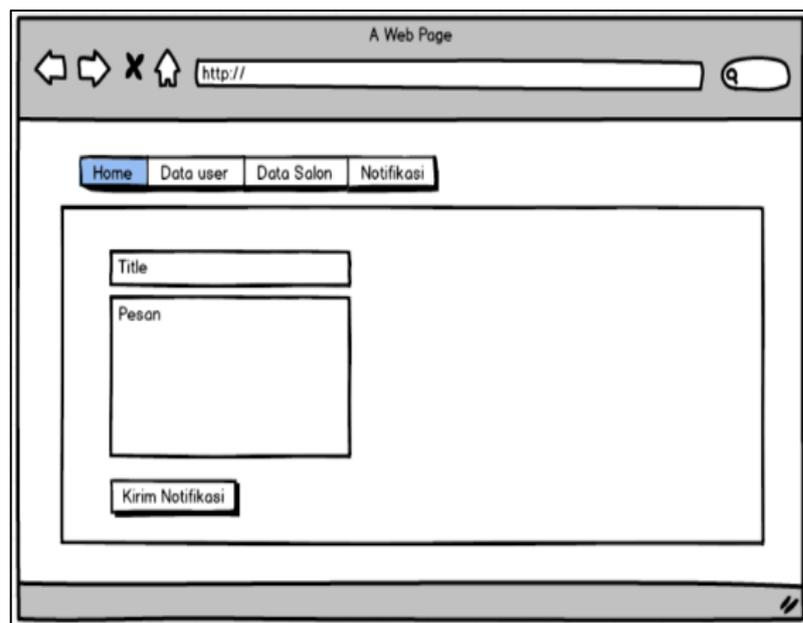
Rancangan halaman ini dapat dilihat pada Gambar 3.29.



Gambar 3.29 Rancangan Halaman Data Perawatan bayi.

## 9. Rancangan Halaman Notifikasi (*Admin*)

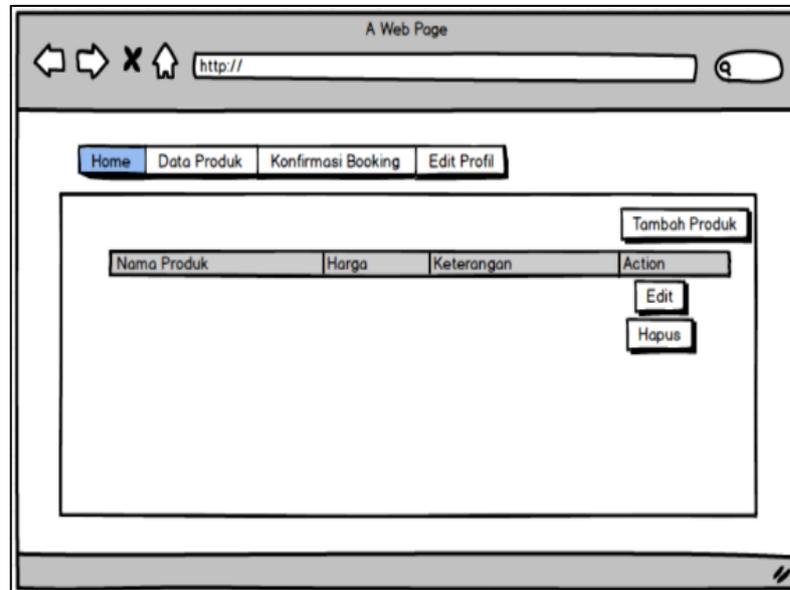
Rancangan halaman ini dapat dilihat pada Gambar 3.30.



Gambar 3.30 Rancangan Halaman Notifikasi.

10. Rancangan Halaman Data Produk (*Owner*)

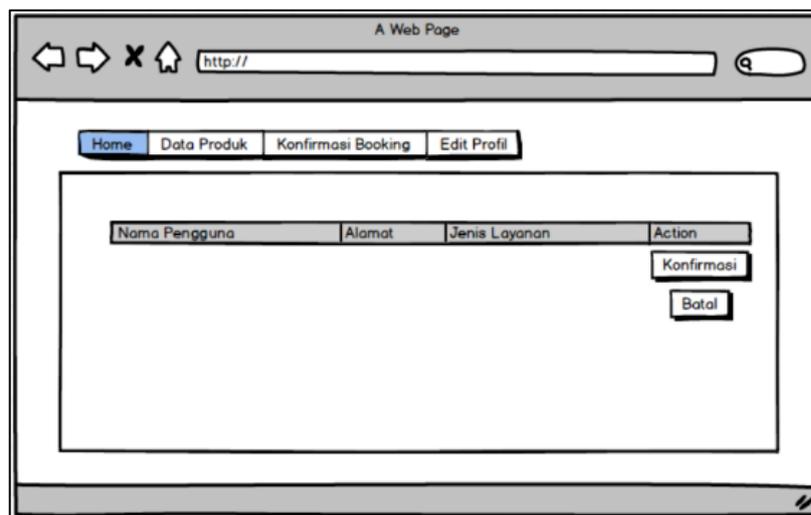
Rancangan halaman ini dapat dilihat pada Gambar 3.31.



Gambar 3.31 Rancangan Halaman Data produk.

11. Rancangan Halaman Konfirmasi *Booking* (*Owner*)

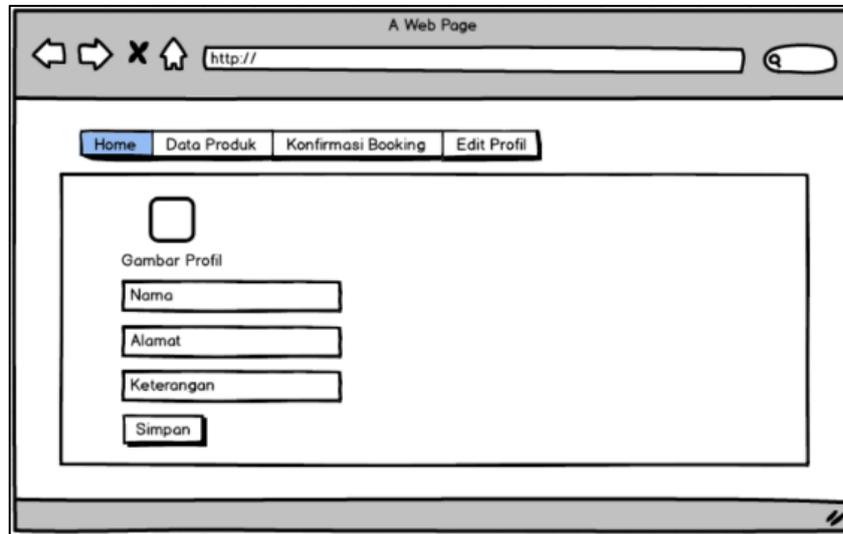
Rancangan halaman ini dapat dilihat pada Gambar 3.32.



Gambar 3.32 Rancangan Halaman Konfirmasi *Booking*.

## 12. Rancangan Halaman Edit Profil Perawatan bayi (*Owner*)

Rancangan halaman ini dapat dilihat pada Gambar 3.33.



Gambar 3.33 Rancangan Halaman Edit Profil perawatan bayi.

### 3.4 Evaluasi *Prototype*

Merupakan tahap dimana rancangan aplikasi yang telah selesai dibangun dilakukan evaluasi oleh *user*, dimana berkaitan dengan penelitian ini *user* atau pengguna dari aplikasi adalah *owner* sebagai pengelola perawatan bayi dan *customer* sebagai pengguna jasa perawatan bayi di Perawatan Bayi Bandar Lampung. Tahapan pengevaluasian ini dilakukan untuk memperjelas spesifikasi kebutuhan *customer* dan *owner* terhadap aplikasi sesuai yang diinginkan.

### 3.5 Proses Kerja Aplikasi Perawatan Bayi

Aplikasi perawatan bayi merupakan aplikasi *android mobile* yang dapat digunakan pelanggan atau pun masyarakat yang ingin melakukan pemesanan atau *booking* perawatan bayi di bandar lampung. Pengguna dapat mengunduh dan menginstall aplikasi tersebut dari Google Playstore, kemudian pengguna langsung bisa melakukan booking perawatan bayi sesuai dengan jarak terdekat dari pengguna yang tersedia pada aplikasi tersebut dengan syarat, pertama telah melakukan proses registrasi untuk mendapatkan *username* dan *password* agar bisa *login* pada aplikasi,

yang kedua pelanggan sudah memiliki perawatan bayi dan jenis jasa terlebih dahulu. Pengguna yang ingin *booking* perawatan bayi, bisa langsung mengklik list perawatan bayi terdekat untuk memilih perawatan bayi mana yang ingin *booking*, selanjutnya pelanggan diminta mengisi *form booking* Setelah melakukan *booking*, pemesan harus melakukan menunggu konfirmasi dari pihak perawatan bayi pada aplikasi.