**DPPL-02**

DESKRIPSI PERANCANGAN PERANGKAT LUNAK

Sistem Informasi Klinik

untuk:

Klinik Pedesaan

Dipersiapkan oleh:

Muhammad Firdaus Basyuni

Hanif Anam Maulana

Ayatullah Naufal

Youga Pratama

Hafizh Fairussufi

Muhammad Bagus Satrio

Program Studi Teknik Informatika

Fakultas Informatika – Telkom University

2018

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| TU-logo-primer-memusat | **Prodi S1- Teknik Informatika**  **Universitas Telkom** | Nomor Dokumen | | Halaman |
| *DPPL-02* | | *33* |
| Revisi | *04* | *02-05-2018 tanggal>* |

DAFTAR PERUBAHAN

|  |  |
| --- | --- |
| Revisi | Deskripsi |
| A | Menambahkan Matriks Kerunutan  Menambahkan Representasi kelas  Menambahkan algoritma |
| B | Mengubah Sequence Diagram  Mengubah Robustness Diagram  Mengubah Diagram Kelas  Menambahkan Query |
| C | Mengubah Sequence Diagram  Mengubah Algoritma dan Query |
| 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

1. Pendahuluan 5

1.1 Tujuan Penulisan Dokumen 5

1.2 Lingkup Masalah 5

1.3 Definisi dan Istilah 5

1.4 Referensi 5

1.5 Sistematika Dokumen 5

2 Deskripsi Perancangan Global 6

2.1 Rancangan Lingkungan Implementasi 6

2.2 Deskripsi Arsitektural 6

2.3 Deskripsi Komponen 7

3 Perancangan Rinci 7

3.1 Realisasi Use Case 7

3.1.1.1 USE CASE #2 <Pendaftaran Pasien> 8

3.1.1.2 USE CASE #3 <Pemeriksaan Kesehatan> 8

3.1.1.3 USE CASE #4 <Tindakan Medis> 10

3.1.1.4 USE CASE #5 <Data Rekam Medis> 10

3.1.1.5 USE CASE #6 <Pemberian Resep> 12

3.1.1.6 USE CASE #7 <Pengambilan Obat> 13

3.1.1.7 USE CASE #8 <Pembayaran Pasien> 14

3.1.1.8 Identifikasi Kelas 14

3.1.1.9 Diagram Kelas 16

3.1.1.10 Sequence Diagram 17

3.2 Perancangan Antarmuka 20

3.3 Perancangan Detil Kelas. 26

3.3.1 Kelas 26

3.3.2 Kelas 29

3.4 Diagram Kelas Keseluruhan 29

3.5 Perancangan Algoritma dan Query 30

3.6 Perancangan Representasi Persistensi Kelas 32

4 Matriks Kerunutan (Requirement Traceability Matrix) 33

# 1. Pendahuluan

## Tujuan Penulisan Dokumen

Dokumen Deskripsi Perancangan Perangkat lunak ini ditujukan untuk mempermudah kegiatan dalam mengelola data pada sebuah Klinik serta untuk memenuhi tugas mata kuliah Analisis dan Pengujian Peranngkat Lunak. Dokumen ini dijadikan sebagai pengolahan data Pasien beserta Dokter, dan Admin di Klinik tersebut. Dokumen ini juga dijadikan sebagai acuan bagi pengembang sebagai pembuatan perangkat lunak dan bagi user atau penggunaka digunakan untuk segala sesuatu kebutuhan pada Kllinik tersebut. Aplikasi Klinik ini juga berfungsi untuk mengolah data menjadi lebih teratur dengan data yang akurat.

## Lingkup Masalah

Sistem Informasi Klinik merupakan sebuah aplikasi yang mengolah data Pasien yang ada di Klinik tersebut serta data para pegawai (Dokter, Admin). Lingkup Masalahnnya adalah untuk meningkatkan kemajuan teknologi di pedesaan, dan menjadikan pengolahan data lebih baik menggunakan aplikasi. Pengguna hanya satu yaitu Admin, yang berhak untuk menginput, mengubah, ataupun mengupdate data di aplikasi Sistem Informasi Klinik ini.

Batasan masalah:

1. Semua data di Aplikasi dikelola Admin
2. Pasien melakukan pendaftaran sendiri atau Self-service
3. Aplikasi ini hanya berisi segala macam data Pasien, Dokter, Admin
4. Aplikasi ini ditujukan untuk daerah Pedesaan

## Definisi dan Istilah

Definisi dan singkatan yang digunakan dalam dokumen ini adalah:

|  |  |  |
| --- | --- | --- |
| **NO** | **Istilah** | **Deskripsi** |
| 1 | Mysql | MySQL adalah sebuah perangkat lunak system manajemen basis data SQL (DBMS) yang multithread, dan multi-user. |
| 2 | Informasi | Sekumpulan data yang diorganisir atau diolah dengan cara tertentu sehingga mempunyai arti. Data yang diolah menjadi sesuatu akan berguna bagi sang penerima informasi tersebut yang memberikan keterangan. |
| 3 | SKPL | Spesifikasi Kebutuhan Perangkat Lunak |
| 4 | DPPL | Dokumen Perancangan Perangkat Lunak |

## Referensi

Dokumen yang digunakan dalam penulisan dokumen ini adalah:

SKLP 002 revisi 4

## Sistematika Dokumen

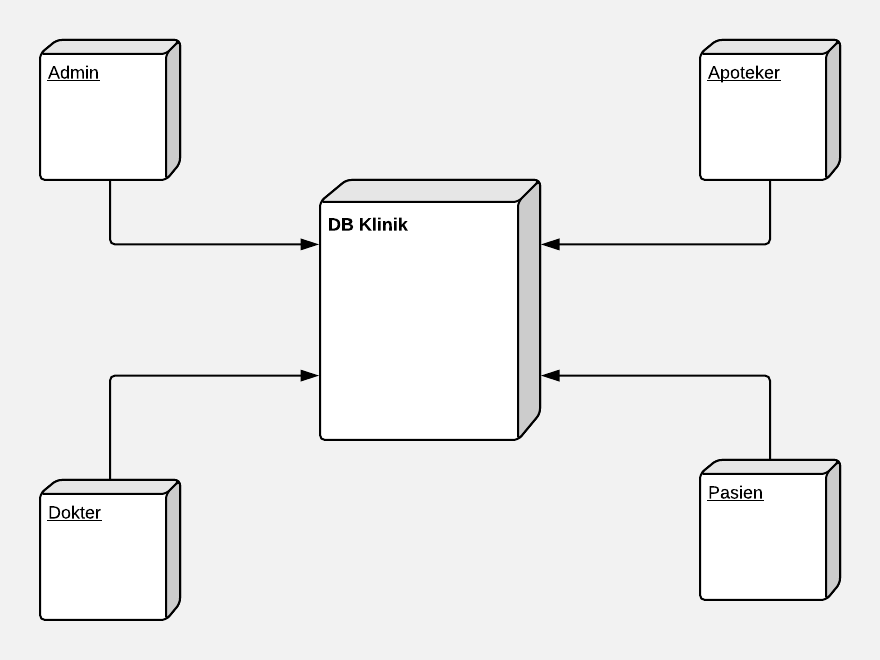
Dokumen Deskripsi Perancangan Perangkat Lunak ini berisi deskripsi kebutuhan untuk sebuah klinik. Dokumen DPPL ini dikelompokkan menjadi tiga bagian, yaitu:

1. Bab 1 Pendahuluan
2. Bab 2 Deskripsi Perancangan Global
3. Bab 3 Deskripsi Perancangan Rinci
4. Bab 4 Matrik Ketenurutan

# Deskripsi Perancangan Global

## Rancangan Lingkungan Implementasi

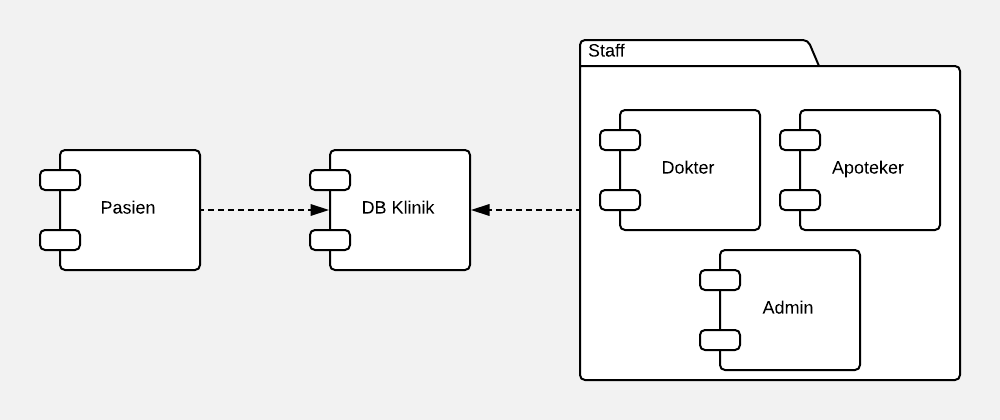
|  |  |
| --- | --- |
| **Sistem Informasi Klinik** | **Spesifikasi** |
| Sistem Operasi | Windows 10 |
| DBMS | Mysql |
| Bahasa Pemrograman | Java |
| Development Tools | NetBeans |

Deployment diagram

MySQL DB

## Deskripsi Arsitektural

*Perangkat lunak ini dibangun sebagai penghubung antara detil informasi klinik (seperti data rekam medis pasien, jadwal praktek dokter, dan lain sebagainya) dengan menu utama yang merupakan fungsional system. Didalam menu utama terdapat beberapa komponen yang bergantung padanya, yaitu Data Pasien, Dokter, Admin. Gambar arsitektur Sistem Informasi Klinik yang akan dibangun sebagai berikut (UML Component Diagram):*



## Deskripsi Komponen

Tabel daftar modul, yang mengacu pada Diagram Komponen

Tabel Daftar Modul sebagai berikut:

| **No** | **Nama Komponen** | **Keterangan / Deskripsi Detil** |
| --- | --- | --- |
|  | Pasien | Modul ini berisi segala data Pasien |
|  | DB Klinik | Modul ini berisi untuk menyimpan data ke database dari Pasien, Dokter, dan Admin |
|  | Dokter | Modul ini berisi segala data Dokter |
|  | Admin | Modul ini berisi segala data Admin |

# Perancangan Rinci

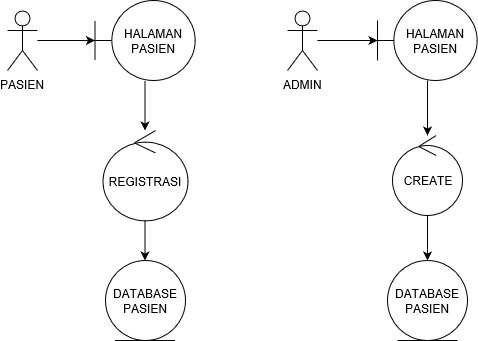
## Realisasi Use Case

Berisi TABEL USE CASE sebagai berikut

|  |  |  |  |
| --- | --- | --- | --- |
| No | Nama UseCase | Deskripi UseCase | Requirement Terkait |
| 2. | Pendaftaran Pasien | Use case yang digunakan Pasien untuk mengisi formulir pendaftaran yang telah dibuat oleh Admin (self-service) dan dikonfirmasi kembali oleh admin | Admin, Pasien. |
| 5. | Tindakan Medis | Use case yang digunakan Dokter akan memberi data Pasien kepada Admin | Dokter, Admin |
| 8. | Pembayaran Pasien | Use case yang digunakan Pasien untuk melakukan pembayaran yang dibantu oleh Admin. | Pasien, Admin. |
|  |  |  |  |

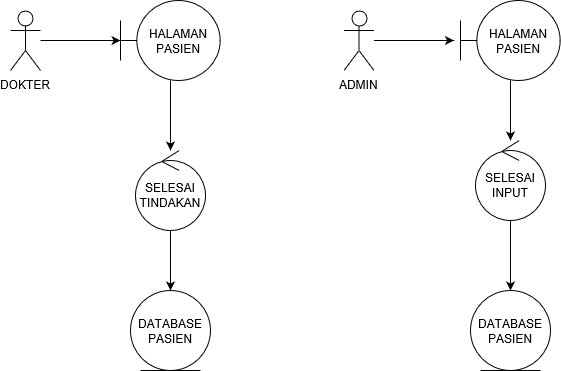
#### 

#### USE CASE #1 <Pendaftaran Pasien>



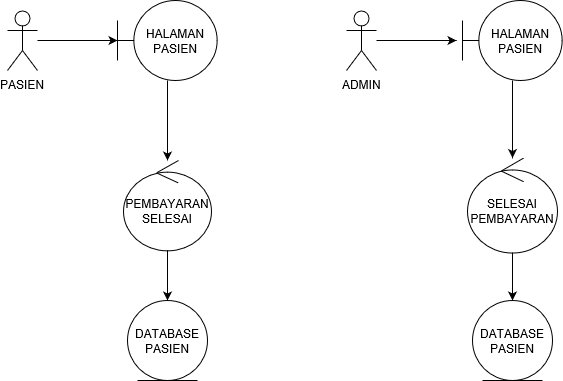
|  |  |  |
| --- | --- | --- |
| Use case | Pendaftaran | |
| Actor | Pasien, System | |
| Pre-Condition | Pasien melakukan pendfatran dengan self-service | |
| Post-Condition | Pasien mendapatkan kartu berobat dari admin | |
| Description |  | |
| **SYSTEM** | **ADMIN** | **PASIEN** |
|  |  | Melakukan pendaftaran sendiri dengan komputer yang telah disediakan berikut halamannya (Pasien hanya mengisi form pendaftaran) |
|  |  | Meng-klik tombol “regis” |
| System berhasil menyimpan data pasien |  |  |
|  | Admin mengkonfirmasi kembali meng-klik tombol “create” dan memberikan kartu berobat |  |

#### USE CASE #2 <Tindakan Medis>



|  |  |  |
| --- | --- | --- |
| Use case | Tindakan Medis | |
| Actor | Dokter, Pasien, System | |
| Pre-Condition | Pasien masuk dengan terdapat nomor ID pasien di layar Dokter | |
| Post-Condition | Data rekam medis diberikan ke admin untuk diinputkan kedalam system | |
| Description |  | |
| **SYSTEM** | **DOKTER** | **ADMIN** |
| Memperlihatkan ID pasien beserta data diri |  |  |
|  | Menyelesaikan tindakan dengan mengklik tombol “Selesai tindakan medis” |  |
|  |  | Menginputkan data pasien berikut resep yang diberi dokter |
|  |  | Meng-klik tombol “selesai input data” |

#### USE CASE #3 <Pembayaran Pasien>



|  |  |  |
| --- | --- | --- |
| Use case | Pembayaran | |
| Actor | Admin, Pasien, System | |
| Pre-Condition | Admin melihat data resep dari dokter | |
| Post-Condition | Pasien mendapatkan Obat dan bukti pembayaran | |
| Description |  | |
| **SYSTEM** | **ADMIN** | **PASIEN** |
| Memperlihatkan Data resep pasien (yang termasuk didalam data tindakan medis) |  |  |
|  | Admin melihat data resep tersebut, lalu menghitung total biaya yang dikeluarkan serta memberi obat dan bukti pembayaran |  |
|  | Meng-klik tombol “Selesai Pembayaran” |  |
|  |  | Pasien menerima bukti pembayaran beserta obat dan meng-klik tombol “Pembayaran Selesai” |

#### Identifikasi Kelas

*Identifikasi kelas yang terkait dengan use case tersebut.*

*Kelas pada tahap perancangan berbeda dengan kelas pada tahap analisis.*

*Gunakan tabel di bawah:*

**TABEL KELAS PERANCANGAN DATABASE ADMINISTRATIF**

|  |  |  |
| --- | --- | --- |
| **No** | **Nama Kelas Perancangan** | **Tipe Kelas** |
| 1. | Halaman Login | Model |
| 2. | c\_admin | Controller |
| 3. | Database Pasien | View |
| 4. | Database Obat | View |
| 5. | Database Transaksi | View |

**TABEL KELAS PERANCANGAN PENDAFTARAN PASIEN**

|  |  |  |
| --- | --- | --- |
| **No** | **Nama Kelas Perancangan** | **Tipe Kelas** |
| 1. | Halaman Pasien | Model |
| 2. | c\_pasien | Controller |
| 3. | Database Pasien | View |

**TABEL KELAS PERANCANGAN PEMERIKSAAN KESEHATAN**

|  |  |  |
| --- | --- | --- |
| **No** | **Nama Kelas Perancangan** | **Tipe Kelas** |
| 1. | Halaman Pasien | Model |
| 2. | c\_pasien | Controller |
| 3. | Database Pasien | View |

**TABEL KELAS PERANCANGAN TINDAKAN MEDIS**

|  |  |  |
| --- | --- | --- |
| **No** | **Nama Kelas Perancangan** | **Tipe Kelas** |
| 1. | Halaman Pasien | Model |
| 2. | c\_pasien | Controller |
| 3. | Database Pasien | View |

**TABEL KELAS PERANCANGAN DATA REKAM MEDIS**

|  |  |  |
| --- | --- | --- |
| **No** | **Nama Kelas Perancangan** | **Tipe Kelas** |
| 1. | Halaman Pasien | Model |
| 2. | c\_pasien | Controller |
| 3. | Database Pasien | View |

**TABEL KELAS PERANCANGAN PEMBERIAN RESEP**

|  |  |  |
| --- | --- | --- |
| **No** | **Nama Kelas Perancangan** | **Tipe Kelas** |
| 1. | Halaman Pasien | Model |
| 2. | c\_pasien | Controller |
| 3. | Database Pasien | View |

**TABEL KELAS PERANCANGAN PENGAMBILAN OBAT**

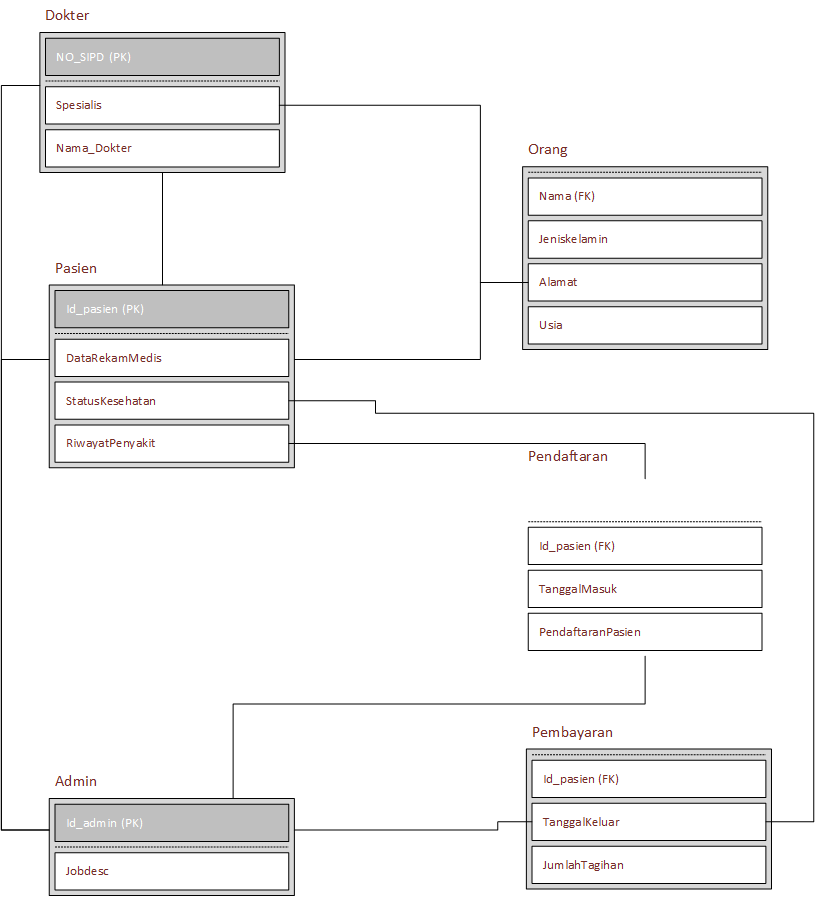
|  |  |  |
| --- | --- | --- |
| **No** | **Nama Kelas Perancangan** | **Tipe Kelas** |
| 1. | Halaman obat | Model |
| 2. | c\_obat | Controller |
| 3. | Database obat | View |

**TABEL KELAS PERANCANGAN PEMBAYARAN PASIEN**

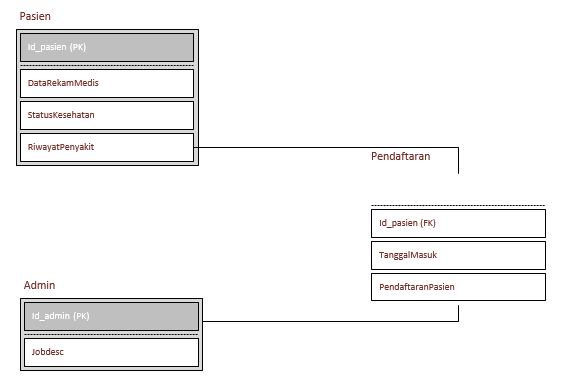
|  |  |  |
| --- | --- | --- |
| **No** | **Nama Kelas Perancangan** | **Tipe Kelas** |
| 1. | Halaman Pasien | Model |
| 2. | c\_pasien | Controller |
| 3. | Database Pasien | View |

|  |  |  |
| --- | --- | --- |
| **No** | **Nama Kelas Perancangan** | **Tipe Kelas** |
| 1. | Halaman Transaksi | Model |
| 2. | c\_pembayaran | Controller |
| 3. | DatabaseTransaksi | View |

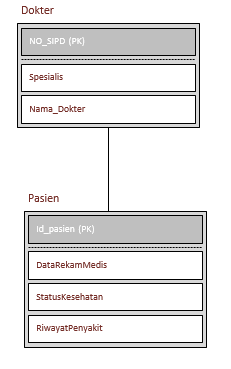
#### Diagram Kelas



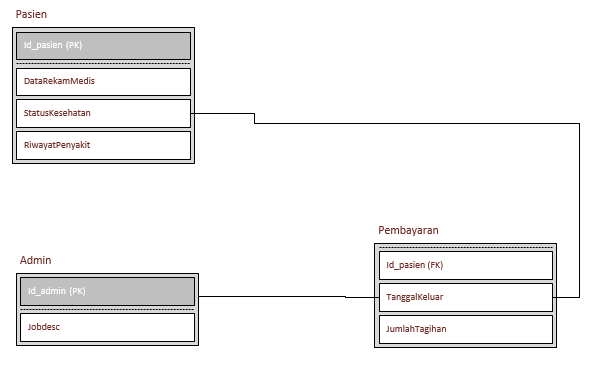
##### Pendaftaran



##### Tindakan Medis



##### Pembayaran Pasien



#### Sequence Diagram

*Sequence Diagram : Pendaftaran Pasien*

*Aktor : Pasien*

*A screenshot of a cell phone

Description generated with high confidence*

*Aktor : Admin*

*A screenshot of a cell phone

Description generated with high confidence*

*Sequnce Diagram : Tindakan Medis*

*Aktor : Dokter*

*A close up of a logo

Description generated with very high confidence*

*Aktor : Admin*

*A screenshot of a cell phone

Description generated with high confidence*

*Sequence Diagram : Pembayaran*

A screenshot of a cell phone

Description generated with very high confidence

## Perancangan Antarmuka

**TABEL IDENTIFIKASI ANTARMUKA / LAYAR/ PAGE**

|  |  |  |
| --- | --- | --- |
| **ID. LAYAR** | **NAMA LAYAR** | **DESKRIPSI** |
| *001* | *Pendaftaran* | *Untuk melakukan pendaftaran yang dilakukan oleh pasien.* |
| *002* | *Tindakan Medis* | *Unuk mencatat tindakan medis yang telah dilakukan oleh dokter ke pasien melalui database.* |
| *003* | *Pembyaran Pasien* | *Untuk melakukan pembayaran yang dilakukn oleh pasien kepada dmin.* |

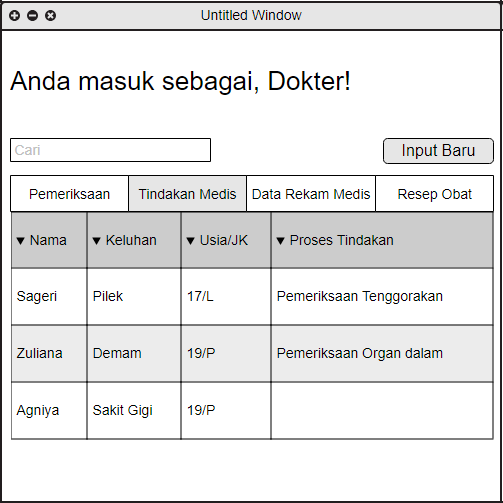
*Antarmuka Pendaftaran:*

A screenshot of a social media post

Description generated with very high confidence

Pasien

| **Id\_Objek** | **JENIS** | **LABEL\*** | **Keterangan\*\*** |
| --- | --- | --- | --- |
| *IN-1-0* | Textbox | Id Pasien | Menerima inputan ID Pasien |
| *IN-2-0* | Button | Submit | Melakukan algoritma validasi() |
| *REG-1-0* | *Textbox* | *Masukan Nama* | Menerima inputan Nama |
| *REG-1-1* | *Textbox* | *Masukan Alamat* | Menerima inputan Alamat |
| *REG-1-2* | *Textbox* | *Masukan Usia dan Jenis Kelamin* | Menerima inputan Usia dan Jenis Kelamiin |
| *REG-1-3* | *Textbox* | *Masukan Id Pasien* | Menerima inputan Id Pasien |
| *REG-2-0* | *Button* | *Registrasi* | Melakukan algoritma registrasi() |

Antarmuka Tindakan Medis

Admin

| **Id\_Objek** | **JENIS** | **LABEL\*** | **Keterangan\*\*** |
| --- | --- | --- | --- |
| *SC-1-0* | Search  box | Cari | Melakukan pencarian sesuai dengan inputan yang diterima |
| *NEW-2-0* | Button | Input Baru | Melakukan algoritma newTindakan() |
| *TAB-1-0* | *Table* | *Pemeriksaan* | Menampilkan isi tabel yang berkaitan dengan Tindakan Pasien |

*Antarmuka Pembayaran:*

A screenshot of a social media post

Description generated with very high confidence

| **Id\_Objek** | **JENIS** | **LABEL\*** | **Keterangan\*\*** |
| --- | --- | --- | --- |
| *IN-1-0* | Textbox | Id Pasien | Menerima inputan ID Pasien |
| *IN-2-0* | Button | Submit | Melakukan algoritma validasi() |
| *REG-1-0* | *Textbox* | *Masukan Nama* | Menerima inputan Nama |
| *REG-1-1* | *Textbox* | *Total Tagihan* | Menerima inputan Alamat |
| *REG-1-2* | *Textbox* | *Detil Tagihan* | Menerima inputan Usia dan Jenis Kelamiin |
| *REG-1-3* | *Textbox* | *Status Pembayaran* | Menerima inputan Id Pasien |
| *REG-2-0* | *Button* | *Ubah Status* | Melakukan algoritma registrasi() |

## Perancangan Detil Kelas.

**TABEL KELAS :**

|  |  |  |
| --- | --- | --- |
| **ID Kelas** | **Nama Kelas Perancangan** | **Nama Kelas Analisis Terkait (Model Domain)** |
|  | Admin | Admin |
|  | Dokter | Dokter |
|  | Pasien | Pasien |
|  | Orang | Orang |
|  | JadwalDokter | Dokter |
|  | Aplikasi | Aplikasi |

### Kelas

*Nama Kelas : Akun\_admin*

|  |  |  |
| --- | --- | --- |
| **Nama Operasi** | **Visibility**  **(private, public)** | **Keterangan** |
| Akun() | Private | Digunakan untuk login kedalam aplikasi |
| setUsername() | Private | Digunakan untuk mendefinisikan username |
| setPassword() | Private | Digunakan untuk mendefinisikan password |
| setNama() | Private | Digunakan untuk mendefinisikan Nama |
| setAlamat() | Private | Digunakan untuk mendefinisikan Alamat |
| setEmail() | Private | Digunakan untuk mendefinisikan Email |
| setNoHp() | Private | Digunakan untuk mendefinisikan NoHp |
| getUsername() | Private | Digunakan untuk mengambil data Username |
| getPassword() | Private | Digunakan untuk mengambil data Password |
| getNama() | Private | Digunakan untuk mengambil data Nama |
| getAlamat() | Private | Digunakan untuk mengambil data Alamat |
| getEmail() | Private | Digunakan untuk mengambil data Email |
| getNoHp | Private | Digunakan untuk mengambil data NoHP |
| **Nama Atribut** | **Visibility**  **(private, public)** | **Tipe** |
| Username | Private | Varchar2 |
| Password | Private | Varchar2 |
| Nama | Private | Varchar2 |

*Kelas Domain : Dokter*

|  |  |  |
| --- | --- | --- |
| **Nama Operasi** | **Visibility**  **(private, public)** | **Keterangan** |
| Dokter() | Private | Untuk menjalankan aksi sebagai dokter |
| setter() | Private | Digunakan untuk mendefinisikan username |
| getter() | Private | Digunakan untuk mendefinisikan password |
| **Nama Atribut** | **Visibility**  **(private, public)** | **Tipe** |
| Id\_dokter | Private | Varchar2 |
| Spesialis | Private | Varchar2 |
| Nama\_dokter | Private | Varchar2 |

*Kelas Domain : Pasien*

|  |  |  |
| --- | --- | --- |
| **Nama Operasi** | **Visibility**  **(private, public)** | **Keterangan** |
| Pasien() | Private | Untuk menjalankan aksi sebagai Pasien |
| setter() | Private | Digunakan untuk mendefinisikan fungsi Pasien |
| getter() | Private | Digunakan untuk mengambil data berkaitan dengan pasien |
| **Nama Atribut** | **Visibility**  **(private, public)** | **Tipe** |
| Id\_Pasien | Private | Varchar2 |
| RiwayatPenyakit | Private | Varchar2 |
| DataRekamMedis | Private | Varchar2 |

*Kelas Domain : Orang*

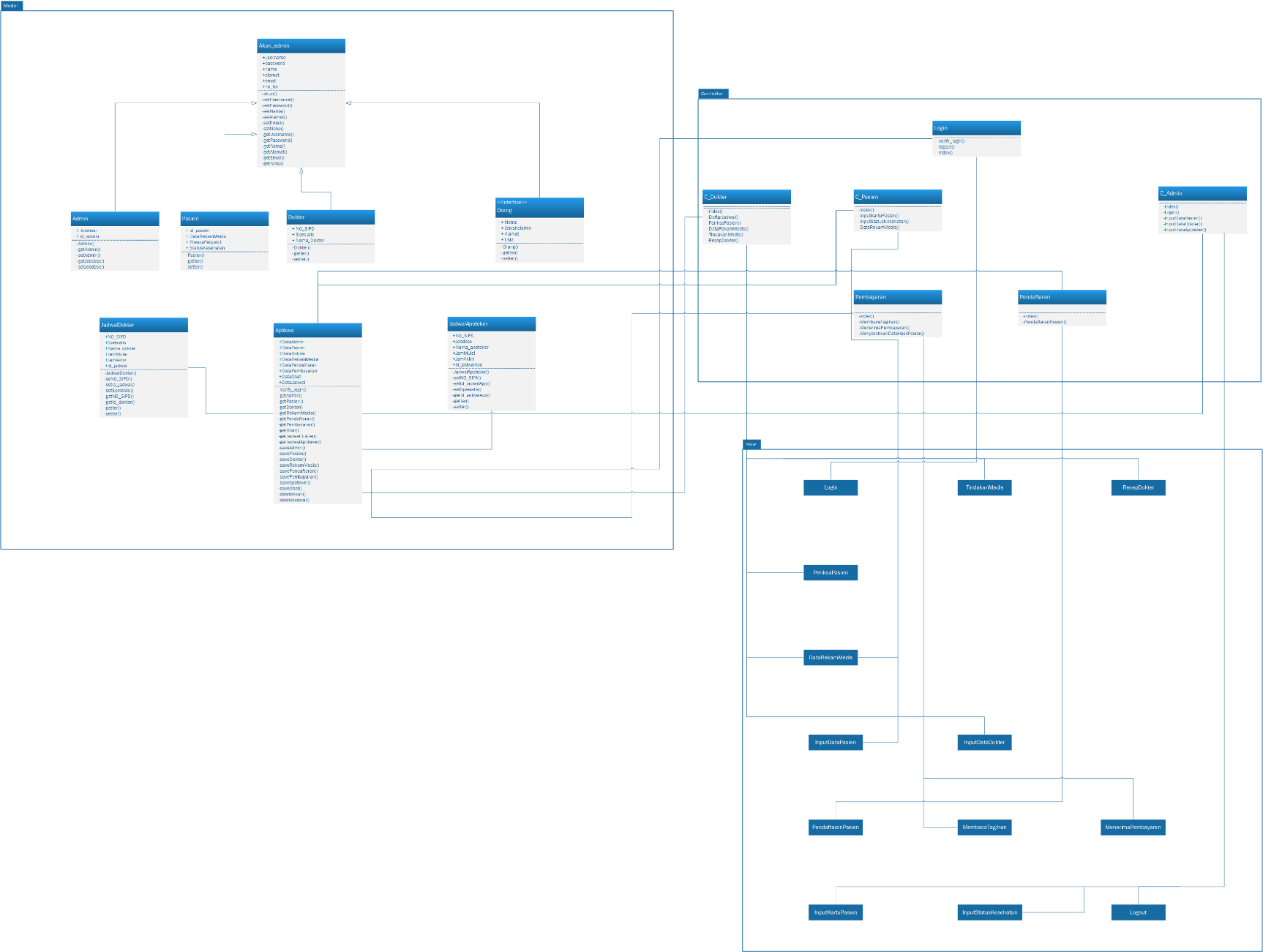
|  |  |  |
| --- | --- | --- |
| **Nama Operasi** | **Visibility**  **(private, public)** | **Keterangan** |
| setter() | Private | Digunakan untuk mendefinisikan nilai pada fungsi Orang |
| getter() | Private | Digunakan untuk mengambil data berkaitan dengan Orang |
| **Nama Atribut** | **Visibility**  **(private, public)** | **Tipe** |
| Nama | Private | Varchar2 |
| Alamat | Private | Varchar2 |
| Usia | Private | Varchar2 |
| Jenis Kelamin | Private | Varchar2 |

*Kelas Domain : Jadwal Dokter*

|  |  |  |
| --- | --- | --- |
| **Nama Operasi** | **Visibility**  **(private, public)** | **Keterangan** |
| JadwalDokter() | Private | Untuk menjalankan aksi sebagai JawalDokter |
| setter() | Private | Digunakan untuk mendefinisikan fungsi JadwalDokter |
| getter() | Private | Digunakan untuk mengambil data berkaitan dengan JadwalDokter |
| **Nama Atribut** | **Visibility**  **(private, public)** | **Tipe** |
| Id\_dokter | Private | Varchar2 |
| Spesialis | Private | Varchar2 |
| Nama\_dokter | Private | Varchar2 |
| Id\_Jadwal | Private | Varchar2 |
| Jam\_Mulai | Private | Varchar2 |
| Jam\_Akhir | Private | Varchar2 |
| Ruangan Dokter | Private | Varchar2 |
| Hari Praktek | Private | Varchar2 |

### Kelas

## Diagram Kelas Keseluruhan

*Bagian ini diisi dengan gambaran diagram kelas keseluruhan.*

## Perancangan Algoritma dan Query

Algoritma dari Sistem Informasi Klinik sebagai berikut

Nama Kelas : Pendaftaran

Nama Operasi : CreateAccount

Algoritma : (Algo-001)

CreateAcc() {

SQL = “INSERT INTO **PASIEN** VALUES (‘.$nama.’,’.$id.’,’.$stats\_payment.’,’.$tindakan.’)

}

Nama Kelas : Pembayaran Pasien

Nama Operasi : StatsPayment

Algoritma : (Algo-002)

StatsPayment() {

UPDATE **PASIEN** SET **STATS\_PAYMENT = ‘**.$stats\_payment.’ WHERE ID = ‘.$id.’

}

Nama Kelas : Tindakan Medis

Nama Operasi : Tindakan

Algoritma : (Algo-003)

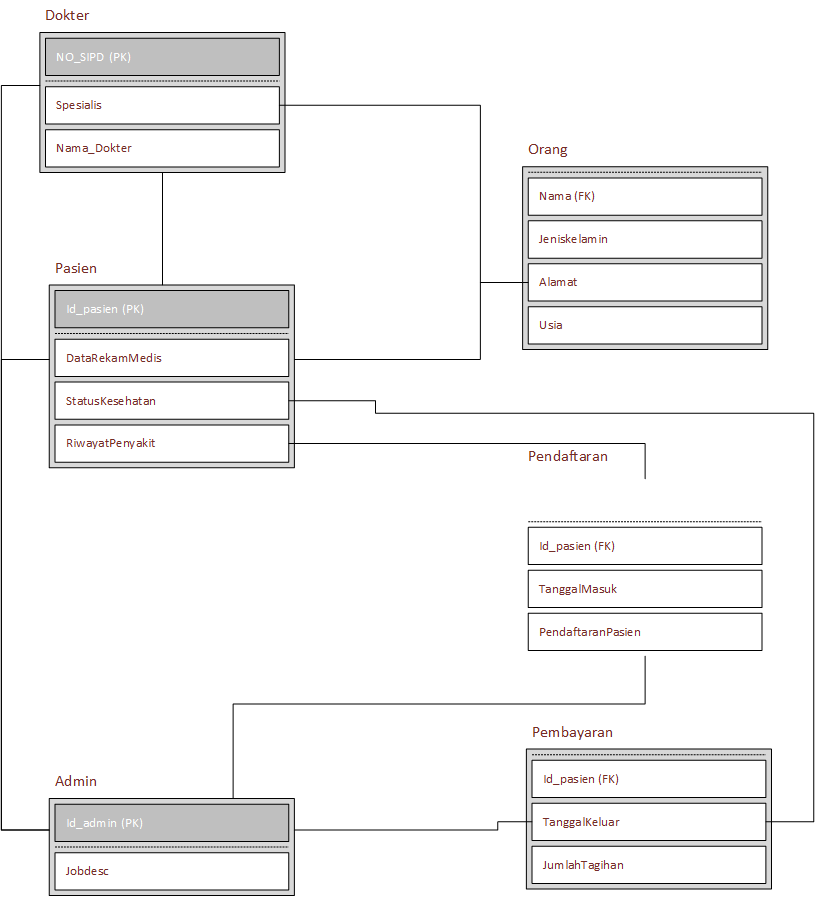
Tindakan() {

UPDATE **PASIEN** SET **TINDAKAN = ‘**.$tindakan.’ WHERE ID = ‘.$id.’

}

## Perancangan Representasi Persistensi Kelas

*Rancangan skema basisdata dan traceability-nya terhadap kelas entity.*



# Matriks Kerunutan (Requirement Traceability Matrix)

Mapping requirement, use case dengan kelas-kelas terkait

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Requirement** | **Use Case** | **Kelas** |
|  | ADM-01 ; | Pendaftaran | Admin, Aplikasi, Pasien |
|  | ADM-02 ; | Tindakan Medis | Dokter, Admin, Aplikasi |
|  | ADM-03 ; | Pembayaran | Admin, Pasien, Aplikasi |