

**KLASIFIKASI PENYAKIT DENGAN GEJALA DEMAM PADA ANAK
MENGUNAKAN ALGORITMA *GENETIC MODIFIED K-NEAREST NEIGHBOR***

TUGAS AKHIR

Tugas Akhir ini sebagai salah satu syarat untuk memperoleh gelar sarjana Informatika Universitas Pembangunan Nasional
“Veteran” Yogyakarta



Disusun Oleh:

MUCHTADHO DHILA ARAFAT

123160108

**PROGRAM STUDI INFORMATIKA
JURUSAN INFORMATIKA
FAKULTAS TEKNIK INDUSTRI
UNIVERSITAS PEMBANGUNAN NASIONAL “VETERAN”
YOGYAKARTA
2021**

PENGESAHAN PEMBIMBING

KLASIFIKASI PENYAKIT DENGAN GEJALA DEMAM PADA ANAK MENGUNAKAN ALGORITMA *GENETIC MODIFIED K-NEAREST NEIGHBOR*

Disusun oleh:

Muchtadho Dhila Arafat
123160108


Telah diperiksa dan disetujui oleh pembimbing untuk di seminarkan
pada tanggal:

oleh:

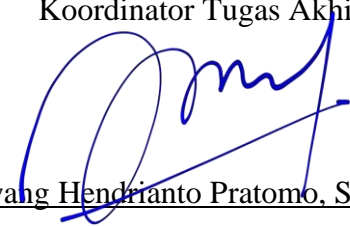
Menyetujui,
Pembimbing I

Menyetujui,
Pembimbing II


Oliver S Simanjuntak, S.T., M.Eng.
NIK. 2 8305 11 0300 1


Dr. Novrido Charibaldi, S.Kom., M.Kom.
NIK. 2 6811 96 00661

Mengetahui,
Koordinator Tugas Akhir


Dr. Awang Hendrianto Pratomo, S.T., M.T.
NIP. 1977 07 25 2005 01 1001

KATA PENGANTAR

Assalamu 'alaikum Wr.Wb

Puji syukur atas kehadiran Allah SWT yang telah melimpahkan rahmat, hidayah, dan inayah-Nya, sehingga penulis dapat menyelesaikan Tugas Akhir dengan judul “Klasifikasi Penyakit Dengan Gejala Demam Pada Anak Menggunakan Algoritma *Genetic Modified K-Nearest Neighbor*”.

Skripsi ini disusun untuk memenuhi salah satu syarat kelulusan di Program Studi Teknik Informatika Universitas Pembangunan Nasional Veteran Yogyakarta, dan dalam proses penyusunan tugas akhir ini, penulis mendapatkan banyak sekali bantuan, bimbingan serta dukungan dari berbagai pihak, sehingga dalam kesempatan ini penulis juga bermaksud menyampaikan rasa terima kasih kepada:

1. Allah SWT yang senantiasa memberikan petunjuk, kemudahan dan kesehatan.
2. Orang Tua dan anggota keluarga yang telah sabar untuk mendukung dan selalu mendoakan penulis dalam menyelesaikan tugas akhir ini.
3. Bapak Oliver Samuel Simanjuntak, S.Kom.M.Eng, selaku Pembimbing I yang selalu memberi bimbingan, masukan dan waktunya untuk membimbing penulis.
4. Bapak Dr.Novrido Charibaldi, S.Kom., M.Kom., selaku pembimbing II yang juga memberikan banyak perhatian serta waktunya untuk membimbing penulis.
5. Seluruh Dosen dan Pegawai Teknik Informatika UPN”Veteran”Yogyakarta atas segala bantuan dan kemudahannya.
6. Keluarga besar teknik informatika khususnya angkatan 2016 yang telah memberikan dukungan berupa doa.
7. Seluruh pihak yang telah memberikan semangat dan motivasi serta bantuan dalam penulisan tugas akhir.
8. Serta pihak-pihak lain yang tidak mungkin penulis sebutkan satu-persatu.

Semoga Allah SWT memberi balasan yang setimpal kepada semuanya. Penulis berharap skripsi yang telah disusun ini bisa memberikan sumbangsih untuk menambah pengetahuan para pembaca, dan akhir kata, dalam rangka perbaikan selanjutnya, penulis akan terbuka terhadap saran dan masukan dari semua pihak karena penulis menyadari skripsi yang telah disusun ini memiliki banyak sekali kekurangan.

Wassalamu 'alaikum Wr.Wb

DAFTAR ISI

HALAMAN JUDUL	i
HALAMAN PENGESAHAN PEMBIMBING.....	ii
KATA PENGANTAR.....	iii
DAFTAR ISI	iv
DAFTAR GAMBAR.....	vii
DAFTAR TABEL	viii
DAFTAR PUSTAKA	
 BAB I PENDAHULUAN	 1
1.1 Latar Belakang Masalah	2
1.2 Rumusan Masalah.....	2
1.3 Batasan Masalah	2
1.4 Tujuan Penelitian.....	2
1.5 Manfaat Penelitian	3
1.6 Metodologi Penelitian dan Pengembangan Sistem	3
1.6.1 Metodologi Penelitian.....	3
1.6.1 Metodologi Pengembangan Sistem	4
1.7 Sistematika Penulisan	5
 BAB II TINJAUAN PUSTAKA	 6
2.1 Pengertian Klasifikasi.....	6
2.2 Klasifikasi Penyakit Bergejala Demam	7
2.2.1 Demam Berdarah	7
2.2.2 Demam Tifoid.....	7
2.2.3 Infeksi Saluran Pernafasan Akut (ISPA)	8
2.2.4 Infeksi Saluran Kencing (ISK)	8
2.3 Sistem Pakar	8
2.4 Algoritma Genetika	8
2.4.1 Struktur Umum Algoritma Genetika	9
2.4.2 Komponen Utama Algoritma Genetika.....	10
2.5 Algoritma Modified K-Nearest Neighbor	11
2.5.1 Perhitungan Jarak Euclidean	12
2.5.2 Perhitungan Nilai Validitas	12
2.5.3 Perhitungan Weighted Voting	12
2.6 Studi Pustaka	13
 BAB III METODOLOGI PENELITIAN DAN PENGEMBANGAN SISTEM.....	 15
3.1 Metodologi Penelitian.....	16
3.2 Pengumpulan Data.....	17
3.2.1 Studi Pustaka	17
3.2.2 Preprocessing Data	17
3.3 Analisis Kebutuhan	19
3.3.1 Kebutuhan Fungsional	19
3.3.2 Kebutuhan Non Fungsional	19
3.3.3 Analisis Algoritma Genetika	20
3.3.4 Analisis Algoritma MKNN	23
3.4 Proses Desain.....	26
3.4.1 Perancangan Sistem.....	26

3.4.2 Perancangan Proses	27
3.4.3 Perancangan Basis Data.....	42
3.4.3.1 Entity Relationship Diagram (ERD).....	42
3.4.3.2 Data Flow Diagram (DFD).....	43
3.4.3.3 Relasi Antar Tabel	46
3.4.4 Struktur Tabel.....	46
3.4.5 Perancangan Antarmuka.....	49

DAFTAR GAMBAR

Gambar 1.1	Metode Waterfall.....	5
Gambar 2.1	Diagram Alir Algoritma Genetika.....	9
Gambar 2.2	Kromosom.....	10
Gambar 3.1	Kerangka Kerja Penelitian.....	16
Gambar 3.2	Parameter Individu MKNN	20
Gambar 3.3	Gen Individu k.....	20
Gambar 3.4	Proses Pindah Silang	22
Gambar 3.5	Proses Mutasi	22
Gambar 3.6	Arsitektur Sistem.....	26
Gambar 3.7	Flowchart Sistem.....	27
Gambar 3.8	Flowchart Algoritma Genetika.....	29
Gambar 3.9	Flowchart Populasi Awal	30
Gambar 3.10	Flowchart Fitness	31
Gambar 3.11	Flowchart Seleksi	32
Gambar 3.12	Flowchart Crossover.....	33
Gambar 3.13	Flowchart Mutasi.....	34
Gambar 3.14	Flowchart Fitness Mutasi	36
Gambar 3.15	Flowchart Populasi Baru	37
Gambar 3.16	Flowchart MKNN.....	38
Gambar 3.17	Flowchart Validitas	39
Gambar 3.18	Flowchart Euclidean.....	40
Gambar 3.19	Flowchart Weighted Voting	41
Gambar 3.20	Entity Relationship Diagram (ERD)	42
Gambar 3.21	Data Flow Diagram Level 0	43
Gambar 3.22	Data Flow Diagram Level 1	44
Gambar 3.23	Data Flow Diagram Level 2 Proses Olahdata	45
Gambar 3.24	Relasi Antar Tabel.....	46
Gambar 3.25	Rancangan Halaman Beranda	50
Gambar 3.26	Rancangan Halaman Login	50
Gambar 3.27	Rancangan Halaman Penyakit.....	51
Gambar 3.28	Rancangan Halaman Gejala	51
Gambar 3.29	Rancangan Halaman Latih Pasien.....	52
Gambar 3.30	Rancangan Halaman Diagnosa.....	52
Gambar 3.31	Rancangan Halaman Gejala Pasien	53
Gambar 3.32	Rancangan Halaman Uji Pasien	54
Gambar 3.33	Rancangan Halaman Diagnosa Pasien	54
Gambar 3.34	Rancangan Halaman Gejala Uji Pasien.....	55
Gambar 3.35	Rancangan Halaman GMKNN.....	55
Gambar 3.36	Rancangan Halaman Akurasi	56
Gambar 3.37	Rancangan Halaman Genetika	56
Gambar 3.38	Rancangan Halaman MKNN.....	57

DAFTAR TABEL

Tabel 2.1	State of the Art	14
Tabel 3.1	Bobot Gejala Demam	17
Tabel 3.2	Lanjutan Bobot Gejala Demam	18
Tabel 3.3	Spesifikasi Perangkat Keras dan Perangkat Lunak	19
Tabel 3.4	Spesifikasi Perangkat Keras dan Perangkat Lunak	19
Tabel 3.5	Nilai Fitness Tiap Individu	21
Tabel 3.6	Optimasi MKNN Menggunakan Algoritma Genetika.....	23
Tabel 3.7	Data Pasien	23
Tabel 3.8	Hasil Perhitungan Euclidean Antar Data Latih	24
Tabel 3.9	Nilai Validitas	24
Tabel 3.10	Hasil Klasifikasi	26
Tabel 3.11	Struktur Tabel Data Admin	46
Tabel 3.12	Struktur Tabel Data Penyakit	47
Tabel 3.13	Struktur Tabel Data Training Pasien	47
Tabel 3.14	Struktur Tabel Data Gejala	47
Tabel 3.15	Struktur Tabel Data Diagnosa	47
Tabel 3.16	Struktur Tabel Data Bobot.....	48
Tabel 3.17	Struktur Tabel Data Uji Pasien.....	48
Tabel 3.18	Struktur Tabel Data Uji Diagnosa	48
Tabel 3.19	Struktur Tabel Data Gejala Pasien	49
Tabel 3.20	Struktur Tabel Data Uji Gejala Pasien	49

BAB I

PENDAHULUAN

1.1 Latar Belakang

Demam merupakan suatu keadaan suhu tubuh diatas normal ($37,5^{\circ}\text{C}$) sebagai akibat peningkatan pusat pengatur suhu di *hipotalamus*. Keadaan hiperpireksia atau hipertermi (demam tinggi) adalah kenaikan suhu tubuh sampai 41°C atau lebih. Penyakit – penyakit yang ditandai dengan adanya demam dapat menyerang sistem tubuh. Selain itu demam mungkin berperan dalam meningkatkan perkembangan imunitas spesifik dan non spesifik dalam membantu pemulihan atau pertahanan terhadap infeksi (Sodikin dan Wardiyah, 2016).

Demam sering ditemui pada anak-anak karena tubuhnya masih rentan terhadap penyakit. Badan Kesehatan Dunia (WHO) mengemukakan jumlah kasus demam di seluruh dunia mencapai 18-34 juta. Anak merupakan kelompok yang paling rentan terkena demam. Sebagian besar demam pada anak merupakan akibat dari perubahan pada pusat panas (*termoregulasi*) di *hipotalamus*. Faktor demografi dimana Indonesia termasuk wilayah beriklim tropis merupakan tempat yang baik bagi kuman untuk berkembang biak. Salah satu penyakit tropis yang sering dijumpai adalah *influenza*, demam berdarah, demam *tifoid*, *tuberculosis*, dan campak.

Kejadian demam berdarah di Indonesia banyak ditemukan pada kelompok usia 2-14 tahun. Hal ini berkaitan dengan kebiasaan nyamuk *Aedes aegypti* yang aktif menggigit pada siang hari dengan puncak aktivitas pukul 08.00-12.00 dan 15.00-17.00. Pada waktu tersebut anak-anak sedang aktif melakukan kegiatan.

Demam tifoid merupakan masalah Kesehatan masyarakat yang sering ditemukan pada anak di negara berkembang. Angka kesakitan demam tifoid di Indonesia paling banyak terjadi pada kelompok usia 2-15 tahun. Sekitar 81,7 kasus per 100.000 anak di Indonesia mengalami demam tifoid.

Infeksi Saluran Pernafasan Akut merupakan penyakit utama penyebab kematian bayi dan balita di Indonesia. Tingkat mortalitas ISPA sangat tinggi pada bayi dan anak-anak terutama pada negara-negara berkembang dengan pendapatan per kapita rendah hingga menengah.

Penelitian yang berkaitan dengan klasifikasi penyakit bergejala demam (DBD, Malaria, dan Tifoid) pernah dilakukan oleh (Wafiyah dkk, 2017) menggunakan metode *Modified K-Nearest Neighbor* menghasilkan akurasi sebesar 88.55% pada pengujian variasi data latih. Perbedaan penelitian penulis dengan penelitian sebelumnya adalah penulis menambahkan jenis-jenis penyakit bergejala demam seperti ISPA, dan gangguan saluran kencing. Peneliti juga akan melakukan penambahan data latih dan data uji, beserta solusi dan cara pencegahan penyakit bergejala demam agar informasi dan data pembelajaran sistem semakin maksimal.

Peneliti akan melakukan pengembangan algoritma yang dilakukan dengan menambahkan algoritma genetika pada algoritma MKNN sehingga menjadi algoritma

GM-KNN (*Genetic Modified K-Nearest Neighbor*). Algoritma GM-KNN adalah algoritma yang digunakan untuk mengatasi masalah optimasi. Berdasarkan penelitian sebelumnya, algoritma Genetika pada algoritma GM-KNN dapat meningkatkan nilai akurasi serta mengurangi kompleksitas yang tidak dimiliki oleh algoritma MKNN untuk melakukan klasifikasi penyakit bergejala demam (Bambang & Nurul, 2018). Keunggulan dari algoritma GMKNN adalah dapat menentukan nilai k optimal pada MKNN dengan otomatis tanpa harus mencoba satu persatu dalam menentukan nilai k .

Penelitian dengan menggunakan algoritma MKNN masih memiliki kekurangan untuk menentukan kategori penyakit dengan gejala demam pada anak disertai gejala lain. Disamping itu untuk menentukan nilai K dan komposisi jumlah data latih dan data uji masih dapat di optimalkan atau dikembangkan untuk mendapatkan akurasi yang lebih baik dengan menggunakan algoritma genetika.

Oleh karena itu, pada penelitian ini penulis akan mengembangkan salah satu algoritma untuk klasifikasi penyakit dengan gejala demam pada anak yang menggunakan metode K-NN menjadi GM-KNN untuk menyelesaikan masalah optimasi dengan data yang didapat dari dokumen rekam medis. Penelitian ini juga dibangun dengan tujuan agar dapat menghasilkan nilai akurasi yang lebih tinggi dari penelitian sebelumnya, sehingga menghasilkan klasifikasi penyakit bergejala demam yang lebih akurat dengan algoritma GM-KNN. Penelitian ini selanjutnya diharapkan dapat dikembangkan sebagai proses *screening* awal penyakit bergejala demam pada anak secara mandiri untuk orang tua. *Screening* penyakit dapat membantu meningkatkan kewaspadaan penyakit bergejala demam yang kemungkinan diderita oleh anak.

1.2 Rumusan Masalah

- Berdasarkan penjabaran masalah pada latar belakang, dapat dirumuskan permasalahan dalam penelitian yaitu bagaimana akurasi dalam klasifikasi penyakit bergejala demam menggunakan algoritma *Genetic Modified K-Nearest Neighbor*?
- Bagaimana menentukan nilai k terbaik menggunakan Algoritma Genetika agar kinerja MKNN lebih baik.

1.3 Batasan Masalah

Penelitian ini terbatas pada permasalahan di bawah ini:

- Data yang digunakan berasal dari Rumah Sakit Umum Daerah Bagas Waras Kabupaten Klaten yang terdiri atas 200 Data yaitu 160 Data Latih dan 40 Data Uji.
- Klasifikasi demam menggunakan algoritma *Genetic Modified K-Nearest Neighbor* berdasarkan 4 penyakit bergejala demam.

1.4 Tujuan Penelitian

Tujuan dari penelitian ini adalah sebagai berikut yaitu :

- Meningkatkan nilai akurasi yang dihasilkan oleh algoritma *Genetic Modified K-Nearest Neighbor* dalam klasifikasi penyakit bergejala demam.
- Menghitung Nilai k terbaik menggunakan Algoritma Genetika.

1.5 Manfaat Penelitian

Manfaat dari penelitian untuk mengetahui nilai akurasi dari Algoritma Genetika dan MKNN untuk klasifikasi penyakit bergejala demam yang sering terjadi pada anak (demam berdarah, demam tifoid, ISPA, dan ISK) sehingga dapat memudahkan orang tua untuk melakukan *screening* awal penyakit secara mandiri sebelum pergi ke fasilitas pelayanan kesehatan untuk mendapatkan penegakan diagnosis dari dokter maupun laboratorium.

1.6 Metodologi Penelitian dan Pengembangan Sistem

Metodologi penelitian yang digunakan dalam penelitian menggunakan pendekatan penelitian kuantitatif. Metodologi penelitian ini terdiri dari metode pengumpulan data dan metode pengembangan sistem.

1.6.1 Metodologi Penelitian

a. Studi Literatur

Melakukan studi dari literatur-literatur yang mendukung penelitian. Literatur dapat berupa buku, paper, jurnal hingga website resmi. Serta mempelajari materi terkait dengan penyelesaian masalah yang berkaitan dengan penelitian ini. Tujuan studi literatur yaitu untuk memahami metode yang akan digunakan dalam penyelesaian masalah dan menghindari duplikasi dari penelitian yang telah dilakukan sebelumnya.

b. Pengumpulan Data

Data yang akan digunakan pada penelitian ini bersumber dari RSUD Bagas Waras Klaten, data yang digunakan melewati tahap *preprocessing*.

c. Analisis dan Perancangan

Analisis dan perancangan sistem akan melakukan analisis terhadap kebutuhan sistem yang akan dibangun. Setelah dilakukan analisis dilanjutkan dengan perancangan sistem yang akan dibangun seperti *data flow diagram*, *entity relationship diagram*, *flowchart* dan tampilan *user interface*.

d. Implementasi Sistem

Implementasi sistem dilakukan dengan mengikuti rancangan sistem yang telah dibuat. Sistem akan dibangun dengan *php*.

e. Pengujian dan Evaluasi

Pada tahap ini dilakukan penulisan laporan dari hasil-hasil atau temuan-temuan pada penelitian yang dilakukan. Pada tahap ini termasuk juga penyusunan laporan, mengumpulkan hasil, serta pengambilan kesimpulan dan saran.

f. Penulisan Laporan

Pada tahap ini dilakukan penulisan laporan dari hasil-hasil atau temuan-temuan pada penelitian yang dilakukan. Pada tahap ini termasuk juga penyusunan laporan, mengumpulkan hasil, serta pengambilan kesimpulan dan saran.

1.6.2 Metodologi Pengembangan Sistem

Pada penelitian ini menggunakan metode pengembangan sistem *waterfall*. *Waterfall* merupakan salah satu metode dalam *SDLC* (*System Development Life Cycle*) yang mempunyai ciri khas pengerjaan setiap fase dalam *waterfall* harus diselesaikan terlebih dahulu sebelum melanjutkan ke fase selanjutnya. Metode *waterfall* adalah pengerjaan dari suatu sistem dilakukan secara berurutan atau secara linear (Yusuf & Fitri, n.d.). Secara garis besar metode *waterfall* mempunyai langkah-langkah sebagai berikut: Analisis kebutuhan, Desain, Pembuatan program, Pengujian, dan Penerapan serta Pemeliharaan (Kadir, 2003).

1. Analisis kebutuhan

Analisis kebutuhan bertujuan untuk memahami kebutuhan dari sistem yang akan dibangun sehingga dapat dikembangkan sistem yang sesuai dengan kebutuhan tersebut. Proses pengumpulan kebutuhan dilakukan secara intensif untuk menspesifikasikan kebutuhan sistem agar dapat dipahami sistem seperti apa yang dibutuhkan oleh pengguna.

2. Desain sistem.

Desain sistem bertujuan untuk membuat rancangan dari hasil analisa yang telah dilakukan pada tahapan sebelumnya. Perancangan yang akan dilakukan pada tahapan desain sistem ini meliputi perancangan diagram *data flow diagram*, perancangan basis data, dan perancangan antar muka.

3. Pembuatan Program

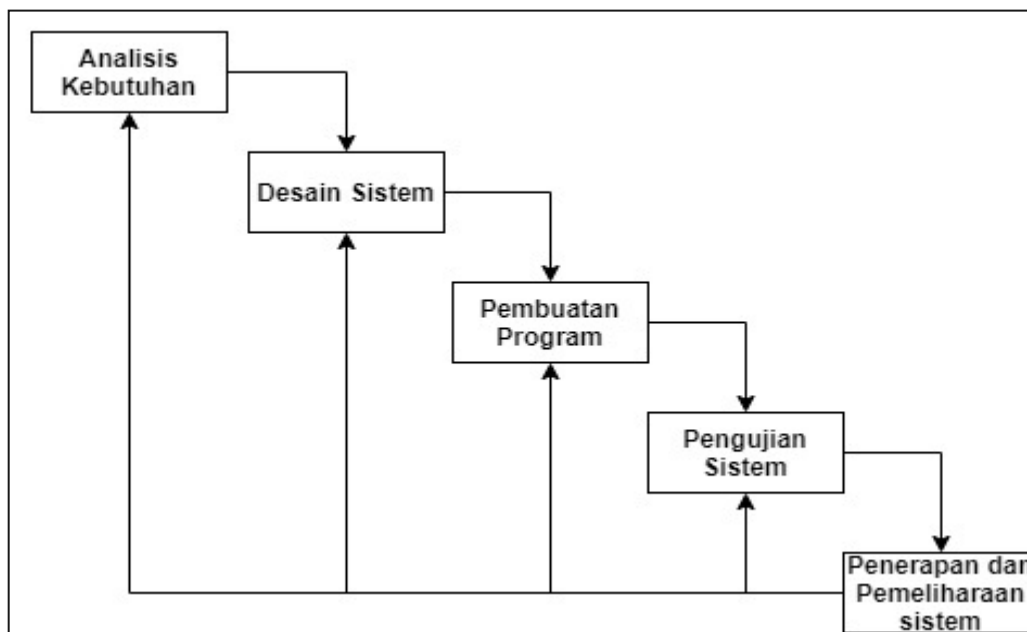
Pembuatan program disebut tahap implementasi sistem atau *coding*. Dengan kata lain, pada tahap pembuatan program ini dilakukan implementasi hasil rancangan ke dalam baris-baris kode program yang dapat dimengerti oleh komputer.

4. Pengujian Sistem

Pengujian fokus pada sistem dari segi logic dan fungsional untuk memastikan bahwa semua bagian sudah diuji. Hal ini dilakukan untuk meminimalisir kesalahan (*error*) dan memastikan keluaran yang dihasilkan sesuai dengan yang diinginkan.

5. Penerapan dan Pemeliharaan Sistem

Tidak menutup kemungkinan sebuah perangkat lunak mengalami perubahan ketika sudah dikirimkan ke user. Perubahan bisa terjadi karena adanya kesalahan yang muncul dan tidak terdeteksi saat pengujian. Tahap pemeliharaan dapat mengulangi proses pengembangan mulai dari analisis spesifikasi untuk perubahan sistem yang sudah ada, tapi tidak untuk membuat sistem baru.



Gambar 1.1 Metode Waterffal (Kadir 2003)

1.7 Sistematika Penulisan

Sistematika penulisan yang digunakan dalam menyusun laporan penelitian ini adalah sebagai berikut :

- BAB I** Pada bagian ini membahas tentang latar belakang masalah, perumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, metodologi penelitian, dan sistematika penulisan.
- BAB II** Tinjauan pustaka memuat tentang dasar teori yang digunakan untuk analisis dan perancangan sistem serta implementasi pada penelitian ini. Selain itu juga sebagai bahan referensi dan pondasi untuk memperkuat argumentasi dalam penelitian ini. Teori-teori yang sesuai dengan penelitian ini antara lain penyakit bergejala demam pada anak, algoritma genetika, dan algoritma MK-NN
- BAB III** Pembahasan bab ini adalah untuk menjabarkan proses-proses perancangan dan analisis implementasi Algoritma *Genetic Modified K-Nearest Neighbor* dalam melakukan klasifikasi penyakit bergejala demam.
- BAB IV** Bab yang membahas tentang implementasi dari algoritma *Genetic Modified K-Nearest Neighbor* yang mengklasifikasikan penyakit bergejala demam pada anak. Pembahasan pada bab ini terdiri atas scenario pengujian, hasil dari pengujian akurasi dari algoritma *Genetic Modified K-Nearest Neighbor* dalam melakukan klasifikasi demam dan analisis dari hasil yang didapatkan.
- BAB V** Bab ini membahas tentang kesimpulan yang didapatkan dari analisis pengujian algoritma *Genetic Modified K-Nearest Neighbor* dalam melakukan klasifikasi penyakit bergejala demam pada anak serta saran untuk pengembangan sistem selanjutnya.

BAB II

TINJAUAN PUSTAKA

2.1 Pengertian Klasifikasi

Klasifikasi adalah teknik untuk merancang fungsi berdasarkan hasil pengamatan dari data dan atribut dari data tersebut sehingga dapat dilakukan pemetaan terhadap data yang belum memiliki kelas kedalam data yang telah terklasifikasi sesuai dengan aturan-aturan yang diberikan. Terdapat banyak algoritma yang digunakan untuk mengklasifikasi data, antara lain *fuzzy*, *artificial neural network*, *support vector machine* dan *k-nearest neighbor* (Wafiyah dkk 2017).

Klasifikasi menggunakan algoritma tersebut secara umum dilakukan berdasarkan 3 tahapan yaitu:

- a. Perancangan Model

Proses membangun solusi untuk menyelesaikan masalah berdasarkan data yang telah terklasifikasi (Data latih).

- b. Implementasi Model

Proses penentuan kelas untuk data uji berdasarkan model fungsi dan parameter-parameter data yang telah ditentukan pada tahap perancangan.

- c. Evaluasi Model

Proses yang bertujuan untuk melakukan evaluasi terhadap hasil implementasi model fungsi dalam mengklasifikasikan data uji berdasarkan parameter-parameter yang telah ditetapkan.

2.2 Klasifikasi Penyakit Bergejala Demam

Demam adalah proses alami tubuh untuk melawan infeksi yang masuk ke dalam tubuh ketika suhu meningkat melebihi suhu tubuh normal ($>37,5^{\circ}\text{C}$). Demam adalah proses alami tubuh untuk melawan infeksi yang masuk ke dalam tubuh. Demam terjadi pada suhu $> 37,2^{\circ}\text{C}$, biasanya disebabkan oleh infeksi (bakteri, virus, jamur atau parasit), penyakit *autoimun*, keganasan, ataupun obat – obatan (Surinah dalam Hartini, 2015).

Menurut (Nurarif 2015) klasifikasi demam adalah sebagai berikut:

- a. Demam Septic

Suhu badan berangsur naik ketinggian yang tinggi sekali pada malam hari dan turun kembali ketinggian diatas normal pada pagi hari. Sering disertai keluhan menggigil dan berkeriangat. Bila demam yang tinggi tersebut turun ketinggian yang normal dinamakan juga demam hektik.

- b. Demam Remiten

Suhu badan dapat turun setiap hari tetapi tidak pernah mencapai suhu badan normal. Penyebab suhu yang mungkin tercatat dapat mencapai dua derajat dan tidak sebesar perbedaan suhu yang dicatat demam septik.

- c. Demam Intermitten

Suhu badan turun ketinggian yang normal selama beberapa jam dalam satu hari. Bila demam seperti ini terjadi dalam dua hari sekali disebut tersiana

dan bila terjadi dua hari terbebas demam diantara dua serangan demam disebut kuartana.

d. Demam Kontinyu

Variasi suhu sepanjang hari tidak berbeda lebih dari satu derajat. Pada tingkat demam yang terus menerus tinggi sekali disebut hiperpireksia.

e. Demam Siklik

Terjadi kenaikan suhu badan selama beberapa hari yang diikuti oleh beberapa periode bebas demam untuk beberapa hari yang kemudian diikuti oleh kenaikan suhu seperti semula.

2.2.1 Demam Berdarah

Demam berdarah *dengue* (DBD) adalah penyakit infeksi yang disebabkan oleh virus *dengue* dan mengakibatkan spektrum manifestasi klinis yang bervariasi antara yang paling ringan, demam *dengue* (DD), DBD dan demam *dengue* yang disertai renjatan atau *dengue shock syndrome* (DSS) ditularkan nyamuk *Aedes aegypti* dan *Ae. albopictus* yang terinfeksi (Candra, 2010).

Gejala awal dari penyakit demam berdarah juga ditandai dengan kondisi tubuh yang mulai lemah, lesu, gelisah, rasa nyeri pada ulu hati, terjadi pendarahan pada kulit berupa titik pendarahan (petechia). Dalam kondisi terburuk penderita demam berdarah mengalami pendarahan pada hidung.

2.2.2 Demam Tifoid

Demam tifoid merupakan penyakit infeksi sistemik akut yang mengenai sistem *retikuloendotelial*, kelenjar *limfe* saluran cerna, dan kandung empedu. Disebabkan terutama oleh *Salmonella enterica serovar typhi* (*S. typhi*) dan menular melalui jalur *fekal-oral* (Pediatri, 2010).

Gejala demam tifoid sangat bervariasi, dari gejala ringan yang tidak memerlukan perawatan hingga gejala berat yang memerlukan perawatan. Masa inkubasi demam tifoid berlangsung antara 10-14 hari. Pada awal periode penyakit ini, penderita demam tifoid mengalami demam. Sifat demam adalah meningkat perlahan-lahan terutama pada sore hingga malam hari (Widodo et al, 2014).

Pada saat demam tinggi, dapat disertai dengan gangguan sistem saraf pusat, seperti kesadaran menurun, penurunan kesadaran mulai dari apatis sampai koma. Gejala sistemik lain yang menyertai adalah nyeri kepala, *malaise*, *anoreksia*, *nausea*, *myalgia*, nyeri perut dan radang tenggorokan. Gejala *gastrointestinal* pada kasus demam tifoid sangat bervariasi. Pasien dapat mengeluh diare, konstipasi, atau konstipasi kemudian disusul dengan diare, lidah tampak kotor dengan warna putih ditengah, hepatomegaly dan splenomegaly (Sumarno et al, 2008).

2.2.3 Infeksi Saluran Pernafasan Akut (ISPA)

Infeksi Saluran Pernapasan Akut (ISPA) merupakan infeksi yang terdapat pada saluran nafas atas maupun saluran nafas bagian bawah. Infeksi akut ini menyerang salah satu bagian/lebih dari saluran nafas mulai hidung sampai *alveoli* termasuk adneksanya (sinus, rongga telinga tengah, pleura) (Depkes RI, 2012). Menurut WHO, ISPA adalah penyakit saluran pernapasan atas atau bawah, biasanya menular, yang dapat menimbulkan berbagai spektrum penyakit yang berkisar dari penyakit tanpa gejala atau infeksi ringan sampai

penyakit yang parah dan mematikan, tergantung pada patogen penyebabnya, faktor lingkungan, dan faktor pejamu (Ching et al., Bulletin WHO 2007).

Menurut Depkes RI tahun 2012, klasifikasi ISPA dapat dibedakan berdasarkan berat ringannya gejala yang ditimbulkan, yaitu tanda dan gejala ringan (bukan pneumonia), sedang (pneumonia sedang/pneumonia), dan berat (pneumonia berat). Penyakit batuk-pilek seperti rinitis, faringitis, tonsilitis dan penyakit jalan napas bagian atas lainnya digolongkan sebagai bukan pneumonia.

2.2.4 Infeksi Saluran Kemih (ISK)

Infeksi saluran kemih (ISK) adalah suatu penyakit yang dimana terbentuknya mikroorganisme dalam jumlah banyak di saluran kemih. ISK merupakan kondisi dimana terdapat mikroorganisme dalam urin yang jumlahnya sangat banyak dan mampu menimbulkan infeksi pada saluran kemih. Gejala yang dapat dirasakan pasien ISK, yaitu mual, muntah, demam, disuria, dan terdesak kencing yang biasanya terjadi bersamaan disertai nyeri suprapubik dan daerah pelvis (Israr, 2009).

2.3 Sistem Pakar

Sistem pakar adalah sistem berbasis komputer yang menggunakan pengetahuan, fakta dan teknik penalaran dalam memecahkan masalah yang biasanya hanya dapat dipecahkan oleh seorang pakar dalam bidang tersebut

Sistem pakar merupakan cabang dari *Artificial Intelligence* (AI) yang sudah lama karena sistem ini telah mulai dikembangkan pada pertengahan tahun 1960. Sistem pakar yang muncul pertama kali adalah *General-purpose problem solver* (GPS) yang dikembangkan oleh Newl dan Simon. Sampai saat ini sudah banyak sistem pakar yang dibuat, seperti MYCIN, DENDRAL, XCON & XSEL, SOPHIE, Prospector, FOLIO, DELTA, dan sebagainya.

Sistem pakar (*expert system*) secara umum adalah sistem yang berusaha mengadopsi pengetahuan manusia ke komputer, agar komputer dapat menyelesaikan masalah seperti yang biasa dilakukan oleh para ahli (Silmi dkk 2013).

2.4 Algoritma Genetika

Algoritma genetika merupakan algoritma evolusioner yang paling populer yang mana algoritma ini menggunakan prinsip dasar dari seleksi alam yang diperkenalkan oleh Charles Darwin. Algoritma genetika diterapkan sebagai pendekatan untuk mengidentifikasi pencarian nilai dan solusi bagi berbagai permasalahan optimasi (Gorunescu, 2011).

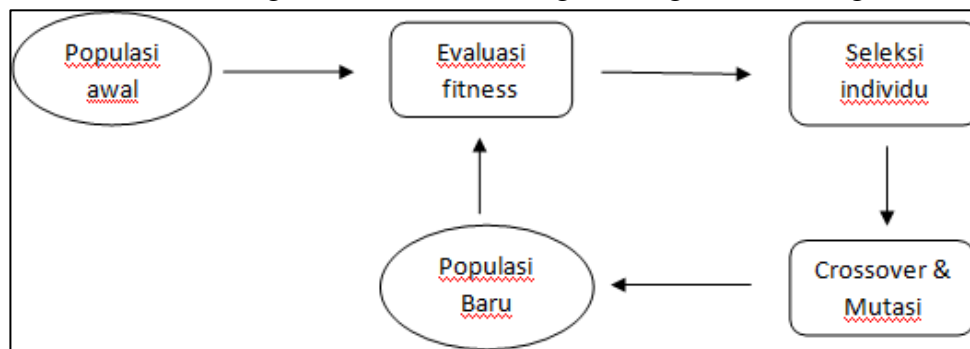
Definisi Penting dalam Algoritma Genetika :

- a. Gen, merupakan nilai yang menyatakan satuan dasar yang membentuk suatu arti tertentu dalam satu kesatuan gen (kromosom).
- b. Kromosom atau individu, merupakan gabungan dari gen-gen yang membentuk arti tertentu dan menyatakan solusi yang mungkin dari suatu permasalahan.
- c. Populasi, merupakan sekumpulan individu yang akan diproses bersama dalam satu siklus evolusi.
- d. Fitness, menyatakan seberapa baik nilai dari suatu individu atau solusi yang didapatkan.

- e. Crossover, merupakan proses pertukaran atau kawin silang antara gen-gen dari dua induk tertentu.
- f. Mutasi, merupakan proses pergantian salah satu gen dengan nilai tertentu.
- Generasi, merupakan urutan iterasi di mana beberapa kromosom bergabung.
- g. Offspring, merupakan kromosom baru yang dihasilkan setelah melewati suatu generasi.

2.4.1 Struktur Umum Algoritma Genetika

Menurut Charles L Karr (1999). Kromosom ini merupakan suatu solusi yang masih berbentuk simbol. Populasi awal dibangun secara acak, sedangkan populasinya merupakan hasil evolusi kromosom-kromosom melalui iterasi yang disebut dengan istilah generasi. Pada setiap generasi kromosom akan melalui proses evaluasi dengan menggunakan alat ukur yang disebut fungsi *fitness*. Nilai *fitness* dari suatu kromosom akan menunjukkan kualitas kromosom dalam populasi tersebut. Proses ini dapat direpresentasikan dalam algoritma sederhana Algoritma genetika sebagai berikut :



Gambar 2.1 Diagram alir algoritma genetika

Secara umum struktur dari suatu algoritma genetik dapat mendefenisikan dengan langkah-langkah sebagai berikut:

- a. Membangkitkan populasi awal
Populasi awal ini dibangkitkan secara random sehingga didapatkan solusi awal. Populasi itu sendiri terdiri atas sejumlah kromosom yang merepresentasikan solusi yang diinginkan.
- b. Membentuk generasi baru
Untuk membentuk generasi baru, digunakan operator reproduksi/seleksi, *crossover* dan mutasi. Proses ini dilakukan berulang-ulang sehingga didapatkan jumlah kromosom yang cukup untuk membentuk generasi baru dimana generasi baru ini merupakan representasi dari solusi baru. Generasi baru ini dikenal dengan istilah anak (*offspring*).
- c. Evaluasi Solusi
Pada tiap generasi, kromosom akan melalui proses evaluasi dengan menggunakan alat ukur yang dinamakan *fitness*. Nilai *fitness* suatu kromosom menggambarkan kualitas kromosom dalam populasi tersebut. Proses ini akan mengevaluasi setiap populasi dengan

menghitung nilai *fitness* setiap kromosom dan mengevaluasinya sampai terpenuhi kriteria berhenti. Bila kriteria berhenti belum terpenuhi maka akan dibentuk lagi generasi baru dengan mengulangi langkah 2. Beberapa kriteria berhenti sering digunakan antara lain: berhenti pada generasi tertentu, berhenti setelah dalam beberapa generasi berturut-turut didapatkan nilai *fitness* tertinggi tidak berubah, berhenti dalam n generasi tidak didapatkan nilai *fitness* yang lebih tinggi.

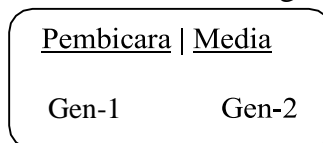
2.4.2 Komponen Utama Algoritma Genetika

Ada enam komponen utama dalam Algoritma Genetika (AG), yaitu:

a. Teknik Pengkodean

Menurut Desiani (2006), pengkodean adalah suatu teknik untuk menyatakan populasi awal sebagai calon solusi suatu masalah ke dalam suatu kromosom sebagai suatu kunci pokok persoalan ketika menggunakan algoritma genetika. Teknik pengkodean meliputi pengkodean gen dari kromosom. Gen merupakan bagian dari kromosom. Satu gen biasanya akan mewakili satu variable.

Gen dapat direpresentasikan dalam bentuk *code* angka, yang terdiri dari Gen-1 sebagai Pembicara, Gen-2 sebagai media.



Gambar 2.2 Kromosom

b. Fungsi Evaluasi

Dalam setiap generasi dilakukan evaluasi fungsi tujuan sehingga menghasilkan nilai *fitness*. Nilai *fitness* adalah sebuah parameter dalam menentukan baik tidaknya sebuah individu dalam populasi. Nilai *fitness* menunjukkan kemampuan bertahan dari suatu solusi untuk tetap ada pada generasi selanjutnya dengan menghasilkan keturunan yang memiliki sifat yang sama dengan menurunkan informasi genetik melalui mekanisme evolusioner (Kusumaningrum, 2017).

c. Seleksi

Seleksi adalah proses memilih individu terbaik dari sebuah populasi dengan mencari kromosom dengan nilai *fitness* terbaik, dan kemudian dilanjutkan ke proses berikutnya yaitu crossover dan mutasi (Irawati, 2010). Metode seleksi yang akan digunakan adalah Tournament selection. Teknik ini melibatkan beberapa turnamen diantara N_u individu yang 24 dipilih secara acak dari populasi, individu terbaik dengan nilai *fitness* tertinggi akan menjadi pemenang. Pemenang dari proses Tournament selection ini akan digunakan untuk tahap Algoritma Genetika selanjutnya yaitu crossover dan mutasi. Jumlah turnamen bisa disesuaikan untuk menentukan tekanan pada proses seleksi, jika jumlah

turnamen semakin besar maka kemungkinan individu dengan nilai fitness kecil untuk terpilih akan menjadi semakin kecil

d. Persilangan (Crossover)

Crossover merupakan proses di dalam Algoritma Genetika yang bekerja untuk meng-gabungkan dua kromosom parent menjadi kromosom baru (offspring) pada suatu waktu (Suhartono, 2015). Tidak semua kromosom parent mengalami crossover, banyaknya pasangan induk yang akan mengalami persilangan akan ditentukan dengan nilai PC yaitu probabilitas crossover. Schaffer, Caruana, Eshelman, dan Das (1989) menyebutkan nilai Pc optimal terletak pada range 0,75-0,95. Teknik crossover dapat dilakukan dalam berbagai cara yaitu one point crossover atau yang disebut satu titik potong dan n-point crossover (Irawati, 2010). Teknik one point crossover adalah teknik memilih satu point secara acak dari parent 1 kemudian digabungkan dengan bagian kedua parent 2. N-point crossover dilakukan dengan cara menukarkan semua gen diantara titik potong dengan induk.

e. Mutasi (*mutation*)

Mutasi adalah proses penggantian gen dengan nilai inversinya. Gen 0 menjadi 1, dan gen 1 menjadi 0, proses ini dilakukan secara acak pada posisi gen tertentu pada individu-individu yang terpilih untuk dimutasikan (Irawati, 2010). Banyaknya individu yang mengalami mutasi ditentukan oleh besarnya probabilitas mutasi Pm. Nilai Pm yang sering pakai pada implementasi Algoritma Genetika berada pada range 0,001 dan 0,05 (Davis, 1991 dalam Ismail & Irhamah, 2008).

2.5 Algoritma *Modified K-Nearest Neighbor* (MKNN)

Algoritma *modified k-nearest neighbor* (MKNN) merupakan pengembangan dari metode KNN dengan penambahan beberapa proses yaitu, perhitungan nilai validitas dan perhitungan bobot. *Algoritma k-nearest neighbor* (KNN) merupakan algoritma *clustering* yang sangat sederhana dengan cara mengelompokkan data baru dengan K tetangga terdekat.

2.5.1 Perhitungan Jarak *Euclidian*

Untuk mendefinisikan jarak antara dua titik yaitu titik pada data *training* (x) dan titik pada data *testing* (y) maka digunakan rumus Euclidean, seperti yang ditunjukkan pada persamaan :

$$d(x_i, y_i) = \sqrt{\sum_{i=0}^n (x_i - y_i)^2}$$

Persamaan di atas adalah Perhitungan Jarak Euclidian dengan d adalah jarak antara titik pada data training x dan titik data testing y yang akan diklasifikasi, dimana $x=x_1, x_2, \dots, x_i$ dan $y=y_1, y_2, \dots, y_i$ dan I merepresentasikan nilai atribut serta n merupakan dimensi atribut.

2.5.2 Perhitungan Nilai Validitas

Dalam algoritma MKNN, setiap data pada data training harus divalidasi terlebih dahulu pada awalnya. Validitas setiap data tergantung pada setiap tetangganya. Proses validasi dilakukan untuk semua data pada data training. Setelah dihitung validitas tiap data maka nilai validitas tersebut digunakan sebagai informasi lebih mengenai data tersebut. Persamaan yang digunakan untuk menghitung nilai validitas pada setiap data training adalah seperti persamaan dibawah ini

$$Validity(x) = \frac{1}{H} \sum_{i=0}^n S(lbl(x), lbl(Ni(x)))$$

Perhitungan Nilai Validitas

Fungsi S digunakan untuk menghitung kesamaan antara titik x dan data ke-i dari tetangga terdekat. Yang dituliskan dalam persamaan di bawah ini mendefinisikan fungsi S pada persamaan berikut :

$$S(a, b) = \begin{cases} 1 & a = b \\ 0 & a \neq b \end{cases}$$

Persamaan Fungsi S

Keterangan:

a = kelas a pada data training.

b = kelas lain selain a pada data training.

2.5.3 Perhitungan *Weighted Voting*

Dalam metode MKNN, pertama weight masing-masing tetangga dihitung dengan menggunakan $1 / (de + 0.5)$. Kemudian, Validitas dari tiap data pada data training dikalikan dengan weighted berdasarkan pada jarak Euclidian. Dalam metode MKNN, weight voting tiap tetangga Persamaan .

$$W(i) = Validity(i) \times \frac{1}{de + a}$$

Perhitungan *Weighted Voting*

Dimana:

$W(i)$: Perhitungan *Weighted Voting*

$Validity(i)$: Nilai Validitas

de : Jarak *Euclidean*

Teknik *weighted voting* ini mempunyai pengaruh yang lebih penting terhadap data yang mempunyai nilai validitas lebih tinggi dan paling dekat dengan data. Selain itu, dengan mengalikan validitas dengan jarak dapat mengatasi kelemahan dari setiap data yang mempunyai jarak dengan weight yang memiliki banyak masalah dalam outlier. Jadi, algoritma MKNN diusulkan secara signifikan lebih kuat daripada metode KNN tradisional yang didasarkan hanya pada jarak. (Parvin, 2008).

2.6 Studi Pustaka

Adapun penelitian yang telah dilakukan sebelum yang berkaitan dengan penelitian ini yang dijadikan sebagai acuan dan referensi:

1. Penelitian yang dilakukan oleh (Wafiyah et al., 2017) dengan judul **Implementasi Algoritma Modified K-Nearest Neighbor (MKNN) untuk Klasifikasi Penyakit Demam** bertujuan mengklasifikasikan penyakit bergejala demam menggunakan algoritma MKNN. Metode tersebut mempelajari pola dari data hasil pemeriksaan sebelumnya berdasarkan 15 gejala penyakit dengan proses perhitungan jarak *euclidian*, perhitungan nilai validitas dan perhitungan *weighted voting* yang hasil akhirnya digunakan untuk penetapan kelas klasifikasi berdasarkan nilai K yang telah ditentukan. Pada penelitian ini rata-rata akurasi untuk pengujian pengaruh nilai K terhadap akurasi sebesar 88.55%.

2. Penelitian yang dilakukan oleh (Rizky Karunia, 2018) dengan judul **Sistem Klasifikasi Penyakit dengan Gejala Demam Pada Anak Menggunakan Metode Fuzzy K-Nearest Neighbor (FK-NN)**. Sistem Klasifikasi Penyakit Anak Dengan Gejala Demam Pada Anak Menggunakan Metode Fuzzy K-Nearest Neighbor dengan memiliki pembagian data latih sebanyak 80%, nilai K sebesar 10, dan nilai M sebesar 2, serta pemilihan kelas ada 2 yaitu dengue dan pnemonia, menghasilkan tingkat akurasi sebesar 0.833.

3. Penelitian yang dilakukan oleh (Nuramilus et al., 2017) dengan judul **Identifikasi Penyakit dengan Gejala Awal Demam Menggunakan K-Nearest Neighbor (K-NN)**. Pada penelitian menggunakan metode KNN dan *Certainty Factor*. Pada penelitian ini data latih dan data uji yang digunakan berjumlah 143 data. Berdasarkan hasil pengujian variasi nilai K didapatkan akurasi sebesar 88.37%. Pada pengujian variasi data latih didapatkan akurasi sebesar 86.04%. Pada pengujian perbandingan metode, metode *k-nearest neighbor-certainty factor* menghasilkan akurasi sebesar 84.79%.

4. Penelitian yang dilakukan oleh (Dananjaya et al., 2019) dengan judul **Decision Support System for Classification of Early Childhood Diseases Using Principal Component Analysis and K-Nearest Neighbors Classifier**. Pada penelitian ini menggunakan metode PCA dan KNN. Hasil evaluasi sistem dengan menggunakan 150 data uji menunjukkan bahwa sistem klasifikasi dengan menerapkan PCA dan KNN Classifier memiliki nilai akurasi sebesar 86%.

5. Penelitian yang dilakukan oleh (Mutrofin et al., 2014) dengan judul **Optimasi Teknik Klasifikasi Modified K Nearest Neighbor Menggunakan Algoritma Genetika**. Pada penelitian ini menggunakan metode Algoritma Genetika dan *Modified K-Nearest Neighbor*. Hasil dari kinerja Algoritma Genetika pada MKNN yaitu dapat menentukan nilai *k* optimal pada MKNN dengan otomatis.

Tabel 2.1 State of the Art

Peneliti	Kasus	Metode	Hasil
Fakihat Wafiyah, Nurul Hidayat, Rizal Setya Perdana (2017)	Penyakit Bergejala Demam	MKNN (Modified K-Nearest Neighbor)	rata-rata akurasi untuk pengujian pengaruh nilai K terhadap akurasi sebesar 88.55%.
Rizky Karunia (2018)	Penyakit Bergejala Demam	FKNN	Tingkat Akurasi 83.3%
Elsa Nuramilus, Rekyan Regasari, Acmad (2017)	Sistem Pakar Penyakit Bergejala Demam	KNN & Certainty Factor	KNearest Neighbor – Certainty Factor dengan K-Nearest Neighbor menghasilkan akurasi yang sama yaitu 84.79%. Untuk J48 yang menghasilkan akurasi 81.39%. Naïve Bayes yang menghasilkan akurasi sebesar 88.37%.
Damar Dananjaya, Indah Werdiningsih, Rini Semiati (2019)	Sistem Pendukung Keputusan untuk Klasifikasi Penyakit Anak	PCA & KNN	Sistem klasifikasi dengan menerapkan PCA dan KNN memiliki akurasi sebesar 86%
Siti Mutrofin, Abidatul Izzah, dkk (2014)	Optimasi Teknik Klasifikasi	Algoritma Genetika & MKNN	Keunggulan dari algoritma GMKNN adalah dapat menentukan nilai k optimal pada MKNN dengan otomatis, tanpa harus mencoba satu persatu dalam menentukan nilai k.
Muchtadho Dhillah Arafat (2021)	Klasifikasi Penyakit Gejala Demam	Algoritma Genetika & MKNN	Nilai k yang dihasilkan paling optimal adalah k=2 dengan akurasi 90%

Berdasarkan penelitian terdahulu yang mempunyai karakteristik yang relatif sama dalam hal tema penelitian. Namun penelitian ini memiliki perbedaan dalam segi metode penelitian, dan hasil dari penelitian. Pada penelitian sebelumnya dengan objek penelitian yang sama yaitu klasifikasi penyakit bergejala demam, dalam penggunaan metode, langkah dan proses pun berbeda.

Berikut perbandingan penelitian ini dengan penelitian sebelumnya :

1. Research gap pada penelitian sebelumnya adalah akurasi yang belum mencapai 90%.
2. Pada penelitian (Wafiyah dkk, 2017) dan (Nuramilus dkk, 2017) menggunakan 15 gejala dan 3 penyakit. Sedangkan pada penelitian ini menggunakan 22 gejala dan 4 penyakit.
3. Penelitian yang di lakukan sebelumnya belum bisa menentukan nilai k menggunakan algoritma optimasi. Sedangkan pada penelitian ini menggunakan algoritma genetika yang digunakan untuk proses optimasi nilai k, sehingga mendapatkan akurasi 90%.
4. Penelitian (Mutrofin dkk, 2014) menggunakan algoritma genetika untuk proses optimasi nilai k. Pada penelitian ini mendapatkan nilai k optimal melalui proses algoritma genetika.

BAB III

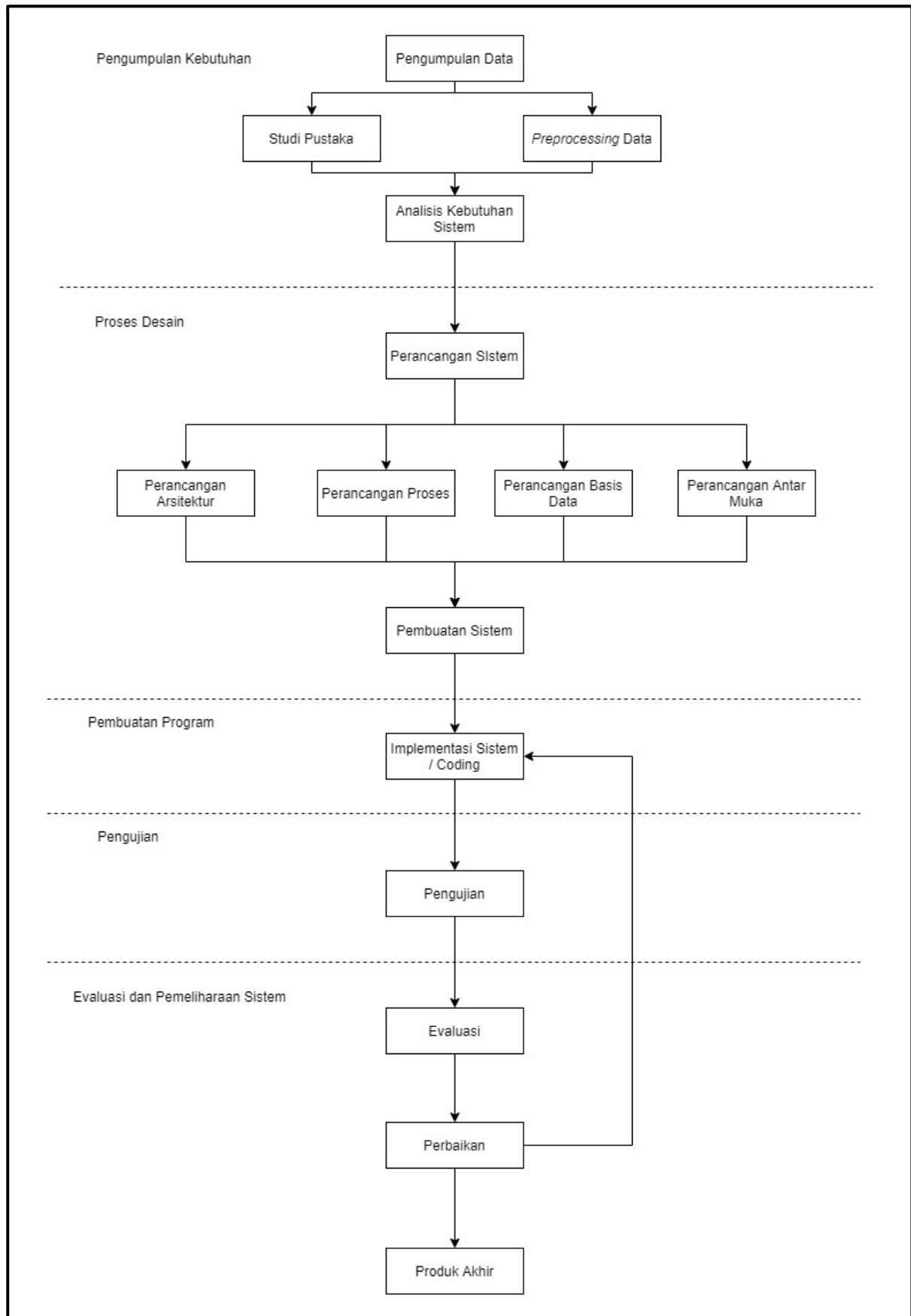
METODOLOGI PENELITIAN DAN PENGEMBANGAN SISTEM

3.1 Metodologi Penelitian

Pada bagian ini akan membahas tentang metodologi penelitian dan pengembangan sistem yang akan digunakan sebagai bahan acuan dalam proses pembuatan penelitian Klasifikasi Penyakit Dengan Gejala Demam Pada Anak Menggunakan Algoritma *Genetic Modified K-Nearest Neighbor*

Metodologi penelitian yang digunakan untuk pengumpulan data pada penelitian ini adalah metode penelitian kuantitatif. Metode penelitian kuantitatif digunakan karena penelitian ini menggunakan teknik pengambilan sampel data. Data penelitian yang akan diambil berupa data dari rekam medis pada rumah sakit.

Metode pengembangan sistem pada penelitian ini menggunakan metode *waterfall*. dengan tahap-tahap yang meliputi analisis, desain system, penulisan kode (coding), pengujian program dan perawatan (*maintenance*). Metode waterfall dalam penelitian ini hanya sampai dengan tahapan pengujian program.



Gambar 3.1 Kerangka Kerja Penelitian

3.2 Pengumpulan Data

Pengumpulan data merupakan proses yang dilakukan untuk mencari informasi yang dibutuhkan oleh sistem. Proses ini dilakukan supaya mendapatkan data yang *valid* dari lapangan, sehingga hasil dan kesimpulan pada penelitian ini sesuai dengan permasalahan yang dibahas.

3.2.1 Studi Pustaka

Studi pustaka merupakan salah satu tahapan pengumpulan data yang digunakan untuk mempelajari berbagai sumber referensi dari buku, jurnal, dan sumber lain yang dapat dipercaya dan dipertanggung jawabkan.

Pada tahapan ini dilakukan pencarian informasi mengenai metode dan teknik analisis sentimen yang berkaitan dengan penelitian sebelumnya untuk pengumpulan informasi tentang masalah penelitian ini. Pengumpulan informasi dibutuhkan dalam mencari referensi dan mendapatkan informasi kelebihan dan kekurangan metode dan teknik analisis tersebut dan menjadi bahan evaluasi yang dapat mendukung pengembangan sistem pada penelitian ini.

3.2.2 Preprocessing Data

Peneliti mengambil data dari Rumah Sakit Umum Daerah Bagas Waras Klaten. Data yang didapatkan dari pihak rumah sakit berupa arsip diagnosa penyakit pada anak berdasarkan gejala yang diderita. Penelitian ini menggunakan 22 gejala penyakit beserta bobot yang telah diperoleh oleh Kepala Dinas Kesehatan dr.Cahyono Widodo.,M.Kes untuk masing-masing gejala dalam beberapa tingkatan dengan total data sebanyak 200 data pasien penderita penyakit demam berdarah, demam tifoid, ISPA, ISK yang memiliki kesamaan gejala. Berikut adalah tabel untuk pembobotan masing-masing gejala pada demam.

Tabel 3.1 Bobot Gejala Demam

NO	VARIABEL GEJALA	SKOR
G1	Demam Intermittent (naik-turun)	
	Ya	0.8
	Tidak	0
G2	Demam Menggigil	
	Berat	0.85
	Sedang	0.5
	Tidak Ada	0
G3	Lama Demam	
	>7 Hari Berat	0.75
	4-7 Hari Sedang	0.5
	1-3 Hari Ringan	0.2
G4	Sakit Kepala	
	Berat	0.8
	Sedang	0.2
	Tidak Ada	0
G5	Mual	
	Berat	0.7
	Sedang	0.3
	Tidak ada	0
G6	Muntah	
	Berat	0.7
	Sedang	0.3

Tabel 3.2 Lanjutan Bobot Gejala Demam

	Tidak ada	0
G7	Nyeri pada Tulang atau Sendi	
	Ya	0.85
	Tidak ada	0
G8	Nyeri Perut	
	Ya	0.75
	Tidak Ada	0
G9	Nyeri Ulu Hati	
	Ya	0.8
	Tidak Ada	0
G10	Bintik Merah pada kulit	
	Ya	0.9
	Tidak Ada	0
G11	Lidah kotor (Coated Tongue)	
	Ya	0.85
	Tidak ada	0
G12	Kejang	
	Ya	0.7
	Tidak	0
G13	Batuk	
	Berat	0.75
	Sedang	0.5
	Tidak ada	0
G14	Pilek	
	Ya	0.8
	Tidak	0
G14	Sesak Nafas	
	Berat	0.8
	Sedang	0.6
	Tidak ada	0
G16	Sulit BAB (Konstipasi)	
	Ya	0.7
	Tidak	0
G17	BAB Cair	
	Ya	0.75
	Tidak	0
G18	BAK Nyeri (Nyeri Kencing)	
	Ya	0.9
	Tidak	0
G19	BAK Kemerahan	
	Ya	0.85
	Tidak	0
G20	Nafsu Makan Turun	
	Ya	0.75
	Tidak	0
G21	Badan Terasa Lemas	
	Ya	0.85
	Tidak	0
G22	Mimisan	
	Ya	0.8
	Tidak	0

3.3. Analisis Kebutuhan

Analisis kebutuhan sistem dilakukan dengan tujuan mendeskripsikan apa yang harus dilakukan oleh sistem. Pada analisis kebutuhan terdapat dua jenis yaitu analisis kebutuhan fungsional dan analisis kebutuhan non fungsional.

3.3.1 Kebutuhan Fungsional

Kebutuhan fungsional pada sistem merupakan gambaran dan informasi yang ada pada sistem serta gambaran apa saja yang dapat dilakukan oleh sistem. Sistem ini akan digambarkan dengan menggunakan *data flow diagram*, *entity relationship diagram* dan *flowchart*. Kebutuhan fungsional pada sistem ini antara lain:

- Sistem ini dapat mengklasifikasikan penyakit berdasarkan *input* gejala oleh *user*.
- Sistem ini dapat melihat hasil kalkulasi pada metode algoritma genetika dan MKNN.
- Sistem ini dapat menampilkan hasil klasifikasi penyakit berdasarkan gejala-gejala yang ada.

3.3.2 Kebutuhan Non Fungsional

Kebutuhan non fungsional pada sistem dilakukan untuk mengetahui spesifikasi penggunaan aplikasi. Spesifikasi yang dibutuhkan aplikasi ini antara lain: perangkat keras dan perangkat lunak.

- Analisis kebutuhan perangkat keras (*hardware*):

Perangkat Keras (Hardware)

Perangkat keras yang dibutuhkan untuk mengembangkan sistem pada penelitian ini dengan spesifikasi yang dapat dilihat pada Tabel 3.3.

Tabel 3.3 Spesifikasi Kebutuhan Perangkat Keras dan Perangkat Lunak

No	Perangkat Keras	Keterangan
1	Processor	Intel i7-7700 HQ
2	RAM	16 GB DDR5
3	Storage	Hardisk 1TB
4	Graphic	NVIDIA GeForce GTX 1050 4GB VRAM
5	Perangkat Input dan Output	Keyboard, Mouse, dan Monitor
6	Koneksi Internet	Wifi dan Kuota Internet

- Analisis kebutuhan perangkat lunak (*software*):

Perangkat lunak yang dibutuhkan untuk mengembangkan sistem pada penelitian ini dapat dilihat pada Tabel 3.4.

Tabel 3.4 Spesifikasi Kebutuhan Perangkat Keras dan Perangkat Lunak

No	Perangkat Lunak	Keterangan
1	Operating System	Windows 10 64 bit
2	Bahasa PHP	Bahasa pemrograman
3	Sublime Text	Perangkat lunak code editor
4	Google Colab	Keyboard, Mouse, dan Monitor
5	XAMPP	Perangkat lunak sebagai server lokal
6	MySQL	Tempat penyimpanan database
7	Google Chrome	Web browser
8	Draw.io	Perangkat lunak untuk membuat desain diagram dan rancang interface

3.3.3 Analisis Algoritma Genetika

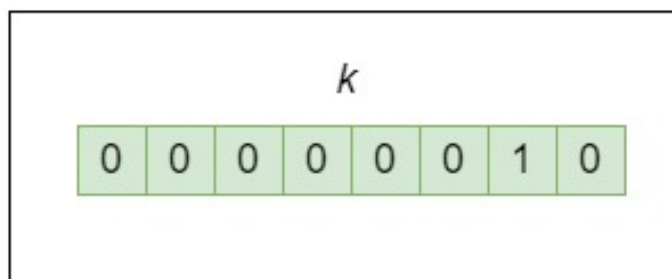
Algoritma genetika dimaksudkan untuk mendapatkan hasil akurasi MKNN yang lebih akurat. Langkah awal yang dilakukan adalah membangkitkan individu sebanyak 6 buah. Individu yang dibangkitkan memiliki delapan buah gen yang mempresentasikan besar nilai k .



Gambar 3.2 Parameter individu MKNN

Pada gambar 3.x menunjukkan ilustrasi sebuah individu yang mempresentasikan nilai k . Individu yang terbentuk tersebut selanjutnya akan menjalani proses algoritma genetika meliputi seleksi, crossover (pindah silang), dan mutasi agar mendapatkan nilai yang paling baik sehingga mengoptimalkan akurasi MKNN.

Pada penelitian ini jumlah individu yang dibangkitkan sebanyak 6 individu yang terdiri dari 8 gen.



Gambar 3.3 Gen individu k .

Gambar diatas merepresentasikan nilai k yang bernilai 2. Pada individu diatas gen berbentuk bilangan biner untuk mendapatkan nilai dapat menggunakan persamaan berikut :

$$\begin{aligned}
 x &= (0 \times 2^7) + (0 \times 2^6) + (0 \times 2^5) + (0 \times 2^4) + (0 \times 2^3) + (0 \times 2^2) + (1 \times 2^1) \\
 &\quad + (0 \times 2^0) \\
 &= 0 + 0 + 0 + 0 + 0 + 0 + 2 + 0 \\
 &= 2
 \end{aligned}$$

Dari hasil diatas nilai decimal dari bagian individu k adalah 2.

Setelah mendefinisikan kromosom, selanjutnya adalah menentukan nilai fitness, nilai fitness merupakan nilai yang akan dijadikan acuan untuk proses selanjutnya meliputi seleksi, pindah silang dan mutasi. Nilai fitness dalam individu ini merupakan nilai akurasi dari hasil prediksi algoritma MKNN. Ilustrasi nilai fitness untuk tiap individu dapat dilihat pada tabel berikut :

Tabel 3.5 Nilai Fitness Tiap Individu

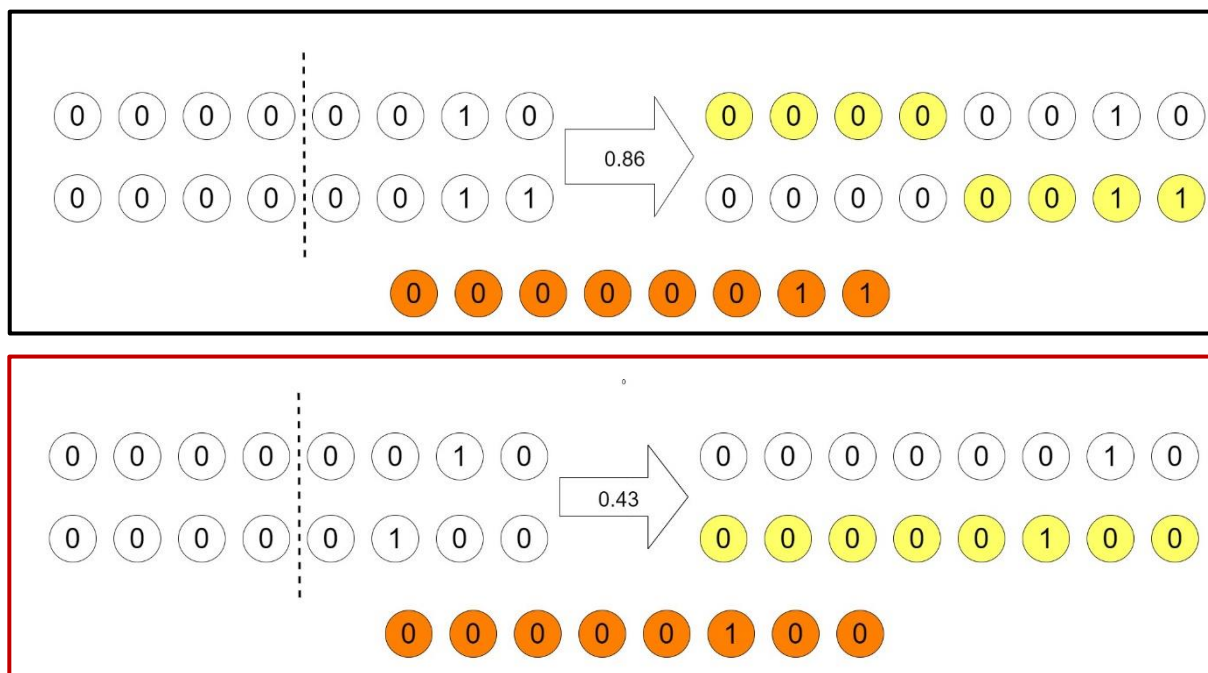
Individu ke	k	<i>Fitness</i>
1	2	47
2	5	42
3	10	40.7
4	9	40.778
5	10	40.7
6	7	40.2857

Proses seleksi yang dilakukan pada penelitian ini adalah seleksi menggunakan metode tournament selection. Tournament selection melibatkan beberapa tournament diantara N_u individu yang dipilih secara acak dari sebuah populasi, pada penelitian ini jumlah individu yang dipilih adalah sebanyak 2. Berdasarkan nilai fitness yang dihasilkan pada table 3.x nilai fitness misalkan diambil 2 individu acak dari populasi $N_1 = 47, N_1 = 42$ maka individu yang dipilih sebagai hasil seleksi adalah N_1 dengan nilai fitness 47. Proses ini akan dilakukan berulang sebanyak $N/2$ proses sehingga didapatkan orang tua untuk generasi selanjutnya sebanyak populasi.

Setelah melakukan proses seleksi, proses selanjutnya adalah proses pindah silang atau crossover. Proses ini hanya dilakukan apabila bilangan acak yang dibangkitkan kurang dari probabilitas pindah silang (P_c), kromosom yang mengalami proses pindah silang akan menjadi orang tua pada proses selanjutnya. Proses pindah silang akan dilakukan $N/2$ kali sehingga orang tua untuk proses selanjutnya akan berjumlah sama dengan populasi. Penentuan kromosom crossover adalah sebagai berikut:

1. Menentukan 2 induk dari tahap seleksi berdasarkan nilai fitness tertinggi ke terendah.
2. Nilai crossover dipilih secara acak untuk dilakukan crossing dengan crossoverrate.
3. Nilai crossoverate yang ditentukan yaitu 0.8. Jika nilai crossover lebih besar daripada crossoverrate maka terjadi penukaran untuk menghasilkan anak.
4. Apabila terjadi crossover maka kepala pada biner individu 1 sebagai kepala anak dan ekor individu 2 menjadi ekor anak. Apabila tidak terjadi crossover maka anak sama seperti individu 2.

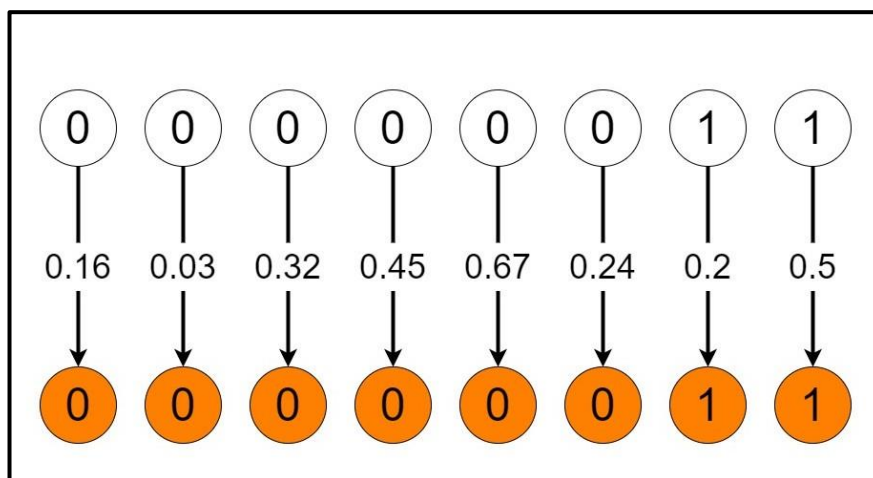
Pada tahap ini dapat diilustrasikan sebagai berikut :



Gambar 3.4 Proses Pindah Silang

Pindah silang dilakukan untuk mendapatkan individu baru dan akan digabungkan dengan orang tua hasil proses seleksi setelah dilakukan proses mutasi. Oleh karena itu jumlah populasi generasi selanjutnya tetap seperti populasi awal.

Dalam algoritma genetika, proses yang selanjutnya setelah pindah silang adalah proses mutasi. Pada proses ini dilakukannya proses mutasi yaitu penggantian gen dengan nilai inversinya. Gen 0 menjadi 1, dan gen 1 menjadi 0. Proses ini dilakukan secara acak pada posisi gen tertentu pada individu yang terpilih untuk dimutasikan. Banyaknya gen yang mengalami mutasi ditentukan oleh besarnya probabilitas mutasi. Pada sistem ini menggunakan mutation rate 0.01. Nilai mutasi akan dipilih secara acak lalu dibandingkan dengan mutation rate, apabila lebih kecil dari mutation rate maka akan terjadi mutasi. Pada tahapan ini dapat diilustrasikan sebagai berikut :



Gambar 3.5 Proses Mutasi

Pada penelitian ini jumlah individu yang dibangkitkan dibatasi sebanyak 6 kromosom dan generasi pelatihan dibatasi tidak lebih dari 5 generasi dikarenakan pertimbangan waktu eksekusi pelatihan model. Berikut merupakan data hasil decode individu dengan nilai fitness paling tinggi pada setiap generasi pada proses optimasi menggunakan Algoritma Genetika:

Tabel 3.6 Optimasi MKNN menggunakan Algoritma Genetika

Iterasi	Nilai <i>k</i> Terbaik	<i>Fitness</i>
1	5	42
2	3	45.333
3	3	45.333
4	2	47
5	2	47

3.3.4 Analisis Algoritma MKNN

Pada pengklasifikasian menggunakan MKNN (Modified K-Nearest Neighbor) terdapat tiga proses utama yaitu hitung validitas, menghitung jarak Euclidean dan menghitung weighted voting. Berikut adalah langkah-langkah klasifikasi menggunakan algoritma MKNN :

1. Proses Validitas

Proses perhitungan validitas bertujuan untuk melakukan validasi terhadap setiap data latih yang digunakan pada sistem. Pertama melakukan perbandingan kelas untuk data latih dengan data uji. Hasil perbandingan tersebut kemudian dijumlahkan dan dibagi dengan nilai *K* yang telah ditentukan.

Tabel 3.7 Data Pasien

ID	G1	G2	G3	G4	G5	G6	G7	G8	G9	G10	G11	G12	G13	G14	G15	G16	G17	G18	G19	G20	G21	G22
1	0.8	0	0.2	0.2	0.3	0.3	0	0.75	0	0	0	0	0	0	0	0	0	0	0	0	0.85	0
2	0	0	0.2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.9	0.85	0	0	0
3	0	0.5	0.5	0	0.3	0.3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0.2	0	0.3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0.2	0	0.3	0	0	0	0	0	0	0	0	0	0	0	0	0.9	0	0	0	0

Pada proses perhitungan validitas pertama kita mencari nilai Euclidean dari data latih. Perhitungan jarak Euclidean antar data latih dilakukan untuk melakukan validasi masing-masing data latih dan digunakan untuk menghitung nilai validitas.

1.1 Perhitungan jarak Euclidean data latih 1 dan data latih 2

$$\begin{aligned}
 &= \sqrt{(0.8 - 0)^2 + (0 - 0)^2 + (0.2 - 0.2)^2 + (0.2 - 0)^2 + (0.3 - 0)^2 + (0.3 - 0)^2} \\
 &= \sqrt{+(0 - 0)^2 + (0.75 - 0)^2 + (0 - 0)^2 + (0 - 0)^2 + (0 - 0)^2 + (0 - 0)^2} \\
 &= \sqrt{+(0 - 0)^2 + (0 - 0)^2 + (0 - 0)^2 + (0 - 0)^2 + (0 - 0)^2 + (0 - 0.9)^2} \\
 &= \sqrt{+(0 - 0.85)^2 + (0 - 0)^2 + (0.85 - 0)^2 + (0 - 0)^2} \\
 &= 3.68
 \end{aligned}$$

1.2 Perhitungan jarak Euclidean data latih 1 dan data latih 3

$$\begin{aligned}
 &= \sqrt{(0.8 - 0)^2 + (0 - 0.5)^2 + (0.2 - 0.5)^2 + (0.2 - 0)^2 + (0.3 - 0.3)^2} \\
 &= \sqrt{+(0.3 - 0.3)^2 + (0 - 0)^2 + (0.75 - 0)^2 + (0 - 0)^2 + (0 - 0)^2 + (0 - 0)^2} \\
 &= \sqrt{++(0 - 0)^2 + (0 - 0)^2 + (0 - 0)^2 + (0 - 0)^2 + (0 - 0)^2 + (0 - 0)^2 +} \\
 &= \sqrt{+(0 - 0)^2(0 - 0)^2 + (0 - 0)^2 + (0.85 - 0)^2 + (0 - 0)^2} \\
 &= 2.3
 \end{aligned}$$

1.3 Perhitungan jarak Euclidean data latih 1 dan data latih 4

$$\begin{aligned}
 &= \sqrt{(0.8 - 0)^2 + (0 - 0)^2 + (0.2 - 0.2)^2 + (0.2 - 0)^2 + (0.3 - 0)^2 + (0.3 - 0)^2} \\
 &= \sqrt{+(0 - 0)^2 + (0.75 - 0)^2 + (0 - 0)^2 + (0 - 0)^2 + (0 - 0)^2 + (0 - 0)^2} \\
 &= \sqrt{+(0 - 0)^2 + (0 - 0)^2 + (0 - 0)^2 + (0 - 0)^2 + (0 - 0)^2 + (0 - 0)^2} \\
 &= \sqrt{+(0 - 0)^2 + (0 - 0)^2 + (0.85 - 0)^2 + (0 - 0)^2} \\
 &= 2.06
 \end{aligned}$$

Perhitungan dilakukan untuk setiap data training seperti pada perhitungan diatas.

Tabel 3.8 Hasil Perhitungan Euclidean antar data latih.

	D1	D2	D3	D4
D1	0	3.68	2.3	2.06
D2	3.68	0	2.05	1.62
D3	2.3	2.0	0	0.43
D4	2.06	1.62	0.43	0

Nilai validitas didapatkan dari hasil perhitungan jarak Euclidean antar data latih dengan $k = 2$, yaitu nilai yang telah dihitung sebelumnya menggunakan algoritma genetika. Lalu menentukan apabila data memiliki label yang sama dengan tetangga bernilai 1 dan jika sebaliknya maka bernilai 0

Tabel 3.9 Nilai Validitas dengan $k = 2$

Data	Tetangga Terdekat	Label Benar	Nilai Validitas
D1	D3 & D4	0	0
D2	D3 & D4	1	0.5
D3	D1 & D2	0	0
D4	D1 & D2	1	0.5

2. Proses Hitung Jarak Euclidean data latih dengan data uji.

Pada proses ini akan dilakukan perhitungan jarak Euclidean data latih terhadap data uji.

2.1 Perhitungan jarak euclidean data latih 1 dan data uji 5

$$\begin{aligned}
 &= \sqrt{(0.8 - 0)^2 + (0 - 0)^2 + (0.2 - 0.2)^2 + (0.2 - 0)^2 + (0.3 - 0.3)^2 + (0.3 - 0)^2} \\
 &= \sqrt{+(0 - 0)^2 + (0.75 - 0)^2 + (0 - 0)^2 + (0 - 0)^2 + (0 - 0)^2 + (0 - 0)^2} \\
 &= \sqrt{+(0 - 0)^2 + (0 - 0)^2 + (0 - 0)^2 + (0 - 0)^2 + (0 - 0)^2 + (0 - 0.9)^2} \\
 &= \sqrt{+(0 - 0)^2 + (0 - 0)^2 + (0.85 - 0)^2 + (0 - 0)^2} \\
 &= 2.87
 \end{aligned}$$

2.2 Perhitungan jarak euclidean data latih 2 dan data uji 5

$$\begin{aligned}
 &= \sqrt{(0-0)^2 + (0-0)^2 + (0.2-0.2)^2 + (0-0)^2 + (0-0.3)^2 + (0-0)^2} \\
 &= \sqrt{+(0-0)^2 + (0-0)^2 + (0-0)^2 + (0-0)^2 + (0-0)^2 + (0-0)^2} \\
 &= \sqrt{+(0-0)^2 + (0-0)^2 + (0-0)^2 + (0-0)^2 + (0-0)^2 + (0.9-0.9)^2} \\
 &= \sqrt{+(0.85-0)^2 + (0-0)^2 + (0-0)^2 + (0-0)^2} \\
 &= 0.81
 \end{aligned}$$

2.3 Perhitungan jarak euclidean data latih 3 dan data uji 5

$$\begin{aligned}
 &= \sqrt{(0-0)^2 + (0.5-0)^2 + (0.5-0.2)^2 + (0-0)^2 + (0.3-0.3)^2 + (0.3-0)^2} \\
 &= \sqrt{+(0-0)^2 + (0-0)^2 + (0-0)^2 + (0-0)^2 + (0-0)^2 + (0-0)^2} \\
 &= \sqrt{+(0-0)^2 + (0-0)^2 + (0-0)^2 + (0-0)^2 + (0-0)^2 + (0-0.9)^2} \\
 &= \sqrt{+(0-0)^2 + (0-0)^2 + (0-0)^2 + (0-0)^2} \\
 &= 1.24
 \end{aligned}$$

2.4 Perhitungan jarak euclidean data latih 4 dan data uji 5

$$\begin{aligned}
 &= \sqrt{(0-0)^2 + (0-0)^2 + (0-0.2)^2 + (0-0)^2 + (0.3-0.3)^2 + (0.3-0)^2} \\
 &= \sqrt{+(0-0)^2 + (0-0)^2 + (0-0)^2 + (0-0)^2 + (0-0)^2 + (0-0)^2} \\
 &= \sqrt{+(0-0)^2 + (0-0)^2 + (0-0)^2 + (0-0)^2 + (0-0)^2 + (0-0.9)^2} \\
 &= \sqrt{+(0-0)^2 + (0-0)^2 + (0-0)^2 + (0-0)^2} \\
 &= 0.81
 \end{aligned}$$

Perhitungan jarak Euclidean antara data latih dan data uji dilakukan untuk mendapatkan nilai yang di proses dalam perhitungan weighted voting.

Proses Hitung Weighted Voting

Perhitungan weighted voting dilakukan dengan menghitung hasil perhitungan jarak Euclidean antara data latih dan data uji serta nilai validitas data latih dan diambil nilai terbesar untuk menemukan kelas klasifikasi. Berikut perhitungan nilai weighted voting :

1. Perhitungan nilai weighted voting antara data uji 5 dan data latih 1.

$$Validity(1) \times \frac{1}{de + 0.5} = 0 \times \frac{1}{2.87 + 0.5} = 0$$

2. Perhitungan nilai weighted voting antara data uji 5 dan data latih 2.

$$Validity(1) \times \frac{1}{de + 0.5} = 0.5 \times \frac{1}{0.81 + 0.5} = 0.38$$

3. Perhitungan nilai weighted voting antara data uji 5 dan data latih 3.

$$Validity(1) \times \frac{1}{de + 0.5} = 0 \times \frac{1}{1.24 + 0.5} = 0$$

4. Perhitungan nilai weighted voting antara data uji 5 dan data latih 4.

$$Validity(1) \times \frac{1}{de + 0.5} = 0.5 \times \frac{1}{0.81 + 0.5} = 0.38$$

Tahap terakhir yaitu menentukan kelas data uji dengan mengurutkan hasil perhitungan weighted voting berdasarkan nilai terbesar lalu menjumlahkan data dengan kelas yang sama. Hasil terbesar diambil sebagai kelas data uji tersebut. Berdasarkan perhitungan diatas maka didapatkan seperti pada tabel dibawah

Tabel 3.10 Hasil Klasifikasi

Infeksi Saluran Kencing	0.76
Demam Tifoid	0
Demam Berdarah	0

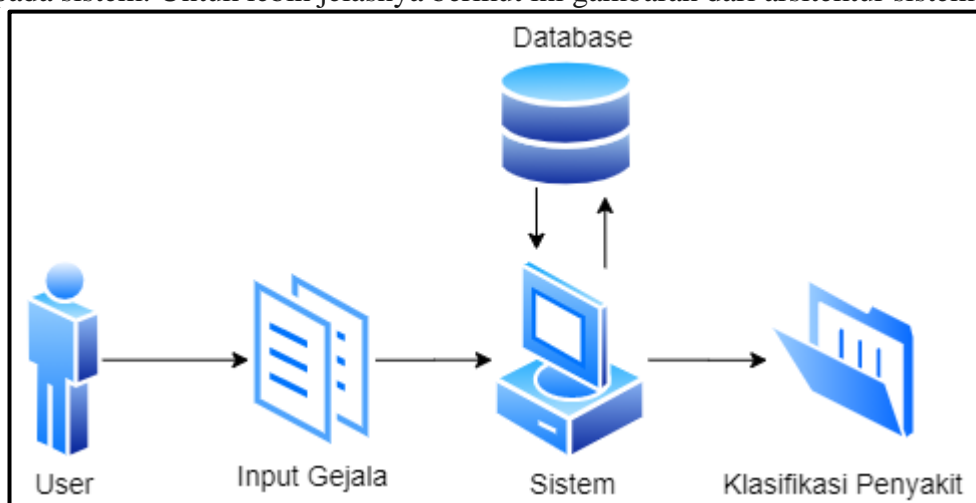
Dari tabel 3.x dapat diketahui bahwa data uji memiliki label data dengan kelas penyakit Infeksi Saluran Kencing.

3.4 Proses Desain

Proses desain adalah tahapan yang membahas proses pembuatan rancangan untuk disajikan kepada pengguna berdasarkan hasil komunikasi antara pengembang dan pengguna sebelumnya. Perancangan yang dibuat bertujuan untuk memberikan gambaran dan mempermudah proses pembuatan sistem yang nantinya akan dibangun. Pada proses desain ini akan dibuat rancangan sistem dan rancangan pengujian.

3.4.1 Perancangan Sistem

Pada tahap ini merupakan langkah awal dalam melakukan pembangunan suatu sistem. Tahapan ini dilakukan dengan membuat arsitektur sistem secara keseluruhan dengan berdasarkan pada hasil tahap komunikasi yang dilakukan sebelumnya agar pengembangan sesuai dengan yang diharapkan. Melalui tahapan ini pengguna dapat melihat gambaran yang terdapat pada sistem. Untuk lebih jelasnya berikut ini gambaran dari arsitektur sistem:



Gambar 3.6 Arsitektur Sistem

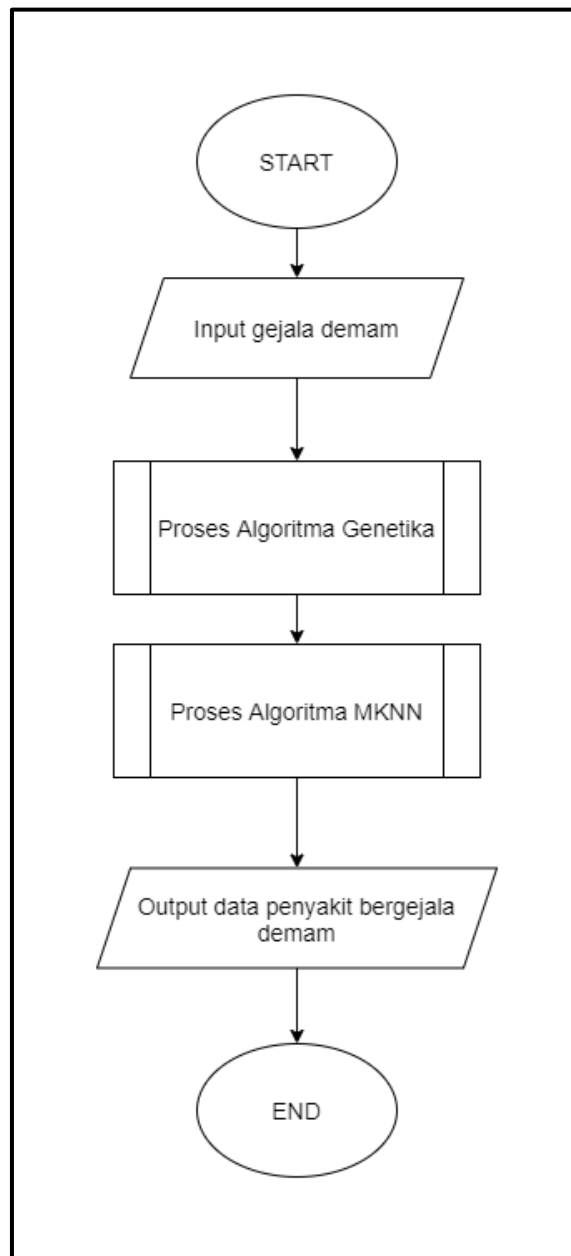
Pada arsitektur sistem gambar 3.6 di atas dimulai dengan adanya *user* memasukkan gejala demam ke dalam sistem. Kemudian dalam sistem tersebut dilakukan proses pencarian nilai k secara otomatis dengan menggunakan algoritma genetika. Selanjutnya, melakukan proses klasifikasi data tersebut menggunakan algoritma $mknn$ dengan data yang ada dalam database kemudian mengklasifikasikan penyakit berdasarkan gejala yang di inputkan oleh *user*.

3.4.2 Perancangan Proses

Perencanaan proses pada penelitian ini menggunakan *Entity Relationship Diagram*, *Data Flow Diagram* dan *flowchart* untuk dapat menggambarkan alur dari proses masukan menjadi hasil keluaran.

1. *Flowchart* Keseluruhan Sistem

Berikut ini merupakan *flowchart* secara keseluruhan sistem hingga detail dari tiap-tiap proses.



Gambar 3.7 *Flowchart* Sistem

Pada Gambar 3.7 menjelaskan proses secara umum yang ada pada aplikasi klafisikasi penyakit bergejala demam. Proses dimulai dari user memasukkan gejala yang diderita, selanjutnya yang akan dilakukan yaitu proses algoritma genetika yang secara otomatis mencari nilai k untuk digunakan pada proses selanjutnya. Pada Proses selanjutnya yaitu proses algoritma MKNN, pada proses ini mengklasifikasikan penyakit berdasarkan gejala

yang dimasukkan oleh user. Pada proses yang terakhir yaitu keluaran dari penyakit yang di diagnosa oleh sistem berdasarkan gejala yang dimasukkan oleh user.

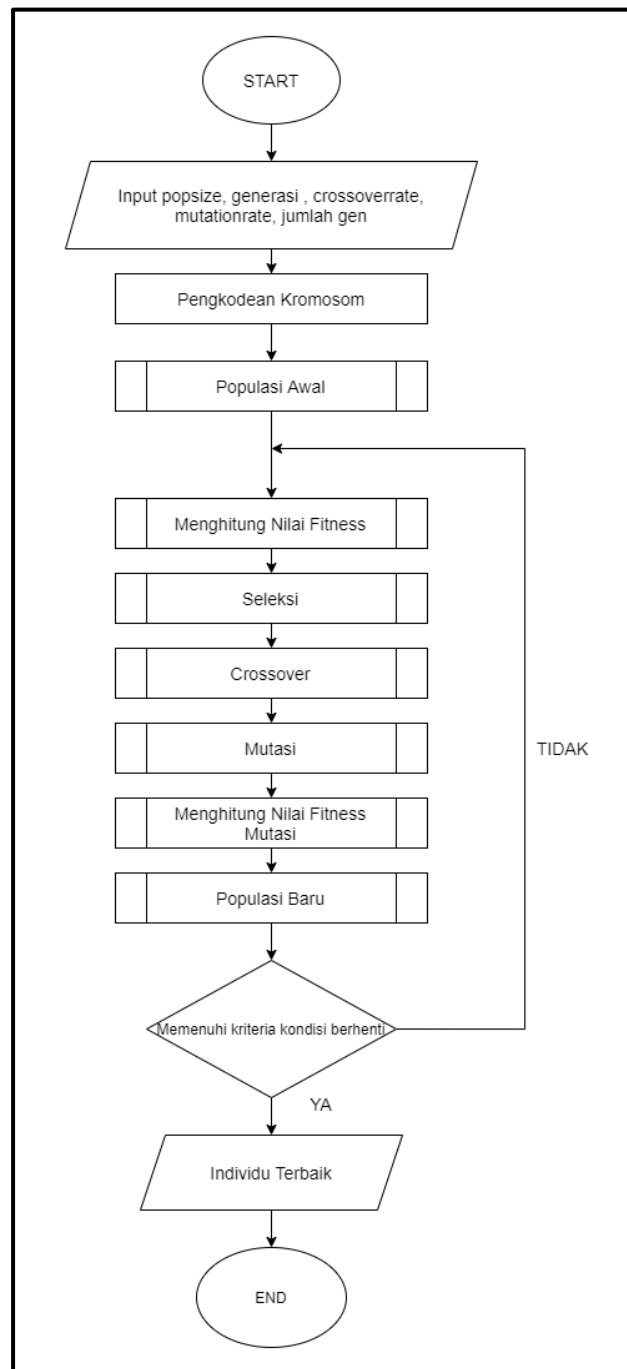
2. *Flowchart* Algoritma Genetika

Perancangan awal sistem akan menggunakan *flowchart* algoritma genetika, algoritma gentika berfungsi untuk mencari nilai k yang merupakan salah satu paremeter algoritma MKNN. Flowchart ini merupakan gambaran umum semua proses algoritma genetika dalam sistem yang dibuat pada penelitian ini.

Alur proses algoritma genetika bisa dilihat pada gambar 3.8.

Berikut merupakan tahapan proses algoritma genetika secara garis besar :

1. Tahap pertama admin menginputkan jumlah populasi , generasi , jumlah gen , mutation rate dan crossover rate.
2. Menginisialisasikan banyaknya populasi.
3. Menghitung nilai fitness individu pada populasi yang telah ditentukan.
4. Melakukan tahap selanjutnya yaitu seleksi pemilihan orangtua berdasarkan 2 nilai individu terbaik pada populasi.
5. Kawin silang antar orangtua dengan nilai fitness terbaik.
6. Cek mutasi per-gen pada anak hasil crossover.
7. Menghitung nilai fitness pada hasil mutase.
8. Mengganti nilai terkecil pada populasi apabila nilai mutase lebih baik dari nilai pada populasi sebelumnya.
9. Mendapatkan individu terbaik.
10. Proses selesai.



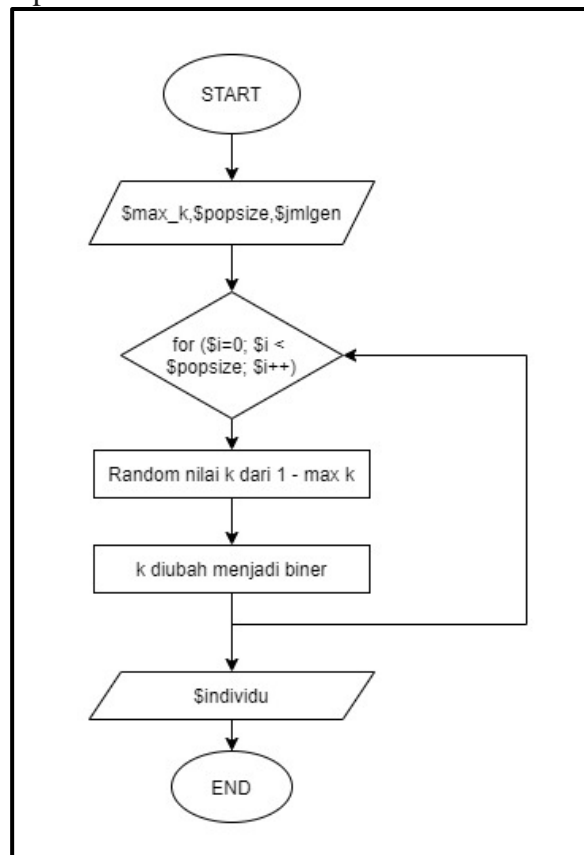
Gambar 3.8 Flowchart Algoritma Genetika

3. Flowchart Populasi Awal

Pada proses populasi awal membangkitkan sejumlah individu secara acak. Inti dari flowchart ini adalah melibatkan pembangkitan bilangan k random yang digunakan. Pada sistem ini menggunakan representasi biner merubah nilai k menjadi bilangan biner. Berikut adalah rumus untuk pembangkitan populasi pada sistem ini :

$$IPOP = \{random(N_{ipop}, N_{bits})\}$$

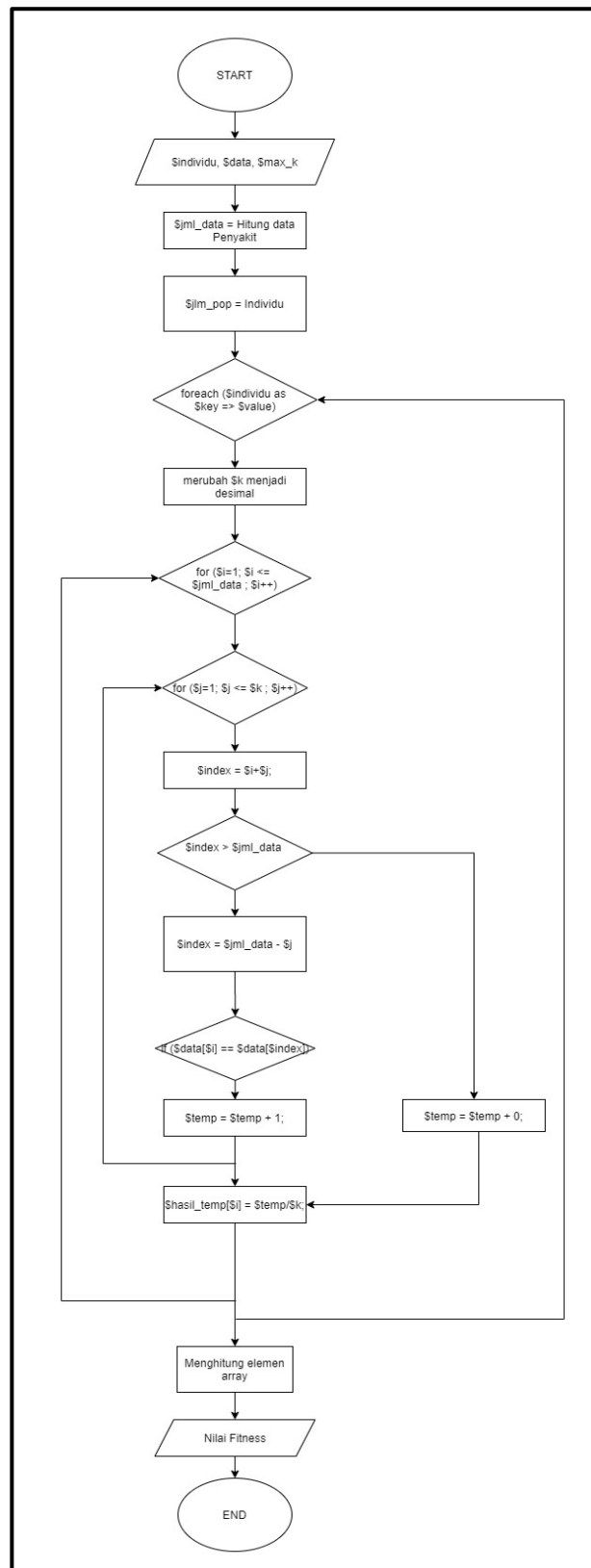
Dimana IPOP adalah individu yang berisi bilangan yang dibangkitkan sebanyak N_{ipop} (Jumlah populasi) dan N_{bits} (Jumlah Gen dalam setiap individu). Alur pembangkitan populasi awal bisa dilihat pada flowchart berikut :



Gambar 3.9 Flowchart Populasi Awal

4. Flowchart Menghitung Nilai *Fitness*

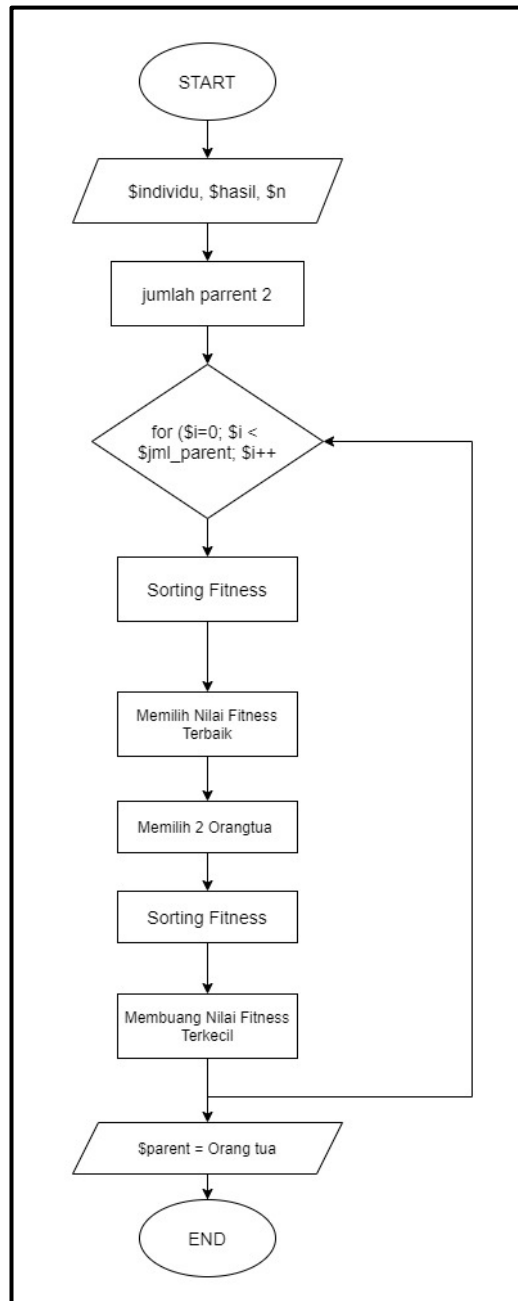
Proses perhitungan validitas bertujuan untuk melakukan validasi setiap data latih yang digunakan dalam sistem yang ditunjukkan pada gambar 3.10. Tahap awal adalah melakukan perhitungan jarak individu antar data latih lalu mengambil sejumlah K tetangga terdekat berdasarkan hasil jarak individu yang terkecil. Tahap selanjutnya adalah melakukan perbandingan kelas untuk data latih tersebut. Hasil perbandingan tersebut kemudian dijumlahkan dan dibagi dengan nilai K yang telah ditentukan. Apabila data sudah sesuai, maka selanjutnya sistem akan melakukan perhitungan elemen *array* dan kemudian akan didapatkan *output* dari nilai *fitness*. Jika belum sesuai, maka sistem akan melakukan perulangan untuk melihat apakah nilai sudah memenuhi syarat. Alur perhitungan nilai *fitness* dapat dilihat pada flowchart berikut :



Gambar 3.10 Flowchart Fitness

5. Flowchart Seleksi

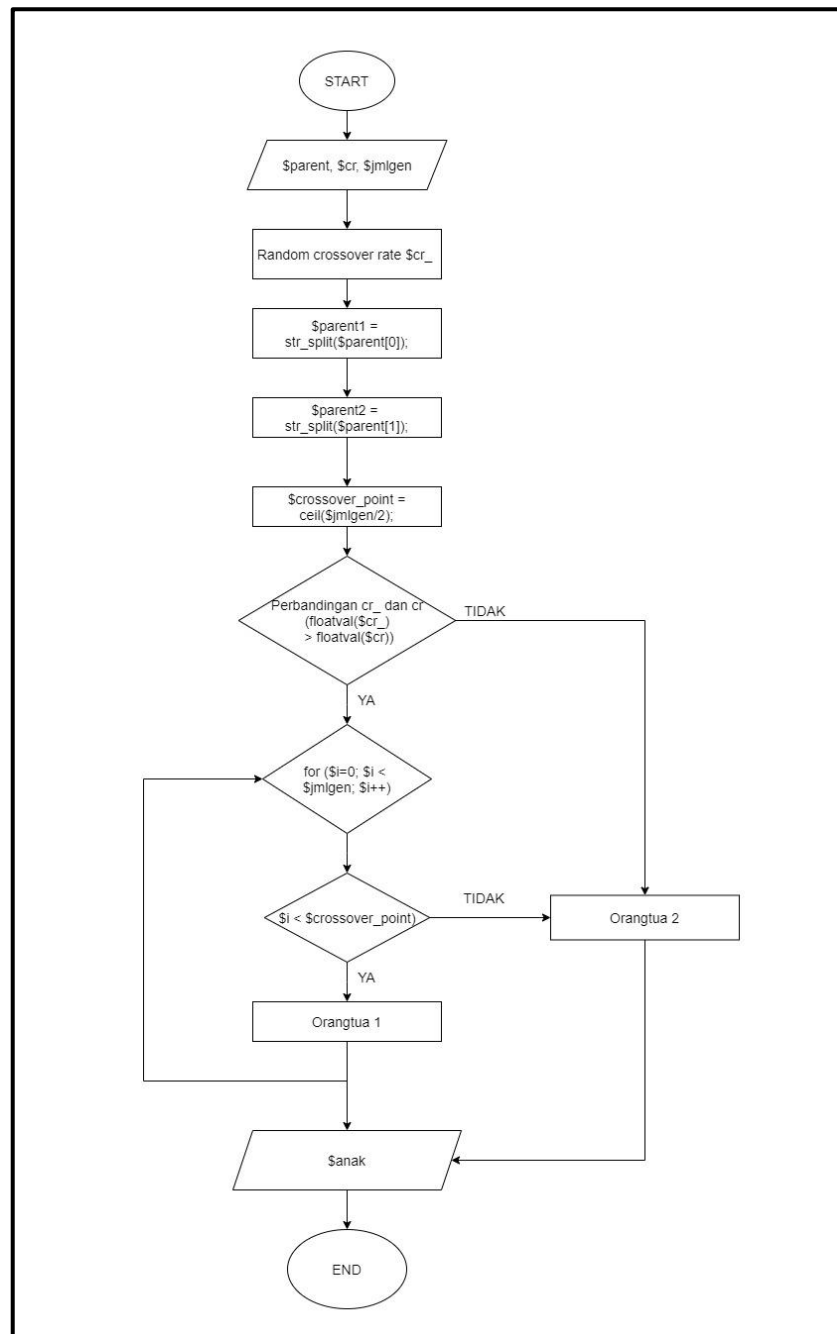
Pada proses ini digunakan untuk memilih individu mana saja yang akan dipilih untuk kawin silang dan mutasi. Seleksi digunakan untuk mendapatkan induk yang baik. Semakin tinggi nilai fitness suatu individu semakin besar kemungkinannya untuk terpilih. Pada algoritma ini menggunakan metode turnamen dimana dipilih 2 individu dengan nilai fitness terbaik untuk dijadikan induk. Alur tahap seleksi bisa dilihat pada gambar berikut :



Gambar 3.11 Flowchart Seleksi

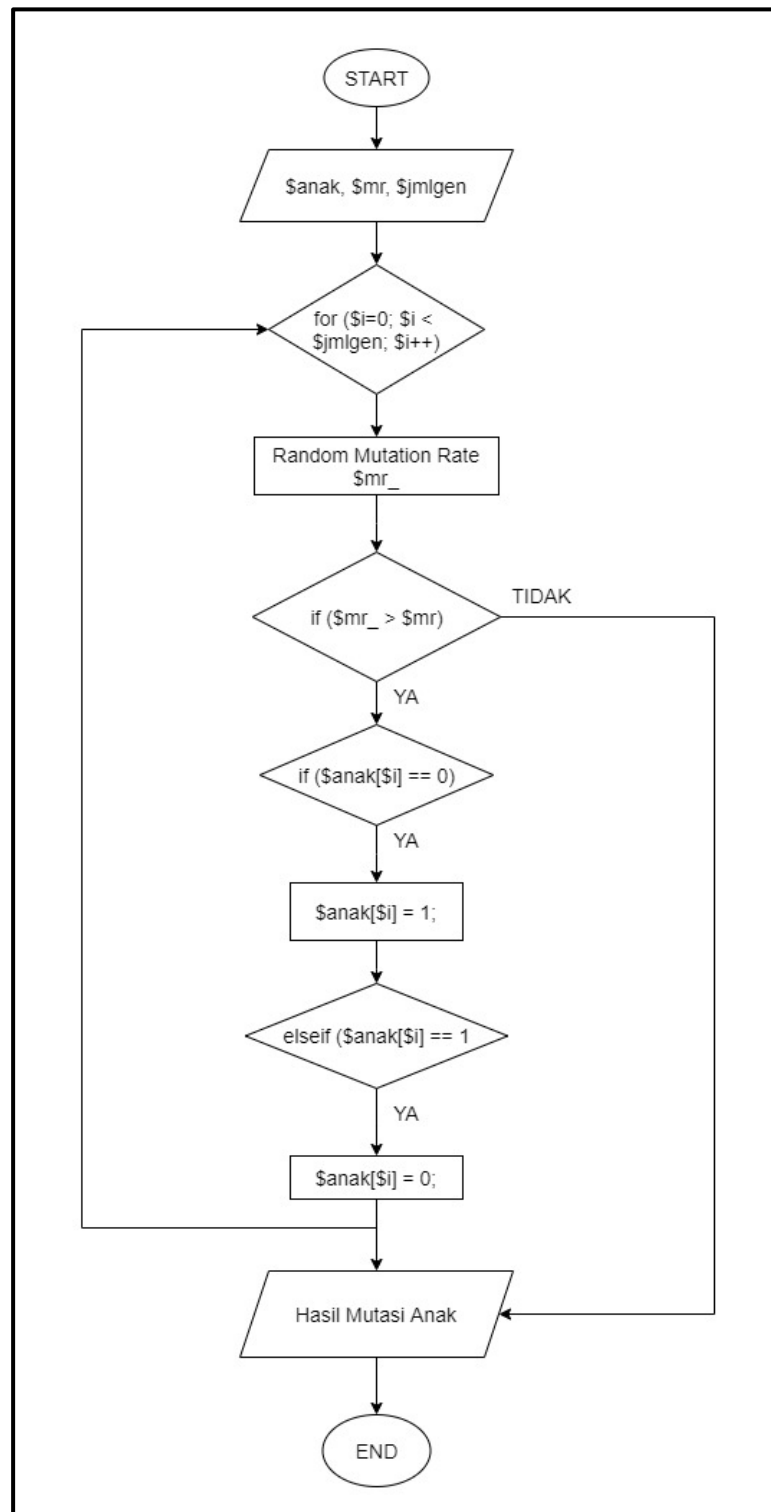
6. Flowchart Crossover

Setelah melakukan tahap seleksi selanjutnya melakukan kawin silang (crossover) dimana crossover sendiri melibatkan 2 induk untuk membentuk individu baru. Prinsip dari pindah silang adalah melakukan operasi pertukaran pada gen yang bersesuaian dari dua induk untuk menghasilkan individu baru. Proses crossover untuk mendapatkan anak yang akan digunakan pada tahap selanjutnya dapat dilihat pada gambar flowchart mutasi berikut:



Gambar 3.12 Flowchart Crossover

7. Flowchart Mutasi

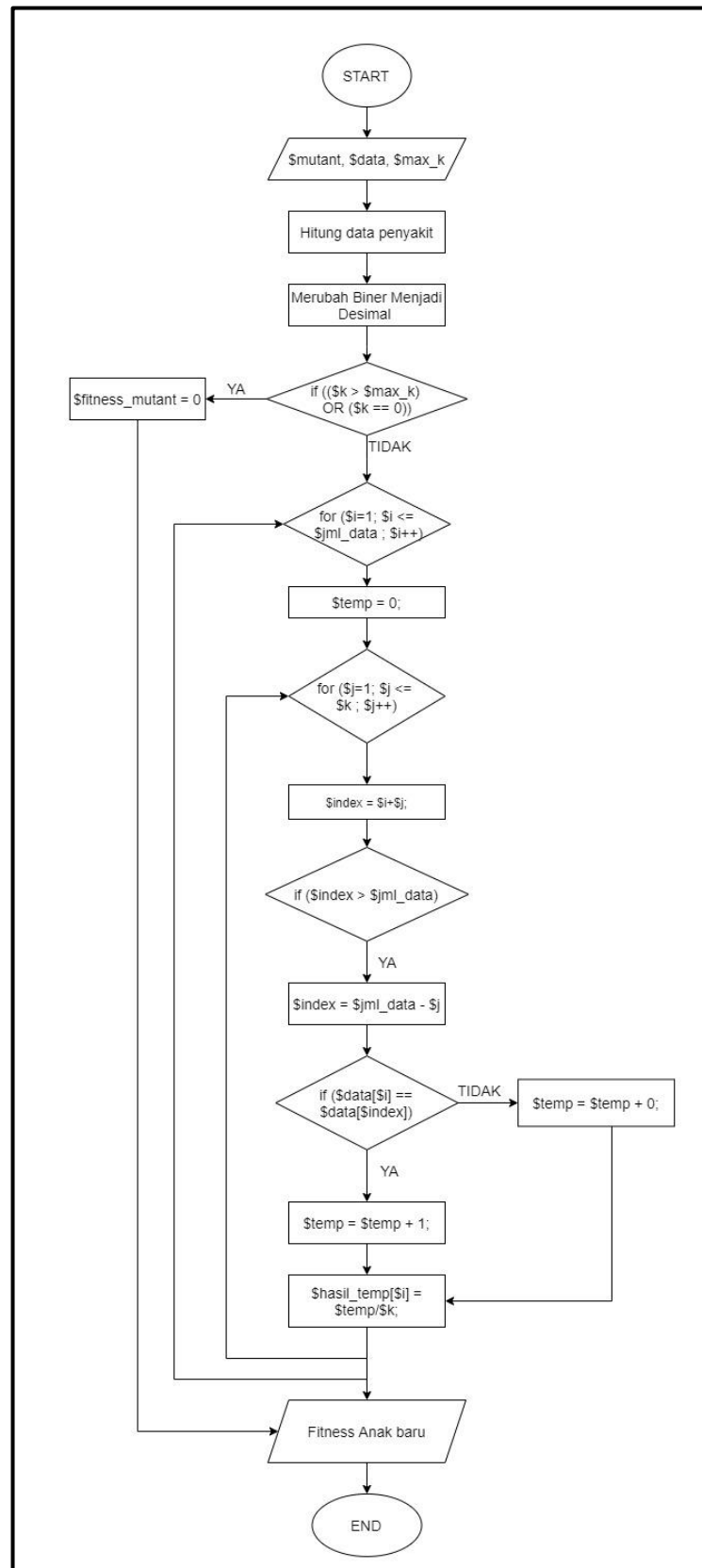


Gambar 3.13 Flowchart Mutasi

Pada proses ini dilakukannya proses mutasi yaitu penggantian gen dengan nilai inversinya. Gen 0 menjadi 1, dan gen 1 menjadi 0, Proses ini dilakukan secara acak pada posisi gen tertentu pada individu yang terpilih untuk dimutasikan. Banyaknya gen yang mengalami mutasi ditentukan oleh besarnya probabilitas mutasi

8. *Flowchart* Fitness Mutasi

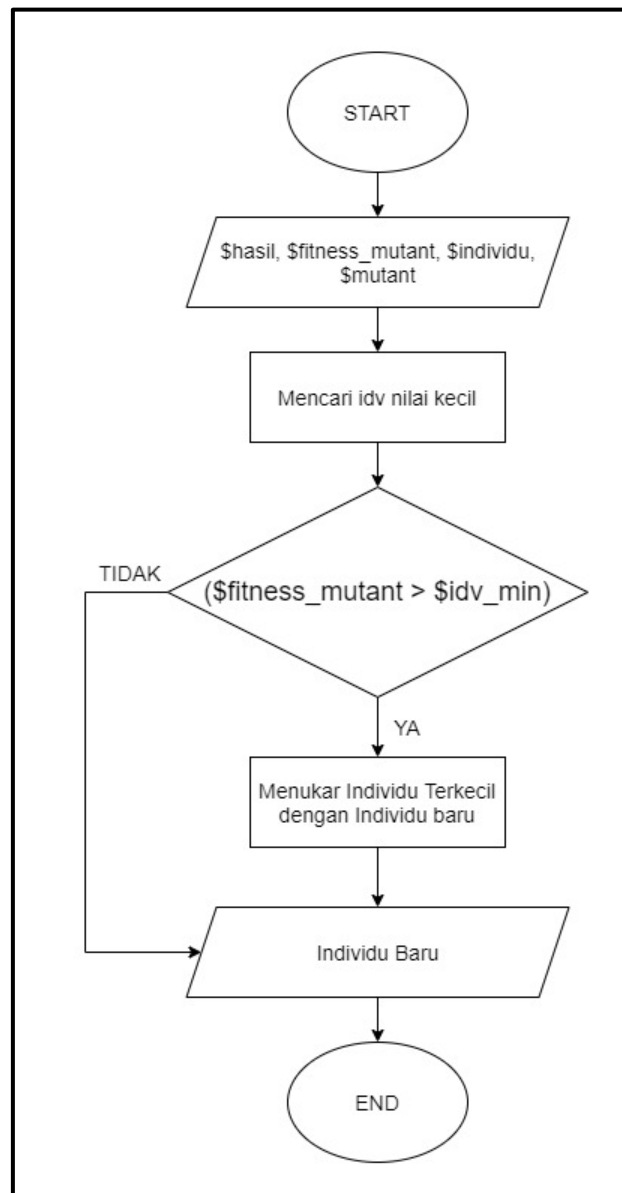
Proses perhitungan validitas bertujuan untuk melakukan validasi setiap data latih yang digunakan dalam sistem yang ditunjukkan pada gambar 3.14 Tahap awal adalah melakukan perhitungan jarak *mutant* antar data latih lalu mengambil sejumlah K tetangga terdekat berdasarkan hasil jarak *mutant* yang terkecil. Tahap selanjutnya adalah melakukan perbandingan kelas untuk data latih tersebut. Hasil perbandingan tersebut kemudian dijumlahkan dan dibagi dengan nilai K yang telah ditentukan. Alur perhitungan nilai *fitness* Mutasi dapat dilihat pada *flowchart* dibawah ini:



Gambar 3.14 Flowchart Fitness Mutasi

9. Flowchart Populasi Baru

Proses algoritma genetika akan dilakukan sampai mencapai suatu kriteria berhenti tertentu. Kriteria penghentian algoritma genetika yang digunakan adalah generasi. Algoritma genetika akan berhenti ketika mencapai generasi yang telah ditentukan, yang kemudian memperoleh individu terbaik dengan nilai fitness tertinggi dan menghilangkan individu dengan nilai fitness terkecil. Alur proses populasi baru seperti pada gambar dibawah ini:

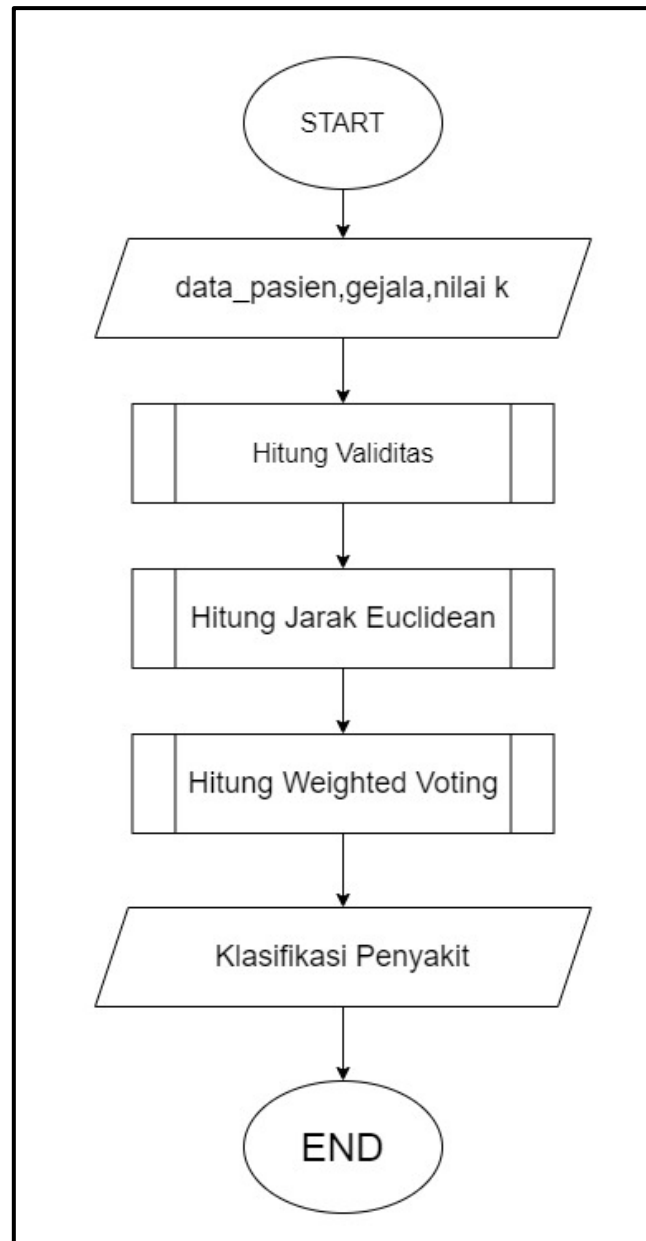


Gambar 3.15 Flowchart Populasi Baru

10. Flowchart MKNN

Klasifikasi demam menggunakan metode MKNN melalui beberapa tahap yaitu, memasukkan data gejala demam, menghitung validitas, menghitung jarak euclidean dan menghitung weighted voting. Perhitungan jarak Euclidean antara data latih dan data uji dilakukan untuk menghasilkan nilai yang selanjutnya di proses dalam perhitungan weighted voting. Hasil perhitungan weighted voting diurutkan dari nilai terbesar ke nilai terkecil dan

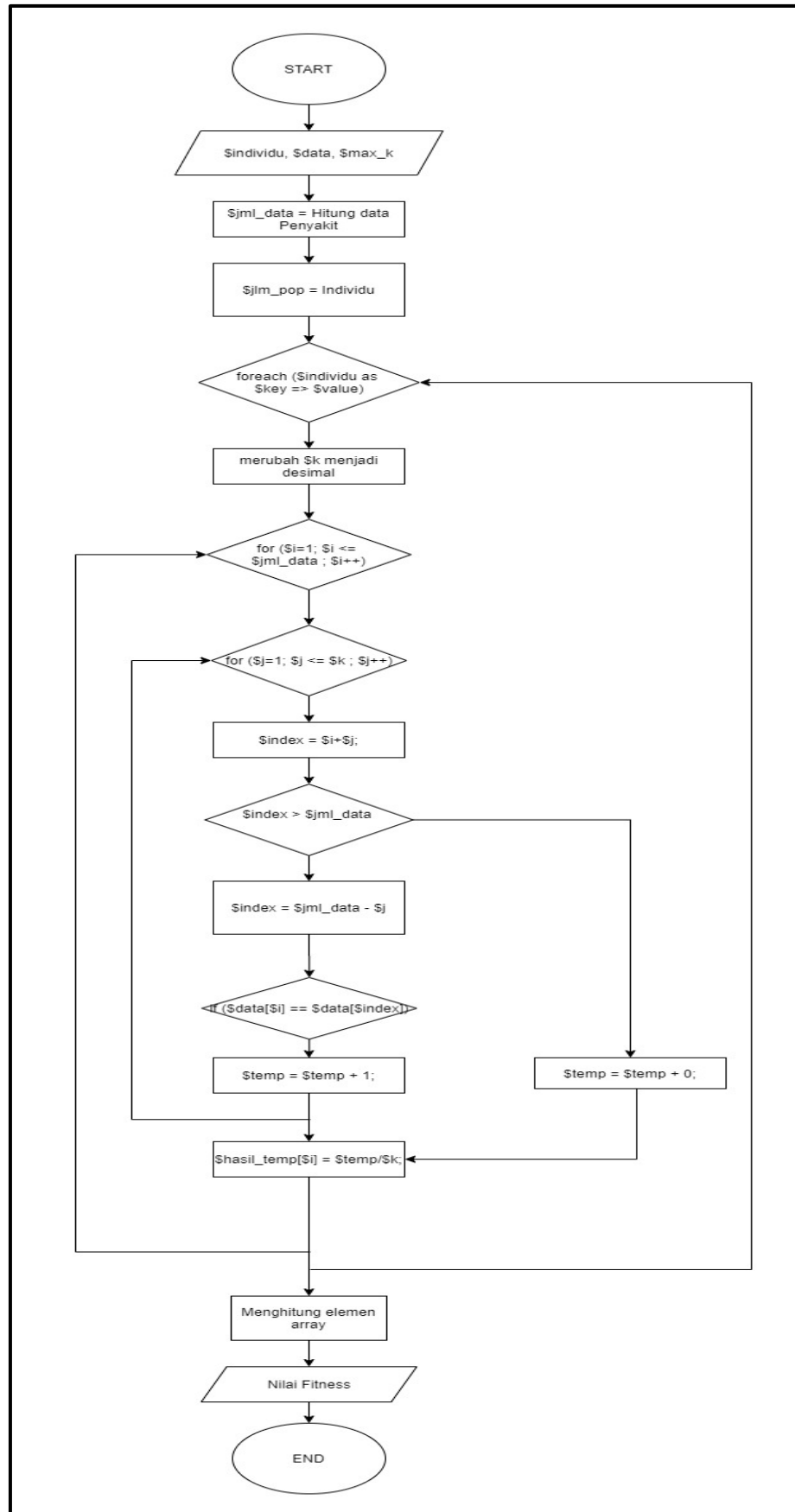
hasil tersebut menjadi hasil klasifikasi yang dilakukan oleh sistem. Proses klasifikasi MKNN data ditunjukkan pada gambar dibawah ini :



Gambar 3.16 Flowchart Klasifikasi MKNN

11. Flowchart Validitas

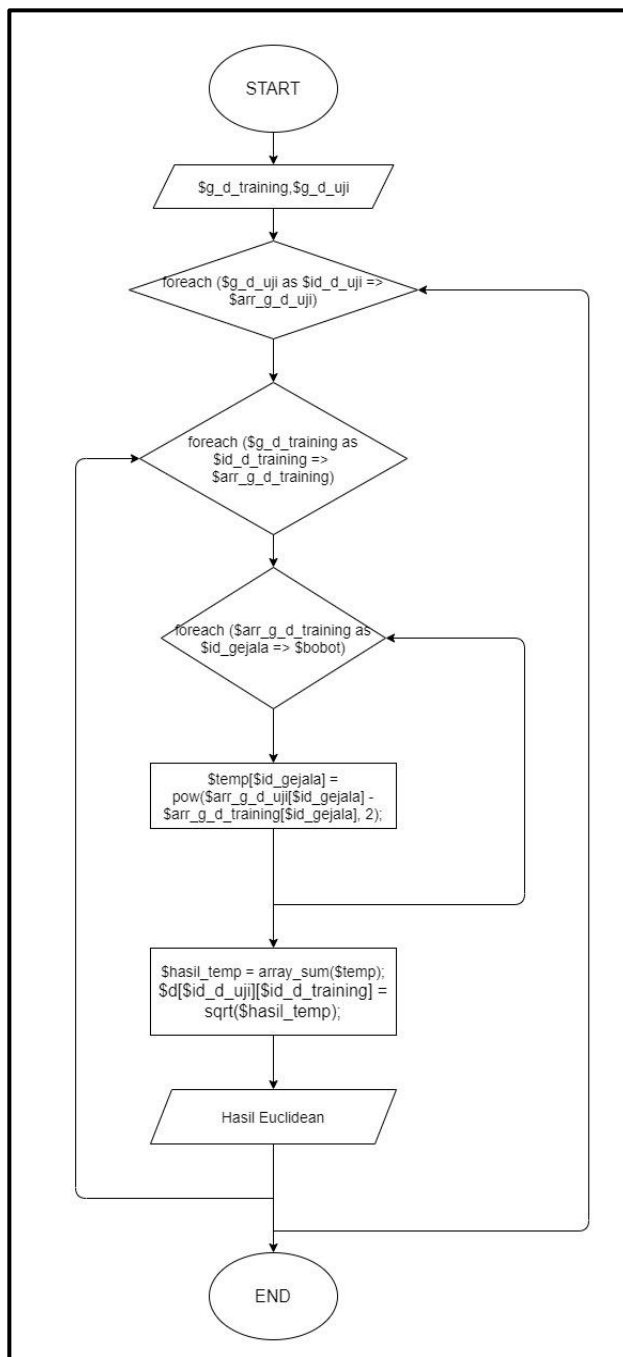
Proses perhitungan validitas bertujuan untuk melakukan validasi setiap data latih yang digunakan dalam sistem yang ditunjukkan pada gambar 3.17 Tahap awal adalah melakukan perhitungan jarak individu antar data latih lalu mengambil sejumlah K tetangga terdekat berdasarkan hasil jarak individu yang terkecil. Tahap selanjutnya adalah melakukan perbandingan kelas untuk data latih tersebut. Hasil perbandingan tersebut kemudian dijumlahkan dan dibagi dengan nilai K yang telah ditentukan. Alur perhitungan nilai *fitness* Mutasi dapat dilihat pada *flowchart* dibawah ini:



Gambar 3.17 Flowchart Validitas

12. Flowchart Euclidean

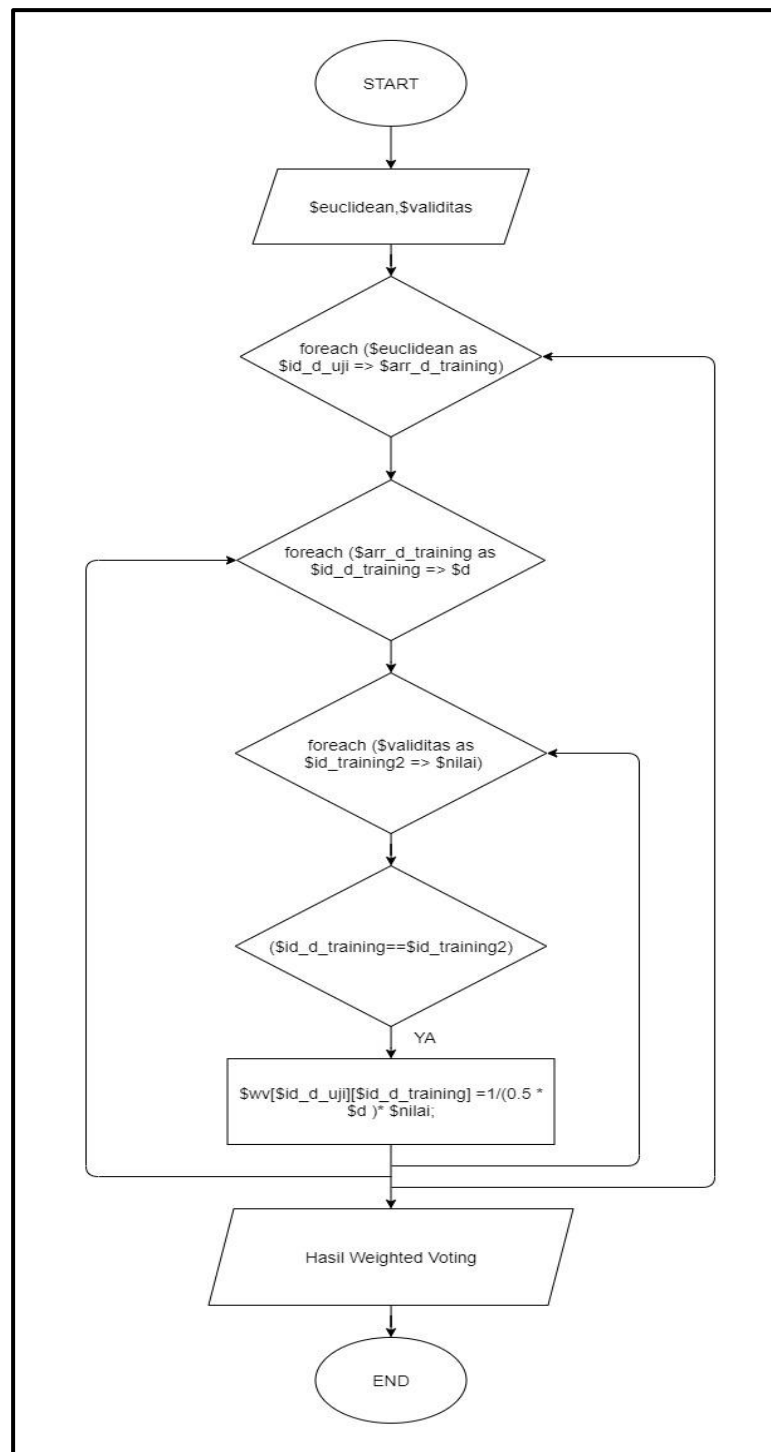
Perhitungan jarak Euclidean dilakukan menghitung nilai jarak Euclidean antar data latih dan data uji. Proses untuk perhitungan jarak Euclidean dijelaskan dalam gambar 3.x dimulai dari memasukkan dataset lalu sistem melakukan perulangan sebanyak data latih yang dimasukkan oleh pengguna sampai didapatkan nilai jarak Euclidean.



Gambar 3.18 Flowchart Perhitungan Euclidean

13. Flowchart Weighted Voting

Proses menghitung hasil weighted voting merupakan tahapan terakhir untuk mendapatkan hasil klasifikasi. Hasil dari weighted voting didapatkan dari nilai validitas dan nilai jarak Euclidean data latih dan data uji. Proses perhitungan weighted dijelaskan dalam gambar 3.x dimulai dari memasukkan nilai jarak antara data latih dan data uji serta nilai validitas untuk selanjutnya dilakukan perulangan sampai mendapatkan nilai dari perhitungan weighted voting.



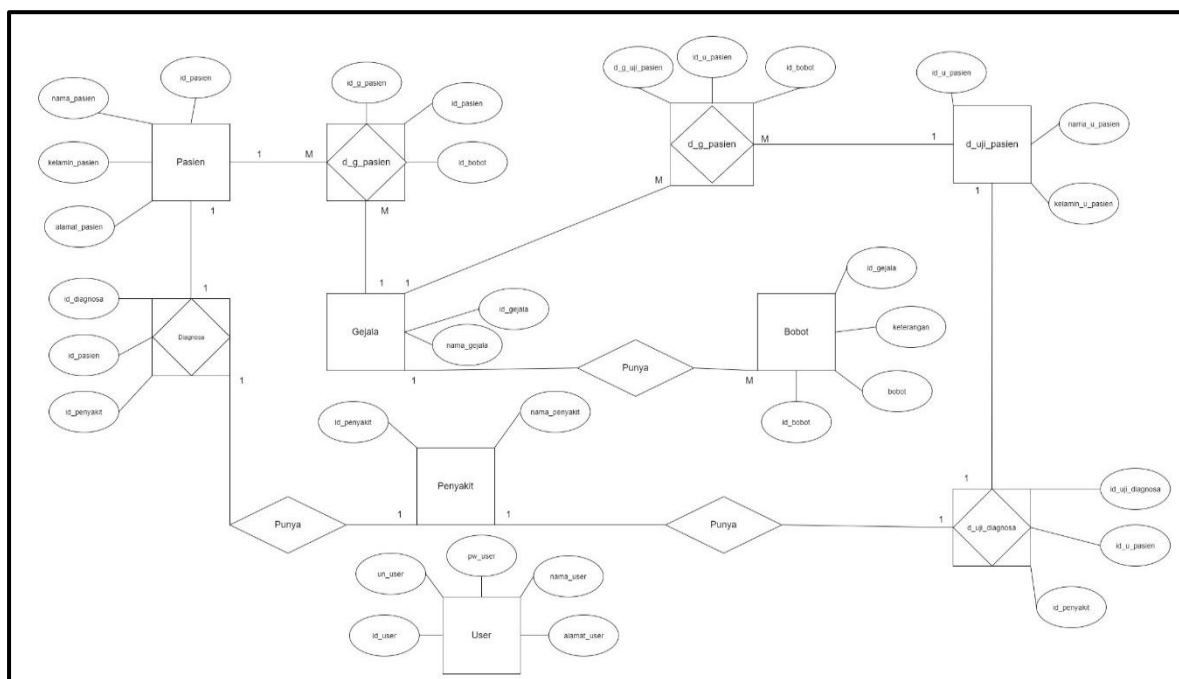
Gambar 3.19 Flowchart Weighted Voting

3.4.3 Perancangan Basis Data

Dalam Penelitian ini membutuhkan *database* sebagai tempat penyimpanan seluruh data yang dibutuhkan dan akan mempermudah pengembang dalam memvisualisasikan data-data yang perlu untuk ditampilkan di dalam sistem. Pada *database* yang bernama “db_gmknn” ini didalamnya terdapat sepuluh tabel, yaitu d_bobot, d_diagnosa, d_gejala, d_g_pasien, d_g_uji_pasien, d_pasien, d_penyakit, d_uji_diagnosa, d_uji_pasien, u_user. Basis data pada sistem ini menggunakan *database MySQL* dan *Server HTTP Apache*.

3.4.3.1. Entity Relationship Diagram (ERD)

Entity Relationship Diagram (ERD) adalah suatu model untuk menjelaskan hubungan antar data dalam basis data berdasarkan objek-objek dasar data mempunyai hubungan antar relasi. Proses ini akan memodelkan struktur data dan hubungan antar data. Terdapat sepuluh buah entitas pada ERD d_pasien, d_diagnosa, d_gejala, d_bobot, d_penyakit, d_g_ujipasien, d_g_pasien, d_uji_diagnosa, d_user dan d_u_pasien. data_training. Rancangan ERD pada penelitian ini dapat dilihat pada gambar dibawah ini.

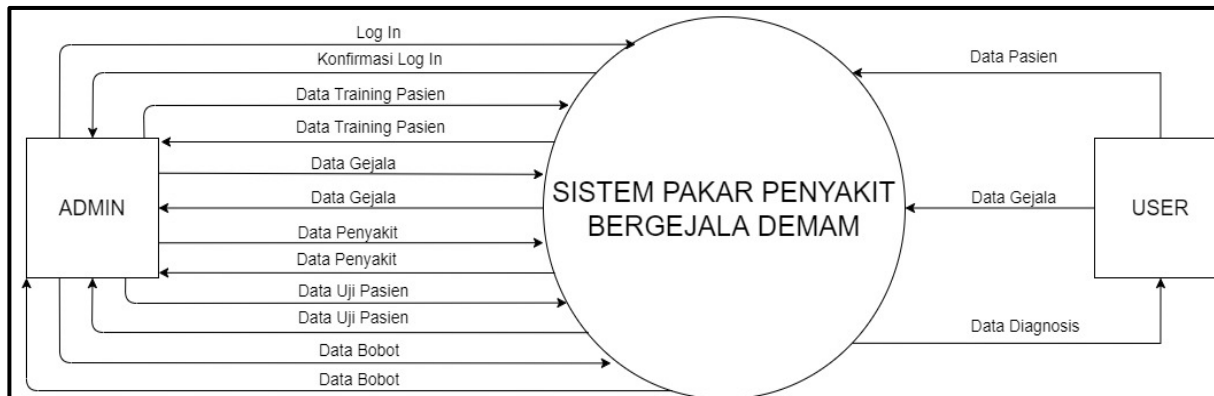


Gambar 3.20 Entity Relationship Diagram (ERD)

3.4.3.2. Data Flow Diagram (DFD)

1. Data Flow Diagram (DFD) Level 0

DFD level 0 berisi gambaran yang berupa garis besar dari apa yang akan dibangun dalam sistem. DFD Level 0 dapat dilihat pada gambar berikut :



Gambar 3.21 Data Flow Diagram Level 0

Pada gambar 3.x DFD level 0 terdapat 2 entitas yang menggunakan sistem yaitu admin dan user dimana admin mempunyai hak istimewa untuk melakukan manajemen user dan upload model classifier. Pada entitas admin terdapat enam aliran data keluar maupun masuk yaitu Login, Data Training Pasien, Data Gejala, Data Penyakit, Data Uji Pasien, dan Data Bobot. Sedangkan pada entitas user hanya dapat melakukan input berupa data pasien, data gejala dan mendapat output sebagai diagnosa penyakit.

2. Data Flow Diagram (DFD) Level 1

Pada DFD level 1 berisi proses turunan dari level sebelumnya, DFD Level 0 dapat diturunkan menjadi 3 proses.

a. DFD Level 1 Proses Login

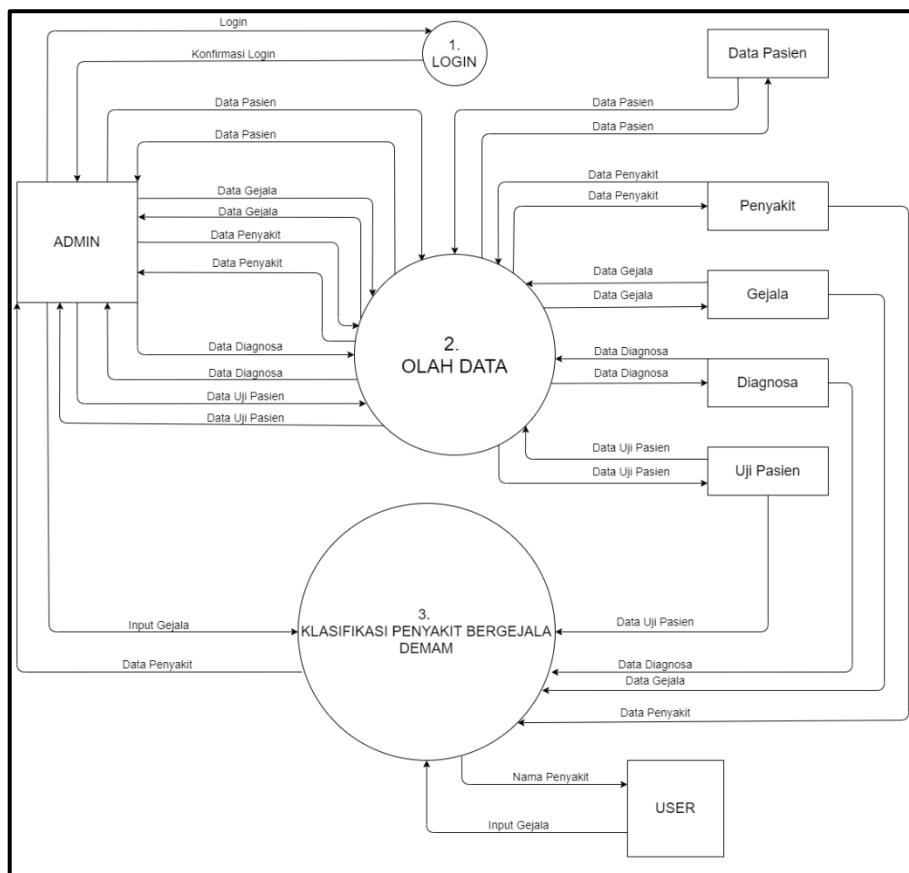
Proses yang pertama adalah proses login, entitas admin memerlukan login untuk dapat mengakses fitur dalam sistem.

b. DFD Level 1 Proses Olah Data

Proses olahdata merupakan proses dalam sistem yang dapat diakses oleh admin. Data yang terkait pada proses ini adalah Data Pasien, Data Penyakit, Data Diagnosa, Data Gejala, Data Uji Pasien, dan Data Uji Diagnosa. Pada proses ini admin dapat menambahkan dan melihat data yang ada pada proses olahdata.

c. DFD Level 1 Proses Sistem Pakar

Proses sistem pakar merupakan proses utama dalam sistem ini, proses ini bisa diakses admin dan user. Pada proses ini admin dan user dapat melakukan input berupa gejala dan mendapat output berupa penyakit. Pada proses ini sistem mengklasifikasikan menggunakan algoritma genetika dan MKNN.



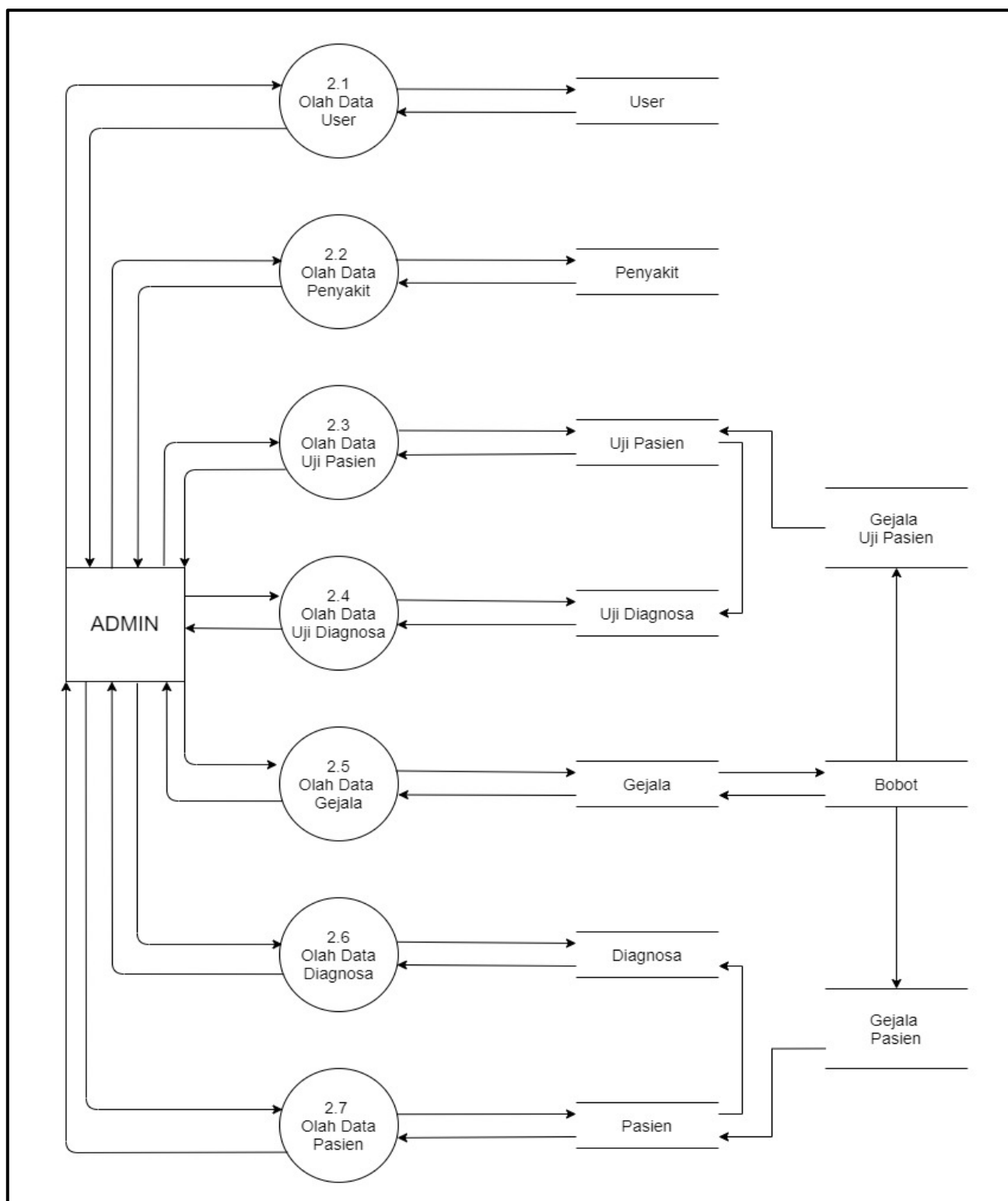
Gambar 3.22 Data Flow Diagram Level 1

3. DFD 2 Olahdata

DFD Level 2 menjelaskan proses secara detail pada DFD Level 1 yang masih bisa dijelaskan lagi lebih detail, terdapat satu buah proses yaitu Proses Olahdata.

Pada proses olahdata, terdiri dari tujuh sub proses yaitu olahdata user, olahdata penyakit, olahdata data uji pasien, olahdata uji diagnosa, olahdata gejala, olahdata diagnosa, olahdata pasien. Pada setiap proses terdapat data masuk dan keluar berupa tambah, hapus dan edit.

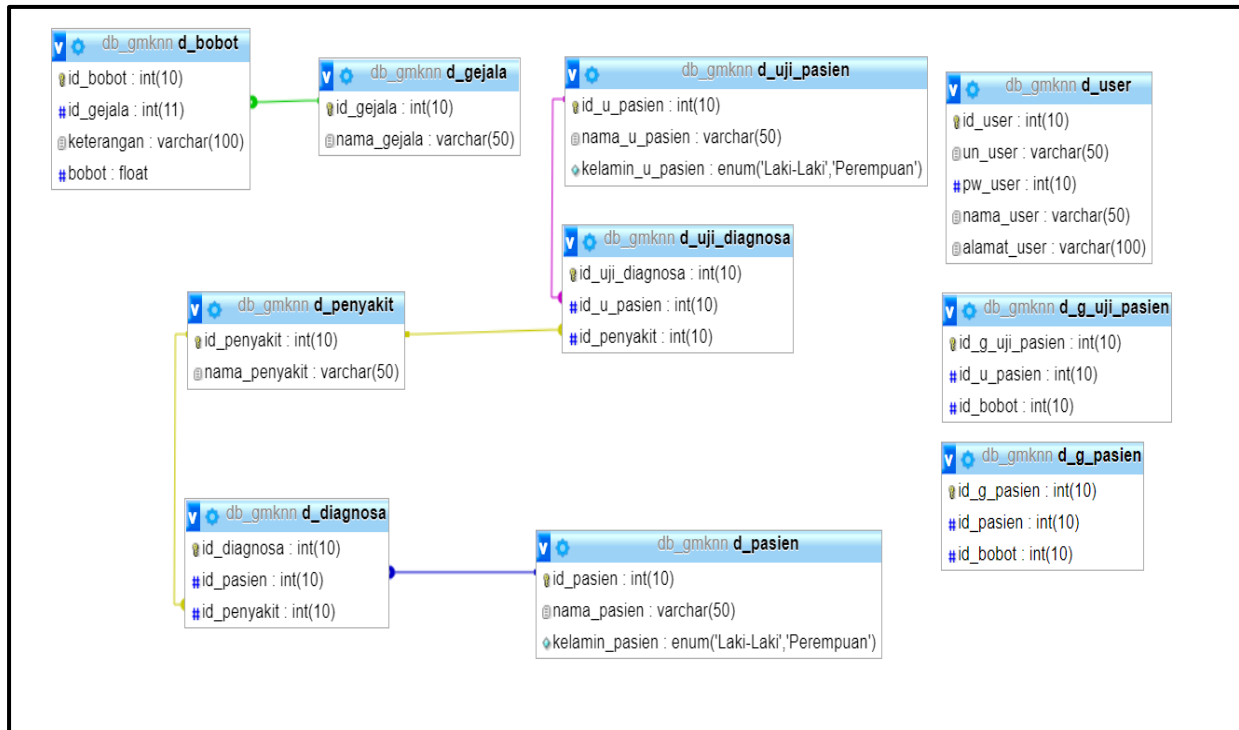
Pada proses gejala terdapat tabel bobot dimana tabel bobot berisi bobot dari setiap gejala yang nantinya digunakan pada algoritma MKNN. Nilai bobot nantinya digunakan pada uji pasien, uji diagnose, data pasien dan data diagnosa.



Gambar 3.23 Data Flow Diagram Level 2 Proses Olahdata

3.4.3.3. Relasi Antar Tabel

Perancangan struktur antar tabel dalam klasifikasi penyakit berdasarkan gejala demam menggunakan algoritma genetika dan MKNN dapat dilihat pada gambar berikut :



Gambar 3.24 Relasi Antar Tabel

3.4.4 Struktur Tabel

Basis data pada penelitian ini terdiri dari beberapa tabel yang masing-masing tabel memiliki struktur tabel dapat dilihat sebagai berikut:

1. Tabel Data Admin

Pada tabel data admin mempunyai 5 buah atribut yaitu id_user, username, nama, alamat, password dimana id_user sebagai *primary key*. Tabel data admin merupakan tabel yang menyimpan semua data yang berhubungan dengan admin sebagai hak akses untuk masuk ke sistem. Struktur tabel data user dapat dilihat pada Tabel 3.11.

Tabel 3.11 Struktur Tabel Data Admin

No	Nama Field	Tipe Data	Constraint	Keterangan
1	id_user	int(10)	Primary key, Auto Increment	Id user
2	un_user	varchar(50)	None	Username user
3	nama_user	varchar(50)	None	Nama user
4	pw_user	int(10)	None	Password user
5	Alamat_user	varchar(100)	None	Alamat user

2. Tabel Data Penyakit

Pada tabel data penyakit mempunyai 2 buah atribut yaitu id_penyakit dan nama penyakit dimana id_penyakit sebagai *primary key*. Tabel data penyakit merupakan tabel yang menyimpan semua data yang berhubungan dengan data penyakit. Pada tabel ini mempunyai relasi antara tabel diagnosa dan uji diagnosa. Struktur tabel data penyakit dapat dilihat pada Tabel 3.12.

Tabel 3.12 Struktur Tabel Data Penyakit

No	Nama Field	Tipe Data	Constraint	Keterangan
1	Id_penyakit	int(10)	Primary key, Auto Increment	Id Penyakit
2	Nama_penyakit	varchar(50)	None	Nama Penyakit

3. Tabel Data *Training* Pasien

Pada tabel data kata dasar mempunyai 3 buah atribut yaitu id_pasien, nama_pasien dan kelamin_pasien dimana id_pasien sebagai *primary key*. Tabel data *training* pasien merupakan tabel yang menyimpan semua data yang berhubungan dengan data pasien. Pada tabel ini terdapat relasi antar id_pasien pada tabel data *training* pasien dan data diagnosa. Struktur tabel data katadasar dapat dilihat pada Tabel 3.13.

Tabel 3.13 Struktur Tabel Data *Training* Pasien

No	Nama Field	Tipe Data	Constraint	Keterangan
1	id_pasien	int(10)	Primary key, Auto Increment	Id Pasien
2	nama_pasien	varchar(50)	None	Nama Pasien
3	kelamin_pasien	Enum('Laki','Perempuan')	None	Kelamin Pasien

4. Tabel Data Gejala

Pada tabel data gejala mempunyai 2 buah atribut yaitu id_gejala dan nama_gejala dimana id_gejala sebagai *primary key*. Tabel data gejala merupakan tabel yang menyimpan semua data yang berhubungan dengan data gejala. Pada tabel ini juga terdapat relasi antar tabel yaitu antara id_gejala pada data gejala dan id_gejala pada data bobot. Struktur tabel data *slangword* dapat dilihat pada Tabel 3.14.

Tabel 3.14 Struktur Tabel Data Gejala

No	Nama Field	Tipe Data	Constraint	Keterangan
1	id_gejala	int(10)	Primary key, Auto Increment	Id gejala
2	nama_gejala	varchar(50)	None	Data Gejala

5. Tabel Data Diagnosa

Pada tabel data kategori mempunyai 3 buah atribut yaitu id_diagnosa, id_pasien dan id_penyakit dimana id_diagnosa sebagai *primary key*. Tabel data kategori merupakan tabel yang menyimpan semua data yang berhubungan dengan data pasien beserta penyakitnya, dimana relasi antar tabel yang terjadi pada id_pasien dan id_penyakit. Struktur tabel data kategori dapat dilihat pada Tabel 3.15.

Tabel 3.15 Struktur Tabel Data Diagnosa

No	Nama Field	Tipe Data	Constraint	Keterangan
1	id_diagnosa	int(10)	Primary key, Auto Increment	Id diagnose
2	id_pasien	int(10)	None	Id Pasien
3	id_penyakit	int(10)	None	Id Penyakit

6. Tabel Data Bobot

Pada tabel data latih mempunyai 4 buah atribut yaitu id_bobot, id_gejala, keterangan, dan bobot dimana id_bobot sebagai *primary key*. Tabel data bobot merupakan tabel yang menyimpan semua data yang berhubungan dengan data bobot. Pada tabel ini terdapat

relasi antar tabel yang terdapat pada id_gejala yang juga dimiliki oleh data gejala. Struktur tabel data latih dapat dilihat pada Tabel 3.16.

Tabel 3.16 Struktur Tabel Data Bobot

No	Nama Field	Tipe Data	Constraint	Keterangan
1	id_bobot	int(10)	Primary key, Auto Increment	Id Bobot
2	Id_gejala	int(11)	None	Id Gejala
3	Keterangan	varchar(100)	None	Keterangan Bobot
4	bobot	float	None	Nilai Bobot

7. Tabel Data Uji Pasien

Pada tabel data kata dasar mempunyai 3 buah atribut yaitu id_u_pasien, nama_u_pasien dan kelamin_u_pasien dimana id_u_pasien sebagai *primary key*. Tabel data uji pasien merupakan tabel yang menyimpan semua data yang berhubungan dengan data uji pasien. Pada tabel ini terdapat relasi antar id_u_pasien pada tabel data uji pasien dan data uji diagnosa. Struktur tabel data katadasar dapat dilihat pada Tabel 3.17.

Tabel 3.17 Struktur Tabel Data Uji Pasien

No	Nama Field	Tipe Data	Constraint	Keterangan
1	id_u_pasien	int(10)	Primary key, Auto Increment	Id Uji Pasien
2	nama_u_pasien	varchar(50)	None	Nama Uji Pasien
3	kelamin_u_pasien	Enum('Laki', 'Perempuan')	None	Kelamin Uji Pasien

8. Tabel Data Uji Diagnosa

Pada tabel data kategori mempunyai 3 buah atribut yaitu id_u_diagnosa, id_u_pasien dan id_penyakit dimana id_u_diagnosa sebagai *primary key*. Tabel data kategori merupakan tabel yang menyimpan semua data yang berhubungan dengan data uji pasien beserta penyakitnya, dimana relasi antar tabel yang terjadi pada id_u_pasien dan id_penyakit. Struktur tabel data kategori dapat dilihat pada Tabel 3.18.

Tabel 3.18 Struktur Tabel Data Uji Diagnosa

No	Nama Field	Tipe Data	Constraint	Keterangan
1	id_u_diagnosa	int(10)	Primary key, Auto Increment	Id uji diagnosa
2	id_u_pasien	int(10)	None	Id uji Pasien
3	id_penyakit	int(10)	None	Id Penyakit

9. Tabel Gejala Pasien

Pada tabel data kategori mempunyai 3 buah atribut yaitu id_g_pasien, id_pasien dan id_bobot dimana id_g_pasien sebagai *primary key*. Tabel data kategori merupakan tabel yang menyimpan semua data yang berhubungan dengan data pasien beserta bobot yang dimilikinya. Struktur tabel data kategori dapat dilihat pada Tabel 3.19.

Tabel 3.19 Struktur Tabel Data Gejala Pasien

No	Nama Field	Tipe Data	Constraint	Keterangan
1	id_g_pasien	int(10)	Primary key, Auto Increment	Id gejala pasien
2	id_pasien	int(10)	None	Id pasien
3	id_bobot	int(10)	None	Id bobot

10. Tabel Uji Gejala Pasien

Pada tabel data kategori mempunyai 3 buah atribut yaitu id_g_uji_pasien, id_u_pasien dan id_bobot dimana id_g_uji_pasien sebagai *primary key*. Tabel data kategori merupakan tabel yang menyimpan semua data yang berhubungan dengan data uji pasien beserta bobot yang dimilikinya. Struktur tabel data kategori dapat dilihat pada Tabel 3.20.

Tabel 3.20 Struktur Tabel Data Uji Gejala Pasien

No	Nama Field	Tipe Data	Constraint	Keterangan
1	id_g_uji_pasien	int(10)	Primary key, Auto Increment	Id uji diagnosa
2	id_u_pasien	int(10)	None	Id uji Pasien
3	id_penyakit	int(10)	None	Id Penyakit

3.4.5 Perancangan Antarmuka

Perancangan antarmuka atau *user interface* adalah model mekanisme komunikasi antara pengguna dengan sistem. Antarmuka atau *user interface* merupakan bentuk sistem dalam tampilan grafis yang dapat menerima informasi dan memberikan informasi pada pengguna dalam bentuk tampilan yang interaktif. Secara garis besar, user interface pada sistem ini akan dibagi menjadi dua aktor, yaitu pengguna dan admin.

Berikut merupakan perancangan *user interface* pada aktor pengguna yang terdiri dari beberapa bagian, diantaranya adalah sebagai berikut:

1. Rancangan Halaman Beranda

Halaman login merupakan halaman awal tempat dimana user dan admin untuk melakukan isi data diri dan melanjutkan ke tahap klasifikasi penyakit berdasarkan gejala yang diinputkan. Rancangan halaman beranda dapat dilihat pada gambar berikut :

Gambar 3.25 Rancangan halaman beranda

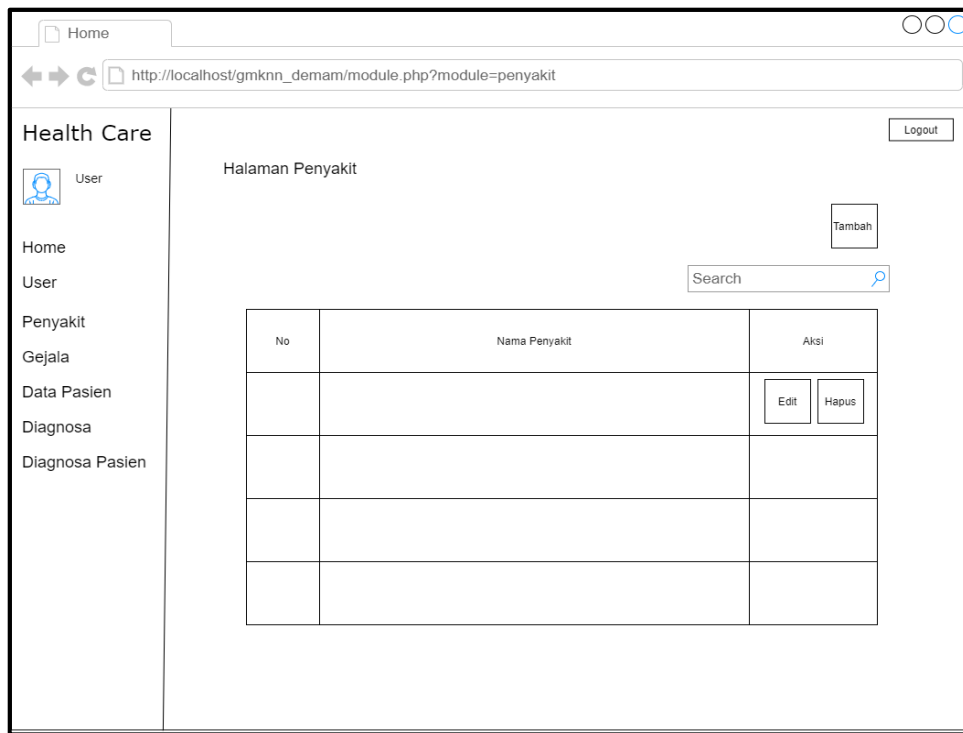
2. Rancangan Halaman Login

Halaman login merupakan halaman tempat admin untuk verifikasi akses ke sistem. Admin mengisi nama pengguna dan password untuk dapat melakukan proses login. Rancangan halaman login dapat dilihat pada gambar berikut :

Gambar 3.26 Rancangan halaman login

3. Rancangan Halaman Penyakit

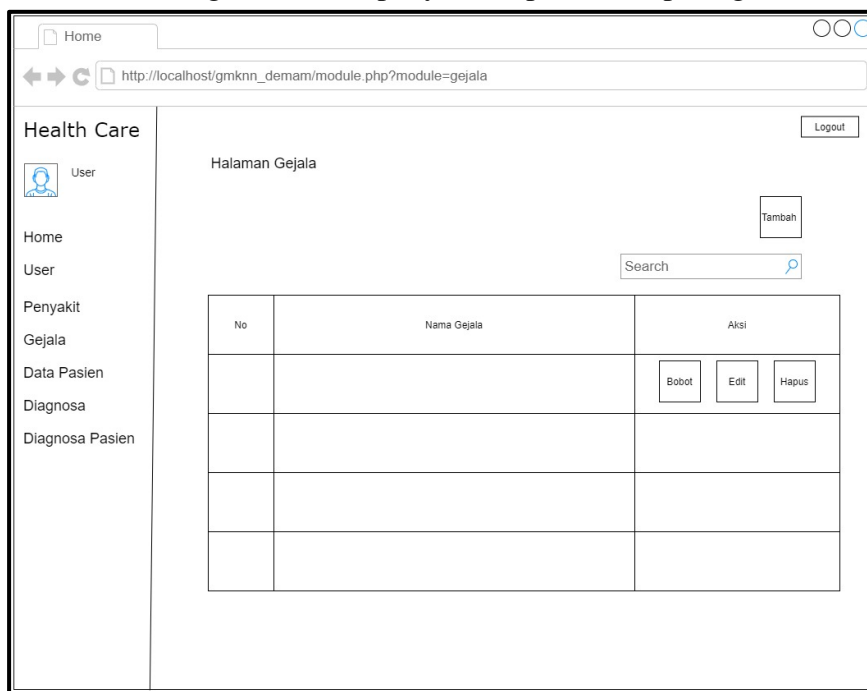
Pada halaman penyakit, admin dapat melihat informasi tentang penyakit yang digunakan dalam sistem. Rancangan halaman penyakit dapat dilihat pada gambar berikut:



Gambar 3.27 Rancangan Halaman Penyakit

4. Rancangan Halaman Gejala

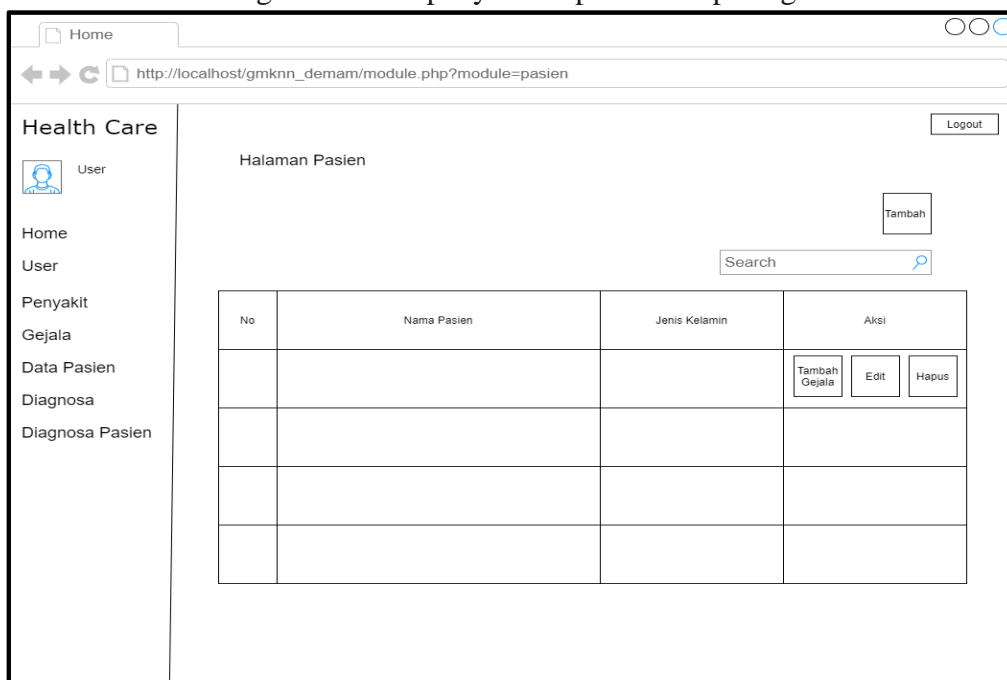
Pada halaman gejala, admin dapat melihat informasi tentang gejala yang digunakan dalam sistem. Rancangan halaman penyakit dapat dilihat pada gambar berikut :



Gambar 3.28 Rancangan Halaman Gejala

5. Rancangan Halaman Data Latih Pasien

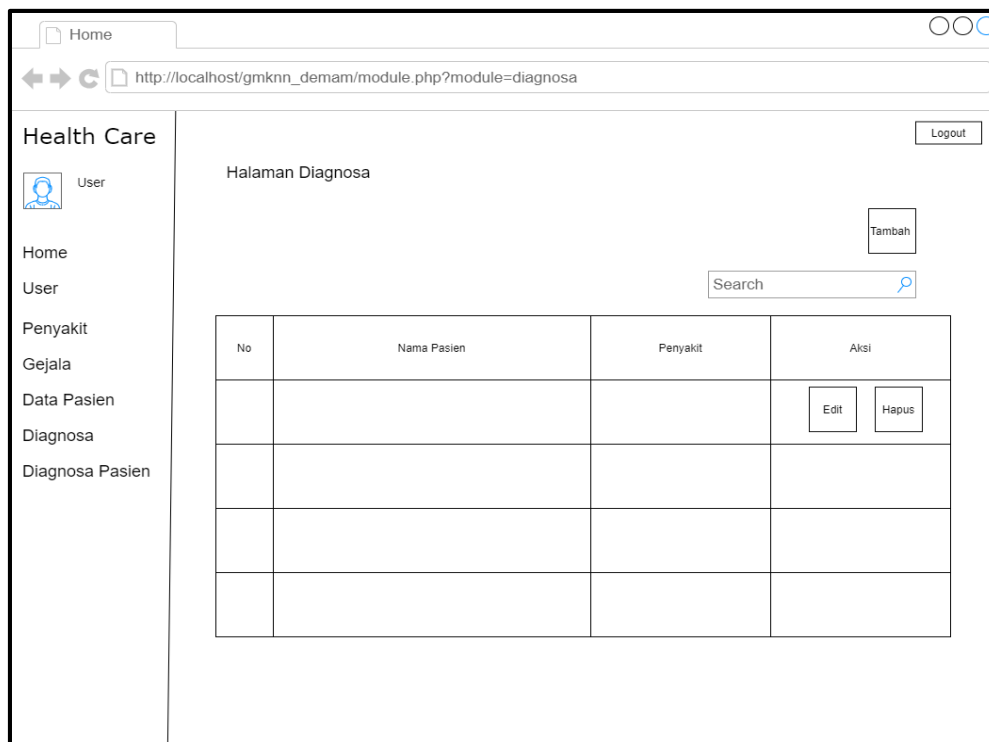
Pada halaman pasien, admin dapat melihat informasi tentang pasien yang digunakan dalam sistem. Rancangan halaman penyakit dapat dilihat pada gambar berikut :



Gambar 3.29 Rancangan Halaman Latih Pasien

6. Rancangan Halaman Diagnosa Latih Pasien

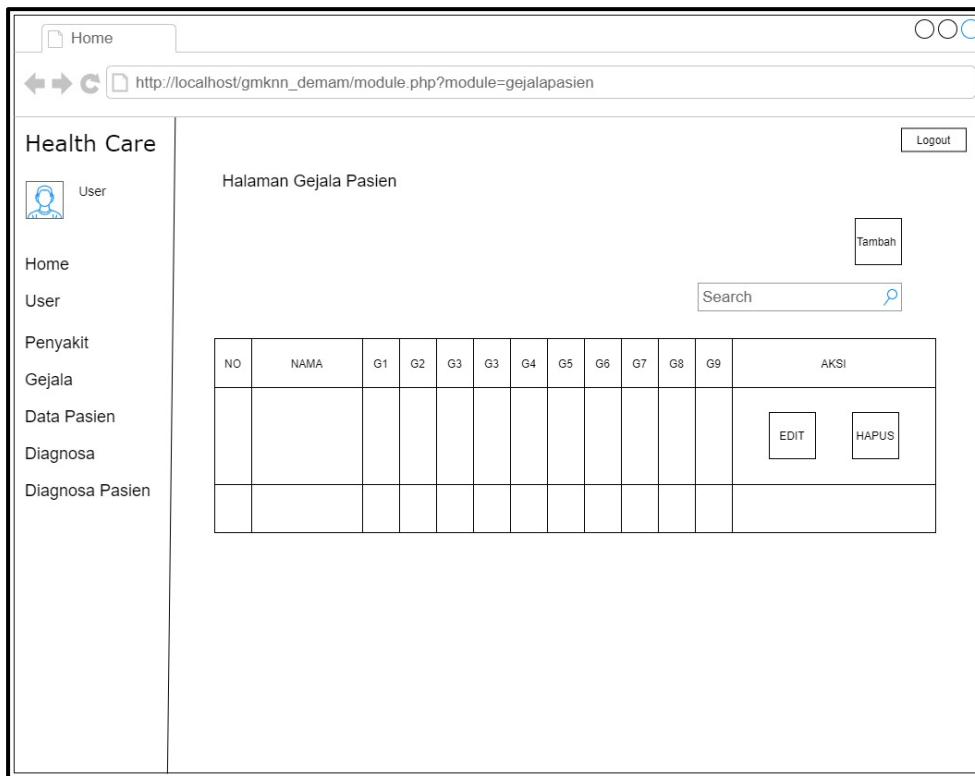
Pada halaman diagnosa admin dapat melihat informasi tentang pasien dan diagnosa penyakit yang digunakan dalam sistem. Rancangan halaman penyakit dapat dilihat pada gambar berikut :



Gambar 3.30 Rancangan Halaman Diagnosa

7. Rancangan Halaman Latih Gejala Pasien

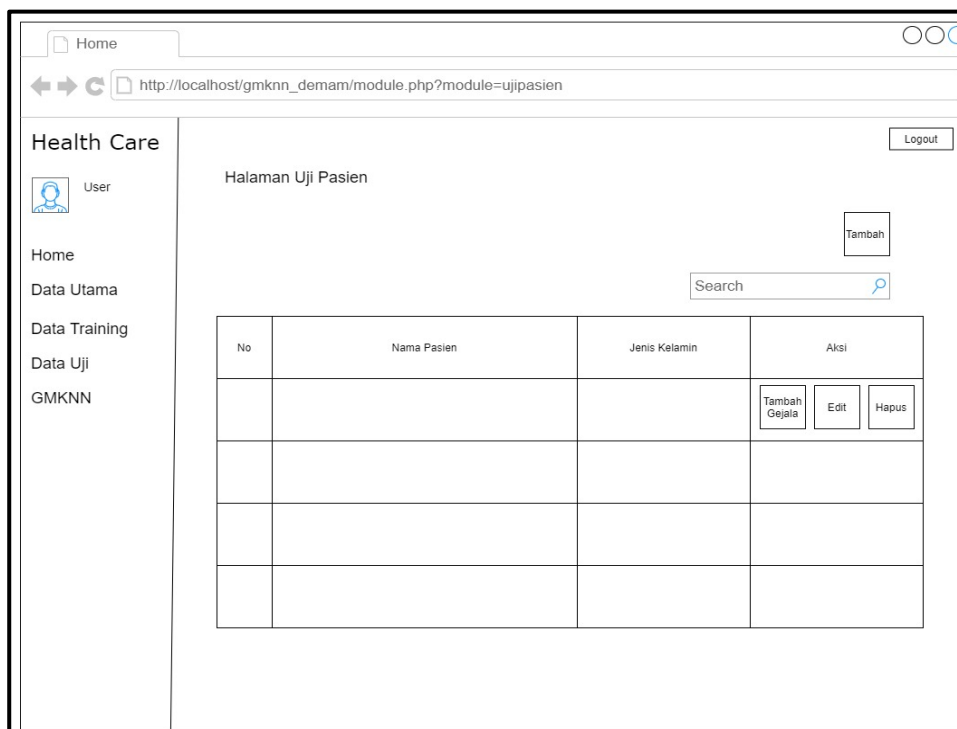
Pada halaman gejala pasien, admin dapat melihat informasi tentang pasien dan gejala yang digunakan dalam sistem. Rancangan halaman penyakit dapat dilihat pada gambar berikut :



Gambar 3.31 Rancang Halaman Gejala Pasien

8. Rancangan Halaman Data Uji Pasien

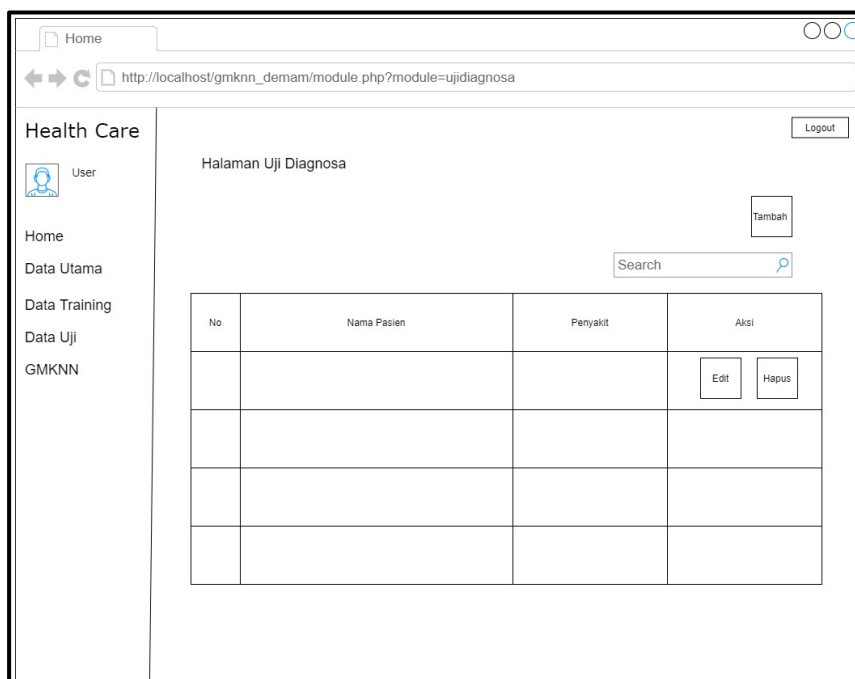
Pada halaman uji pasien, admin dapat melihat informasi tentang uji pasien yang digunakan dalam sistem. Rancangan halaman penyakit dapat dilihat pada gambar berikut:



Gambar 3.32 Rancang Halaman Uji Pasien

9. Rancangan Halaman Diagnosa Uji Pasien

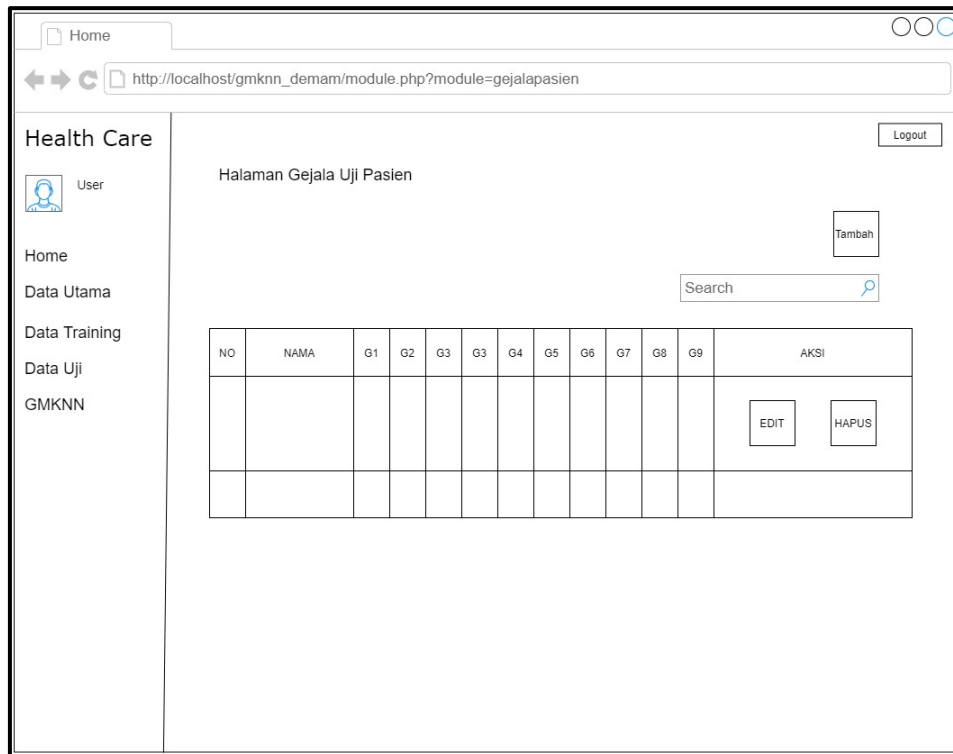
Pada halaman diagnosa uji admin dapat melihat informasi tentang data uji pasien dan diagnosa penyakit yang digunakan dalam sistem. Rancangan halaman penyakit dapat dilihat pada gambar berikut :



Gambar 3.33 Rancang Halaman Uji Diagnosa Pasien

10. Rancangan Halaman Uji Gejala Pasien

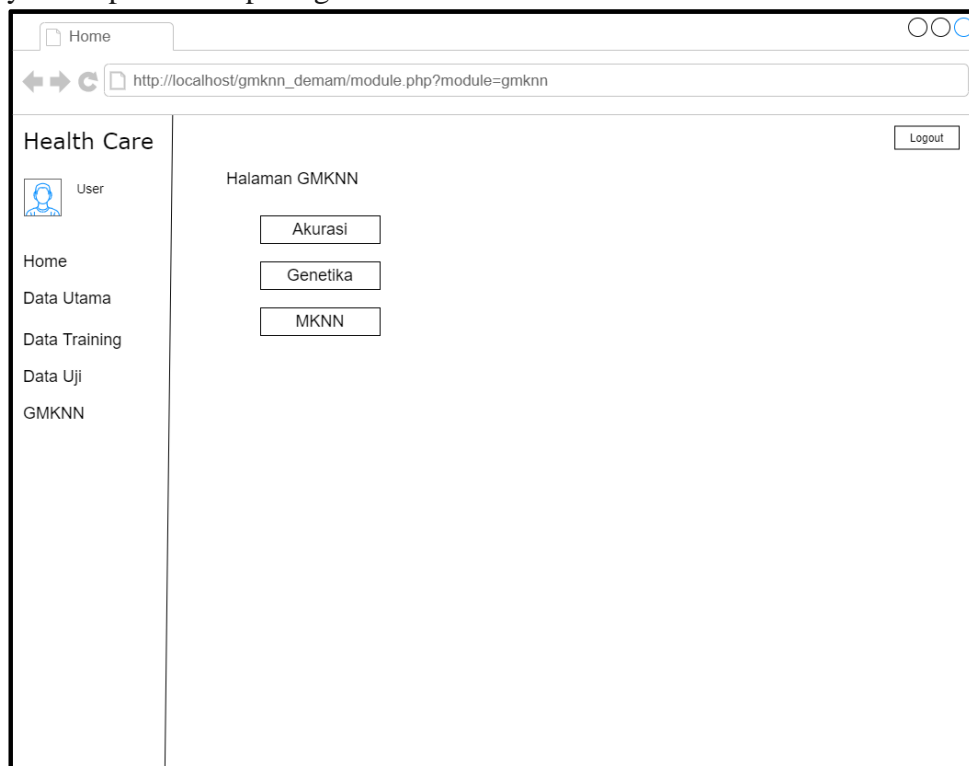
Pada halaman gejala uji pasien, admin dapat melihat informasi tentang data uji pasien dan gejala yang digunakan dalam sistem. Rancangan halaman penyakit dapat dilihat pada gambar berikut :



Gambar 3.34 Rancangan Halaman Gejala Uji Pasien

11. Rancangan Halaman GMKNN

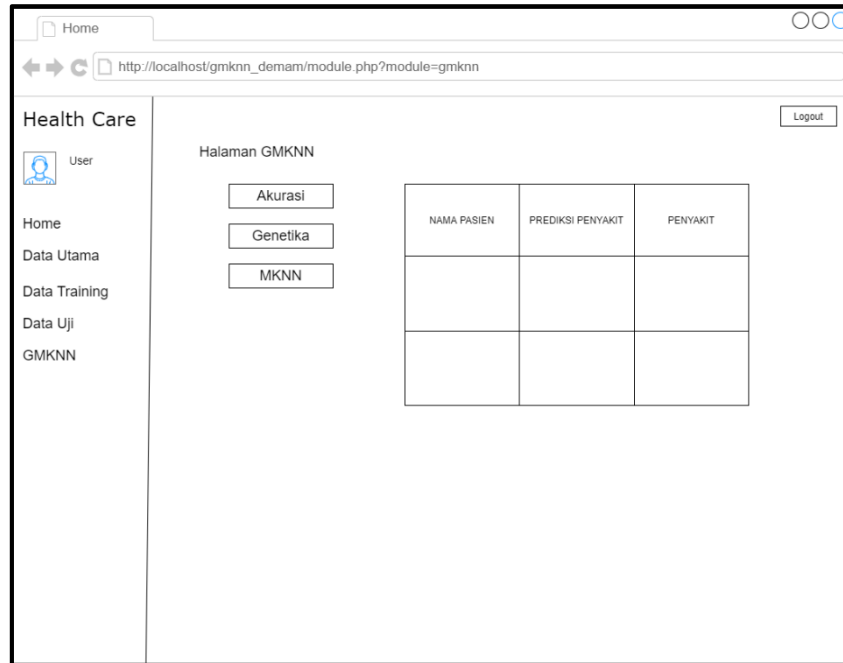
Pada halaman gmkn, admin dapat melihat informasi tentang akurasi, kalkulasi genetika dan kalkulasi mknn yang digunakan dalam sistem. Rancangan halaman penyakit dapat dilihat pada gambar berikut :



Gambar 3.35 Rancangan Halaman GMKNN

12. Rancangan Halaman Akurasi

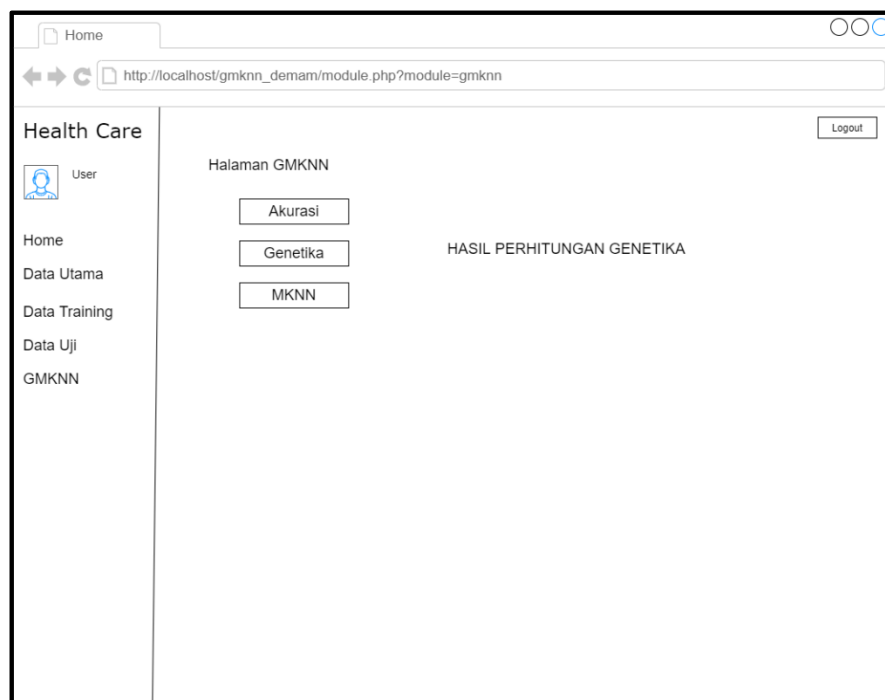
Pada halaman akurasi, admin dapat melihat informasi tentang akurasi yang digunakan dalam sistem. Rancangan halaman penyakit dapat dilihat pada gambar berikut:



Gambar 3.36 Rancangan Halaman Akurasi

13. Rancangan Halaman Genetika

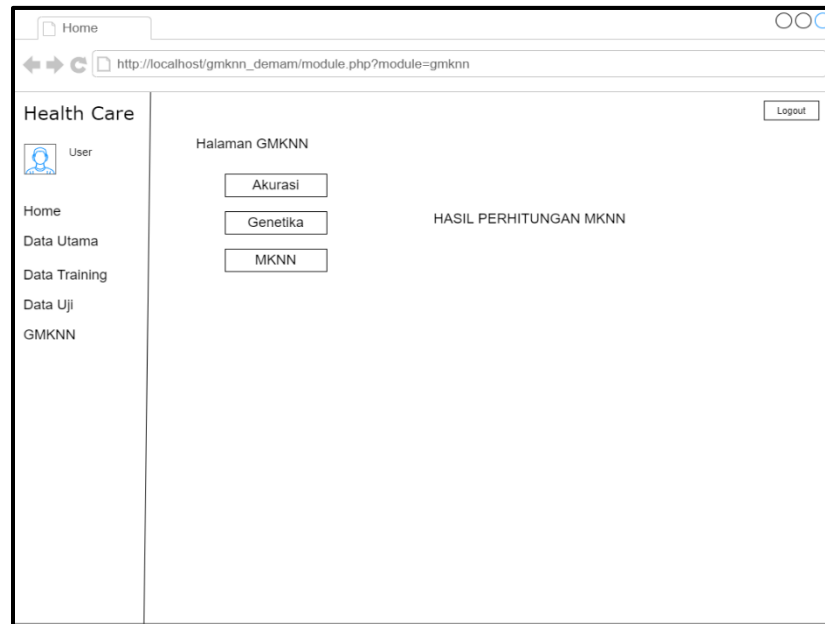
Pada halaman genetika, admin dapat melihat informasi tentang kalkulasi genetika yang digunakan dalam sistem. Rancangan halaman penyakit dapat dilihat pada gambar berikut :



Gambar 3.37 Rancangan Halaman Genetika

14. Rancangan Halaman MKNN

Pada halaman MKNN, admin dapat melihat informasi tentang kalkulasi MKNN yang digunakan dalam sistem. Rancangan halaman penyakit dapat dilihat pada gambar berikut:



Gambar 3.38 Rancangan Halaman MKNN

DAFTAR PUSTAKA

- Candra, A. (2010). Demam Berdarah Dengue: Epidemiologi, Patogenesis, dan Faktor Risiko Penularan. *ASPIRATOR-Journal of Vector-borne Disease Studies*, 2(2).
- Depkes RI. (2013). Riset Kesehatan Dasar. Jakarta: Badan Penelitian dan Pengembangan Kesehatan Kementerian Kesehatan RI.
- Dananjaya, D., Werdiningsih, I., & Semiati, R. (2019). Decision Support System for Classification of Early Childhood Diseases Using Principal Component Analysis and K-Nearest Neighbors Classifier. *Journal of Information Systems Engineering and Business Intelligence*, 5(1), 13-22.
- Gorunescu, F., Belciug, S., Gorunescu, M., & Badea, R. (2012). Intelligent decision-making for liver fibrosis stadialization based on tandem feature selection and evolutionary-driven neural network. *Expert Systems with Applications*, 39(17), 12824-12832.
- Hartini, S., & Pertiwi, P. P. (2016). Efektifitas kompres air hangat terhadap penurunan suhu tubuh anak demam usia 1-3 tahun di SMC RS Telogorejo Semarang. *Karya Ilmiah*.
- Ismanto, B., & Amalia, N. (2018). Peningkatan Akurasi Pada Modified K-NN Untuk Klasifikasi Pengajuan Kredit Koperasi Dengan Menggunakan Algoritma Genetika. *IC-Tech*, 13(2).
- Irawati. (2010). OPTIMISASI PARAMETER SUPPORT VECTOR MACHINE (SVM) MENGGUNAKAN ALGORITME GENETIKA.
- Suhartono, E. (2015). Optimasi Penjadwalan Mata Kuliah dengan Algoritma Genetika (Studi Kasus di AMIK JTC Semarang). *Infokam*, 2, 132–146.
- Ismail, Z., & Irhamah. (2008). Adaptive Permutation-Based Genetic Algorithm for Solving VRP with Stochastic Demands. *Journal of Applied Science* 8(18), 3228-3234.
- Israr, Y. A. (2009). Infeksi Saluran Kemih (ISK). *Riau: Fakultas Kedokteran Universitas Negeri Riau*.
- Kadir, A. (2003). Pengenalan sistem informasi edisi revisi. *Edisi Revisi, January*. <https://doi.org/10.13140/2.1.2637.6328>
- Kusumaningrum, A. P. (2017). *OPTIMASI PARAMETER SUPPORT VECTOR MACHINE MENGGUNAKAN GENETIC ALGORITHM UNTUK KLASIFIKASI MICROARRAY DATA*.
- Lubis, I. N. D., & Lubis, C. P. (2016). Penanganan demam pada anak. *Sari Pediatri*, 12(6), 409-18.
- Mutrofin, S., Izzah, A., Kurniawardhani, A., & Masrur, M. (2015). Optimasi teknik klasifikasi modified k nearest neighbor menggunakan algoritma genetika. *Jurnal Gamma*, 10(1).
- Parvin, H., Alizadeh, H., & Minaei-Bidgoli, B. (2008, October). MKNN: Modified k-nearest neighbor. In *Proceedings of the world congress on engineering and computer science* (Vol. 1). Newswood Limited.

- Scott, D. A., Karr, C. L., & Schinstock, D. E. (1999). Genetic algorithm frequency-domain optimization of an anti-resonant electromechanical controller. *Engineering Applications of Artificial Intelligence*, 12(2), 201-211.
- Shofia, E. N., Putri, R. R. M., & Arwan, A. (2017). Sistem Pakar Diagnosis Penyakit Demam: DBD, Malaria dan Tifoid Menggunakan Metode K-Nearest Neighbor–Certainty Factor. *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, e-ISSN.
- Silmi, M., & Sarwoko, E. A. (2013). Kushartantya. 2013. Sistem Pakar Berbasis Web dan Mobile untuk Mendiagnosis Penyakit Darah pada Manusia dengan Menggunakan Metode Inferensi Forward Chaining. *Journal of Informatics and Technology*, 2(3), 2013.
- Sodikin, Prinsip Perawatan Demam Pada Anak, Pustaka Belajar, Yogyakarta, 2012
- Wafiyah, F., Hidayat, N., & Perdana, R. S. (2017). Implementasi Algoritma Modified K-Nearest Neighbor (MKNN) untuk Klasifikasi Penyakit Demam. *J. Pengemb. Teknol. Inf. dan Ilmu Komput. Univ. Brawijaya*, 1(10), 1210-1219.
- Wardiyah, A., Setiawati, S., & Romayati, U. (2016). Perbandingan efektifitas pemberian kompres hangat dan tepid sponge terhadap penurunan suhu tubuh anak yang mengalami demam di ruang alamanda RSUD dr. H. Abdul Moeloek Provinsi Lampung Tahun 2015. *Holistik Jurnal Kesehatan*, 10(1), 36-44.
- Widodo, D., & Tifoid, D. (2014). Dalam: Buku Ajar Ilmu Penyakit Dalam Jilid I.
- Yusuf, A. A., & Fitri, M. O. (n.d.). *Implementasi sistem informasi akademik pesantren immim putra berbasis web*. 1–7.