

BAB III

METODE PENELITIAN

3.1 Metode Pengembangan Perangkat Lunak

Metode perangkat lunak yang digunakan untuk merancang sistem prediksi dini penyakit jantung adalah metode air terjun. Penjelasan langkah-langkah dalam menjelajahi air terjun adalah sebagai berikut.

3.1.1 Pengumpulan Data

Pada tahap ini melakukan pengumpulan data melalui situs *web* resmi Kaggle, kemudian melakukan penelitian literatur, dimana mencari dan mempelajari tentang sistem pakar, algoritma *K-Nearest Neighbor*, dan mencari informasi tentang penyakit jantung, kemudian melakukan observasi untuk mendapatkan informasi yang diperlukan secara akurat.

3.1.2 Analisis Sistem

Pada tahap ini dianalisis kebutuhan implementasi fungsi-fungsi sistem agar dapat memenuhi tujuan sistem yang dibuat, dan spesifikasi sistem didefinisikan secara rinci sebelum diimplementasikan dalam pembuatan sistem.

3.1.3 Perancangan Sistem

Pada tahap perencanaan dilakukan pemodelan sistem berdasarkan hasil analisis kebutuhan. Selain itu, pemodelan berbasis data juga dibuat untuk menggambarkan hubungan antar data. Perancangan sistem akan menggunakan *flowchart* dan *Class Diagram*.

3.1.4 Implementasi Sistem

Implementasi sistem, yaitu tahap penerapan sistem atau proses coding yang dilaksanakan berdasarkan rancangan yang telah dibuat sebelumnya sesuai dengan kebutuhan para pihak.

3.1.5 Pengujian Sistem

Pengujian sistem akan dilakukan dengan menggunakan metode pengujian kotak hitam (*black box testing*) untuk menguji fungsionalitas sistem dan akan menguji model KNN yang dibangun dengan mengukur tingkat akurasi dari model KNN tersebut.

3.1.6 Pengoperasian dan Pemeliharaan

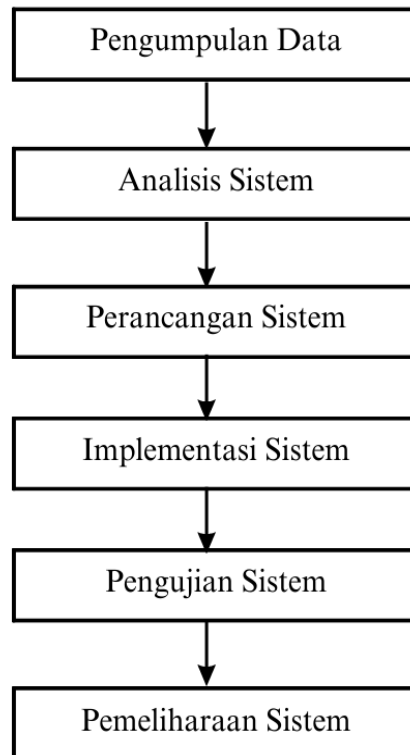
Setelah sistem sudah diuji coba berkali-kali untuk melihat apakah ada kesalahan atau tidak, maka aplikasi tersebut dapat dipasang dan digunakan. Jika selama penggunaan terdapat kekurangan atau terjadi masalah, maka akan dilakukan pengembangan dan perbaikan.

3.2 Metode Penelitian

Dalam melakukan penelitian, penulis menggunakan metode kualitatif yang didapat oleh peneliti dari sumber data secara langsung yang disebut dengan data primer. Data primer pada penelitian ini diperoleh dengan menggunakan sumber data kuesioner dari objek yang diteliti yaitu data berupa (kadar gula darah, umur, jenis kelamin, tekanan darah, kolesterol, elektrokardiografi). Dalam kuesioner digunakan skala pengukuran berupa skala likert. Pada kuesioner, responden memberikan hasil *medical check up* untuk setiap pernyataan dalam kuesioner.

3.2.1 Diagram Alir Penelitian

Langkah-langkah metodologi penelitian ini ditunjukkan pada Gambar 3.1 di bawah ini.



Gambar 3. 1 Diagram Alir Penelitian

3.3 Analisis Kebutuhan Sistem

Analisis kebutuhan sistem merupakan langkah untuk menentukan kebutuhan yang diperlukan dalam penelitian ini untuk membuat sistem deteksi dini penyakit jantung menggunakan algoritma KNN. Analisis kebutuhan sistem adalah sebagai berikut.

3.3.1 Perangkat Keras

Perangkat keras yang dapat digunakan untuk membangun sebuah sistem sebagai berikut, Spesifikasi perangkat keras yang digunakan penulis dalam penelitian adalah sebagai berikut.

- a) Processor Intel Core i5 intel iRISXe
- b) Hardisk 500 GB.
- c) RAM 4 GB.
- d) *Keyboard dan mouse.*

3.3.2 Perangkat Lunak

Perangkat lunak yang digunakan untuk membangun sistem adalah sebagai berikut.

- a) Sistem Operasi Windows 10 64bit.
- b) Bahasa Pemrograman PHP, dan javascript.
- c) MySQL Database
- d) Web Browser
- e) Visual Studio Code
- f) XAMPP

3.4 Desain Sistem

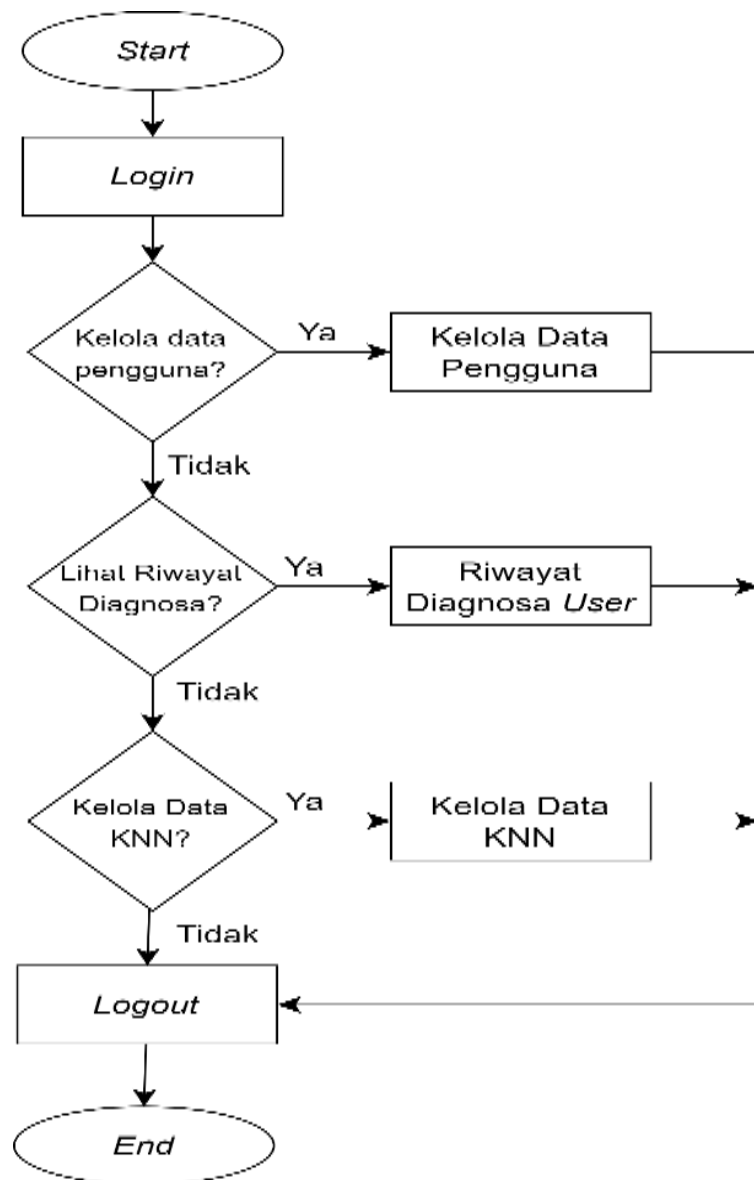
Desain sistem mendeskripsikan bagaimana alur sistem, peran aktor terhadap sistem, hubungan antar kelas dan juga tampilan rancangan antar muka dari sistem yang akan dibangun. Desain sistem adalah sebagai berikut.

3.4.1 Flowchart

Flowchart menggambarkan bagaimana alur dari penggunaan sistem. Berikut adalah alur sistem untuk admin dan juga pengguna dengan menggunakan *flowchart*.

1. *Flowchart* Admin

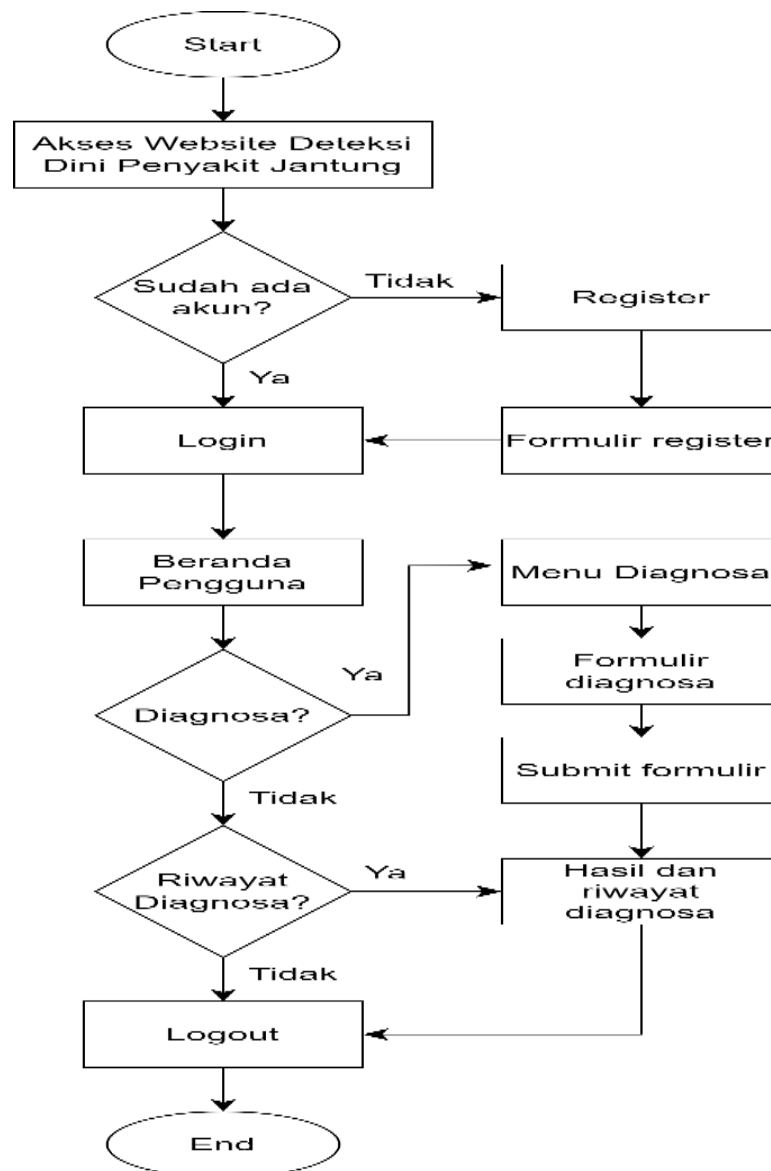
Flowchart admin akan menggambarkan alur penggunaan sistem bagi admin, mulai dari proses login, mengelola data pengguna, mengelola data KNN, melihat hasil diagnosa pengguna, serta dapat mengubah informasi akun admin. *Flowchart* admin dapat dilihat pada gambar 3.2.



Gambar 3. 2 Flowchart Admin

2. Flowchart Pengguna (User)

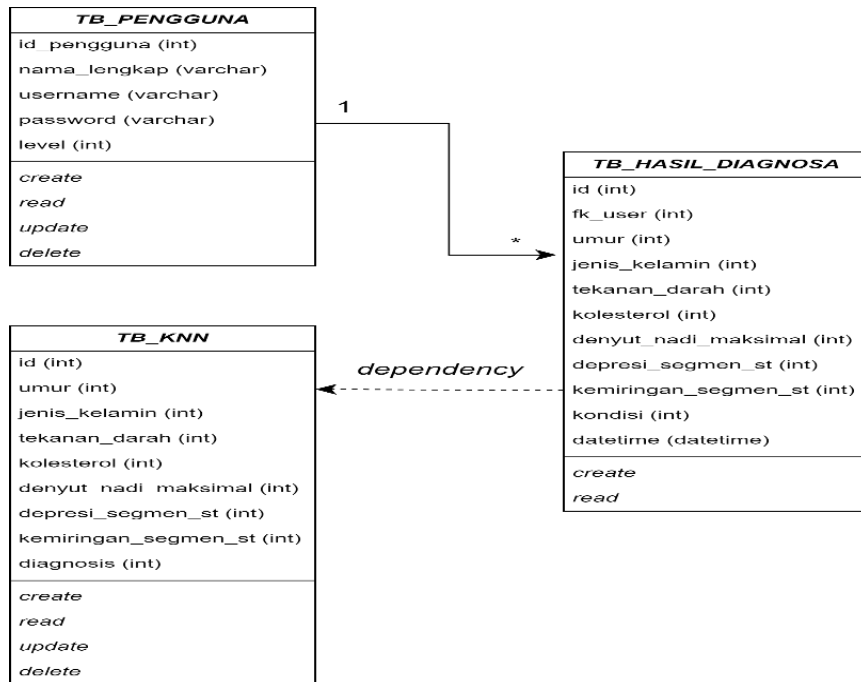
Flowchart pengguna akan menggambarkan bagaimana alur dari penggunaan sistem bagi pengguna, mulai dari proses masuk jika sudah ada akun, proses pendaftaran akun jika belum memiliki akun. Hingga proses diagnosa. *Flowchart user* dapat dilihat pada gambar 3.3.



Gambar 3.3 *Flowchart User*

3.4.2 *Class Diagram*

Class diagram akan menjelaskan hubungan antar kelas-kelas pada sistem. *Class diagram* dapat dilihat pada gambar 3.5.



Gambar 3. 4 Class Diagram

Basis data yang ada pada gambar 3.5 di atas memaparkan bahwa basis data yang akan dibangun dan digunakan pada sistem memiliki 3 tabel, yaitu *tb_pengguna*, *tb_knn*, dan *tb_hasil diagnosa*. Struktur tabel pada ketiga tabel dalam basis data tersebut adalah sebagai berikut.

1. Tabel Pengguna

Tabel pengguna memiliki sifat *one to many* terhadap tabel hasil diagnosa. Hal ini menandakan bahwa 1 pengguna memiliki lebih dari 1 data hasil diagnosa (dapat melakukan diagnosa lebih dari 1 kali). Struktur dari tabel pengguna dapat dilihat pada tabel 3.1.

Tabel 3. 1 Struktur Tabel Pengguna

Kolom	Type Data	Atribut
id_pengguna	int	PRIMARY_KEY
nama_lengkap	varchar (100)	-
username	varchar (100)	-
password	varchar (100)	-
Level	int	-

2. Tabel KNN

Tabel KNN memiliki sifat *dependency* terhadap tabel hasil diagnosa yang menandakan bahwa tabel hasil diagnosa memiliki kebutuhan terhadap tabel knn untuk melakukan diagnosa Struktur dari tabel knn dapat dilihat pada tabel 3.2.

Tabel 3. 2 Struktur Tabel KNN

Kolom	Tipe Data	Atribut
id	int	PRIMARY_KEY
umur	int	-
jenis_kelamin	int	-
tekanan_darah	int	-
kolesterol	int	-
denyut_nadi_maksimal	int	-
depresi_segmen_st	int	-
kemiringan_segmen_st	int	-
diagnosis	int	-

3. Tabel Hasil Diagnosa

Tabel diagnosa merupakan tabel untuk menyimpan hasil diagnosa yang dilakukan oleh pengguna. Struktur dari tabel hasil diagnosa dapat dilihat pada tabel 3.3.

Tabel 3. 3 Struktur Tabel Hasil Diagnosa

Kolom	Tipe Data	Atribut
id	int	PRIMARY_KEY
fk_user	int	FOREIGN_KEY
umur	int	-
jenis_kelamin	int	-
tekanan_darah	int	-
kolesterol	int	-

denyut_nadi_maksimal	int	-
depresi_segmen_st	int	-
kemiringan_segmen_st	int	-
kondisi	int	-
datetime	datetime	-

3.4.3 Wireframe

Wireframe merupakan rancangan tampilan antarmuka dari sistem yang akan dibangun. Tampilan rancangan antarmuka sistem adalah sebagai berikut.

1. Halaman Masuk

Halaman masuk akan digunakan oleh admin dan pengguna untuk masuk ke dalam siste. Tampilan rancangan antarmuka dapat dilihat pada gambar 3.6.

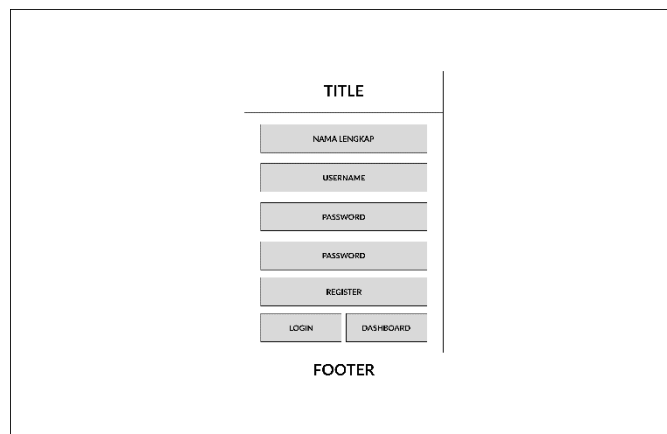
The wireframe shows a central login form with the following elements:

- TITLE**: A label above the form.
- USERNAME**: A text input field.
- PASSWORD**: A text input field.
- LOGIN**: A button.
- REGISTER**: A button.
- DASHBOARD**: A button.
- FOOTER**: A label below the form.

Gambar 3. 5 *Wireframe* Halaman Masuk

2. Halaman Daftar Akun

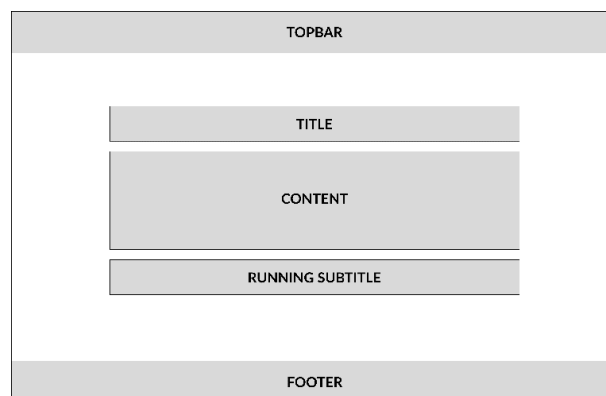
Halaman daftar akun digunakan oleh *user* untuk membuat akun. Tampilan rancangan antar muka halaman daftar akun dapat dilihat pada gambar 3.7.



Gambar 3. 6 *Wireframe* Daftar Akun

3. Halaman Beranda Umum

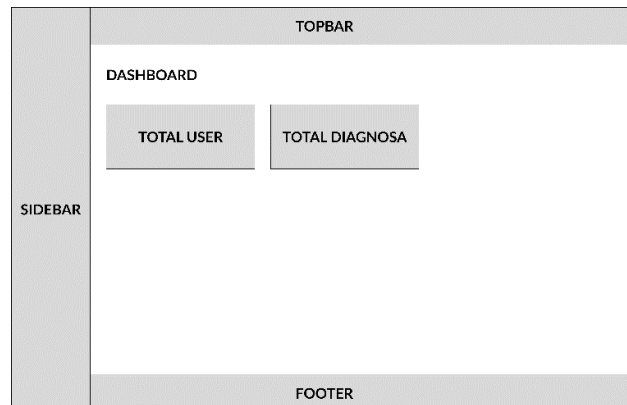
Halaman beranda umum digunakan oleh *user* maupun admin saat pertama kali mengunjungi sistem. Tampilan rancangan antarmuka beranda umum dapat dilihat pada gambar 3.8.



Gambar 3. 7 *Wireframe* Beranda Umum

4. Halaman Beranda Admin

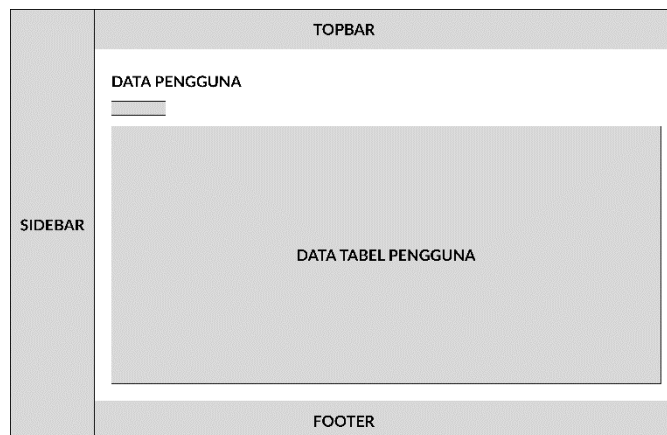
Halaman beranda admin akan digunakan oleh admin, halaman ini muncul pertama kali ketika admin telah berhasil *login*. Tampilan rancangan antarmuka halaman beranda admin dapat dilihat pada gambar 3.9.



Gambar 3. 8 *Wireframe* Beranda Admin

5. Halaman Data Pengguna

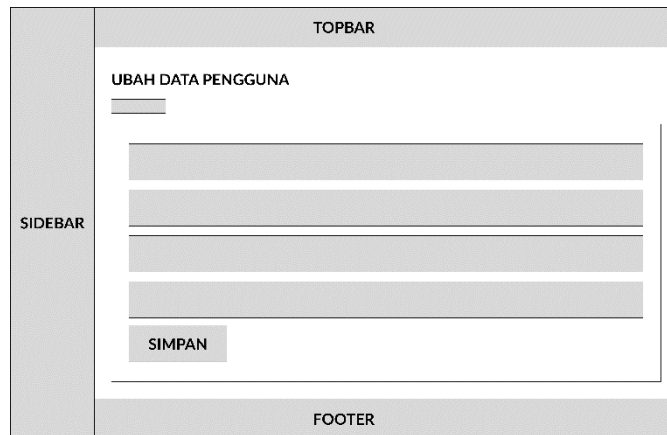
Halaman data pengguna akan digunakan oleh admin untuk mengelola data pengguna sistem. Tampilan rancangan antarmuka halaman data pengguna dapat dilihat pada gambar 3.10.



Gambar 3. 9 *Wireframe* Kelola Data Pengguna

6. Halaman Ubah Data Pengguna

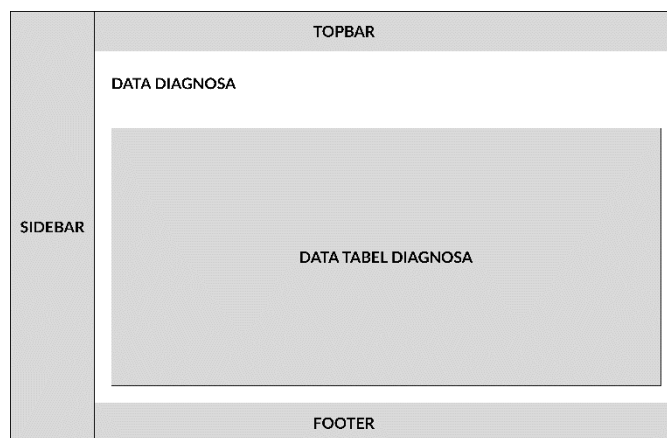
Halaman ubah data pengguna digunakan oleh admin untuk mengubah data pengguna yang ingin diubah. Tampilan rancangan antarmuka ubah data pengguna dapat dilihat pada gambar 3.11.



Gambar 3. 10 *Wireframe* Ubah Data Pengguna

7. Halaman Data Diagnosa

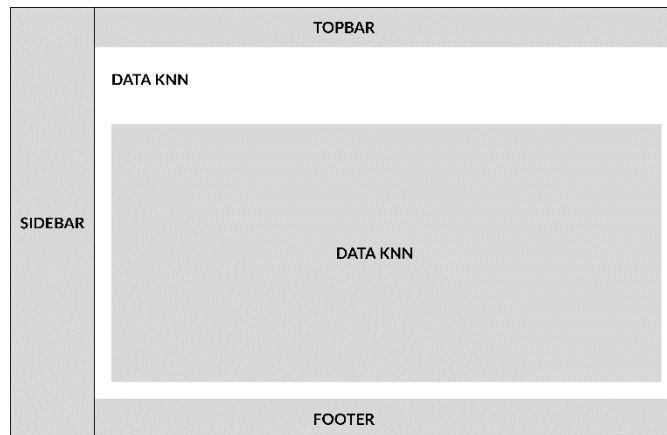
Halaman data diagnosa digunakan oleh admin untuk melihat keseluruhan data diagnosa yang dilakukan oleh pengguna. Tampilan rancangan antarmuka data diagnosa dapat dilihat pada gambar 3.12.



Gambar 3. 11 *Wireframe* Data Diagnosa

8. Halaman Data KNN

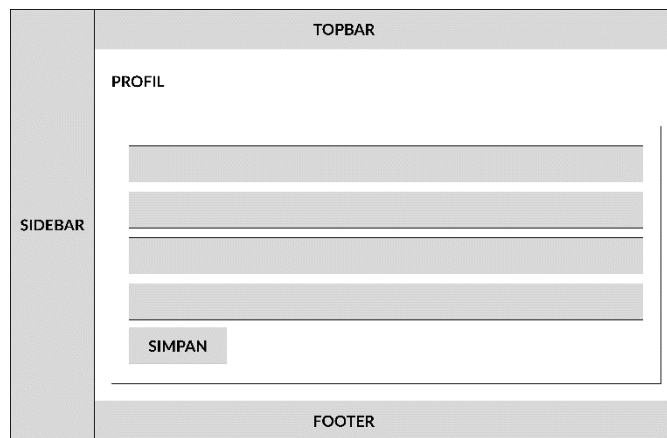
Halaman data KNN digunakan oleh admin untuk menambahkan data latih untuk algoritma KNN yang digunakan. Tampilan rancangan antarmuka data KNN dapat dilihat pada gambar 3.13.



Gambar 3. 12 *Wireframe* Data KNN

9. Halaman Profil

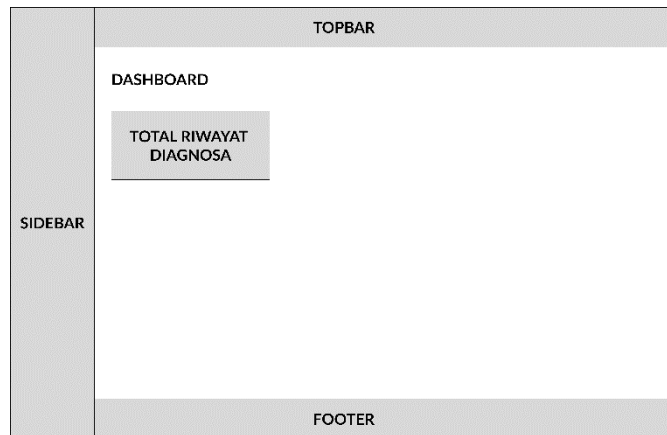
Halaman profil merupakan halaman yang akan digunakan oleh admin dan *user* untuk mengubah data akun. Tampilan rancangan antarmuka halaman profil dapat dilihat pada gambar 3.14.



Gambar 3. 13 *Wireframe* Profil

10. Halaman Beranda Pengguna

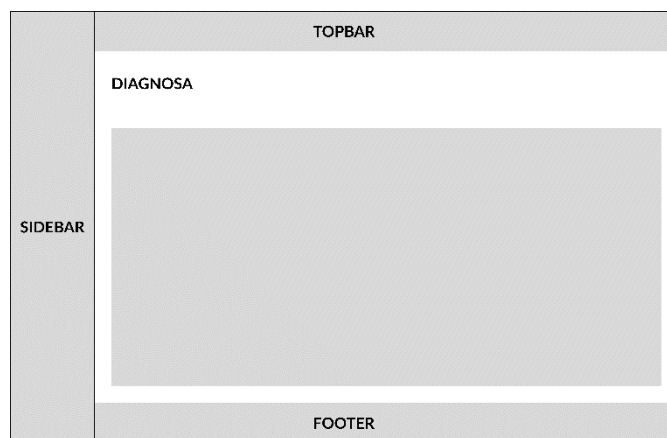
Halaman beranda pengguna merupakan halaman yang muncul pertama kali saat pengguna berhasil *login*. Tampilan rancangan antarmuka beranda pengguna dapat dilihat pada gambar 3.15.



Gambar 3. 14 *Wireframe* Beranda Pengguna

11. Halaman Diagnosa

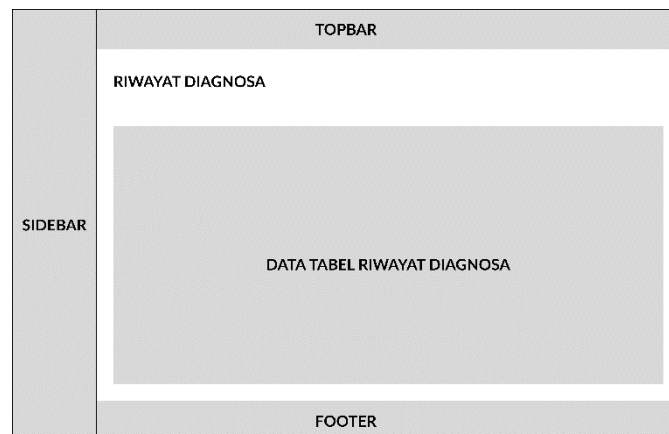
Halaman diagnosa akan digunakan oleh *user* untuk melakukan diagnosa dini penyakit jantung dengan menjawab pertanyaan. Tampilan rancangan antarmuka halaman diagnosa dapat dilihat pada gambar 3.16.



Gambar 3. 15 *Wireframe* Diagnosa

12. Halaman Riwayat Diagnosa

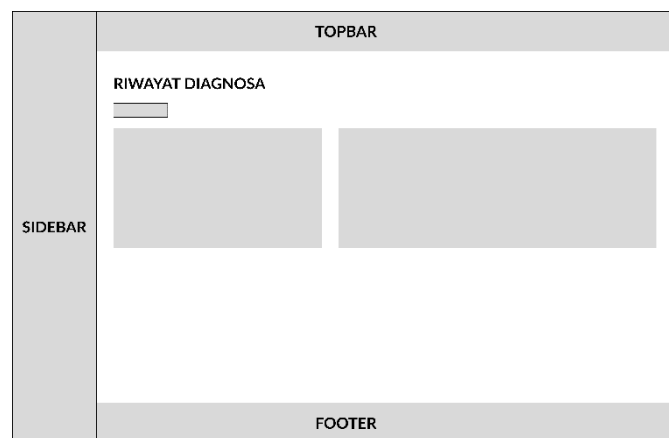
Halaman riwayat diagnosa digunakan oleh *user* untuk melihat riwayat diagnosa yang telah dilakukan. Tampilan rancangan antarmuka riwayat diagnosa dapat dilihat pada gambar 3.17.



Gambar 3. 16 *Wireframe* Riwayat Diagnosa

13. Halaman Detail Diagnosa

Halaman detail diagnosa digunakan oleh admin dan *user* untuk melihat detail hasil diagnosa. Tampilan rancangan antarmuka detail diagnosa dapat dilihat pada gambar 3.18.

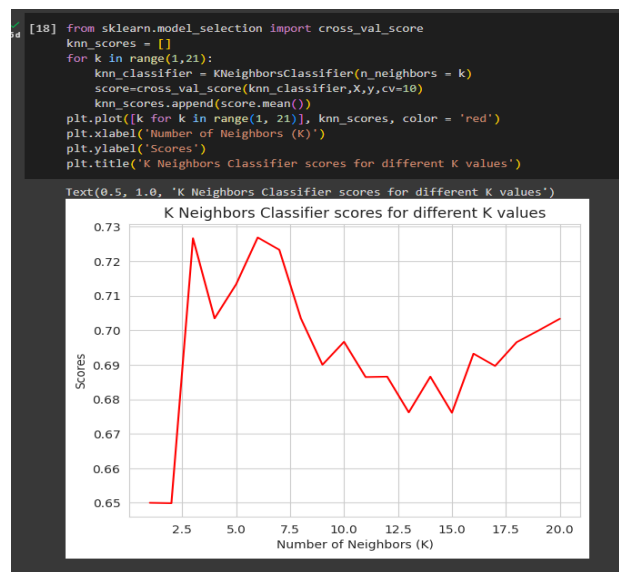


Gambar 3. 17 *Wireframe* Detail Diagnosa

3.5 Algoritma *K Nearest Neighbor*

Pada sistem diagnostik jantung yang akan dibangun pada penelitian ini akan digunakan sebuah algoritma untuk melakukan diagnostik apakah data yang dimasukkan oleh pengguna didiagnosis terindikasi penyakit jantung atau tidak. Secara singkat algoritma KNN akan memilih data dari tetangga terdekat sesuai dengan jumlah K yang telah ditentukan.

Nilai K pada penelitian ini akan menggunakan bantuan dari Google Collabs dengan bahasa pemrograman *python* dan *library KNN* untuk memilih nilai K yang paling optimal berdasarkan data yang digunakan, yaitu dataset diagnostik jantung yang didapatkan dari *open datasouce* kaggle sebanyak 601 data dan juga untuk mengukur tingkat ke akurasian dari model dengan nilai K yang telah didapatkan. Nilai K yang didapatkan dapat dilihat pada gambar 3.19.



Berdasarkan gambar 3.19 didapatkan nilai K yang optimal, yaitu $K = 3$. Dari nilai K yang didapatkan ini nantinya akan dimasukkan ke dalam sistem sebagai nilai K yang optimal. Adapun perhitungan menggunakan algoritma KNN dengan nilai $K = 3$ adalah sebagai berikut.

Data latih dapat dilihat pada tabel 3.4 berikut.

Tabel 3. 4 *Sample Data KNN*

Umur	JK	Tekanan Darah	Kolesterol	Denyut Nyadi Max	Diagnosis
69	1	160	234	131	0
69	0	140	239	151	0
66	0	150	226	114	0
65	1	138	282	174	1

61	1	134	234	145	1
58	0	136	319	152	1

Diketahui terdapat 6 data seperti data yang disajikan pada tabel 3.4 di atas sebagai data latih. Terdapat 3 data yang mendiagnosis merupakan indikasi penyakit jantung yang diberi label 1 dan lainnya dengan label 0 yang merupakan tidak terindikasi penyakit jantung. Jenis kelamin 1 menyatakan sebagai laki-laki dan 0 menyatakan perempuan. Diketahui data yang akan diuji adalah sebagai berikut.

56	1	120	240	168	?
----	---	-----	-----	-----	---

Langkah pertama yang akan dilakukan adalah mencari jarak antara data uji dengan tiap data latih yang ada pada tabel 3.1 dengan menggunakan rumus *euclidean distance*. Perhitungan jarak dengan *euclidean distance* adalah sebagai berikut.

$$distance = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$$

Sehingga perhitungannya dengan menggunakan rumus *euclidean distance* adalah sebagai berikut.

distance 1

$$= \sqrt{(69 - 56)^2 + (1 - 1)^2 + (160 - 120)^2 + (234 - 240)^2 + (131 - 168)^2}$$

$$= 56,3382641$$

distance 2

$$= \sqrt{(69 - 56)^2 + (0 - 1)^2 + (140 - 120)^2 + (239 - 240)^2 + (151 - 168)^2}$$

$$= 29,3257566$$

distance 3

$$= \sqrt{(66 - 56)^2 + (0 - 1)^2 + (150 - 120)^2 + (226 - 240)^2 + (114 - 168)^2}$$

$$= 64,132675$$

distance 4

$$= \sqrt{(65 - 56)^2 + (1 - 1)^2 + (138 - 120)^2 + (282 - 240)^2 + (174 - 168)^2}$$

$$= 46,9574275$$

distance 5

$$= \sqrt{(61 - 56)^2 + (1 - 1)^2 + (134 - 120)^2 + (234 - 240)^2 + (145 - 168)^2}$$

$$= 28,0356915$$

distance 6

$$= \sqrt{(58 - 56)^2 + (0 - 1)^2 + (136 - 120)^2 + (319 - 240)^2 + (152 - 168)^2}$$

$$= 82,2070557$$

Setelah menghitung jarak antara data latih dengan data uji langkah selanjutnya adalah mengelompokkan data tersebut dan mengurutkannya secara *ascending*.

Data	Label	Jarak	Urutan
1	Sehat	56,3382641	4
2	Sehat	29,3257566	2
3	Sehat	64,132675	5
4	Penyakit Jantung	46,9574275	3
5	Penyakit Jantung	28,0356915	1
6	Penyakit Jantung	82,2070557	6

Setelah data diurutkan dan dikelompokkan maka diambil jumlah data sebesar jumlah K. Jumlah K yang ditetapkan adalah sebesar 3 sehingga diambil 3 data teratas, yaitu sebagai berikut.

Data	Label	Jarak	Urutan
5	Penyakit Jantung	28,0356915	1
2	Sehat	29,3257566	2
4	Penyakit Jantung	46,9574275	3

