

DPPL-0002

DESKRIPSI PERANCANGAN PERANGKAT LUNAK

HEWAN KITA - HETA

untuk:

Dosen


Dipersiapkan oleh:

Alif Farhan Widyanto	1301184289	Project Leader
Amal Khairin	1301180248	Back-End
Risma Julia Wardani	1301184094	Front-End
Heri Setyo Nugoho	1301184382	Back-End
Wardah Hanifah	1301184257	UI/UX

Program Studi Informatika

Fakultas Informatika

Jl. Telekomunikasi 1, Dayeuhkolot Bandung

	Prodi S1- Informatika Universitas Telkom	Nomor Dokumen		Halaman
		<i>DPPL-0002</i>		<i>1/57</i>
		Revisi	-	<i>14 April 2020</i>

DAFTAR PERUBAHAN

Revisi	Deskripsi
A	
B	
C	
D	
E	
F	
G	

INDEX TGL	-	A	B	C	D	E	F	G
Ditulis oleh								
Diperiksa oleh								
Disetujui oleh								

Daftar Halaman Perubahan

Halaman	Revisi	Halaman	Revisi

DAFTAR ISI

Table of Contents

DAFTAR PERUBAHAN	2
Daftar Halaman Perubahan	3
DAFTAR ISI	4
DAFTAR TABLE	6
DAFTAR GAMBAR	7
1. Pendahuluan	8
1.1 Tujuan Penulisan Dokumen	8
1.2 Lingkup Masalah	8
1.3 Definisi dan Istilah	8
1.4 Referensi	8
1.5 Ikhtisar Dokumen	9
2 Deskripsi Perancangan Global	9
2.1 Rancangan Lingkungan Implementasi	9
2.2 Deskripsi Arsitektural	10
2.3 Deskripsi Komponen	10
3 Perancangan Rinci	11
3.1 Realisasi Use Case	11
3.1.1 Use Case <Registrasi Akun>	12
3.1.2 Use Case <Login Akun>	14
3.1.3 Use Case <Konsultasi Dokter>	16
3.1.4 Use Case <Chat Dokter & Pemilik Hewan>	18
3.1.5 Use Case <Membayar Tarif Konsultasi>	21
3.1.6 Use Case <Reservasi Dokter>	23
3.1.7 Use Case <Konfirmasi Reservasi>	26
3.2 Perancangan Detil Kelas	31
3.2.1 Kelas <Dokter (extends dari user)>	31
3.2.2 Kelas <Pemilik Hewan>	32
3.2.3 Kelas <Database>	32
3.2.4 Kelas <Klinik>	33
3.2.5 Kelas <C_Registrasi >	34
3.2.6 Kelas <V_Registrasi >	34
3.2.7 Kelas <C_Login >	34
3.2.8 Kelas <V_Login >	35
3.2.9 Kelas <C_Konsultasi >	35
3.2.10 Kelas <V_Konsultasi >	35
3.2.11 Kelas <C_Pembayaran>	35
3.2.12 Kelas <V_Pembayaran>	36
3.2.13 Kelas <C_Reservasi>	36
3.2.14 Kelas <V_Reservasi>	36
3.2.15 Kelas <C_Chat>	37

3.2.16	Kelas <V_Chat>	37
3.2.17	Kelas <C_detailReservasi>	38
3.2.18	Kelas <V_detailReservasi>	38
3.2.19	Kelas <C_history>	38
3.2.20	Kelas <V_history>	39
3.2.21	Kelas <User>	39
3.3	Diagram Kelas Keseluruhan	40
3.4	Algoritma/Query	41
3.5	Diagram Statechart	45
3.6	Perancangan Antarmuka	46
3.7	Perancangan Representasi Persistensi Kelas	55
4	Matriks Kerunutan	56

DAFTAR TABEL

Tabel 1 Component Diagram Tabel	10
Tabel 2 Use Case Scenario Registrasi	12
Tabel 3 Identifikasi Kelas Registrasi	12
Tabel 4 Use Case Scenario Login	14
Tabel 5 Identifikasi Kelas Login	15
Tabel 6 Use Case Scenario Konsultasi Dokter	16
Tabel 7 Identifikasi Kelas Konsultasi Dokter	17
Tabel 8 Use Case Scenario Chat	18
Tabel 9 Identifikasi Kelas Chat	19
Tabel 10 Use Case Scenario Membayar Tarif	21
Tabel 11 Identifikasi Kelas Membayar Tarif	22
Tabel 12 Use Case Scenario Reservasi	23
Tabel 13 Identifikasi Kelas Reservasi	24
Tabel 14 Use Case Scenario Konfirmasi Reservasi	26
Tabel 15 Identifikasi Kelas Konfirmasi Reservasi	27
Tabel 16 Use Case Scenario Melihat History	28
Tabel 17 Identifikasi Kelas Melihat History	29
Tabel 18 Perancangan Detail Kelas Dokter	31
Tabel 19 Perancangan Detail Kelas Pemilik Hewan	32
Tabel 20 Perancangan Detail Kelas Database	32
Tabel 21 Perancangan Detail Kelas Klinik	33
Tabel 22 Perancangan Detail Kelas C_Registrasi	34
Tabel 23 Perancangan Detail Kelas V_Registrasi	34
Tabel 24 C_Login Perancangan Detail Kelas	34
Tabel 25 Perancangan Detail Kelas V_Login	35
Tabel 26 Perancangan Detail Kelas C_Konsultasi	35
Tabel 27 Perancangan Detail Kelas V_Konsultasi	35
Tabel 28 Perancangan Detail Kelas V_Pembayaran	36
Tabel 29 Perancangan Detail Kelas C_Reservasi	36
Tabel 30 Perancangan Detail Kelas V_Reservasi	36
Tabel 31 Perancangan Detail Kelas C_Chat	37
Tabel 32 Perancangan Detail Kelas V_Chat	37
Tabel 33 Perancangan Detail Kelas C_detailReservasi	38
Tabel 34 Perancangan Detail Kelas V_detailReservasi	38
Tabel 35 Perancangan Detail Kelas C_history	38
Tabel 36 Perancangan Detail Kelas V_history	39
Tabel 37 Perancangan Detail Kelas User	39
Tabel 38 Algoritma Register, getResepDokter, getAllHistoryByStatus	41
Tabel 39 Query Register, getResepDokter, getAllHistoryByStatus	41
Tabel 40 Algoritma getDokterAll(), getKlinikAll(), getAllHistoryFilter(filter)	42
Tabel 41 Query getDokterAll(), getKlinikAll(), getAllHistoryFilter(filter)	42
Tabel 42 Algoritma getAllDokterByID(), getAllKlinikByID(), getAllHistory()	43
Tabel 43 Query getAllDokterByID(), getAllKlinikByID(), getAllHistory()	43
Tabel 44 Algoritma Login, getMetodePembayaran, insertResevasi	43
Tabel 45 Query Login, getMetodePembayaran, insertResevasi	44
Tabel 46 Antarmuka Login	46
Tabel 47 Antarmuka Registrasi	47
Tabel 48 Antarmuka Home	48
Tabel 49 Antarmuka Halaman Konsultasi	49
Tabel 50 Antarmuka Chat room	50
Tabel 51 Antarmuka Pembayaran	51
Tabel 52 Antarmuka Reservasi	52
Tabel 53 Antarmuka Konfirmasi Reservasi	53
Tabel 54 Antarmuka Riwayat Transaksi (History)	54
Tabel 55 Matriks Keruntutan	56

DAFTAR GAMBAR

Gambar 1 Component Diagram	10
Gambar 2 Use Case Diagram	11
Gambar 3 Sequence Diagram Registrasi	13
Gambar 4 Diagram Kelas Registrasi	14
Gambar 5 Sequence Diagram Login	15
Gambar 6 Diagram Kelas Login	16
Gambar 7 Sequence Diagram Konsultasi	18
Gambar 8 Diagram Kelas Konsultasi	18
Gambar 9 Sequence Diagram Chat	20
Gambar 10 Diagram Kelas Chat	20
Gambar 11 Sequence Diagram Pembayaran	22
Gambar 12 Diagram Kelas Pembayaran	23
Gambar 13 Sequence Diagram Reservasi	25
Gambar 14 Diagram Kelas Reservasi	25
Gambar 15 Sequence Diagram Konfirmasi Reservasi	27
Gambar 16 Diagram Kelas Konfirmasi Reservasi	28
Gambar 17 Sequence Diagram Melihat History	30
Gambar 18 Diagram Kelas Melihat History	30
Gambar 19 Class Diagram Keseluruhan	40
Gambar 20 Diagram State Chart	45
Gambar 21 Wireframe Login	46
Gambar 22 Wireframe Register	47
Gambar 23 Wireframe Home	48
Gambar 24 Wireframe Konsultasi	49
Gambar 25 Wireframe Chat Dokter & Pemilik_hewan	50
Gambar 26 Wireframe Pembayaran	51
Gambar 27 Wireframe Reservasi Dokter	52
Gambar 28 Wireframe Konfirmasi Reservasi	53
Gambar 29 Wireframe History	54
Gambar 30 Skema Relasi	55

1. Pendahuluan

1.1 Tujuan Penulisan Dokumen

Dokumen ini berisi penjelasan mengenai Deskripsi Perancangan Perangkat Lunak (DPPL) untuk Aplikasi Hewan Kita (HETA). Tujuan penulisan dokumen ini adalah untuk mendeskripsikan secara rinci mengenai perangkat lunak yang akan dibangun. Dokumen ini akan digunakan untuk acuan pada proses pembangunan perangkat lunak dan evaluasi diakhir pembangunan perangkat lunak. Penggunaan dari dokumen ini adalah tim pengembang dari perangkat lunak dan stakeholders yang terlibat dalam sistem ini. Dengan disusunnya dokumen DPPL ini, diharapkan pembangunan perangkat lunak akan lebih terkonsep dan tidak menimbulkan ambiguitas pada saat pembangunannya.

1.2 Lingkup Masalah

Perangkat lunak “HETA” yang akan dikembangkan dalam makalah ini membahas tentang Konsultasi dan Reservasi Dokter untuk mempermudah membuat konsultasi hewan dan membuat jadwal reservasi dokter

1.3 Definisi dan Istilah

DPPL : Dokumen yang mendeskripsikan dan menjabarkan secara terperinci mengenai perancangan perangkat lunak yang akan dibangun

SKPL : Singkatan dari spesifikasi kebutuhan perangkat lunak atau sering disebut juga sebagai Software Requirements Specification (SRS) merupakan spesifikasi dari perangkat lunak perangkat lunak yang dikembangkan

Dart : Dart adalah Bahasa pemrograman yang dioptimalkan klien untuk aplikasi pada berbagai platform. Ini dikembangkan oleh Google dan digunakan untuk membangun aplikasi mobile, desktop, server, dan web.

Javascript : JavaScript adalah bahasa pemrograman tingkat tinggi dan dinamis. JavaScript populer di internet dan dapat bekerja di sebagian besar penjelajah web populer seperti Google Chrome, Internet Explorer, Mozilla Firefox, Netscape dan Opera.

MySQL : MySQL adalah sebuah perangkat lunak sistem manajemen basis data SQL DBMS yang multialur, multipengguna, dengan sekitar 6 juta instalasi di seluruh dunia.

Visual Studio Code : Visual Studio Code adalah editor kode sumber yang dikembangkan oleh Microsoft untuk Windows , Linux dan macOS . Ini termasuk dukungan untuk debugging , kontrol Git yang tertanam dan GitHub , penyorotan sintaksis , penyelesaian kode cerdas , snippet , dan refactoring kode

1.4 Referensi

Kelompok 3 IF-39-03. (2017). DPPL Sistem Informasi Citra Mart.
Kelompok 2 IF-42-06. (2020). SKPL Hewan Kita – HETA
Kelompok 1 IF-41-XX. (2019). DPPL Sistem Informasi Apoteker
Flanagan, D. (2006). *JavaScript: the definitive guide*. " O'Reilly Media, Inc.".
Welling, L., & Thomson, L. (2003). *PHP and MySQL Web development*. Sams Publishing.
Dart (Programming Language). (2020). Retrieved 20 April, 2020 from [https://en.wikipedia.org/wiki/Dart_\(programming_language\)](https://en.wikipedia.org/wiki/Dart_(programming_language))
Visual Studio Code. (2020). Retrieved 20 April, 2020 from https://en.wikipedia.org/wiki/Visual_Studio_Code

1.5 Ikhtisar Dokumen

Dokumen DPPL ini berisikan deskripsi rancangan perangkat lunak sistem informasi perpustakaan yang akan dikembangkan berdasarkan dokumen SKPL. Pada dokumen DPPL ini akan dijelaskan rincian dari rancangan perangkat lunak sehingga dapat diimplementasikan. Dokumen ini secara garis besar terdiri dari empat bab dengan perincian sebagai berikut:

A. Pendahuluan

Pendahuluan berisi penjelasan tentang dokumen DPPL yang mencakup tujuan pembuatan dokumen ini, lingkup masalah yang diselesaikan oleh perangkat lunak yang dikembangkan, definisi, aturan penamaan dan penomoran, referensi, dan ikhtisar dokumen.

B. Deskripsi Perancangan Global

Deskripsi perancangan global berisi tentang rancangan dari perangkat lunak yang akan dibangun meliputi, rancangan lingkungan implementasi, deskripsi arsitektural, dan deskripsi komponen.

C. Perancangan Rinci

Perancangan rinci pada dokumen ini berisi tentang realisasi use case, perancangan detil kelas, deskripsi diagram kelas, algoritma/query, perancangan antarmuka, dan perancangan representasi persistensi kelas.

D. Matriks Keterunutan

Matriks keterunutan berisi tentang hal fungsional yang terdapat pada dokumen SKPL.

2 Deskripsi Perancangan Global

Deskripsi perancangan global meliputi penjelasan tentang rancangan lingkungan implementasi, deskripsi arsitektural, dan deskripsi komponen yang membangun perangkat lunak.

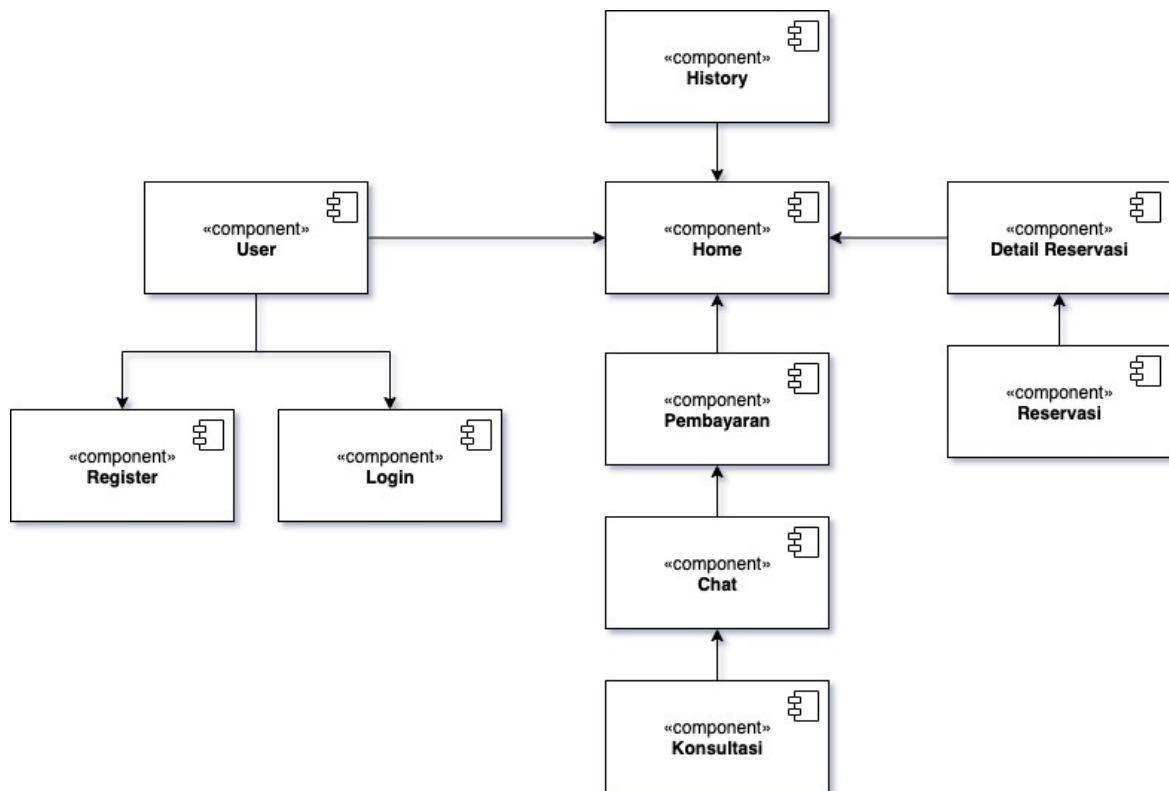
2.1 Rancangan Lingkungan Implementasi

Sistem ini diimplementasikan dalam lingkungan sebagai berikut :

- a. Sistem operasi : Windows
- b. Bahasa Pemrograman : Dart, Javascript
- c. DBMS : MySQL
- d. Development Tool: Visual Studio Code

2.2 Deskripsi Arsitektural

Merupakan gambaran arsitektur atau komponen yang akan diterapkan pada perangkat lunak “Hewan Kita - HETA” untuk mempermudah pengembang dalam mengembangkan atau mengimplementasikan perangkat lunak ini.



Gambar 1 Component Diagram

2.3 Deskripsi Komponen

Tabel 1 Component Diagram Tabel

No	Nama Komponen	Keterangan
1.	Home	Komponen yang berisi semua menu fungsi tampilan
2.	Login	Komponen yang berisi tampilan fungsi login
3.	Register	Komponen yang berisi tampilan fungsi register
4.	Reservasi	Komponen yang berisi tampilan fungsi reservasi
5.	Detail Reservasi	Komponen yang berisi tampilan fungsi detail reservasi
6.	Konsultasi	Komponen yang berisi tampilan fungsi konsultasi
7.	Chat	Komponen yang berisi tampilan fungsi chat
8.	Pembayaran	Komponen yang berisi tampilan fungsi pembayaran
9.	History	Komponen yang berisi tampilan fungsi history
10.	User	Komponen yang berisi tampilan fungsi user

3 Perancangan Rinci

Dalam bab perancangan rinci akan dijelaskan mengenai realisasi use case, perancangan detil kelas, diagram kelas keseluruhan, algoritma/query, diagram statechart, perancangan antarmuka, dan perancangan representasi persistensi kelas

3.1 Realisasi Use Case

Subbab ini menjelaskan tentang realisasi semua use case yang telah dirancang pada dokumen spesifikasi kebutuhan perangkat lunak.



Gambar 2 Use Case Diagram

3.1.1 Use Case <Registrasi Akun>

Tabel 2 Use Case Scenario Registrasi

Nama Use case :	Registrasi Akun
Aktor :	Pemilik_hewan, dan Dokter
Pre-condition :	Aktor belum terdaftar pada dalam system, aktor belum memiliki akun.
Post-condition :	Aktor telah terdaftar dalam system, dan aktor telah memiliki akun.
Deskripsi :	Aktor melakukan registrasi akun, dengan menginput data yang diperlukan dari system.
SKENARIO UTAMA	
Pemilik_hewan	System
1) Membuka halaman registrasi	
2) Memasukkan data	
	3) Memeriksa data yang telah dimasukkan oleh aktor
	4) Memverifikasi data yang telah dimasukkan oleh aktor
5) Menyelesaikan tahap registrasi	

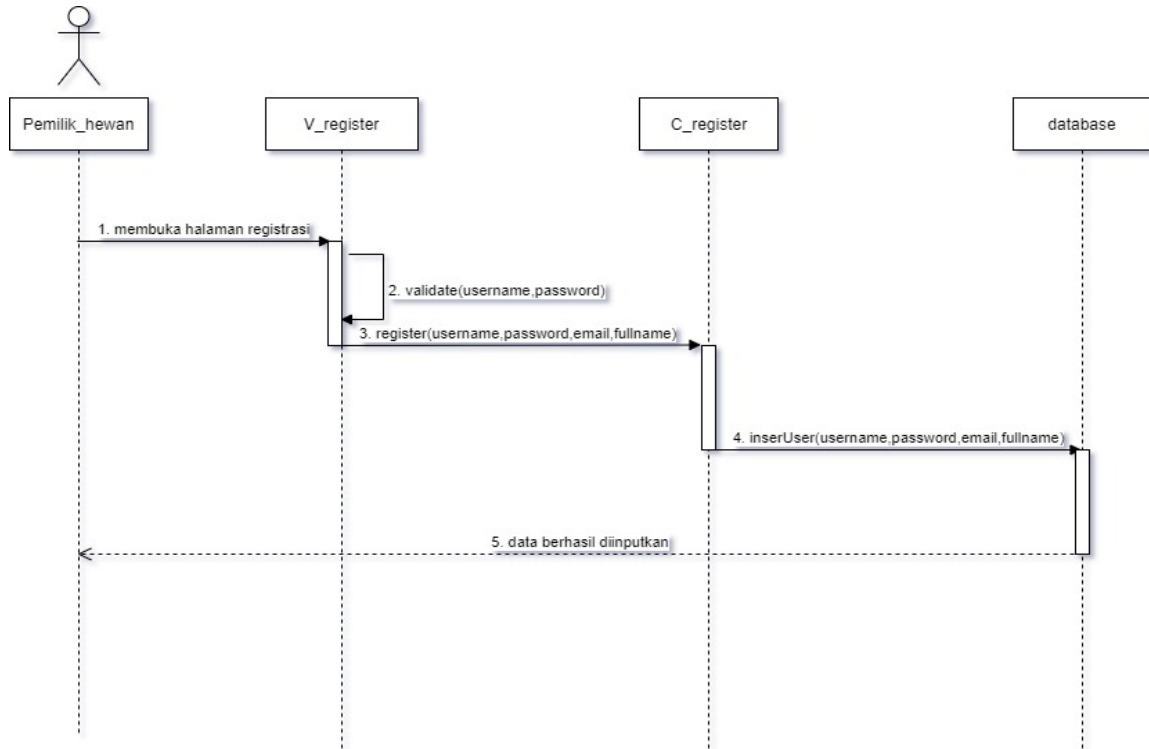
3.1.1.1 Identifikasi Kelas

Tabel 3 Identifikasi Kelas Registrasi

No	Nama Kelas Perancangan	Tipe Kelas
1.	User	User

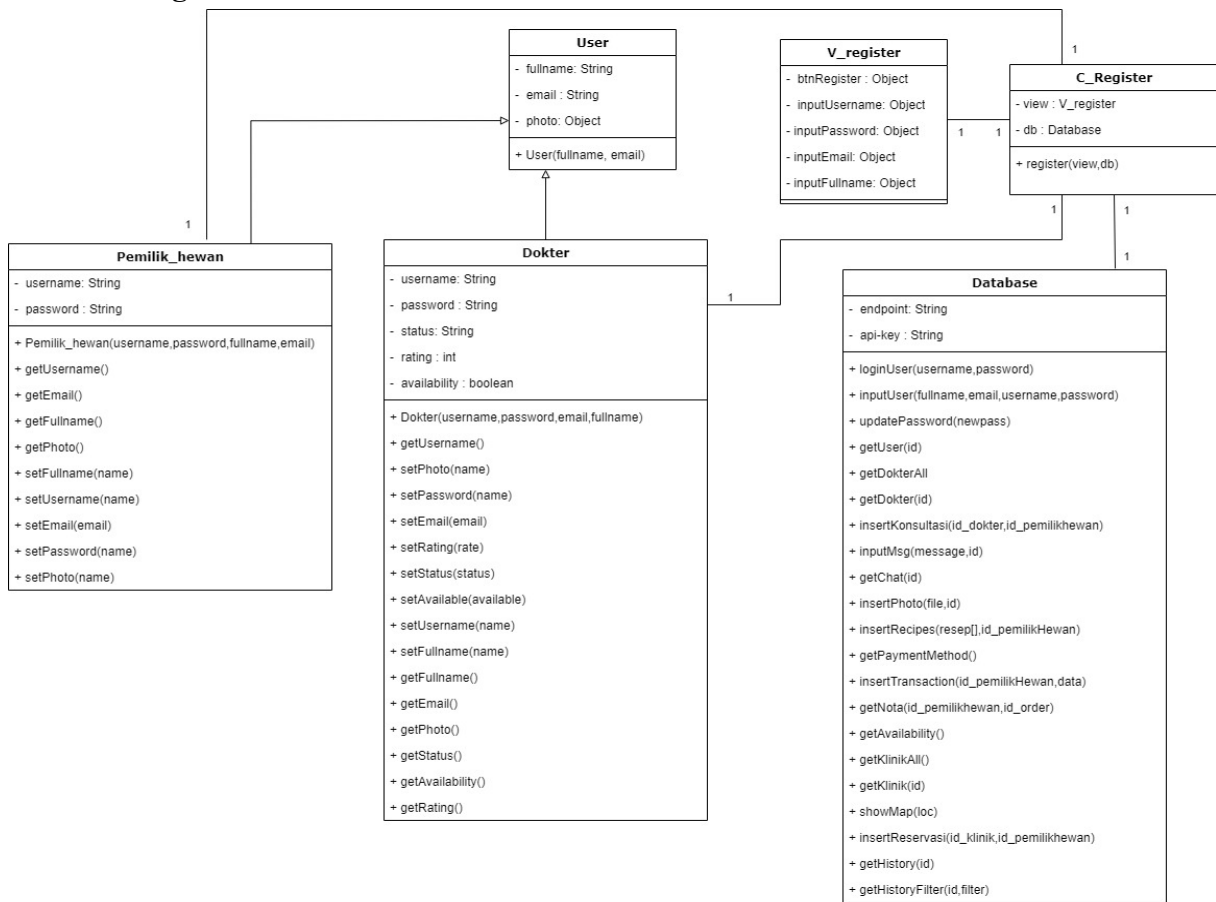
2.	<i>Pemilik hewan</i>	<i>Pemilik hewan</i>
3.	<i>Dokter hewan</i>	<i>Dokter hewan</i>
4.	<i>Database</i>	<i>Database</i>
5.	<i>V Registrasi</i>	<i>View</i>
6.	<i>C Registrasi</i>	<i>Controller</i>

3.1.1.2 Sequence Diagram



Gambar 3 Sequence Diagram Registrasi

3.1.1.3 Diagram Kelas



3.1.2 Use Case <Login Akun>

Gambar 4 Diagram Kelas Registrasi

Tabel 4 Use Case Scenario Login

Nama Use case :	Login Akun
Aktor :	Pemilik_hewan, dan Dokter
Pre-condition :	Aktor ingin melakukan login, dan belum login.
Post-condition :	Aktor telah masuk ke dalam akun masing-masing.
Deskripsi :	Aktor masuk ke dalam akun masing-masing untuk menggunakan fitur aplikasi.
SKENARIO UTAMA	
Pemilik_hewan	System
1) Membuka halaman login.	
2) Memasukkan data username dan password pada form yang telah disediakan.	
	3) Memverifikasi data yang dimasukkan oleh aktor.
	4) Jika data yang dimasukkan benar, maka lanjut ke tahap 5, jika belum benar, maka dialihkan ke tahap 2.
	5) Aktor dialihkan ke menu home.

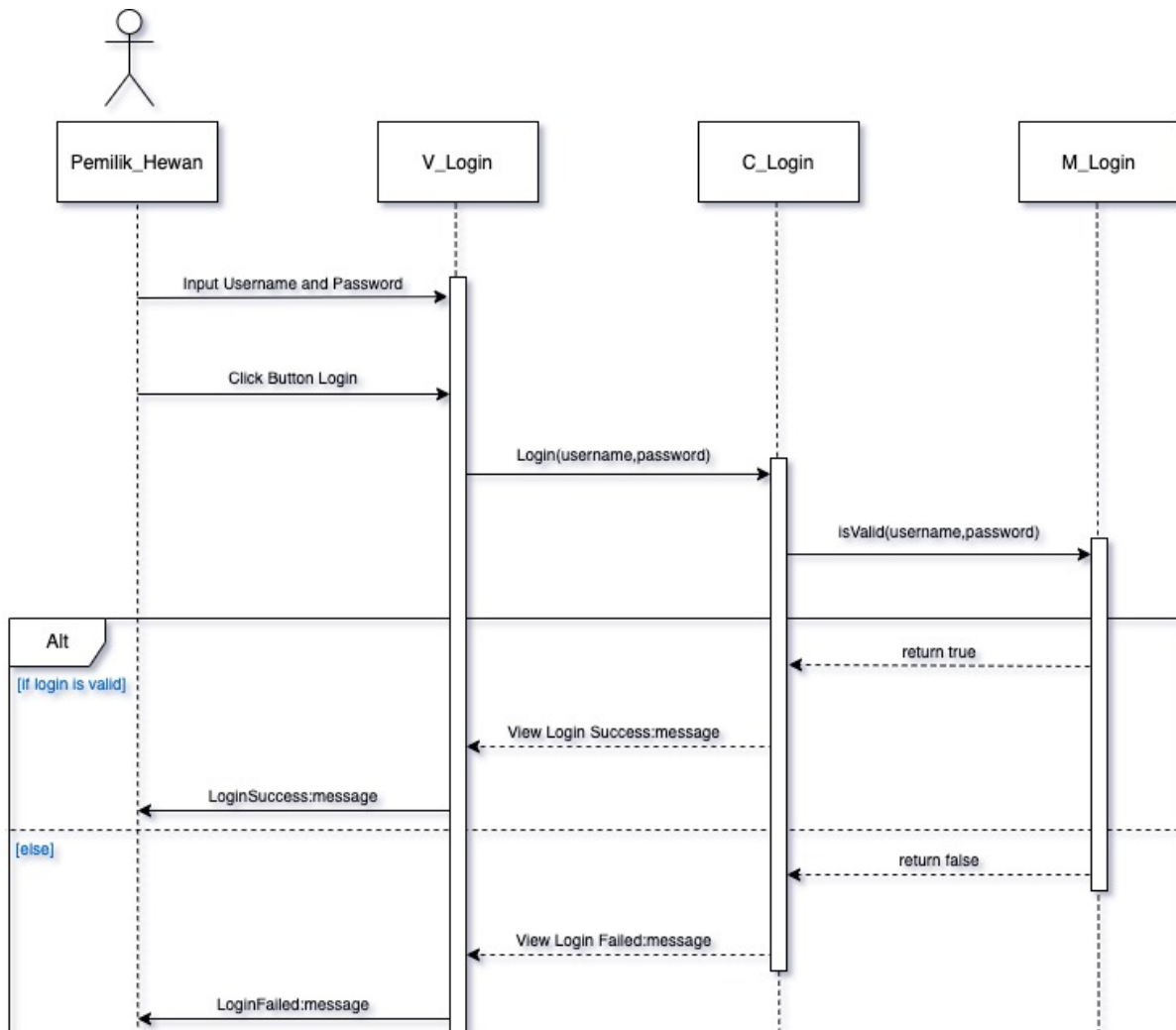
6) Aktor berhasil masuk kedalam akun.

3.1.2.1 Identifikasi Kelas

Tabel 5 Identifikasi Kelas Login

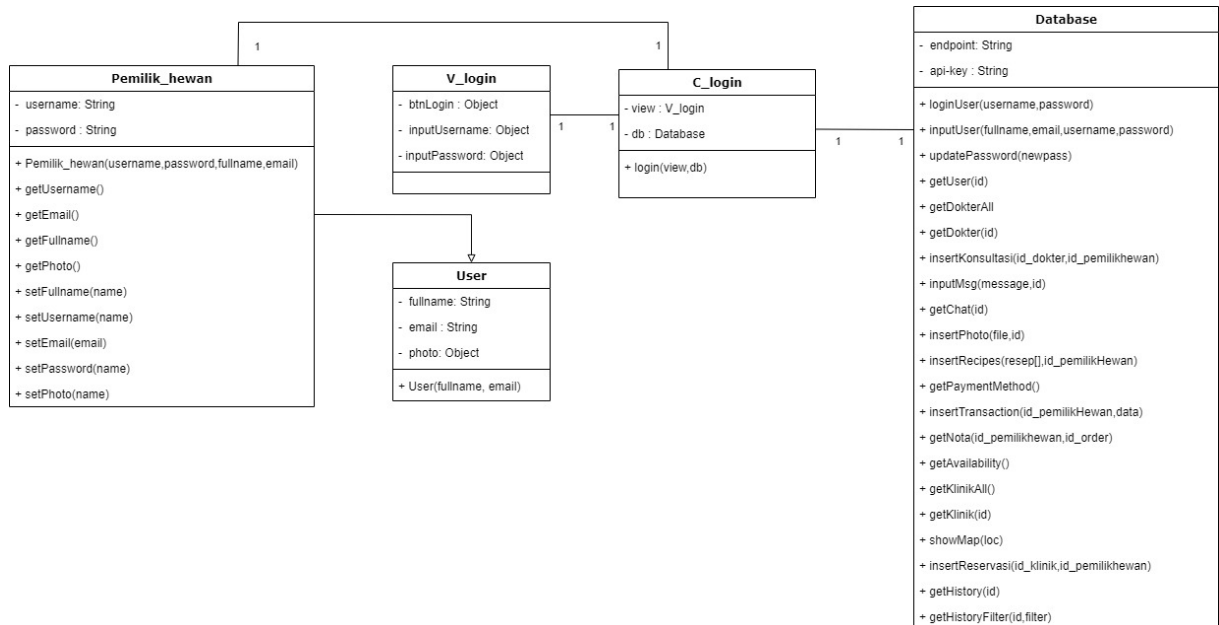
No	Nama Kelas Perancangan	Nama Kelas Analisis Terkait
1.	User	User
2.	Pemilik hewan	Pemilik hewan
3.	Dokter hewan	Dokter hewan
4.	Database	Database
5.	V_Login	View
6.	C_Login	Controller

3.1.2.2 Sequence Diagram



Gambar 5 Sequence Diagram Login

3.1.2.3 Diagram Kelas



Gambar 6 Diagram Kelas Login

3.1.3 Use Case <Konsultasi Dokter>

Tabel 6 Use Case Scenario Konsultasi Dokter

Nama Use case :	Konsultasi Dokter
Aktor :	Pemilik_hewan
Pre-condition :	Aktor ingin melakukan konsultasi dengan dokter, aktor belum melakukan konsultasi.
Post-condition :	Aktor telah melakukan konsultasi dengan dokter.

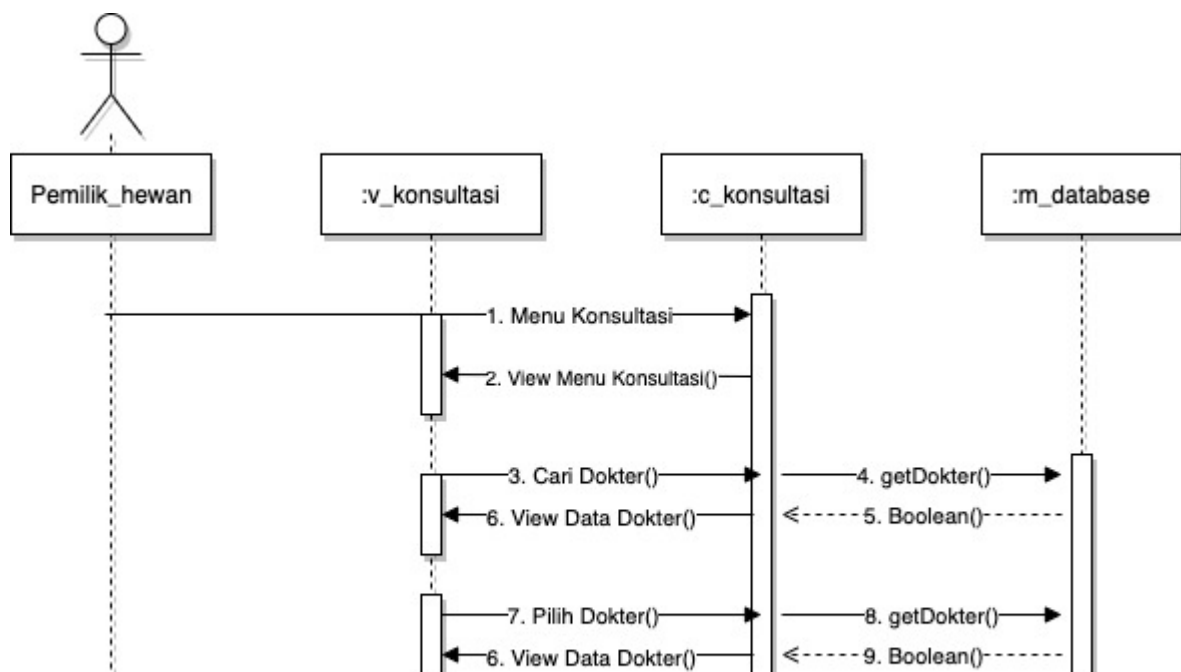
Deskripsi :	Aktor melakukan konsultasi dengan memulai percakapan dengan dokter.
SKENARIO UTAMA	
Pemilik_hewan	System
1) Memilih menu konsultasi	
	2) Menampilkan menu konsultasi
3) Mencari Dokter	
	4) Menampilkan dokter yang dicari
5) Memilih Dokter	
	6) Menerima permintaan aktor, dan mengalihkan aktor ke tahap percakapan.

3.1.3.1 Identifikasi Kelas

Tabel 7 Identifikasi Kelas Konsultasi Dokter

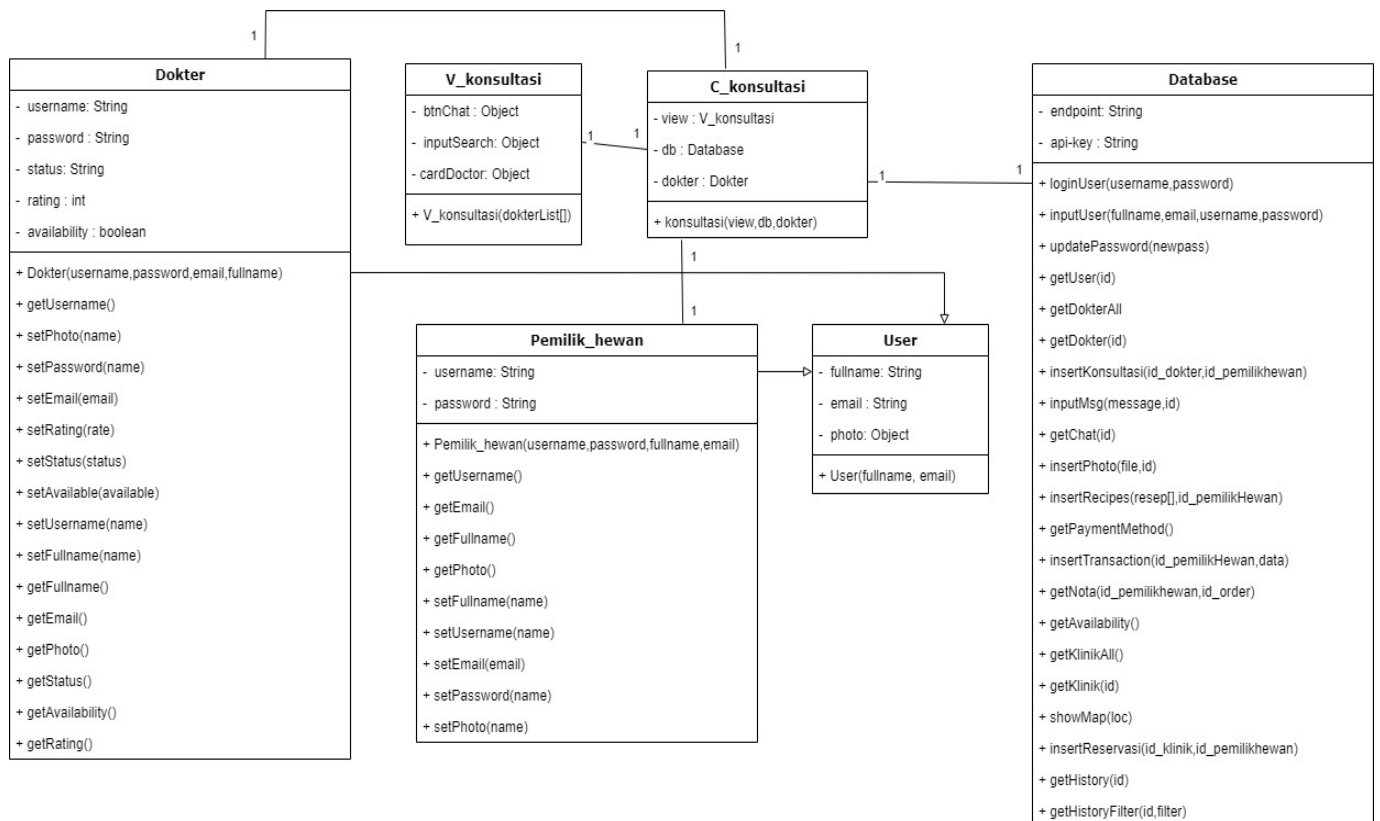
No	Nama Kelas Perancangan	Nama Kelas Analisis Terkait
1.	User	User
2.	Pemilik hewan	Pemilik hewan
3.	Dokter hewan	Dokter hewan
4.	Database	Database
5.	C Konsultasi	Controller
6.	V Konsultasi	View

3.1.3.2 Sequence Diagram



Gambar 7 Sequence Diagram Konsultasi

3.1.3.3 Diagram Kelas



Gambar 8 Diagram Kelas Konsultasi

3.1.4 Use Case <Chat Dokter & Pemilik Hewan>

Tabel 8 Use Case Scenario Chat

Nama Use case :	Chat Dokter & Pemilik_hewan
Aktor :	Pemilik_hewan, Dokter

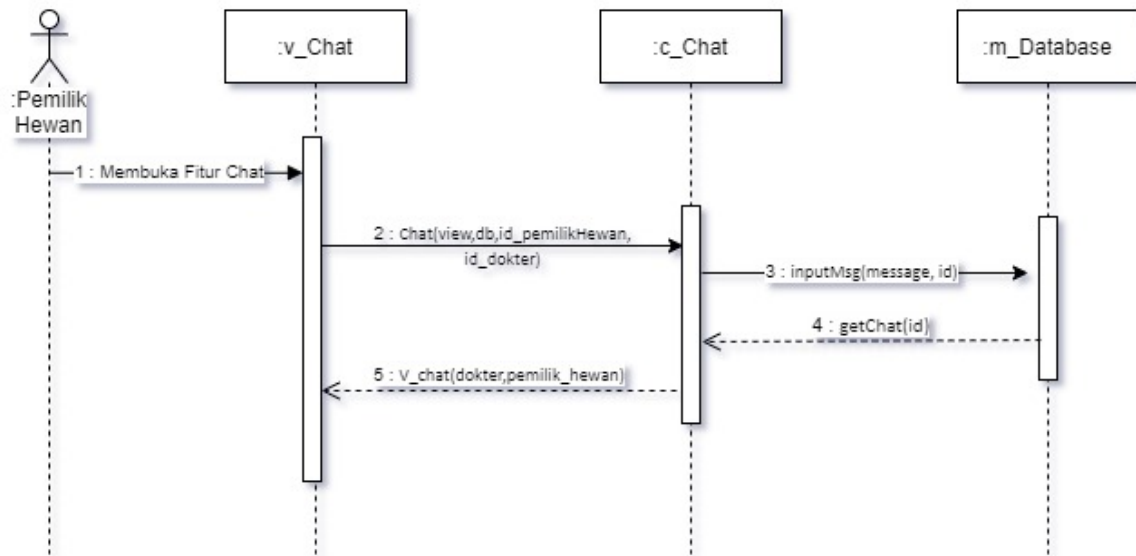
Pre-condition :	Aktor belum memulai percakapan apapun dengan dokter/pemilik_hewan
Post-condition :	Aktor memulai percakapan dengan dokter/pemilik_hewan, aktor
Deskripsi :	Aktor memulai percakapan dengan dokter/pemilik_hewan, aktor (pemilik_hewan) menyampaikan keluhan mengenai hewan peliharaan, serta meminta saran kepada dokter, aktor (Dokter) memberikan saran atau resep obat ke pemilik_hewan.
SKENARIO UTAMA	
Pemilik_hewan	System
1) Membuka fitur chat	
2) Mengirimkan pesan/foto kepada dokter	
3) Menyelesaikan percakapan	
	4) Mengalihkan aktor ke tahap selanjutnya

3.1.4.1 Identifikasi Kelas

Tabel 9 Identifikasi Kelas Chat

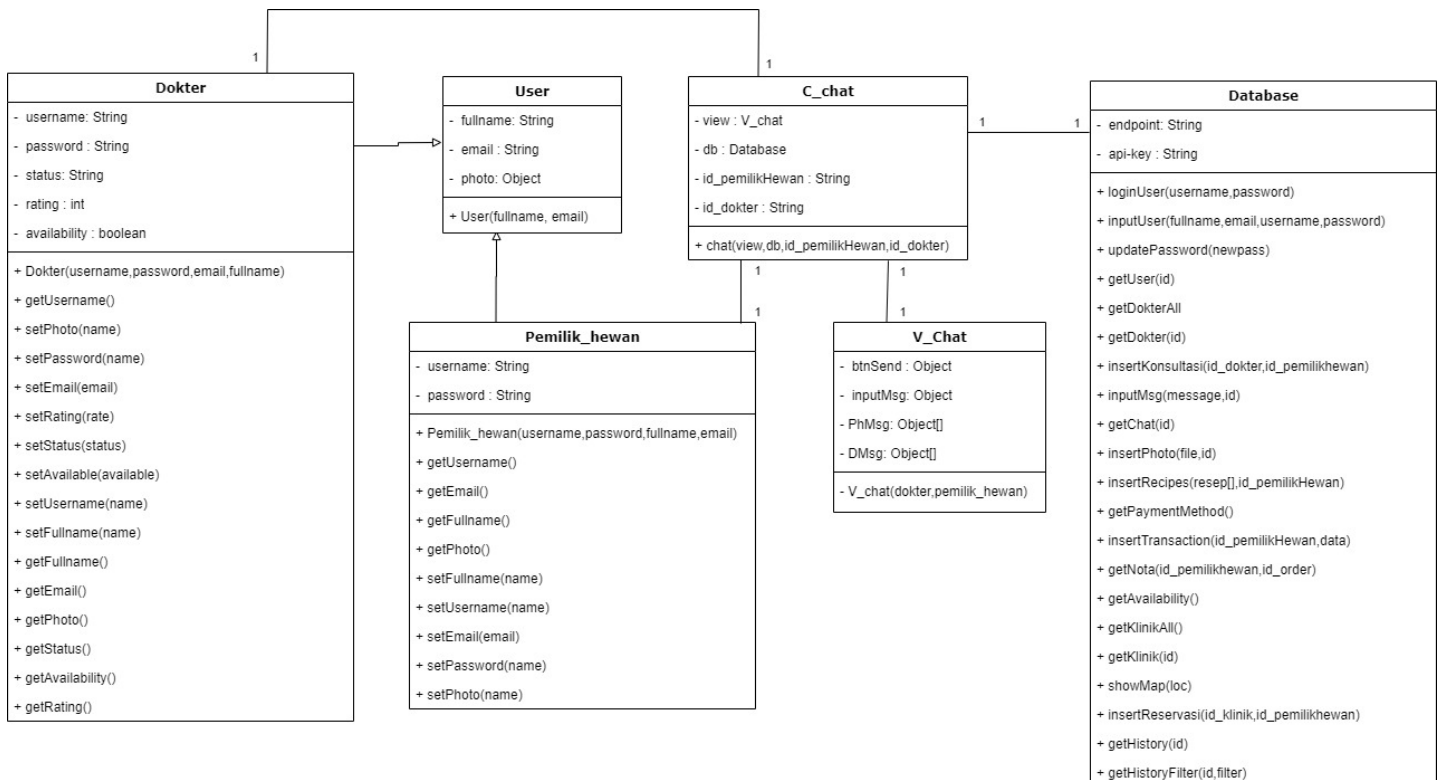
No	Nama Kelas Perancangan	Nama Kelas Analisis Terkait
1.	User	User
2.	Pemilik hewan	Pemilik hewan
3.	Dokter hewan	Dokter hewan
4.	Database	Database
5.	C Chat	Controller
6.	V Chat	View

3.1.4.2 Sequence Diagram



Gambar 9 Sequence Diagram Chat

3.1.4.3 Diagram Kelas



Gambar 10 Diagram Kelas Chat

3.1.5 Use Case <Membayar Tarif Konsultasi>

Tabel 10 Use Case Scenario Membayar Tarif

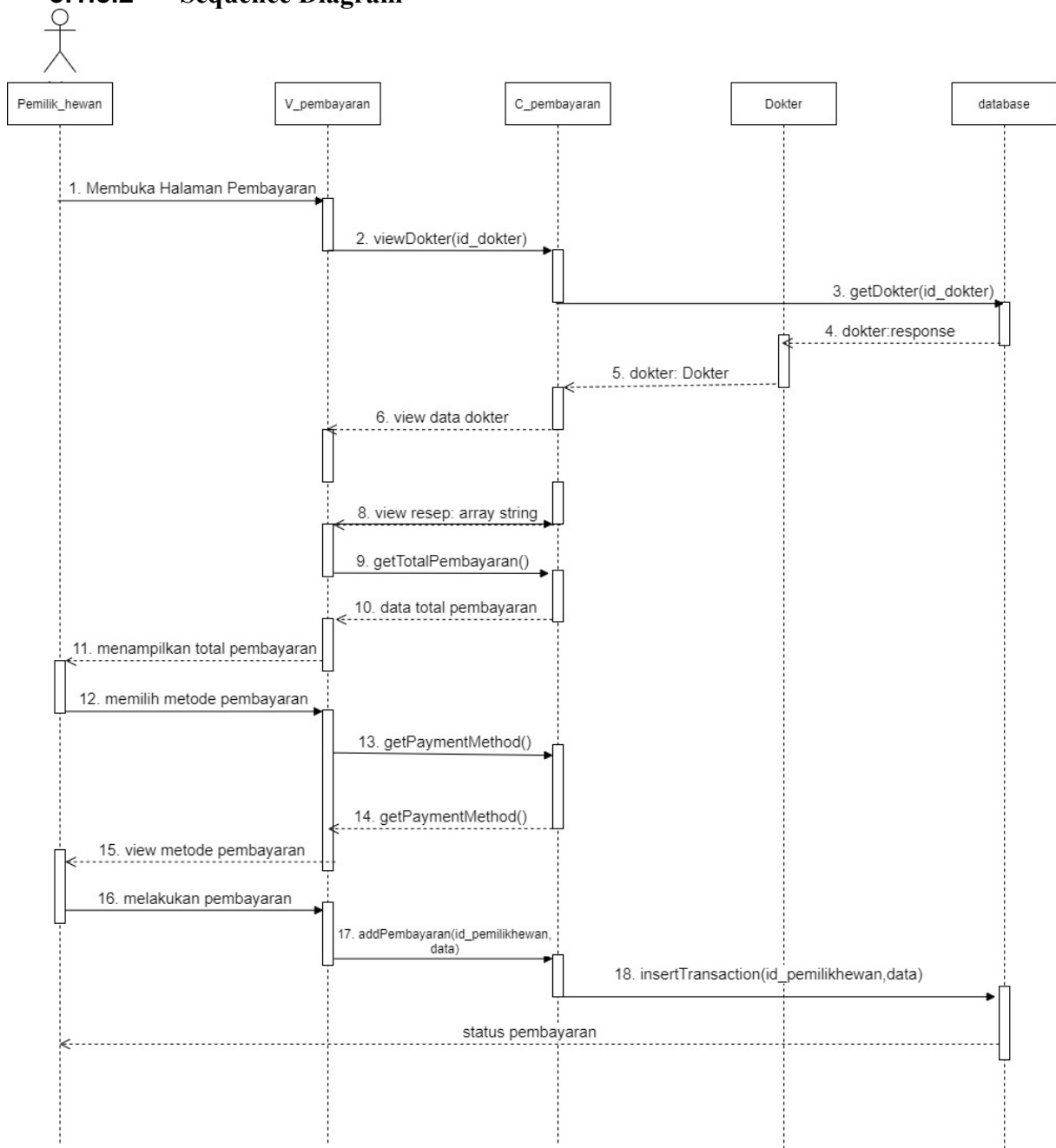
Nama Use case :	Membayar tarif konsultasi
Aktor :	Pemilik_hewan
Pre-condition :	Aktor belum melakukan pembayaran apapun.
Post-condition :	Aktor telah melakukan pembayaran
Deskripsi :	Aktor membayar tarif konsultasi yang sebelumnya telah dilakukan di sesi konsultasi bersama dokter
SKENARIO UTAMA	
Pemilik_hewan	System
1) Membuka halaman pembayaran	
	2) Menampilkan Info data Dokter
	3) Menampilkan list obat yang disarankan oleh dokter (jika ada)
	4) Mengakumulasikan total tarif dari sesi konsultasi antara kedua aktor
	5) Menampilkan total pembayaran
6) Melihat total tarif konsultasi yang perlu dibayar.	
7) Memilih metode pembayaran yang telah disediakan dari system.	
8) Melakukan pembayaran.	

3.1.5.1 Identifikasi Kelas

Tabel 11 Identifikasi Kelas Membayar Tarif

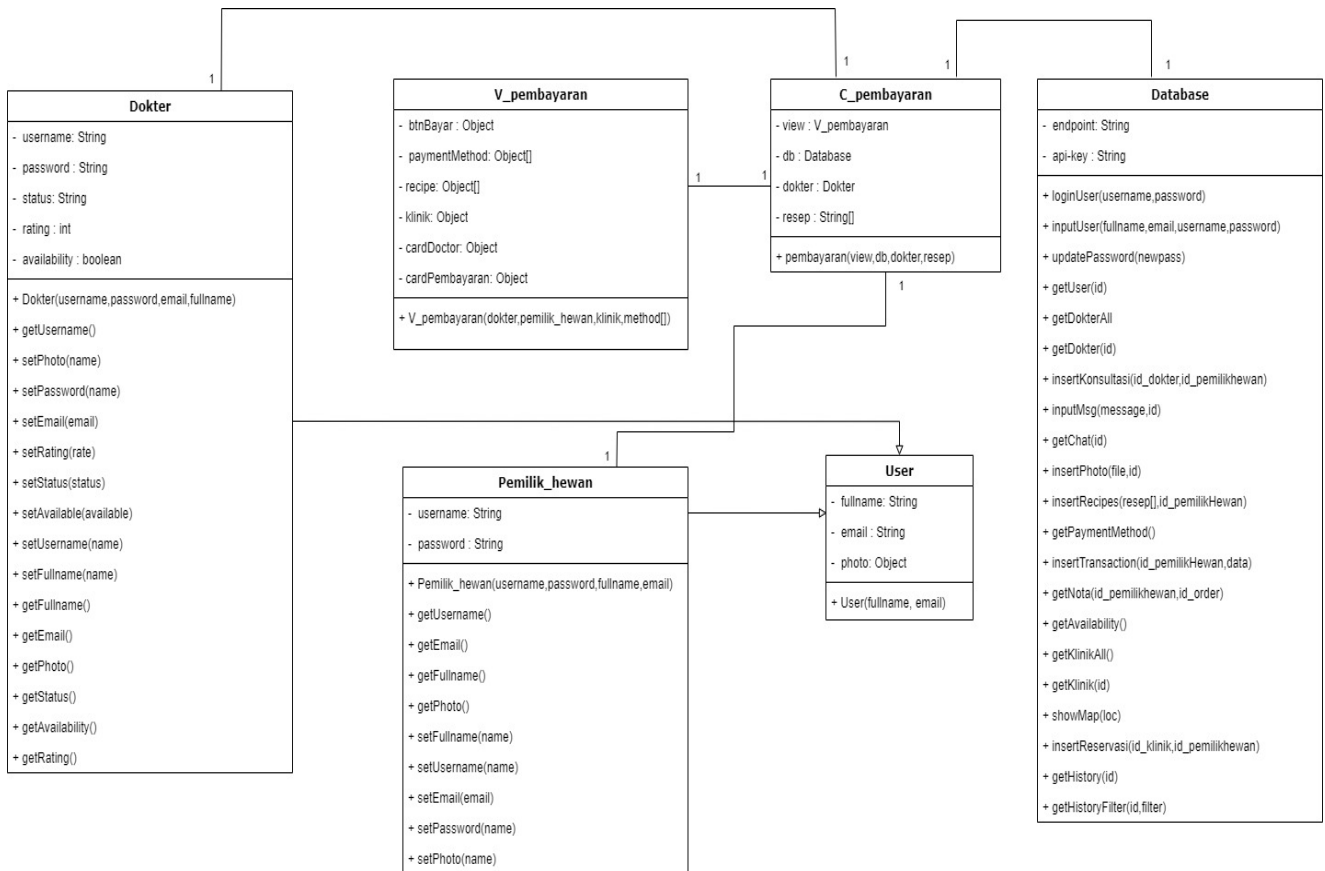
No	Nama Kelas Perancangan	Nama Kelas Analisis Terkait
7.	User	User
	Pemilik hewan	Pemilik hewan
	Dokter hewan	Dokter hewan
	Database	Model
	C Pembayaran	Controller
	V Pembayaran	View

3.1.5.2 Sequence Diagram



Gambar 11 Sequence Diagram Pembayaran

3.1.5.3 Diagram Kelas



Gambar 12 Diagram Kelas Pembayaran

3.1.6 Use Case <Reservasi Dokter>

Tabel 12 Use Case Scenario Reservasi

Nama Use case :	Reservasi Dokter
Aktor :	Pemilik_hewan
Pre-condition :	Aktor ingin melakukan reservasi dokter, aktor belum melakukan reservasi.
Post-condition :	Aktor telah melakukan reservasi.

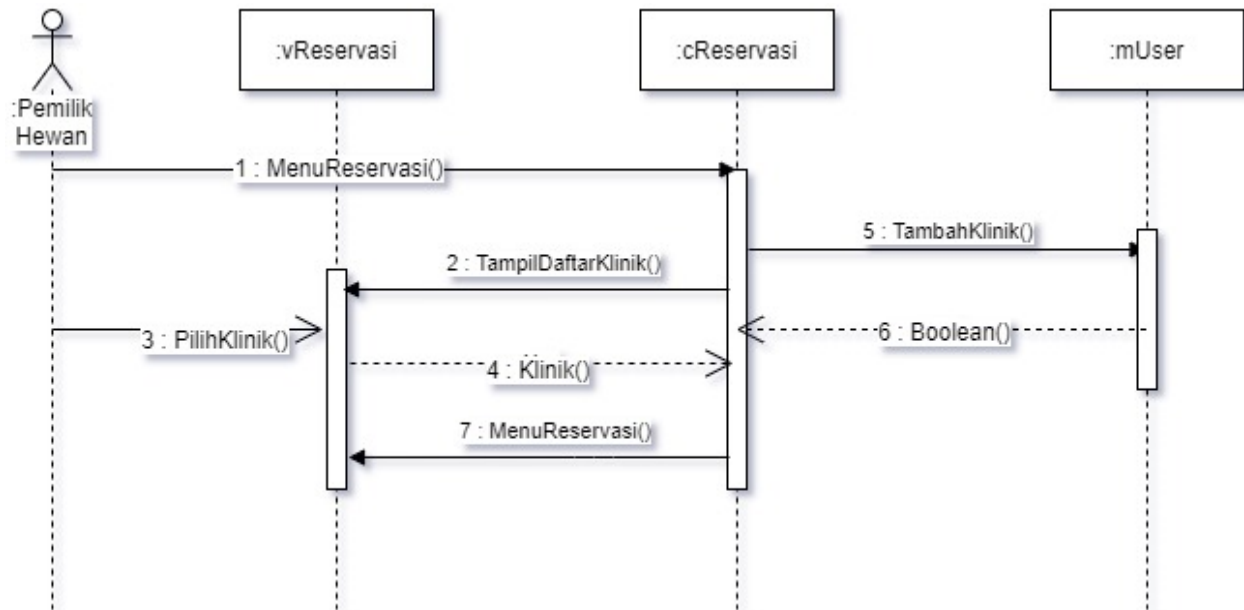
Deskripsi :	Aktor melakukan reservasi dokter, dengan memilih klinik yang disediakan oleh system, aktor dapat memfilter lokasi klinik.
SKENARIO UTAMA	
Pemilik_hewan	System
1) Memilih menu reservasi.	
	2) Menampilkan info klinik terdekat dari aktor.
3) Memfilter lokasi sesuai lokasi pemilik_hewan.	
4) Memilih klinik yang diinginkan.	
	5) Mengalihkan aktor ke halaman konfirmasi reservasi.

3.1.6.1 Identifikasi Kelas

Tabel 13 Identifikasi Kelas Reservasi

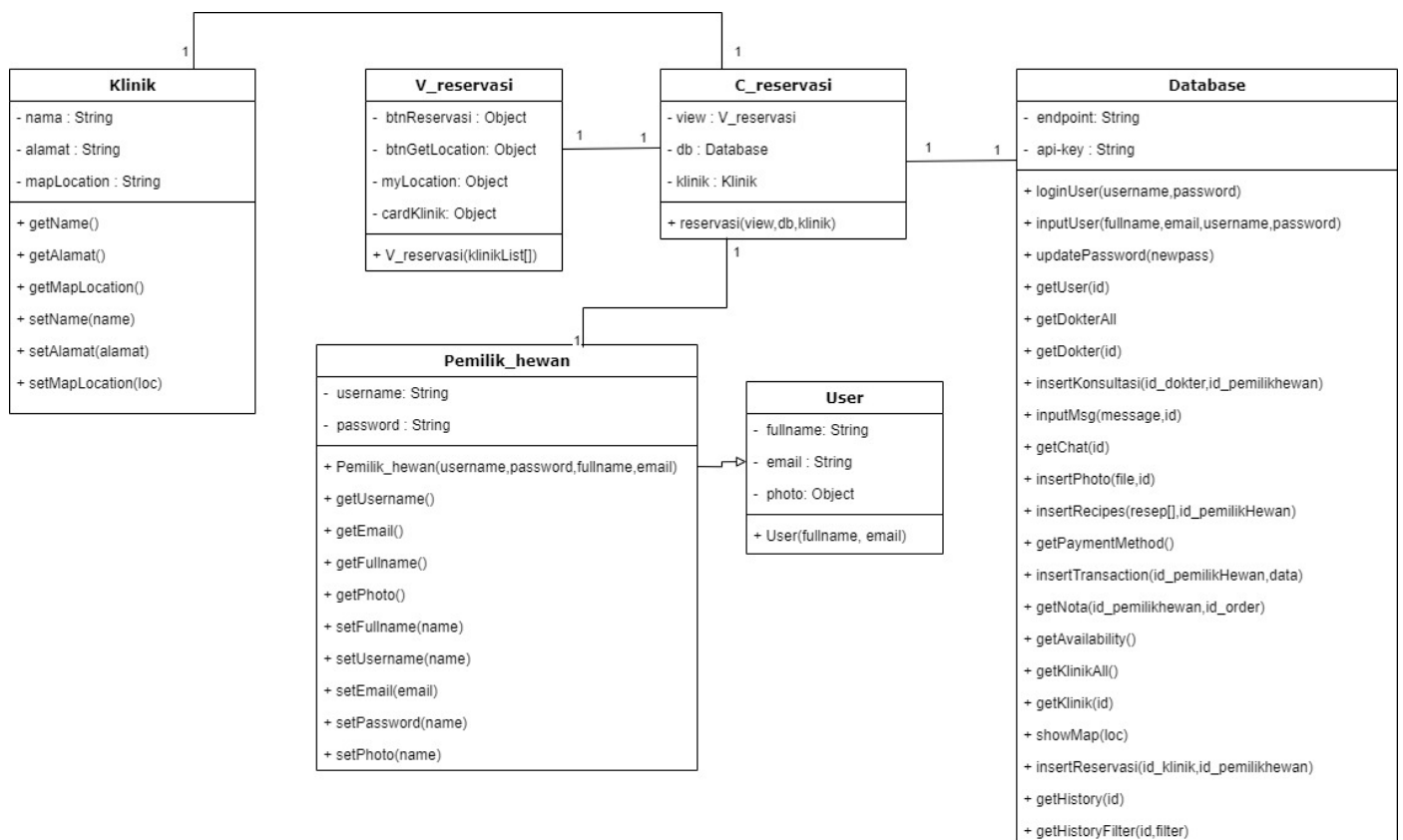
No	Nama Kelas Perancangan	Nama Kelas Analisis Terkait
8.	User	Model
	Pemilik hewan	Pemilik hewan
	V Reservasi	View
	C Reservasi	Controller

3.1.6.2 Sequence Diagram



Gambar 13 Sequence Diagram Reservasi

3.1.6.3 Diagram Kelas



Gambar 14 Diagram Kelas Reservasi

3.1.7 Use Case <Konfirmasi Reservasi>

Tabel 14 Use Case Scenario Konfirmasi Reservasi

Nama Use case :	Konfirmasi Reservasi
Aktor :	Pemilik_hewan
Pre-condition :	Aktor ingin melakukan konfirmasi reservasi
Post-condition :	Aktor telah melakukan konfirmasi reservasi.
Deskripsi :	Aktor melakukan konfirmasi reservasi dokter, dengan memilih klinik yang disediakan oleh system, aktor dapat memfilter lokasi klinik.
SKENARIO UTAMA	
Pemilik_hewan	System
1) Membuka halaman konfirmasi reservasi	
	1) Menampilkan info klinik sesuai pilihan aktor
	2) Menampilkan form tanggal, dan waktu yang wajib diisi oleh aktor.
	3) Menampilkan form tanggal, dan waktu yang wajib diisi oleh aktor.
4) Mengatur jadwal dengan mengisi form tanggal dan waktu yang telah disediakan.	

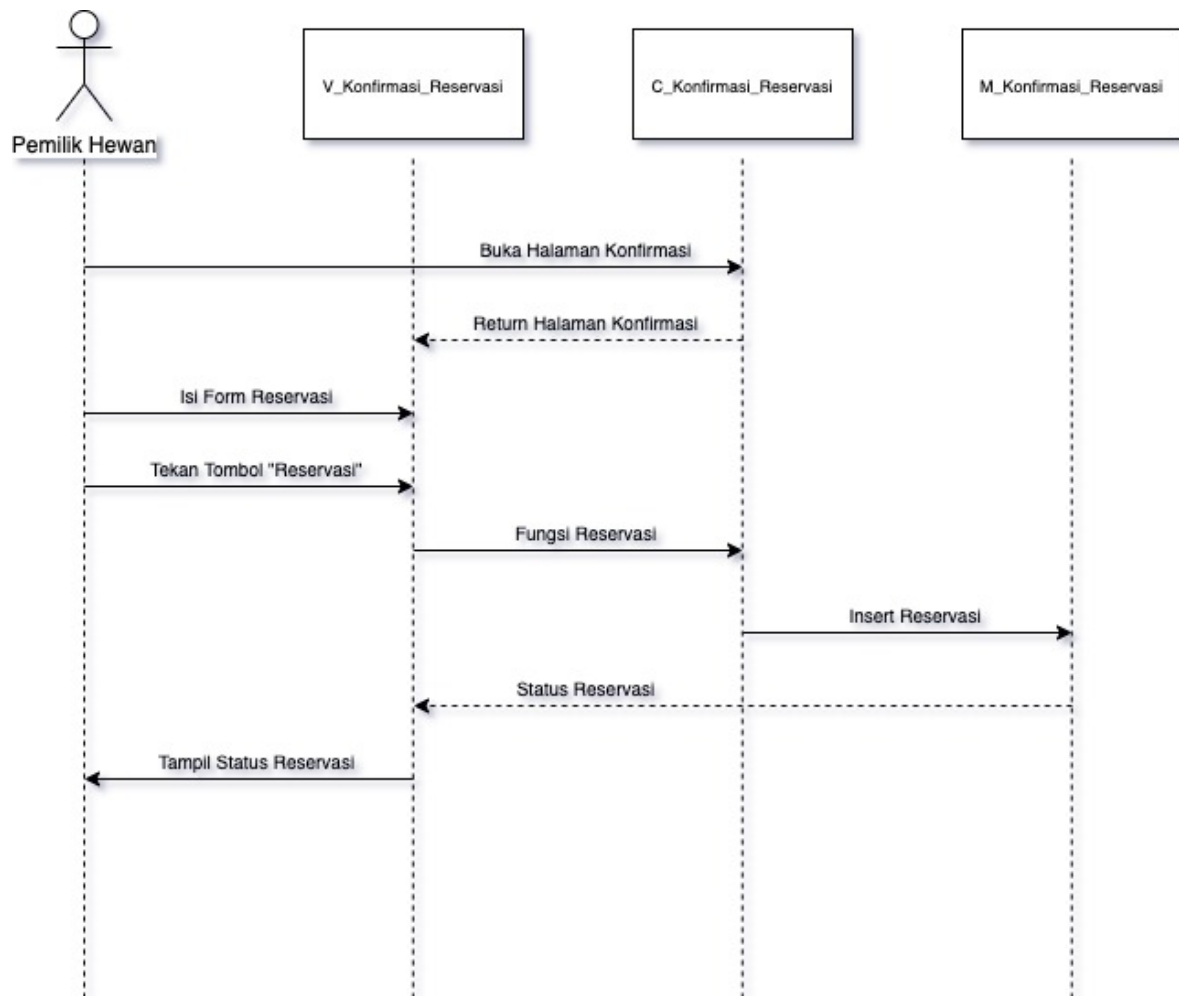
5) Menyelesaikan tahap reservasi dengan menekan tombol “Reservasi”.

3.1.7.1 Identifikasi Kelas

Tabel 15 Identifikasi Kelas Konfirmasi Reservasi

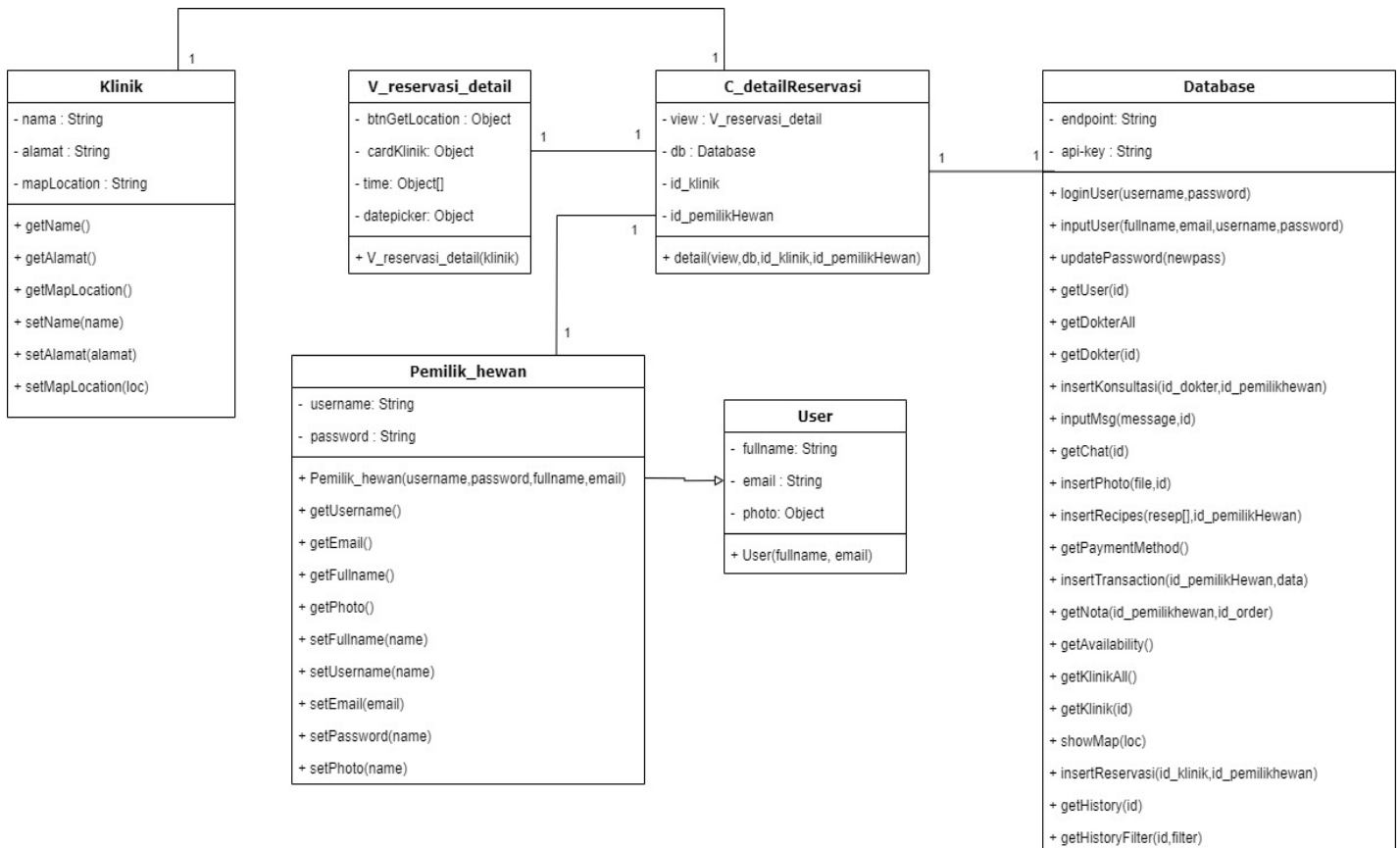
No	Nama Kelas Perancangan	Nama Kelas Analisis Terkait
9.	User	User
	Pemilik hewan	Pemilik hewan
	Dokter hewan	Dokter hewan
	V Konfirmasi Reservasi	View
	C Konfirmasi Reservasi	Controller

3.1.7.2 Sequence Diagram



Gambar 15 Sequence Diagram Konfirmasi Reservasi

3.1.7.3 Diagram Kelas



Gambar 16 Diagram Kelas Konfirmasi Reservasi

3.1.8 Use Case <Melihat History>

Tabel 16 Use Case Scenario Melihat History

Nama Use case :	Melihat history
Aktor :	Pemilik_hewan, dan Dokter
Pre-condition :	Aktor ingin memeriksa history transaksi, aktor belum melakukan pengecekan.
Post-condition :	Aktor telah memeriksa history transaksi.
Deskripsi :	Aktor melakukan pengecekan pada menu history transaksi yang telah dilakukan selama menggunakan aplikasi.

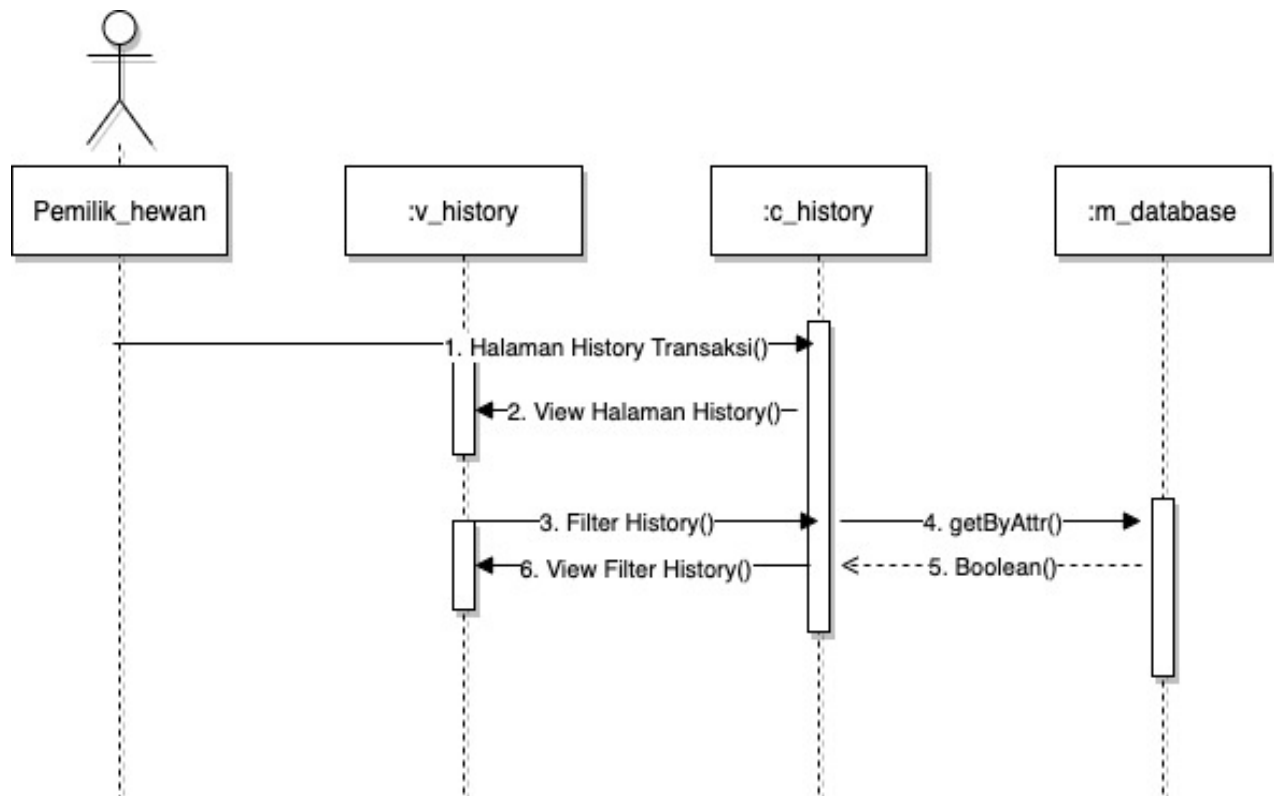
SKENARIO UTAMA	
Pemilik_hewan	System
1) Membuka fitur history	
	2) Menampilkan history aktor
3) Aktor dapat memfilter history berdasarkan 2 tipe: ongoing, dan completed.	
	4) Jika aktor memilih untuk memfilter history maka: - Jika aktor memilih ongoing maka sistem akan menampilkan history aktor yang sedang berlangsung saja. - Jika aktor memilih completed maka sistem akan menampilkan history transaksi aktor yang telah selesai.
5) Aktor melihat history transaksi.	

3.1.8.1 Identifikasi Kelas

Tabel 17 Identifikasi Kelas Melihat History

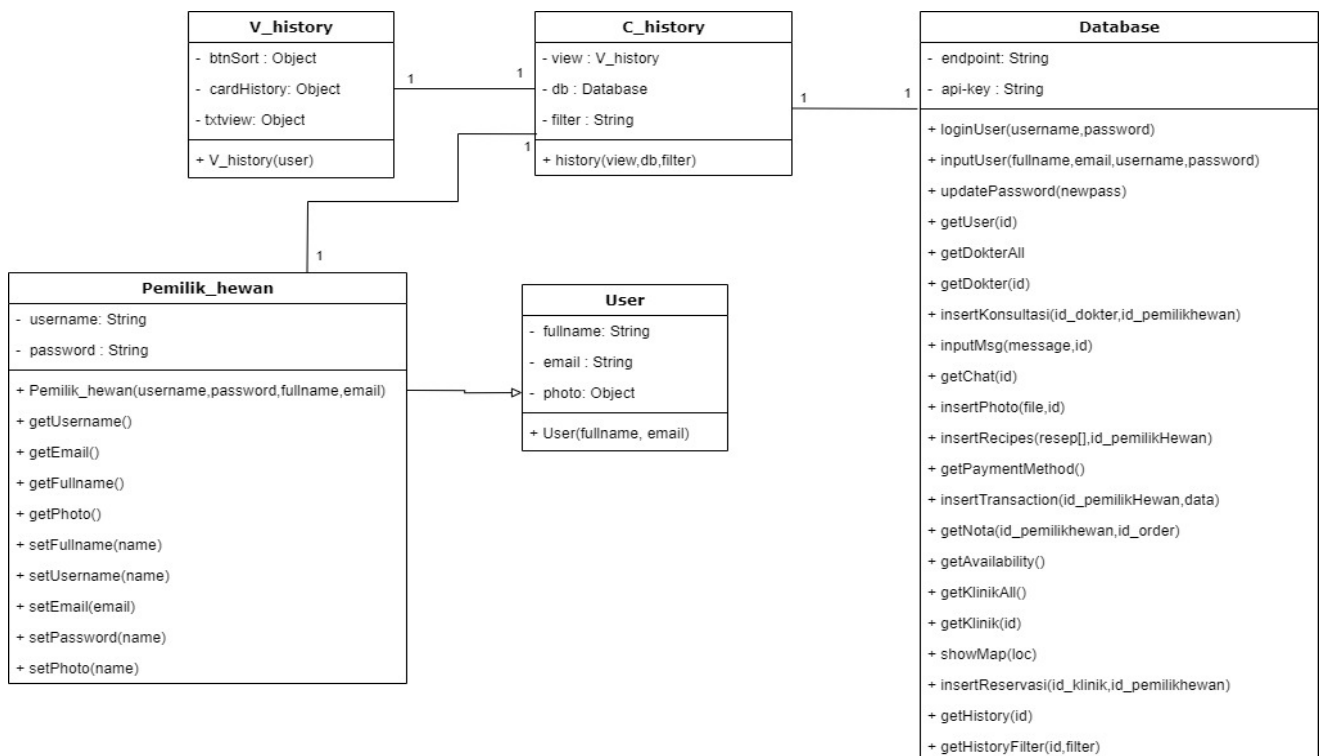
No	Nama Kelas Perancangan	Nama Kelas Analisis Terkait
10	User	User
	Pemilik hewan	Pemilik hewan
	Dokter hewan	Dokter hewan
	Database	Model
	V history	View
	C history	Controller

3.1.8.2 Sequence Diagram



Gambar 17 Sequence Diagram Melihat History

3.1.8.3 Diagram Kelas



Gambar 18 Diagram Kelas Melihat History

3.2 Perancangan Detil Kelas

3.2.1 Kelas <Dokter (extends dari user)>

Tabel 18 Perancangan Detail Kelas Dokter

Nama Operasi	Visibility (private,public)	Keterangan
Dokter(username,password, email,fullname : string)	Public	Constructor dokter untuk membuat objek dokter.
getUsername()	Public	Button untuk mendapatkan username.
setPhoto(name)	Public	Menambahkan foto.
setPassword(password)	Public	Menambahkan password
setEmail(email)	Public	Menambahkan email
setRating(rate)	Public	Menambahkan rating
setStatus(status)	Public	Menambahkan status
setAvailable(available)	Public	Menambahkan availability
setUsername(name)	Public	Menambahkan username
setFullname(name)	Public	Menambahkan nama lengkap
getEmail()	Public	Button untuk mendapatkan Email.
getFullname()	Public	Button untuk mendapatkan Nama lengkap.
getPhoto()	Public	Button untuk mendapatkan Foto.
getStatus()	Public	Button untuk mendapatkan Status.
getAvailability()	Public	Button untuk mendapatkan availability.
getRating()	Public	Button untuk mendapatkan rating.
Nama Atribut	Visibility (private,public)	Keterangan
Username	Private	String
Password	Private	String
Status	Private	String
Rating	Private	Integer
Availability	Private	Boolean

3.2.2 Kelas <Pemilik Hewan>

Tabel 19 Perancangan Detail Kelas Pemilik_Hewan

Nama Operasi	Visibility (private, public)	Keterangan
Pemilik_hewan	Public	Constructor untuk membuat objek pemilik hewan
getUsername()	Public	mendapatkan data username
getEmail()	Public	mendapatkan data email
getFullname()	Public	mendapatkan data full name
getPhoto()	Public	mendapatkan data foto
setFullname(name)	Public	menambahkan data full name
setUsername(name)	Public	menambahkan data username
setEmail(email)	Public	menambahkan data email
setPassword(name)	Public	menambahkan data password
setPhoto(name)	Public	menambahkan data foto
Nama Atribut	Visibility (private, public)	Keterangan
username	Private	String
password	Private	String

3.2.3 Kelas <Database>

Tabel 20 Perancangan Detail Kelas Database

Nama Operasi	Visibility (private, public)	Keterangan
loginUser(username, password)	Public	Constructor untuk menyimpan data login user
inputUser(fullname, email, username, password)	Public	Menginputkan data pribadi user
updatePassword(newpass)	Public	Mengubah password yang digunakan
getUser(id)	Public	Button untuk mendapatkan data user berdasarkan id
getDokterAll	Public	Button untuk mendapatkan semua data dokter
insertKonsultasi(id_dokter,id_pemilikhewan)	Public	Menginputkan data konsultasi antara pemilik hewan dengan dokter hewan

inputMsg(message, id)	Public	Menginputkan <i>message</i> chat
getChat(id)	Public	Button untuk mendapatkan data chat berdasarkan id
insertPhoto(file, id)	Public	Menginputkan foto
insertRecipes(resep[], id_pemilik hewan)	Public	Menginputkan data resep dokter kepada pemilik hewan
getPaymentMethod()	Public	Button untuk mendapatkan fungsi pembayaran
getNota(id_pemilik hwan, id_order)	Public	Button untuk mendapatkan nota pembayaran
insertTransaction(id_pemilikHewan, data)	Public	Menginputkan transaksi baru
getAvailability()	Public	Button untuk mendapatkan data ketersediaan
getKlinikAll()	Public	Button untuk mendapatkan data semua klinik
getKlinik(id)	Public	Button untuk mendapatkan data klinik berdasarkan id
showMap(loc)	Public	Menampilkan petunjuk lokasi/ <i>Map</i>
insertReservasi(id_klinik, id_pemilik hewan)	Public	Menginputkan data reservasi klinik
getHistory(id)	Public	Button untuk mendapatkan data history berdasarkan id
getHistoryFilter(id, filter)	Public	Button untuk mendapatkan data history berdasarkan filter
Nama Atribut	Visibility (private, public)	Keterangan
endpoint	Private	String
api-key	Private	String

3.2.4 Kelas <Klinik>

Tabel 21 Perancangan Detail Kelas Klinik

Nama Operasi	Visibility (private,public)	Keterangan
getName()	Public	Mendapatkan data Nama
getAlamat()	Public	Mendapatkan data Alamat
getMapLocation()	Public	Mendapatkan data Lokasi
setName(name)	Public	Menambahkan data Nama
setAlamat(alamat)	Public	Menambahkan data Alamat
setMapLocation(loc)	Public	Menambahkan data Lokasi
Nama Atribut	Visibility (private,public)	Keterangan
Nama	Private	String
Alamat	Private	String
mapLocation	Private	String

3.2.5 Kelas <C_Registrasi >

Tabel 22 Perancangan Detail Kelas C_Registrasi

Nama Operasi	Visibility (private, public)	Keterangan
register(view,db)	public	method controller untuk registrasi
Nama Atribut	Visibility (private, public)	Keterangan
view	Private	V_register
db	Private	Database

3.2.6 Kelas <V_Registrasi >

Tabel 23 Perancangan Detail Kelas V_Registrasi

Nama Operasi	Visibility (private, public)	Keterangan
-	-	-
Nama Atribut	Visibility (private, public)	Keterangan
btnRegister	Private	button register
inputUsername	Private	input text username
inputPassword	Private	input text password
inputEmail	private	input text email
inputFullname	private	input text full name

3.2.7 Kelas <C_Login >

Tabel 24 C_Login Perancangan Detail Kelas

Nama Operasi	Visibility (private,public)	Keterangan
login(view,db)	Public	Constructor untuk memanggil menu fungsi login
Nama Atribut	Visibility (private,public)	Keterangan
view	Private	V_login
db	Private	Database

3.2.8 Kelas <V_Login >

Tabel 25 Perancangan Detail Kelas V_Login

Nama Operasi	Visibility (private,public)	Keterangan
-	-	-
Nama Atribut	Visibility (private,public)	Keterangan
btnLogin	Private	Object
inputUsername	Private	Object
inputPassword	Private	Object

3.2.9 Kelas <C_Konsultasi >

Tabel 26 Perancangan Detail Kelas C_Konsultasi

Nama Operasi	Visibility (private,public)	Keterangan
Konsultasi(view,db,dokter)	Public	Constructor untuk memanggil menu konsultasi.
Nama Atribut	Visibility (private,public)	Keterangan
view	Private	V_reservasi_detail
db	Private	Database
dokter	Private	Dokter

3.2.10 Kelas <V_Konsultasi >

Tabel 27 Perancangan Detail Kelas V_Konsultasi

Nama Operasi	Visibility (private,public)	Keterangan
V_konsultasi(dokterList[])	Public	Constructor untuk menampilkan menu konsultasi yang berisi list dokter.
Nama Atribut	Visibility (private,public)	Keterangan
btnChat	Private	Object
inputSearch	Private	Object
cardDoctor	Private	Object

3.2.11 Kelas <C_Pembayaran>

Nama Operasi	Visibility (private,public)	Keterangan
chat(view,db,dokter,resep)	Public	Constructor untuk memanggil menu fungsi chat

Nama Atribut	Visibility (private,public)	Keterangan
view	Private	V_pembayaran
db	Private	Database
dokter	Private	Dokter
resep	Private	String[]

3.2.12 Kelas <V_Pembayaran>

Tabel 28 Perancangan Detail Kelas V_Pembayaran

Nama Operasi	Visibility (private,public)	Keterangan
V_pembayaran(dokter,pemilik_hewan,klinik,method[])	Public	Constructor untuk menampilkan menu pembayaran
Nama Atribut	Visibility (private,public)	Keterangan
btnBayar	Private	Object
paymentMethod	Private	Object[]
recipe	Private	Object[]
klinik	Private	Object
cardDoctor	Private	Object
cardPembayaran	Private	Object

3.2.13 Kelas <C_Reservasi>

Tabel 29 Perancangan Detail Kelas C_Reservasi

Nama Operasi	Visibility (private, public)	Keterangan
reservasi(view,db,klinik)	public	method controller untuk reservasi
Nama Atribut	Visibility (private, public)	Keterangan
view	Private	V_register
db	Private	Database
klinik	private	Klinik

3.2.14 Kelas <V_Reservasi>

Tabel 30 Perancangan Detail Kelas V_Reservasi

Nama Operasi	Visibility (private, public)	Keterangan
---------------------	---	-------------------

V_reservasi(klinikList[])	public	constructor untuk membuat objek V_reservasi dengan input parameter list klinik
Nama Atribut	Visibility (private, public)	Keterangan
btnReservasi	Private	button reservasi
btnGetLocation	Private	button get lokasi klinik
myLocation	Private	text view lokasi user
cardKlinik	private	card berisi data klinik

3.2.15 Kelas <C_Chat>

Tabel 31 Perancangan Detail Kelas C_Chat

Nama Operasi	Visibility (private, public)	Keterangan
Chat(view,db,id_pemilikHewan, id dokter)	Public	Constructor untuk memanggil menu chat
Nama Atribut	Visibility (private, public)	Keterangan
View	Private	V_chat
Db	Private	Database
Id_pemilikHewan	Private	String
Id_dokter	Private	String

3.2.16 Kelas <V_Chat>

Tabel 32 Perancangan Detail Kelas V_Chat

Nama Operasi	Visibility (private, public)	Keterangan
V_chat(dokter,pemilik_hewan)	Private	Constructor untuk menampilkan chat pemilik hewan dengan dokter
Nama Atribut	Visibility (private, public)	Keterangan
btnSend	Private	Object
inputMsg	Private	Object
PhMsg	Private	Objetc[]
DMsg	Private	Object[]

3.2.17 Kelas <C_detailReservasi>

Tabel 33 Perancangan Detail Kelas C_detailReservasi

Nama Operasi	Visibility (private,public)	Keterangan
Detail(view,db,id_klinik, id_pemilikHewan)	Public	Constructor untuk memanggil reservasi detail user.
Nama Atribut	Visibility (private,public)	Keterangan
view	Private	V_reservasi_detail
db	Private	Database
Id_klinik	Private	-
Id_pemilikHewan	Private	-

3.2.18 Kelas <V_detailReservasi>

Tabel 34 Perancangan Detail Kelas V_detailReservasi

Nama Operasi	Visibility (private,public)	Keterangan
V_detail_reservasi(klinik)	Public	Constructor untuk menampilkan reservasi detail user.
Nama Atribut	Visibility (private,public)	Keterangan
btnGetLocation	Private	Object
cardKlinik	Private	Object
time	Private	Object[]
datepicker	Private	Object

3.2.19 Kelas <C_history>

Tabel 35 Perancangan Detail Kelas C_history

Nama Operasi	Visibility (private, public)	Keterangan
History(view,db,filter)	Public	Constructor untuk memanggil menu history
Nama Atribut	Visibility (private, public)	Keterangan
View	Private	V_history
Db	Private	Database
filter	Private	filter

3.2.20 Kelas <V_history>

Tabel 36 Perancangan Detail Kelas V_history

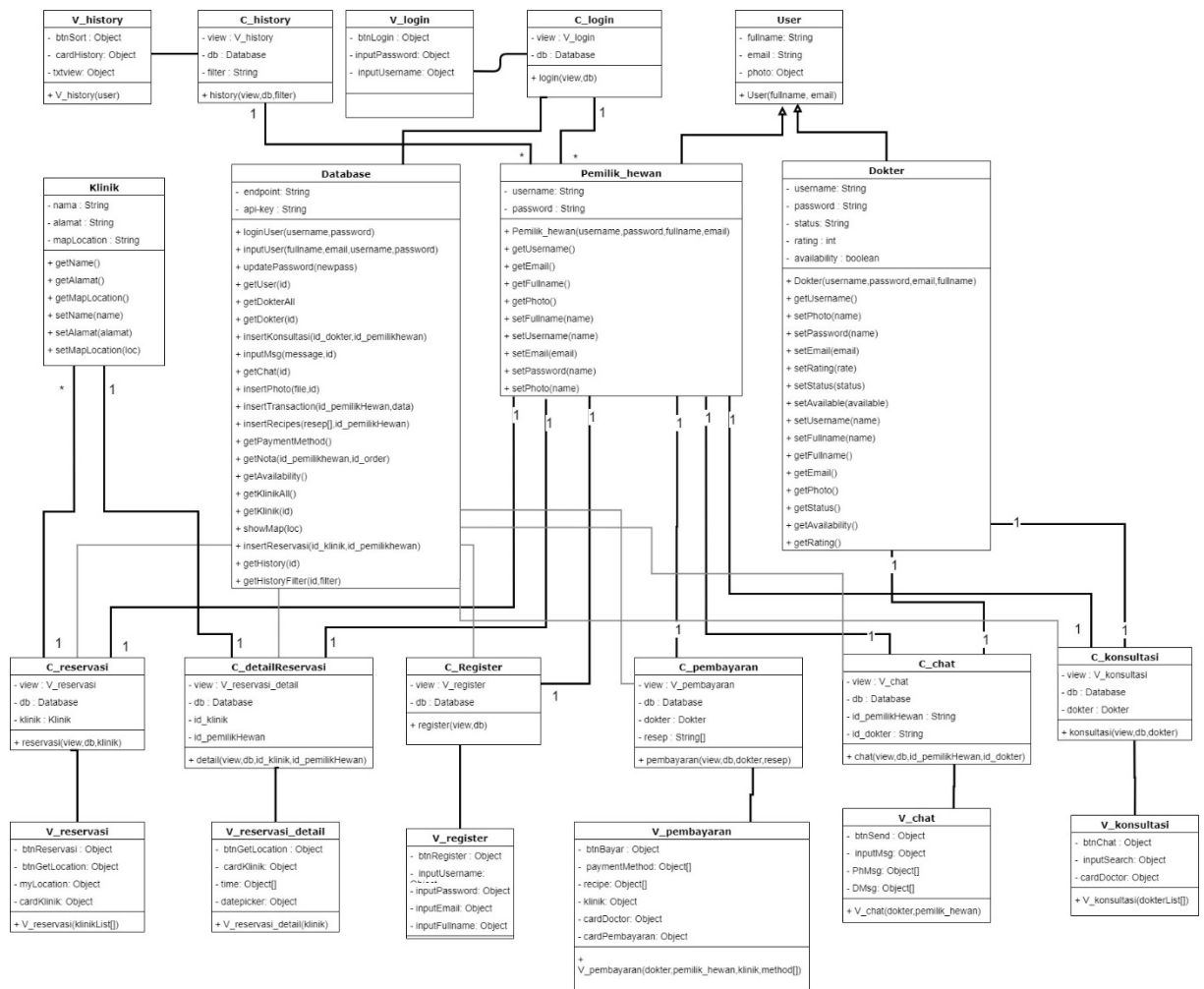
Nama Operasi	Visibility (private, public)	Keterangan
V_history(user)	Public	Constructor untuk menampilkan history
Nama Atribut	Visibility (private, public)	Keterangan
btnSort	Private	Object
cardHistory	Private	Object
txtview	Private	Object

3.2.21 Kelas <User>

Tabel 37 Perancangan Detail Kelas User

Nama Operasi	Visibility (private,public)	Keterangan
User(fullname, email)	Public	Constructor untuk memanggil data user
Nama Atribut	Visibility (private,public)	Keterangan
fullname	Private	String
Email	Private	String
Photo	Private	Object

3.3 Diagram Kelas Keseluruhan



Gambar 19 Class Diagram Keseluruhan

3.4 Algoritma/Query

Tabel 38 Algoritma Register, getResepDokter, getAllHistoryByStatus

Nama kelas : User, Pemilik_hewan, Dokter, Nama operasi : Register, getResepDokter, getAllHistoryByStatus Algoritma : (Algo-00x)
<p>Register</p> username = input fullname = input email = input password = input If (select * from user where username = username) then begin If(username != null and password != null and fullname != null and email != null) then begin query = insert into user values (username,fullname,email,password) end Else output("field tidak boleh kosong") End if end Else output("username sudah terpakai") End if <p>getResepDokter</p> username, id_konsultasi = input select * from resep where username = username and id_konsultasi = id_konsultasi <p>getAllHistoryByStatus</p> status = input select * from history where status = status

No Query	Query	Keterangan
Q-001	insert into user values (value_column1,value_column2, .. .)	melakukan insert user baru kedalam tabel user
Q-002	select * from resep where username = value_username and id_konsultasi = value_id	mengambil data resep dokter dari tabel resep berdasarkan user dan id konsultasi
Q-003	select * from history where status = value	mengambil data history dari tabel history berdasarkan status

Tabel 39
Query
Register,

getResepDokter, getAllHistoryByStatus

Tabel 40 Algoritma getDokterAll(), getKlinikAll(), getAllHistoryFilter(filter)

Nama kelas : User, Pemilik_hewan, Dokter, Database Nama operasi : getDokterAll(), getKlinikAll(), getAllHistoryFilter(filter) Algoritma : (Algo-00x)
<p>getDokterAll() select *from Dokter;</p> <p>getKlinikAll() select * from Klinik;</p> <p>getAllHistoryFilter(filter_history) filter_history = input select * from History where filter = filter;</p>

Tabel 41 Query getDokterAll(), getKlinikAll(), getAllHistoryFilter(filter)

No Query	Query	Keterangan
Q-004	SELECT * FROM Dokter;	Mengambil data seluruh dokter dari table dokter.
Q-005	SELECT * FROM Klinik;	mengambil data seluruh klinik dari table klinik.
Q-006	SELECT * FROM Database WHERE filter = value;	mengambil data history dari tabel history berdasarkan status.

Tabel 42 Algoritma *getAllDokterByID()*, *getAllKlinikByID()*, *getAllHistory()*

<p>Nama kelas : user, pemilik_hewan, dokter, klinik, database</p> <p>Nama Operasi : <i>getAllDokterByID()</i>, <i>getAllKlinikByID()</i>, <i>getAllHistory()</i></p> <p>Algoritma : (Algo-00x)</p>
<p>getAllDokterByID() id_dokter = input select * from dokter where id_dokter = id_dokter</p> <p>getAllKlinikByID() id_klinik = input select * from klinik where id_klinik = id_klinik</p> <p>getAllHistory() select * from history</p>

Tabel 43 Query *getAllDokterByID()*, *getAllKlinikByID()*, *getAllHistory()*

No Query	Query	Keterangan
Q-007	select * from dokter where id_dokter = value_id	Mengambil data dokter dari tabel dokter berdasarkan id dokter
Q-008	select * from klinik where id_klinik = value_id	Mengambil data klinik dari tabel klinik berdasarkan id klinik
Q-009	select * from history	Mengambil semua data history dari tabel history

Tabel 44 Algoritma *Login*, *getMetodePembayaran*, *insertResevasi*

<p>Nama kelas : User, Pemilik_hewan, Dokter, Klinik, Database</p> <p>Nama operasi : Login, <i>getMetodePembayaran</i>, <i>insertResevasi</i></p> <p>Algoritma : (Algo-00x)</p>
<p>Login username = input password = input If (select * from user where username = username and password = password) then begin query = select * from user where username = username and password = password end</p>

```

Else
    output("username atau password salah")
End if

getMetodePembayaran
SELECT * FROM Metode_Pembayaran

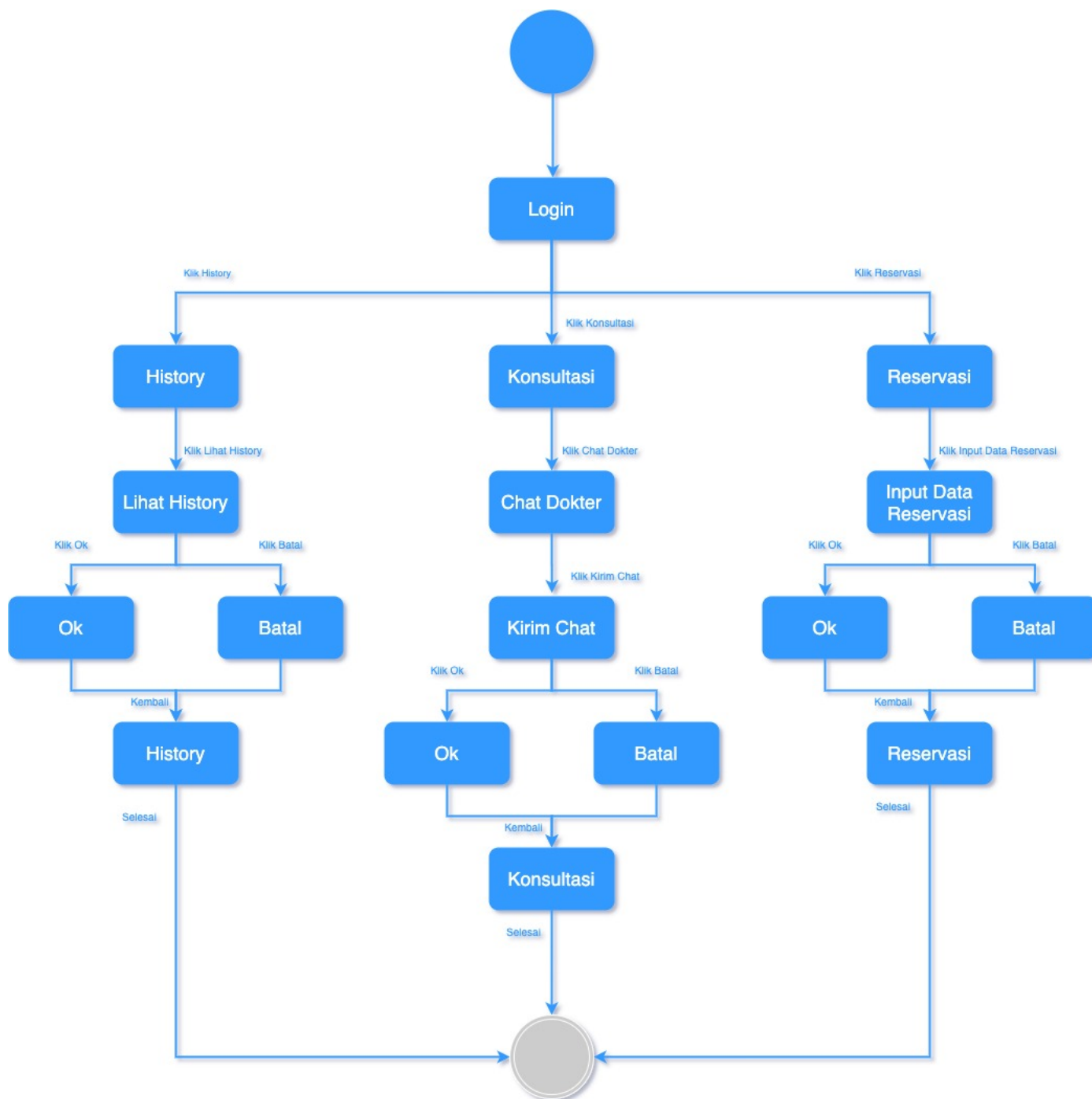
insertReservasi
klinik_id = input
tanggal_reservasi = input
jam_reservasi = input
username = input
If (select * from user where username = username) then
begin
    query = insert into Reservasi (klinik_id, tanggal_reservasi, jam_reservasi, username)
    values (value_klinik_id, value_tanggal_reservasi, value_jam_reservasi, username)
end
Else
    output("username tidak sesuai")
End if

```

Tabel 45 Query Login, getMetodePembayaran, insertReservasi

No Query	Query	Keterangan
Q-010	SELECT * FROM user WHERE username = value_username and password = value_password	Mengambil data dan pengkondisian username dan password
Q-011	SELECT * FROM Metode_Pembayaran	Mengambil data Metode Pembayaran
Q-012	INSERT INTO <i>Reservasi</i> (<i>klinik_id, tanggal_reservasi, jam_reservasi</i>) VALUES (<i>value_klinik_id, value_tanggal_reservasi, value_jam_reservasi</i>)	Memasukan data Reservasi

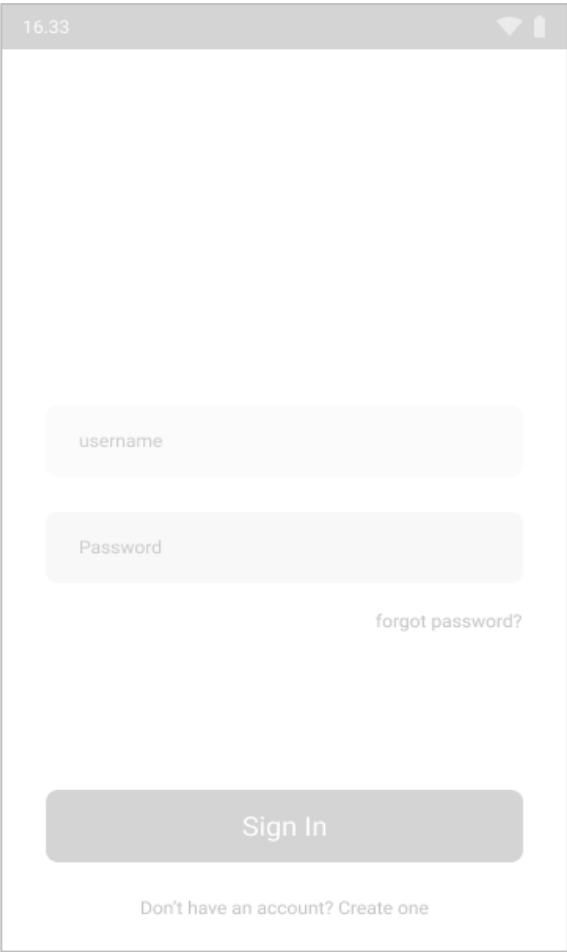
3.5 Diagram Statechart



Gambar 20 Diagram State Chart

3.6 Perancangan Antarmuka

SIGN IN (LOGIN)



Gambar 21 Wireframe Login

Antarmuka : 1 (Login)

Tabel 46 Antarmuka Login

Id Objek	Jenis	Nama	Keterangan
TextBox1	TextBox	Username	Diisi dengan username yang telah terdaftar
TextBox2	TextBox	Password	Diisi dengan password yang telah terdaftar
Hyperlink1	Hyperlink	Forgot Password?	Jika di klik, akan mengaktifkan proses pembuatan ulang password
Button1	Button	Sign In	Jika di klik, akan mengaktifkan proses validasi username dan password. Jika validasi berhasil, maka user akan diarahkan pada halaman menu
Hyperlink2	Hyperlink	Create One	Jika di klik, maka akan mengarahkan user pada halaman registrasi akun

SIGN UP (REGISTER)

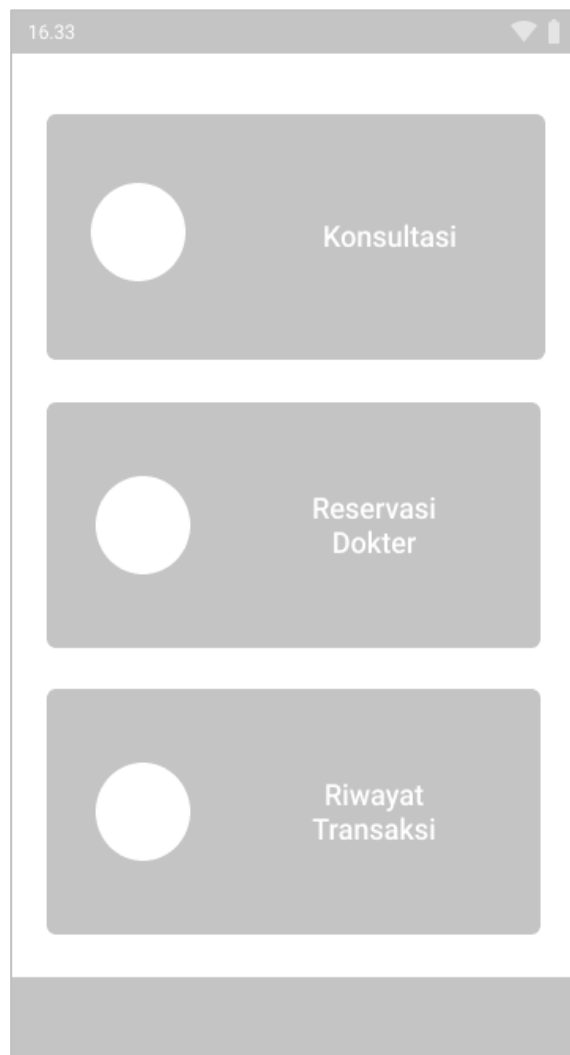
Gambar 22 Wireframe Register

Antarmuka : 2 (Registrasi Akun)

Tabel 47 Antarmuka Registrasi

Id Objek	Jenis	Nama	Keterangan
TextBox1	TextBox	Full Name	Diisi dengan nama lengkap user yang akan didaftarkan
TextBox2	TextBox	email	Diisi dengan email user yang akan didaftarkan
TextBox3	TextBox	Username	Diisi dengan username user yang akan didaftarkan
TextBox4	TextBox	Password	Diisi dengan password user yang akan didaftarkan
Button1	Button	Sign Up	Jika di klik, maka data user akan disimpan pada database dan user telah terdaftar sebagai user baru

HOME



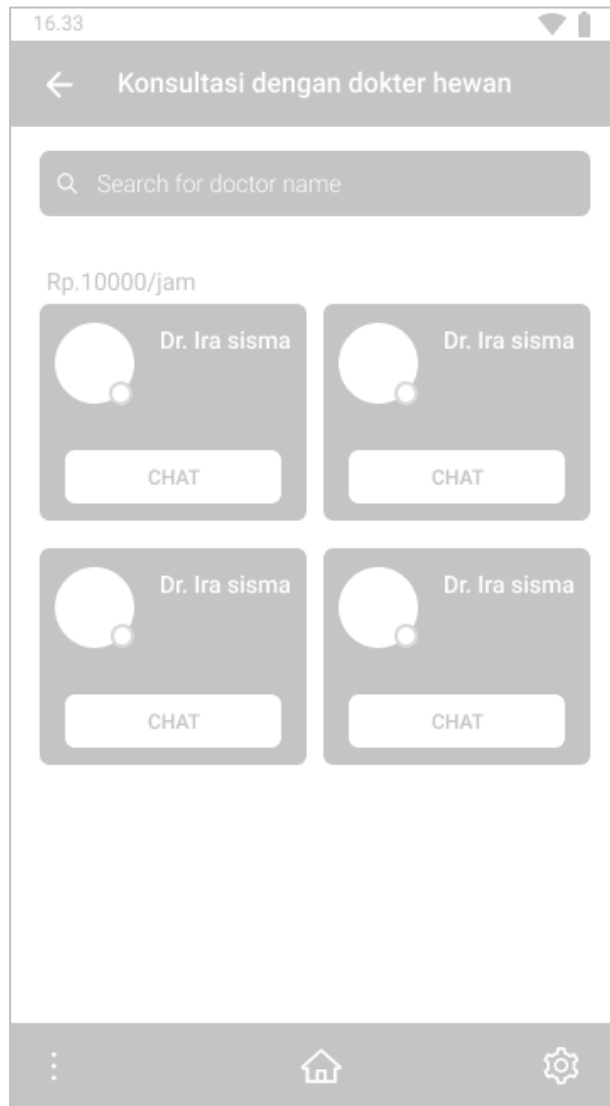
Gambar 23 Wireframe Home

Antarmuka : 3 (Menu / Home)

Tabel 48 Antarmuka Home

<u>Id_Objek</u>	<u>Jenis</u>	<u>Nama</u>	<u>Keterangan</u>
Button1	Button	Konsultasi	Jika di klik, maka akan mengarahkan user pada halaman konsultasi
Button2	Button	Reservasi Dokter	Jika di klik, maka akan mengarahkan user pada halaman reservasi dokter
Button3	Button	Riwayat Transaksi	Jika di klik, maka akan mengarahkan user pada halaman riwayat transaksi

KONSULTASI



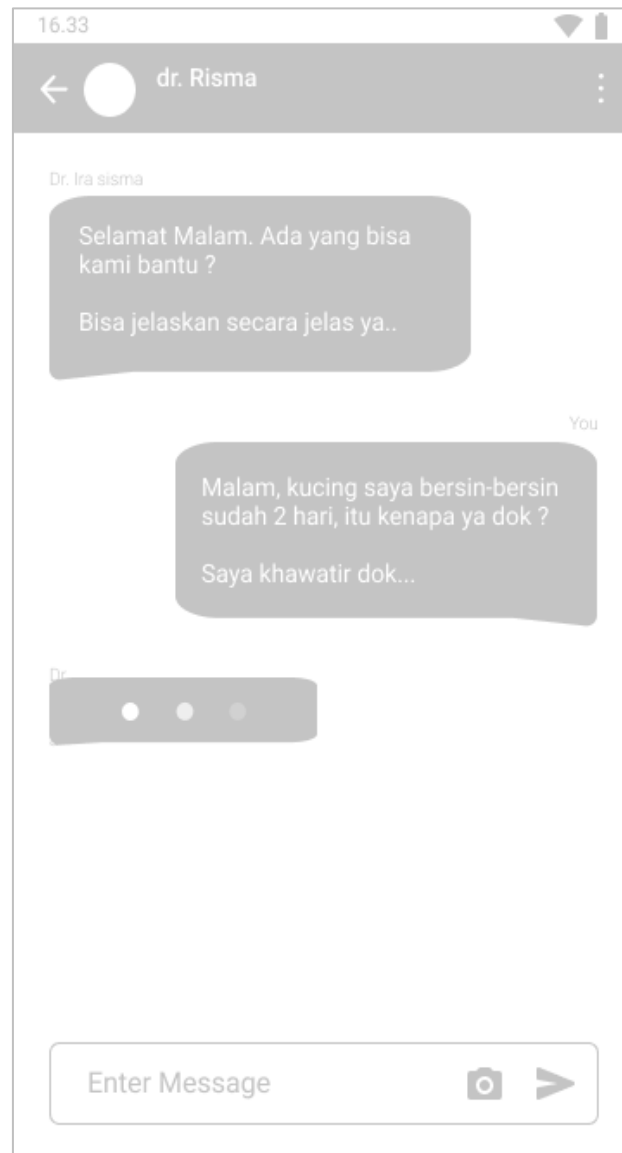
Gambar 24 Wireframe Konsultasi

Antarmuka : 4 (Halaman Konsultasi)

Tabel 49 Antarmuka Halaman Konsultasi

Id Objek	Jenis	Nama	Keterangan
Button1	Button		Jika di klik, maka akan mengarahkan user pada halaman sebelumnya
search	searchBox	Search for doctor name	Diisi dengan nama dokter yang ingin dicari, dan sistem akan mencari dokter berdasarkan nama dokter yang telah diinputkan
Button2	Button	Chat	Jika di klik, maka akan mengarahkan user pada halaman chat sesuai dengan dokter yang diinginkan

CHAT



Gambar 25 Wireframe Chat Dokter & Pemilik_hewan

Antarmuka : 5 (Chat Room)

Tabel 50 Antarmuka Chat room

Id_Objek	Jenis	Nama	Keterangan
Button1	Button		Jika di klik, maka akan mengarahkan user pada halaman sebelumnya
Button2	Button		Jika di klik, maka akan menampilkan menu pilihan untuk detail chat seperti media, view contact, more
TextBox1	TextBox	Enter Message	Diisi dengan text chat yang akan dikirimkan ke lawan bicara pada chat room
Button3	Button		Jika di klik, maka akan mengarahkan user pada kamera aplikasi
Button4	Button		Jika di klik, maka akan mengirimkan text yang berada di dalam TextBox1 ke lawan bicara pada chat room

PEMBAYARAN

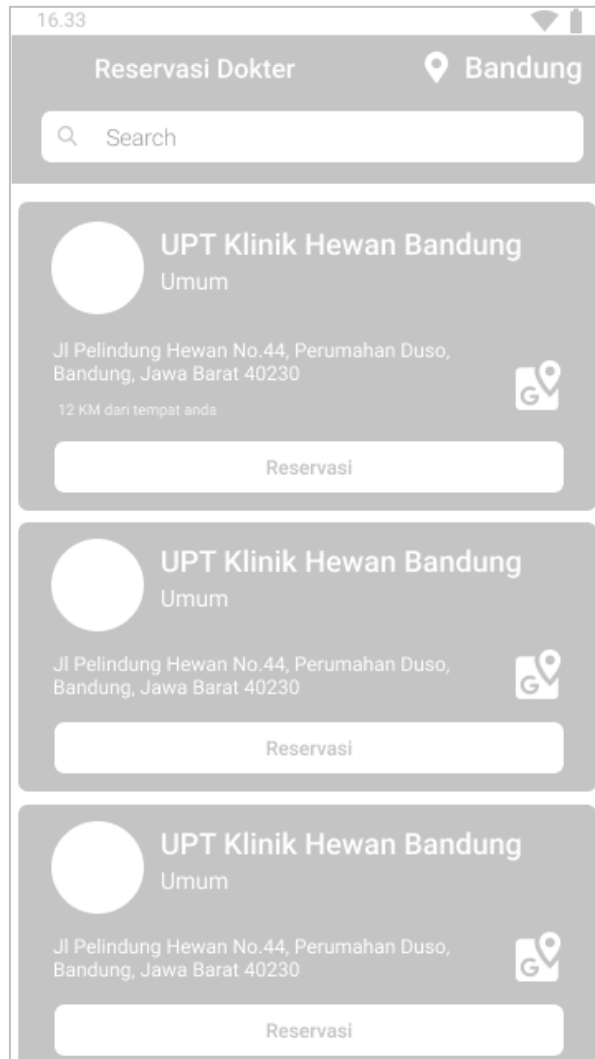
Gambar 26 Wireframe Pembayaran

Antarmuka : 6 (Halaman Pembayaran)

Tabel 51 Antarmuka Pembayaran

Id Objek	Jenis	Nama	Keterangan
Button1	Button		Jika di klik, menampilkan menu pilihan payment method
Button2	Button	Bayar	Jika di klik, maka akan dilakukan proses transaksi yang kemudian akan disimpan kedalam database

RESERVASI



Gambar 27 Wireframe Reservasi Dokter

Antarmuka : 7 (Halaman Reservasi)

Tabel 52 Antarmuka Reservasi

Id Objek	Jenis	Nama	Keterangan
Search	SearchBox	Search	Diisi dengan nama atau alamat klinik yang akan dicari
Button1	Button		Jika di klik, maka akan menampilkan detail lokasi klinik pada google maps
Button2	Button	Reservasi	Jika di klik, maka akan mengarahkan user ke halaman detail reservasi

KONFIRMASI RESERVASI

Gambar 28 Wireframe Konfirmasi Reservasi

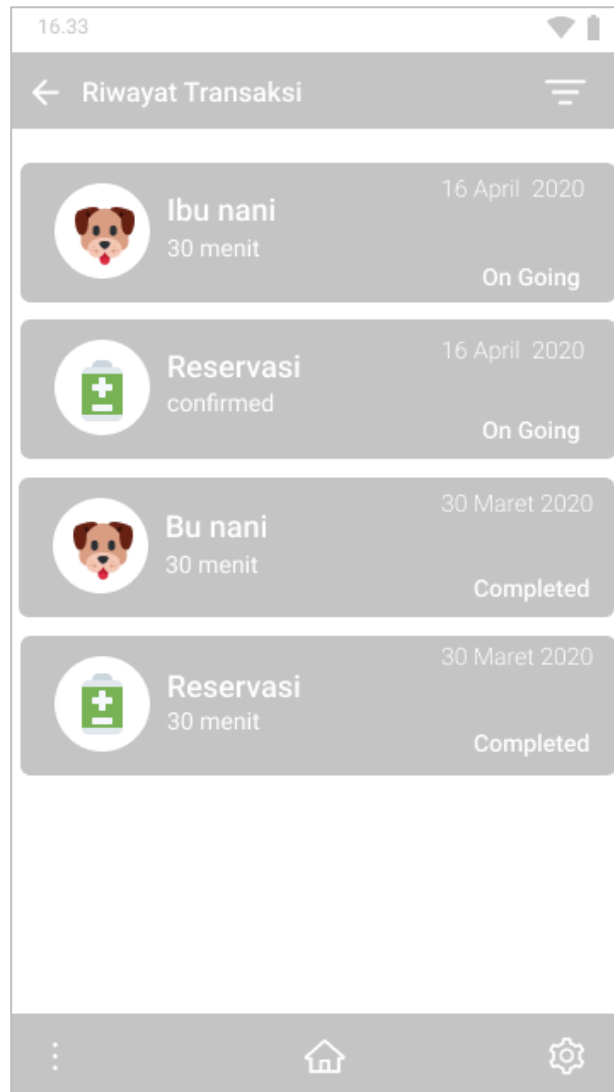
Antarmuka : 8 (Halaman Konfirmasi Reservasi)

Tabel 53 Antarmuka Konfirmasi Reservasi

Id Objek	Jenis	Nama	Keterangan
Button0	Button		Jika di klik, maka akan mengarahkan user ke halaman sebelumnya
Date	DateBox	02/10/2019	Jika di klik, maka akan menampilkan kalender yang dapat dipilih tanggalnya oleh user
Button1	Button	10:00	Jika di klik, maka jam yang dipilih akan terdaftar kedalam detail reservasi
Button2	Button	11:00	Jika di klik, maka jam yang dipilih akan terdaftar kedalam detail reservasi
Button3	Button	12:00	Jika di klik, maka jam yang dipilih akan terdaftar kedalam detail reservasi
Button4	Button	13:00	Jika di klik, maka jam yang dipilih akan terdaftar kedalam detail reservasi

Button5	Button	14:00	Jika di klik, maka jam yang dipilih akan terdaftar kedalam detail reservasi
Button6	Button	15:00	Jika di klik, maka jam yang dipilih akan terdaftar kedalam detail reservasi
Button7	Button	Reservasi	Jika di klik, maka akan menyimpan data konfirmasi reservasi kedalam database

HISTORY



Gambar 29 Wireframe History

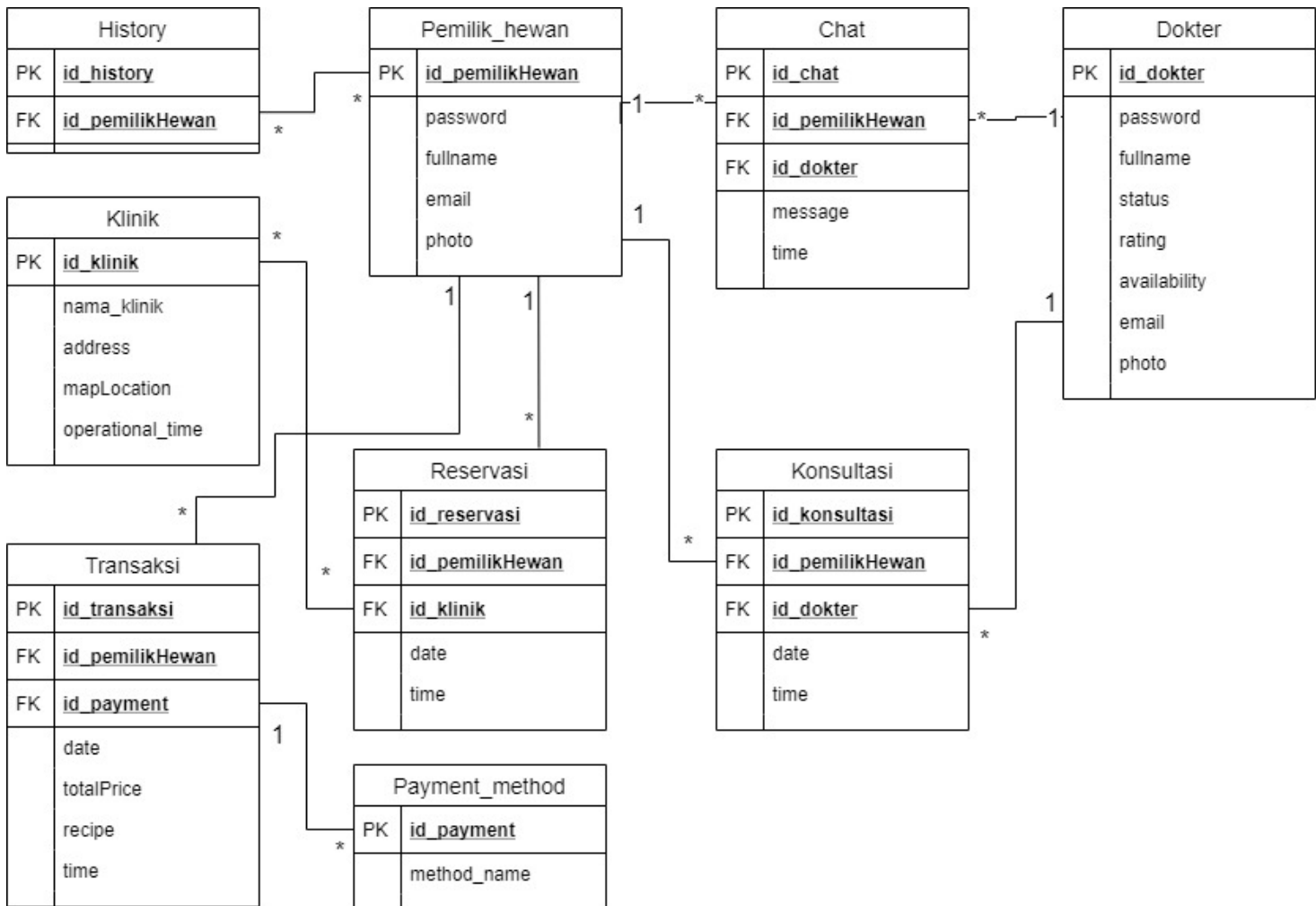
Antarmuka : 9 (Riwayat Transaksi)

Tabel 54 Antarmuka Riwayat Transaksi (History)

Id_Objek	Jenis	Nama	Keterangan
Button1	Button		Jika di klik, maka akan mengarahkan user ke halaman sebelumnya
Button2	Button		Jika di klik, maka akan menampilkan menu filter berdasarkan status seperti On Going/Completed.

			Setelh menu filter di klik, maka akan dijalankan proses filter riwayat transaksi
--	--	--	--

3.7 Perancangan Representasi Persistensi Kelas



Gambar 30 Skema Relasi

4 Matriks Kerunutan

Tabel 55 Matriks Kerunutan

Kode Kebutuhan	Usecase Terkait	Kelas
FR-01	Registrasi Akun	User, database, pemilik_hewan, dokter, C_register, V_register
FR-02	Konsultasi	User, database, dokter, pemilik_hewan, C_konsultasi, V_konsultasi
FR-03	Chat	User, database, dokter, pemilik_hewan, C_chat, V_chat
FR-04	Membayar tarif konsultasi	User, database, dokter, pemilik_hewan, v_pembayaran, c_pembayaran
FR-05	Reservasi	User, database, pemilik_hewan, klinik, V_reservasi, C_reservasi
FR-06	Konfirmasi Reservasi	User, database, pemilik_hewan, klinik, V_reservasiDetail, C_reservasiDetail
FR-07	Melihat History	User, database, pemilik_hewan, C_history, V_history
FR-08	Login	User, database, pemilik_hewan, C_login, V_login