

BAB IV. ANALISIS DAN PERANCANGAN

4.1 Analisis Masalah

Tahap ini merupakan tahapan menganalisis masalah dalam Sistem Pendeteksian Penyakit Diabetes Mellitus Secara Dini Menggunakan Metode *Case Based Reasoning*. Didalam sistem ini terdiri 3 aktor yaitu admin, pakar, dan pasien. Pasien yang mengakses sistem ini yaitu masyarakat umum yang sedang merasakan beberapa gejala tidak nyaman pada kondisi tubuhnya. Namun dengan banyaknya dokter/pakar yang tersebar di setiap daerah tidak menutup kemungkinan pasien tidak mempunyai waktu atau biaya untuk memeriksakan langsung pada pakar tersebut, apalagi pemerintah sudah menyediakan program kesehatan yang dapat membantu dalam hal pemeriksaan dan perawatan. Sehingga hal mengenai gejala yang dirasa pasien tersebut diremehkan dan tidak jarang pasien menganggapnya hanya masalah biasa dan tidak menyebutkan bahwa hal tersebut adalah penyakit, bahkan ada pula pasien yang hanya mengandalkan cara tradisional untuk mengatasi masalah tersebut. Padahal banyak penyakit yang menyerang namun dengan gejala yang sama satu sama lain, ada pula yang menunjukkan gejala yang tidak terlalu terlihat namun pada nyatanya itu merupakan sebuah penyakit yang harus diatasi oleh pakar dan merupakan hal yang sangat serius. Karena jika tidak segera diketahui dan diatasi oleh pakar dapat menyebabkan komplikasi bagi orang tersebut, tidak jarang pula akan menyebabkan kematian. Dengan adanya permasalahan tersebut, dalam era saat ini sangat bisa dibantu dan diatasi dengan adanya program komputer.

Data yang dibutuhkan untuk pembuatan sistem ini yaitu daftar gejala penyakit, daftar penyakit, dan daftar penanganan pertamanya. Data ini diperoleh dari proses wawancara kepada pakar dan studi literatur. Data yang sudah didapat nantinya akan digunakan sistem untuk mencari solusi berdasarkan metode yang digunakan. Selanjutnya untuk menentukan bobot kriteria yang diberikan dengan cara konsultasi dengan pakar untuk mengetahui prioritas masing-masing kriteria gejala yang diinginkan untuk penentuan penyakit. Setelah itu daftar gejala penyakit yang sudah tersedia masing-masing diberi

bobot. Untuk mengetahui diagnosis penyakitnya, maka pasien harus memasukan gejala yang dirasakannya. Kemudian sistem akan memprosesnya dengan cara mencari kesamaan berdasarkan gejala-gejala yang sudah ada pada masing-masing penyakit dan dengan menggunakan metode *case-based reasoning*. Metode *case-based reasoning* adalah salah satu metode untuk membangun sistem pakar dengan pengambilan keputusan dari kasus yang baru dengan berdasarkan solusi dari kasus – kasus sebelumnya. Konsep dari metode *case based reasoning* ditemukan dari ide untuk menggunakan pengalaman – pengalaman yang terdokumentasi untuk menyelesaikan masalah yang baru. Dalam metode *case-based reasoning* sendiri terdapat langkah-langkah penyelesaiannya, yaitu : *Retrieve* (memperoleh Kembali), *Reuse* (menggunakan kembali), *Revise* (peninjauan kembali), *Retain* (menyimpan). Diharapkan sistem ini dapat membantu pasien untuk mengetahui diagnosis sementara dengan diberikannya informasi persentase keyakinan dari hasil yang diberikan dan dapat memberikan informasi berupa solusi cara penanganan pertamanya sebelum pasien tersebut yakin untuk memeriksakan diri ke pakarnya langsung. Sistem ini dibuat dengan tetap dibawah pengawasan ahli/pakar.

4.2 Analisis Kebutuhan

Tahap ini merupakan tahapan awal dalam pembentukan aplikasi Sistem Pendeteksian Penyakit Diabetes Mellitus Secara Dini Menggunakan Metode *Case Based Reasoning*. Kebutuhan sistem yang diperlukan berupa memberikan diagnosis penyakit diabetes mellitus berdasarkan gejala yang dipilih, admin melakukan proses *load* data dan melakukan proses perhitungan dengan menggunakan metode *case based reasoning*. Pakar melakukan proses revisi kasus jika diperlukan untuk memperbaiki solusi pada kasus yang baru. Sedangkan pasien memilih gejala sesuai yang dirasakannya. Kebutuhan sistem terbagi menjadi dua yaitu kebutuhan sistem fungsional dan kebutuhan sistem non-fungsional.

4.2.1 Kebutuhan Fungsional

Kebutuhan fungsional adalah kebutuhan yang berisi mengenai proses-proses apa saja yang nantinya akan dilakukan oleh sistem. Dalam sistem ini terdapat 3 aktor, yaitu admin dan pakar (dokter spesialis penyakit dalam) yang berperan sebagai pengelola serta pasien yang merupakan masyarakat umum yang sedang mengalami gejala yang kurang nyaman pada tubuhnya.

- Perbedaan fungsi admin dan pakar akan dijelaskan pada tabel 4.1 sebagai berikut :

Tabel 4. 1 Perbedaan Fungsi Admin dan Pakar

Fungsi	Admin	Pakar
Dapat menambah, mengubah, dan menghapus data.	Bisa, namun tetap dalam pengawasan pakar.	Tidak Bisa
Dapat melakukan proses revisi jika pada proses <i>retrieve</i> sistem tidak dapat memberikan hasil diagnosis yang tepat.	Tidak Bisa	Bisa

- Kebutuhan Fungsional dijelaskan pada tabel 4.2 sebagai berikut :

Tabel 4. 2 Kebutuhan Fungsional

Data yang Diperlukan	Data rekam medis pasien yang terdahulu Data gejala penyakit Data penyakit Data basis kasus
Admin	Menginputkan data yang akan digunakan, seperti data gejala, data penyakit, dan data basis kasus.

Pakar	Melakukan proses revisi (<i>revise</i>) dari data kasus baru yang mempunyai nilai keyakinan dibawah 50%.
Pasien	Dapat melihat informasi mengenai penyakit diabetes mellitus, dapat melihat informasi yang terdapat pada aplikasi, dan dapat memilih gejala-gejala penyakit yang dirasakannya untuk mengetahui diagnosis sementara penyakit serta solusi penanganan pertamanya.

4.2.2 Kebutuhan Non-Fungsional

Kebutuhan non fungsional adalah kebutuhan pendukung yang dimiliki oleh sistem, yaitu :

- Kebutuhan Keamanan
Sistem ini dilengkapi dengan penggunaan *password* saat pengelola melakukan *login* guna untuk menjaga keamanan data yang terdapat didalamnya.
- Kebutuhan Informasi
Digunakan untuk menampilkan tata cara pasien dalam melakukan proses konsultasi.

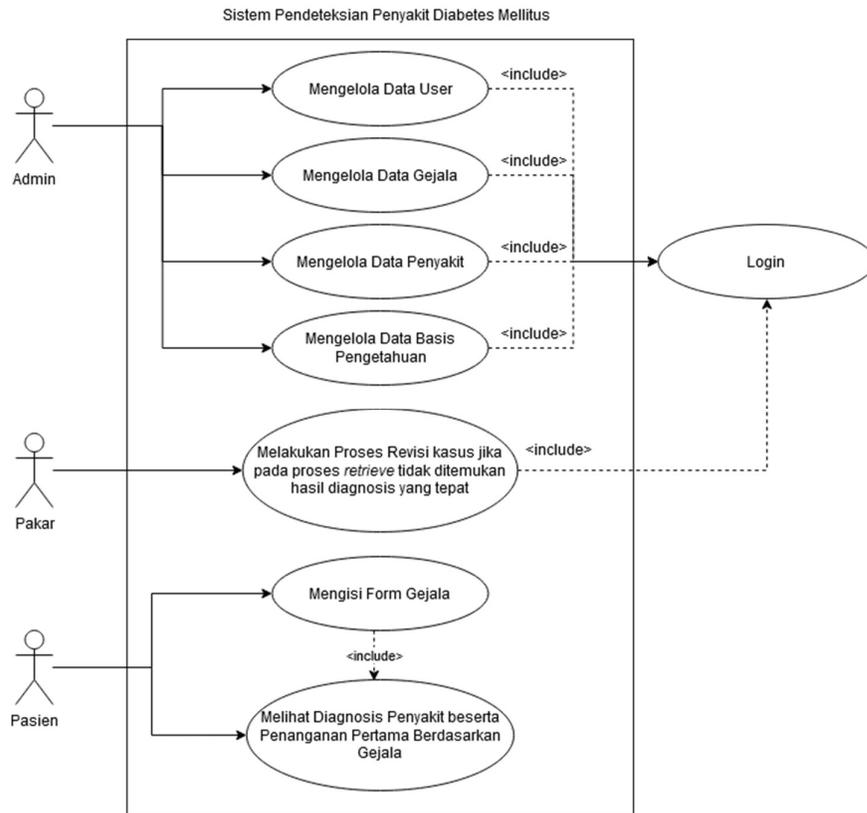
4.3 Perancangan Sistem

Perancangan adalah penjelasan dari hasil analisa kebutuhan Sistem Pendeteksian Penyakit Diabetes Mellitus Secara Dini Menggunakan Metode *Case Based Reasoning*. Tujuan perancangan sistem untuk memenuhi kebutuhan pemakaian sistem (*user*) dan memberikan gambaran yang jelas dalam pembuatan sistem.

4.3.1 Perancangan *Use Case*

Use Case Diagram merupakan teknik untuk menggambarkan secara ringkas siapa yang menggunakan sistem dan apa saja yang bisa dilakukannya. Dibawah ini merupakan desain *use case diagram* dari aplikasi Sistem

Pendeteksian Penyakit Diabetes Mellitus Secara Dini Menggunakan Metode *Case Based Reasoning*.



Gambar 4. 1 *Use Case Diagram*

a. *Skenario Use Case*

Skenario Use Case menuliskan alur jalannya sistem. Setiap skenario ditulis menurut sudut pandang aktor yang berinteraksi dengan sistem dalam berbagai cara. Alur skenario *use case* diatas memiliki 3 aktor dengan hak akses yang berbeda. Admin adalah orang yang bertanggung jawab dalam mengelola data secara keseluruhan pada sistem kecuali proses revisi. Pakar adalah seorang dokter spesialis penyakit dalam yang bertugas untuk melakukan revisi kasus, yang akan menjadi pakar pada aplikasi ini adalah dr. Achmad Zainudin Arif, SpPD yang merupakan dokter spesialis penyakit dalam di klinik Hayunanto Medical Center. Pasien adalah seseorang yang melakukan pendeteksian

penyakit pada aplikasi. Berikut adalah hak akses dari masing-masing aktor :

1. Data User

Tabel 4.3 merupakan tabel deskripsi skenario dari *use case* data gejala. Dalam tabel ini menjelaskan bagaimana skenario untuk menambahkan dan mengelola data gejala.

Tabel 4. 3 Skenario *Use Case* Data *User*

CRUD Data <i>User</i>	
Tujuan	Mengelola data <i>user</i>
Aktor	Admin
Skenario	Proses ini adalah sebuah kegiatan untuk memasukan data <i>user</i> yang baru dan menghapus data <i>user</i> yang sudah ada .
Kondisi Akhir	Sistem akan menampilkan data <i>user</i> .

2. Data Gejala

Tabel 4.4 merupakan tabel deskripsi skenario dari *use case* data gejala. Dalam tabel ini menjelaskan bagaimana skenario untuk menambahkan dan mengelola data gejala.

Tabel 4. 4 Skenario *Use Case* Data Gejala

CRUD Data Gejala	
Tujuan	Mengelola data gejala
Aktor	Admin
Skenario	Proses ini adalah sebuah kegiatan untuk memasukan data gejala yang baru, mengubah, ataupun menghapus data gejala yang sudah ada .
Kondisi Akhir	Sistem akan menampilkan data gejala.

3. Data Penyakit

Tabel 4.5 merupakan tabel deskripsi skenario dari *use case* data penyakit. Dalam tabel ini menjelaskan bagaimana skenario untuk menambahkan dan mengelola data penyakit.

Tabel 4. 5 Skenario *Use Case* Data Penyakit

CRUD Data Penyakit	
Tujuan	Mengelola data penyakit
Aktor	Admin
Skenario	Proses ini adalah sebuah kegiatan untuk memasukan data penyakit yang baru, mengubah, ataupun menghapus data penyakit yang sudah ada .
Kondisi Akhir	Sistem akan menampilkan data penyakit.

4. Data Basis Pengetahuan

Tabel 4.6 merupakan tabel deskripsi skenario dari *use case* data basis pengetahuan. Dalam tabel ini menjelaskan bagaimana skenario untuk menambahkan dan mengelola data basis pengetahuan.

Tabel 4. 6 Skenario *Use Case* Data Basis Pengetahuan

CRUD Data Basis Pengetahuan	
Tujuan	Mengelola data basis pengetahuan
Aktor	Admin
Skenario	Proses ini adalah sebuah kegiatan untuk memasukan data basis kasus yang baru, mengubah, ataupun menghapus data basis kasus yang sudah ada .

CRUD Data Basis Pengetahuan	
Kondisi Akhir	Sistem akan menampilkan data basis kasus.

5. Melakukan revisi kasus

Tabel 4.7 merupakan tabel deskripsi skenario dari *use case* data pemeriksaan. Dalam tabel ini menjelaskan bagaimana skenario untuk melakukan revisi pada hasil diagnosis *user* yang kurang tepat.

Tabel 4. 7 Skenario *Use Case* Data Revisi Kasus Baru

Melakukan revisi kasus	
Tujuan	Melakukan revisi kasus jika pada proses <i>retrieve</i> tidak ditemukan hasil diagnosis yang tepat
Aktor	Pakar
Skenario	Proses ini adalah sebuah kegiatan pakar untuk melakukan proses revisi kasus yang baru karena ketika proses <i>retrieve</i> berlangsung hasil diagnosis yang didapat tidak tepat/tidak sesuai.
Kondisi Akhir	Sistem akan menampilkan data-data hasil pemeriksaan beserta hasil presentasinya.

6. Mengisi Form Gejala

Tabel 4.8 merupakan tabel deskripsi skenario dari *use case* mengisi form gejala. Dalam tabel ini menjelaskan bagaimana skenario *user* mengisi form gejala penyakit.

Tabel 4. 8 Skenario *Use Case* Dalam Mengisi Form Gejala

Mengisi Form Gejala	
Tujuan	Mengisi form gejala
Aktor	Pasien
Skenario	Proses ini adalah sebuah kegiatan <i>user</i> mengisi form gejala guna untuk melakukan konsultasi berdasarkan gejala yang dirasakan.
Kondisi Akhir	Sistem akan menampilkan data-data gejala yang umum dirasakan dalam bentuk form.

7. Melihat hasil diagnosis penyakit beserta penanganan pertama

Tabel 4.9 merupakan tabel deskripsi skenario dari *use case* melihat diagnosis penyakit berdasarkan gejala. Dalam tabel ini menjelaskan bagaimana skenario *user* melihat diagnosis penyakit berdasarkan gejala beserta dengan solusi penanganan pertamanya.

Tabel 4. 9 Skenario *Use Case* Melihat Hasil Diagnosis Pasien

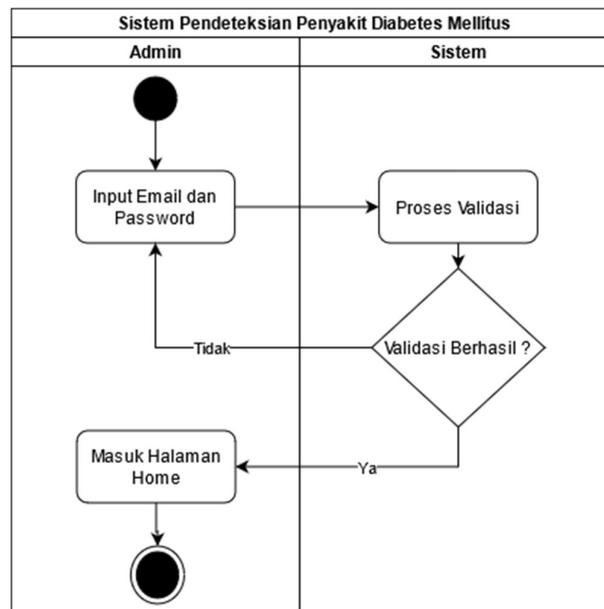
Melihat hasil diagnosis penyakit beserta penanganan pertama	
Tujuan	Melihat hasil diagnosis
Aktor	Pasien
Skenario	Proses ini adalah sebuah kegiatan pasien untuk melihat hasil diagnosis sementara penyakit beserta solusi penanganan pertamanya.
Kondisi Akhir	Sistem akan menampilkan data hasil diagnosis beserta solusi penanganan pertamanya.

4.3.2 Perancangan *Activity Diagram*

Activity Diagram merupakan alur kerja dari sebuah sistem. Dari awal hingga akhir, diagram ini menunjukkan langkah-langkah dalam proses kerja sistem yang dibuat. Perancangan penelitian yang dilakukan pada proses diagnosis dimodelkan dalam bentuk *Activity Diagram* seperti pada gambar berikut ini :

1. *Activity Diagram Login*

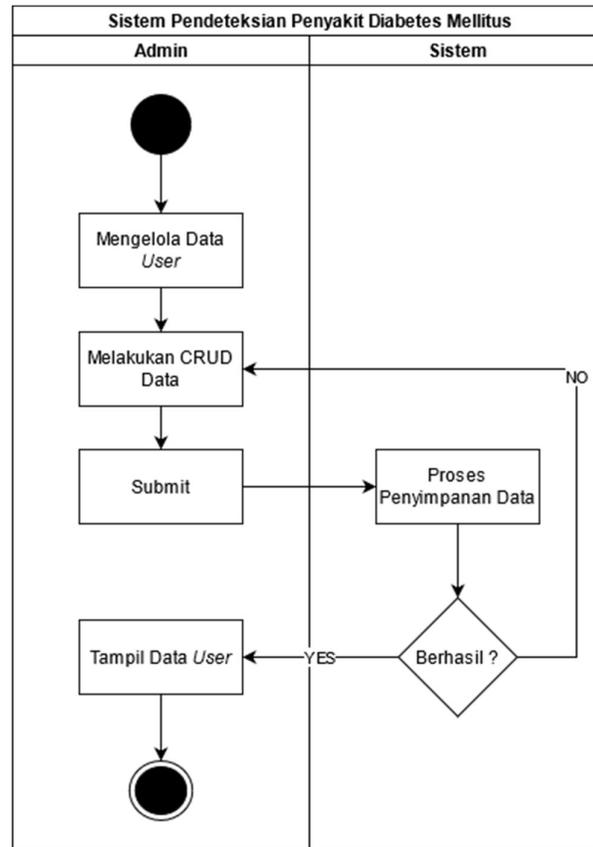
Proses ini dilakukan oleh pengelola, yaitu admin dan pengelola. Pengelola menginputkan *email* dan *password* untuk login. Kemudian sistem akan melakukan proses validasi, setelah berhasil maka akan diarahkan ke halaman pengelola.



Gambar 4. 2 *Activity Diagram Login Admin dan Pakar*

2. *Activity Diagram Mengelola Data User* pada Admin

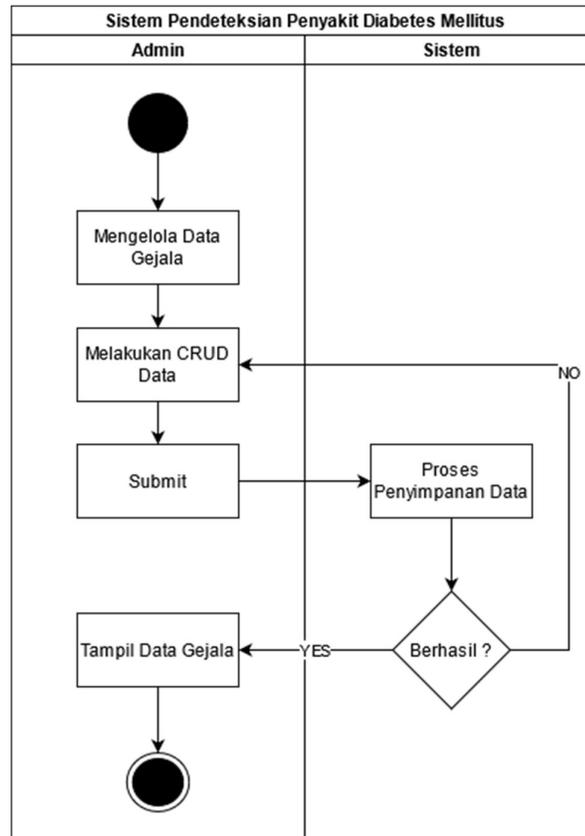
Activity diagram dibawah menggambarkan alur untuk mengelola data *user*. Terdapat 3 proses yang dapat dilakukan, yaitu menambah, mengubah dan menghapus data. *Activity diagram* mengelola data *user* dapat dilihat pada Gambar 4.3 berikut ini :



Gambar 4. 3 *Activity Diagram* CRUD Data

3. *Activity Diagram* Mengelola Data Gejala pada Admin

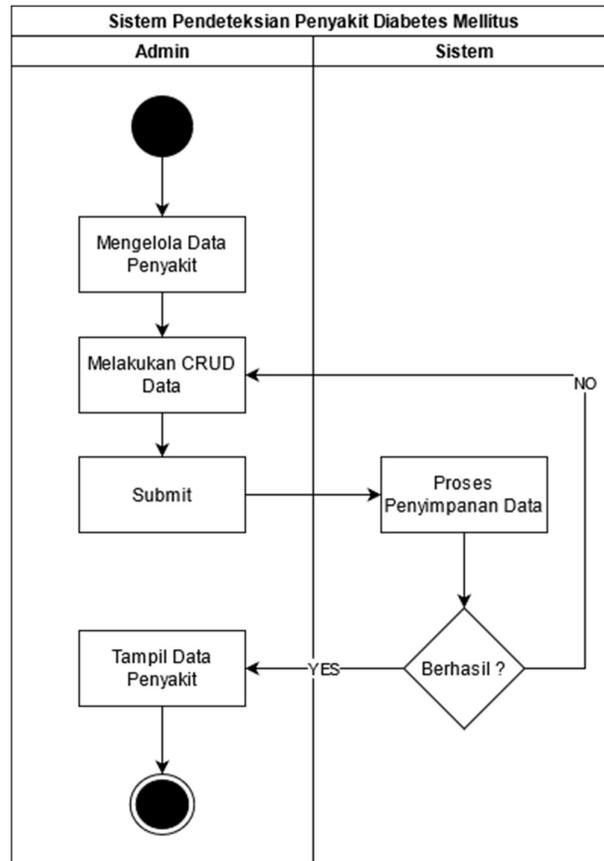
Activity diagram dibawah menggambarkan alur untuk mengelola data gejala. Terdapat 3 proses yang dapat dilakukan, yaitu menambah, mengubah dan menghapus data. *Activity diagram* mengelola data gejala dapat dilihat pada Gambar 4.4 berikut ini :



Gambar 4. 4 *Activity Diagram* CRUD Data Gejala

4. *Activity Diagram* Mengelola Data Penyakit pada Admin

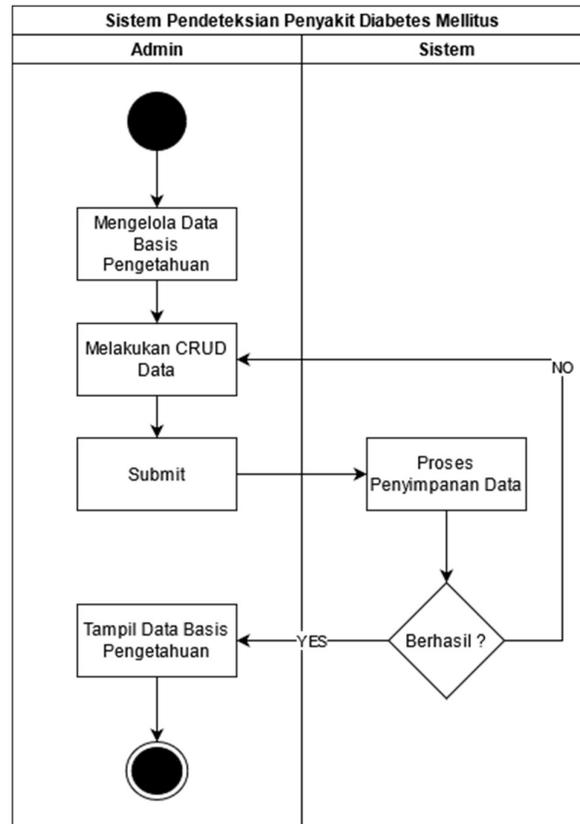
Activity diagram dibawah menggambarkan alur untuk mengelola data penyakit. Terdapat 3 proses yang dapat dilakukan, yaitu menambah, mengubah dan menghapus data. *Activity diagram* mengelola data penyakit dapat dilihat pada Gambar 4.5 berikut ini :



Gambar 4. 5 *Activity Diagram* CRUD Data Penyakit

5. *Activity Diagram* Mengelola Data Basis Pengetahuan pada Admin

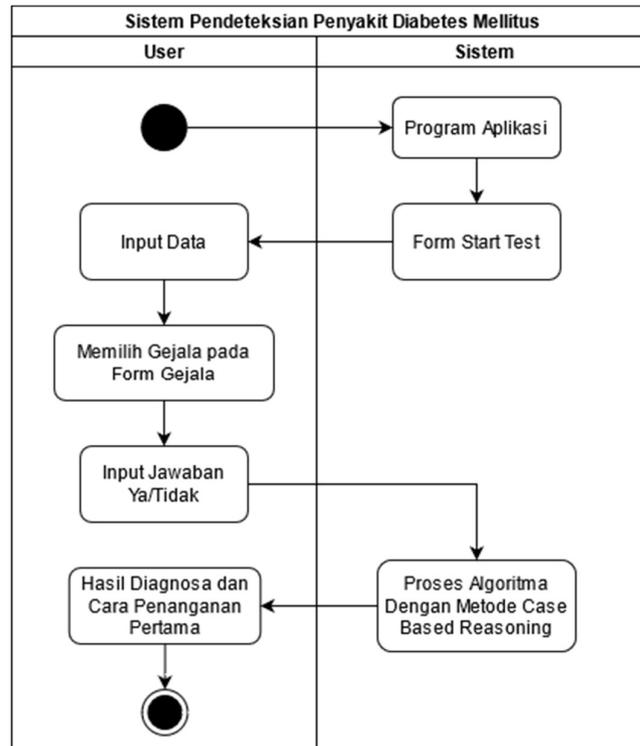
Activity diagram dibawah menggambarkan alur untuk mengelola data Basis Pengetahuan. Terdapat 3 proses yang dapat dilakukan, yaitu menambah, mengubah dan menghapus data. *Activity diagram* mengelola data Basis Pengetahuan dapat dilihat pada Gambar 4.6 berikut ini :



Gambar 4. 6 Activity Diagram CRUD Data Basis Pengetahuan

6. *Activity Diagram* Proses Diagnosis pada Pasien

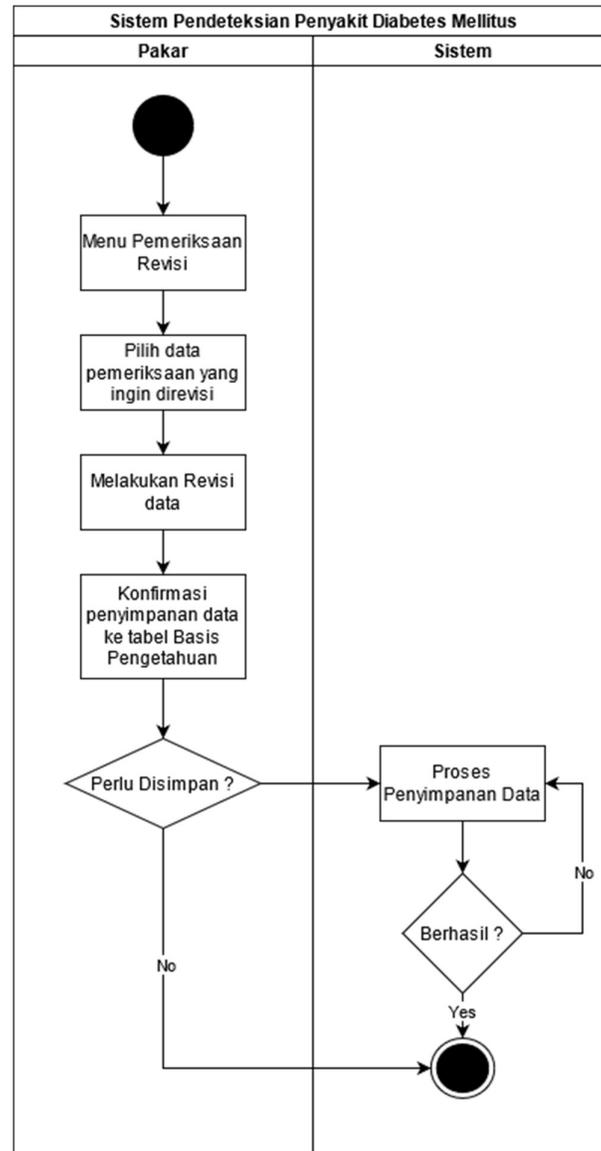
Activity diagram dibawah menggambarkan alur ketika proses diagnosis penyakit yang dilakukan oleh pasien. Dimana sebelum proses dilakukan, pasien terlebih dahulu mengisi data pada form gejala konsultasi. *Activity diagram* mengelola data dapat dilihat pada Gambar 4.7 berikut ini :



Gambar 4. 7 *Activity Diagram* Proses Diagnosis Penyakit pada Pasien

7. *Activity Diagram* Proses Revisi oleh Pakar

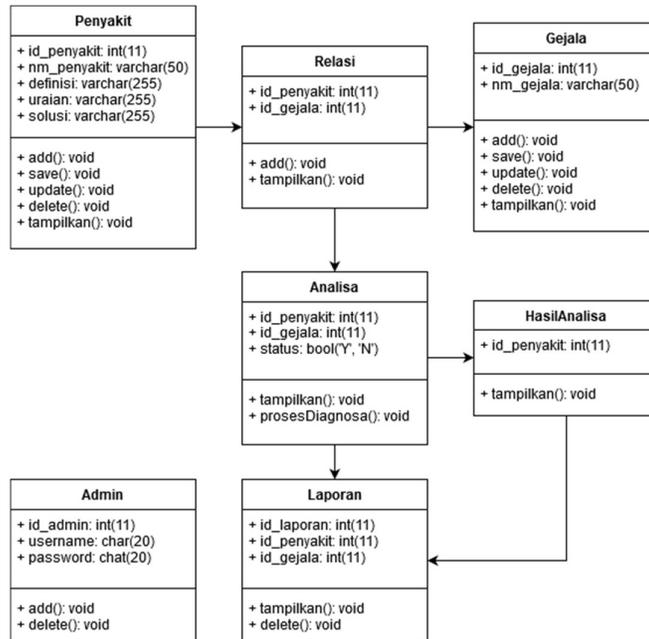
Activity diagram menggambarkan alur ketika proses revisi kasus dilakukan. Dimana proses ini dilakukan jika hasil pemeriksaan oleh pasien menghasilkan persentase kurang dari 50%. *Activity diagram* proses revisi dapat dilihat pada Gambar 4.8 berikut ini :



Gambar 4. 8 *Activity Diagram* Proses Revisi oleh Pakar

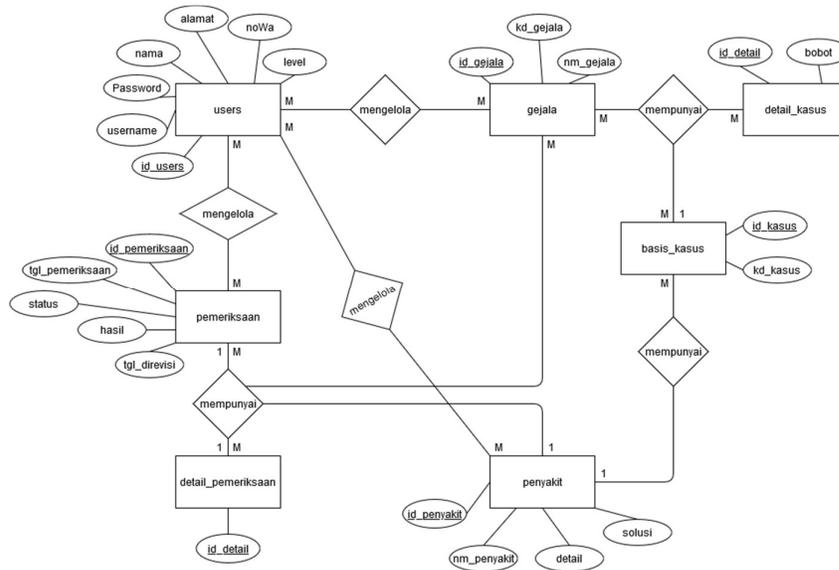
4.3.3 Perancangan *Class Diagram*

Perancangan penelitian yang dilakukan pada proses diagnosis dimodelkan dalam bentuk *Class Diagram* seperti pada gambar 4.9 :

Gambar 4. 9 *Class Diagram Metode*

4.3.4 Perancangan *Entity Relationship Diagram*

ERD atau *Entity Relationship Diagram* adalah suatu bentuk diagram yang menjelaskan hubungan antar objek-objek data yang mempunyai hubungan antar relasi. ERD digunakan untuk menyusun struktur data dan hubungan antar data, dan untuk menggambarannya digunakan notasi, simbol, bagan, dan lain sebagainya. ERD juga merupakan salah satu model yang digunakan untuk mendesain database dengan tujuan menggambar data yang berelasi pada sebuah *database*. Perancangan ERD ditunjukkan pada gambar 4.10 berikut ini :



Gambar 4. 10 Perancangan *Entity Relationship Diagram*

4.4 Perancangan *Database*

Perancangan Basis Data adalah proses untuk menentukan isi dan pengaturan data yang dibutuhkan untuk mendukung berbagai rancangan sistem. Berikut ini merupakan rancangan basis data untuk Sistem Pendeteksian Penyakit Diabetes Mellitus Secara Dini Menggunakan Metode *Case Based Reasoning*.

4.4.1 Struktur Tabel

Struktur tabel berfungsi sebagai rancangan dalam database sistem informasi yang akan dibuat, adapun struktur tabel yang ada dari perancangan Sistem Pendeteksian Penyakit Diabetes Mellitus Secara Dini Menggunakan Metode *Case Based Reasoning* yaitu sebagai berikut :

1. Tabel *Users*

Tabel 4.9 merupakan tabel *users*. Tabel *users* digunakan pada use case login. Login adalah hal yang harus dilakukan pertama kali ketika masuk ke dalam sistem. Tabel *users* hanya digunakan untuk menyimpan data admin dan pakar saja.

Tabel 4. 10 Tabel *Users*

Nama Atribut	Tipe Data	Keterangan
Id_users	INT (11)	NOT NULL, PRIMARY_KEY, AUTO_INCREMENT
Email	VARCHAR (100)	NOT NULL
Password	VARCHAR (100)	NOT NULL
Nama	VARCHAR (200)	NOT NULL
Alamat	VARCHAR (255)	NOT NULL
NoWa	CHAR (15)	NOT NULL
level	ENUM ('Admin', 'Pakar')	NOT NULL

2. Tabel Gejala

Tabel gejala digunakan untuk menyimpan daftar gejala yang umum dirasakan oleh para penderita diabetes mellitus. Yang ditunjukkan pada tabel 4.10 dibawah ini :

Tabel 4. 11 Tabel Gejala

Nama Atribut	Tipe Data	Keterangan
Id_gejala	INT (11)	NOT NULL, PRIMARY_KEY, AUTO_INCREMENT
Kd_gejala	VARCHAR (8)	NOT NULL
Nm_gejala	VARCHAR (255)	NOT NULL
Ktg_gejala	ENUM('Non Gestasional', 'Gestasional dan Non Gestasional', 'Gestasional')	NOT NULL

3. Tabel Penyakit

Tabel penyakit digunakan untuk menyimpan daftar penyakit diabetes mellitus. Yang ditunjukkan pada tabel 4.11 dibawah ini :

Tabel 4. 12 Tabel Penyakit

Nama Atribut	Tipe Data	Keterangan
Id_penyakit	INT (11)	NOT NULL, PRIMARY_KEY, AUTO_INCREMENT
Kd_penyakit	CHAR (5)	NOT NULL
Nm_penyakit	VARCHAR (100)	NOT NULL
Detail	TEXT	NOT NULL
Solusi	TEXT	NOT NULL

4. Tabel Basis Kasus atau Basis Pengetahuan

Tabel basis_kasus digunakan untuk menyimpan daftar basis kasus atau basis pengetahuan yang sering terjadi pada penderita diabetes mellitus. Yang ditunjukkan pada tabel 4.12 dibawah ini :

Tabel 4. 13 Tabel Basis Pengetahuan atau Basis Kasus

Nama Atribut	Tipe Data	Keterangan
Id_kasus	INT (11)	NOT NULL, PRIMARY_KEY, AUTO_INCREMENT
Kd_kasus	CHAR (8)	NOT NULL
Fk_penyakit	INT (11)	NOT NULL, FOREIGN_KEY

5. Tabel Detail Kasus atau Pengetahuan

Tabel detail_kasus digunakan untuk menyimpan daftar gejala dari masing-masing basis kasus atau basis pengetahuan yang sering terjadi pada penderita diabetes. Yang ditunjukkan pada tabel 4.13 dibawah ini :

Tabel 4. 14 Tabel Detail Kasus atau Pengetahuan

Nama Atribut	Tipe Data	Keterangan
Id_detail	INT (11)	NOT NULL, PRIMARY_KEY, AUTO_INCREMENT
Fk_gejala	INT (11)	NOT NULL, FOREIGN_KEY
Fk_kasus	INT (11)	NOT NULL, FOREIGN_KEY

Bobot	ENUM ('1','3','5')	NOT NULL
-------	--------------------	----------

6. Tabel Pemeriksaan

Tabel pemeriksaan digunakan untuk menyimpan daftar hasil pemeriksaan yang dilakukan oleh pasien. Yang ditunjukkan pada tabel 4.14 dibawah ini :

Tabel 4. 15 Tabel Pemeriksaan

Nama Atribut	Tipe Data	Keterangan
Id_pemeriksaan	INT (11)	NOT NULL, PRIMARY_KEY, AUTO_INCREMENT
Tgl_pemeriksaan	DATE	NOT NULL
Status	ENUM ('1','2','3','4')	NOT NULL
Fk_penyakit	INT (11)	NULL, FOREIGN_KEY
Hasil	INT (11)	NULL
Fk_user	INT (11)	NULL, FOREIGN_KEY
Tgl_direvisi	DATE	NULL

7. Tabel Detail Pemeriksaan

Tabel detail_pemeriksaan digunakan untuk menyimpan daftar gejala dari masing-masing hasil pemeriksaan yang telah dilakukan oleh pasien. Yang ditunjukkan pada tabel 4.15 dibawah ini :

Tabel 4. 16 Tabel Detail Pemeriksaan

Nama Atribut	Tipe Data	Keterangan
Id_detail	INT (11)	NOT NULL, PRIMARY_KEY, AUTO_INCREMENT
Fk_pemeriksaan	INT (11)	NOT NULL, FOREIGN_KEY
Fk_gejala	INT (11)	NOT NULL, FOREIGN_KEY

8. Tabel Komentar

Tabel komentar digunakan untuk menyimpan daftar komentar yang diberikan oleh *user*. Yang ditunjukkan pada tabel 4.16 dibawah ini :

Tabel 4. 17 Tabel Komentar

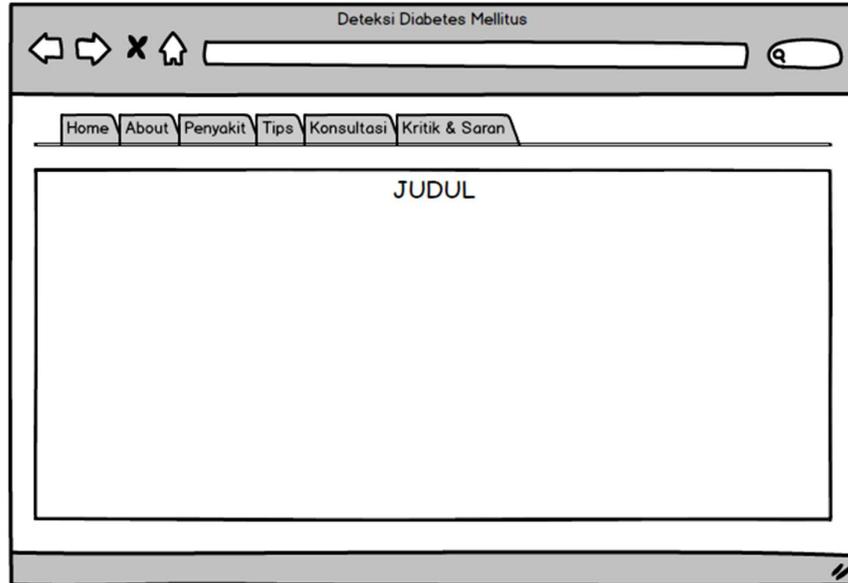
Nama Atribut	Tipe Data	Keterangan
Id_komen	INT (11)	NOT NULL, PRIMARY_KEY, AUTO_INCREMENT
Komen	TEXT	NOT NULL

4.5 Perancangan Antarmuka

Perancangan antarmuka merupakan proses penggambaran bagaimana sebuah sistem dibentuk. Merancang antarmuka merupakan salah satu bagian penting dari merancang sistem. Perancangan antarmuka atau *mockup* digunakan untuk mempermudah dalam perancangan desain sistem yang akan dibuat kepada *user* dengan konsep mudah dipahami dan mudah digunakan. Berikut ini adalah tampilan perancangan antarmuka dari Sistem Pendeteksian Penyakit Diabetes Mellitus Secara Dini Menggunakan Metode *Case Based Reasoning*

1. Perancangan Antarmuka Halaman Utama Sistem

Halaman ini adalah halaman awal ketika sistem dijalankan, di mana pada halaman ini semua *user* dapat melihat beberapa informasi mengenai sistem maupun kesehatan. Untuk lebih jelasnya dapat dilihat pada gambar 4.11 berikut.



Gambar 4. 11 *Mockup* Halaman Utama Sistem

2. Perancangan Antarmuka Halaman Form Gejala untuk Konsultasi Pasien

Halaman form gejala pasien adalah halaman yang memuat data gejala-gejala yang umum dirasakan oleh penderita diabetes dan juga merupakan halaman untuk proses konsultasi untuk *user* atau pasien. Rancangan antarmuka halaman form gejala dapat dilihat pada Gambar 4.12 berikut.

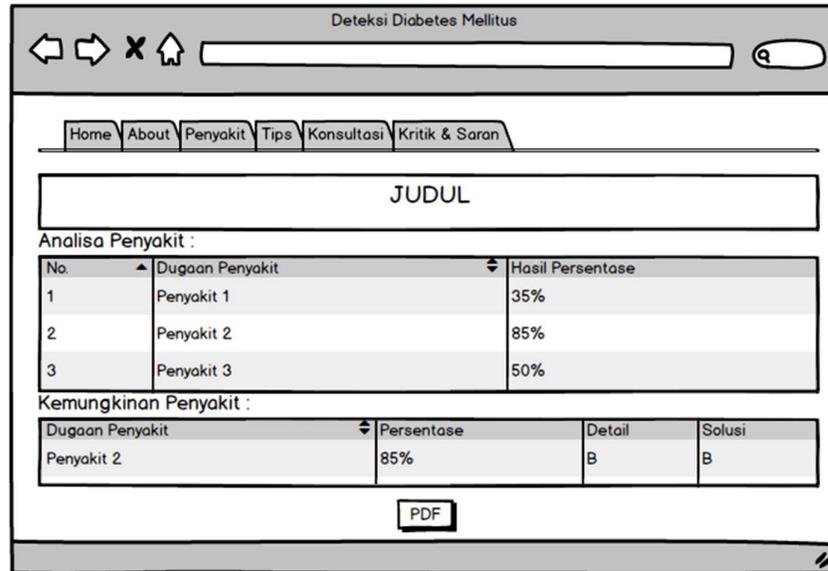
No.	Gejala	Action
1	Gejala 1	<input type="checkbox"/>
2	Gejala 2	<input type="checkbox"/>
3	Gejala 3	<input type="checkbox"/>

SUBMIT

Gambar 4. 12 *Mockup* Form Gejala untuk Proses Konsultasi Pasien

3. Perancangan Antarmuka Halaman Hasil Konsultasi Pasien

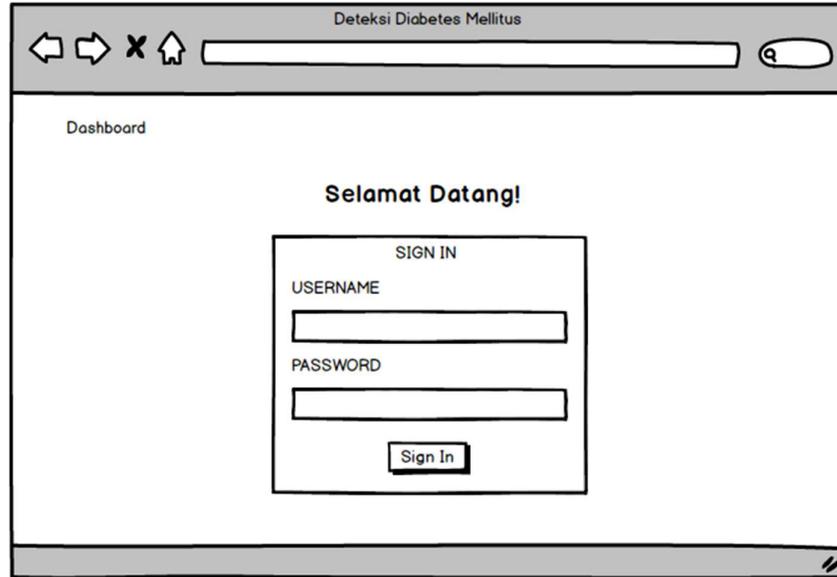
Halaman hasil konsultasi pasien adalah halaman yang memuat hasil diagnosis penyakit yang didapat setelah pasien memasukkan gejala-gejala yang dirasakan di form gejala. Rancangan antarmuka halaman hasil konsultasi dapat dilihat pada Gambar 4.13 berikut.



Gambar 4. 13 *Mockup* Halaman Hasil Diagnosis Pasien

4. Perancangan Antarmuka Halaman *Login*

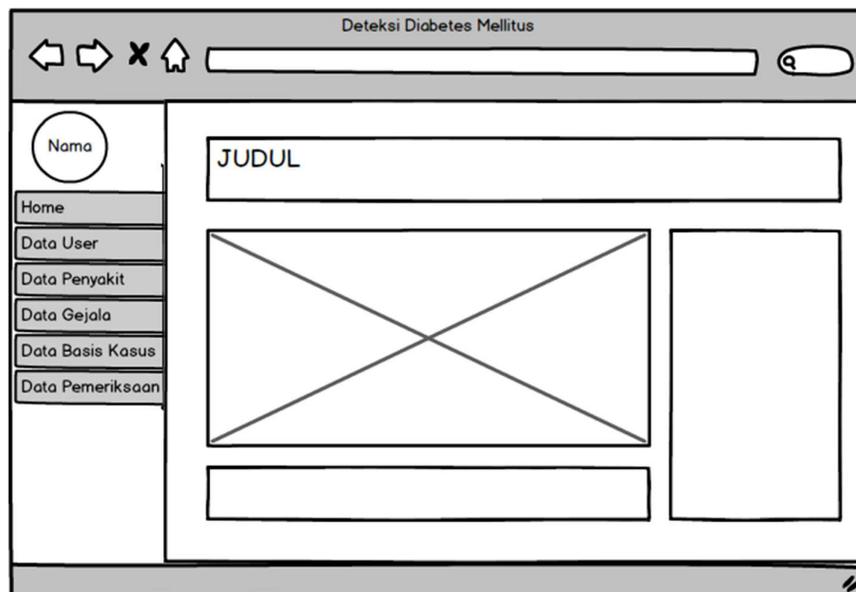
Halaman ini adalah halaman *login* untuk pengelola, yaitu admin dan pakar, dimana pengelola diminta untuk memasukkan email dan password agar dapat mengakses sistem. Untuk lebih jelasnya dapat dilihat pada gambar 4.14 berikut.



Gambar 4. 14 *Mockup* Halaman *Login* untuk Admin dan Pakar

5. Perancangan Antarmuka Halaman *Home* Pengelola

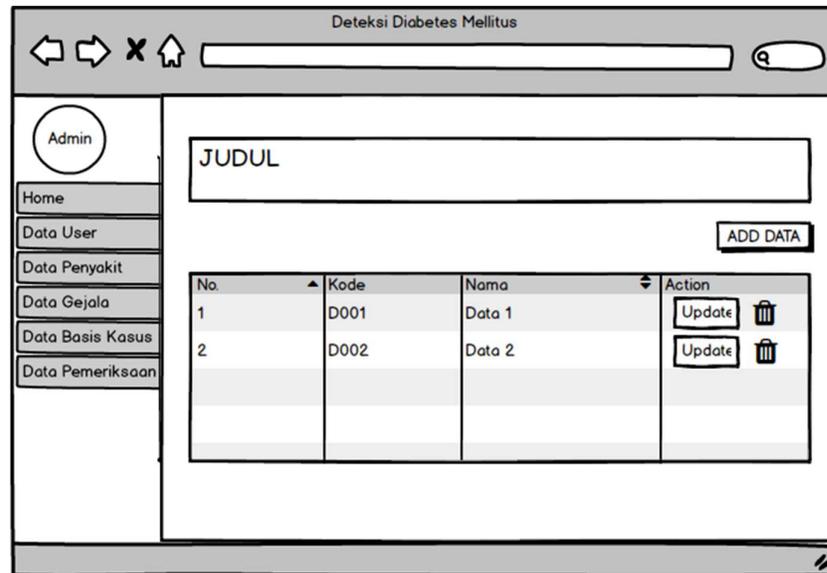
Setelah berhasil login secara otomatis akan muncul halaman pengelola dari Sistem Pendeteksian Penyakit Diabetes Mellitus Secara Dini. Rancangan antarmuka halaman *home* dapat dilihat pada Gambar 4.15.



Gambar 4. 15 *Mockup* Halaman *Home* Pengelola

6. Perancangan Antarmuka Halaman Mengelola Data pada Admin

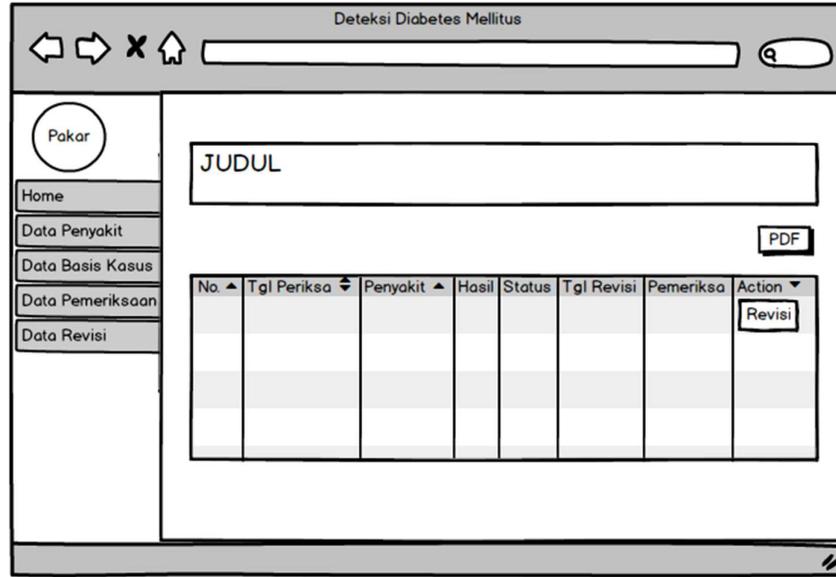
Halaman mengelola data adalah halaman yang memuat informasi tentang data *user*, gejala, penyakit, basis kasus atau basis pengetahuan, dan komentar. Selain itu pada halaman mengelola data, admin juga dapat memasukkan data, mengedit, maupun menghapus data. Rancangan antarmuka halaman mengelola data dapat dilihat pada gambar 4.16 berikut.



Gambar 4. 16 *Mockup* Halaman CRUD Data pada Admin

7. Perancangan Antarmuka Halaman Data Pemeriksaan Revisi pada Pakar

Halaman data pemeriksaan revisi adalah halaman yang memuat informasi tentang data pemeriksaan yang telah dilakukan oleh pasien dan perlu peninjauan oleh pakar karena hasil yang didapat kurang dari 50%. Selain itu pada halaman data pemeriksaan revisi ini pakar juga dapat meninjau data detail pemeriksaan. Rancangan antarmuka halaman pemeriksaan revisi dapat dilihat pada gambar 4.17 berikut.



Gambar 4. 17 *Mockup* Halaman Data Pemeriksaan Revisi pada Pakar

8. Perancangan Antarmuka Halaman Mengelola Data Detail Pemeriksaan Revisi pada Pakar

Setelah meninjau detail pemeriksaan pada halaman data pemeriksaan, maka secara otomatis pakar akan diarahkan pada halaman detail pemeriksaan revisi yang berisi data-data gejala hasil konsultasi pasien yang harus direvisi oleh pakar. Rancangan antarmuka halaman detail pemeriksaan revisi dapat dilihat pada Gambar 4.18.

The mockup shows a web browser window titled "Deteksi Diabetes Mellitus". The browser's address bar is empty. On the left side, there is a navigation menu with a circular profile icon labeled "Pakar" and several menu items: "Home", "Data Penyakit", "Data Basis Kasus", "Data Pemeriksaan", and "Data Revisi". The main content area contains a form with the following elements:

- A text input field labeled "JUDUL".
- A dropdown menu labeled "Penyakit" with the selected option "Dugaan Penyakit".
- A table with three columns: "No.", "Gejala", and "Bobot".

No.	Gejala	Bobot
1.	Gejala 1	Pilih Bobot

Below the table are three buttons: "SIMPAN", "Kembali", and "Hapus".

Gambar 4. 18 *Mockup* Halaman Data Detail Pemeriksaan Revisi pada Pakar