

**SISTEM PAKAR DIAGNOSA PENYAKIT DIABETES
MELLITUS MENGGUNAKAN METODE
*CERTAINTY FACTOR***

(Studi Kasus : Rumah Sakit Islam Kota Gorontalo)

Oleh :
AHMAD JUBAIR DAI
T3114129

SKRIPSI

**Untuk Memenuhi Salah Satu Syarat Ujian
Guna Memperoleh Gelar Sarjana**



**PROGRAM SARJANA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS ICHSAN GORONTALO
GORONTALO
2021**

HALAMAN PERSETUJUAN

PENGESAHAN SKRIPSI

SISTEM PAKAR DIAGNOSA PENYAKIT DIABETES MELLITUS MENGGUNAKAN METODE *CERTAINTY FACTOR*

(Studi Kasus : Rumah Sakit Islam Kota Gorontalo)

Oleh :

AHMAD JUBAIR DAI

T3114129

SKRIPSI

Untuk memenuhi salah satu syarat Ujian Akhir
Guna memperoleh gelar Sarjana Program Studi Teknik Informatika,
ini telah disetujui oleh tim pembimbing

Gorontalo, 29 Juni 2021

Pembimbing Utama

Budy Santoso, S.Kom, M.Eng
NIDN. 0908048403

Pembimbing Pendamping

Warid Yunus, M.Kom
NIDN. 0914059001

PENGESAHAN SKRIPSI

PERSETUJUAN SKRIPSI

SISTEM PAKAR DIAGNOSA PENYAKIT DIABETES MELLITUS MENGGUNAKAN METODE *CERTAINTY FACTOR*

Oleh

AHMAD JUBAIR DAI
T3114129

Diperiksa Oleh Panitia Ujian Strata Satu (S1)
Universitas Ichsan Gorontalo

Gorontalo, 8 Juli 2021

1. Ketua Pengudi
Asmaul Husna N, M.Kom
2. Anggota
Sunarto Taliki, M.Kom
3. Anggota
Maryam Hasan, M.Kom
4. Anggota
Budy Santoso, S.Kom, M.Eng
5. Anggota
Warid Yunus, M.Kom



PERNYATAAN SKRIPSI

PERNYATAAN SKRIPSI

Dengan ini saya menyatakan bahwa:

1. Karya tulis (Skripsi) Saya ini adalah asli dan belum pernah diajukan untuk mendapatkan gelar akademik (Sarjana) baik di Universitas Ichsan Gorontalo maupun di perguruan tinggi lainnya
2. Karya tulis (Skripsi) Saya ini adalah murni gagasan, rumusan, dan penelitian saya sendiri,tanpa bantuan pihak lain, kecuali arahan dari tim pembimbing.
3. Dalam karya tulis (Skripsi) Saya ini tidak terdapat karya atau pendapat yang telah dipublikasikan orang lain, kecuali secara tertulis dicantumkan pula dalam daftar pustaka.
4. Pernyataan ini Saya buat dengan sesungguhnya dan apabila dikemudian hari terdapat penyimpangan dan ketidak benaran dalam pernyataan ini, maka saya bersedia menerima sanksi akademik berupa pencabutan gelar yang diperoleh karna karya tulis ini, serta sanksi lainnya sesuai norma-norma yang berlaku di Universitas Ichsan Gorontalo.

Gorontalo , 8 Juli 2021

Yang Membuat Pernyataan



Ahmad Jubair Dai

ABSTRACT

AHMAD JUBAIR DAI. T3114129. EXPERT SYSTEM FOR DIABETES MELLITUS DIAGNOSIS USING CERTAINTY FACTOR METHOD

Patients with Diabetes Mellitus (DM) in Gorontalo City have been increasing from year to year. The inexistence of a computerized system to determine Diabetes Mellitus (DM) diagnose of the community. To find it, the diagnose Diabetes Mellitus employs the Certainty Factor method. In accord with the level of application, this study is Applied Research. while the research method used is the descriptive method. The object of research in this study is Diabetes Mellitus. This application can be used by simply inputting the symptoms suffered by the patients, making it easier to diagnose diabetes mellitus. This expert system needs to be developed into a portable application. The community's reach of this system can be wider.

Keywords: *certainty factor, diabetes, diagnosis, portable application*



ABSTRAK

AHMAD JUBAIR DAI. T3114129. SISTEM PAKAR DIAGNOSA PENYAKIT DIABETES MELLITUS MENGGUNAKAN METODE CERTAINTY FACTOR

Penderita penyakit Diabetes Mellitus (DM) di Kota Gorontalo dari tahun ke tahun mengalami peningkatan. Belum adanya sistem terkomputerisasi untuk menentukan masyarakat mendiagnosa penyakit Diabetes Mellitus (DM). Untuk mengetahui cara diagnosa penyakit diabetes melitus menggunakan metode Certainty Factor. Dipandang dari tingkat penerapannya penelitian ini merupakan penelitian terapan sedangkan metode penelitian yang digunakan adalah metode deskriptif. Yang menjadi objek penelitian pada penelitian ini adalah penyakit diabetes melitus. Aplikasi ini dapat digunakan dengan cara cukup menginput gejala-gejala yang diderita oleh pasien sehingga mempermudah dalam mendiagnosa penyakit diabetes melitus. Sistem pakar ini perlu dikembangkan lagi dalam bentuk aplikasi portable sehingga jangkauan masyarakat terhadap sistem ini dapat lebih luas lagi.

Kata kunci: certainty factor, diabetes, diagnosa, aplikasi portable



KATA PENGANTAR

Bismillahirrahmanirrahim

Puji Syukur penulis panjatkan kehadiran Allah SWT, karena atas berkat dan rahmat-Nya penulis dapat menyelesaikan penelitian ini dengan judul, “Sistem Pakar Diagnosa Penyakit Diabetes Melitus Menggunakan Metode *Certainty Factor*” Studi kasus Rumah Sakit Islam Kota Gorontalo. Untuk memenuhi salah satu syarat mendapat gelar sarjana Program Studi Teknik Informatika Fakultas Ilmu Komputer Universitas Ichsan Gorontalo. Penelitian ini dapat terlaksana dengan baik berkat dukungan dari banyak pihak. Oleh karena itu penulis menyampaikan terimakasih kepada:

1. Bapak Muhamad Ichsan Gaffar, S.E, M.AK, selaku Ketua Yayasan Pengembangan Ilmu Pengetahuan dan Teknologi (YPIPT) Universitas Ichsan Gorontalo.
2. Bapak Dr. Abdul Gaffar Latjoke, M.Si, selaku Rektor Universitas Ichsan Gorontalo.
3. Ibu Zohrahayaty, S.Kom, M.Kom, selaku Dekan Fakultas Ilmu Komputer Universitas Ichsan Gorontalo.
4. Bapak Sudirman S. Panna, M.Kom, selaku Wakil Dekan I Bidang Akademik Fakultas Ilmu Komputer Universitas Ichsan Gorontalo.
5. Ibu Irma Surya Kumala Idris, M.Kom, selaku Wakil Dekan II Bidang Administrasi Umum dan Keuangan Fakultas Ilmu Komputer Universitas Ichsan Gorontalo
6. Bapak Sudirman Melangi, M.Kom, selaku Wakil Dekan III Bidang Kemahasiswaan Fakultas Ilmu Komputer Universitas Ichsan Gorontalo.
7. Bapak Irwan Abraham Salihi, M.Kom, selaku Ketua Jurusan Teknik Informatika Fakultas Ilmu Komputer Universitas Ichsan Gorontalo.
8. Bapak Budy Santoso, M.Eng, selaku Pembimbing Utama dalam penelitian ini yang telah membimbing penulis hingga penelitian ini selesai.

9. Bapak Warid Yunus, S.Kom, M.Kom, sebagai Pembimbing Pendamping dalam penelitian ini yang telah membimbing penulis selama menyelesaikan penelitian ini.
10. Bapak dan Ibu Dosen yang telah mendidik dan membimbing dan mengajarkan berbagai disiplin ilmu kepada penulis.
11. Kepada bapak, Ibu, Kakak, Adik dan Keluaraga yang selalu memberikan dorongan moral maupun materil dari awal sampai akhir perkuliahan.
12. Teman-teman di jurusan Teknik Informatika dan semua pihak yang ikut membantu penulis dalam menyelesaikan Skripsi ini.

Walaupun demikian, penulis menyadari masih banyak kekurangan dalam penyusunan penelitian ini. Oleh karena itu, diharapkan saran dan kritik untuk penyempurnaan penulisan lebih lanjut. Semoga penelitian ini dapat bermanfaat bagi pihak yang berkepentingan terutama bagi penulis sendiri.

Gorontalo, Desember 2021

Penulis

DAFTAR ISI

HALAMAN PERSETUJUAN	v
PENGESAHAN SKRIPSI	vi
PERNYATAAN SKRIPSI.....	vii
<i>ABSTRACT</i>	vii
ABSTRAK	vi
KATA PENGANTAR	vii
DAFTAR ISI.....	ix
DAFTAR GAMBAR	xi
DAFTAR TABEL.....	xii
BAB I PENDAHULUAN	1
1.1 Latar Belakang.....	1
1.2 Identifikasi Masalah.....	4
1.3 Rumusan Masalah.....	5
1.4 Tujuan Penelitian.....	5
1.5 Manfaat Penelitian.....	5
BAB II LANDASAN TEORI	6
2.1 Tinjauan Studi.....	6
2.2 Tinjauan Pustaka.....	8
2.2.1 Diabetes Melitus.....	8
2.2.2 Klasifikasi Diabetes Melitus	8
2.2.3 Diagnosis Diabetes Melitus.....	9

2.2.4	<i>Manifestasi Klinis</i> Diabetes Melitus	10
2.2.5	Sistem Pakar.....	10
2.2.6	Ciri-ciri Sistem Pakar.....	11
2.2.7	Jenis-Jenis Sistem Pakar	12
2.2.8	Refresentasi Pengetahuan.....	13
2.2.9	Faktor kepastian (<i>Certainty Factor</i>).....	13
2.2.10	Siklus Hidup Pengembangan Sistem	22
2.2.11	Desain Sistem.....	25
2.2.12	Teknik Pengujian Sistem.....	34
2.3	KerangkaPemikiran.....	41
	BAB III METODE PENELITIAN.....	42
3.1	Jenis, Metode dan Objek Penelitian.....	42
3.2	Pengumpulan Data.....	42
3.3	Konstruksi Sistem.....	43
3.4	Analisa Sistem.....	43
3.5	Tahap Pengujian.....	43
	BAB IV HASIL PENELITIAN	44
4.1	HasilPengembangan Sistem.....	44
4.1.1	<i>Sequence Diagram</i>	45
4.1.2	Kamus Data.....	49
4.1.3	Desain Keluaran Secara Umum	53
4.1.4	Desain <i>Input</i> Secara Umum	53
4.1.5	Desain <i>Database</i> Secara Umum.....	54
4.1.5	Desain Sistem Secara Terinci.....	54
4.1.6	Desain <i>Database</i> Secara Terinci	57

4.2 Pengujian Sistem.....	60
4.2.1 <i>Listing</i> Program Uji <i>White Box</i>	60
4.2.2 <i>Flowchart</i> Program Uji <i>White Box</i>	61
BAB V PEMBAHASAN PENELITIAN.....	62
5.1 Pembahasan Sistem.....	62
5.1.1 Tampilan Halaman Utama	62
5.1.2 Tampilan Halaman <i>Login Admin</i>	63
5.1.3 Tampilan Halaman Admin	64
5.1.4 Tampilan <i>Form Input</i> Data Admin	65
5.1.5 Tampilan <i>Form Input</i> Data Penyakit.....	66
5.1.6 Tampilan <i>Form Input</i> Data Gejala	67
5.1.7 Tampilan <i>Form Input</i> Pengetahuan 1.....	68
5.1.8 Tampilan <i>Form Input</i> Pengetahuan 2.....	69
5.1.9 Tampilan <i>Form Input</i> Diagnosa	70
5.1.10 Tampilan <i>Form Input</i> Nilai CF <i>User 1</i>	71
5.1.11 Tampilan <i>Form Input</i> Nilai CF <i>User 2</i>	72
5.1.12 Tampilan <i>Form</i> Hasil Perhitungan.....	73
5.1.13 Tampilan <i>Form</i> Hasil Diagnosa	74
BAB VI KESIMPULAN DAN SARAN	75
6.1 Kesimpulan.....	75
6.2 Saran.....	75
DAFTAR PUSTAKA	
LAMPIRAN	

DAFTAR GAMBAR

Gambar 2. 1 Proses Akuisisi Pengetahuan.....	13
Gambar 2. 2 <i>Use Case Diagram</i> : Whitten & Bentley (2007:246) [14].....	27
Gambar 2. 3 <i>Activity Diagram</i> : (Whitten & Bentley, 2007:392) [14].....	31
Gambar 2. 4 <i>Sequence Diagram</i> : (Whitten & Bentley, 2007:396) [14].....	33
Gambar 2. 5 Bagan Air: Pressman.Roger S.....	36
Gambar 2. 6 <i>Flowgraph</i> :Roger S. Pressman.....	37
Gambar 2. 7 Kerangka Pemikiran.....	41
Gambar 4. 1 <i>Use Case Diagram</i> Sistem Pakar Diabetes.....	44
Gambar 4. 2 <i>Sequence Diagram Input Admin</i>	45
Gambar 4. 3 <i>Sequence Diagram Tampilan Admin</i>	45
Gambar 4. 4 <i>Sequence Diagram Input Data Penyakit</i>	46
Gambar 4. 5 <i>Sequence Diagram Tampil Data Penyakit</i>	46
Gambar 4. 7 <i>Sequence Diagram Input Data Gejala</i>	47
Gambar 4. 8 <i>Sequence Diagram Tampil Data Gejala</i>	47
Gambar 4. 9 <i>Sequence Diagram Tampil Data Pengetahuan</i>	48
Gambar 4. 10 <i>Sequence Diagram Diagnosa</i>	48
Gambar 4. 11 Desain <i>Input Data Admin</i>	54
Gambar 4. 12 Desain <i>Input Data Penyakit</i>	55
Gambar 4. 13 Desain <i>Input Data Gejala</i>	55
Gambar 4. 14 Desain <i>Input Data Pengetahuan</i>	56
Gambar 4. 15 Desain <i>Input Data User</i> dan Gejala yang Dirasakan.....	56
Gambar 4. 16 Desain <i>Input Data Nilai User</i>	57
Gambar 4. 17 <i>Flowchart</i> Program Uji <i>White Box</i>	61
Gambar 5. 1 Halaman Utama.....	62
Gambar 5. 2 Halaman Login Admin.....	63
Gambar 5. 3 Halaman Admin	64
Gambar 5. 4 Form Input Data Admin	65
Gambar 5. 5 Form Input Data Penyakit	66
Gambar 5. 6 Form Input Data Gejala.....	67

Gambar 5. 7 Form Input Pengetahuan 1	68
Gambar 5. 8 Form Input Pengetahuan 2	69
Gambar 5. 9 Form Input Diagnosa.....	70
Gambar 5. 10 Form Input Nilai CF User 1	71
Gambar 5. 11 Form Input Nilai CF User 2	72
Gambar 5. 12 Form Hasil Perhitungan	73
Gambar 5. 13 Form Hasil Diagnosa.....	74

DAFTAR TABEL

Tabel 1. 1Data pasien penyakit Diabetes Mellitus (DM) dari tahun 2018 - 2020 di Rumah Sakit Islam Kota Gorontalo.....	3
Tabel 2. 1 Penelitian Tentang Sistem pakar Menggunakan Metode Certainty Factor	6
Tabel 2. 2 Input gejala oleh user atau pengguna.....	20
Tabel 2. 3 Perhitungan CF kombinasi.....	20
Tabel 2. 4 Notasi Use Case Diagram	27
Tabel 2. 5 Notasi Class Diagram.....	29
Tabel 2. 6 Diagram Activity Notasi	32
Tabel 2. 7 Notasi Diagram Sequence.....	33
Tabel 2. 8 Hubungan antara Cyclomatic Complexity dan Resiko	38
Tabel 4. 1 Kamus Data : Data Admin	49
Tabel 4. 2 Kamus Data : Data Penyakit	50
Tabel 4. 3 Kamus Data : Data Gejala.....	50
Tabel 4. 4 Kamus Data : Data Nilai CF Pakar	51
Tabel 4. 5 Kamus Data : Data Gejala User	51
Tabel 4. 6 Kamus Data : Hasil Diagnosa	52
Tabel 4. 7 Rancangan Output Secara Umum	53
Tabel 4. 8 Rancangan Input secara umum	53
Tabel 4. 9 Rancangan Database Secara Umum	54
Tabel 4. 13 Data Tabel D_Admin	57
Tabel 4. 14 Data Tabel D_Nilai CF Pakar	58
Tabel 4. 15 Data Tabel D_Gejala.....	58
Tabel 4. 16 Data Tabel D_Penyakit	58
Tabel 4. 17 Data Tabel D_Hasil Diagnosa.....	59

BAB I

PENDAHULUAN

1.1 Latar Belakang

Penyakit Diabetes merupakan salah satu penyakit kronis yang diderita oleh manusia selama hidupnya hingga progresifitas terus meningkat yang pada suatu saat dapat menimbulkan komplikasi dengan penyakit lain. Penyakit ini dalam istilah kedokteran dikenal dengan Diabetes Mellitus (DM). Diabetes Mellitus pada awalnya memiliki gejala-gejala yang ringan sampai berat. Namun, untuk beberapa kasus, penyakit ini dapat menyebabkan kematian baik dari komplikasi akut maupun kronis. Dikutip dari Pusat Data dan Informasi Departemen Kesehatan tahun 2020 bahwa diabetes ada dua jenis, yaitu diabetes yang timbul karena kekurangan insulin disebut sebagai DM tipe 1 atau insulin dependent diabetes mellitus (IDDM), sedangkan diabetes disebabkan oleh Insulin tidak bekerja dengan baik, yang dikenal sebagai DM tipe 2 atau diabetes mellitus tidak tergantung insulin (NIDDM)[1].

Selain itu, Pusat Data dan informasi Kementerian Kesehatan RI tahun 2020 menyatakan bahwa diabetes tidak hanya menyebabkan kematian prematur di seluruh dunia tetapi juga penyakit Diabetes Mellitus (DM) ini juga menjadi penyebab utama kebutaan, penyakit jantung dan gagal ginjal. Oleh karena itu, Diabetes mellitus telah menjadi penyebab kematian terbesar keempat di dunia dan jumlahnya dari tahun ke tahun makin bertambah. Organisasi International Diabetes Federation (IDF) memperkirakan sedikitnya terdapat 463 juta orang pada usia 20-79 tahun di dunia menderita Diabetes Mellitus (DM). Sementara itu, pada tahun 2019, Indonesia menduduki peringkat ke 7 dunia dari 10 negara dengan jumlah penderita Diabetes Mellitus (DM) dengan presentasi 10,7% [2].

Hampir semua provinsi di Indonesia menunjukkan peningkatan prevalensi, hal ini dapat dilihat dari data Kementerian Kesehatan Republik Indonesia tahun 2018 yaitu pada tahun 2013-2018 penyakit Diabetes mengalami peningkatan

pravalensi pada beberapa provinsi, kecuali Nusa Tengara Timur. Dari data tersebut pada tahun 2013 terdapat empat provinsi dengan prevalensi tertinggi yakni pada provinsi DI Yogyakarta, DKI Jakarta, Sulawesi Utara dan Kalimantan Timur. Sementara itu pada tahun 2018, terdapat beberapa provinsi dengan peningkatan preverensi tertinggi, yaitu Riau, DKI Jakarta, Banten, Gorontalo dan Papua Barat [2]. Provinsi Gorontalo berdasarkan data Kementerian Kesehatan pada Tahun 2018 menduduki peringkat ke 8 dengan presentasi 2,4% dari total penderita penyakit Diabetes Mellitus (DM) di Indonesia.

Prevalensi kasus Diabetes Mellitus di Provinsi Gorontalo mengalami peningkatan selama 3 tahun terakhir, hal tersebut diperoleh dari data sekunder Dinas Kesehatan Provinsi Gorontalo,yaitu pada tahun 2017 mengalami peningkatan sebesar 3,5%,tahun 2018 mengalami peningkatan sebesar 5,1% dan tahun2019 mengalami peningkatan sebesar 7,4%. Oleh karena itu, pada tahun 2019 Kota Gorontalo mendudukiperingkat kedua yang mengalami peningkatan *prevalensi* penyakit Diabetes Mellitus tertinggi setelah Kabupaten Gorontalo dari seluruh Kabupaten/Kota yang ada di Provinsi Gorontalo yaitu sebanyak 2.881 kasus (5,6%).

Rumah Sakit Islam Kota Gorontalo merupakan salah satu rumah sakit yang menerima pasien penyakit Diabetes Mellitus (DM), Rumah Sakit Islam Kota Gorontalo menerima pasien dengan diagnosa DM kategori rawat inap yang dirawat oleh dokter spesialis penyakit dalam. Ada beberapa pasien rawat inap diagnosa DM yang memiliki luka/*ganggren*, biasa di rawat oleh dokter spesialis bedah, untuk pasien kategori rawat jalan dirawat oleh dokter spesialis bedah/dokter spesialis dalam (jika masuk di poliklinik) dan di rawat dokter umum (jika masuk di Poli umum/UGD). Berikut ini data pasien penyakit Diabetes Mellitus (DM) selama 3 tahun terakhir :

Tabel 1. Data pasien penyakit Diabetes Mellitus (DM) dari tahun 2018 - 2020 di Rumah Sakit Islam Kota Gorontalo.

No	Tahun	Jumlah Pasien
1	2018	27
2	2019	253
3	2020	348

(Sumber: Rumah Sakit Islam Kota Gorontalo, 2020)

Berdasarkan data pasien penyakit Diabetes Mellitus (DM) dari tahun 2018-2020 di Rumah Sakit Islam Kota Gorontalo mengalami peningkatan yang signifikan, hal ini menjadi permasalahan bagi pihak pengelolah Rumah Sakit Islam Kota Gorontalo yang dikarenakan minimnya dokter penyakit dalam yang hanya berjumlah 2 orang. Kurangnya dokter-dokter spesialis diabetes mellitus yang masih terbatas yang ada di Rumah Sakit Islam Kota Gorontalo dan Kota Gorontalo menjadi masalah disamping pengetahuan yang kurang mengenai gejala dan cara menangani penyakit diabetes mellitus, dan harga pemeriksaan di dokter praktek yang mahal merupakan salah satu sebab meningkatnya jumlah orang yang terkena penyakit Diabetes Mellitus (DM) di Kota Gorontalo. Hal ini menjadi awal untuk merancang sebuah aplikasi sistem pakar berbasis web yang dapat mengatasi nilai derajat kepercayaan atau faktor kepastian data yang diperoleh dari hasil konsultasi dengan pasien.

Dikutip dari jurnal Khairina Eka Setyaputri, Abdul fadlil dan Sunardi tahun 2018 menyatakan bahwa Sistem Pakar adalah cabang ilmu komputer yang bekerja atas dasar kecerdasan buatan. Sistem ini mentransfer pengetahuan manusia ke komputer sehingga komputer dapat memecahkan masalah seperti yang biasa dilakukan para ahli[3]. Guna mengetahui tingkat penyakit DM, maka dibuat sistem pakar diagnosa Diabetes berbasis web dengan mengimplementasikan salah satu metode, yaitu metode *Certainty Factor*. Aplikasi diagnosa penyakit Diabetes Mellitus (DM) dengan menggunakan metode *Certainty Factor* dilakukan bertujuan untuk mendeteksi dini jenis penyakit Diabetes Mellitus (DM) dan perawatan penyakit Diabetes Mellitus (DM) secara komputerisasi. Adapun hasil dari

penelitian ini adalah sebuah aplikasi yang dirancang agar dapat mendeteksi dan menelusuri masalah yang dimulai dari adanya faktor yang mempengaruhi sehingga dapat disimpulkan masalah apa yang dihadapi.

Metode *Certainty Factor* (CF) adalah sebuah metode yang membuktikan suatu fakta apakah pasti atau tidak yang berbentuk *metric* yang digunakan sistem pakar[4]. Metode ini bekerja dengan menunjukkan derajat kepastian terhadap suatu fakta atau aturan yang ada. Konsep dari metode CF ini adalah berargumentasi layaknya seorang ahli demi menjaga nilai kepercayaan. Proses perhitungan untuk metode CF dilakukan dengan menghitung nilai perkalian antara nilai *user cf* dengan nilai *expert cf* dan menghasilkan nilai CF gabungan. Nilai CF kombinasi tertinggi adalah keputusan akhir dari metode CF[5].

Pada penelitian sebelumnya yang dilakukan oleh Indyah Hartami Santi, Bina Andari, 2019. Dengan judul Sistem Pakar Untuk Mengidentifikasi Jenis Kulit Wajah dengan Metode *Certainty Factor*. Penelitian ini dilakukan melalui beberapa tahapan yaitu : Data diperoleh melalui 40 responden wanita yang hasilnya terdapat 100% responden tidak memahami jenis kulit wajah, dan 76% responden mengatakan mereka membutuhkan ahli, sehingga sebanyak 95% responden membutuhkan aplikasi sistem pakar. Dari hasil penelitian tersebut menyatakan bahwa 88% dari desain sistem sangat baik , 91% dari sistem mudah digunakan, dan 98% mengatakan operasi itu dengan apa yang dibutuhkan[5].

Berdasarkan hal tersebut di atas, maka penulis menyimpulkan akan mengangkat Judul Penelitian tentang “**Sistem Pakar Diagnosa Penyakit Diabetes Melitus Menggunakan Metode *Certainty Factor***” Studi kasus Rumah Sakit Islam Kota Gorontalo.

1.2 Identifikasi Masalah

Berdasarkan latar belakang di atas maka permasalahan yang di ambil yaitu:

1. Penderita penyakit Diabetes Mellitus (DM) di Kota Gorontalo dari tahun ke tahun mengalami peningkatan.
2. Belum adanya sistem terkomputerisasi untuk menentukan masyarakat mendiagnosa penyakit Diabetes Mellitus (DM).

1.3 Rumusan Masalah

Dari uraian latar belakang masalah diatas, dapat dirumuskan masalah pokok yang berkaitan yaitu :

1. Bagaimana cara diagnosa penyakit diabetes melitus menggunakan metode *Certainty Factor*?
2. Bagaimana hasil penerapan metode *Certainty Factor* untuk diagnosa penyakit diabetes melitus?

1.4 Tujuan Penelitian

Adapun tujuan dari pembuatan software ini yaitu:

1. Untuk mengetahui cara diagnosa penyakit diabetes melitus menggunakan metode *Certainty Factor*.
2. Untuk mengetahui hasil penerapan metode *Certainty Factor* untuk diagnosa penyakit diabetes melitus.

1.5 Manfaat Penelitian

Penelitian ini dimohonkan membawa faedah secara teoretis dan praktis kepada pihak terkait, sebagai pertimbangan, masukan dan pedoman serta evaluasi.

1. Secara Teoritis, Penelitian ini diharapkan dapat mengembangkan ilmu pengetahuan bagi akademis sebagai bahan masukan pemikiran mengenai permasalahan diagnosa penyakit diabetes melitus dengan efektif dan efesien, dan memberikan sumber informasi bagi mahasiswa apabila melakukan penelitian yang sejenis.
2. Secara Praktis, penelitian ini diharapkan dapat mengembangkan ilmu pengetahuan bagi perusahaan sebagai bahan informasi agar dapat mengoptimalkan aplikasi sistem pakar diagnosa penyakit diabetes melitus dengan terperinci secara terus menerus.

BAB II

LANDASAN TEORI

2.1 Tinjauan Studi

Beberapa penelitian yang melekat tentang Sistem pakar dan penerapan metode *Certainty Factor*, seperti di bawah ini:

Tabel 2. 1 Penelitian Tentang Sistem pakar Menggunakan Metode Certainty Factor

Peneliti	Judul	Hasil
Indyah Hartami Santi, Bina Andari, 2019.[5]	Sistem Pakar Untuk Mengidentifikasi Jenis Kulit Wajah dengan Metode <i>Certainty Factor</i>	Penelitian ini dilakukan dalam beberapa tahapan yaitu: Pendataan dilakukan pada 40 responden wanita, dengan 100% responden tidak memahami jenis kulit wajah dan 76% mengatakan membutuhkan ahli, 95% membutuhkan ahli aplikasi sistem. Sedangkan hasil evaluasi sistem aplikasi yang dibuat oleh responden menyatakan bahwa 88% desain sistem sangat baik dan sangat baik, 91% sistem mudah digunakan, dan 98% mengatakan perlu sesuatu untuk dioperasikan.

Adi Sucipto, Yusra Fernando, Rohmat Indra Borman, Nisa Mahmuda, 2019. [6]	Penerapan Metode <i>Certainty Factor</i> Pada Diagnosa Penyakit Saraf Tulang Belakang	Sistem pakar dapat membantu dalam mendiagnosis penyakit, dimana sistem ini dimaksudkan untuk merekonstruksi pengetahuan spesialis dan kemampuan nalar seorang pakar. Penelitian ini mengimplementasikan faktor keamanan dalam penerapan diagnosis penyakit tulang belakang. Sistem diuji pada serangkaian input dan hasil pengujian yang diperoleh memberikan hasil sesuai dengan perhitungan manual. Hasil pengujian dengan uji coba pada rangkaian input menunjukkan bahwa pengujian memberikan hasil yang sesuai dengan perhitungan manual. Saat menguji kebenaran kesesuaian data uji yang diterima dari pakar dengan keluaran sistem pakar, hasil keluaran yang sesuai adalah 90%.
Mohammad Arifin, Slamin, Windi Eka Yulia Retnani, 2017.[7]	Penerapan Metode <i>Certainty Factor</i> Untuk Sistem Pakar Diagnosis Hama Dan Penyakit Pada Tanaman Tembakau	Sistem pakar diagnosa hama dan penyakit pada tanaman tembakau pada penelitian ini menggunakan metode Certainty Factor untuk menentukan derajat kepastian suatu hama atau penyakit berdasarkan data gejala yang dipilih. Proses konsultasi yang dilakukan oleh pengguna untuk mendapatkan hasil data hama atau penyakit secara persentase, nilai tertinggi yang dicapai adalah 99.985729744%. walaupun tidak pernah mencapai 100%, namun penggunaan metode faktor kepastian dalam penerapan sistem pakar masih sangat cocok

2.2 Tinjauan Pustaka

2.2.1 Diabetes Melitus

Menurut *American Diabetes Association* (ADA) 2012 dalam jurnal P. Hikmat tahun 2017 bahwa diabetes mellitus adalah sekelompok penyakit metabolism yang ditandai dengan hiperglikemia akibat gangguan sekresi insulin, gangguan kerja insulin, atau keduanya dan berbagai komplikasi kronis pada mata, ginjal, saraf dan saraf, serta pembuluh darah [1]. Selain itu, data Kementerian Kesehatan Republik Indonesia tahun 2020 menunjukkan bahwa diabetes melitus merupakan gangguan metabolisme karbohidrat, protein, dan lemak yang ditandai dengan *hiperglikemia*, *aterosklerosis*, *mikroangiopati*, dan *neuropati*. *Hiperglikemia* terjadi sebagai akibat dari kekurangan insulin atau berkurangnya efek insulin [2]. Dari kedua definisi di atas dapat disimpulkan bahwa diabetes melitus merupakan penyakit yang diderita manusia akibat kelainan insulin dalam tubuh.

2.2.2 Klasifikasi Diabetes Melitus

Dikutip dari jurnal P. Hikmat tahun 2017 Diabetes melitus dapat diklasifikasikan menjadi 4 kategori klinis yaitu [1]:

1. Diabetes mellitus tipe 1. Pada tipe ini, diabetes mellitus disebabkan oleh kerusakan sel beta pankreas, sehingga terjadi defisiensi insulin *absolut*. Umumnya, penyakit berkembang menjadi *ketoasidosis diabetik* yang menyebabkan kematian. Dalam bentuk ini, diabetes mellitus biasanya terjadi sebelum usia 30 tahun dan harus disuplai dengan insulin eksternal. Beberapa faktor risiko pada diabetes mellitus tipe ini adalah: penyakit *autoimun*, infeksi virus, riwayat keluarga diabetes mellitus.
2. Diabetes mellitus tipe 2. Diabetes mellitus tipe 1 lainnya biasanya menyerang pankreas, yang memproduksi insulin relatif tetapi insulin tidak berfungsi dengan sempurna karena resistensi insulin yang disebabkan oleh obesitas. Faktor genetik dan gaya hidup juga menjadi penyebabnya. Faktor risiko diabetes tipe 2 adalah obesitas, stres fisik dan emosional, kehamilan di atas 40 tahun, pengobatan, dan riwayat keluarga

diabetes mellitus. Hampir 90% penderita diabetes melitus adalah diabetes melitus tipe 2.

3. Diabetes mellitus dengan kehamilan atau *gestational diabetes mellitus* (GDM) adalah penyakit diabetes mellitus yang terjadi selama kehamilan, meskipun sebelumnya kadar gula darah selalu normal. Tipe ini akan kembali normal setelah melahirkan. Faktor risiko DMG adalah wanita hamil dengan usia lebih dari 25 tahun, disertai riwayat keluarga diabetes mellitus, infeksi berulang, melahirkan bayi dengan berat badan lebih dari 4 kg.
4. Diabetes tipe lain disebabkan oleh defek genetik pada fungsi sel beta, defek genetik pada fungsi insulin, gangguan *eksokrin pankreas*, *endokrinopati* obat atau kimiawi, infeksi, dan *sindrom* genetik lain yang berhubungan dengan diabetes mellitus. Beberapa hormon seperti hormon pertumbuhan, *kortisol*, *glukagon* dan *adrenalin* memiliki efek antagonis atau melawan efek insulin. Kelebihan hormon ini dapat menyebabkan diabetes mellitus jenis ini

2.2.3 Diagnosis Diabetes Melitus

Kriteria diagnosis DM menurut Perkumpulan *Endokrinologi* Indonesia adalah jika ditemukan gejala sebagai berikut [1]:

1. Keluhan klasik DM yaitu *polidipsia* (sering minum), *poliuria* (sering kencing), *polifagia* (sering makan) dan penurunan pada berat badan.
2. Keluhan lain yang sering terjadi pada penderita diabetes melitus adalah kelemahan, kesemutan, gatal-gatal, penglihatan kabur, dan *disfungsi ereksi* pada pria, serta *pruritus vulva* pada wanita..
3. Tes *glukosa plasma* puasa 126 mg/dL. Puasa adalah keadaan di mana tidak ada asupan kalori selama minimal 8 jam.
4. Cek *glukosa plasma*, yaitu 200 mg/dL 2 jam setelah tes toleransi *glukosa oral* (TTGO).
5. Pemeriksaan *glukosa plasma* sewaktu $\geq 200\text{mg/dL}$ bersamaan dengan keluhan klasik DM.

6. Pemeriksaan HbA1c $\geq 6,5\%$.

2.2.4 *Manifestasi Klinis Diabetes Melitus*

Menurut Kementerian Kesehatan RI tahun 2020, ditemukan bahwa *manifestasi klinis* diabetes mellitus dapat dibagi menjadi gejala akut dan kronis [2]:

- 1) Gejala Akut Penyakit Diabetes Melitus

Gejala diabetes mellitus bervariasi dari pasien ke pasien, mereka mungkin tidak menunjukkan gejala untuk beberapa waktu. Gejala yang terjadi pada penderita diabetes awalnya banyak makan (*polifagia*), banyak minum (*polidipsia*), dan banyak buang air kecil (*poliuria*).

Situasi ini, jika tidak segera diobati, menyebabkan gejala akut. pada penderita diabetes mellitus banyak minum, banyak buang air kecil, nafsu makan hilang/berat badan cepat turun (menurun 5-10 kg dalam 3-4 minggu), mudah lelah, walaupun tidak segera diobati, terjadi mual, oleh karena itu penderita diabetes mellitus jatuh ke dalam koma yang dikenal sebagai koma diabetes.

- 2) Gejala Kronik Diabetes Mellitus

Gejala lain yang terjadi pada penderita diabetes melitus M adalah gejala kronis. Gejala kronis ini ditandai dengan sensasi kesemutan, panas atau kesemutan pada kulit seperti kesemutan, rasa tebal di kulit, kram, mata kabur yang biasanya sering berganti kacamata, sedikit mengantuk, gigi mudah lepas dan rontok, gatal-gatal di sekitar alat kelamin, terutama wanita, penurunan kemampuan seksual, bahkan impotensi, dan ibu hamil sering mengalami keguguran atau kematian janin dalam kandungan atau bayi dengan berat 4 kg.

2.2.5 Sistem Pakar

Sistem pakar adalah paket perangkat keras dan perangkat lunak yang digunakan sebagai pengambil keputusan dan/atau pemecah masalah; yang dapat mencapai tingkat atau kadang-kadang bahkan lebih tinggi dari seorang ahli / ahli

dalam bidang masalah tertentu dan biasanya lebih sempit dan merupakan cabang dari aplikasi Kecerdasan Buatan [8].

Sistem pakar adalah aplikasi berbasis komputer yang digunakan untuk memecahkan masalah yang dipikirkan oleh para ahli. Pakar adalah orang-orang dengan keahlian khusus yang dapat memecahkan masalah yang tidak dapat dipecahkan oleh orang normal. Teknisi servis, misalnya, adalah seorang ahli yang mampu mengidentifikasi kerusakan mesin pada kendaraan dan memperbaiki kerusakan pada kendaraan tersebut. Tidak semua orang dapat membuat keputusan tentang mendiagnosis dan mengobati suatu penyakit [9].

Ide dasarnya sederhana, pengetahuan ditransfer dari seorang ahli ke komputer. Pengetahuan ini kemudian disimpan di sana atau pengguna bisa mendapatkan saran khusus. Komputer dapat mencari, memproses, dan menampilkan kesimpulan yang ditargetkan. Dan seperti seorang ahli, saran ini dapat digunakan oleh non-ahli bersama dengan penjelasan yang mencakup logika alasan di balik saran tersebut.

Sistem pakar memiliki 2 komponen utama, yaitu basis data pengetahuan dan mesin *inferensi*. Basis pengetahuan adalah tempat di mana pengetahuan disimpan dalam memori komputer, di mana pengetahuan itu diakses dari pengetahuan ahli [9].

2.2.6 Ciri-ciri Sistem Pakar

Sistem pakar memiliki ciri-ciri sebagai berikut [9] :

- 1) Fasilitas informasi
- 2) Kemampuan modifikasi
- 3) Berdasarkan pada kaidah/*rule* tertentu
- 4) Terbatas pada domain keahlian tertentu
- 5) *Transportability*
- 6) Kemampuan belajar adaptif
- 7) Dapat memberikan penalaran untuk data-data yang tidak pasti
- 8) Dirancang untuk dapat dikembangkan secara bertahap.
- 9) Outputnya bersifat nasihan atau anjuran

- 10) Outputnya tergantung dari dialog dengan *user*
- 11) *Knowledge base* dan *inference engine* terpisah

2.2.7 Jenis-Jenis Sistem Pakar

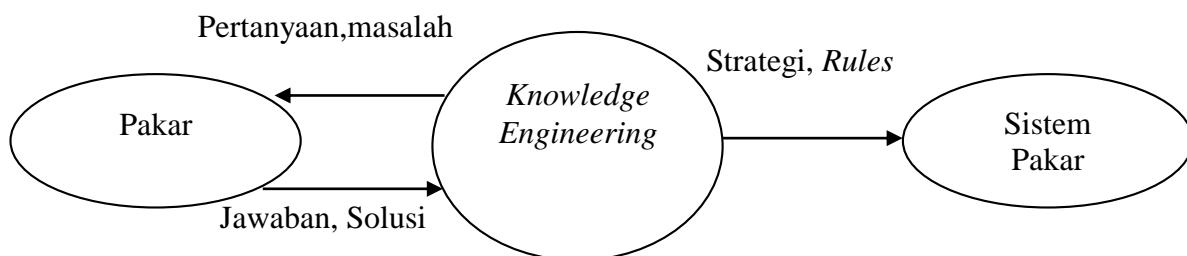
Sistem pakar muncul dalam berbagai variasi, seperti tersebut dibawah ini [8] :

- 1) Sistem pakar vs *Knowledge-based system*. Sistem pakar adalah suatu sistem yang bekerja dengan mendapatkan pengetahuan dari para pakar, sedangkan KBS dari mendapatkan pengetahuan dari sumber-sumber terdokumentasi. Oleh karena itu, KBS lebih murah dan lebih cepat dibangun dibandingkan sistem pakar.
- 2) *Rule-Based Expert System* adalah suatu Pengetahuan yang direpresentasikan sebagai serangkaian rule-rule (production rules)
- 3) *Frame-Based System* adalah suatu Pengetahuan direpresentasikan sebagai frame, yaitu representasi dari pendekatan pemrograman berbasis objek (OOP)
- 4) *Hybrid Systems* yakni sistem yang melibatkan berbagai pendekatan representasi pengetahuan, seperti frame dan rule, tapi biasanya yang digunakan lebih dari itu.
- 5) *Model-Based System* yaitu sistem yang terstruktur di sekitar model yang mensimulasikan struktur dan fungsi sistem yang sedang dipelajari. Model ini digunakan untuk menghitung nilai yang dibandingkan dengan nilai yang diamati. Perbandingan memicu tindakan atau diagnosis lebih lanjut.
- 6) Sistem diklasifikasikan menurut sifatnya yang terdiri dari 3 jenis, yaitu
 - (1) terkait dengan pengumpulan bukti, (2) sistem penyempurnaan bertahap. Sistem ini menangani sejumlah besar keluaran dari tingkat detail berikutnya. (3) perakitan bertahap, di mana domain subjek dapat memiliki jumlah kemungkinan keluaran yang luar biasa besar. Jenis sistem ini dikenal sebagai pemilihan katalog. Sistem ini menangani masalah seperti pemilihan bahan kimia yang benar, dan bahan baja.

- 7) Sistem siap pakai (*off-the shelf Systems*) adalah suatu sistem yang hasilnya dari produksi masal sehingga membuatnya lebih murah dibandingkan dengan sistem yang memenuhi keinginan user (*customized system*).
- 8) *Real-Time Expert System*. Sistem ini berkenaan dengan waktu, jadi harus cukup cepat mengontrol proses terkomputerisasi. Sistem ini menghasilkan respon sesuai waktu yang diperlukan.

2.2.8 Refresentasi Pengetahuan

Representasi pengetahuan adalah sebuah metode untuk menggabungkan pengetahuan dari sistem pakar dalam basis pengetahuan [10]. Pengetahuan dapat bersumber dari keahlian pakar, kasus-kasus, dokumen, gambar-gambar, laporan, dan data-data lainnya. Proses menyerap pengetahuan dari sumber-sumber pengetahuan dan mentransformasinya ke dalam bentuk representasi internal berdasarkan kerangka representasi pengetahuan disebut dengan akuisisi pengetahuan [9]. Berikut gambarnya:



Gambar 2. 1 Proses Akuisisi Pengetahuan

2.2.9 Faktor kepastian (*Certainty Factor*)

2.2.9.1 Ketidakpastian dengan teori *Certainty Factor* (Faktor Kepastian)

Ketika menghadapi suatu masalah, seringkali ditemukan jawaban yang tidak memiliki kepastian penuh. Ketidakpastian ini dapat berupa probabilitas atau probabilitas, yang tergantung pada hasil dari suatu peristiwa. Hasil *unsafe* disebabkan oleh faktor yaitu *unsafe rules* dan *unsafe user response* terhadap

pertanyaan yang diajukan oleh sistem. Hal ini sangat mudah dilihat dalam sistem diagnosis penyakit dimana para ahli tidak dapat dengan yakin mendefinisikan hubungan antara gejala dan penyebabnya dan pasien tidak dapat merasakan suatu gejala dengan pasti. Pada akhirnya, banyak kemungkinan diagnosis ditemukan [7].

Sistem pakar harus mampu bekerja dalam ketidakpastian. Sejumlah teori telah ditemukan untuk menyelesaikan ketidakpastian termasuk *probabilitas klasik*, *probabilitas Bayesian*, teori *Hartley* berdasarkan himpunan klasik, teori *Shannon* berdasarkan probabilitas, teori *Depmster-Shafer*, teori *fuzzy Zadeh*, dan faktor kepastian. (*Certainty Factor*) [5].

2.2.9.2 Ketidakpastian Aturan

Ada tiga penyebab ketidakpastian aturan yaitu aturan tunggal, penyelesaian konflik dan ketidakcocokan (*incompatibility*) aturan konsikuen dalam aturan. aturan yang dapat menyebabkan ketidakpastian di pengaruhi oleh tiga hal, yaitu: kesalahan, probalitas, dan kombinasi gejala (*evidence*). Kesalahan dapat terjadi karena [5] :

- a. Ambiguitas, sesuatu didefinisikan dengan lebih dari satu cara
- b. Ketidaklengkapan data
- c. Kesalahan informasi
- d. Ketidakpercayaan terhadap suatu alat

Certainty Factors (CF) menyatakan *belief* dalam suatu event (atau fakta, atau hipotesis) didasarkan kepada *evidence* atau *expert's assessment*.

- CF [P,H] = MB [P,E] – MD [P,E] Dimana:

CF = *certainty factor* MB = *measure of belief*

MD = *measure of disbelief* P = *probability* E = *evidence*, atau event

- Mengkombinasikan beberapa CF dalam satu rule:

Operator AND:

CF (A AND B AND C) = min [CF(A). CF(B). CF(C)]

Contoh:

IF inflation is high, [CF = 0,5, (A)] AND

IF unemployment rate is above 7%, [CF = 0,7, (B)],

AND IF bond prices decline [CF=0,1, (C)]

THEN stock price decline

$$\begin{aligned} \text{CF(A AND B AND C)} &= \min [\text{CF(A)}, \text{CF(B)}, \text{CF(C)}] \\ &= \min (0,5,0,7,1,0) \\ &= \min 0.5 \end{aligned}$$

Operator OR

$\text{CF(A OR B)} = \text{MAX} [\text{CF(A)}, \text{CF(B)}]$

Kombinasi dari 2 (dua) atau lebih Rules:

Dua Rules (*independent*)

$$\text{CF(R1,R2)} = \text{CF(R1)} + \text{CF(R2)} \times [1 - \text{CF(R1)}] \text{ atau}$$

$$\text{CF(R1,R2)} = \text{CF(R1)} + \text{CF(R2)} - \text{CF(R1)} \times \text{CF(R2)}$$

Dua Rules (*independent*)

$$\text{CF(R1,R2)} = \text{CF(R1)} \times \text{CF(R2)}$$

Tiga Rules (*independent*):

$$\text{CF(R1,R2,R3)} = \text{CF(R1,R2)} + \text{CF(R3)} \times [1 - \text{CF(R1,R2)}]$$

Contoh:

R1 : IF the inflation rate is less than 5 %

THEN stock market prices go up (CF = 0,7)

R2 : IF unemployment level is less than 7 %

THEN stock market prices go up (CF = 0,6)

R3 : IF bond prices increases,

THEN stock prices go up (CF = 0,85)

Hitung : -CF(R1,R2) dan

- CF(R1.R2.R3)

- Solusi:

Dua Rules (*independent*)

$$\text{CF(R1,R2)} = \text{CF(R1)} = \text{CF(R2)} \times [1 - \text{CF(R1)}]$$

$$= 0.7 + 0.6 \times (1-0.7)$$

$$= 0.7 + 0.18$$

$$= 0.88$$

TIGA RULES (*independent*)

$$\begin{aligned}
 CF(R1,R2,R3) &= CF(R1,R2) + CF(R3) \times [1 - CF(R1,R2)] \\
 &= 0.88 + 0.85 \times (1 - 0.88) \\
 &= 0.88 + 0.85 \times 0.12 \\
 &= 0.88 + 0.102 \\
 &= 0.982
 \end{aligned}$$

2.2.9.3 Pengertian Faktor Kepastian (*Certainty Factor*)

Faktor kepastian (CF) menunjukkan jaringan keyakinan tentang hipotesis berdasarkan beberapa fakta atau gejala di bidang medis. CF positif berarti fakta yang mendukung hipotesis, karena $MB > MD$. $CF = 1$ berarti fakta, menurut definisi, membuktikan hipotesis. $CF = 0$ berarti salah satu dari dua kemungkinan, yaitu pertama $CF = MB-MD = 0$ baik MB maupun MD sama dengan nol, yang berarti tidak ada fakta. Kemungkinan kedua adalah bahwa $MB = MD$ dan keduanya bukan nol, yang berarti ketidakpastian menghilangkan atau menghilangkannya. CF negatif berarti fakta menunjukkan negasi hipotesis karena $MB < MD$. Dengan kata lain, lebih masuk akal untuk tidak mempercayai hipotesis dari pada mempercayainya [5].

Faktor kepastian (*certainty factory*) Diperkenalkan oleh *Shortliffe Buchanan* dalam pembuatan MYCIN. *Confidence Factor* (CF) adalah nilai parametrik klinis yang diberikan oleh MYCIN untuk menunjukkan tingkat kepercayaan *Certainty Factory* (CF) menunjukkan ukuran kepastian terhadap suatu fakta atau aturan. *Certainty Factory* (CF) didefinisikan sebagai berikut :

$$CF(H,E) = MB(H,E) - MD(H,E)$$

Dengan :

$CF(H,E)$: *certainty factory* dari hipotesis H yang dipengaruhi oleh gejala (*evidence*) E. Besarnya CF berkisar antara -1 sampai dengan 1. Nilai -1 menunjukkan ketidakpercayaan mutlak sedangkan nilai 1 menunjukkan kepercayaan mutlak.

$MB(H,E)$: ukuran kenaikan kepercayaan (*measure of increased belief*) terhadap hipotesis H yang dipengaruhi oleh gejala E.

MD(H,E) : ukuran kenaikan ketidakpercayaan (*measure of increased disbelief*) terhadap hipotesis H yang dipengaruhi oleh gejala E

Cara perhitungan manual untuk mendapatkan nilai *certainty factory*, yaitu sebagai berikut.

Pertama-tama, mereka akan memilih usia anak terlebih dahulu, lalu menjawab salah satu gejala yang dialami pasien, misalnya yang pertama dipilih seperti menangis atau tertawa tiba-tiba tanpa malu, menolak untuk dipeluk, atau sedikit tertunda dalam pembangunan, untuk memimpin dengan contoh dalam berjalan. Dengan nilai kepercayaan menunjukkan tebelatur.

Rumus Awal :

IF E THEN H

Adalah :

$CF(H,e)$

$CF(E,e) CF(H,E)$

Dimana:

$CF(E,e)$: faktor kepastian dari fakta E membuat

antecedent dari kaidah berdasarkan pada

$Ecf(H,E)$: faktor kepastian dalam hipotesa dengan asumsi bahwa fakta di ketahui dengan

Pasti, bila $CF(E,e) = 1$

$CF(H,e)$: faktor kepastian hipotesis yang didasarkan pada ketidakpastian fakta e.

Jika semua fakta dalam *antecedent* di ketahui dengan pasti rumus faktor kepastian menjadi:

$CF(H,E) = CF(E,e)$, karena $CF(E,e) = 1$ penalaran pengantar kecerdasan Buatan (AK045218)

Contoh : kaidah *streptococcus* (bakteri)

IF

1. Zat dari organisme adalah gram positif AND
2. Morfologi dari organisme adalah *coccus* AND
3. Penyesuaian diri dari organisme adalah merantai

THEN

Ada bukti suggesstif (0.7)

Bahwa indentifikasi dari organisme tersebut adalah *streptococcus*

Dimana faktor kepastian dari hipotesis dengan kepastian fakta adalah

$$CF(H,E) = CF(H,E1)$$

$$\cap E2$$

$$\cap E3 = 0.7$$

Dan disebut *Attenuation factor Attenuation factor* dadasarkan pada asumsi bahwa semua fakta E1,E2 dan E3 diketahui dengan pasti,yaitu :

$CF(E1,e) = CF(E2,e) = CF(E3,e) = 1$ jika diasumsikan : $CF(E1,e) = 0.5$

$CF(E2,e) = 0.6$ $CF(E3,e) = 0.3$ Maka $CF(E,e) = CF(E1 \cap E2 \cap E3,e) = 0.7 = \text{Min}[CF(E1,e),e]$

$CF(E3,e) = \min[0.5;0.6;0.3] = 0.3$ $CF(H,e) = (0.3).(0.7) = 0.21$ karena CF dari *antecedent* $CF(E,e) > 0.2$; *antecedent* diyatakan benar dan kaidah diaktifkan.

Setiap perhitungan dibandingkan dengan nilai masing-masing jenis penyakit sehingga masing-masing penyakit dengan nilai CF yang besar dapat dibandingkan sehingga dapat diperoleh tingkat kepastiannya.(sumber: MATERI KULIAH. Html artikel diakses 12 Mei 2013)

Keterkaitan rumus dan proses diajukan (kelebihan metode *certainty factory* (CF). Aturan metode *certainty factory* (CF):

1. McAllister menggambarkan aturan untuk menambahkan dua faktor *Certainty* positif adalah:

$$(CFaCFb) = CFa + CFb * (1-Cfa)$$

2. Aturan untuk menambahkan dua *certainty factory* yang negatif adalah:

$$(CFcCFd) = CFc + CFd + CFc * CFd$$

3. Aturan untuk menambahkan *certainty factory* positif dan *certainty factory* negatif lebih kompleks :

$$(CFeCFf) = \frac{CFe + CFf}{1 - \min\{|CFe|, |CFf|\}}$$

Kelebihan metode *certainty factory* (CF) adalah:

1. Metode ini cocok digunakan pada sistem pakar untuk mengukur apakah ada sesuatu yang pasti atau tidak pasti dalam diagnosis suatu penyakit sebagai salah satu contohnya.
2. Akun yang menggunakan cara ini dalam satu akun hanya dapat memproses 2 data sehingga keakuratan data dapat terjaga.

Kekurangan metode *Certainty Factor* adalah:

1. Gagasan umum pemodelan ketidakpastian manusia menggunakan metode faktor kepastian numerik biasanya dibahas. Beberapa orang mungkin berpendapat bahwa rumus untuk metode faktor kepastian yang disebutkan di atas memiliki sedikit kebenaran.

Metode ini hanya mengelola ketidakpastian/kepastian hanya 2 data saja. Perlu beberapa kali pengolahan data yang lebih baik dari 2 buah

2.2.9.4 Penerapan Metode *Certainty Factor*

Dalam jurnal M.Arifin, Slamin dan Windi E.J.R tahun 2017 yang meneliti tentang Penerapan Metode *Certainty Factor* pada Identifikasi Penyakit Tanaman Tembakau diperoleh bahwa, terdapat empat gejala dengan masing-masing nilai berat serangan yaitu gejala 1 : daun mendadak layu seperti disiram air panas yakni dengan persentase sebanyak 98%, gejala 2 : Bercak daun cokelat kehitaman agak basah dengan persentase sebanyak 65%, gejala 3 : bercak berwarna hijau kelabu kotor dengan persentase sebanyak 80%, dan gejala 4 : Jika pangkal batang dibelah, empulur mengering dengan persentase sebanyak 80%. Hasil dari penelitian tersebut dapat dilihat pada tabel berikut.

Tabel 2. 2 Input gejala oleh user atau pengguna

Kode Gejala	Nama Gejala	Input User (%)
L01	Daun yang mendadak layu seperti disiram air panas	98
L02	Bercak daun berwarna cokelat kehitaman agak basah	65
L03	Bercakberwarna hijau kelabu dan kotor	80
L04	Jika pangkal batang dibelah maka empulur mongering	80

Selanjutnya peneliti melakukan perhitungan dengan menggunakan metode *Certainty Factor* dengan menggabungkan *User CF* yang diperoleh dengan *Expert CF* yang merupakan nilai yang diperoleh dari kepercayaan para ahli untuk masing-masing basis. Sistem akan membagi nilai bobot yang diberikan oleh pengguna dengan angka 100 karena penggunaan metode faktor kepastian, rentang bobot yang dapat ditentukan adalah dari 0 hingga 1.

CF gabungan diperoleh dengan mengalikan CF yang digunakan dan CF *Expert*. Untuk gejala yang tidak ditentukan, sistem akan secara otomatis menetapkan nilai 0.

Tabel 2. 3 Perhitungan CF kombinasi

Kode Gejala	Nama Variabel Gejala	Penyakit	Nilai CF User	Nilai CF Pakar	CF Kombinasi
L01	Daun mendadak layu seperti disiram air panas	Lanas	0.98	0.2	0.196
L02	Bercak daun cokelat kehitaman agak basah		0.65	0.6	0.39
L03	Bercak berwarna hijau kelabu kotor		0.8	0.8	0.64

Kode Gejala	Nama Variabel Gejala	Penyakit	Nilai CF User	Nilai CF Pakar	CF Kombinasi
L04	Jika pangkal batang dibelah, empulur mongering		0.8	1	0.8
B05	Daun layu sebagian	Layu	0.3	0.2	0.06
B06	Tulang dan tepi daun menguning	Bakteri	0	0.4	0
B07	Daun mengering dan menjadi seperti selaput		0	0.6	0
B08	Terjadi pembusukan pada pangkal batang		0	0.8	0
B09	Keluar lendir putih kotor bila pangkal batang dipotong		0	1	0
K10	Daun muda tampak berkerut dan mengeriting	Keriting	0	0.2	0
K11	Tepi daun melengkung keatas		0	0.6	0
K12	Tulang daun bengkok dan menebal		0	0.8	0
K13	Daun tua tambak berkerut dan mengeriting		0	1	0
M14	Bercak berwarna kuning	Mosaik	0	0.2	0
M15	Daun mempunyai gambaran mosaik (belang) terutama daun muda		0	0.4	0
M16	Tulang daun lebih jernih dan ada gambaran mosaic		0	0.6	0

Kode Gejala	Nama Variabel Gejala	Penyakit	Nilai CF User	Nilai CF Pakar	CF Kombinasi
M17	Pertumbuhan terhambat dan ada gambaran mosaik		0	1	0

Berdasarkan nilai CF Kombinasi pada tabel 2, ditentukan nilai CF gabungan dari setiap rules dengan menggunakan rumus sebagai berikut.

$$\text{CFgabungan} (\text{CF1}, \text{CF2}) = \text{CF1} + \text{CF2} - (\text{CF1} * \text{CF2})$$

CF Gabungan untuk Penyakit Lanas:

1. CFgabungan (L01, L02) = $0,196 + 0,39 - (0,196 * 0,39) = 0,586 - 0,07644 = \mathbf{0,50956}$
2. CF Gabungan (old, L03) = $0,50956 + 0,64 - (0,50956 * 0,64) = 1,14956 - 0,3261184 = \mathbf{0,8234416}$
3. CF Gabungan (old, L04) = $0,8234416 + 0,8 - (0,8234416 * 0,8) = 1,6234416 - 0,65875 = \mathbf{0,96468}$

Selanjutnya untuk mendapatkan nilai persentase penentuan penyakit tanaman tembakau dihitung senyawa $\text{CF} * 100\%$. Sehingga sistem akan memberikan informasi persentase identifikasi penyakit laring sebesar 96,46%.

2.2.10 Siklus Hidup Pengembangan Sistem

Analisa sistem (*System Analisa*) Dapat diartikan sebagai penguraian suatu sistem informasi yang utuh menjadi bagian-bagian komponennya dengan tujuan untuk mengidentifikasi dan mengevaluasi masalah dan hambatan yang terjadi serta kebutuhan yang diharapkan sehingga dapat disarankan perbaikannya [11].

Analisis sistem adalah spesialis yang mempelajari masalah organisasi dan perlu menentukan bagaimana orang, data, proses, dan teknologi informasi dapat memajukan bisnis dengan baik.

Analisis sistem adalah *Stakeholder* yang berperan sebagai fasilitator atau pelatih, menjebatani jurang komunikasi yang dapat secara alamiah berkembang antara pemilik dan pengguna *system nonteknis* atau desainer dan perkembangan sistem teknis.

Whitten, et al. [12] mengungkapkan “*System analysis* adalah *study domain* masalah bisnis untuk merekomendasikan perbaikan dan menspesifikasi persyaratan dan prioritas bisnis untuk solusi”.

Impak teknologi objek sangat berarti dalam dunia analisis dan desain sistem. Sebelum ada teknologi objek, kebanyakan bahasa pemrograman didasarkan pada apa yang disebut metode yang terstruktur (*structured method*). Contohnya COBOL bahasa yang domain 0, C, Fortan, Pascal, dan PL/i. Maka, metode analisis dan desain berorientasi objek telah muncul sebagai pendekatan terpilih untuk membangun kebanyakan sistem informasi saat ini.

Sebagai tambahan keahlian analisis dan desain sistem formal, seorang analis harus mengembangkan atau memiliki keahlian lain, pengetahuan, dan karakter untuk menyelesaikan pekerjaan. Hal ini termasuk:

1. Pengalaman dan keahlian pemrograman komputer.

Sulit membayangkan bagaimana analis sistem dapat secara memadai mempersiapkan spesifikasi bisnis dan teknis untuk pemrogram jika mereka tidak memiliki pengalaman pemrograman. Sebagian besar analis sistem harus mahir dalam satu atau lebih bahasa pemrograman tingkat tinggi.

2. Pengetahuan umum proses dan teknologi bisnis.

Analis sistem harus dapat berkomunikasi dengan para ahli untuk memahami masalah dan kebutuhan. Untuk analisis, setidaknya sebagian dari pengetahuan ini hanya berasal dari pengalaman. Pada saat yang sama, analis yang menginspirasi harus memanfaatkan setiap kesempatan untuk menyelesaikan kursus teori bisnis dasar.

Tahap analisis merupakan tahap yang kritis dan sangat penting, karena setiap kesalahan yang terjadi pada tahap ini akan menimbulkan kesalahan pada tahap selanjutnya. Tahap analisis sistem ini meliputi studi kelayakan analisis kebutuhan.

a. Studi Kelayakan.

Studi kelayakan digunakan untuk menentukan probabilitas keberhasilan solusi yang diusulkan selama penelitian. Tahap ini berguna untuk memastikan bahwa solusi yang diusulkan benar-benar dapat dicapai dengan sumber daya dan dengan mempertimbangkan kendala yang ada di perusahaan dan dampaknya terhadap lingkungan sekitar. Tugas-tugas yang tercakup dalam studi kelayakan meliputi:

1. Penentuan masalah dan peluang yang ditangani oleh sistem.
2. Penetapan tujuan sistem baru secara keseluruhan.
3. Identifikasi pengguna sistem.
4. Penentuan ruang lingkup sistem.

Selain itu, selama dalam tahapan ini, sistem analisis juga melakukan tugas-tugas sebagai berikut:

1. Usulan perangkat lunak dan perangkat keras untuk sistem baru.
2. Pembuatan analisis untuk membuat atau membeli aplikasi..
3. Membuat analisis biaya/manfaat.
4. Penilaian risiko proyek..

Menurut Kadir Abdul tahun 2003 dalam bukunya menyatakan Studi kelayakan ini diukur dengan memperhatikan aspek teknologi, ekonomi, faktor organisasi dan kendala hukum, etika, dan yang lain [13].

b. Analisis kebutuhan.

Analisis kebutuhan dilakukan untuk menghasilkan spesifikasi kebutuhan. Spesifikasi kebutuhan adalah spesifikasi rinci tentang apa yang akan dilakukan sistem saat dijalankan selama pencarian. Spesifikasi ini juga digunakan untuk membuat kesepakatan antara pengembang sistem, mitra (seperti *auditor internal*) dan pengguna yang akan menggunakan sistem manajemen di masa mendatang.

Analisis kebutuhan ini diperlukan untuk menentukan keluaran yang akan dihasilkan sistem, lingkup proses yang digunakan untuk mengolah masukan menjadi keluaran,masukan yang diperlukan sistem,

volume data yang akan ditangani sistem, jumlah pemakai dan kategori pemakai, serta kontrol terhadap sistem.

Menurut Kadir Abdul pada tahun 2003 menyatakan bahwa dalam tahap analisis sistem ini terdapat langkah-langkah dasar yang harus dilakukan oleh analisis sistem, yaitu sebagai berikut [13]:

1. *Identify*, yaitu mengidentifikasi masalah.

Ini adalah langkah pertama dalam tahap analisis sistem. Masalah (*problems*) dapat diartikan sebagai suatu pertanyaan yang ingin dipecahkan. Pada tahap identifikasi, sebuah pertanyaan ingin dipecahkan. Tahap identifikasi masalah ini sangat penting karena akan menentukan keberhasilan pada langkah-langkah selanjutnya.

2. *Understand*, yaitu memahami kerja dari sistem yang ada.

Langkah kedua dari tahap analisis sistem adalah memahami cara kerja sistem yang ada. Langkah ini dapat dilakukan dengan mempelajari operasi sistem yang dapat diperoleh dengan melakukan penelitian.

3. *Analyze*, yaitu menganalisis sistem tanpa report.

Langkah ini dilakukan berdasarkan data yang telah diperoleh dari hasil penelitian yang telah dilakukan.

4. *Report*, yaitu membuat laporan hasil analisis.

Tujuan utama pembuatan laporan hasil yang dilakukan yaitu;

- a. Melaporkan bahwa analisis telah selesai.
- b. Memperbaiki kesalahpahaman tentang apa yang telah ditemukan dan dianalisis oleh analis sistem tetapi tidak sesuai menurut manajemen

2.2.11 Desain Sistem

Setelah tahap analisis sistem selesai, analisis sistem memiliki gambaran yang jelas tentang apa yang perlu dilakukan. Pada saat ini analisis sistem mempertimbangkan bagaimana sistem dikonfigurasi atau biasa disebut dengan desain sistem (*system design*). Dalam perancangan sistem diperlukan alat bantu.

Salah satu *tools* yang dapat digunakan dalam pembuatan sistem ini adalah *Unified Modeling Language* (UML). [14].

Menurut Whitten & Bentley (2007:371), *Unified Modeling Language* (UML) adalah sebuah bahasa pemodelan yang digunakan untuk mendefinisikan atau menggambarkan sistem perangkat lunak berdasarkan elemen-elemen dalam sistem. Alat UML ini tidak menentukan metode mana yang harus digunakan dalam pengembangan sistem, tetapi hanya mendefinisikan simbol standar yang biasa digunakan untuk pemodelan objek pada sistem. [14].

Dengan menggunakan UML, kita dapat membuat model untuk semua jenis aplikasi perangkat lunak, aplikasi ini dapat berjalan di perangkat, sistem operasi, dan jaringan apa pun, dan ditulis dalam bahasa pemrograman apa pun. Tetapi karena UML juga menggunakan kelas dan operasi dalam konsep intinya, maka lebih cocok untuk menulis program dalam bahasa berorientasi objek seperti C++, Java, C#, atau VB.NET. Namun, UML masih dapat digunakan untuk memodelkan aplikasi prosedural di VB atau C [15].

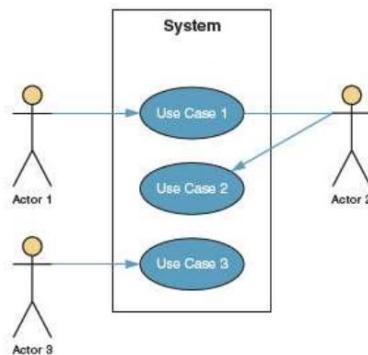
2.2.11.1 Use Case Diagram

Menurut Whitten & Bentley (2007:246) *Use-case diagram* adalah Diagram yang menggambarkan interaksi sistem dengan bagian luar sistem dan dengan pengguna. Secara grafis, *use case* diagram ini menggambarkan siapa yang akan menggunakan sistem dalam penelitian dan bagaimana pengguna berharap untuk berinteraksi dengan sistem saat menganalisis data. [14].

Whitten & Bentley (2007) mengemukakan bahwa *Use case* diagram memiliki unsur yang harus dipenuhi,yakni.

- a. *Use Cases*, adalah Seperangkat fungsi yang terdapat dalam sistem. Di mana aktor (*user*) dapat melakukan fungsi-fungsi ini. Untuk melakukan tugasnya dengan sistem saat ini pada saat analisis data.
- b. *Actors*, yaitu Segala sesuatu yang berinteraksi dengan sistem untuk berbagi informasi, baik pengguna maupun sistem dari luar negeri..

c. *Relationships*, yaitu garis yang menghubungkan subjek dengan *use case* yang menunjukkan hubungan antara subjek dan *use case* itu sendiri.



Gambar 2. 2 *Use Case Diagram*: Whitten & Bentley (2007:246) [14].

Tabel 2. 4 Notasi *Use Case Diagram*

Nama Komponen	Keterangan	Simbol
<i>Use Case</i>	<i>Use case</i> dapat digambarkan sebagai lingkaran <i>elips</i> , dengan nama <i>use case</i> dituliskan dalam lingkaran <i>elips</i> .	
<i>Actor</i>	Aktor adalah pengguna sistem dimana aktor tersebut tidak terbatas pada orang saja, tetapi ketika sebuah sistem berkomunikasi dengan aplikasi lain dan membutuhkan <i>input</i> atau <i>output</i> , aplikasi juga dianggap sebagai aktor.	
<i>Association</i>	Asosiasi sering digunakan untuk menghubungkan aktor dengan kasus penggunaan. Asosiasi ini diwakili oleh garis lurus yang menghubungkan aktor <i>use case</i> dan <i>use case</i> .	<hr/>

(Sumber: Whitten & Bentley, 2007:246) [14].

2.2.11.2 *Class Diagram*

Menurut Whitten & Bentley (2007:382), *class diagram* adalah diagram yang menggambarkan struktur objek dari sistem yang ada, dimana *class diagram*

ini memperlihatkan *object class* yang menyusun diagram ini beserta hubungan antara *object class* tersebut [14].

Menurut Whitten & Bentley (2007:400-405), Terdapat beberapa tahapan dalam pembentukan *class diagram*, antara lain [14]:

1. Identifikasi asosiasi dan keragaman kelas yang ada Obyek.

Dalam fase ini, asosiasi kelas yang ada diidentifikasi. Objek. Klub yang disebutkan di sini adalah tentang informasi apa yang harus diketahui antara suatu objek dan objek aneka ragam.

2. Mengidentifikasi hubungan yang general dan hubungan khusus atas *class*.

Setelah mengetahui asosiasi dan keragaman kelas yang ada, perlu diketahui apakah hubungan antar kelas meliputi: hubungan umum atau hubungan khusus. Hubungan umum atau khusus yang dimaksud di sini adalah pengklasifikasian suatu hierarki, yaitu hubungan berdasarkan *supertype class (abstract/parent)* dan *subtype class (concrete/child)*.

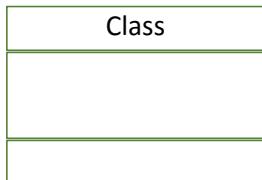
3. Mengidentifikasi hubungan agregasi / komposisi dari suatu *class*.

Pada tahap ini perlu ditentukan apakah terdapat hubungan agregasi/komposisi antar kelas. Hubungan agregasi yang dimaksud adalah jenis hubungan unik dari suatu objek yang merupakan bagian dari objek tertentu.

4. Menyiapkan *class diagram* itu sendiri.

Pada fase ini, diagram kelas dibuat berdasarkan informasi tentang hubungan antar kelas yang ada, baik hubungan asosiasi, hubungan umum/khusus, maupun hubungan agregasi yang terjadi antar kelas tersebut.

Tabel 2. 5 Notasi Class Diagram

Symbol	Penjelasan
	Class: Yaitu deskripsi objek yang terbagi menjadi 3 bagian, yaitu nama kelas di bagian atas, atribut di tengah dan operasi di bagian bawah.
	Aggregation: Suatu bentuk khusus dari hubungan asosiasi yang memiliki hubungan khusus antara kelompok dan bagian. Koleksi ini diwakili oleh berlian yang tidak terisi.
	Association: Yaitu, menggambarkan hubungan terstruktur antara kelas terkait.
	Generalization: Artinya, hubungan yang memperhatikan suatu kelas bisa lebih umum atau khusus dari kelas lainnya.
<i>Contains</i>  0..* 1	Multiplicity: Yaitu, menggambarkan jumlah objek yang terlibat dalam hubungan antar kelas.

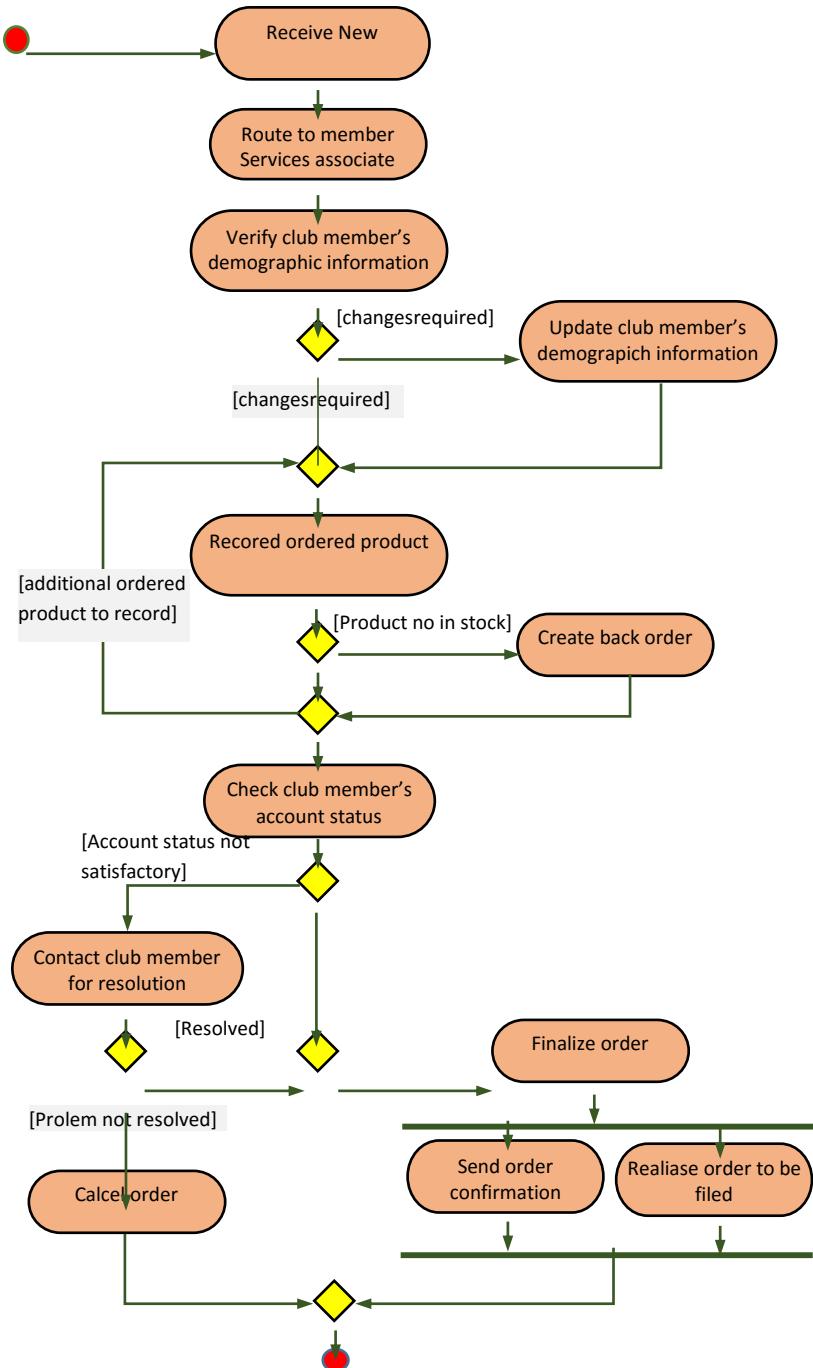
(Sumber: Whitten & Bentley, 2007:406) [14]

2.2.11.3 Activity Diagram

Menurut Whitten & Bentley (2007:394), *activity diagram* adalah sebuah diagram yang dapat digunakan untuk menggambarkan secara grafis alur dari sebuah proses bisnis, langkah-langkah dari sebuah *use-case* yang ada, atau logika

dari sebuah objek. *Activity diagram* sangat berguna untuk model *action* yang akan dikerjakan ketika sebuah operasi dieksekusi serta hasil dari *action* tersebut terjadi. [14].

Tidak semua *use-case* harus digambarkan dalam sebuah *activity diagram*. *Activity diagram* biasanya digunakan untuk *use-case* yang memiliki logika yang cukup kompleks sehingga *activity diagram* ini dapat membantu untuk berpikir tentang logika dari sebuah sistem yang ada.



Gambar 2. 3 Activity Diagram: (Whitten & Bentley, 2007:392) [14].

Tabel 2. 6 Diagram Activity Notasi

Komponen	Simbol	Penjelasan
<i>Initial Node</i>		Ini adalah awal dari proses.
<i>Action</i>		Ini adalah langkah-langkah individu yang membentuk aktivitas total yang ditunjukkan dalam diagram.
<i>Flow</i>		Tampilkan kemajuan tindakan.
<i>Decission</i>		Menunjukkan aktivitas pilihan yang mengarah pada keputusan.
<i>Fork</i>		Menunjukkan tindakan yang terjadi pada waktu yang sama.
<i>Join</i>		Menunjukkan akhir dan penggabungan proses bersamaan.
<i>Activity Final</i>		Ini adalah akhir dari proses.

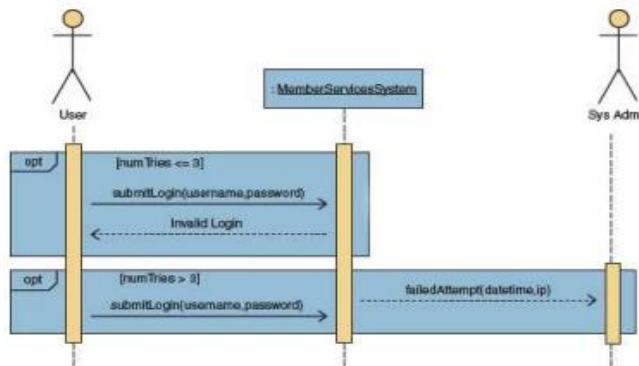
(Sumber: Whitten & Bentley, 2007:392) [14].

2.2.11.4 Sequence Diagram

Menurut Whitten & Bentley (2007:394), *sequence diagram* adalah sebuah diagram yang menggambarkan interaksi antara *actor* dan *system* untuk sebuah skenario *use-case*. Pada tahap pembuatan *sequence diagram*, kita belum

menganalisa lebih lanjut *individual object class*, namun hanya memikirkan keseluruhan sistem yang ada [14].

Sequence diagram membantu kita untuk mengidentifikasi setiap data yang masuk dan keluar dari sebuah sistem. Pada *sequence diagram* hanya sebuah skenario dari sebuah *use-case*, sehingga sebuah *use-case* dapat memiliki beberapa *sequence diagram* untuk menggambarkan keseluruhan *use-case* tersebut [14]



Gambar 2. 4 Sequence Diagram: (Whitten & Bentley, 2007:396) [14]

Tabel 2. 7 Notasi Diagram Sequence

Simbol	Nama	Keterangan
	<i>Object Lifeline</i>	Menyatakan kehidupan suatu objek.
	<i>Actor</i>	Orang atau divisi yang terlibat dalam suatu sistem.
	<i>Message</i>	Menyatakan arah tujuan antara <i>Object Lifeline</i> .
	<i>Message (return)</i>	Menyatakan arah kembali dalam 1 <i>Object Lifeline</i> .
	<i>Message (return)</i>	Menyatakan arah kembali antara <i>Object Lifeline</i> .
	<i>Activation</i>	Menyatakan objek dalam keadaan aktif dan berinteraksi.

(Sumber: Whitten & Bentley, 2007:396) [14]

2.2.12 Teknik Pengujian Sistem

Pada pendekatan berorientasi objek, pengujian merupakan suatu persoalanyang lebih kompleks dibanding dengan pendekatan konvensional, karena keberadaan pewarisan, *polymorphism*, dan pengkapsulan pada pengembangan sistem berorientasi objek menimbulkan suatu persoalan yang baru untuk perancangan kasus pengujian dan analisis hasil.

Hariyanto [16] mengungkapkan bahwa: fitur-fitur berikut berpengaruh dalam teknik-teknik pengujian yang perlu dilakukan:

- Pengkapsulan (*encapsulation*)
- Penyusunan objek-objek (*object composition*)
- Pewarisan (*inheritance*)
- Interaksi (*interaction*)
- *Polymorphism*
- Pengikatan dinamis (*dynamic binding*)
- Guna ulang (*reuse*)
- *Genericity* dan kelas abstrak

Dari kompleksnya fitur-fitur yang mempengaruhi dalam pengujian sistemberorientasi objek maka strategi pengujian dilakukan pada:

1. Pengujian unit, dimana pengujian unit dilakukan hingga beberapa level dengan alasan adanya konsep pewarisan. Pengujian unit ini bertujuan untuk menjamin setiap unit memenuhi spesifikasi. Kelas-kelas merupakan sasaran pengujian unit.
2. Pengujian integrasi, pengujian ini dilakukan untuk memverifikasi implementasi dari satu use case yang telah bekerja seperti yang diharapkan. Pengujian validitas, pengujian ini dilakukan untuk menjamin fungsi-fungsi sistem/aplikasi telah dilakukan secara benar, pengujian di eksekusi ketika satu sistem (*subsistem*) yang lengkap telah di rakit. Pengujian validasi ini meliputi rincian-rincian objek yang tidak tampak, fokus pada masukan dan keluaran yang tampak oleh pemakai.

2.2.12.1 WhiteBox

White Box Testing atau pengujian *glass box* adalah metode desain *test case* menggunakan struktur kontrol desain prosedural untuk mendapatkan *test case*. Dengan menggunakan metode *White Box* analisis sistem akan memperoleh Test Case yang [17] :

- a) Menjamin seluruh *Independent Path* di dalam modul yang dikerjakan sekurang-kurangnya sekali.
- b) Mengerjakan seluruh keputusan logical
- c) Mengerjakan seluruh *loop* yang sesuai dengan batasannya
- d) Mengerjakan seluruh struktur data internal yang menjamin validitas [17].

Untuk melakukan proses pengujian *Test Case* terlebih dahulu dilakukan penerjemahan *flowchart* kedalam notasi *flowgraph* (aliran kontrol). Ada beberapa cara istilah saat pembuatan *flowgraph*, yaitu:

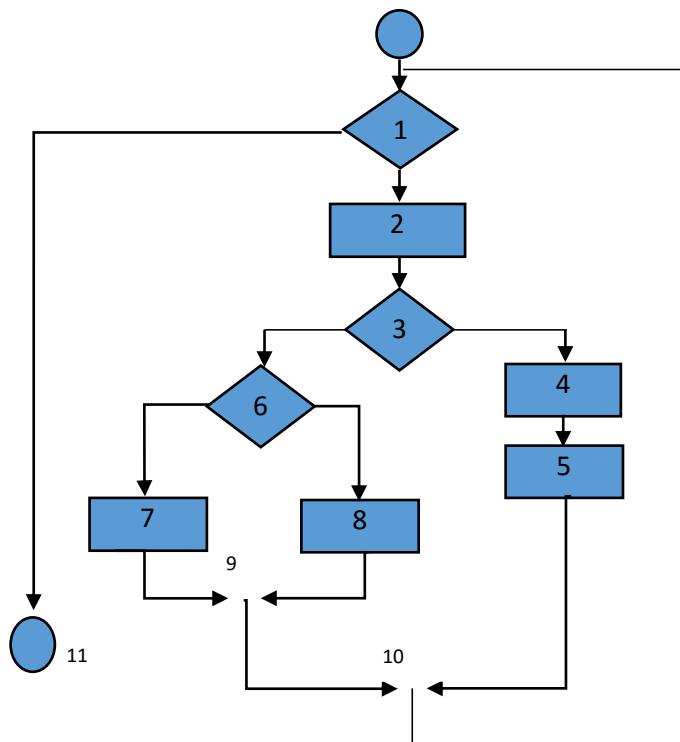
1. *Node* yaitu lingkaran pada *flowgraph* yang menggambarkan satu atau lebih perintah prosedural.
2. *Edge* yaitu tanda panah yang menggambarkan aliran kontrol dari setiap *node* harus mempunyai tujuan *node*.
3. *Region* yaitu daerah yang dibatasi oleh *node* dan *edge* dan untuk menghitung daerah diluar *flowgraph* juga harus dihitung.
4. *Predicate Node* yaitu kondisi yang terdapat pada *node* dan mempunyai karakteristik dua atau lebih *edge* lainnya.
5. *Cyclomatic Complexity* yaitu metrik perangkat lunak yang menyediakan ukuran kuantitatif dari kekompleksan logikal program dan dapat digunakan untuk mencari jumlah path dalam suatu *flowgraph*.
6. *Independen Path* yaitu jalur melintasi atau melalui program dimana sekurang-kurangnya terdapat proses perintah yang baru atau kondisi yang baru [17].

Rumus-rumus untuk menghitung jumlah *Independen Path* dalam suatu *flowgraph* yaitu:

1. Jumlah *regionflowrgaph* mempunyai hubungan dengan *Cyclomatic Complexity (CC)*.
2. $V(G)$ untuk *flowgraph* dapat dihitung dengan rumus :
 - a) $V(G) = E - N + 2$
Dimana :
 E = Jumlah *edge* pada *flowrgaph*
 N = Jumlah *node* pada *flowrgaph*
 - b) $V(G) = P + 1$
Dimana :
 P = Jumlah *predicate node* pada *flowrgaph*

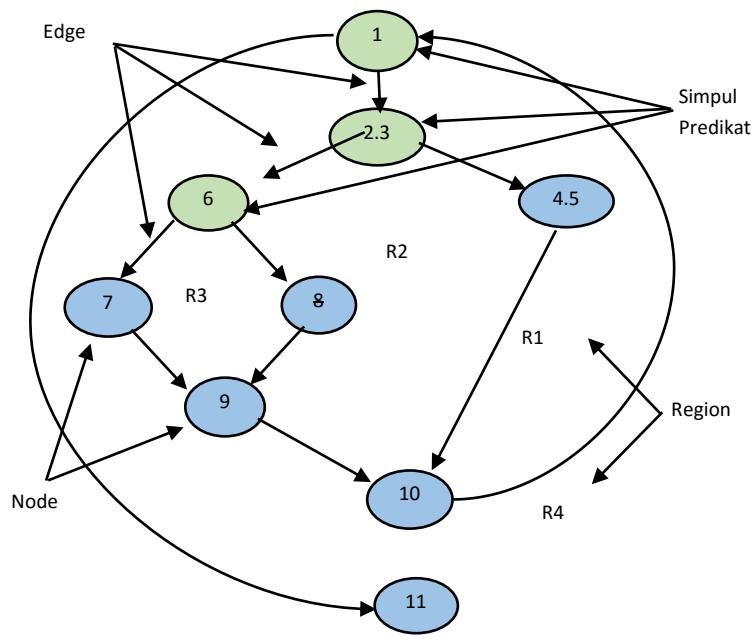
Teknik pelaksanaan pengujian *White Box* ini mempunyai tiga langkah yaitu:

1. Menggambar *flowgraph* yang ditransfer oleh flowchart
2. Menghitung *Cylomatic Complexity* untuk *flowgraph* yang telah dibuat
3. Menentukan jalur pengujian dari *flowgraph* yang berjumlah sesuai dengan *Cyclomatic Complexity* yang telah ditentukan.



Gambar 2. 5 Bagan Air: Pressman.Roger S.

Diagram alir di atas digunakan untuk menggambarkan struktur kontrol program dan juga digunakan untuk menggambarkan diagram alir yang harus diperhatikan representasi desain prosedural pada diagram alir. Pada gambar berikut, *flowchart* memetakan *flowchart* ke *flowchart* yang sesuai (dengan asumsi tidak mengandung kondisi yang dikombinasikan dengan dialog keputusan diagram alur). Setiap lingkaran, yang disebut simpul grafik aliran, dapat mewakili satu atau lebih pernyataan prosedural. Urutan kisi proses dan permata keputusan ini dapat dipetakan ke satu titik, dan panah ini, yang disebut tepi atau tautan, dapat mewakili aliran kontrol dan mirip dengan panah dalam diagram alir. Tepi harus berhenti di simpul bahkan jika simpul tidak mewakili pernyataan prosedural [17].



Gambar 2. 6 *Flowgraph*:Roger S. Pressman

Dari gambar *flowgraph* di atas didapat:

Path 1 = 1 – 11

Path 2 = 1 – 2 – 3 – 4 – 5 – 10 – 1 – 11

Path 3 = 1 – 2 – 3 – 6 – 8 – 9 – 10 – 1 – 11

Path 4 = 1 – 2 – 3 – 6 – 7 – 9 – 10 – 1 – 11

Path 1,2,3,4 yang telah didefinisikan diatas merupakan basis set untuk diagram alir.

Cyclomatic complexity digunakan untuk mencari jumlah path dalam satu flowgraph. Dapat dipergunakan rumusan sebagai berikut:

1. Jumlah region grafik alir sesuai dengan *cyclomatic complexity*.
 2. *Cyclomatic complexity* $V(G)$ untuk grafik alir dihitung dengan rumus:

$$V(G) = E - N + 2 \dots\dots\dots (1)$$

Dimana:

E = jumlah edge pada grafik alir

N = jumlah node pada grafik alir

3. Cyclomatic complexity $V(G)$ juga dapat dihitung dengan rumus:

$$V(G) = P + 1 \dots \dots (2)$$

Dimana $P = \text{jumlah } predicate node \text{ pada grafik alir}$

Dari Gambar di atas dapat dihitung *cyclomatic complexity*:

1. Flowgraph mempunyai 4 region
 2. $V(G) = 11 \text{ edge} - 9\text{node} + 2 = 4$
 3. $V(G) = 3 \text{ predicate node} + 1 = 4$

Jadi cyclomatic complexity untuk flowgraph adalah 4

Cyclomatic Complexity yang tinggi menunjukkan prosedur kompleks yang sulit untuk dipahami, diuji dan dipelihara. Ada hubungan antara *Cyclomatic Complexity* dan resiko dalam suatu prosedur.

Tabel 2. 8 Hubungan antara *Cyclomatic Complexity* dan Resiko

CC	Type of Procedure	Risk
1-4	<i>A simple procedure</i>	<i>Low</i>
5-10	<i>A well structured and stable procedure</i>	<i>Low</i>
11-20	<i>A more complex procedure</i>	<i>Moderate</i>
21-50	<i>A complex procedure, alarming</i>	<i>High</i>
>50	<i>An error-prone, extremely troublesome, untestable procedure</i>	<i>Very high</i>

2.2.12.2 BlackBox

Menurut Pressman [17] *Black-Box testing* berfokus pada persyaratan fungsional perangkat lunak yang memungkinkan *engineers* untuk memperoleh set kondisi *input* yang sepenuhnya akan melaksanakan persyaratan fungsional untuk sebuah program. *Black-Box testing* berusaha untuk menemukan kesalahan dalam kategori berikut:

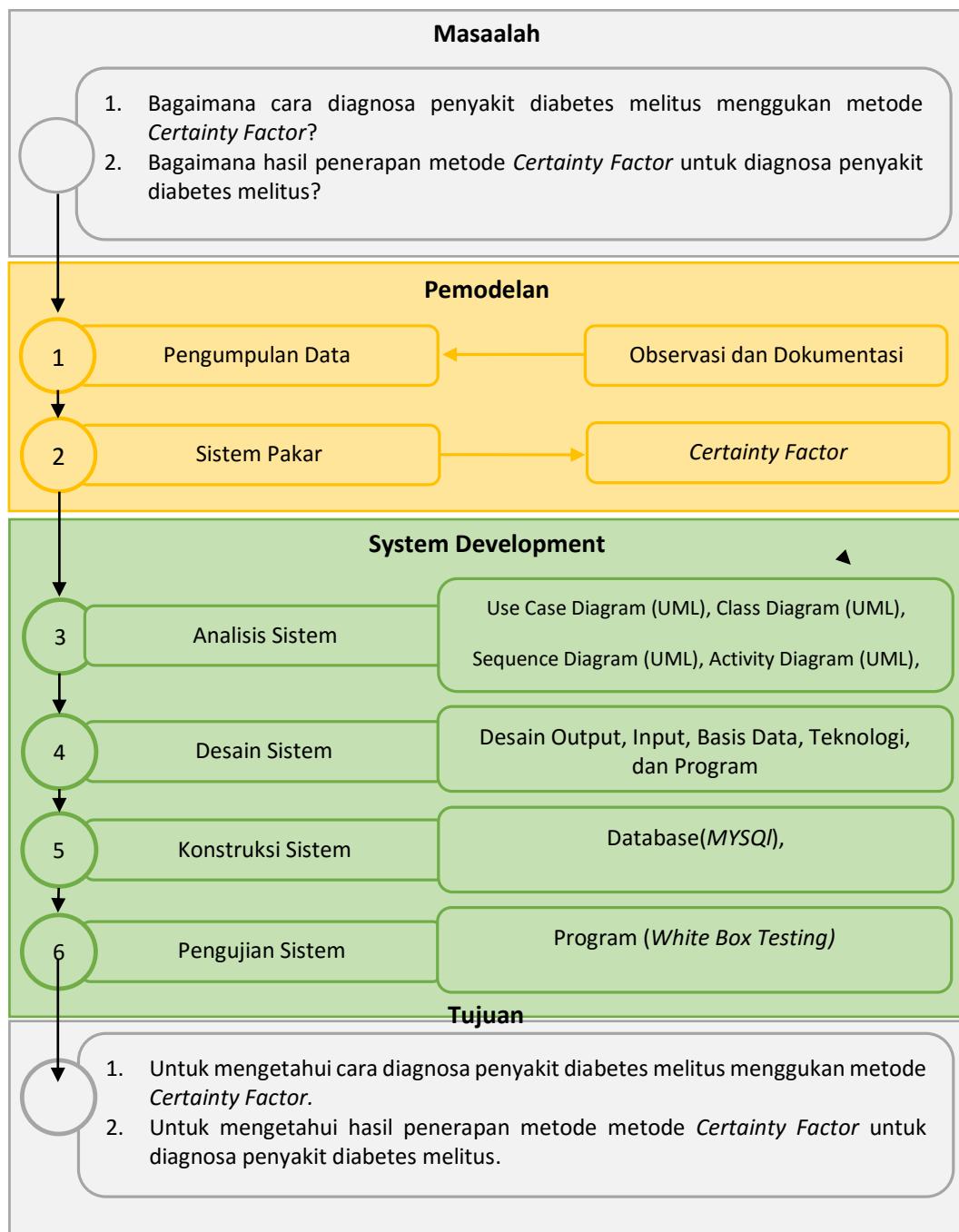
1. Fungsi yang tidak benar atau fungsi yang hilang
2. Kesalahan antarmuka
3. Kesalahan dalam struktur data atau akses *database* eksternal
4. Kesalahan perilaku (*behavior*) atau kesalahan kinerja
5. Inisialisasi dan pemutusan kesalahan

Tes ini dirancang untuk menjawab beberapa pertanyaan-pertanyaan berikut ini:

- a. Bagaimana validitas fungsional diuji?
- b. Bagaimana perilaku dan kinerja sistem diuji?
- c. Apa kelas *input* akan membuat kasus uji yang baik?
- d. Apakah sistem *sensitive* terhadap nilai input tertentu?
- e. Bagaimana batas-batas kelas data yang terisolasi?
- f. Kecepatan dan volume data seperti apa yang dapat ditolerir sistem?
- g. Efek apakah yang akan menspesifikasikan kombinasi data dalam sistem operasi?
1. Ciri-Ciri *Black Box Testing*
 - a. *Black box testing* berfokus pada kebutuhan fungsional pada *software*, berdasarkan pada spesifikasi kebutuhan dari *software*.
 - b. *Black box testing* bukan teknik alternatif daripada *white box testing*. Lebih daripada itu, ia merupakan pendekatan pelengkap dalam mencakup *error* dengan kelas yang berbeda dari metode *white box testing*.
 - c. *Black box testing* melakukan pengujian tanpa pengetahuan detil struktur internal dari sistem atau komponen yang dites. juga disebut

- sebagai *behavioral testing, specification-based testing, input/output testing* atau *functional testing*
2. Jenis teknik *design* tes yang dapat dipilih berdasarkan pada tipe testing yang akan digunakan.
 - a. *Equivalence Class Partitioning*
 - b. *Boundary Value Analysis*
 - c. *State Transitions Testing*
 - d. *Cause-Effect Graphing*
 3. Kategori *error* yang akan diketahui melalui *black box testing*
 - a. Fungsi yang hilang atau tak benar
 - b. *Error* dari antar-muka
 - c. *Error* dari struktur data atau akses eksternal database
 - d. *Error* dari kinerja atau tingkah laku
 - e. *Error* dari inisialisasi dan terminasi

2.3 Kerangka Pemikiran



Gambar 2. 7 Kerangka Pemikiran

BAB III

METODE PENELITIAN

3.1 Jenis, Metode dan Objek Penelitian

Dipandang dari tingkat penerapannya penelitian ini merupakan penelitian terapan sedangkan metode penelitian yang digunakan adalah metode deskriptif. Yang menjadi objek penelitian pada penelitian ini adalah penyakit diabetes melitus. Penelitian ini di mulai pada November 2020 sampai dengan Februari 2021.

3.2 Pengumpulan Data

Tahap pengumpulan data dapat diperoleh secara langsung dari objek penelitian. Cara-cara yang mendukung untuk mendapatkan data primer adalah sebagai berikut:

a) *Study sistem*

Yaitu teknik pengumpulan data dengan mengadakan suatu penelitian secara langsung, dengan cara yaitu:

1. Wawancara (*Interview*)

Wawancara yaitu teknik pengumpulan data dengan cara mengadakan tanya jawab secara langsung dengan narasumber yang terkait dengan permasalahan yang diambil.

2. Pengamatan (*Observasi*)

Dengan teknik ini, penulis mengamati dan mencatat secara cermat dan teliti tentang penyakit diabetes melitus.

b) *Study pustaka(Literatur)*

Teknik ini penulis lakukan untuk menunjang penelitian, dengan membaca dan mempelajari buku-buku yang berhubungan dengan penelitian penulis.

3.3 Konstruksi Sistem

Pada tahap ini menerjemahkan hasil kedalam kode-kode program kemudian membangun sistemnya. Alat bantu yang digunakan pada tahap ini adalah *PHP* dengan menggunakan *database*.

3.4 Analisa Sistem

Analisis sistem menggunakan pendekatan berorientasi *procedural/structural*:

- a) Use Case Diagram, menggunakan alat bantu UML
- b) Class Diagram, menggunakan alat bantu UML
- c) Sequence Diagram menggunakan alat bantu UML
- d) Activity Diagram menggunakan alat bantu UML
- e) Kamus Data menggunakan alat bantu Ms. Word.

3.5 Tahap Pengujian

Tahap ini di lakukan setelah semua model selesai di buat, dan program dapat berjalan, di mana seluruh perangkat lunak, program tambahan,dan semua program yang terlibat dalam pembangunan sistem diuji untuk memastikan sistem dapat berjalan sesuai dengan rancangan atau belum, pengujian yang di lakukan dengan dua teknik pengujian, yaitu:

- a. *Whitebox Testing*

Dalam pengujian *whitebox* dengan membuat bagan alir program, *litsning* program, grafik alir, pengujian *basis path* serta perhitungan *cicломatic complexity*.

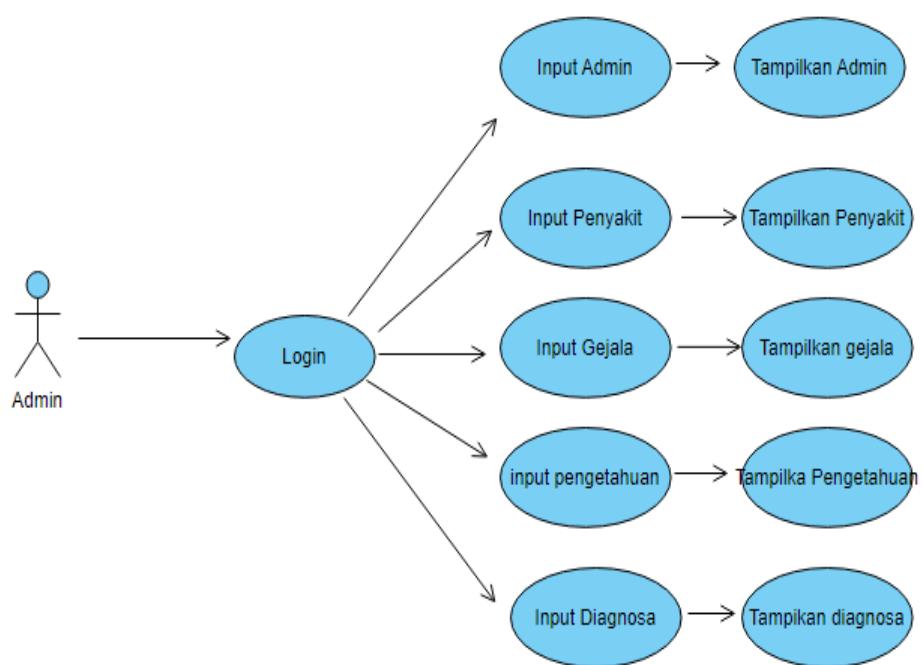
- b. *Blackbox Testing*

Pengujian *blackbox* yang termasuk dalam tahap ini yaitu menguji antarmuka sistem, apakah sebuah sistim setelah di berikan ke pengguna dapat di operasikan atau tidak.

BAB IV

HASIL PENELITIAN

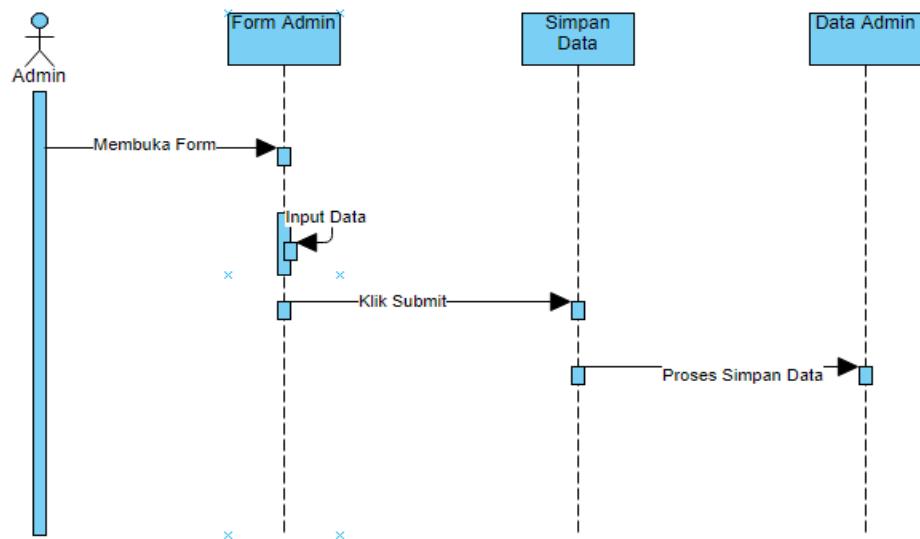
4.1 Hasil Pengembangan Sistem



Gambar 4. 1 *Use Case Diagram Sistem Pakar Diabetes*

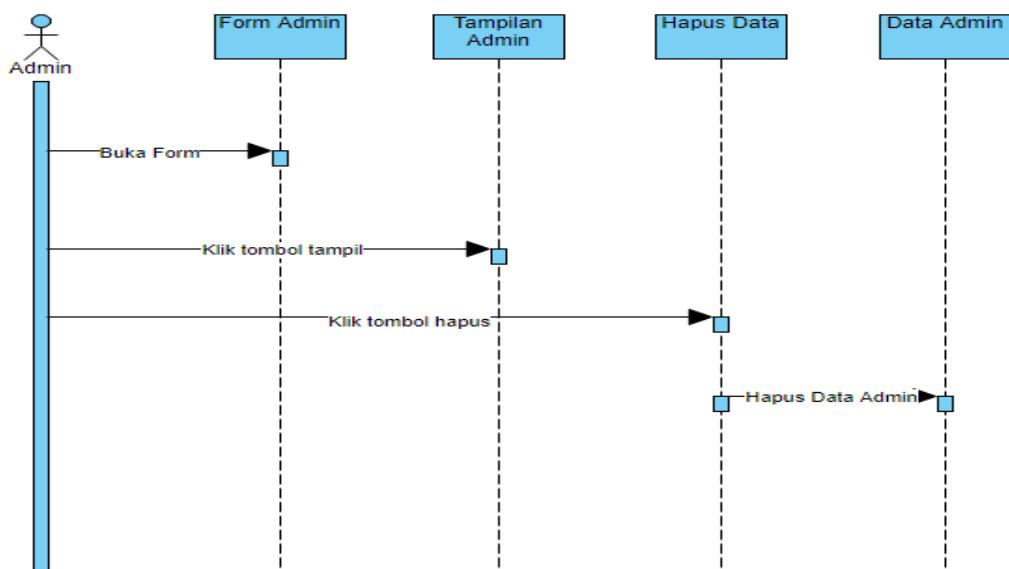
4.1.1 Sequence Diagram

4.1.1.1 Sequence Diagram Input Admin



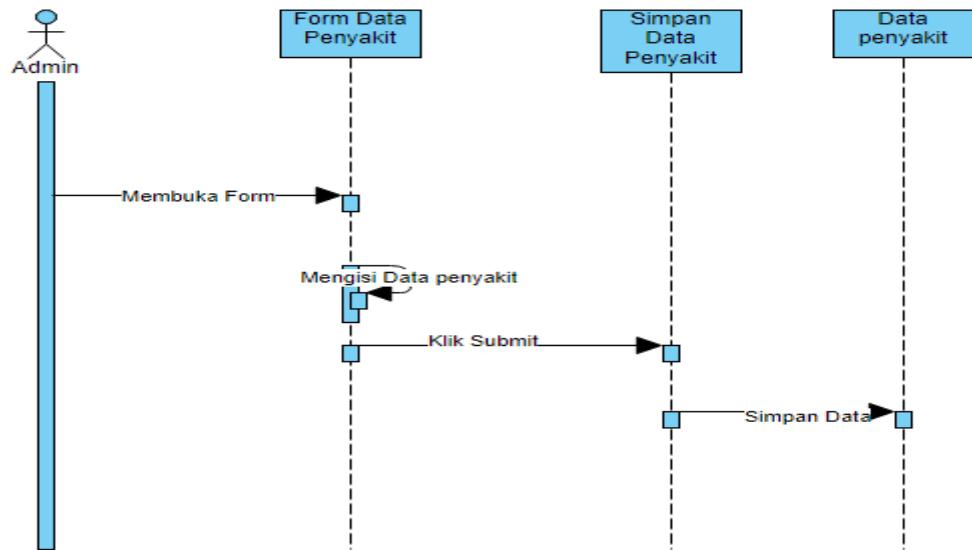
Gambar 4. 2 Sequence Diagram Input Admin

4.1.1.2 Sequence Diagram Tampilan Admin



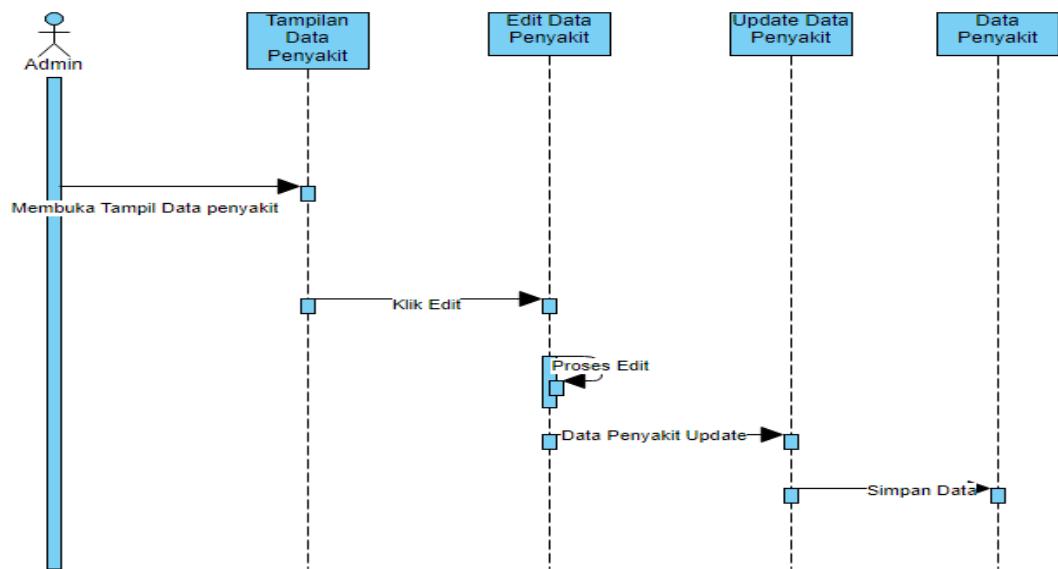
Gambar 4. 3 Sequence Diagram Tampilan Admin

4.1.1.3 Sequence Diagram Input Data Penyakit



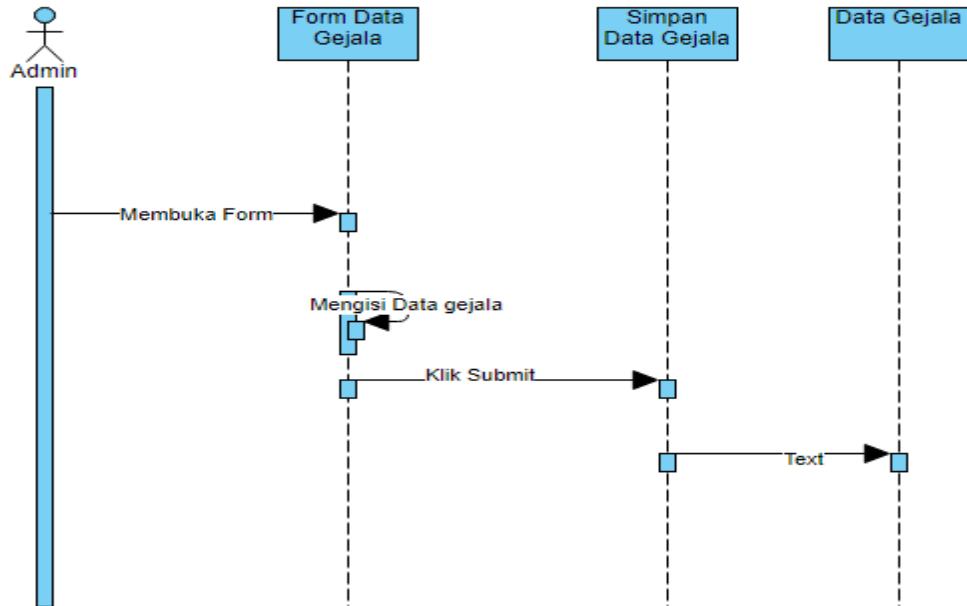
Gambar 4. 4 Sequence Diagram Input Data Penyakit

4.1.1.4 Sequence Diagram Tampil Data Penyakit



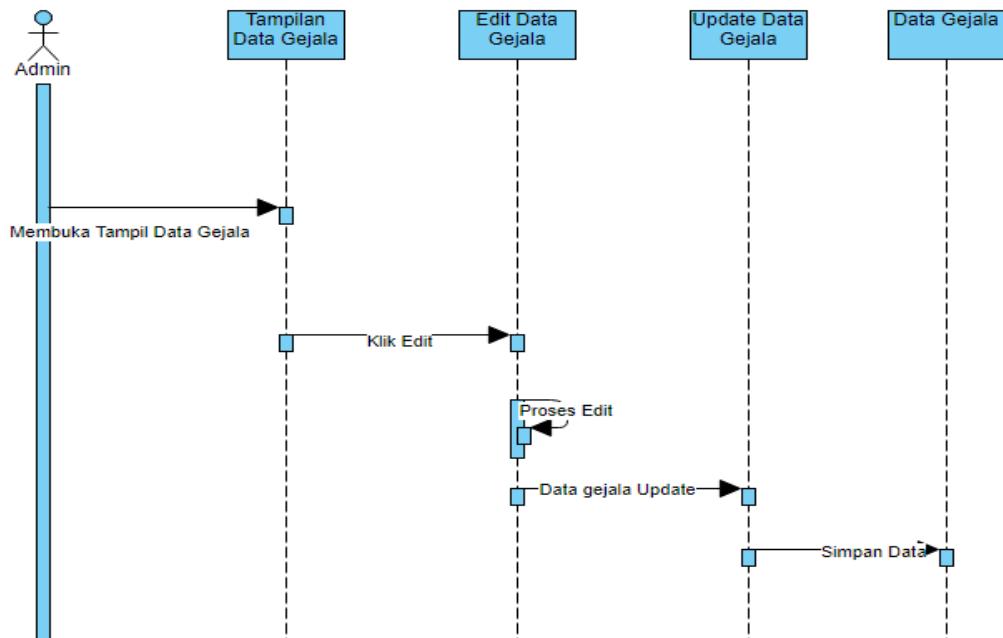
Gambar 4. 5 Sequence Diagram Tampil Data Penyakit

4.1.1.5 Sequence Diagram Input Data Gejala



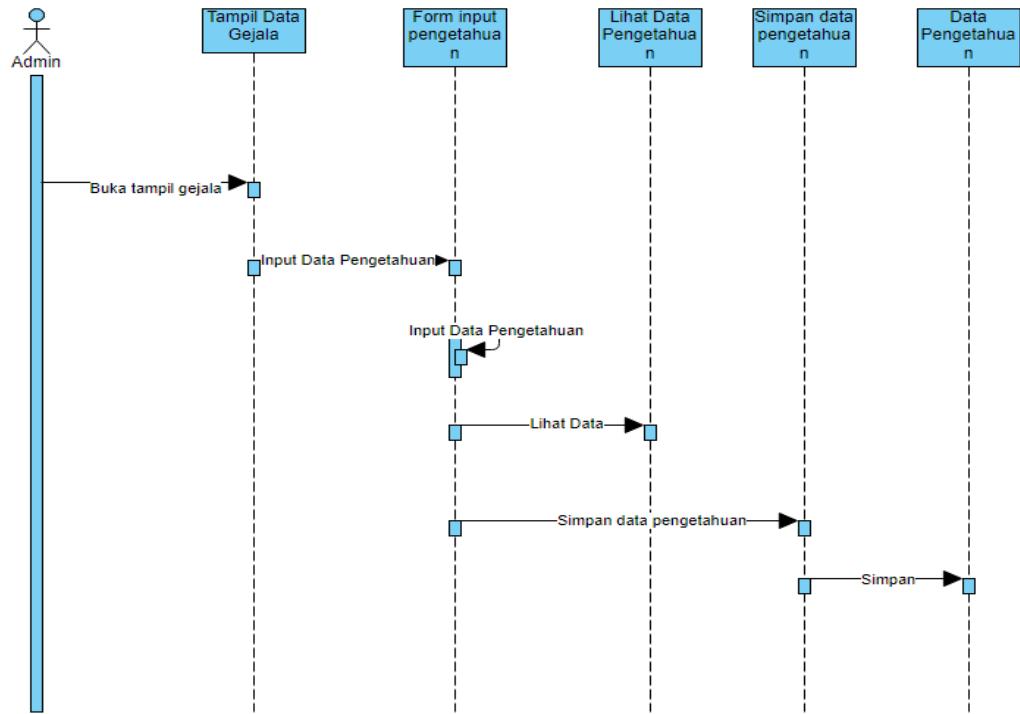
Gambar 4. 7 Sequence Diagram *Input Data Gejala*

4.1.1.6 Sequence Diagram Tampil Data Gejala



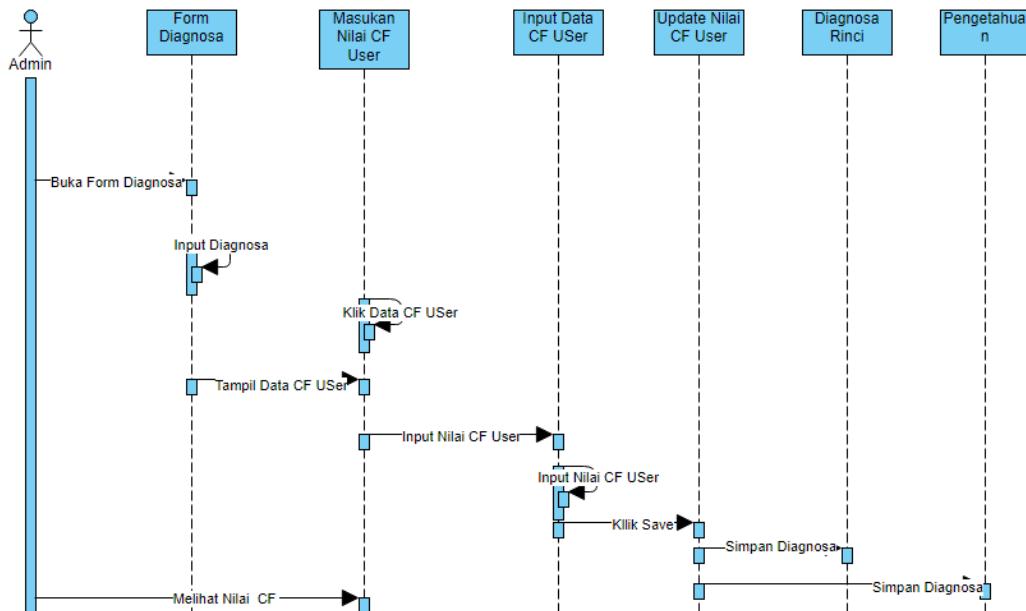
Gambar 4. 8 Sequence Diagram *Tampil Data Gejala*

4.1.1.7 Sequence Diagram Tampil Data Pengetahuan



Gambar 4. 9 Sequence Diagram Tampil Data Pengetahuan

4.1.1.8 Sequence Diagram Diagnosa



Gambar 4. 10 Sequence Diagram Diagnosa

4.1.2 Kamus Data

Kamus data merupakan suatu penjelasan tertulis mengenai suatu data yg berada di dalam database. Kamus data untuk DAD sitem pakar Diagnosa Penyakit Diabetes Melitus adalah :

1. Kamus Data Admin

Tabel 4. 1Kamus Data : Data Admin

Nama Arus Data	D_Admin		
Bentuk Data	Field		
Arus Data	A-1.1p,1.1p-F1		
Penjelasan	Admin diberikan hak penuh mengakses system		
Periode	Bulanan		
Volume	1-4		
Struktur Data			
No	Keterangan	Tipe data	Ukuran
1	Id_user	Varchar	30
2	Nama_user	Varchar	30
3	Password	Varchar	30

2. Kamus Data Penyakit

Tabel 4. 2Kamus Data : Data Penyakit

Nama Arus Data	D_Penyakit		
Bentuk Data	Field		
Arus Data	A-1.2p,1.2p-F2		
Penjelasan	Data Penyakit Diabetes		
Periode	Bulanan		
Volume	1-10		
Struktur Data			
No	Keterangan	Tipe data	Ukuran
1	Kode_Penyakit	Varchar	10
2	Nama_Penyakit	Varchar	30
3	Pengobatan	Varchar	100

3. Kamus Data Gejala

Tabel 4. 3Kamus Data : Data Gejala

Nama Arus Data	D_Gejala		
Bentuk Data	Field		
Arus Data	A-1.3p,1.3p-F3		
Penjelasan	Data Gejala Penyakit Diabetes		
Periode	Bulanan		
Volume	1-10		
Struktur Data			
No	Keterangan	Tipe data	Ukuran
1	Kode_Gejala	Varchar	10
2	Nama_Gejala	Varchar	100

4. Kamus Data Nilai CF Pakar

Tabel 4. 4Kamus Data : Data Nilai CF Pakar

Nama Arus Data	Data Nilai CF Pakar		
Bentuk Data	Field		
Arus Data	A-2p,2p-F4,F4-3p		
Penjelasan	Data Nilai CF Pakar		
Periode	Bulanan		
Volume	1-40		
Struktur Data			
No	Keterangan	Tipe data	Ukuran
1	Kode_CF	Varchar	20
2	Kode_Gejala	Varchar	10
3	Kode_Penyakit	Varchar	10
4	Nilai_CFPakar	Number	5

5. Kamus Data Gejala User

Tabel 4. 5 Kamus Data : Data Gejala User

Nama Arus Data	Data Gejala User		
Bentuk Data	Field		
Arus Data	A-2p,2p-F4,F4-3p		
Penjelasan	Data Gejala User		
Periode	Harian		
Volume	1-10		
Struktur Data			

No	Keterangan	Tipe data	Ukuran
1	Kode_Diagnosa	Varchar	20
2	Nama_user	Varchar	50
3	Kode_gejala	Varchar	10
4	Nilai_User	Number	5

6. Kamus Data Hasil Diagnosa

Tabel 4. 6Kamus Data : Hasil Diagnosa

Nama Arus Data	Data Hasil Diagnosa		
Bentuk Data	Field		
Arus Data	4p-F6,4p-B		
Penjelasan	Data Hasil Diagnosa semua gejala		
Periode	Bulanan		
Volume	1-10		
Struktur Data			
No	Keterangan	Tipe data	Ukuran
1	Kode_CF	Varchar	20
2	Kode_Gejala	Varchar	10
3	Kode_Penyakit	Varchar	10
4	Nilai_CFUser	Number	5
5	Nilai_CFPakar	Number	5
6	Nilai_Combinasi	Number	5
7	Nilai_Gabungan	Number	5

4.1.3 Desain Keluaran Secara Umum

Tabel 4. 7Rancangan *Output* Secara Umum

No/Kode Output	Nama Output	Type Output	Media Output	Alat Output	Jumlah Tembusan	Distribusi	Periode
DO-01	Hasil Perhitungan CF	Ekternal	Kertas	Printer	2	User	Harian
DO-02	Hasil Diagnosa Penyakit Diabetes	Eksternal	Kertas	Printer	2	User	Harian

4.1.4 Desain *Input* Secara Umum

Tabel 4. 8Rancangan *Input* secara umum

No/Kode Input	Nama Form Input	Arus Data	Penjelasan	Periode	Struktur Data
FI-01	Form Input Data Admin	Admin	Input user id dan nama user	Bulanan	Id user, nama user dan password
FI-02	Form input penyakit	Admin	Input data penyakit	Bulanan	Kode penyakit , nama penyakit dan pengobatan
FI-03	Form input gejala	Admin	Input data gejala	Bulanan	Kode gejala, nama gejala
FI-04	Form input Nilai CF Pakar	Admin	Input data nilai CF daripakar	Bulanan	Kode CF, Kodegejala, kode penyakit dan nilai CF Pakar
FI-04	Form input Gejala User	User	Input data gejala yang dirasakan user	Harian	Kode Diagnosa, nama user, kode gejala, nilai user

4.1.5 Desain Database Secara Umum

Tabel 4. 9Rancangan Database Secara Umum

Kode File	Nama File	Tipe File	Kunci File	Keterangan
F1	D_Admin	Master	Id_User	Menyimpan data admin
F2	D_Penyakit	Master	Kode_Penyakit	Menyimpan data penyakit
F3	D_Gejala	Master	Kode_Gejala	Menyimpan data gejala
F4	D_CF	Transaksi	Kode_CF	Menyimpan data CF
F5	Gejala_User	Transaksi	Kode_Gejala	Menyimpan gejala user
F6	Diagnosa	Transaksi	Kode_CF	Menyimpan hasil diagnose

4.1.5 Desain Sistem Secara Terinci

4.1.5.1 Desain Input Secara Terinci

1. Desain Input Data Admin

Data Admin

INPUT DATA ADMIN

Nama User	<input type="text" value="admin"/>
Masukkan Nama User	
Pasword	<input type="password" value="*****"/>
Masukkan Pasword	
<input type="button" value="Save"/> <input type="button" value="Cancel"/> Lihat Data	

Gambar 4. 11 Desain Input Data Admin

2. Desain *Input* Data Penyakit

Data Penyakit

INPUT DATA PENYAKIT

Kode Penyakit	<input type="text"/>
	Masukkan Kode Penyakit contoh P01
Nama Penyakit	<input type="text"/>
	Masukkan Nama Penyakit
Pengobatan	<input type="text"/>
	Masukkan Pengobatan
Pencegahan	<input type="text"/>
	Masukkan Pencegahan
<input type="button" value="Save"/> <input type="button" value="Cancel"/> Lihat Data	

Gambar 4. 12 Desain *Input* Data Penyakit

3. Desain *Input* Data Gejala

Data Gejala

INPUT DATA GEJALA

Kode Gejala	<input type="text"/>
	Masukkan Nama Gejala
Nama Gejala	<input type="text"/>
	Masukkan Nama Gejala
<input type="button" value="Save"/> <input type="button" value="Cancel"/> Lihat Data	

Gambar 4. 13 Desain *Input* Data Gejala

4. Desain *Input* Data Pengetahuan

Data Pengetahuan

INPUT DATA PENGETAHUAN

Kode Gejala	<input type="text"/>	
Nama Penyakit	P01 Diabetes Tipe 1 	
Nilai CF	<input type="text"/>	
Nilai 0 - 100		
<input type="button" value="Save"/>	<input type="button" value="Cancel"/>	<input type="button" value="Lihat Data"/>

Gambar 4. 14 Desain *Input* Data Pengetahuan

5. Desain *Input* Data User dan Gejala yang Dirasakan

Data Diagnosa

INPUT DATA USER DAN GEJALA YANG DIRASAKAN

Kode Diagnosa	<input type="text" value="D1"/>															
Nama User	<input type="text"/>															
Tanggal Diagnosa	<input type="text"/>															
<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>No</th> <th>Kode Gejala</th> <th>Nama Gejala</th> </tr> </thead> <tbody> <tr><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td></tr> </tbody> </table>		No	Kode Gejala	Nama Gejala												
No	Kode Gejala	Nama Gejala														
<input type="button" value="Save"/>	<input type="button" value="Cancel"/>															

Gambar 4. 15 Desain *Input* Data User dan Gejala yang Dirasakan

6. Desain *Input Data Nilai User*

Data Nilai User

INPUT DATA NILAI USER

Kode	D1P01G04
Kode Diagnosa	D1
Kode CF	P01G04
Nama Gejala	Berat Badan Turun
Nilai User	Sangat Pasif

Save **Cancel**

Gambar 4. 16 Desain *Input Data Nilai User*

4.1.6 Desain *Database* Secara Terinci

1. Tabel D_Admin

Tabel 4. 10Data Tabel D_Admin

No	Name	Type	Collation	Attribut
1	Nama_user	Varchar(30)	Utf8mb4_general_ci	
2	Password	Varchar(32)	Utf8mb4_general_ci	

2. Tabel Data Nilai CF Pakar

Tabel 4. 11Data Tabel D_Nilai CF Pakar

No	Name	Type	Collation	Attribut
1	Kode_cf	Varchar(20)	Utf8mb4_general_ci	
2	Kode_gejala	Varchar(10)	Utf8mb4_general_ci	
3	Kode penyakit	Varchar(10)	Utf8mb4_general_ci	
4	Nilai_CFPakar	Double		

3. Tabel Data Gejala

Tabel 4. 12Data Tabel D_Gejala

No	Name	Type	Collation	Attribut
1	Kode_gejala	Varchar(10)	Utf8mb4_general_ci	
2	Nama_gejala	Varchar(100)	Utf8mb4_general_ci	

4. Tabel Data Penyakit

Tabel 4. 13 Data Tabel D_Penyakit

No	Name	Type	Collation	Attribut
1	Kode_penyakit	Varchar(10)	Utf8mb4_general_ci	
2	Nama_penyakit	Varchar(30)	Utf8mb4_general_ci	
3	Pengobatan	Varchar(200)	Utf8mb4_general_ci	
4	Pencegahan	Varchar(100)	Utf8mb4_general_ci	

5. Tabel Data Hasil Diagnosa

Tabel 4. 14 Data Tabel D_Hasil Diagnosa

No	Name	Type	Collation	Attribut
1	Kode_cf	Varchar(20)	Utf8mb4_general_ci	
2	Kode_gejala	Varchar(10)	Utf8mb4_general_ci	
3	Kode penyakit	Varchar(10)	Utf8mb4_general_ci	
4	Nilai_cfpakar	Double		
5	Nilai_cfuser	Double		
6	Nilai_kombinasi	Double		
7	Nilai_gabungan	Double		

No	Name	Type	Collation	Attribut
1	Kode	Varchar(20)	Utf8mb4_general_ci	
2	Kode_diagnosa	Varchar(20)	Utf8mb4_general_ci	
3	Kode_cf	Varchar(20)	Utf8mb4_general_ci	
4	Nilai_user	Double		

No	Name	Type	Collation	Attribut
1	Kode_diagnosa	Varchar(20)	Utf8mb4_general_ci	
2	Tanggal_diagnosa	Varchar(15)	Utf8mb4_general_ci	
3	Nama_user	Varchar(50)	Utf8mb4_general_ci	
4	Kode penyakit	Varchar(10)	Utf8mb4_general_ci	
5	Nilai_hasil	Double		

4.2 Pengujian Sistem

4.2.1 Listing Program Uji White Box

Listing Kode

```

<?php
    include_once("config.php");      1

    $kode=$_POST['kode'];
    $kodetcf=substr($_POST['kode'],2,6);      2
    $nilai=$_POST['nilaiuser']/100;

    $result = mysqli_query($mysqli, "SELECT * FROM diagnosa WHERE      3
        Kode_cf='$kodetcf');

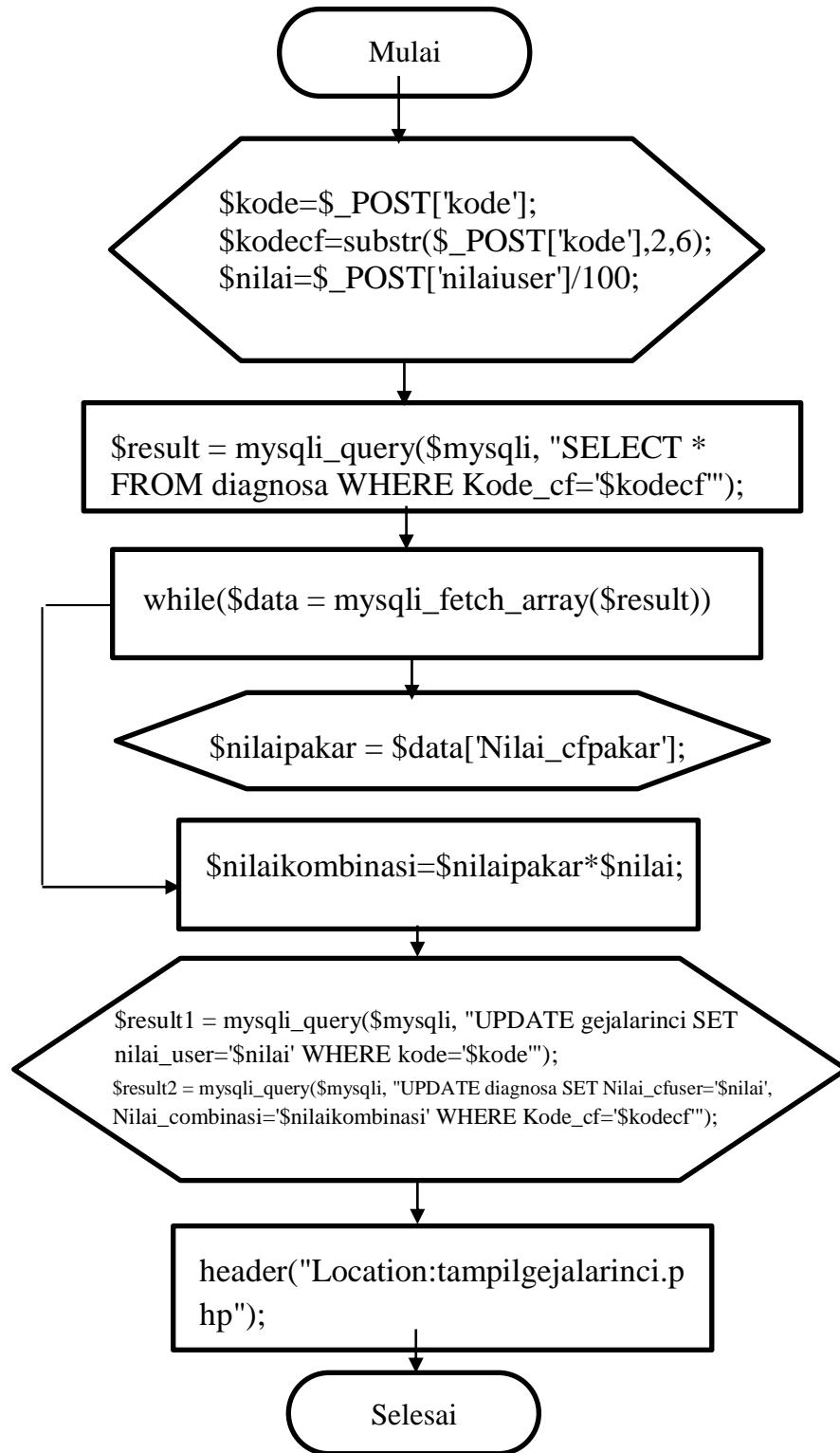
    while($data = mysqli_fetch_array($result))      4
    {
        $nilaipakar = $data['Nilai_cfpakar'];      5
    }
    $nilaikombinasi=$nilaipakar*$nilai;      6

    //include database connection file

    $result1 = mysqli_query($mysqli, "UPDATE gejalarinci SET
        nilai_user='$nilai' WHERE kode='$kode'");
    $result2 = mysqli_query($mysqli, "UPDATE diagnosa SET      7
        Nilai_cfuser='$nilai', Nilai_combinasi='$nilaikombinasi'
        WHERE
        Kode_cf='$kodetcf');
    header("Location:tampilgejalarinci.php");      8
?>

```

4.2.2 Flowchart Program Uji White Box



Gambar 4. 17 Flowchart Program Uji White Box

BAB V

PEMBAHASAN PENELITIAN

5.1 Pembahasan Sistem

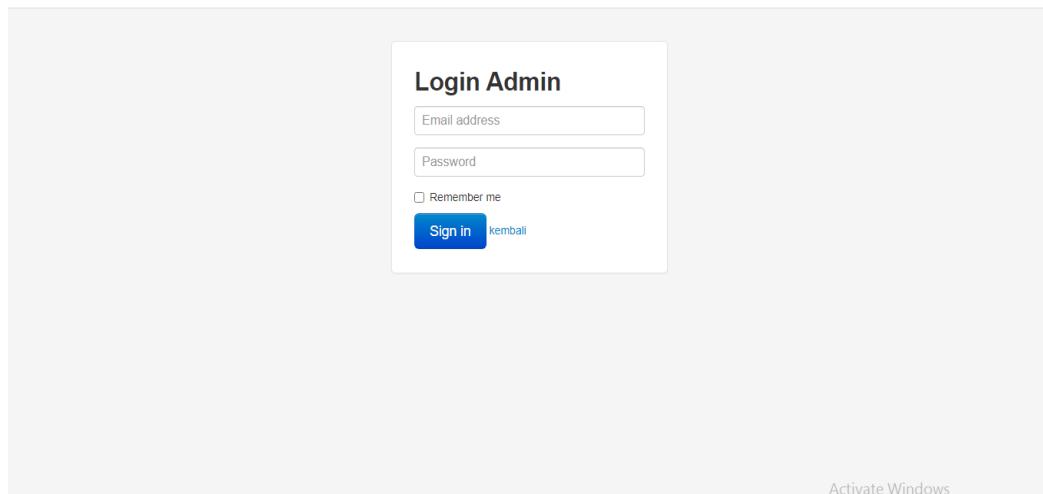
5.1.1 Tampilan Halaman Utama



Gambar 5. 1Halaman Utama

Form menu utama tampil ketika kita mengetikan alamat url sistem pakar diagnosa penyakit diabetes melitus. Dalam tampilan tersebut tampak menu “home” dan menu “login”. Menu login tersebut digunakan untuk masuk dalam beranda admin.

5.1.2 Tampilan Halaman *Login Admin*



Gambar 5. 2 Halaman Login Admin

Setelah mengklik tombol login pada halaman utama tadi, maka akan tampil halaman login admin. Halaman login admin ini merupakan syarat masuk agar bisa mengakses beranda admin, yakni dapat dilakukan dengan mengetikan data user dan pasword admin yang selanjutnya sistem akan memproses data login tersebut sehingga menampilkan beranda admin.

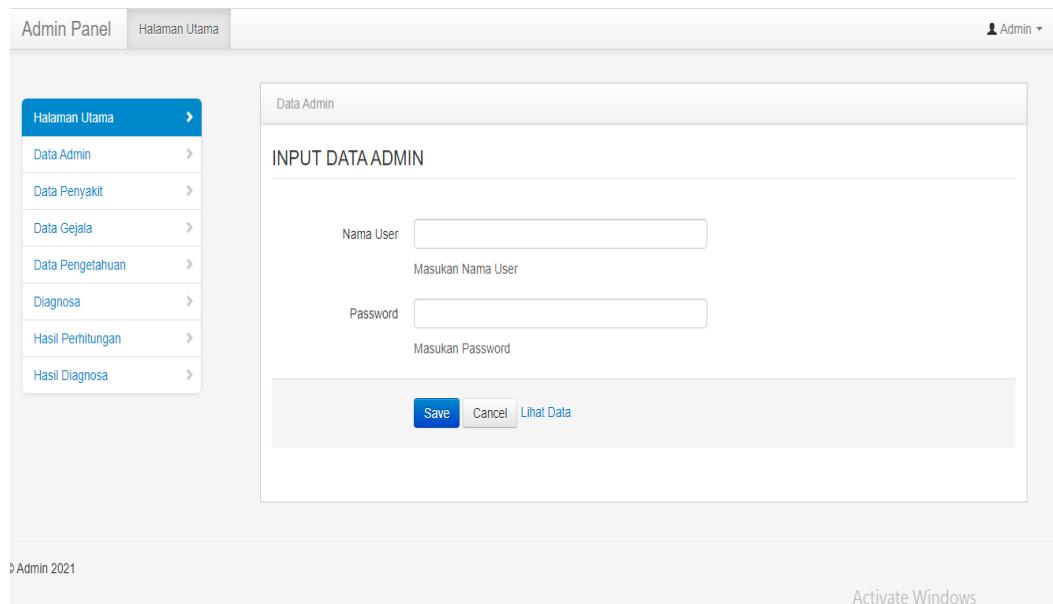
5.1.3 Tampilan Halaman Admin



Gambar 5. 3 Halaman Admin

Tampilan beranda admin setelah admin berhasil memasukkan data user dan password,yakni menampilkan penjelasan secara umum penyakit diabetes, serta terdapat tujuh tombol menu yakni halaman utama, data admin, data penyakit, data gejala, data pengetahuan, diagnosa, hasil perhitungan dan hasil diagnosa yang nanti dapat digunakan ketika mendiagnosa penyakit diabetes melitus menggunakan program sistem pakar ini.

5.1.4 Tampilan *Form Input Data Admin*



Gambar 5. 4*Form Input Data Admin*

Langkah yang dilakukan dalam form input data admin yakni dengan memasukkan nama user dan pasword yang benar agar sistem dapat memproses input data admin tersebut.

5.1.5 Tampilan Form Input Data Penyakit

The screenshot shows a web-based administrative interface. At the top, there's a header with 'Admin Panel' and 'Halaman Utama' tabs, and a user profile 'Admin'. On the left, a sidebar menu lists several options: 'Halaman Utama', 'Data Admin', 'Data Penyakit' (which is currently selected), 'Data Gejala', 'Data Pengetahuan', 'Diagnosa', 'Hasil Perhitungan', and 'Hasil Diagnosa'. The main content area is titled 'Data Penyakit' and contains a sub-section titled 'INPUT DATA PENYAKIT'. This section has four input fields: 'Kode Penyakit' (placeholder: 'Masukan Kode Penyakit contoh P01'), 'Nama Penyakit' (placeholder: 'Masukan Nama Penyakit'), 'Pengobatan' (placeholder: 'Masukan Pengobatan'), and 'Pencegahan' (placeholder: 'Masukan Pencegahan'). Below these fields are three buttons: 'Save' (blue), 'Cancel' (grey), and 'Lihat Data' (blue). A watermark 'Activate Windows' is visible at the bottom right.

Gambar 5. 5Form Input Data Penyakit

Setelah mengsave form input data admin selanjutnya kita memasukkan data penyakit dengan mengklik menu data penyakit, sehingga akan tampil form input data penyakit. Dalam form input data penyakit tersebut dapat diakses dengan memasukkan kode penyakit, nama penyakit, pengobatan dan pencegahan.

5.1.6 Tampilan Form Input Data Gejala

The screenshot shows a web-based administrative interface. At the top, there's a header with 'Admin Panel' and 'Halaman Utama' tabs, and a user profile 'Admin'. On the left, a sidebar menu lists 'Halaman Utama', 'Data Admin', 'Data Penyakit', 'Data Gejala' (which is currently selected), 'Data Pengetahuan', 'Diagnosa', 'Hasil Perhitungan', and 'Hasil Diagnosa'. The main content area is titled 'Data Gejala' and contains a sub-section titled 'INPUT DATA GEJALA'. It has two input fields: 'Kode Gejala' with placeholder 'Masukan Kode Gejala contoh G01' and 'Nama Gejala' with placeholder 'Masukan Nama Gejala'. Below these fields is a button bar with 'Save' (highlighted in blue), 'Cancel', and 'Lihat Data'.

Gambar 5. 6Form Input Data Gejala

Setelah data penyakit, selanjutnya yang diinput adalah data gejala. Hal ini dilakukan dengan cara mengklik menu data gejala, selanjutnya akan tampil form input data gejala. Pada form tersebut di perintahkan untuk mengisi kode gejala dan nama gejala, kemudian di *save*.

5.1.7 Tampilan Form Input Pengetahuan 1

The screenshot shows a web-based administrative interface. At the top, there's a header with 'Admin Panel' and 'Halaman Utama' tabs, and a user profile icon labeled 'Admin'. The main content area has a sidebar on the left with various menu items. The central part of the screen displays a table titled 'Data Gejala' (Symptom Data) with 9 rows of information.

No	Kode Gejala	Nama Gejala
1	G01	Sering buang air kecil, terutama pada malam hari
2	G02	Sering haus
3	G03	Sering merasa lapar (polyphagia)
4	G04	Berat badan turun.
5	G05	Pandangan kabur
6	G06	Kelelahan
7	G07	Mudah diserang penyakit infeksi
8	G08	Luka yang lama sembuh
9	G09	Merasa kaku atau kesemutan pada kaki

At the bottom of the content area, there is a link labeled 'Tampil Data'.

Gambar 5. 7Form Input Pengetahuan 1

Pada form input pengetahuan 1 akan menampilkan kode gejala dan nama gejala penyakit diabetes melitus yang nanti dapat digunakan untuk memproses hasil diagnosa penyakit tersebut.

5.1.8 Tampilan Form Input Pengetahuan 2

Admin Panel Halaman Utama

Admin

Halaman Utama >

Data Admin >

Data Penyakit >

Data Gejala >

Data Pengetahuan > **INPUT DATA PENGETAHUAN**

Diagnosa >

Hasil Perhitungan >

Hasil Diagnosa >

Kode Gejala G01

Kode Penyakit P01 | Diabetes Tipe 1

Nilai CF

Nilai 0-100

Save Cancel Lihat Data

© Vincent Gabriel 2013

Gambar 5. 8Form Input Pengetahuan 2

Setelah menampilkan form input pengetahuan 1, selanjutnya mengisi form input pengetahuan 2 dengan cara memasukan kode gejala, kode penyakit dan nilai cf kemudian di simpan dengan mengklik tombol *save*.

5.1.9 Tampilan Form Input Diagnosa

The screenshot shows a web-based administrative interface. At the top left is a sidebar titled 'Admin Panel' with links to 'Halaman Utama', 'Data Admin', 'Data Penyakit', 'Data Gejala', 'Data Pengetahuan', 'Diagnosa', 'Hasil Perhitungan', and 'Hasil Diagnosa'. The main content area has a header 'Data Diagnosa' and a sub-header 'INPUT DATA USER DAN GEJALA YANG DIRASAKAN'. It contains three input fields: 'Kode Diagnosa' (D1), 'Nama User' (empty), and 'Tanggal Diagnosa' (empty). Below these is a table with columns 'No', 'Kode Gejala', and 'Nama Gejala'. The table rows are:

No	Kode Gejala	Nama Gejala
1	<input type="checkbox"/> G01	Sering buang air kecil, terutama pada malam hari
2	<input type="checkbox"/> G02	Sering haus
3	<input type="checkbox"/> G03	Sering merasa lapar (polyphagia)
4	<input type="checkbox"/> G04	Berat badan turun.
5	<input type="checkbox"/> G05	Pandangan kabur
6	<input type="checkbox"/> G06	Kelelahan
7	<input type="checkbox"/> G07	Mudah diserang penyakit infeksi
8	<input type="checkbox"/> G08	Luka yang lama sembuh

At the bottom right of the table, there is a watermark: 'Activate Windows Go to Settings to activate Windows.'

Gambar 5. 9Form Input Diagnosa

Setelah menyimpan data pengetahuan, selanjutnya ke tahap menginput diagnosa dengan cara mengklik menu diagnosa kemudian akan tampil form input diagnosa. Pada form ini akan ditampilkan kode diagnosa, nama user dan tanggal diagnosa yang harus diisi kemudian mengisi gejala user dengan mencentang pada kotak yang telah disediakan.

5.1.10 Tampilan *Form Input Nilai CF User 1*

No	Kode	Kode Diagnosa	Kode CF	nilai_user	Isi Nilai User
1	D1P01G01	D1	P01G01	0	Update Nilai
2	D1P01G03	D1	P01G03	0	Update Nilai
3	D1P02G01	D1	P02G01	0	Update Nilai
4	D1P02G03	D1	P02G03	0	Update Nilai

Hitung P1

Gambar 5. 10*Form Input Nilai CF User 1*

Setelah menginput diagnosa sesuai dengan gejala user selanjutnya akan ditampilkan *form input nilai CF user 1* yang berisi tentang informasi kode diagnosa, kode cf, nilai user yang dapat diisi dengan cara mengupdate nilai user.

5.1.11 Tampilan Form Input Nilai CF User 2

The screenshot shows the 'Admin Panel' interface. The left sidebar has a blue header 'Halaman Utama' and lists several menu items: Data Admin, Data Penyakit, Data Gejala, Data Pengetahuan, Diagnosa, Hasil Perhitungan, and Hasil Diagnosa. The 'Halaman Utama' item is highlighted with a blue background. The main content area has a title 'Data Nilai User' and a sub-section 'INPUT DATA NILAI USER'. It contains five input fields: 'Kode' with value 'D1P01G01', 'Kode Diagnosa' with value 'D1', 'Kode CF' with value 'P01G01', 'Nama Gejala' with value 'Sering buang air kecil, terutama pada malam hari', and 'Nilai User' with a dropdown menu showing 'Sangat Pasti'. At the bottom are 'Save' and 'Cancel' buttons.

Gambar 5. 11Form Input Nilai CF User 2

Pada *form input* nilai CF user 2 ini akan ditampilkan form untuk mengisi nilai *user* agar terupdate, kemudian di simpan dengan mengklik tombol *save*.

5.1.12 Tampilan Form Hasil Perhitungan

Perhitungan CF							
No	Kode CF	Kode Gejala	Kode Penyakit	Nilai CF Pakar	Nilai CF User	Nilai Kombinasi	Perhitungan
1	P01G01	G01	P01	0.78	0	0	0
2	P01G02	G02	P01	0.66	0	0	0
3	P01G03	G03	P01	0.78	0	0	0
4	P01G04	G04	P01	0.67	0	0	0
5	P01G05	G05	P01	0.98	0	0	0
6	P01G06	G06	P01	0.78	0	0	0
7	P01G07	G07	P01	0.75	0	0	0
8	P01G08	G08	P01	0.65	0	0	0
9	P01G09	G09	P01	0.82	0	0	0
10	P02G01	G01	P02	0.88	0	0	0
11	P02G02	G02	P02	0.77	0	0	0
12	P02G03	G03	P02	0.88	0	0	0
13	P02G04	G04	P02	0.79	0	0	0
14	P02G05	G05	P02	0.9	0	0	0

Gambar 5. 12Form Hasil Perhitungan

Setelah menginput nilai user kemudian akan ditampilkan hasil perhitungan CF sesuai yang tampil pada form hasil perhitungan.

5.1.13 Tampilan *Form* Hasil Diagnosa

LAPORAN HASIL DIAGNOSA				
Gejala User Terinci				
No	Kode CF	Kode Gejala	Kode Penyakit	Nilai CF Pakar
1	D1P01G01	D1	P01G01	0
2	D1P01G03	D1	P01G03	0
3	D1P02G01	D1	P02G01	0
4	D1P02G03	D1	P02G03	0
HASIL DIAGNOSA				
No	Kode Diagnosa	Tanggal Diagnosa	Nama	Nilai
1	D1		P00	Tidak ada 0
Pengobatan dan Pencegahan				
No	Kode Penyakit	Nama Penyakit	Pengobatan	Pencegahan
1	P00	Tidak ada	-	Activate Windows Go to Settings to activate Window

Gambar 5. 13*Form* Hasil Diagnosa

Pada *form* hasil diagnosa akan ditampilkan laporan hasil diagnosa yang berisikan gejala user yang terinci setelah perhitungan, hasil diagnosa dan tipe penyakit diabetes yang diderita serta pengobatan dan pencegahannya.

BAB VI

KESIMPULAN DAN SARAN

6.1 Kesimpulan

Dari hasil perancangan aplikasi ini, penulis dapat mengambil beberapa kesimpulan, yaitu sebagai berikut :

1. Dapat dibuat cara mendiagnosa penyakit diabetes melitus di Rumah Sakit Islam Kota Gorontalo menggunakan aplikasi dengan sistem pakar
2. Aplikasi ini dapat digunakan dengan cara cukup menginput gejala-gejala yang diderita oleh pasien sehingga mempermudah dalam mendiagnosa penyakit diabetes melitus
3. Aplikasi sistem pakar ini menggunakan metode *certainty factor*.

6.2 Saran

Penulis menyadari sistem ini masih sangat jauh dari kesempurnaan sehingga membutuhkan pengembangan yang lebih lagi. Adapun saran dari penulis untuk pengembangan sistem ini kedepannya, yaitu :

1. Pengembangan Sumber Daya Manusia (SDM) sangat diperlukan agar pengembangan sistem dapat optimal.
2. Sistem pakar ini perlu dikembangkan lagi dalam bentuk aplikasi *portable* sehingga jangkauan masyarakat terhadap sistem ini dapat lebih luas lagi.

DAFTAR PUSTAKA

- [1] P. Hikmat, “Komplikasi Kronik dan Penyakit Penyerita pada Diabetes,” *Med. Care*, pp. 1–5, 2017.
- [2] Kementerian kesehatan republik indonesia, “Tetap Produktif, Cegah Dan Atasi Diabetes Mellitus,” *pusat data dan informasi kementerian kesehatan RI*. 2020.
- [3] K. E. Setyaputri, A. Fadlil, and S. Sunardi, “Analisis Metode Certainty Factor pada Sistem Pakar Diagnosa Penyakit THT,” *J. Tek. Elektro*, vol. 10, no. 1, pp. 30–35, 2018, doi: 10.15294/jte.v10i1.14031.
- [4] H. Fahmi, “Sistem Pakar Mendiagnosa Penyakit Mata KatarakDengan Metode Certainty Factor Berbasis Web,” *Matics*, vol. 11, no. 1, p. 27, 2019, doi: 10.18860/mat.v11i1.7673.
- [5] I. H. Santi and B. Andari, “Sistem Pakar Untuk Mengidentifikasi Jenis Kulit Wajah dengan Metode Certainty Factor,” *INTENSIF J. Ilm. Penelit. dan Penerapan Teknol. Sist. Inf.*, vol. 3, no. 2, p. 159, 2019, doi: 10.29407/intensif.v3i2.12792.
- [6] A. Sucipto, Y. Fernando, R. I. Borman, and N. Mahmuda, “Penerapan Metode Certainty Factor Pada Diagnosa Penyakit Saraf Tulang Belakang,” *J. Ilm. FIFO*, vol. 10, no. 2, p. 18, 2019, doi: 10.22441/fifo.2018.v10i2.002.
- [7] M. Arifin, S. Slamin, and W. E. Y. Retnani, “Penerapan Metode Certainty Factor Untuk Sistem Pakar Diagnosis Hama Dan Penyakit Pada Tanaman Tembakau,” *Berk. Sainstek*, vol. 5, no. 1, p. 21, 2017, doi: 10.19184/bst.v5i1.5370.
- [8] Subakti, Irfan , & Rahmat Hidayatullah. “ Aplikasi Sistem Pakar Untuk Diagnosa Awal Gangguan Kesehatan Secara Mandiri Menggunakan Variabel-Centered Intellegent Rule Sistem”, 2007.
- [9] Kusrini. *Sistem pakar. Teori dan Aplikasi*. Yogyakarta : Andi Offset, 2006.
- [10] Suyanto,“Artificial Intelligence”,Informatika. Bandung, 2007
- [11] Sutarbi, Tata. 2013. *Analisis Sistem Informasi*. Yogyakarta: Andi.
- [12] Witten, Jeffrey L, et all, *Metode Desain & Analisis Sistem*, Edisi 6, Edisi International, Mc Graw Hill, Andi, Yogyakarta: 2004.

- [13] Kadir, Abdul. 2003. *Pengenalan Sistem Informasi*. Edisi I. Yogyakarta. Andi Yogyakarta.
- [14] Bently, Lonnie D, Jeffrey L Whitten, (2007). Systems Analisys and Design for the Global Enterprise Seventh Edition, New York: McGraw-Hill.
- [15] Sri Dharwiyanti & Romi Satria Wahono, 2013. Kuliah Umum Ilmu Komputer. Jakarta.
- [16] Hariyanto, Bambang, 2004. *Sistem Informasi Basis Data: Pemodelan, Perancangan, dan Terapannya*. Informatika, Bandung.
- [17] Pressman, R.S. 2002. *Rekayasa Perangkat Lunak : Pendekatan Praktis (Buku I)*. Yogyakarta : Andi Yogyakarta.

LAMPIRAN

Lampiran 1. Listing Program

Listing Program Awal Diagnosa

```

<!DOCTYPE html>
<?php
// Create database connection using config file
include_once("config.php");
$kodep1="P01";
$no=1;
$cf1=0;
$hasil=0;
$result = mysqli_query($mysqli, "SELECT * FROM diagnosa");
while($data = mysqli_fetch_array($result)){
    $cf2=$data['Nilai_combinasi'];
    $kode=$data['Kode_penyakit'];
    $kodetcf=$data['Kode_cf'];
    if ($kode!=$kodep1){
        $cf1=0;
        $kodep1="P02";
    }
    else{
        $cf1=$hasil;
    }
    $hasil=$cf1+$cf2-($cf1*$cf2);

    $result1 = mysqli_query($mysqli, "UPDATE diagnosa SET
Nilai_gabungan='$hasil' WHERE Kode_cf='$kodetcf'");
}

$maks=0;
$koded="D1";
$result2 = mysqli_query($mysqli, "SELECT * FROM diagnosa");
while($data = mysqli_fetch_array($result2)){
    $Nilai=$data['Nilai_gabungan'];
    $kode=$data['Kode_penyakit'];
    if($Nilai>$maks){
        $maks=$Nilai;
        $kodep=$kode;
        $result3 = mysqli_query($mysqli, "UPDATE gejalauser
SET Kode_penyakit='$kodep',Nilai_hasil='$maks' WHERE
Kode_diagnosa='$koded'");
    }
}

$result4 = mysqli_query($mysqli, "SELECT * FROM diagnosa");
?>
<html>
```

```

<head>
<title>Sistem Pakar Diabetes</title>
<!-- Bootstrap -->
<link href="bootstrap/css/bootstrap.min.css" rel="stylesheet"
media="screen">
<link href="bootstrap/css/bootstrap-responsive.min.css" rel="stylesheet"
media="screen">
<link href="assets/styles.css" rel="stylesheet" media="screen">
<link href="assets/DT_bootstrap.css" rel="stylesheet" media="screen">
<!--[if lte IE 8]><script language="javascript" type="text/javascript"
src="vendors/fLOT/excanvas.min.js"></script><![endif]-->
<!-- HTML5 shim, for IE6-8 support of HTML5 elements -->
<!--[if lt IE 9]>
<script
src="http://html5shim.googlecode.com/svn/trunk/html5.js"></script>
<![endif]-->
<script src="vendors/modernizr-2.6.2-respond-1.1.0.min.js"></script>
</head>

<body>
<div class="navbar navbar-fixed-top">
<div class="navbar-inner">
<div class="container-fluid">
<a class="btn btn-navbar" data-toggle="collapse" data-target=".nav-
collapse"><span class="icon-bar"></span>
<span class="icon-bar"></span>
<span class="icon-bar"></span>
</a>
<a class="brand" href="#">Admin Panel</a>
<div class="nav-collapse collapse">
<ul class="nav pull-right">
<li class="dropdown">
<a href="#" role="button" class="dropdown-toggle" data-
toggle="dropdown"><i class="icon-user"></i> Admin <i
class="caret"></i>
</a>
<ul class="dropdown-menu">
<li>
<a tabindex="-1" href="#">Profile</a>
</li>
<li class="divider"></li>
<li>
<a tabindex="-1" href="logout.php">Logout</a>
</li>
</ul>
</li>
</ul>
<ul class="nav">

```

```
<li class="active">
<a href="#">Halaman Utama</a>
</li>

</ul>
</div>
<!--.nav-collapse -->
</div>
</div>
</div>
<div class="container-fluid">
<div class="row-fluid">
<div class="span3" id="sidebar">
<ul class="nav nav-list bs-docs-sidenav nav-collapse collapse">
<li class="active">
<a href="index.php"><i class="icon-chevron-right"></i> Halaman
Utama</a>
</li>
<li>
<a href="admin.php"><i class="icon-chevron-right"></i> Data
Admin</a>
</li>
<li>
<a href="penyakit.php"><i class="icon-chevron-right"></i> Data
Penyakit</a>
</li>
<li>
<a href="gejala.php"><i class="icon-chevron-right"></i> Data Gejala</a>
</li>
<li>
<a href="pengetahuan.php"><i class="icon-chevron-right"></i> Data
Pengetahuan</a>
</li>
<li>
<a href="diagnosa.php"><i class="icon-chevron-right"></i>
Diagnosa</a>
</li>
<li>
<a href="lap_perhitungan.php"><i class="icon-chevron-right"></i> Hasil
Perhitungan</a>
</li>
<li>
<a href="lap_diagnosa.php"><i class="icon-chevron-right"></i>Hasil
Diagnosa</a>
</li>
</ul>
</div>
<!--span-->
<div class="span9" id="content">
```

```

<div class="row-fluid">
<!-- block -->
<div class="block">
<div class="navbar navbar-inner block-header">
<div class="muted pull-left">Perhitungan CF</div>
</div>
<div class="block-content collapse in">
<div class="span12">

    <table class="table">
        <thead>
            <tr>
                <th>No</th>
                <th>Kode CF</th>
                <th>Kode Gejala</th>
                <th>Kode Penyakit</th>
                <th>Nilai CF Pakar</th>
                <th>Nilai CF User</th>
                <th>Nilai Kombinasi</th>
                <th>Perhitungan</th>
            </tr>
        </thead>
        <tbody>
            <tr>
                <td>$no</td>
                <td>$data['Kode_cf']</td>
                <td>$data['Kode_gejala']</td>
                <td>$data['Kode_penyakit']</td>
                <td>$data['Nilai_cfpakar']</td>
                <td>$data['Nilai_cfuser']</td>
                <td>$data['Nilai_combinasi']</td>
            </tr>
        </tbody>
    </table>
</div>

```

```

echo"<td>".$data['Nilai_gabungan']."</td>";
echo"</tr>";
echo"</tbody>";

$no++;

}

?>
</table>

</div>
</div>
</div>
<!-- /block -->
</div>
</div>
</div>
<hr>
<footer>
<p>&copy; Admin 2021</p>
</footer>
</div>
<!--/.fluid-container-->

<script src="vendors/jquery-1.9.1.js"></script>
<script src="bootstrap/js/bootstrap.min.js"></script>
<script src="vendors/datatables/js/jquery.dataTables.min.js"></script>

<script src="assets/scripts.js"></script>
<script src="assets/DT_bootstrap.js"></script>
<script>
    $(function() {

        });

    </script>
</body>

</html>

```

Listing Program Hitung Diagnosa

```

4. <!DOCTYPE html>
5. <?php
6. // Create database connection using config file
7. include_once("config.php");
8. session_start();
9.
10. if (!isset($_SESSION["username"])) {
11.     header("Location:login.php");

```

```

12.     exit;
13. }
14.
15. $username=$_SESSION["username"];
16.
17. $result1 = mysqli_query($mysqli, "SELECT * FROM diagnosa");
18. while($data = mysqli_fetch_array($result1)) {
19.     $kodecf=$data['Kode_cf'];
20.     $nilai=0;
21.     $result = mysqli_query($mysqli, "UPDATE diagnosa SET
    Nilai_cfuser='$nilai',Nilai_combinasi=$nilai,Nilai_gabungan=$nilai WHERE
    Kode_cf='$kodecf'");
22. }
23.
24. $result3 = mysqli_query($mysqli, "DELETE FROM gejalarinci");
25. $result4 = mysqli_query($mysqli, "DELETE FROM gejalauser");
26.
27. if(isset($_POST['tampil'])){
28.     if(empty($_POST['checkbox_list'])){
29.         echo"Pilih Gejala...";
30.     }
31.     else{
32.         $koded=$_POST['kodediagnosa'];
33.         $tanggald=$_POST['tanggal'];
34.         $namad=$_POST['namauuser'];
35.         $kodep="P00";
36.         $hasil=0;
37.         $nilaiu=0;
38.         $result = mysqli_query($mysqli, "INSERT INTO
            gejalauser(Kode_diagnosa,Tanggal_diagnosa,Nama_user,Kode_penyakit,Nilai
            _hasil) VALUES('$koded','$tanggald','$namad','$kodep','$hasil')");
39.
40.
41.         foreach($_POST['checkbox_list'] as $item){
42.             $kodep="P01";
43.             $kodecf=$kodep.$item;
44.             $koder=$koded.$kodecf;
45.             $result1 = mysqli_query($mysqli, "INSERT INTO
                gejalarinci(kode,Kode_diagnosa,kode_cf,nilai_user)
                VALUES('$koder','$koded','$kodecf','$nilaiu')");
46.         }
47.
48.         foreach($_POST['checkbox_list'] as $item){
49.             $kodep="P02";
50.             $kodecf=$kodep.$item;
51.             $koder=$koded.$kodecf;
52.             $result1 = mysqli_query($mysqli, "INSERT INTO
                gejalarinci(kode,Kode_diagnosa,kode_cf,nilai_user)
                VALUES('$koder','$koded','$kodecf','$nilaiu')");
53.         }

```

```
54.         header("Location:tampilgejalarinci.php");
55.     }
56. }
57.
58. $result = mysqli_query($mysqli, "SELECT * FROM data_gejala");
59. ?>
60. <html>
61.
62. <head>
63. <title>Sistem Pakar Diabetes</title>
64. <!-- Bootstrap -->
65. <link href="bootstrap/css/bootstrap.min.css" rel="stylesheet" media="screen">
66. <link href="bootstrap/css/bootstrap-responsive.min.css" rel="stylesheet"
   media="screen">
67. <link href="assets/styles.css" rel="stylesheet" media="screen">
68. <!--[if lte IE 8]><script language="javascript" type="text/javascript"
   src="vendors/fLOT/excanvas.min.js"></script><![endif]-->
69. <!-- HTML5 shim, for IE6-8 support of HTML5 elements -->
70. <!--[if lt IE 9]>
71. <script src="http://html5shim.googlecode.com/svn/trunk/html5.js"></script>
72. <![endif]-->
73. <script src="vendors/modernizr-2.6.2-respond-1.1.0.min.js"></script>
74. </head>
75.
76. <body>
77. <div class="navbar navbar-fixed-top">
78. <div class="navbar-inner">
79. <div class="container-fluid">
80. <a class="btn btn-navbar" data-toggle="collapse" data-target=".nav-
   collapse"><span class="icon-bar"></span>
81. <span class="icon-bar"></span>
82. <span class="icon-bar"></span>
83. </a>
84. <a class="brand" href="#">Admin Panel</a>
85. <div class="nav-collapse collapse">
86. <ul class="nav pull-right">
87. <li class="dropdown">
88. <a href="#" role="button" class="dropdown-toggle" data-
   toggle="dropdown"><i class="icon-user"></i> Admin <i class="caret"></i>
89.
90. </a>
91. <ul class="dropdown-menu">
92. <li>
93. <a tabindex="-1" href="#">Profile</a>
94. </li>
95. <li class="divider"></li>
96. <li>
97. <a tabindex="-1" href="logout.php">Logout</a>
98. </li>
99. </ul>
```

```
100.    </li>
101.    </ul>
102.    <ul class="nav">
103.        <li class="active">
104.            <a href="#">Halaman Utama</a>
105.        </li>
106.
107.    </ul>
108.    </div>
109.    <!--.nav-collapse -->
110.    </div>
111.    </div>
112.    </div>
113.    <div class="container-fluid">
114.        <div class="row-fluid">
115.            <div class="span3" id="sidebar">
116.                <ul class="nav nav-list bs-docs-sidenav nav-collapse collapse">
117.                    <li class="active">
118.                        <a href="index.php"><i class="icon-chevron-right"></i> Halaman
          Utama</a>
119.                    </li>
120.                    <li>
121.                        <a href="admin.php"><i class="icon-chevron-right"></i> Data
          Admin</a>
122.                    </li>
123.                    <li>
124.                        <a href="penyakit.php"><i class="icon-chevron-right"></i> Data
          Penyakit</a>
125.                    </li>
126.                    <li>
127.                        <a href="gejala.php"><i class="icon-chevron-right"></i> Data
          Gejala</a>
128.                    </li>
129.                    <li>
130.                        <a href="pengetahuan.php"><i class="icon-chevron-right"></i> Data
          Pengetahuan</a>
131.                    </li>
132.                    <li>
133.                        <a href="diagnosa.php"><i class="icon-chevron-right"></i>
          Diagnosa</a>
134.                    </li>
135.                    <li>
136.                        <a href="lap_perhitungan.php"><i class="icon-chevron-right"></i>
          Hasil Perhitungan</a>
137.                    </li>
138.                    <li>
139.                        <a href="lap_diagnosa.php"><i class="icon-chevron-right"></i>Hasil
          Diagnosa</a>
140.                    </li>
141.                </ul>
```

```

142.      </div>
143.      <!--/span-->
144.      <div class="span9" id="content">
145.          <!-- morris stacked chart -->
146.
147.          <div class="row-fluid">
148.              <!-- block -->
149.              <div class="block">
150.                  <div class="navbar navbar-inner block-header">
151.                      <div class="muted pull-left">Data Diagnosa</div>
152.                  </div>
153.                  <div class="block-content collapse in">
154.                      <div class="span12">
155.                          <form class="form-horizontal" method="POST" id="form1">
156.                              <fieldset>
157.                                  <legend>INPUT DATA USER DAN GEJALA YANG
158.                                      DIRASAKAN</legend>
159.                                  <div class="control-group">
160.                                      <label class="control-label" for="typeahead">Kode Diagnosa</label>
161.                                      <div class="controls">
162.                                          <input type="text" class="span6" id="typeahead"
163.                                              name="kodediagnosa" value="D1">
164.                                      </div>
165.                                      </div>
166.                                      <div class="control-group">
167.                                          <label class="control-label" for="typeahead">Nama User</label>
168.                                          <div class="controls">
169.                                              <input type="text" class="span6" id="typeahead" name="namauser">
170.                                              </div>
171.                                              <div class="control-group">
172.                                                  <label class="control-label" for="typeahead">Tanggal
173.                                                      Diagnosa</label>
174.                                                  <div class="controls">
175.                                                      <input type="text" class="span6" id="typeahead" name="tanggal">
176.                                                      </div>
177.
178.          <div class="block-content collapse in">
179.              <div class="span12">
180.                  <table class="table">
181.                      <thead>
182.                          <tr>

```

```

182.                                     <th>No</th>
183.                                     <th>Kode
    Gejala</th>
184.                                     <th>Nama
    Gejala</th>
185.                                     </tr>
186.                                     </thead>
187.                                     <?php
188.                                     $no=1;
189.                                     while($data = mysqli_fetch_array($result)) {
190.                                         echo"<tbody>";
191.                                         echo"<tr>";
192.                                         echo"<td>".$no."</td>";
193.                                         echo"<td><input type='checkbox' name='checkbox_list[]' value='".$data['Kode_gejala']."'>&nbsp;&nbsp".$data['Nama_gejala']."</td>";
194.                                         echo"<td>".$data['Nama_gejala']."</td>";
195.                                         echo"</tr>";
196.                                         echo"</tbody>";
197.                                         $no++;
198.                                         }
199.                                         ?>
200.                                     </table>
201.                                     </div>
202.                                     </div>
203.
204.
205.                                     <div class="form-actions">
206.                                         <button type="submit" class="btn btn-primary" form="form1" name="tampil">Save</button>
207.                                         <button type="reset" class="btn">Cancel</button>

```

```

208.

209.      </div>
210.      </fieldset>
211.      </form>
212.
213.      </div>
214.      </div>
215.      </div>
216.      <!-- /block -->
217.      </div>
218.                  </div>
219.              </div>
220.          </div>
221.          <!-- /block -->
222.                  </div>
223.      <!-- /validation -->
224.
225.
226.      </div>
227.      </div>
228.      <hr>
229.      <footer>
230.          <p>&copy; Admin 2021</p>
231.      </footer>
232.      </div>
233.      <!--/.fluid-container-->
234.      <link href="vendors/datepicker.css" rel="stylesheet" media="screen">
235.      <link href="vendors/uniform.default.css" rel="stylesheet"
   media="screen">
236.      <link href="vendors/chosen.min.css" rel="stylesheet"
   media="screen">
237.
238.      <link href="vendors/wysiwyg/bootstrap-wysihtml5.css"
   rel="stylesheet" media="screen">
239.
240.      <script src="vendors/jquery-1.9.1.js"></script>
241.      <script src="bootstrap/js/bootstrap.min.js"></script>
242.      <script src="vendors/jquery.uniform.min.js"></script>
243.      <script src="vendors/chosen.jquery.min.js"></script>
244.      <script src="vendors/bootstrap-datepicker.js"></script>
245.
246.      <script src="vendors/wysiwyg/wysihtml5-0.3.0.js"></script>
247.      <script src="vendors/wysiwyg/bootstrap-wysihtml5.js"></script>
248.
249.      <script src="vendors/wizard/jquery.bootstrap.wizard.min.js"></script>
250.
251.          <script type="text/javascript" src="vendors/jquery-
   validation/dist/jquery.validate.min.js"></script>
252.          <script src="assets/form-validation.js"></script>

```

```

253.
254.        <script src="assets/scripts.js"></script>
255.    <script>
256.
257.        jQuery(document).ready(function() {
258.            FormValidation.init();
259.        });
260.
261.
262.        $(function() {
263.            $(".datepicker").datepicker();
264.            $(".uniform_on").uniform();
265.            $(".chzn-select").chosen();
266.            $('.textarea').wysihtml5();
267.
268.            $('#rootwizard').bootstrapWizard({onTabShow: function(tab,
269.                navigation, index) {
270.                    var $total = navigation.find('li').length;
271.                    var $current = index+1;
272.                    var $percent = ($current/$total) * 100;
273.                    $('#rootwizard').find('.bar').css({width:$percent+'%'});
274.                    // If it's the last tab then hide the last button and show the
275.                    // finish instead
276.                    if($current >= $total) {
277.                        $('#rootwizard').find('.pager .next').hide();
278.                        $('#rootwizard').find('.pager .finish').show();
279.                        $('#rootwizard').find('.pager .finish').removeClass('disabled');
280.                    } else {
281.                        $('#rootwizard').find('.pager .next').show();
282.                        $('#rootwizard').find('.pager .finish').hide();
283.                    }
284.                });
285.            $('#rootwizard .finish').click(function() {
286.                alert('Finished!, Starting over!');
287.                $('#rootwizard').find("a[href*='tab1']").trigger('click');
288.            });
289.        });
290.
291.    </script>
292.    </body>
293.
294.    </html>

```

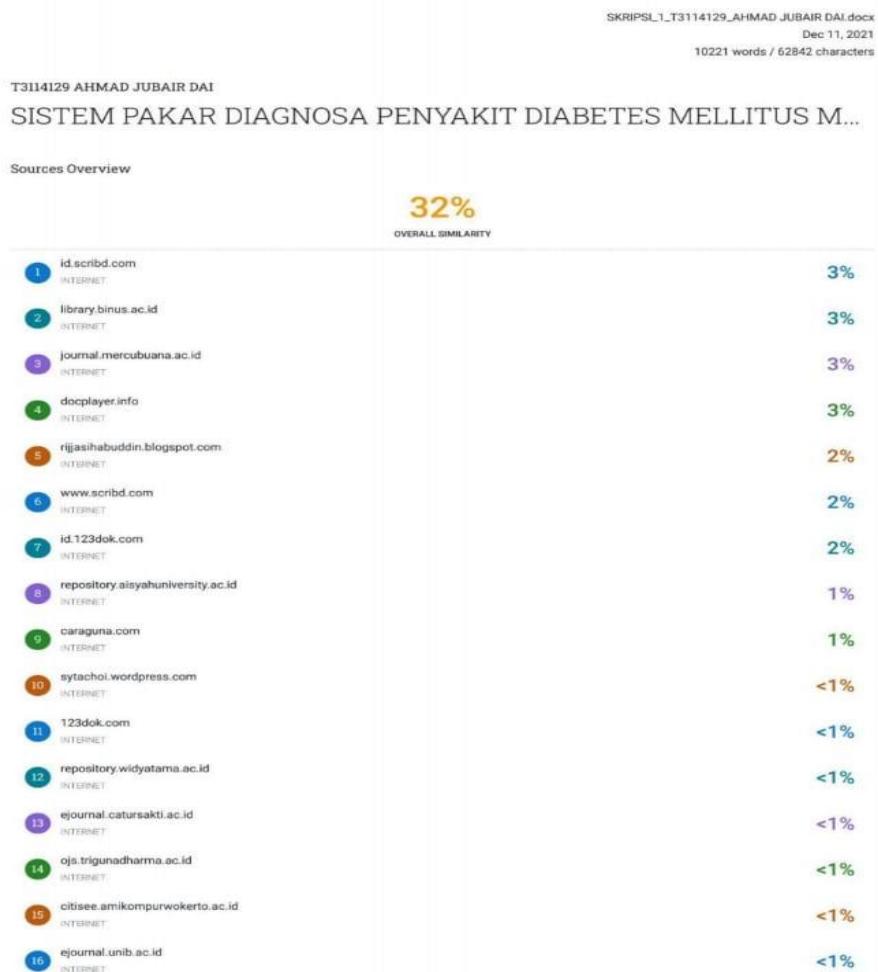
Lampiran 2. Rekomendasi Bebas Pustaka



Lampiran 3. Surat Balasan Penelitian



Lampiran 4. Hasil Turnitin



Scanned by TapScanner

17	ridhotampan99.blogspot.com INTERNET	<1%
18	ojs.stmik-banjarbaru.ac.id INTERNET	<1%
19	widuri.raharja.info INTERNET	<1%
20	www.mikroskil.ac.id INTERNET	<1%
21	mafiadoc.com INTERNET	<1%
22	edoc.site INTERNET	<1%

Lampiran 5. Surat Rekomendasi Bebas Plagiasi



**KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET DAN
UNIVERSITAS ICHSAN
(UNISAN) GORONTALO**

SURAT KEPUTUSAN MENDIKNAS RI NOMOR 84/D/O/2001
Jl. Achmad Nadjamuddin No. 17 Telp (0435) 829975 Fax (0435) 829976 Gorontalo

SURAT REKOMENDASI BEBAS PLAGIASI

No. 1088/UNISAN-G/S-BP/XII/2021

Yang bertanda tangan di bawah ini :

Nama	:	Sunarto Taliki, M.Kom
NIDN	:	0906058301
Unit Kerja	:	Pustikom, Universitas Ichsan Gorontalo

Dengan ini Menyatakan bahwa :

Nama Mahasiswa	:	AHMAD JUBAIR DAI
NIM	:	T3114129
Program Studi	:	Teknik Informatika (S1)
Fakultas	:	Fakultas Ilmu Komputer
Judul Skripsi	:	SISTEM PAKAR DIAGNOSA PENYAKIT DIABETES MELLITUS MENGGUNAKAN METODE CERTAINTY FACTOR

Sesuai dengan hasil pengecekan tingkat kemiripan skripsi melalui aplikasi Turnitin untuk judul skripsi di atas diperoleh hasil Similarity sebesar 32%, berdasarkan SK Rektor No. 237/UNISAN-G/SK/IX/2019 tentang Panduan Pencegahan dan Penanggulangan Plagiarisme, bahwa batas kemiripan skripsi maksimal 35% dan sesuai dengan Surat Pernyataan dari kedua Pembimbing yang bersangkutan menyatakan bahwa isi softcopy skripsi yang diolah di Turnitin SAMA ISINYA dengan Skripsi Aslinya serta format penulisannya sudah sesuai dengan Buku Panduan Penulisan Skripsi, untuk itu skripsi tersebut di atas dinyatakan BEBAS PLAGIASI dan layak untuk diujiankan.

Demikian surat rekomendasi ini dibuat untuk digunakan sebagaimana mestinya.

Gorontalo, 11 Desember 2021
Tim Verifikasi,



Sunarto Taliki, M.Kom
NIDN. 0906058301

Tembusan :

1. Dekan
2. Ketua Program Studi
3. Pembimbing I dan Pembimbing II
4. Yang bersangkutan
5. Arsip

Lampiran 6. Riwayat Hidup

DAFTAR RIWAYAT HIDUP

A. Biodata

Nama : Ahmad Jubair Dai
NIM : T3114129
Tempat, Tanggal Lahir : Liquica, 19 April 1996
Agama : Islam
Jenis Kelamin : Laki-Laki
No. Telepon : 0822-9063-6330
Email : ariefdai056@gmail.com



B. Riwayat Pendidikan

1. Tahun 2008, menyelesaikan Pendidikan dasar di Sekolah Dasar Negeri Tongo (SDN 7 Tongo), Kecamatan Bonepantai, Kabupaten Bonebolango,
2. Tahun 2011, menyelesaikan pendidikan di Sekolah Menengah Pertama Negeri 1 Bonepantai (SMPN 1 Bonepantai), Kecamatan Bonepantai, Kabupaten Gorontalo,
3. Tahun 2014, menyelesaikan pendidikan di Sekolah Menengah Atas Negeri 1 Bonepantai (SMAN 1 Bonepantai), Kecamatan Bonepantai Kabupaten Gorontalo,
4. Tahun 2014 diterima menjadi Mahasiswa di Perguruan Tinggi Universitas Ichsan Gorontalo Jurusan Teknik Informatika