**DOKUMEN**

**SPESIFIKASI KEBUTUHAN 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 |
| Jourdy Hanif Murzani | 1301180152 | Back-End |
| Wardah Hanifah | 1301184257 | UI/UX |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Program Studi S1 Informatika**  **-**  **Fakultas Informatika** | **Nomor Dokumen** | | **Halaman** |
| **SKPL-** | | < 1/27> |
| **Revisi** | < 0 > | Tgl : 8 Maret 2020 |

# **Daftar Perubahan**

|  |  |
| --- | --- |
| **Revisi** | **Deskripsi** |
| **A** |  |
| **B** |  |
| **C** |  |
| **D** |  |
| **E** |  |
| **F** |  |
| **G** |  |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **INDEX** | **-** | **A** | **B** | **C** | **D** | **E** | **F** | **G** |
| Tgl |  |  |  |  |  |  |  |  |
| Ditulis oleh |  |  |  |  |  |  |  |  |
| Diperiksa oleh |  |  |  |  |  |  |  |  |
| Disetujui oleh |  |  |  |  |  |  |  |  |

# 

# **Daftar Halaman Perubahan**

|  |  |  |  |
| --- | --- | --- | --- |
| **Halaman** | **Revisi** | **Halaman** | **Revisi** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**DAFTAR ISI**

[**Daftar Perubahan 2**](#_heading=h.30j0zll)

[**Daftar Halaman Perubahan 4**](#_heading=h.17dp8vu)

[**1. Pendahuluan 6**](#_heading=h.2et92p0)

[**1.1.** **Tujuan Penulisan Dokumen** 6](#_heading=h.tyjcwt)

[**1.2.** **Ruang Lingkup / Cakupan Dokumen** 6](#_heading=h.3dy6vkm)

[**1.3.** **Definisi, Singkatan, dan Akronim** 6](#_heading=h.1t3h5sf)

[**1.4.** **Referensi** 7](#_heading=h.4d34og8)

[**2.1.** ***Statement of Objective* Perangkat Lunak** 7](#_heading=h.2s8eyo1)

[**2.3.** **Profil dan Karakteristik Pengguna** 8](#_heading=h.3rdcrjn)

[**2.4.** **Arsitektur Lingkungan Operasi Perangkat Lunak (digambarkan)** 9](#_heading=h.26in1rg)

[**2.5.** **Kebutuhan Perangkat Keras** 10](#_heading=h.lnxbz9)

[**2.6.** **Asumsi dan Batasan Perangkat Lunak** 10](#_heading=h.35nkun2)

[**3. Deskripsi Rinci Perangkat Lunak 11**](#_heading=h.1ksv4uv)

[**3.1.** **Deskripsi Kebutuhan** 11](#_heading=h.44sinio)

[***3.1.1.******Kebutuhan Fungsional*** *11*](#_heading=h.2jxsxqh)

[***3.1.2.******Kebutuhan Non-Fungsional*** *12*](#_heading=h.z337ya)

[**3.2**](#_heading=h.3j2qqm3) **Pemodelan Analisis** 12

[***3.2.1. Data Flow Diagram*** *12*](#_heading=h.3whwml4)

[***3.2.1.2 PSPEC*** *16*](#_heading=h.2bn6wsx)

[***3.2.2.******Usecase Diagram*** *20*](#_heading=h.1y810tw)

[***3.2.2.1. Usecase Skenario*** *20*](#_heading=h.4i7ojhp)

[**3.3.** **Class Diagram** 27](#_heading=h.qsh70q)

# **1.** **Pendahuluan**

## **1.1.** **Tujuan Penulisan Dokumen**

Dokumen ini berisi penjelasan Spesifikasi Kebutuhan Perangkat Lunak (SKPL) atau *Software Requirement Specification* (*SRS*) dengan pendekatan berorientasi objek dari perangkat lunak yang akan dibuat.

## **1.2.** **Ruang Lingkup / Cakupan Dokumen**

Tujuan penulisan dokumen ini adalah untuk menguraikan proses-proses tahapan pembuatan perangkat lunak yang akan dibangun. Bagi pihak pengembang, SKPL ini dapat digunakan sebagai acuan dalam mengembangkan perangkat lunak secara detail agar perangkat lunak yang dibangun sesuai dengan kebutuhan yang diinginkan pengguna. Diharapkan dengan adanya dokumen SKPL ini pengembangan perangkat lunak lebih terarah kepada tujuan dari pengembangan perangkat lunak ini. Sedangkan bagi pengguna, dokumen SKPL ini digunakan untuk mencatat spesifikasi kebutuhan perangkat lunak yang akan dibangun dan sesuai dengan harapan yang diinginkan pengguna.

## **1.3.** **Definisi, Singkatan, dan Akronim**

Berikut adalah daftar definisi dan istilah penting yang digunakan dalam dokumen SKPL ini:

- SKPL: Dokumen hasil analisis yang berisi spesifikasi kebutuhan perangkat lunak.

*- Use Case Diagram*: Gambaran grafik dari beberapa atau semua aktor, *Use case,* dan interaksi nya.

- *Class Diagram*:Class adalah sebuah spesifikasi yang jika diinstansiasi akan menghasilkan sebuah objek dan merupakan inti dari pengembangan desain berorientasi objek. Class menggambarkan keadaan (atribut/properti) suatu sistem, sekaligus menawarkan layanan untuk memanipulasi keadaan tersebut (metoda/fungsi).

## **1.4.** **Referensi**

1) Dharwiyanti, Sri, and Romi Satria Wahono. “Pengantar Unified Modeling Language (UML).” *Ilmu Komputer* (2003): 1-13.

2) Khoiriah, An Nisa Sari, and Radiant Victor Imbar. “Pembuatan Sistem Akademik Pada Sma Pgii Dengan Metode Simple Additive Weighting (SAW).” *Jurnal Teknik Informatika dan Sistem Informasi* 1.1 (2015): 22-26.

**2.** **Deskripsi Global Perangkat Lunak**

## **2.1.** ***Statement of Objective* Perangkat Lunak**

HETA merupakan aplikasi yang serupa dengan halodoc dan berbasis mobile, akan tetapi berbeda dengan halodoc yang dimana merupakan forum konsultasi dan pelayanan untuk pengobatan manusia. HETA mempunyai beberapa fitur diantaranya adalah, fitur untuk konsultasi dengan dokter hewan yang sudah bermitra dengan HETA, mengenai kesehatan hewan peliharaan. Sebelum melakukan konsultasi, user dapat melakukan pembayaran setelah layanan diterima dan pembayaran dapat dilakukan secara mobile.

**2.2.** **Perspektif dan Fungsi Perangkat Lunak**

Perangkat lunak ini memiliki beberapa fungsi utama, antara lain:

1. Registrasi Akun
2. *Login*
3. *Update Profile*
4. *Update Password*
5. *Inpu*t Data Hewan Peliharaan
6. *Input* Data Dokter Hewan
7. Pencarian Dokter Hewan
8. Konsultasi Dokter Hewan Via *Chat*
9. Reservasi Dokter Hewan
10. Pembayaran *Online*
11. *History* Transaksi

## **2.3.** **Profil dan Karakteristik Pengguna**

|  |  |  |  |
| --- | --- | --- | --- |
| **Kategori**  **Pengguna** | **Tugas** | **Hak Akses Ke Aplikasi** | **Kemampuan Yang Harus Dimiliki** |
| Pemilik Hewan Peliharaan | Menginputkan informasi hewan peliharaan, melakukan konsultasi dengan dokter hewan melalui *chat*, melakukan reservasi dokter hewan,dan melakukan pembayaran | -*Login*  -*Update Profile*  -*Update Password*  -*Input* Data Hewan Peliharaan  -Pencarian Dokter Hewan  -Konsultasi Dokter Hewan Via *Chat*  -Reservasi Dokter Hewan  -Pembayaran *Online*  -*History* Transaksi | -Mengoperasikan *handphone* |
| Dokter Hewan | Memberikan konsultasi kepada pemilik hewan peliharaan melalui chat, dan menerima pembayaran | -*Login*  -*Update Profile*  -*Update Password*  -I*nput* Data Dokter Hewan  -Konsultasi Dokter Hewan Via *Chat*  -*History* Transaksi | -Mengoperasikan *handphone* |

## **2.4.** **Arsitektur Lingkungan Operasi Perangkat Lunak (digambarkan)**

Asumsi-asumsi pada perangkat lunak ini adalah:

1. *Use*r yang ada pada aplikasi adalah pemilik hewan peliharaan
2. Semua *user* dapat *login, update profile*, dan *update password*.
3. Pemilik hewan peliharaan dapat melakukan pencarian dokter hewan, melakukan konsultasi dengan dokter hewan melalui *chat*, membuat janji temu dengan melakukan reservasi dokter hewan, melakukan pembayaran dan dapat melihat *history* transaksi.

## 

## **2.5.** **Kebutuhan Perangkat Keras**

**2.5.1 Kebutuhan Perangkat Keras**

* Intel Processor Dual Core Keatas
* RAM 5 GB
* HardDisk 500GB
* Android 6.0 (API 23)
* Internet

**2.5.2 Fakta Perangkat Keras Server**

* Intel Processor i3
* RAM 4GB
* HDD 256 GB

**2.5.3 Fakta Perangkat Keras *Smartphone***

* Android 6.0 (API 23)
* RAM 2 GB
* Internet

## **2.6.** **Asumsi dan Batasan Perangkat Lunak**

**2.6.1 Kebutuhan Perangkat Lunak**

* *Operating System*
* Node JS

**2.6.2 Fakta Perangkat Lunak**

* *Operating System* Linux

# **3.** **Deskripsi Rinci Perangkat Lunak**

## **3.1.** **Deskripsi Kebutuhan**

### **3.1.1.** **Kebutuhan Fungsional**

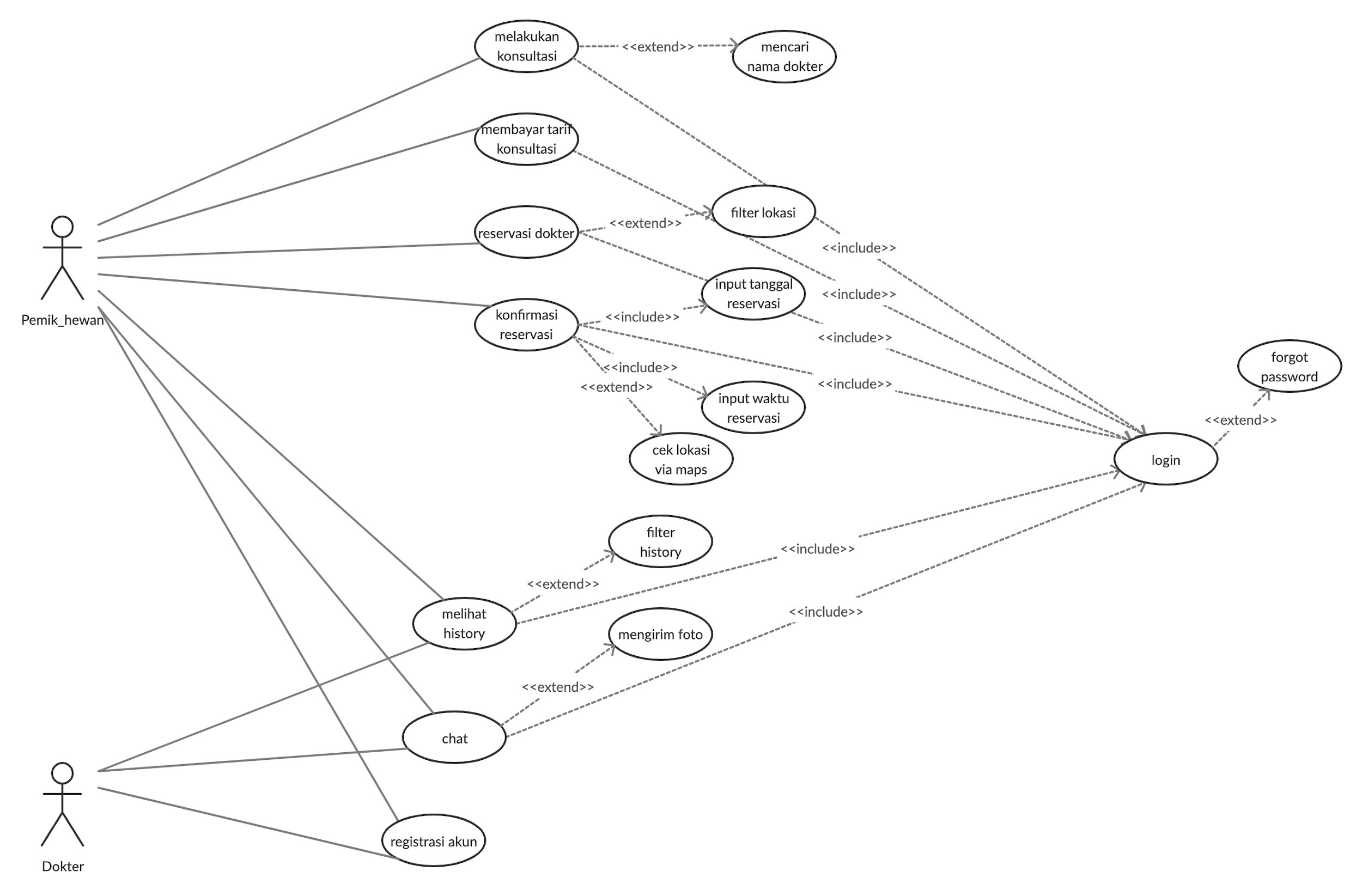
|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Kode Kebutuhan** | **Fungsi** | **Deskripsi** |
| 1. | FR-01 | Registrasi | Fungsi untuk registrasi dan menginputkannya ke dalam sistem |
| 2. | FR-02 | Konsultasi | Fungsi untuk konsultasi |
| 3. | FR-03 | Chat dokter | Fungsi untuk chat dokter |
| 4. | FR-04 | Membayar tarif konsultasi | Fungsi untuk membayar tarif konsultasi dan menginputkannya ke dalam sistem |
| 5. | FR-05 | Reservasi Dokter | Fungsi untuk Reservasi Dokter |
| 6. | FR-06 | Konfirmasi reservasi | Fungsi Konfirmasi Reservasi |
| 7. | FR-07 | Melihat History | Fungsi Melihat History |
| 8. | FR-08 | Chat dengan Pasien | Fungsi Chat dengan Pasien |
| 9. | FR-09 | Login | Fungsi Login |

### **3.1.2.** **Kebutuhan Non-Fungsional**

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Quality** | **Kode Kebutuhan** | **Deskripsi** |
| 1. | Security  Safety | NFR-01 | Memastikan Data User Terenkripsi dengan baik |
| 2. | User Interface | NFR-02 | Memastikan User Interface ter-deliver dengan baik |
| 3. |  |  |  |
| 4. |  |  |  |

## **3.2** **Pemodelan Analisis**

### **3.2.2.** **Usecase Diagram**



### **3.2.2.1. Usecase Skenario**

1)

- 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.

|  |  |
| --- | --- |
| **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 |  |

2)

- Nama usecase : 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.

|  |  |
| --- | --- |
| **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)

- Nama Use case : Melakukan konsultasi

- Aktor : pemilik\_hewan

- Pre-condition : Aktor ingin melakukan konsultasi dengan dokter, aktor belum melakukan konsultasi.

- Post-condition : Aktor telah melakukan konsultasi dengan dokter.

- Deskirpsi : Aktor melakukan konsultasi dengan memulai percakapan dengan dokter.

|  |  |
| --- | --- |
| **Pemilik\_hewan** | **System** |
| 1) Memilih menu konsultasi |  |
| 2) Mencari nama dokter |  |
| 3) Memilih dokter |  |
|  | 4) Menerima permintaan aktor, dan mengalihkan aktor ke tahap percakapan. |

4)

- 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 telah memulai dan melakukan percakapan.

- 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.

|  |  |
| --- | --- |
| **Pemilik\_hewan** | **System** |
| 1) Membuka fitur chat |  |
| 2) Mengirimkan pesan/foto kepada dokter |  |
| 3) Menyelesaikan percakapan |  |
|  | 4) Mengalihkan aktor ke tahap selanjutnya |

5)

- 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

|  |  |
| --- | --- |
| **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. |  |

6)

- 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.

|  |  |
| --- | --- |
| **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. |

7)

- Nama use case : Konfirmasi Reservasi

- 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.

|  |  |
| --- | --- |
| **Pemilik\_hewan** | **System** |
| 1) Membuka halaman konfirmasi reservasi |  |
|  | 2) Menampilkan info klinik sesuai pilihan aktor |
|  | 3) Menampilkan form tanggal, dan waktu yang wajib diisi oleh aktor. |
|  | 4) Menampilkan form tanggal, dan waktu yang wajib diisi oleh aktor. |
| 5) Mengatur jadwal dengan mengisi form tanggal dan waktu yang telah disediakan. |  |
| 6) Menyelesaikan tahap reservasi dengan menekan tombol “Reservasi”. |  |

8)

- Nama use case : Melihat history

- Aktor : Pemilik\_hewan, dan Dokter

- Pre-condition : Aktor ingin memeriksa history transaksi, aktor belum melakukan pengecekan.

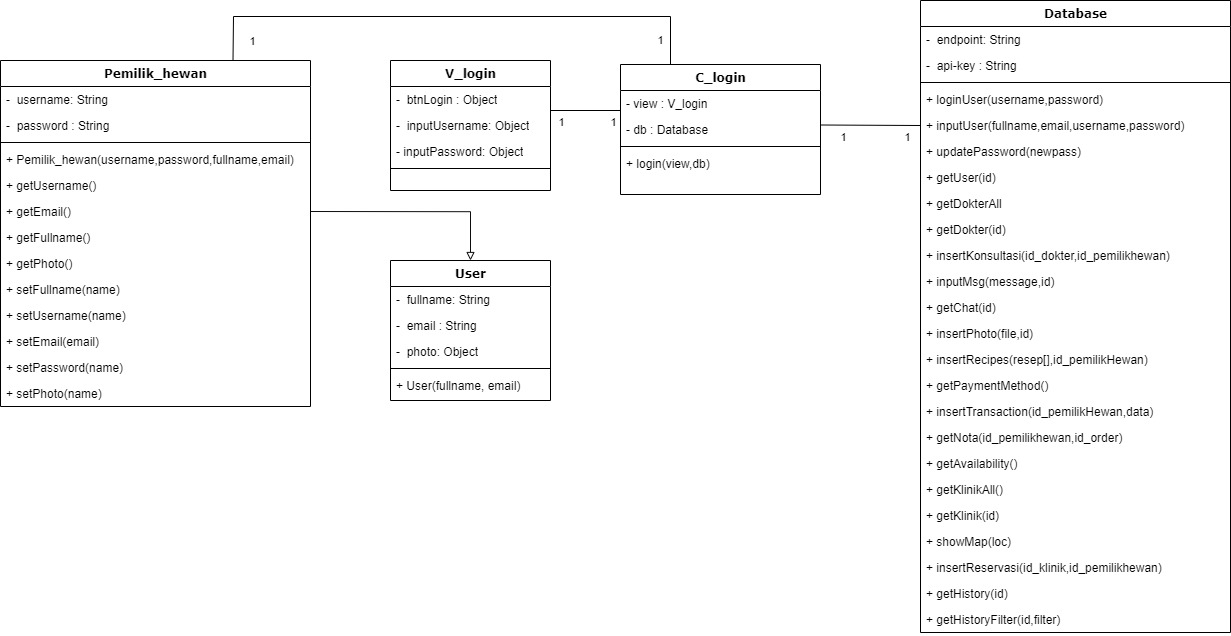
- Post-condition: Aktor telah memeriksa history transaksi.

- Deskripsi: Aktor melakukan pengecekan pada menu history transaksi yang telah dilakukan selama menggunakan aplikasi.

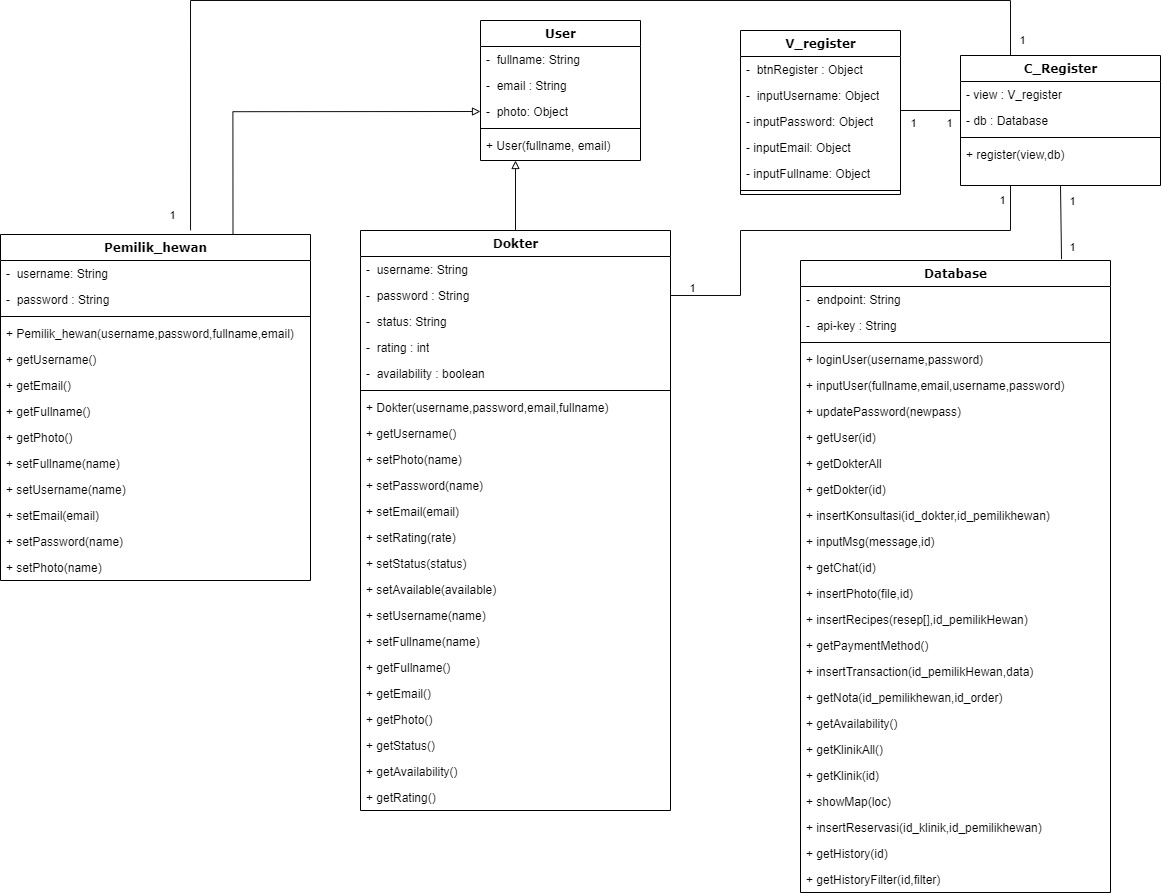
|  |  |
| --- | --- |
| **Pemilik\_hewan** | **System** |
| 1) Membuka fitur history |  |
|  | 2) Menampilkan history aktor |
| 3) Aktor dapat memfilter history berdasarkan 2 tipe: ongoing, dan completed. |  |
|  | 4) Jika aktor memilih untuk memfilter history maka:   * Jika aktor memilih ongoing maka 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. |  |

## **3.3.** **Class Diagram**

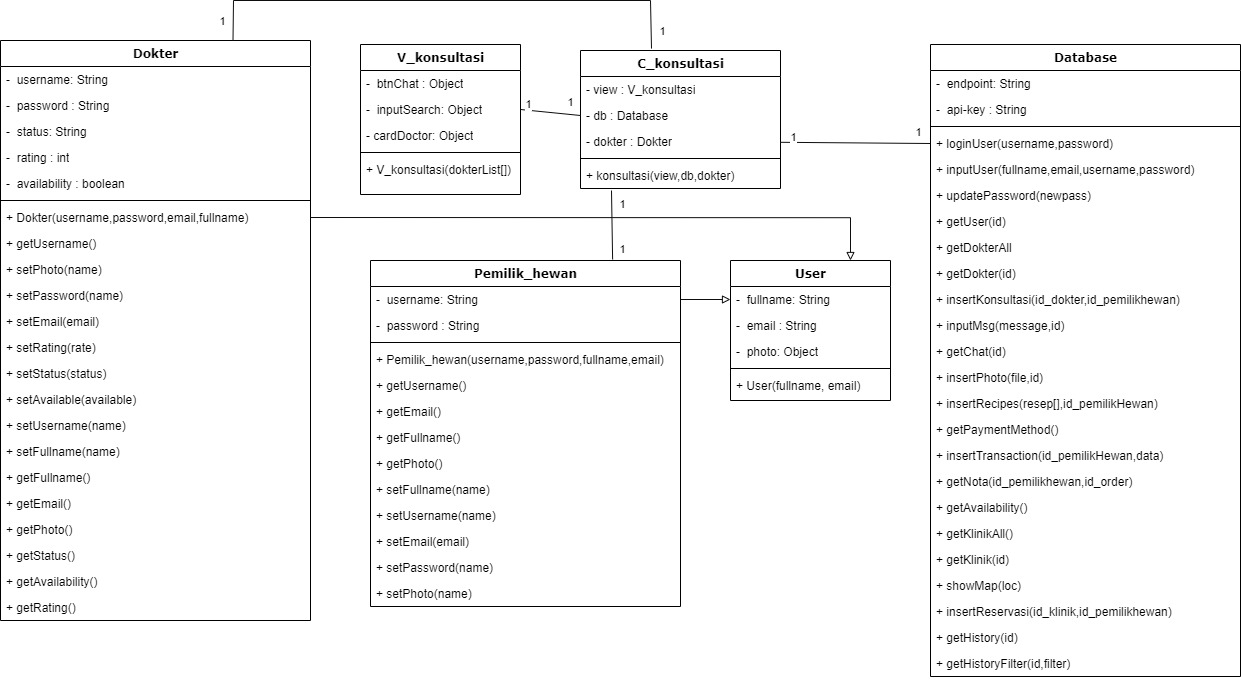
3.3.1. Class Diagram Login



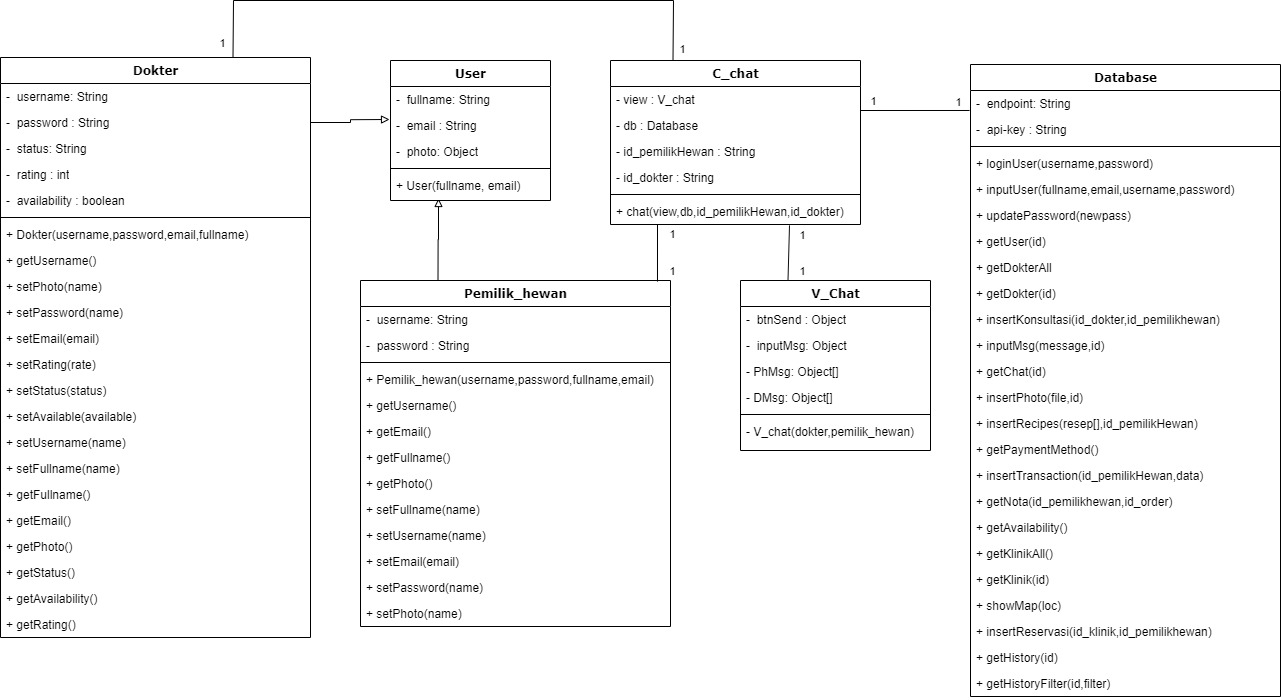
3.3.2. Class Diagram Register



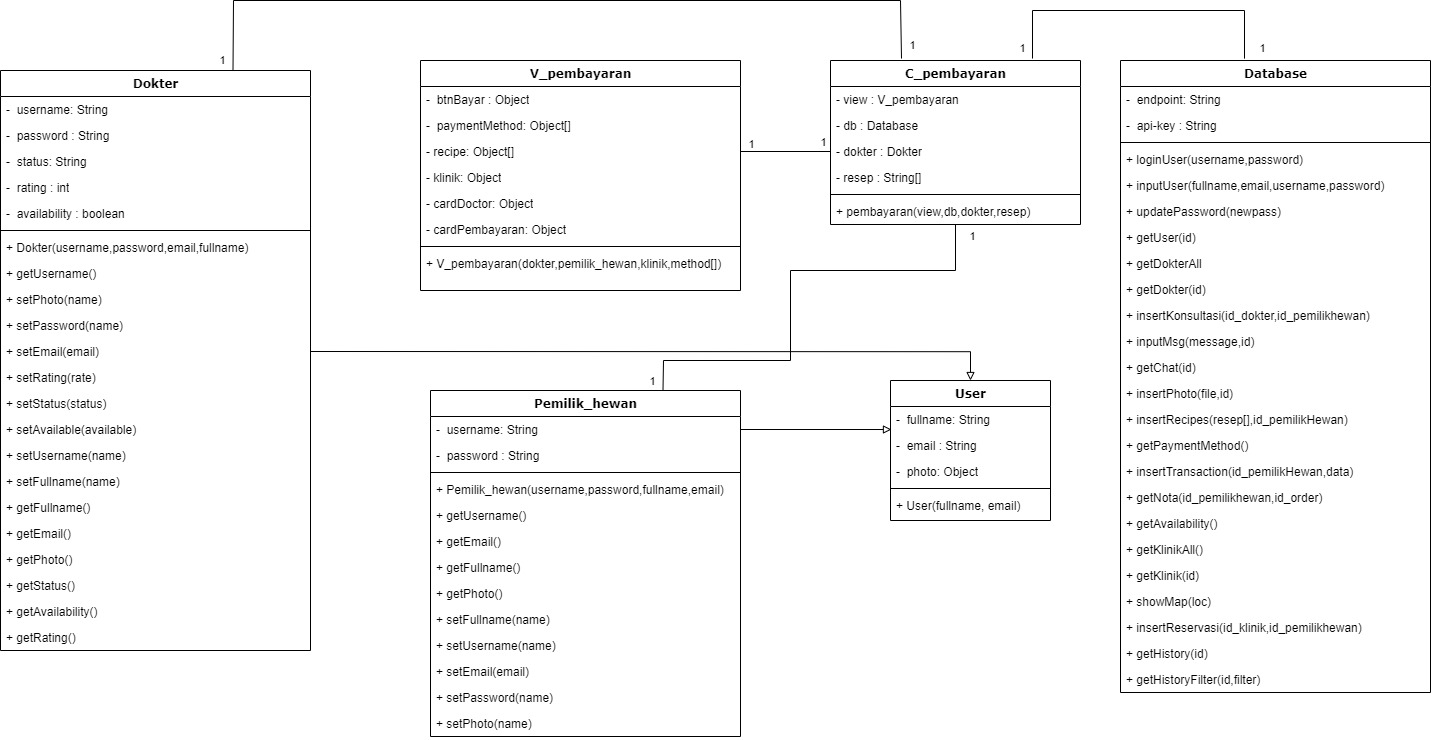
3.3.3. Class Diagram Konsultasi



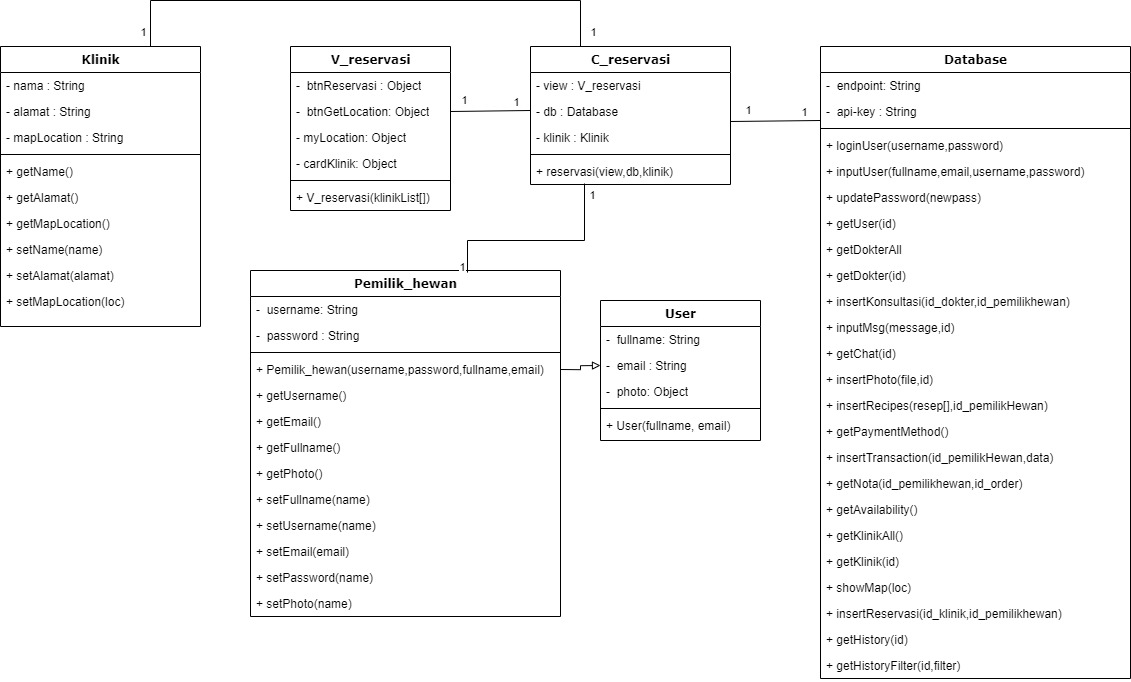
3.3.4. Class Diagram Chat Dokter



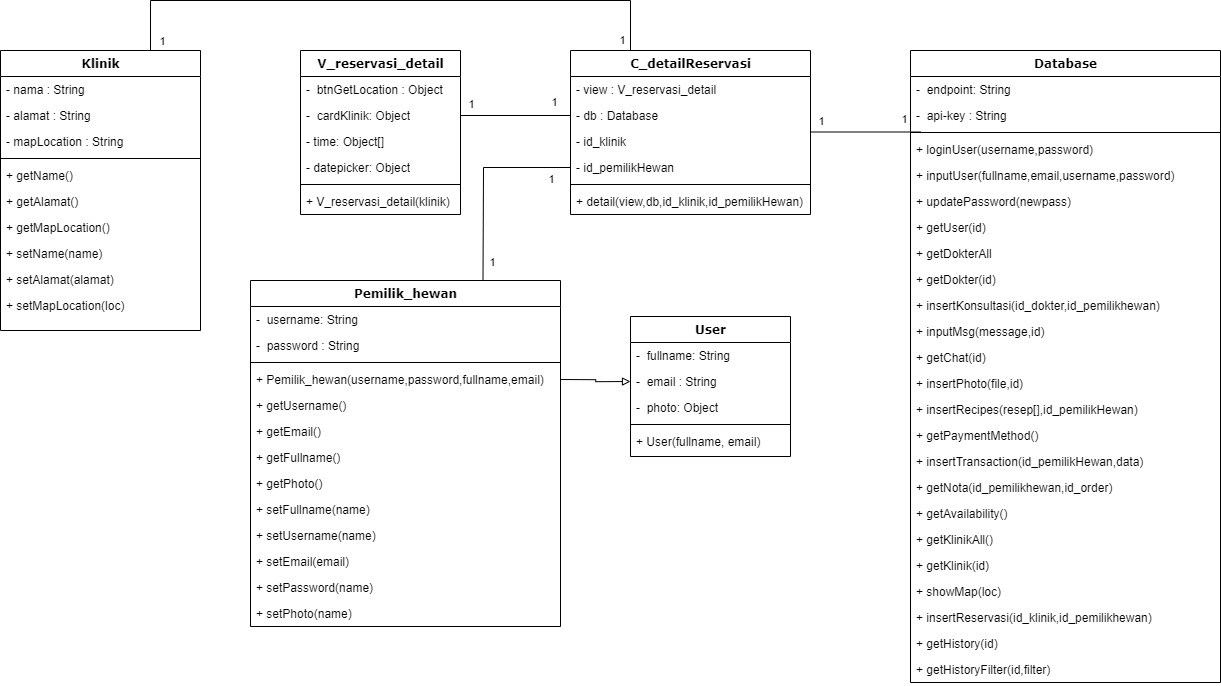
3.3.5. Class Diagram Pembayaran Tarif



3.3.6. Class Diagram Reservasi



3.3.7. Class Diagram Konfirmasi Reservasi



3.3.8. Class Diagram History

