

**SISTEM PENDUKUNG KEPUTUSAN DETEKSI DINI PENYAKIT STROKE  
MENGUNAKAN METODE *DEMPSTER SHAFER***

**SKRIPSI**

**LABORATORIUM KOMPUTASI CERDAS DAN VISUAL**

Untuk memenuhi persyaratan memperoleh gelar Sarjana Komputer



Disusun oleh:

**DEBY PUTRI INDRASWARI**

**NIM. 135150109111006**

**KEMENTERIAN RISET TEKNOLOGI DAN PENDIDIKAN TINGGI  
UNIVERSITAS BRAWIJAYA  
PROGRAM TEKNOLOGI INFORMASI DAN ILMU KOMPUTER  
PROGRAM STUDI INFORMATIKA/ILMU KOMPUTER  
MALANG  
2015**

**LEMBAR PERSETUJUAN**

**SISTEM PENDUKUNG KEPUTUSAN DETEKSI DINI PENYAKIT STROKE  
MENGUNAKAN METODE *DEMPSTER SHAFER***

**SKRIPSI**

**LABORATORIUM KOMPUTASI CERDAS DAN VISUAL**

Untuk memenuhi persyaratan memperoleh gelar Sarjana Komputer



Disusun oleh:

**DEBY PUTRI INDRASWARI**

**NIM. 135150109111006**

**Skripsi ini telah diperiksa dan disetujui oleh  
Dosen Pembimbing**

**Pembimbing I**

**Pembimbing II**

**Arief Andy Soebroto, S.T., M.Kom**

**NIP.19720425 199903 1 002**

**dr. Eko Arisetijono M., Sp.S (K)**

**19690206 200604 1 007**

**LEMBAR PENGESAHAN**

**SISTEM PENDUKUNG KEPUTUSAN DETEKSI DINI PENYAKIT STROKE  
MENGUNAKAN METODE *DEMPSTER SHAFER***

**SKRIPSI**

Disusun oleh:

**DEBY PUTRI INDRASWARI**

**NIM. 135150109111006**

Skripsi ini telah diuji dan dinyatakan lulus  
tanggal 11 Agustus 2015

**Dosen Penguji I**

**Dosen Penguji II**

---

**NIP.**

---

**NIP.**

**Dosen Penguji III**

---

**NIP.**

Mengetahui,  
Ketua Program Studi Teknik Informatika/Ilmu Komputer

**Drs. Marji, MT**

**NIP. 19670801 199203 1001**



## LEMBAR PERNYATAAN

Saya menyatakan dengan sebenar-benarnya bahwa sepanjang pengetahuan saya, di dalam naskah SKRIPSI ini tidak terdapat karya ilmiah yang pernah diajukan oleh orang lain untuk memperoleh gelar akademik di suatu perguruan tinggi dan tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis dikutip dalam naskah ini dan disebutkan dalam sumber kutipan dan daftar pustaka.

Apabila ternyata dalam naskah SKRIPSI ini dapat dibuktikan terdapat unsur-unsur PLAGIASI, saya bersedia SKRIPSI ini digugurkan dan gelar akademik yang saya peroleh (SARJANA) dibatalkan, serta diproses sesuai dengan peraturan perundang-undangan yang berlaku (UU No. 20 Tahun 2003, Pasal 25 ayat 2 dan Pasal 70).

Malang, Juni 2015

**Deby Putri Indraswari**

**135150109111006**

## KATA PENGANTAR

Puji dan syukur penulis panjatkan kehadirat Tuhan yang Maha Esa, yang telah melimpahkan berkat dan anugerah-Nya kepada penulis sehingga penulis dapat menyelesaikan skripsi ini yang berjudul **“SISTEM PENDUKUNG KEPUTUSAN DETEKSI DINI PENYAKIT STROKE MENGGUNAKAN METODE *DEMPSTER SHAFER*”**

Pada penyusunan Skripsi ini tidak semata-mata hasil kerja penulis sendiri, melainkan juga berkat dan bimbingan dan dorongan dari pihak-pihak yang telah membantu, baik secara materi maupun non materi. Maka dari itu penulis ingin mengucapkan banyak terima kasih yang tak terhingga serta penghargaan yang setinggi-tingginya kepada orang-orang yang telah membantu penulis secara langsung maupun tidak langsung kepada yang terhormat:

1. Arief Andy Soebroto, S.T.,M.Kom dan dr. Eko Arisetijono Marhendraputro, Sp.S(K) selaku dosen pembimbing skripsi yang telah membimbing dan mengarahkan penulis sehingga dapat menyelesaikan skripsi ini.
2. Ir. Sutrisno, M.T, Ir. Heru Nurwasito, M.Kom, Himawat Aryadita, S.T, M.Sc, dan Edy Santoso, S.Si., M.Kom selaku Ketua, Wakil Ketua 1, Wakil Ketua 2 dan Wakil Ketua 3 Program Teknologi Informasi dan Ilmu Komputer Universitas Brawijaya.
3. Drs. Marji, M.T selaku Ketua Prodi Informatika / Ilmu Komputer yang telah memberikan kesempatan kepada penulis untuk menyelesaikan skripsi ini.
4. Ibunda Alfi Syafi'i, Ayahanda Bambang Santoso, Dindies Lupitasari, Daneswara Alfikri Santoso dan seluruh keluarga besar atas segala nasehat, kasih sayang, perhatian dan kesabarannya di dalam membesarkan dan mendidik penulis, serta yang senantiasa tiada henti-hentinya memberikan doa dan semangat demi terselesaikannya skripsi ini.
5. Segenap bapak dan ibu dosen program studi Informatika / Ilmu Komputer beserta seluruh staff administrasi yang telah membantu selama perkuliahan.
6. Seluruh Civitas Akademika Teknik Informatika Universitas Brawijaya yang telah banyak memberi bantuan dan dukungan selama penulis menempuh studi di Teknik Informatika Universitas Brawijaya dan selama penyelesaian skripsi ini.

7. Teman – teman Teknik Informatika angkatan SAP 2013 yang telah memberikan masukan dan inspirasi kepada penulis selama menempuh studi dan menyelesaikan skripsi ini.
8. Anggota tim Penelitian Deteksi Dini Penyakit Stroke yaitu Voni Nurfaizzah, Eka Hastian Astuti, Anis Fitri Masruriyah, dan Tety Dewi Prastiti yang selalu bertukar pikiran dan semangat dengan penulis selama menyelesaikan skripsi ini.
9. Seluruh pihak yang telah membantu kelancaran penulisan tugas akhir yang tidak dapat kami sebutkan satu persatu.

Penulis menyadari bahwa penulisan skripsi ini jauh dari kata sempurna. Oleh karena itu, penulis mengharapkan masukan berupa saran dan kritik dari semua pihak demi tercapainya kesempurnaan dalam skripsi ini. Akhir kata semoga penulisan skripsi ini dapat memberikan manfaat bagi semua pihak.

Malang, Juni 2015

Penulis

**ABSTRAK**

**Deby Putri Indraswari. 2015. : Sistem Pendukung Keputusan Deteksi Dini Penyakit Stroke Menggunakan Metode *Dempster Shafer*.**

**Skripsi Program Studi Teknik Informatika, Program Teknologi Informasi dan Ilmu Komputer, Universitas Brawijaya.**

**Dosen Pembimbing: Arief Andy Soebroto, ST., M.Kom dan dr. Eko Ari Setijono, SpS(K)**

Stroke adalah gangguan fungsi saraf yang disebabkan oleh gangguan aliran darah dalam otak. Penyakit stroke ini merupakan penyebab kematian ketiga tersering di negara maju, setelah penyakit jantung dan kanker. Hal ini menyebabkan penyakit stroke harus diwaspadai. Pencegahan secara dini melalui pemeriksaan dokter perlu dilakukan untuk mengurangi tingginya angka resiko penyakit stroke. Pendeteksian resiko penyakit stroke mudah ditentukan apabila mengetahui kriteria faktor resiko secara lengkap dan terstruktur. Namun terkadang pendeteksian resiko penyakit stroke sulit ditentukan apabila terdapat faktor resiko yang terlupakan atau tidak terstruktur sehingga dokter dapat mengalami kendala atau ambigu dalam melakukan diagnosa. Untuk mengatasi permasalahan pola semi terstruktur tersebut, maka dapat diselesaikan menggunakan sistem pendukung keputusan (SPK) dengan komputasi cerdas. SPK deteksi dini penyakit stroke dibangun menggunakan metode *Dempster Shafer*. Pada penelitian dapat mendeteksi tingkat resiko penyakit stroke yaitu resiko tinggi, sedang, dan rendah dengan masukan 8 faktor resiko. Berdasarkan data yang digunakan dalam sistem ini diperoleh akurasi sebesar 90% sehingga dapat disimpulkan bahwa SPK yang dibangun menggunakan metode *Dempster Shafer* dapat berfungsi dengan baik untuk mendeteksi penyakit stroke.

**Kata Kunci :** *Dempster Shafer*, Stroke, Sistem Pendukung Keputusan

**ABSTRACT**

**Deby Putri Indraswari. 2015. : Sistem Pendukung Keputusan Deteksi Dini Penyakit Stroke Menggunakan Metode Dempster Shafer.**

**Skripsi Program Studi Teknik Informatika, Program Teknologi Informasi dan Ilmu Komputer, Universitas Brawijaya.**

**Dosen Pembimbing: Arief Andy Soebroto, ST., M.Kom dan dr. Eko Ari Setijono, SpS(K)**

*Stroke is a neurological function disorders caused by impaired blood flow in the brain. Stroke is the third most common cause of death in developed countries, after heart disease and cancer. This causes a stroke to watch. Early prevention through medical examination needs to be done to reduce the high rate of risk of stroke. The detection of the risk of stroke is determined when knowing the criteria of risk factors is complete and structured. But sometimes the detection of the risk of stroke is difficult to determine if there are risk factors that have forgotten or not structured so that doctors can experience problems or ambiguous to make diagnosis. To overcome the problem of semi-structured pattern, it can be solved using decision support systems (DSS) with intelligent computing. DSS early detection of stroke constructed using methods Dempster Shafer. In the study can detect the level of risk of stroke is high risk, medium, and low with 8 input risk factors. Based on the data used in this system is obtained accuracy of 90%. So that it can be concluded that DSS is constructed with Dempster Shafer method to function well for detecting stroke.*

**Keywords:** *Dempster Shafer, Stroke, Decision Support System*

**DAFTAR ISI**

<b>JUDUL .....</b>	<b>i</b>
<b>LEMBAR PERSETUJUAN .....</b>	<b>ii</b>
<b>LEMBAR PENGESAHAN .....</b>	<b>iii</b>
<b>LEMBAR PERNYATAAN .....</b>	<b>iv</b>
<b>KATA PENGANTAR.....</b>	<b>v</b>
<b>ABSTRAK .....</b>	<b>vii</b>
<b>DAFTAR ISI.....</b>	<b>ix</b>
<b>DAFTAR GAMBAR.....</b>	<b>xi</b>
<b>DAFTAR TABEL .....</b>	<b>xii</b>
<b>DAFTAR PERSAMAAN.....</b>	<b>xiv</b>
<b>DAFTAR LAMPIRAN .....</b>	<b>Error! Bookmark not defined.</b>
<b>BAB I PENDAHULUAN.....</b>	<b>1</b>
1.1. Latar Belakang .....	1
1.2. Rumusan Masalah .....	3
1.3. Batasan Masalah.....	3
1.4. Tujuan.....	4
1.5. Manfaat.....	4
1.6. Sistematika Penulisan.....	5
<b>BAB II TINJAUAN PUSTAKA.....</b>	<b>7</b>
2.1. Kajian Pustaka.....	7
2.2. Sistem .....	13
2.3. Sistem Pendukung Keputusan (SPK).....	14
2.3.1 Tahapan Pengambilan Keputusan .....	15
2.3.2 Karakteristik dan Kapabilitas SPK.....	17
2.3.3 Arsitektur Pemodelan SPK .....	18
2.4. Ketidakpastian .....	19
2.5. <i>Dempster Shafer</i> .....	20
2.6. Stroke.....	21



2.6.1.	Jenis Stroke .....	22
2.6.2.	Faktor Resiko .....	24
2.7.	<i>Data Flow Diagram</i> (DFD).....	28
2.8.	<i>Entity Relationship Diagram</i> (ERD) .....	33
<b>BAB III METODOLOGI PENELITIAN .....</b>		<b>36</b>
3.1.	Studi Literatur.....	37
3.2.	Pengumpulan Data .....	37
3.3.	Analisa Kebutuhan .....	37
3.4.	Desain dan Perancangan Sistem.....	38
3.4.1.	Arsitektur Sistem Pendukung Keputusan .....	39
3.5.	Implementasi Sistem .....	42
3.6.	Pengujian SPK.....	42
3.7.	Pengambilan Kesimpulan.....	43
<b>BAB IV PENGUJIAN DAN ANALISA .....</b>		<b>45</b>
6.1.	Pengujian Fungsionalitas.....	46
6.1.1.	Skenario Pengujian Fungsionalitas .....	46
6.1.2	Analisis Pengujian Fungsionalitas .....	61
6.2	Pengujian Akurasi .....	62
6.2.1.	Skenario Pengujian Akurasi.....	62
6.2.2	Analisis Pengujian Akurasi.....	69
<b>BAB V PENUTUP .....</b>		<b>70</b>
7.1	Kesimpulan.....	70
7.2	Saran.....	70
<b>DAFTAR PUSTAKA .....</b>		<b>71</b>

**DAFTAR GAMBAR**

Gambar 2.1 Diagram Perbandingan Faktor Resiko Deteksi Dini Penyakit Stroke..... 13

Gambar 2.2 Tahapan Perhitungan Metode Naïve Bayes .....**Error! Bookmark not defined.**

Gambar 2.3 Tahapan Perhitungan Metode CF..... **Error! Bookmark not defined.**

Gambar 2.4 Tahapan Perhitungan Metode Dempster Shafer ....**Error! Bookmark not defined.**

Gambar 2.5 Tahapan Pengambilan Keputusan ..... 16

Gambar 2.6 Arsitektur Pemodelan Sistem Pendukung Keputusan..... 19

Gambar 2.7 Diagram Persamaan Penyakit Berdasarkan Gejala **Error! Bookmark not defined.**

Gambar 2.8 Stroke Iskemik ..... 23

Gambar 2.9 Stroke Hemoragik ..... 24

Gambar 2.10 Simbol Terminator ..... 28

Gambar 2.11 Jenis Terminator..... 29

Gambar 2.12 Simbol Proses..... 29

Gambar 2.13 Jenis Proses ..... 30

Gambar 2.14 Simbol Data Store ..... 30

Gambar 2.15 Jenis Data Store..... 31

Gambar 2.16 Simbol Data Flow..... 31

Gambar 2.17 Konsep Data Flow..... 32

Gambar 2.21 Macam-macam Relasi..... 34

Gambar 2.22 Simbol ERD ..... 35

Gambar 3.1 Diagram Alir Metodologi.....36

Gambar 3.2 Arsitektur SPK Deteksi Dini Penyakit Stroke Metode *Dempster Shafer*39

Gambar 3.3 Diagram Blok SPK Deteksi Dini Penyakit Stroke ..... 41

Gambar 4.19 Perancangan Antarmuka Halaman Kelola Data Aturan.....**Error! Bookmark not defined.**

Gambar 4.20 Perancangan Antarmuka Halaman Deteksi.....**Error! Bookmark not defined.**

Gambar 4.21 Perancangan Antarmuka Halaman Riwayat Deteksi . **Error! Bookmark not defined.**

**DAFTAR TABEL**

Tabel 2.1 Kajian Pustaka..... 8

Tabel 2.2 Nilai Densitas untuk setiap Gejala yang Terpilih ..... **Error! Bookmark not defined.**

Tabel 2.3 Aturan kombinasi untuk  $m_3$  ..... **Error! Bookmark not defined.**

Tabel 2.4 Aturan kombinasi untuk  $m_5$  ..... **Error! Bookmark not defined.**

Tabel 2.5 Sorting Nilai Probabilitas Posterior Hasil Perhitungan Naive Bayes ..**Error! Bookmark not defined.**

Tabel 2.6 Tabel Rule dan Nilai CF Pakar ..... **Error! Bookmark not defined.**

Tabel 2.7 Tabel Karakteristik Metode ..... **Error! Bookmark not defined.**

Tabel 4.1 Penjelasan Kasus Uji untuk Pengujian Proses Login.....47

Tabel 4.2 Skenario Pengujian Fungsionalitas Proses Login ..... 47

Tabel 4.3 Penjelasan Kasus Uji untuk Tambah Data Pengguna ..... 48

Tabel 4.4 Skenario Pengujian Fungsionalitas Proses Tambah Data Pengguna ..... 49

Tabel 4.5 Penjelasan Kasus Uji untuk Ubah Data Pengguna..... 49

Tabel 4.6 Skenario Pengujian Fungsionalitas Proses Ubah Data Pengguna..... 50

Tabel 4.7 Penjelasan Kasus Uji untuk Hapus Data Pengguna ..... 51

Tabel 4.8 Skenario Pengujian Fungsionalitas Proses Hapus Data Pengguna ..... 51

Tabel 4.9 Penjelasan Kasus Uji untuk Lihat Data Faktor Resiko ..... 52

Tabel 4.10 Skenario Pengujian Fungsionalitas Proses Lihat Data Faktor Resiko ..... 52

Tabel 4.11 Penjelasan Kasus Uji untuk Lihat Data Tingkat Resiko Stroke..... 53

Tabel 4.12 Skenario Pengujian Fungsionalitas Proses Lihat Data Resiko Stroke ..... 53

Tabel 4.13 Penjelasan Kasus Uji untuk Tambah Data Aturan..... 53

Tabel 4.14 Skenario Pengujian Fungsionalitas Proses Tambah Data Aturan ..... 54

Tabel 4.15 Penjelasan Kasus Uji untuk Ubah Data Aturan ..... 55

Tabel 4.16 Skenario Pengujian Fungsionalitas Proses Ubah Data Aturan ..... 55

Tabel 4.17 Penjelasan Kasus Uji untuk Hapus Data Aturan..... 56

Tabel 4.18 Skenario Pengujian Fungsionalitas Proses Hapus Data Aturan..... 56

Tabel 4.19 Penjelasan Kasus Uji untuk Lihat Data Aturan ..... 57



Tabel 4.20 Skenario Pengujian Fungsionalitas Proses Lihat Data Aturan.....	57
Tabel 4.21 Penjelasan Kasus Uji Proses Deteksi Dini Penyakit Stroke .....	57
Tabel 4.22 Skenario Pengujian Fungsionalitas Proses Deteksi Dini .....	58
Tabel 4.23 Penjelasan Kasus Uji untuk Lihat Hasil Deteksi .....	58
Tabel 4.24 Skenario Pengujian Fungsionalitas Proses Lihat Hasil Deteksi.....	59
Tabel 4.25 Penjelasan Kasus Uji Menampilkan Riwayat Deteksi.....	59
Tabel 4.26 Skenario Pengujian Fungsionalitas Proses Menampilkan Riwayat .....	59
Tabel 4.27 Penjelasan Kasus Uji Menampilkan Detail Riwayat Deteksi .....	60
Tabel 4.28 Skenario Pengujian Fungsionalitas Proses Menampilkan Riwayat .....	60
Tabel 4.29 Penjelasan Kasus Uji Logout .....	61
Tabel 4.30 Skenario Pengujian Fungsionalitas Proses Deteksi Dini .....	61
Tabel 4.31 Tabel Pengujian Akurasi Hasil Deteksi Sistem dengan Pakar.....	63

**DAFTAR PERSAMAAN**

Persamaan 2-1 Teori *Dempster Shafer*.....23

Persamaan 2-2 Persamaan *Plausability*.....23

Persamaan 2-3 Persamaan Kombinasi *Dempster Shafer*.....24

Persamaan 2-4 Persamaan Teorema Bayes.....27

Persamaan 2-5 Persamaan Naïve Bayes.....27

Persamaan 2-6 Rule CF.....30

Persamaan 2-7 Rule *Evidence* dan Hipotesis Tunggal CF.....30

Persamaan 2-8 Rule Kombinasi Rule *Evidence* Berbeda CF.....31

Persamaan 2-9 Persamaan Indeks Massa Tubuh.....41



## BAB I

### PENDAHULUAN

#### 1.1.Latar Belakang

Stroke atau *Cerebrovascular Accident* (CVA) merupakan gangguan fungsi saraf yang disebabkan oleh gangguan aliran darah dalam otak dan menyebabkan gangguan pada aktivitas fungsional [1, p. 60]. Stroke merupakan penyebab kematian ketiga tersering di negara maju, setelah penyakit jantung dan kanker [1, p. 58]. Insidensi tahunan adalah 2 per 1000 populasi [2, p. 89]. Pada data *South East Asian Medical Information Centre* (SEAMIC) diketahui bahwa angka kematian stroke terbesar dan tertinggi terjadi di Indonesia [3, p. 57]. Hal ini menyebabkan penyakit stroke harus diwaspadai. Pola hidup tidak sehat dapat menimbulkan penyakit stroke. Pola hidup yang tidak teratur seperti pola makan yang tidak teratur, kurangnya aktivitas olahraga, jam kerja berlebihan, serta konsumsi makanan cepat saji [1, p. 58]. Pencegahan secara dini melalui pemeriksaan dokter perlu dilakukan karena tingginya angka resiko penyakit stroke. Pendeteksian resiko penyakit stroke mudah ditentukan apabila mengetahui kriteria yang lengkap dan terstruktur. Namun terkadang pendeteksian resiko penyakit stroke sulit ditentukan apabila ada faktor resiko yang tidak terstruktur sehingga dokter dapat mengalami kendala atau ambigu dalam melakukan diagnosa. Untuk mengatasi permasalahan pola semi terstruktur tersebut, maka dapat diselesaikan menggunakan sistem pendukung keputusan (SPK) dengan komputasi cerdas. SPK ini sangat efektif dalam membantu pengambilan keputusan dalam situasi yang semiterstruktur dan tidak terstruktur, dimana tak seorangpun tahu secara pasti bagaimana keputusan seharusnya dibuat [4, pp. 15-16].

Beberapa penelitian terkait penggunaan SPK dapat membantu pengambilan keputusan dengan baik. Penelitian oleh Sholeh dengan judul “Aplikasi Pendukung Keputusan Untuk Deteksi Dini Risiko Penyakit Stroke Menggunakan Logika *Fuzzy* Mamdani Studi Kasus DI RS XYZ” dapat membantu pengambilan keputusan tingkat resiko penyakit stroke [5, p. 1]. Penelitian SPK lainnya dengan menggunakan metode

*Dempster Shafer* dilakukan oleh Sisilia Daeng dengan judul “Sistem Pendukung Keputusan Seleksi Beasiswa Menggunakan Teorema Bayes dan *Dempster Shafer*”. Pada penelitian tersebut metode *Dempster Shafer* digunakan untuk mendukung pengambilan keputusan pemberian beasiswa secara selektif dan tepat sasaran. Metode *Dempster Shafer* mampu melakukan kalkulasi dengan baik dan menghasilkan calon penerima mahasiswa yang efektif dan efisien berdasarkan 4 kriteria, yaitu nilai IPK, jenjang semester, penghasilan orangtua, dan jumlah saudara kandung [6, p. 23].

Penelitian lainnya berjudul “*A Cooperative Watchdog Model Based On Dempster-Shafer For Detecting Misbehaving Vehicles*” melakukan pengawasan pada kendaraan yang tidak dapat berfungsi dengan baik. Terdapat beberapa pengawas kendaraan yang dapat menentukan sifat kendaraan. Namun terkadang setiap pengawas memiliki kesimpulan yang berbeda terhadap sifat kendaraan tersebut. Sistem ini menggunakan metode *Dempster Shafer* dalam membantu menyelesaikan penentuan sifat kendaraan melalui 3 kriteria yaitu *selfish*, *cooperative*, dan *uncertain* dari setiap pengawas kendaraan yang berbeda. Dengan metode *Dempster Shafer* ini maka dapat ditentukan sifat kendaraan termasuk *selfish*, atau *cooperative*. Metode *Dempster Shafer* ini mampu menambah keakuratan deteksi 40% dan mengurangi kesalahan dalam mendeteksi 30% [7, pp. 50,51,54].

Penelitian lain dengan metode *Dempster Shafer* dilakukan oleh Mustika dengan judul “Aplikasi Sistem Pakar Pendeteksian Dan Penanganan Dini Pada Penyakit Sapi Dengan Metode *Dempster-Shafer* Berbasis Web” menjelaskan mengenai pendeteksian pada penyakit sapi menggunakan metode *Dempster Shafer*. Sistem ini melakukan pendeteksian berdasarkan gejala penyakit yang dimasukkan oleh user. Dengan menggunakan metode *Dempster Shafer*, sistem ini mampu melakukan pendeteksian dengan keakuratan 88,89 % [8, p. 157]. Penelitian oleh Elyza dengan judul “Prototype Sistem Pakar untuk Mendeteksi Tingkat Resiko Penyakit Jantung Koroner dengan Metode *Dempster-Shafer*”. Sistem ini melakukan deteksi penyakit jantung koroner menggunakan metode *Dempster Shafer*. Sistem ini akan menghasilkan tingkat resiko jantung koroner berdasarkan 8 faktor resiko, yaitu

kolesterol, gula darah, tekanan darah, BMI, usia, jenis kelamin, batuk, sesak nafas. Penelitian ini menghasilkan keakuratan 100% nilai benar [9, p. 143].

Menurut objek dan uraian yang telah dipaparkan, maka peneliti melakukan penelitian dengan judul “Sistem Pendukung Keputusan Deteksi Dini Penyakit Stroke Menggunakan Metode *Dempster Shafer*”. Penelitian ini menggunakan metode *Dempster Shafer* dalam mendeteksi dini penyakit stroke. Metode *Dempster Shafer* digunakan karena mampu mengatasi permasalahan yang memiliki banyak unsur ketidakpastian kerap kali ditemukan dalam melakukan pendeteksian penyakit. *Dempster-Shafer* merupakan metode kecerdasan buatan, dimana metode ini dianggap lebih mudah dalam mempresentasikan fakta-fakta dan keakuratan data dapat terjaga, karena pada kenyataannya banyak permasalahan yang tidak dapat terselesaikan secara lengkap dan konsisten. Untuk mengatasi ketidakkonsistenan tersebut, maka dapat menggunakan teori *Dempster-Shafer*.

### **1.2. Rumusan Masalah**

Berdasarkan uraian latar belakang di atas, maka dapat dirumuskan permasalahan pada skripsi ini yaitu sebagai berikut :

1. Bagaimana desain dan perancangan SPK deteksi dini penyakit stroke menggunakan metode *Dempster Shafer*.
2. Bagaimana implementasi dari SPK deteksi dini penyakit stroke menggunakan metode *Dempster Shafer*.
3. Bagaimana pengujian dan analisis SPK deteksi dini penyakit stroke menggunakan metode *Dempster Shafer*.

### **1.3. Batasan Masalah**

Berdasarkan latar belakang dan rumusan masalah yang telah disampaikan di atas, penelitian ini mempunyai batasan masalah sebagai berikut :

1. Metode yang digunakan untuk penerapan adalah metode *Dempster Shafer*.
2. Keluaran sistem yang dihasilkan yaitu penyakit stroke dengan tingkat resiko rendah, sedang, atau tinggi sesuai dengan kriteria faktor resiko yang dimasukkan.

3. Terdapat 8 kriteria faktor resiko deteksi dini penyakit stroke yang digunakan dalam penelitian ini yaitu tekanan darah, riwayat fibrilasi atrium, merokok, kolesterol, diabetes, aktivitas fisik, diet, dan riwayat keluarga.
4. Data yang digunakan dalam penelitian ini adalah data kondisi kesehatan masyarakat Kota Malang.
5. Bahasa pemrograman yang digunakan adalah PHP dengan database MySQL.

#### 1.4. Tujuan

Tujuan yang ingin dicapai dalam penelitian ini adalah untuk membangun dan menganalisa SPK deteksi dini penyakit stroke menggunakan metode *Dempster Shafer*.

#### 1.5. Manfaat

Penulisan skripsi ini diharapkan mempunyai manfaat yang baik dan berguna bagi pembaca dan penulis. Adapun manfaat yang diharapkan adalah sebagai berikut :

1. Bagi Penulis
  - a. Penulis dapat memperoleh pengalaman untuk membangun sistem cerdas yang berguna bagi masyarakat.
  - b. Penelitian ini merupakan salah satu bentuk sumbangsih penulis dalam memajukan teknologi.
2. Bagi Pengguna
  - a. Sistem ini dapat membantu untuk memecahkan masalah dalam mendeteksi dini penyakit stroke.
  - b. Pengguna dapat melakukan deteksi dini pada penyakit stroke.
3. Bagi Universitas Brawijaya
  - a. Penelitian ini dapat digunakan sebagai tolak ukur kesiapan mahasiswa untuk terjun di masyarakat.
  - b. Universitas dapat mengetahui kualitas yang dimiliki oleh mahasiswa sehingga dapat digunakan sebagai acuan bagi perguruan tinggi untuk meningkatkan mutu pendidikan selanjutnya.

## 1.6. Sistematika Penulisan

Sistematika penyusunan laporan ditujukan untuk memberikan gambaran dan uraian dari penyusunan skripsi ini secara garis besar yang meliputi beberapa bab, sebagai berikut :

### **BAB I           Pendahuluan**

Bab ini berisi latar belakang, rumusan masalah, tujuan, batasan masalah serta manfaat yang terkait dengan SPK deteksi dini penyakit stroke menggunakan metode *Dempster Shafer*.

### **BAB II           Tinjauan Pustaka**

Bab ini digunakan untuk mendeskripsikan dasar teori yang akan digunakan dan mendukung penelitian SPK deteksi dini penyakit stroke menggunakan metode *Dempster Shafer*. Dasar teori terdiri dari sistem, SPK, ketidakpastian, *Dempster Shafer*, stroke, *Data Flow Diagram* (DFD), dan *Entity Relationship Diagram* (ERD).

### **BAB III         Metodologi Penelitian**

Metodologi penelitian menguraikan metode dan langkah kerja yang dilakukan untuk membangun SPK deteksi dini penyakit stroke. Metodologi penelitian terdiri dari studi literatur, pengumpulan data, analisis dan perancangan sistem SPK, implementasi SPK, pengujian SPK, dan pengambilan kesimpulan.

### **BAB IV         Perancangan**

Menganalisa kebutuhan SPK dan melakukan perancangan SPK deteksi dini penyakit stroke dengan metode *Dempster Shafer*.

### **BAB V         Implementasi**

Mengimplementasikan dan membahas SPK deteksi dini penyakit stroke dengan metode *Dempster Shafer*.

**BAB VI Pengujian dan Analisis**

Pengujian dan analisis mendeskripsikan rencana pengujian serta menganalisis hasil pengujian pada SPK deteksi dini penyakit stroke yang telah diimplementasikan. Pengujian akan dilakukan melalui dua tahap yaitu pengujian fungsional (*blackbox*) dan pengujian akurasi.

**BAB VII Penutup**

Bab ini memuat kesimpulan berdasarkan hasil penelitian SPK deteksi dini penyakit stroke yang telah dilakukan serta memberikan saran yang berguna untuk penelitian berikutnya.



## BAB II

### TINJAUAN PUSTAKA

Bab ini terdiri dari kajian pustaka dan dasar teori yang berhubungan dengan SPK deteksi dini penyakit stroke menggunakan metode *Dempster Shafer*. Kajian pustaka membahas mengenai penelitian yang telah ada dan penelitian yang diusulkan. Penelitian tersebut antara lain dilakukan oleh Sisilia [6], Mustikadewi dkk [8], Mukhlis [10], Omar dkk [7], Sholeh [5], Elyza dkk [9]. Dasar teori membahas teori yang dibutuhkan dalam penyusunan penelitian yang diusulkan. Pada penelitian ini dasar teori yang dibutuhkan antara lain sistem, SPK, ketidakpastian, *Dempster Shafer*, stroke, *Data Flow Diagram* (DFD), dan *Entity Relationship Diagram* (ERD).

#### 2.1. Kajian Pustaka

Berikut merupakan beberapa kajian pustaka yang digunakan penulis berdasarkan penelitian sebelumnya tentang permasalahan yang diangkat. Beberapa penelitian sebelumnya adalah penelitian yang dilakukan oleh Mukhlis dengan judul, “Diagnosa Kemungkinan Pasien Terkena Stroke dengan Menggunakan Metode *Naïve Bayes* dan Metode Jaringan Syaraf Tiruan Berbasis Web” [10]. Sholeh dengan judul “Aplikasi Pendukung Keputusan Untuk Deteksi Dini Risiko Penyakit Stroke Menggunakan Logika *Fuzzy Mamdani* Studi Kasus di RS XYZ” [5]. Sisilia dengan judul “Sistem Pendukung Keputusan Seleksi Beasiswa Menggunakan Teorema Bayes dan *Dempster Shafer*” [6]. Omar dkk dengan judul “*A Cooperative Watchdog Model Based On Dempster-Shafer For Detecting Misbehaving Vehicles*” [7]. Elyza dkk dengan judul “Prototype Sistem Pakar untuk Mendeteksi Tingkat Resiko Penyakit Jantung Koroner dengan Metode *Dempster-Shafer*” [9]. Mustikadewi dkk dengan judul “Aplikasi Sistem Pakar Pendeteksian Dan Penanganan Dini Pada Penyakit Sapi Dengan Metode *Dempster-Shafer* Berbasis Web” [8]. Uraian dari penelitian ditunjukkan pada tabel 2.1 berikut.

Tabel 2.1 Kajian Pustaka

No.	Judul	Objek		Metode Proses	Output
		Input	Kriteria		
1.	Diagnosa Kemungkinan Pasien Terkena Stroke Dengan Menggunakan Metode <i>Naïve Bayes</i> Dan Metode Jaringan Syaraf Tiruan Berbasis Web [10]	Diagnosa Kemungkinan Pasien Terkena Stroke 1. Umur 2. Nadi 3. Jenis Kelamin 4. Penyakit Darah Tinggi 5. Penyakit Diabetes 6. Jantung 7. Darah	Pasien	<p><i>Naïve Bayes</i> :</p> <ol style="list-style-type: none"> <li>Perhitungan nilai probabilitas awal</li> <li>Perhitungan nilai probabilitas <i>prior</i>.</li> <li>Perhitungan nilai probabilitas <i>likelihood</i></li> <li>Perhitungan nilai probabilitas <i>posterior</i></li> <li>Melakukan <i>sorting</i> nilai probabilitas <i>posterior</i>.</li> </ol> <p>Jaringan Saraf Tiruan :</p> <ol style="list-style-type: none"> <li>Inisialisasi penimbang kriteria</li> <li>Menentukan nilai <i>learning rate</i></li> <li>Pengujian selisih target dan output (<i>error</i>)</li> <li>Menghitung pembaruan nilai penimbang kriteria</li> <li>Hitung keluaran JST</li> </ol>	<p>Hasil Penelitian</p> <p>Diagnosa Stroke Pada Pasien</p> <ol style="list-style-type: none"> <li><i>Suspect Stroke</i></li> <li>Tidak <i>Suspect Stroke</i></li> </ol>
2.	Aplikasi Pendukung Keputusan Untuk Deteksi Dini Risiko Penyakit Stroke Menggunakan Logika Fuzzy Mamdani Studi Kasus DIRS XYZ [5]	Deteksi Dini Risiko Penyakit Stroke 1. Tekanan Darah 2. Kadar gula darah 3. Kolesterol total 4. Kolesterol LDL ( <i>Density Lipoprotein</i> ) 5. Usia 6. Asam urat 7. Jenis Kelamin 8. BUN ( <i>Blood Urea Nitrogen</i> ) 9. Kreatinin	Penyakit Stroke	<p>Metode Logika Fuzzy Mamdani:</p> <ol style="list-style-type: none"> <li>Menentukan variabel masukan</li> <li>Pembentukan himpunan fuzzy</li> <li>Membuat <i>rule/aturan</i></li> <li>Komposisi fuzzy (<i>Compositional rule of inference</i>)</li> <li>Defuzzyfikasi (penegasan)</li> </ol>	<p>Diagnosa Status Resiko Stroke</p> <ol style="list-style-type: none"> <li>Rendah</li> <li>Normal</li> <li>Tinggi</li> </ol>
3.	Sistem Pendukung Keputusan	Seleksi Calon Penerima Beasiswa	Seleksi Calon Penerima Beasiswa	Metode Dempster Shafer	Seleksi Calon Penerima

	Beasiswa		
<p>Seleksi Beasiswa Menggunakan Teorema Bayes dan Dempster Shafer [6]</p>	<p>1. Nilai IPK 2. Jenjang Semester 3. Penghasilan Orang Tua 4. Jumlah Saudara Kandung</p>	<p>1. Mencari data kriteria beasiswa 2. Menentukan nilai densitas kriteria antara 0-1 3. Menentukan <i>frame of discernment</i> dalam pengaitan ukuran kepercayaan elemen 4. Menentukan probabilitas densitas (m) 5. Melakukan perhitungan kombinasi densitas setiap kriteria 6. Identifikasi nilai hasil kombinasi tertinggi menjadi variabel keluaran</p> <p>Metode Dempster Shafer: 1. Mendapatkan hasil pengawasan kendaraan dari setiap pengawasan kendaraan yang berbeda 2. Menentukan nilai densitas sifat kendaraan antara 0-1 3. Menentukan <i>frame of discernment</i> dalam pengaitan ukuran kepercayaan elemen 4. Menentukan probabilitas densitas (m) 5. Melakukan perhitungan kombinasi densitas setiap kriteria 6. Identifikasi nilai hasil kombinasi tertinggi menjadi variabel keluaran</p>	<p>Calon Penerima Beasiswa dengan nilai densitas tertinggi.</p>
<p>4. A Cooperative Watchdog Model Based On Dempster-Shafer For Detecting Misbehaving Vehicles [7]</p>	<p>Deteksi Pengawas pada Mobil yang Tidak berfungsi Dengan Baik</p> <p>1. <i>Selfish</i> 2. <i>Cooperative</i> 3. <i>Uncertain</i></p>	<p>1. Mendapatkan hasil pengawasan kendaraan dari setiap pengawasan kendaraan yang berbeda 2. Menentukan nilai densitas sifat kendaraan antara 0-1 3. Menentukan <i>frame of discernment</i> dalam pengaitan ukuran kepercayaan elemen 4. Menentukan probabilitas densitas (m) 5. Melakukan perhitungan kombinasi densitas setiap kriteria 6. Identifikasi nilai hasil kombinasi tertinggi menjadi variabel keluaran</p>	<p>Deteksi Pengawas pada Mobil yang Tidak berfungsi Dengan Baik</p> <p>1. <i>Selfish</i> 2. <i>Cooperative</i> 3. <i>Uncertain</i></p>
<p>5. Prototype Sistem Pakar untuk Mendeteksi Tingkat Resiko Penyakit Jantung Koroner dengan Metode Dempster-Shafer [9]</p>	<p>Deteksi Tingkat Resiko Penyakit Jantung Koroner</p> <p>1. Kolesterol 2. Gula darah 3. Tekanan darah 4. BMI 5. Usia</p>	<p>Metode Dempster Shafer: 1. Mencari data kriteria gejala 2. Menentukan nilai densitas gejala antara 0-1 3. Menentukan <i>frame of discernment</i> dalam pengaitan ukuran kepercayaan elemen</p>	<p>Deteksi Tingkat Resiko Penyakit Jantung Koroner</p> <p>1. Jantung Koroner Berat 2. Jantung Koroner Sedang 3. Jantung Koroner Ringan</p>

	<p>6. Jenis Kelamin 7. Batuk 8. Sesak nafas</p>	<p>6. Jenis Kelamin 7. Batuk 8. Sesak nafas</p>	<p>4. Menentukan probabilitas densitas (m) 5. Melakukan perhitungan kombinasi densitas setiap gejala 6. Identifikasi nilai hasil kombinasi tertinggi menjadi variabel keluaran</p>	<p>Deteksi Dan Penanganan Dini Pada Penyakit Sapi Penyakit pada sapi sesuai dengan hasil kalkulasi Dempster Shafer.</p>
<p>6. Aplikasi Sistem Pakar Pendeteksian Dan Penanganan Dini Pada Penyakit Sapi Dengan Metode Dempster-Shafer Berbasis Web [8]</p>	<p>Deteksi Dan Penanganan Dini Pada Penyakit Sapi Gejala yang ada pada penyakit sapi</p>	<p>Metode Dempster Shafer: 1. Mencari data gejala yang terdapat pada penyakit sapi 2. Menentukan nilai densitas gejala antara 0-1 3. Menentukan <i>frame of discernment</i> dalam pengaitan ukuran kepercayaan elemen 4. Menentukan probabilitas densitas (m) 5. Melakukan perhitungan kombinasi densitas setiap gejala 6. Identifikasi nilai hasil kombinasi tertinggi menjadi variabel keluaran</p>	<p>Metode Dempster Shafer: 1. Mencari data faktor resiko penyakit stroke 2. Menentukan nilai densitas gejala antara 0-1 3. Menentukan <i>frame of discernment</i> dalam pengaitan ukuran kepercayaan elemen 4. Menentukan probabilitas densitas (m) 5. Melakukan perhitungan kombinasi densitas setiap gejala 6. Identifikasi nilai hasil kombinasi tertinggi menjadi variabel keluaran</p>	<p>Deteksi Dini Penyakit Stroke Pada Pasien Terkena</p>
<p>7. Sistem Pendukung Keputusan Deteksi Dini Penyakit Stroke Menggunakan Metode Dempster Shafer [Usulan]</p>	<p>Deteksi Dini Penyakit Stroke 1. Tekanan darah 2. Riwayat fibrilasi atrium 3. Merokok 4. Kolesterol 5. Diabetes 6. Aktivitas fisik 7. Diet 8. Riwayat keluarga stroke</p>	<p>Deteksi Dini Penyakit Stroke 1. Tekanan darah 2. Riwayat fibrilasi atrium 3. Merokok 4. Kolesterol 5. Diabetes 6. Aktivitas fisik 7. Diet 8. Riwayat keluarga stroke</p>	<p>Metode Dempster Shafer: 1. Mencari data faktor resiko penyakit stroke 2. Menentukan nilai densitas gejala antara 0-1 3. Menentukan <i>frame of discernment</i> dalam pengaitan ukuran kepercayaan elemen 4. Menentukan probabilitas densitas (m) 5. Melakukan perhitungan kombinasi densitas setiap gejala 6. Identifikasi nilai hasil kombinasi tertinggi menjadi variabel keluaran</p>	<p>Deteksi Dini Penyakit Stroke Pada Pasien 1. Resiko Tinggi 2. Resiko Sedang 3. Resiko Rendah</p>

Sumber : [10] [5] [6] [7] [9] [8][Usulan]

Hasil penelitian pertama adalah dari M. Khusnul Mukhlis pada tahun 2011 dengan judul *Diagnosa Kemungkinan Pasien Terkena Stroke Dengan Menggunakan Metode Naive Bayes Dan Metode Jaringan Syaraf Tiruan Berbasis Web*. Dalam penelitian tersebut menggunakan metode *Naive Bayes* dan Jaringan Syaraf Tiruan dengan objek yang dipilih adalah penyakit stroke dengan 8 kriteria. Pada jurnal tersebut dikaji mengenai permasalahan diagnosa kemungkinan pasien terkena stroke atau tidak dengan kriteria penilaian yaitu umur, nadi, jenis kelamin, penyakit, darah tinggi, penyakit diabetes, jantung, dan darah. Keakuratan dari sistem ini yaitu 97% [10, p. 1].

Hasil penelitian yang kedua adalah dari Ahmad Fashel Sholeh tahun 2012 dengan judul *Aplikasi Pendukung Keputusan Untuk Deteksi Dini Risiko Penyakit Stroke Menggunakan Logika Fuzzy Mamdani : Studi Kasus Di RS XYZ*. Dalam penelitiannya, mereka menggunakan metode Logika *Fuzzy Mamdani* dengan objek deteksi dini penyakit stroke. Dimana berdasarkan hasil kalkulasi perhitungan berdasarkan metode Logika *Fuzzy Mamdani*, maka akan diputuskan kategori tingkat resiko penyakit stroke yang diderita pasien, berdasarkan 9 kriteria, yaitu tekanan darah, kadar gula darah, kolesterol total, kolesterol LDL (*Low Density Lipoprotein*), usia, asam urat, jenis kelamin, BUN (*Blood Urea Nitrogen*), dan kreatinin. Aplikasi ini memiliki akurasi total sebesar 82,98% [5, p. 1].

Hasil penelitian yang ketiga adalah dari Sisilia Daeng tahun 2014 dengan judul *Sistem Pendukung Keputusan Seleksi Beasiswa Menggunakan Teorema Bayes dan Dempster-Shafer*. Dalam penelitiannya, menggunakan metode *Dempster Shafer* dengan objek calon penerima beasiswa. Berdasarkan hasil kalkulasi maka akan ditentukan calon penerima beasiswa secara efektif dan tepat sasaran berdasarkan 4 faktor resiko, yaitu nilai IPK, jenjang semester, penghasilan orangtua, dan jumlah saudara kandung. SPK ini memiliki keakurasian. Pada penelitian ini membandingkan teorema bayes dan *Dempster Shafer* dalam mengambil keputusan seleksi beasiswa, hasilnya kedua metode ini menghasilkan *output* yang sama [6, p. 23].

Hasil penelitian yang keempat adalah dari Omar dkk tahun 2014 dengan judul *A Cooperative Watchdog Model Based On Dempster-Shafer For Detecting*

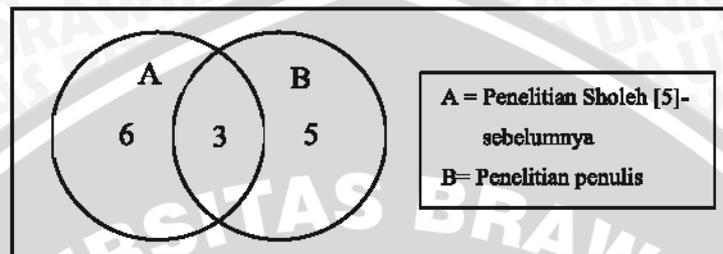
*Misbehaving Vehicles*. Penelitian ini melakukan pengawasan pada kendaraan yang tidak dapat berfungsi dengan baik. Terdapat beberapa pengawas kendaraan yang dapat menentukan sifat kendaraan. Namun terkadang setiap pengawas memiliki kesimpulan yang berbeda terhadap sifat kendaraan tersebut. Sistem ini menggunakan metode *Dempster Shafer* dalam membantu menyelesaikan penentuan sifat kendaraan melalui 3 kriteria yaitu *selfish*, *cooperative*, dan *uncertain* dari setiap pengawas kendaraan yang berbeda. Dengan metode *Dempster Shafer* ini maka dapat ditentukan sifat kendaraan termasuk *selfish* atau *cooperative*. Metode *Dempster Shafer* ini mampu menambah keakuratan deteksi 40% dan mengurangi kesalahan dalam mendeteksi 30% [7, pp. 50,51,54]

Hasil penelitian yang kelima adalah dari Elyza dkk tahun 2013 dengan judul *Prototype Sistem Pakar untuk Mendeteksi Tingkat Resiko Penyakit Jantung Koroner dengan Metode Dempster-Shafer*. Dalam penelitiannya, menggunakan metode *Dempster Shafer* dengan objek deteksi penyakit jantung koroner. Berdasarkan hasil kalkulasi maka akan ditentukan tingkat resiko penyakit jantung koroner berdasarkan 8 faktor resiko. Faktor resiko tersebut yaitu kolesterol, gula darah, tekanan darah, BMI, usia, jenis kelamin, batuk, sesak nafas. Penelitian ini menghasilkan keakuratan 100% nilai benar [9, p. 133].

Hasil penelitian yang keenam adalah dari Mustikadewi tahun 2013 dengan judul *Aplikasi Sistem Pakar Pendeteksian Dan Penanganan Dini Pada Penyakit Sapi Dengan Metode Dempster-Shafer Berbasis Web*. Penelitian ini menggunakan metode *Dempster Shafer* dengan objek deteksi penyakit pada sapi. Gejala yang digunakan untuk deteksi sebanyak 167 gejala dan penyakit yang dapat dideteksi melalui sistem ini sebanyak 26 penyakit. Berdasarkan hasil kalkulasi maka akan ditentukan penyakit berdasarkan dengan gejala yang dimasukkan. Penelitian ini menghasilkan akurasi 88,89% [8, p. 157].

Pada penelitian sebelumnya membahas penelitian mengenai SPK untuk deteksi dini risiko penyakit stroke menggunakan logika *fuzzy* mamdani yang dilakukan oleh Sholeh dengan judul “Aplikasi Pendukung Keputusan Untuk Deteksi Dini Risiko Penyakit Stroke Menggunakan Logika *Fuzzy* Mamdani : Studi Kasus Di

RS XYZ” [5, p. 1]. Perbedaan yang dibuat peneliti dengan penelitian ini adalah dari kriteria faktor resiko penyakit stroke yang dideteksi oleh sistem dan metode yang digunakan pada SPK. Perbedaan kriteria faktor resiko dapat dilihat pada Gambar 2.1 berikut :



Gambar 2.1 Diagram Perbandingan Faktor Resiko Deteksi Dini Penyakit Stroke

SPK pada penelitian ini akan menerima 8 kriteria yaitu berupa faktor resiko penyakit stroke. Faktor resiko tersebut adalah :

1. Tekanan darah
2. Riwayat fibrilasi atrium
3. Merokok
4. Kolesterol
5. Diabetes
6. Aktivitas fisik
7. Diet
8. Riwayat keluarga

Berdasarkan penjelasan Gambar 2.1 diatas, terdapat tiga gejala yang sama pada penelitian sebelumnya yaitu tekanan darah, diabetes, dan kolesterol. Sedangkan lima gejala yang berbeda adalah usia, asam urat, jenis kelamin, BUN (*Blood Urea Nitrogen*), dan kreatinin. Dengan kriteria *input* yang berbeda tentunya akan mempengaruhi dalam penentuan *output* tingkat resiko penyakit stroke.

## 2.2. Sistem

Sistem adalah kombinasi beberapa komponen yang dapat melakukan pekerjaan tertentu secara bersama-sama. Komponen tersebut dapat berdiri sendiri maupun

komponen yang saling berhubungan antar satu komponen dengan lainnya. Komponen utama dari suatu sistem yaitu [19, p. 1]:

1. *Input* atau masukan adalah komponen yang dapat berupa data atau informasi.
2. Proses adalah operasi atau perkembangan yang berlangsung secara menerus ditandai dengan perubahan kecil yang berurutan dengan cara relatif tetap dan menuju ke suatu keadaan atau hasil tertentu.
3. *Output* atau keluaran adalah hasil perubahan yang telah dilakukan terhadap data atau informasi yang diberikan pada *input*.

### 2.3. Sistem Pendukung Keputusan (SPK)

Sistem adalah kombinasi beberapa komponen yang dapat melakukan pekerjaan tertentu secara bersama-sama [19, p. 1]. Terdapat beberapa jenis sistem yang sering digunakan yaitu sistem informasi, SPK, dan sistem pakar. Sistem informasi adalah sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian yang mendukung fungsi operasi organisasi yang bersifat manajerial [20, p. 46]. Sistem informasi digunakan untuk menyelesaikan permasalahan yang terstruktur [21, p. 16]. SPK adalah bagian dari sistem informasi berbasis komputer yang dipakai untuk mendukung pengambilan keputusan menggunakan metode basis pengetahuan tertentu guna untuk mengambil suatu keputusan yang bersifat semi terstruktur dan tidak terstruktur [4, p. 15].

Pengambilan keputusan yang bersifat tidak terstruktur juga dapat menggunakan alat pendukung keputusan lanjutan, salah satunya yaitu sistem pakar [21, p. 17]. SPK tidak harus membutuhkan pakar dalam mengambil suatu keputusan, namun hal ini berbeda dengan sistem pakar. Sistem pakar adalah salah satu bagian kecerdasan buatan yang mengandung pengetahuan pakar yang digunakan untuk memecahkan permasalahan yang biasanya dipecahkan oleh pakar [15, p. 13]. Sistem pakar juga dapat memberikan solusi dari permasalahan (*treatment*) kepada *user*.

SPK secara khusus dikembangkan untuk mendukung solusi dari permasalahan yang semi terstruktur dan tidak terstruktur untuk meningkatkan kualitas pengambilan keputusan. Dengan demikian dapat ditarik satu definisi tentang SPK yaitu suatu

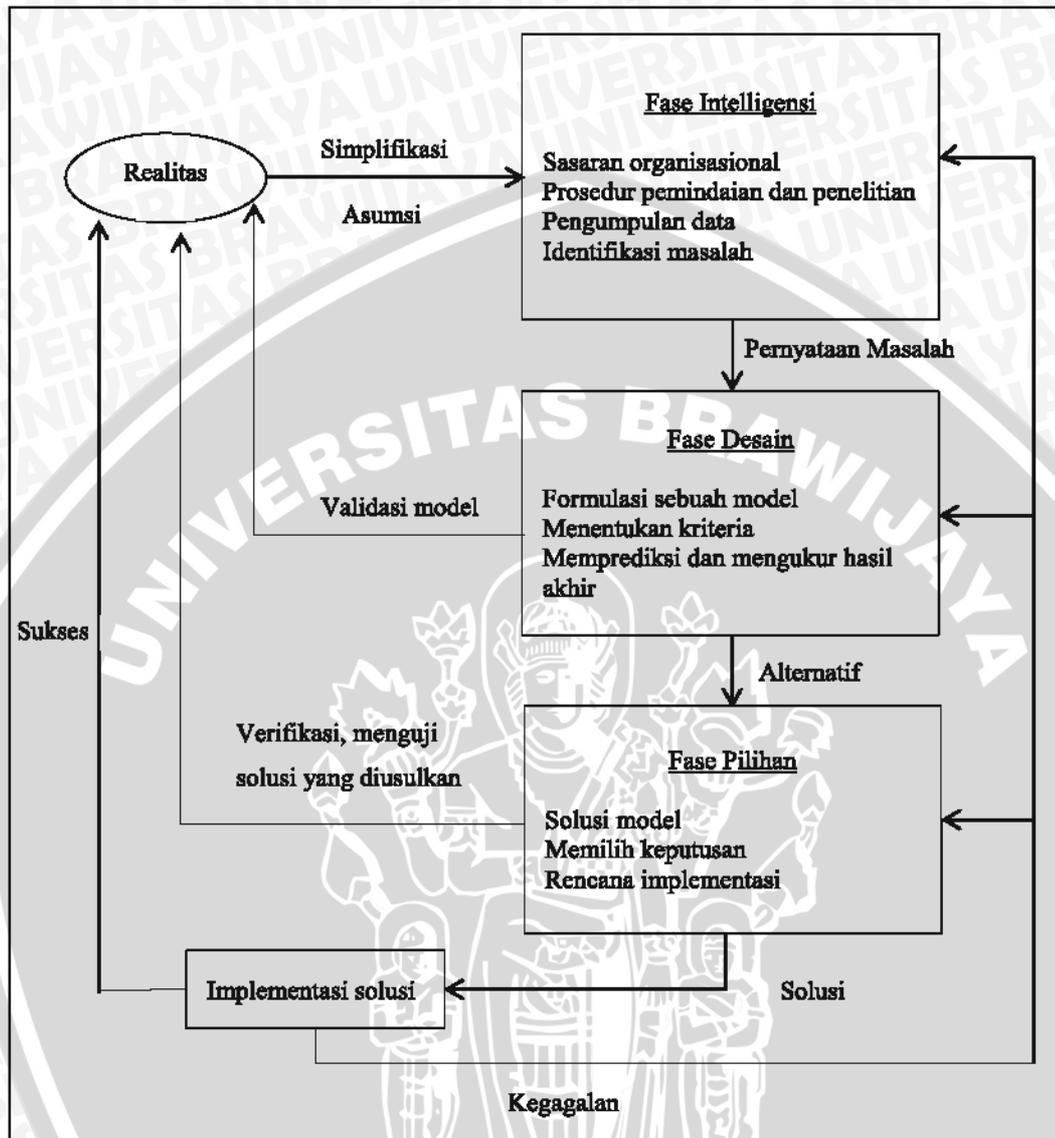
sistem berbasis komputer yang adaptif, fleksibel, dan interaktif yang digunakan untuk memecahkan masalah-masalah semi terstruktur dan tidak terstruktur sehingga meningkatkan nilai keputusan yang diambil [21, pp. 136-138].

SPK memberikan dukungan langsung pada permasalahan dengan menyediakan keputusan dan menekankan pada efektifitas pengambilan keputusan dalam upaya untuk menghasilkan keputusan yang lebih baik. Pada sistem ini yang memegang peranan penting adalah pengambil keputusan, karena sistem hanya menyediakan keputusan sedangkan keputusan akhir tetap diambil oleh pengambil keputusan. SPK biasanya dibangun untuk mendukung solusi atas suatu masalah, yang seperti itu disebut aplikasi SPK [4, p. 16].

### **2.3.1 Tahapan Pengambilan Keputusan**

Pengambilan keputusan adalah pemilihan keputusan untuk mencapai satu atau beberapa tujuan yang telah diterapkan [22]. Pada dasarnya pengambilan keputusan dilakukan melalui mekanisme tertentu dengan harapan akan menghasilkan suatu keputusan yang terbaik.

Menurut Turban, terdapat empat fase utama dalam proses pengambilan keputusan, yaitu inteligensi, desain, pilihan dan implementasi [22, p. 64]. Fase ini saling berkaitan sehingga sangat berpengaruh dalam proses pengambilan keputusan sehingga sesuai dengan yang diharapkan. Fase ini tentunya sangat membantu dalam memutuskan suatu keputusan yang tidak terstruktur. Fase proses pengambilan keputusan akan dijelaskan melalui Gambar 2.5 berikut ini.



Gambar 2.2 Tahapan Pengambilan Keputusan  
 Sumber : [21, p. 65]

Saat melakukan pemodelan dalam pembangunan SPK, dilakukan langkah sebagai berikut [21, pp. 67-93]:

1. Intelegensi/Pemahaman (*Intelligence*)

Tahap ini menentukan, mencari prosedur, mengumpulkan data, dan identifikasi permasalahan yang dibutuhkan berkaitan dengan persoalan dalam keputusan yang akan diambil.

2. Perancangan (*Design*)

Tahap ini merumuskan model yang akan digunakan dan menentukan kriteria-kriteria yang dibutuhkan. Tahapan ini mempresentasikan model sistem yang akan dibangun berdasarkan pada asumsi yang telah ditetapkan.

3. Pemilihan (*Choice*)

Tahap ini merupakan tahap pengambilan keputusan. Pada tahap ini dilakukan pengambilan suatu keputusan yang nyata dan diambil suatu komitmen untuk mengikuti suatu tindakan tertentu.

4. Implementasi (*Implementation*)

Tahap ini merupakan melaksanakan dan merealisasikan dari hasil keputusan yang telah dipilih agar dapat menyelesaikan permasalahan yang telah diidentifikasi sebelumnya.

### 2.3.2 Karakteristik dan Kapabilitas SPK

Karakteristik dan kapabilitas kunci dari SPK menurut Turban adalah [21, pp. 140-143]:

1. Dukungan untuk pengambil keputusan pada situasi semi terstruktur dan tidak terstruktur.
2. Dukungan untuk individu dan kelompok.
3. Dukungan untuk keputusan individu dan sekuensial.
4. Dukungan di semua fase proses pengambilan keputusan (inteligensi, desain, pilihan, dan implementasi).
5. Dukungan di berbagai proses dan gaya pengambilan keputusan.
6. Peningkatan terhadap keefektifan pengambilan keputusan ketimbang pada efisiensinya.
7. Kontrol penuh oleh pengambil keputusan terhadap semua proses pengambilan keputusan.

8. Akses disediakan untuk berbagai sumber data, format, dan tipe.
9. Dapat digunakan sebagai alat *standalone* yang digunakan oleh seorang pengambil keputusan pada satu lokasi atau didistribusikan di lain tempat.

### 2.3.3 Arsitektur Pemodelan SPK

Aplikasi SPK bisa terdiri dari beberapa subsistem, yaitu [4, pp. 25-26]:

#### 1. Subsistem Manajemen Data

Subsistem manajemen data memasukkan satu *database* yang berisi data yang relevan untuk suatu situasi dan dikelola oleh perangkat lunak yang disebut sistem manajemen database (DBMS). Subsistem manajemen data bisa diinterkoneksi dengan data yang relevan dengan pengambilan keputusan.

#### 2. Subsistem Manajemen Model

Merupakan paket perangkat lunak yang memasukkan model keuangan, statistik, ilmu manajemen, atau model kuantitatif lainnya yang dapat memberikan kapabilitas analitik dan manajemen perangkat lunak yang tepat. Bahasa pemodelan untuk membangun model-model kustom juga dimasukkan. Perangkat lunak itu sering disebut sistem manajemen basis model (*MBMS/ Model Based Management System*). Komponen tersebut bisa dikoneksikan ke penyimpanan korporat atau eksternal yang ada pada model.

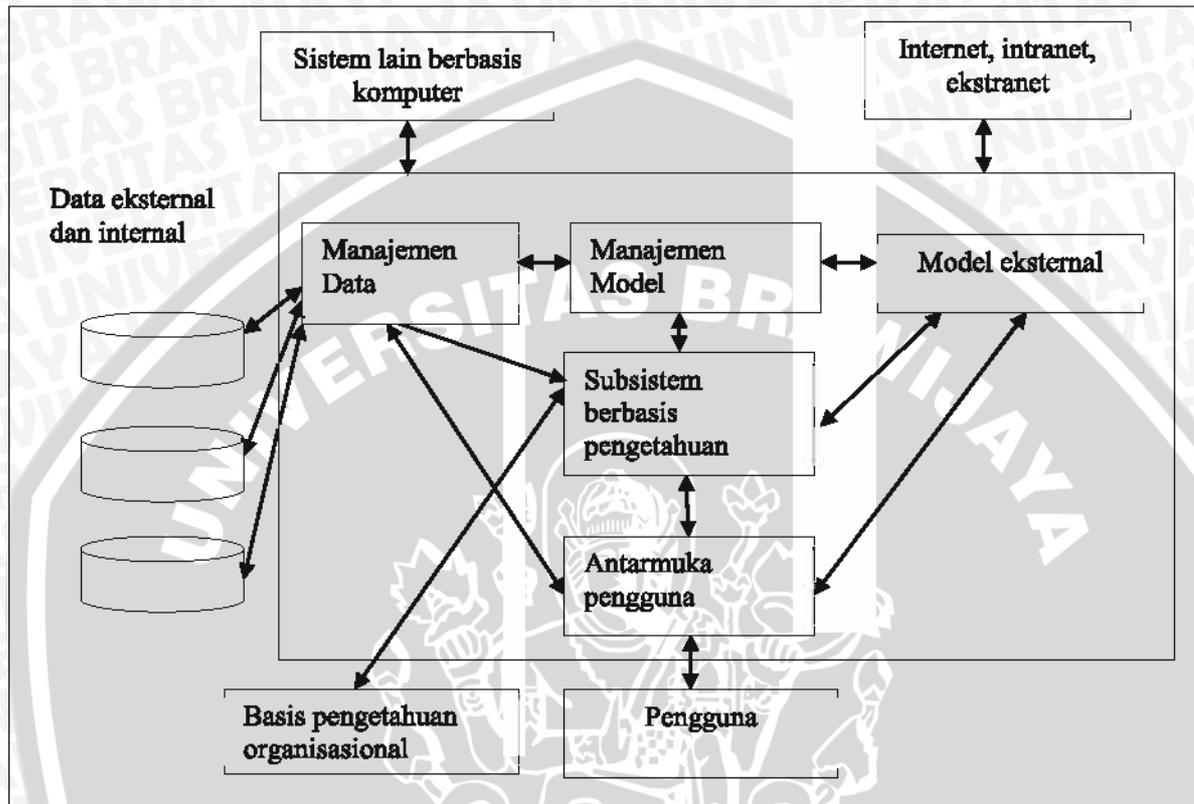
#### 3. Subsistem Antarmuka Pengguna

Pengguna berkomunikasi dengan dan memerintahkan SPK melalui subsistem tersebut. Pengguna adalah bagian yang dipertimbangkan dari sistem. Para peneliti menegaskan bahwa beberapa kontribusi unik dari SPK berasal dari interaksi yang intensif antara komputer dan pembuat keputusan.

#### 4. Subsistem Manajemen Berbasis Pengetahuan

Subsistem tersebut mendukung semua subsistem lain atau bertindak langsung sebagai suatu komponen independen dan bersifat opsional. Selain memberikan inteligensi untuk memperbesar pengetahuan si pengambil keputusan, subsistem tersebut bisa diinterkoneksi dengan repositori pengetahuan (bagian dari sistem

manajemen pengetahuan), yang kadang-kadang disebut basis pengetahuan organisasional.



Gambar 2.3 Arsitektur Pemodelan Sistem Pendukung Keputusan  
Sumber : [4, p. 26]

#### 2.4. Ketidakpastian

Jika sistem kecerdasan buatan yang dikembangkan memiliki pengetahuan yang lengkap tentang permasalahan yang akan ditangani, maka sistem tersebut dapat dengan mudah memberikan solusi dengan menggunakan pendekatan logika dengan model yang sangat lengkap dan konsisten. Namun pada kenyataannya, banyak masalah di dunia ini yang tidak dapat dimodelkan secara lengkap dan konsisten [16, p. 91]. Sistem hampir tidak pernah dapat mengakses seluruh fakta yang ada dalam lingkungan permasalahan yang akan ditanganinya, sehingga sistem harus bekerja dalam ketidakpastian dan kesamaran. Untuk itu, sistem harus menggunakan teknik-

teknik khusus yang dapat menangani ketidakpastian dan kesamaran dalam menyelesaikan permasalahan yang ditanganinya [11, p. 30].

Terdapat tiga teknik yang dapat digunakan untuk menangani ketidakpastian dan kesamaran pengetahuan, yaitu :

1. Teknik Probabilitas, yang dikembangkan dengan memanfaatkan teorema Bayes yang menyajikan hubungan sebab akibat yang terjadi diantara *evidence* yang ada. Pendekatan alternatif lainnya yang dapat digunakan adalah teori *dempster shafer*.
2. Faktor Ketidakpastian, merupakan teknik penalaran tertua, yang digunakan pada sistem MCYIN. Teknik ini bersifat semi probabilitas, karena tidak sepenuhnya menggunakan notasi probabilitas.
3. Logika Fuzzy, merupakan teknik baru yang diperkenalkan oleh Zadeh. Setiap variabel dalam teknik ini memiliki rentang nilai tertentu, yang akan digunakan untuk menghitung nilai fungsi keanggotaannya.

**2.5. Dempster Shafer**

Ada berbagai macam penalaran dengan model yang lengkap dan sangat konsisten, tetapi pada kenyataannya banyak permasalahan yang tidak dapat terselesaikan secara lengkap dan konsisten [16, p. 91]. Untuk mengatasi ketidakconsistenan tersebut maka dapat menggunakan penalaran dengan teori *Dempster-Shafer*.

Secara umum teori *Dempster-Shafer* ditulis dalam suatu interval seperti pada Persamaan 2-1 dan 2-2 [16, pp. 102-103]. Persamaan *Dempster Shafer* dapat dituliskan pada persamaan 2-1 berikut:

$$[Belief, Plausibility] \dots\dots\dots (2-1)$$

- *Belief* (Bel) adalah ukuran kekuatan *evidence* dalam mendukung suatu himpunan proposisi. Jika bernilai 0 maka mengindikasikan bahwa tidak ada *evidence*, dan jika bernilai 1 menunjukkan adanya kepastian.
- Persamaan *Plausibility* (Pl) dituliskan sebagai berikut:

$$Pl(s) = 1 - Bel (\neg s) \dots\dots\dots (2-2)$$

*Plausibility* juga bernilai 0 sampai 1. Jika yakin akan  $\neg s$ , maka dapat dikatakan bahwa  $Bel(\neg s) = 1$ , dan  $Pl(\neg s) = 0$ .

Pada teori *Dempster-Shafer* dikenal adanya *frame of discernment* yang dinotasikan dengan  $\theta$ . Frame ini merupakan semesta pembicaraan dari sekumpulan hipotesis. Tujuannya adalah mengaitkan ukuran kepercayaan elemen-elemen  $\theta$ . Tidak semua *evidence* secara langsung mendukung tiap-tiap elemen. Untuk itu perlu adanya probabilitas fungsi densitas ( $m$ ). Nilai  $m$  tidak hanya mendefinisikan elemen-elemen  $\theta$  saja, namun juga semua subsetnya. Sehingga jika  $\theta$  berisi  $n$  elemen, maka subset  $\theta$  adalah  $2^n$ . Jumlah semua  $m$  dalam subset  $\theta$  sama dengan 1. Apabila tidak ada informasi apapun untuk memilih hipotesis, maka nilai  $m\{\theta\} = 1,0$  [16, p. 103].

Apabila diketahui  $X$  adalah subset dari  $\theta$ , dengan  $m_1$  sebagai fungsi densitasnya, dan  $Y$  juga merupakan subset dari  $\theta$  dengan  $m_2$  sebagai fungsi densitasnya, maka dapat dibentuk fungsi kombinasi  $m_1$  dan  $m_2$  sebagai  $m_3$  sehingga didapatkan persamaan kombinasi *Dempster Shafer* sebagai berikut [16, p. 103]:

$$m_3(Z) = \frac{\sum_{X \cap Y = Z} m_1(X) \cdot m_2(Y)}{1 - \sum_{X \cap Y = \emptyset} m_1(X) \cdot m_2(Y)} \dots \dots \dots (2-3)$$

Keterangan :

$m$  = Nilai Densitas

$XYZ$  = Himpunan *Evidence*

$\emptyset$  = Himpunan Kosong

**2.6. Stroke**

Stroke adalah gangguan fungsi saraf yang disebabkan oleh gangguan aliran darah dalam otak yang dapat timbul secara mendadak dalam beberapa detik atau secara cepat dalam beberapa jam dengan gejala atau tanda-tanda sesuai dengan daerah yang terganggu [1, p. 60]. Stroke juga merupakan sindrom yang terdiri dari tanda dan atau gejala hilangnya fungsi sistem saraf pusat fokal (global) yang berkembang cepat dalam detik atau menit. Gejala ini berlangsung lebih dari 24 jam atau menyebabkan kematian [2, p. 89].



Menurut Neil F. Gordon : stroke adalah gangguan potensial yang fatal pada suplai darah bagian otak. Tidak ada satupun bagian tubuh manusia yang dapat bertahan bila terdapat gangguan suplai darah dalam waktu relatif lama sebab darah sangat dibutuhkan dalam kehidupan, terutama oksigen pengangkut bahan makanan yang dibutuhkan pada otak dan otak adalah pusat kontrol sistem tubuh termasuk perintah dari semua gerakan fisik [1, p. 60].

Dengan kata lain, stroke merupakan manifestasi keadaan pembuluh darah cerebral yang tidak sehat sehingga bisa disebut juga "*cerebral arterial disease*" atau "*cerebrovascular disease*". Cedera dapat disebabkan oleh sumbatan bekuan darah, penyempitan pembuluh darah, sumbatan dan penyempitan atau pecahnya pembuluh darah, semua ini menyebabkan kurangnya pasokan darah yang memadai [1, p. 60].

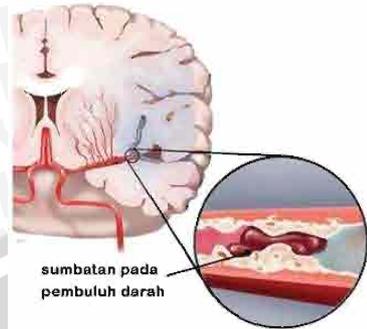
Stroke seringkali terjadi pada orang-orang golongan usia diatas 50 tahun, tetapi mungkin saja terjadi pada usia muda yang seringkali disebabkan karena adanya kelainan jantung yang mengakibatkan timbulnya embolisasi [1, p. 60].

### **2.6.1. Jenis Stroke**

Secara umum, stroke terdiri dari dua jenis yakni iskemik (bukan pendarahan) dan hemoragik (perdarahan). Berikut merupakan penjelasan dari kedua jenis stroke tersebut :

#### **1. Stroke Iskemik**

Stroke iskemik merupakan stroke yang disebabkan sumbatan oleh pembekuan darah, penyempitan sebuah arteri, atau beberapa arteri yang mengarah ke otak, atau *embolus* (kotoran) yang terlepas dari jantung atau arteri *ekstrakranial* (arteri yang berada di luar tengkorak) yang menyebabkan sumbatan di satu atau beberapa arteri intrakrani (arteri yang berada di dalam tengkorak). Pada orang berusia lanjut lebih dari 65 tahun, penyumbatan atau penyempitan pada pembuluh darah otak dapat disebabkan oleh *aterosklerosis* (mengerasnya arteri) [1, p. 61].



Gambar 2.4 Stroke Iskemik  
Sumber : [1, p. 61]

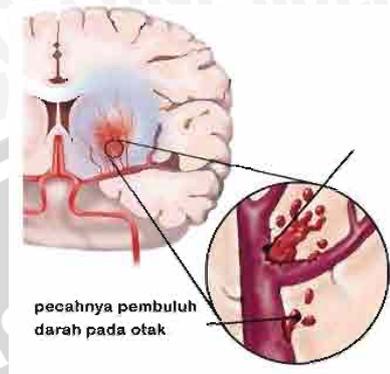
Sebagian stroke iskemik terjadi di *hemisfer* otak, meskipun sebagian terjadi di serebelum (otak kecil) atau batang otak. Beberapa stroke iskemik di *hemisfer* tampaknya bersifat ringan. Stroke iskemik di hemisfer ini *asimtomatik* (tak bergejala) atau hanya menimbulkan kecanggungan, kelemahan ringan, atau masalah daya ingat. Namun stroke ringan ganda dan berulang dapat juga menimbulkan cacat berat, penurunan kognitif dan demensia pada penderita. Hampir sebagian besar penderita stroke mengalami jenis stroke demikian sekitar 83%. Serangan muncul secara tiba-tiba dan langsung menyebabkan penderita tidak sadarkan diri [1, p. 62].

## 2. Stroke Hemoragik

Stroke hemoragik disebabkan oleh perdarahan ke dalam jaringan otak (disebut hemoragia intraserebrum atau hemantum intrserebrum) atau ke dalam ruangan *subaraknoid* yaitu ruang sempit antara permukaan otak dan lapisan jaringan yang menutupi otak. Stroke ini adalah jenis stroke yang mematikan tetapi relatif hanya menyusun sebagian kecil bagi stroke [1, p. 62].

Perdarahan dari sebuah arteri intraktanium biasanya disebabkan oleh aneurisma yang pecah atau karena suatu penyakit. Penyakit yang menyebabkan dinding arteri menipis dan rapuh adalah penyebab tersering perdarahan intraserebrum. Penyakit semacam ini adalah hipertensi (dimana terjadi pengendapan protein di dinding arteri-arteri kecil di otak). Jika seseorang mengalami perdarahan intraserebrum, darah

dipaksa masuk ke dalam jaringan otak, merusak *neuron* sehingga bagian otak yang terkena tidak dapat berfungsi dengan baik [1, p. 63].



Gambar 2.5 Stroke Hemoragik  
Sumber : [1, p. 63]

Stroke hemoragik memiliki dampak yang sangat mencelakakan, karena cairan yang mengelilingi otak (*cerebrospinal*) akan mengalir mengelilingi otak dan menyebabkan pembuluh darah di sekitarnya menjadi kejang sehingga menyumbat pasokan darah ke otak. Karena itulah, *subarachnoid hemorrhage* dapat meninggalkan dampak kelumpuhan yang sangat luas, bahkan resiko kematiannya 50% [23, p. 14]. Kadang satu-satunya gejala perdarahan adalah nyeri kepala, tetapi bila diabaikan gejala ini dapat berakibat fatal. Nyeri kepala khas pada perdarahan timbul mendadak, parah, dan tanpa sebab yang jelas. Nyeri kepala ini dapat disertai oleh muntah, kaku leher, atau kehilangan kesadaran sementara. [1, p. 63].

#### 2.6.2. Faktor Resiko

Semua faktor yang menentukan timbulnya manifestasi stroke dikenal sebagai faktor resiko stroke. Terdapat dua jenis faktor resiko pada penyakit stroke yaitu :

##### 1. Faktor resiko tak terkontrol

Faktor resiko tak terkontrol merupakan faktor resiko yang tidak dapat dikendalikan dan tidak dapat diubah. Faktor resiko tersebut adalah umur, jenis kelamin, keturunan stroke dalam keluarga, dan ras/etnik [23, pp. 31-33].

##### 2. Faktor resiko terkontrol

Faktor resiko terkendali merupakan faktor resiko yang dapat diubah. Hal ini berarti faktor resiko ini dapat diobati atau dikendalikan dengan tujuan untuk mengurangi resiko penyakit stroke. Faktor ini adalah tekanan darah, diabetes, kolesterol, merokok, alkohol, diet yang tidak sehat, aktivitas fisik, dan riwayat fibrilasi atrium [23, pp. 33-39].

Semua faktor tersebut sangat berpengaruh dalam menentukan timbulnya penyakit stroke. Berdasarkan *Score Risk ScoreCard* pada NSA (*National Stroke Association*) berikut merupakan faktor resiko yang paling berpengaruh dalam menentukan penyakit stroke [24]:

1. Tekanan darah

Tekanan darah merupakan faktor resiko yang potensial. Penentuan tekanan darah dapat berdasarkan sistolik dan diastolik pada pasien beresiko stroke. Penderita hipertensi (tekanan darah tinggi) memiliki faktor resiko stroke empat hingga enam kali lipat dibandingkan dengan orang yang tanpa hipertensi dan sekitar 40 hingga 90 persen pasien stroke ternyata menderita hipertensi sebelum stroke [23, p. 34]. Tekanan darah tinggi (Hipertensi) dapat mengakibatkan pecahnya maupun menyempitnya pembuluh darah otak. Apabila pembuluh darah otak pecah, maka timbullah perdarahan otak dan apabila pembuluh darah otak menyempit, maka aliran darah ke otak akan terganggu dan sel-sel otak akan mengalami kematian [25, p. 43].

2. Riwayat fibrilasi atrium

Riwayat fibrilasi atrium adalah jenis denyut jantung tidak teratur. Meskipun dapat terjadi pada semua usia, hal ini lebih sering terjadi pada orang 65 tahun dan lebih tua. Riwayat fibrilasi atrium lebih sering terjadi pada orang dengan tekanan darah tinggi, penyakit jantung, atau diabetes. Riwayat fibrilasi atrium menimbulkan risiko stroke karena memungkinkan darah ke dalam hati. Ketika darah, cenderung membentuk gumpalan yang kemudian dapat dibawa ke otak, menyebabkan stroke [24].

3. Merokok

Merokok merupakan faktor resiko stroke yang sebenarnya dapat diubah. Perokok berat memiliki resiko yang lebih besar dibandingkan perokok ringan. Merokok hampir melipatgandakan resiko stroke iskemik dan dapat juga meningkatkan resiko subarahnoid hemoragik hingga 3,5%. Merokok adalah penyebab nyata kejadian stroke yang lebih banyak terjadi pada usia dewasa muda dibandingkan usia tengah baya atau lebih tua [23, p. 37]. Merokok merupakan faktor resiko utama untuk terjadinya infark jantung [25, p. 44]. Pada pasien perokok, kerusakan yang diakibatkan stroke jauh lebih parah karena dinding bagian dalam (*endothelial*) pada sistem pembuluh darah otak biasanya sudah melemah. Ini dapat menyebabkan kerusakan yang lebih besar lagi pada otak sebagai akibat bila terjadi stroke tahap kedua [23, p. 37].

#### 4. Kolesterol

Penelitian menunjukkan bahwa makanan kaya lemak jenuh dan kolesterol dapat meningkatkan kadar kolesterol dalam tubuh dan berpengaruh pada resiko penebalan pembuluh [23, p. 36]. Meningginya angka kolesterol dalam darah, terutama *low density lipoprotein* (LDL), merupakan faktor resiko penting untuk terjadinya arteriosclerosis (menebalnya dinding pembuluh darah yang kemudian diikuti penurunan elastisitas pembuluh darah) [25, p. 43].

#### 5. Diabetes

Penderita diabetes memiliki tingkat resiko tiga kali lipat terkena stroke dan mencapai tingkat tertinggi pada usia 50-60 tahun. Setelah itu, resiko tersebut akan menurun [23, p. 36]. Diabetes mampu menebalkan dinding pembuluh darah otak yang berukuran besar. Menebalnya dinding pembuluh darah otak akan menyempitkan diameter pembuluh darah tadi dan penyempitan tersebut akan mengganggu kelancaran aliran ke otak, yang pada akhirnya akan menyebabkan infark sel-sel otak [24].

#### 6. Aktivitas Fisik

Aktivitas fisik meliputi apa saja yang membuat tubuh Anda bergerak. Sebuah penelitian baru menunjukkan bahwa orang yang berolahraga lima kali atau lebih

per minggu memiliki risiko stroke berkurang. Rekomendasi terbaru dari CDC pada Aktivitas Fisik Pedoman untuk orang dewasa adalah sebagai berikut :

- *Moderate Activity*. 150 menit aktivitas intensitas sedang per minggu (jalan cepat, aerobik air, bersepeda) dan 2 atau lebih hari seminggu latihan beban.
- *Vigorous Activity*. atau 75 menit aktivitas kuat intensitas per minggu (jogging, lari, berenang ) dan 2 atau lebih hari seminggu latihan latihan beban.
- *Break it up*. Jika tidak dapat memasukkan kegiatan 30 menit dalam satu waktu, memecahnya menjadi segmen 10 menit . Ingat bahwa semua aktivitas fisik penting.
- *Make physical activity fun*. Berjalan atau bersepeda dengan seorang teman, mengambil kelas tari, dan mencoba waktu yang berbeda dari hari.

Aktivitas fisik secara teratur akan meningkatkan kesehatan Anda secara keseluruhan dan kebugaran, dan mengurangi risiko Anda untuk penyakit kronis [24].

#### 7. Diet

Diet adalah jumlah makanan yang dikonsumsi oleh seseorang. Jenis diet sangat dipengaruhi oleh latar belakang asal individu atau keyakinan yang dianut masyarakat tertentu. Walaupun manusia pada dasarnya adalah omnivora, suatu kelompok masyarakat biasanya memiliki preferensi atau pantangan terhadap beberapa jenis makanan. Berikut ini adalah rekomendasi diet :

1. Makan lebih banyak makanan dari tumbuhan, seperti sayuran dan kacang-kacangan, biji-bijian , dan kacang-kacangan .
2. Makan lebih banyak makanan laut di tempat daging merah, unggas, dan telur.
3. Batasi asupan natrium, lemak padat, ditambahkan gula, dan biji-bijian olahan.
4. Mengurangi kalori yang Anda makan dan minum dan meningkatkan kalori yang Anda bakar melalui aktivitas fisik.

Kelebihan berat badan menempatkan beban pada seluruh sistem peredaran darah. Hal ini juga dapat membuat orang lebih cenderung memiliki kolesterol tinggi, tekanan darah tinggi dan diabetes, yang semuanya dapat meningkatkan risiko

stroke [24]. Diet dapat ditentukan dengan IMT (Indeks Massa Tubuh) dengan persamaan sebagai berikut [26]:

$$IMT = \frac{\text{berat badan}}{(\text{tinggi badan})^2} \dots \dots \dots (2-9)$$

Keterangan :

Berat badan = berat badan dalam ukuran kilogram (kg).

Tinggi badan = tinggi badan dalam ukuran meter (m).

8. Riwayat Keluarga Stroke

Resiko stroke meningkat jika anggota keluarga (orang tua, kakek-nenek, atau saudara) telah mengalami stroke atau serangan jantung pada usia dini [24]. Faktor genetik yang sangat berperan antara lain adalah tekanan darah tinggi, penyakit jantung, diabetes, dan cacat pada pembuluh darah [23, p. 32].

**2.7. Data Flow Diagram (DFD)**

DFD adalah alat yang menggambarkan aliran data melalui sistem dan kerja atau pengolahan yang dilakukan oleh sistem tersebut [27, p. 326]. DFD merupakan alat perancangan sistem yang berorientasi pada alur data dengan konsep dekomposisi dapat digunakan untuk penggambaran analisa maupun rancangan sistem yang mudah dikomunikasikan oleh professional sistem kepada pemakai maupun pembuat program [28, p. 1]. Berikut ini adalah simbol-simbol yang digunakan dalam DFD:

1. Terminator/Entitas Luar

Terminator/Entitas luar merupakan suatu entitas yang berkomunikasi dengan sistem yang dibuat. Biasanya terminator dikenal dengan entitas luar (*external entity*) [28, p. 2]. Simbol terminator dapat ditunjukkan pada gambar berikut ini:

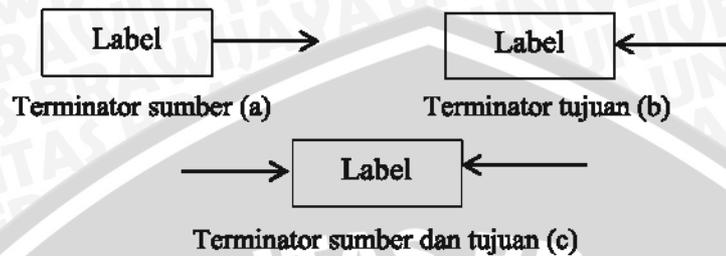


Gambar 2.6 Simbol Terminator  
Sumber : [4, p. 41]

Terminator dibagi menjadi 2 jenis yaitu [28]:

- Terminator sumber (*source*) adalah terminator yang menjadi sumber sistem, ditunjukkan pada Gambar 2.11 (a).

- Terminator tujuan (*sink*) adalah terminator yang menjadi tujuan data/informasi sistem, terminator jenis ini dapat ditunjukkan pada Gambar 2.11 (b).



Gambar 2.7 Jenis Terminator  
Sumber : [28, p. 2]

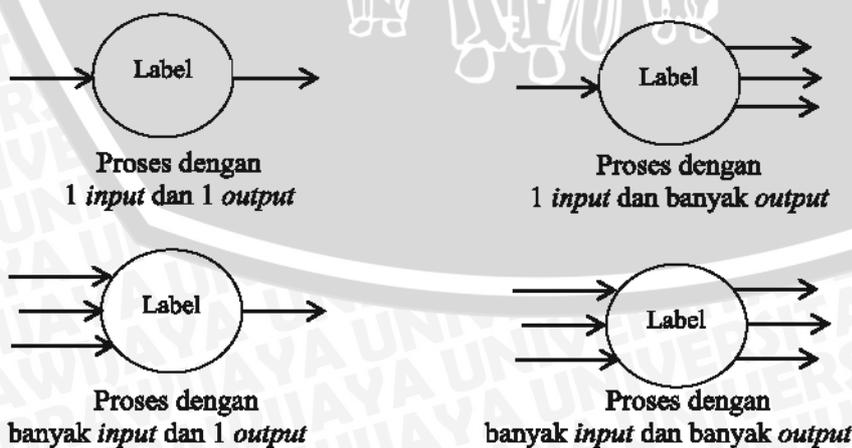
## 2. Proses

Proses merupakan suatu gambaran bagian dari sistem yang dibuat dengan mengubah *input* menjadi *output*, simbol proses dapat dilihat pada Gambar 2.12.[28, p. 1].



Gambar 2.8 Simbol Proses  
Sumber : [28, p. 2]

Proses dapat dibedakan menjadi 4 kemungkinan terjadinya proses yang dapat dilihat pada Gambar 2.13 [28, p. 3].



Gambar 2.9 Jenis Proses  
Sumber : [28, p. 3]

### 3. *Data Store*

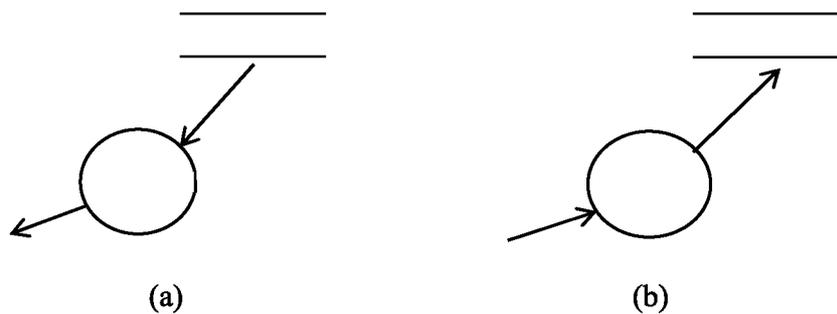
*Data store* merupakan suatu komponen yang digunakan untuk pembuatan model kumpulan data pada suatu sistem yang dibangun seperti file, database komputer, arsip, catatan manual, kotak tempat data, tabel acuan, dan buku. *Data store* juga disebut sebagai tempat penyimpanan data pengikat data yang ada dalam sistem. *Data store* dapat disimbolkan menjadi dua jenis yaitu dengan sepasang dua garis sejajar atau dua bentuk persegi panjang yang terbagi dua. Proses dapat mengambil data dari atau memberikan data ke database. Simbol *data store* tersebut dapat dilihat pada Gambar 2.14 [27, p. 346].



Gambar 2.10 Simbol *Data Store*  
Sumber : [27, p. 346]

Terdapat beberapa jenis aliran data yang digunakan dalam gambaran alir di *data store*. Jenis aliran data ini tentunya memiliki fungsi yang berbeda. Berdasarkan aliran data yang ada pada *data store* tersebut, maka aliran data pada *data store* tersebut dapat dibedakan menjadi dua jenis yaitu [28, p. 5]:

- Alur data dari *data store*  
Alur data dari *data store* merupakan pengaksesan data dalam sistem tetapi tidak terjadi perubahan dalam *data store*. Alur data dari *data store* dapat ditunjukkan pada Gambar 2.15 (a).
- Alur data ke *data store*  
Alur data ke *data store* merupakan *update* atau pengubahan data dalam sistem sehingga terjadi perubahan dalam data store seperti menghapus atau mengubah data dalam sistem. Alur data ke *data store* dapat ditunjukkan pada Gambar 2.15 (b).



Gambar 2.11 Jenis *Data Store*  
Sumber : [28, p. 5]

#### 4. *Data Flow*

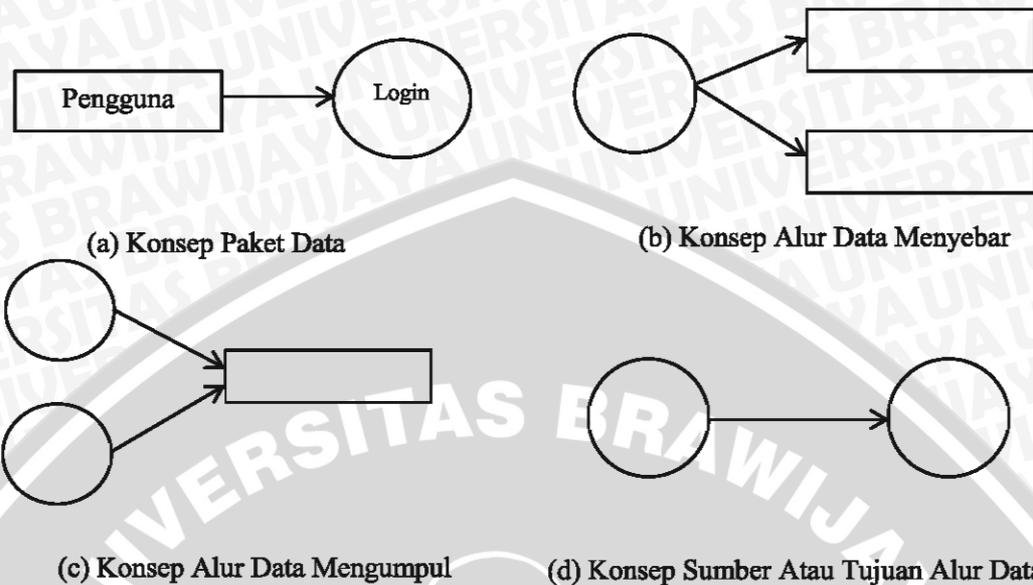
*Data flow* merupakan alur data yang digunakan untuk menjelaskan proses aliran data yang digambarkan dengan anak panah sekaligus menunjukkan arah aliran data menuju atau keluar dari proses. Simbol *data flow* dapat dilihat pada Gambar 2.16 [28, p. 6].



Gambar 2.12 Simbol *Data Flow*  
Sumber : [4, p. 41]

Terdapat 4 konsep yang digunakan dalam penggambaran alur data, yaitu [28, p. 6]:

- Konsep paket data, merupakan alur data yang menggabungkan beberapa data berhubungan menjadi satu paket data yang sama. Dengan ketentuan alur data tersebut mempunyai sumber dan tujuan yang sama seperti pada Gambar 2.17 (a).
- Konsep alur data menyebar, merupakan alur data yang mempunyai sumber sama dan menyebar ke tujuan yang berbeda seperti pada Gambar 2.17 (b).
- Konsep alur data mengumpul, merupakan alur data dari beberapa sumber data yang berbeda dan mengumpul ke tujuan yang sama seperti pada Gambar 2.17 (c).
- Konsep sumber atau tujuan alur data, merupakan alur data dengan ketentuan harus mengandung minimal satu proses. Terdapat beberapa contoh penggunaan diantaranya seperti pada Gambar 2.17 (d):
  - Alur data yang bersumber atau bertujuan pada suatu proses.
  - Alur data yang bersumber pada suatu proses dan bertujuan pada suatu proses.



Gambar 2.13 Konsep Data Flow

Sumber : [28, p. 8]

Berikut ini adalah tahapan perancangan dengan menggunakan DFD [29, pp. 72-73]:

1. Membuat DFD Level 0 atau sering disebut juga *Context Diagram*.

DFD Level 0 menggambarkan sistem yang akan dibuat sebagai suatu entitas tunggal yang berinteraksi dengan orang maupun sistem lain. DFD Level 0 digunakan untuk menggambarkan interaksi antara sistem yang akan dikembangkan dengan entitas luar.

2. Membuat DFD Level 1.

DFD Level 1 digunakan untuk menggambarkan modul-modul yang ada dalam sistem yang akan dikembangkan. DFD Level 1 merupakan hasil *breakdown* DFD Level 0 yang sebelumnya sudah dibuat.

3. Membuat DFD Level 2.

Modul-modul pada DFD Level 1 dapat di-*breakdown* menjadi DFD Level 2. Modul mana saja yang harus di-*breakdown* lebih detail tergantung pada tingkat kedetailan modul tersebut. Apabila modul tersebut sudah cukup detail dan rinci maka modul tersebut sudah tidak perlu untuk di-*breakdown* lagi. Untuk sebuah

sistem, jumlah DFD Level 2 sama dengan jumlah modul pada DFD Level 1 yang di-*breakdown*.

4. Membuat DFD Level 3 dan seterusnya.

DFD Level 3,4,5 dan seterusnya merupakan *breakdown* dari modul pada DFD Level di-atasnya. *Breakdown* pada level 3,4,5, dan seterusnya aturannya sama persis dengan DFD Level 1 atau Level 2.

### 2.8. *Entity Relationship Diagram (ERD)*

ERD adalah model data yang menggunakan beberapa notasi untuk menggambarkan data dalam konteks entitas dan hubungan yang digambarkan oleh data tersebut [27, p. 281]. Terdapat komponen pada ERD seperti yang dijelaskan berikut:

- Entitas

Entitas adalah pemodelan sistem yang membantu dalam menentukan setiap konsep ke dalam suatu bentuk. Entitas merupakan perwakilan dari orang, tempat, obyek, kejadian, atau konsep yang apa yang dibutuhkan untuk menyimpan data [27]. Penamaan entitas biasanya lebih ke kata benda dan belum merupakan nama tabel [29, p. 50]. Simbol dari entitas, terdapat pada Gambar 2.22 (a).

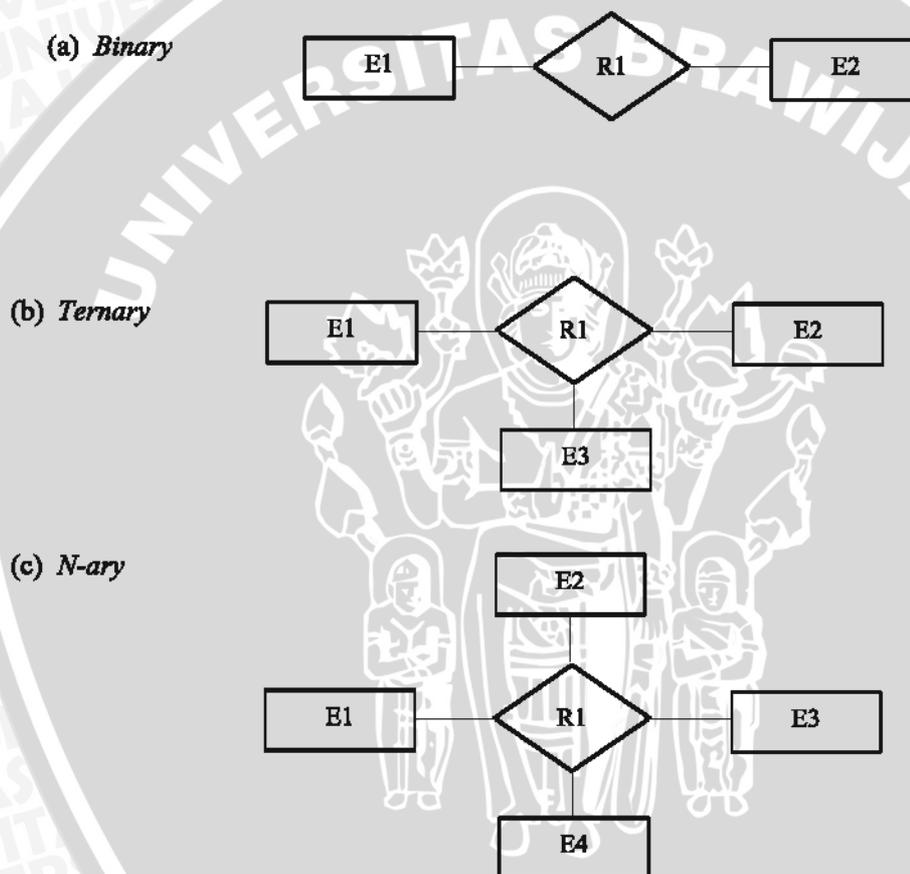
- Atribut

Atribut merupakan sifat atau karakteristik deskriptif yang dimiliki oleh entitas. Setiap diagram hubungan entitas bisa terdapat lebih dari satu atribut [27]. *Field* atau kolom data yang dibutuhkan disimpan dalam suatu entitas [29, p. 50]. Atribut digambarkan dalam bentuk elips. Simbol dari atribut terdapat pada Gambar 2.22 (b).

- Relasi

Relasi merupakan hubungan antara satu entitas atau lebih entitas. Relasi ini mewakili suatu peristiwa antar entitas atau hanya sebagai relasi logis antar entitas. Pada penggambaran diagram hubungan entitas, relasi adalah perekat yang menghubungkan suatu entitas dengan entitas lainnya [27]. Nama relasi diawali dengan kata kerja [29, p. 51]. Simbol dari relasi ditunjukkan pada Gambar 2.22

(c). ERD memiliki hubungan *binary* (satu relasi menghubungkan dua buah entitas). Beberapa metode perancangan menoleransi hubungan relasi *ternary* (satu relasi menghubungkan tiga buah relasi) atau *N-ary* (satu relasi menghubungkan banyak entitas). Tetapi banyak metode perancangan yang tidak mengizinkan hubungan *ternary* dan *N-ary* [29, pp. 51-52]. Berikut adalah contoh bentuk relasi dalam ERD:

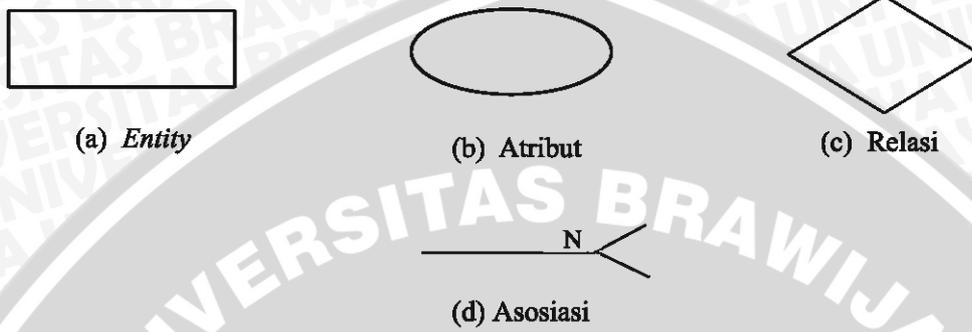


Gambar 2.14 Macam-macam Relasi  
Sumber : [29, pp. 51-52]

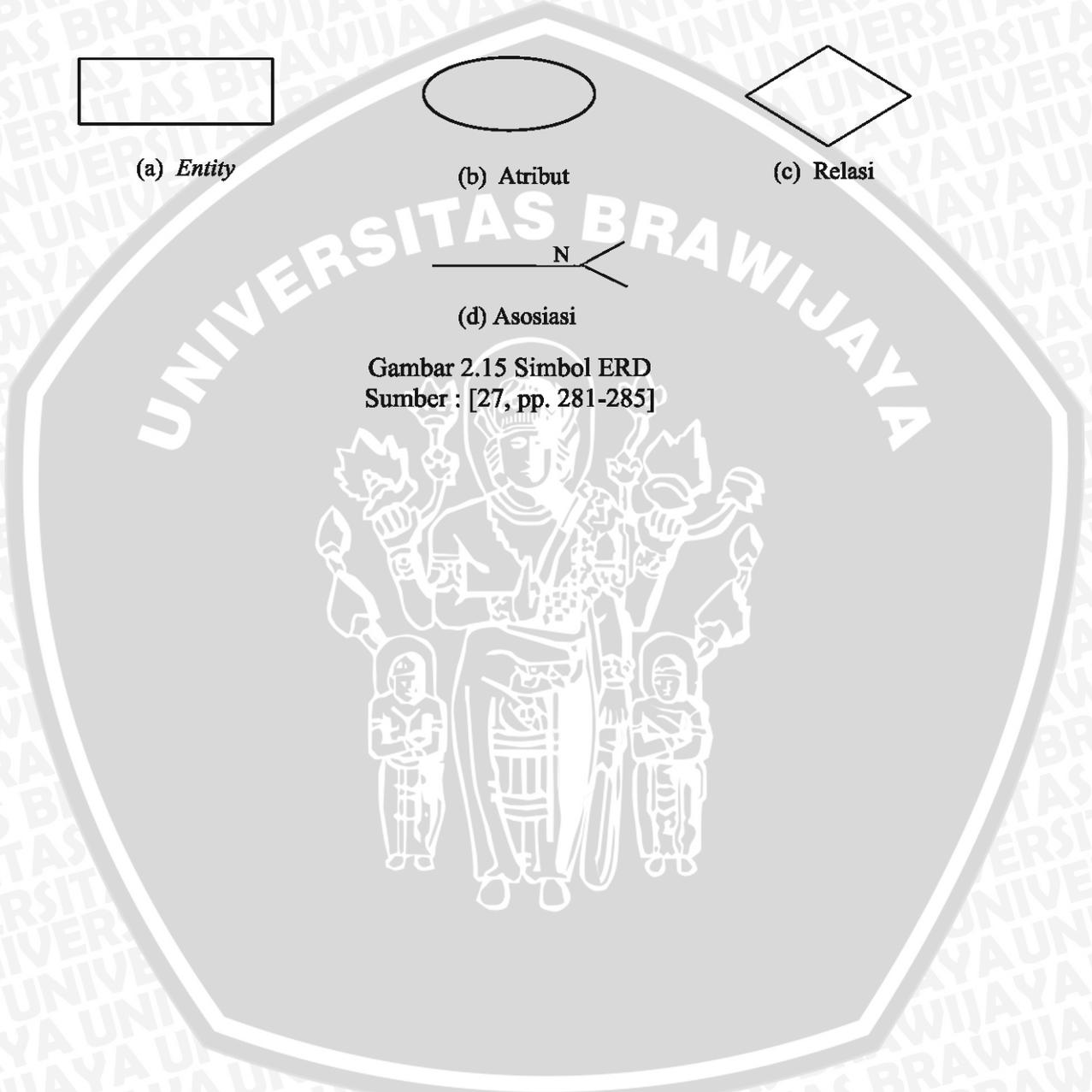
- **Asosiasi/Association**

Asosiasi merupakan penghubung antara relasi dan entitas. Pada ujung relasi dan entitas terdapat *multiplicity* yang merupakan kemungkinan jumlah pemakaian.

Kemungkinan jumlah maksimum keterhubungan antara satu entitas dengan entitas lainnya yang disebut kardinalitas 1 ke N atau disebut *one to many* [29, p. 51]. Pada gambar 2.22 (d) merupakan simbol asosiasi.

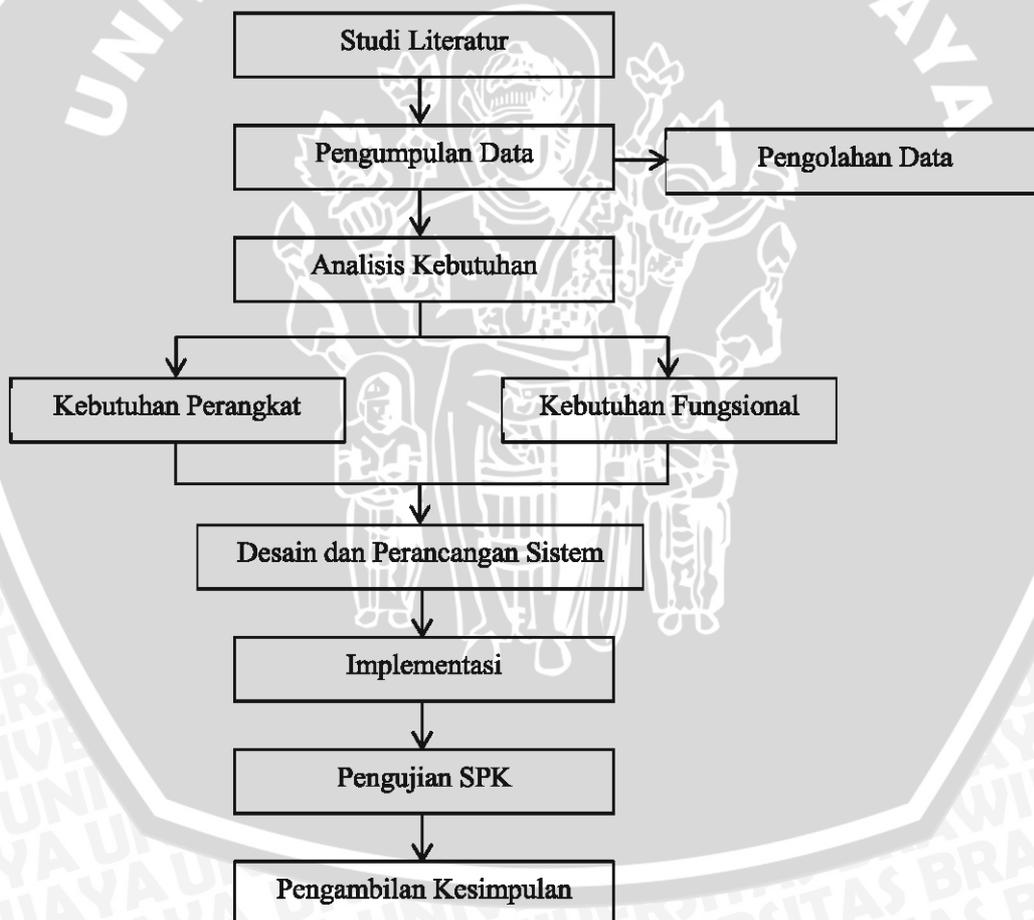


Gambar 2.15 Simbol ERD  
 Sumber : [27, pp. 281-285]



### BAB III METODOLOGI PENELITIAN

Bab ini berisi tentang penjelasan langkah-langkah yang dilakukan dalam pembuatan Sistem Pendukung Keputusan Deteksi Dini Penyakit Stroke Menggunakan Metode *Dempster Shafer*. Langkah-langkah tersebut meliputi studi literatur, pengumpulan data, analisa kebutuhan, perancangan sistem, implementasi sistem, pengujian serta pengambilan keputusan. Berikut ini merupakan diagram alir yang menjelaskan mengenai metodologi yang digunakan seperti dijelaskan pada Gambar 3.1.



Gambar 3.1 Diagram Alir Metodologi  
Sumber : [Metodologi]

### 3.1. Studi Literatur

Setelah menentukan objek penelitian yang akan dikerjakan, hal yang selanjutnya dilakukan adalah mencari dan memahami studi pustaka/literatur untuk memahami konsep rancangan aplikasi yang harus dipelajari agar dalam perancangan aplikasi tidak menemukan kendala dalam pemahaman konsep. Dalam penelitian ini literatur yang digunakan adalah :

- 1) Kajian Pustaka
- 2) Stroke
- 3) Sistem Pendukung Keputusan (SPK)
- 4) Pengertian Metode *Dempster Shafer*
- 5) *Data Flow Diagram* (DFD)
- 6) *Entity Relationship Diagram* (ERD)

### 3.2. Pengumpulan Data

Dalam mendapatkan informasi yang dibutuhkan maka dilakukan proses pengumpulan data dengan cara wawancara ke dr. Eko Arisetijono M., Sp.S(K) sebagai dokter spesialis saraf (konsultan) dan observasi data di puskesmas Kota Malang. Pengumpulan data pada penelitian SPK deteksi dini penyakit stroke menggunakan data yang didapat dari sumber penelitian. Data yang didapatkan merupakan data kasus pasien beresiko stroke pada wilayah Kota Malang sesuai dengan kriteria yang telah ditetapkan oleh *National Stroke Association* (NSA) yaitu tekanan darah, riwayat fibrilasi atrium, merokok, diabetes, kolesterol, aktivitas fisik, diet, dan riwayat keluarga.

### 3.3. Analisa Kebutuhan

Analisa kebutuhan bertujuan untuk menentukan kebutuhan yang diperlukan dalam membuat sistem. Analisa kebutuhan disesuaikan dengan variabel penelitian, lokasi penelitian, dan kebutuhan penelitian. Metode analisa dilakukan dengan pemodelan *data flow diagram* (DFD) yang bertujuan untuk mendeskripsikan kebutuhan dan pengguna. Spesifikasi kebutuhan yang digunakan dalam pembuatan sistem pendukung keputusan ini antara lain :

- a. Kebutuhan perangkat keras (*hardware*), meliputi :
  - Laptop dengan *Processor* Intel® Core™ i5-450M CPU @ 2.4 GHz
  - *Memory* DDR-3 2 GB
- b. Kebutuhan perangkat lunak (*software*) meliputi :
  - Microsoft Windows 7 sebagai sistem operasi
  - XAMPP sebagai server *localhost* dengan menggunakan DBMS MySQL sebagai sistem manajemen database dari aplikasi.
  - Notepad ++
  - Bahasa pemrograman PHP sebagai pengembang aplikasi.
- c. Kebutuhan data meliputi :
  - Data penyakit stroke dan data kasus pasien beresiko stroke

#### 3.4.Desain dan Perancangan Sistem

Perancangan sistem merupakan uraian secara umum jalannya sistem yang akan dibuat. Perancangan sistem dilakukan untuk mempermudah implementasi, pengujian, dan analisis. Langkah-langkah yang dilakukan dalam perancangan sistem ini adalah :

1. Perancangan diagram blok SPK dan *Data Flow Diagram* (DFD)

Diagram blok SPK menjelaskan penguraian logis dari fungsi sistem dan hubungannya satu sama lain. DFD digunakan sebagai perangkat yang memodelkan sistem untuk mempermudah memahami sistem secara jelas dan terstruktur.

2. Perancangan subsistem manajemen data

Subsistem manajemen data mengatur penyimpanan data yang dibutuhkan pada sistem dalam database.

3. Perancangan subsistem manajemen berbasis pengetahuan

Subsistem manajemen berbasis pengetahuan berisi pengetahuan terkait penentuan deteksi dini penyakit stroke untuk mendukung kebutuhan subsistem manajemen lainnya.

4. Perancangan subsistem manajemen model

Pada subsistem manajemen model menggunakan metode *Dempster Shafer* sebagai model untuk pengambilan keputusan yang terbaik. Subsistem manajemen model menjelaskan penggunaan metode perhitungan *Dempster Shafer* sebagai penentuan tingkat resiko penyakit stroke untuk mendukung pengujian SPK.

5. Perancangan subsistem antarmuka pengguna

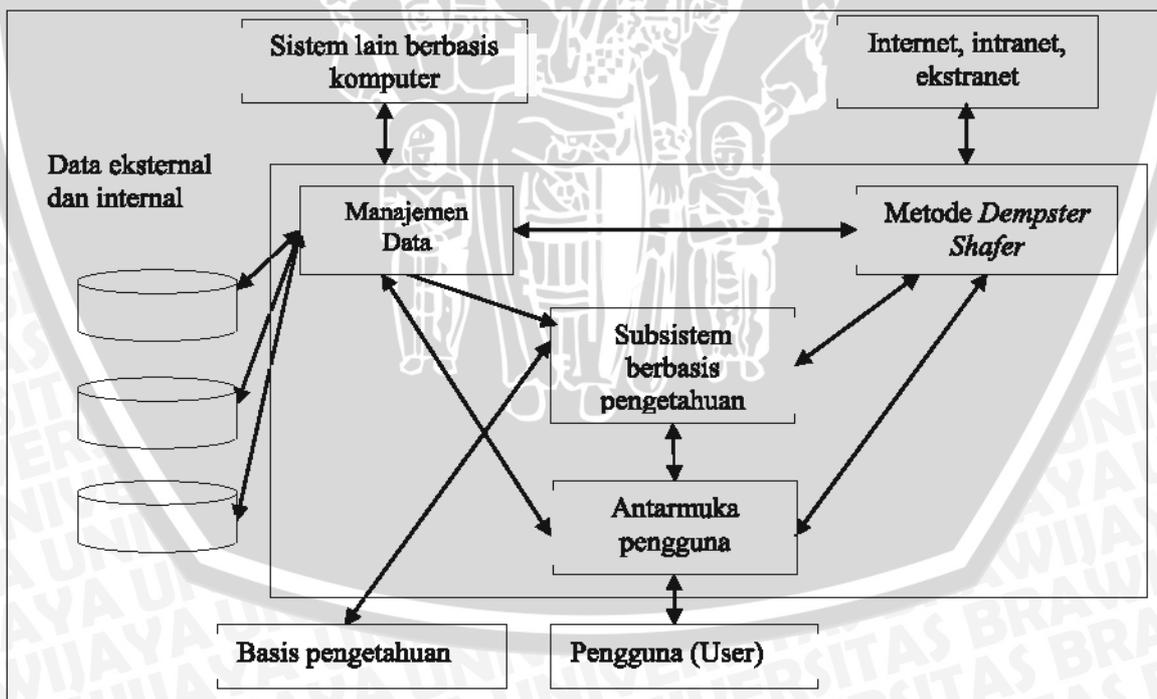
Perancangan subsistem memudahkan pengguna untuk menggunakan sistem yang dibangun.

6. Perancangan algoritma

Perancangan algoritma metode *Dempster Shafer* dalam melakukan proses pendeteksian dini penyakit stroke.

3.4.1. Arsitektur Sistem Pendukung Keputusan

SPK deteksi dini penyakit stroke pada penelitian ini akan menggunakan metode *Dempster Shafer* sebagai metode dalam pencarian alternatif. Arsitektur SPK deteksi dini penyakit stroke metode *Dempster Shafer* terdapat pada Gambar 3.2.



Gambar 3.2 Arsitektur SPK Deteksi Dini Penyakit Stroke Metode *Dempster Shafer*  
 Sumber : [Metodologi]

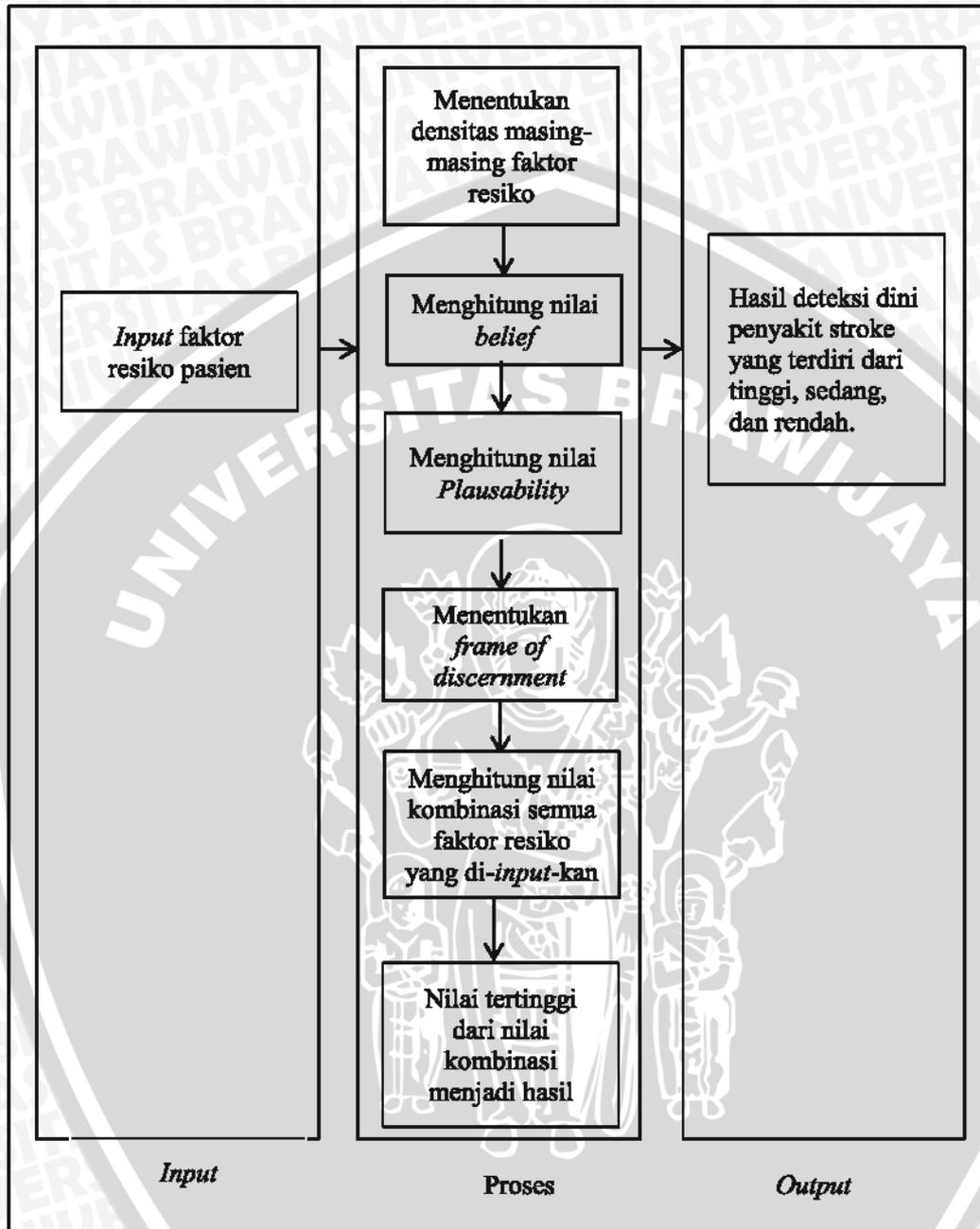
Pada Gambar 3.2 dijelaskan tentang SPK deteksi dini penyakit stroke dengan metode *Dempster Shafer* yang dibagi menjadi beberapa komponen SPK, yaitu :

- a. Subsistem manajemen data diwakili oleh data internal dan eksternal yang berfungsi untuk pengelolaan data yang berkaitan dengan deteksi penyakit stroke.
- b. Subsistem manajemen model terlihat pada penggunaan metode *Dempster Shafer* yang berfungsi untuk melakukan perhitungan dalam penyelesaian masalah SPK.
- c. Subsistem berbasis pengetahuan menjelaskan alternatif dari basis pengetahuan yang sesuai dengan kriteria dan data penyakit stroke.
- d. Antarmuka pengguna berfungsi sebagai perantara antara pengguna dengan sistem yang ada.
- e. User merupakan sebagai pengguna SPK.
- f. Garis penghubung panah bolak-balik untuk menggambarkan proses data masuk dan data keluar.

#### 3.4.1. Diagram Blok Sistem

Diagram blok sistem merupakan diagram yang menggambarkan aliran proses dari sistem yang akan dibangun. Diagram blok sistem menjelaskan cara kerja sistem dari segi matematis. Diagram blok menggambarkan sejumlah komponen yang berperan dalam sistem, komponen tersebut digambarkan dalam bentuk blok blok terhadap aliran proses. Tiap tiap blok terhubung satu sama lain melalui sinyal yang ditransmisikan.

Diagram blok ini dapat menjelaskan proses sistem berjalan dimulai dengan *input* yang dimasukkan, proses metode *Dempster Shafer*, dan keluaran yang dihasilkan. Proses metode *Dempster Shafer* pada sistem melalui beberapa tahapan perhitungan sebelum akhirnya mendapatkan hasil keluaran. Diagram blok sistem pada SPK deteksi dini penyakit stroke menggunakan metode *Dempster Shafer* dapat ditunjukkan pada Gambar 3.3 berikut:



Gambar 3.3 Diagram Blok SPK Deteksi Dini Penyakit Stroke  
 Sumber : [Perancangan]

Pada Gambar 3.3 diagram blok tersebut menjelaskan mengenai cara kerja SPK deteksi dini penyakit stroke menggunakan metode *Dempster Shafer* yang dibagi menjadi beberapa komponen utama, yaitu :

a. *Input*

*Input* pada sistem ini merupakan masukan dari pengguna berdasarkan kondisi pasien yang berupa tekanan darah, riwayat fibrilasi atrium, merokok, kolesterol, diabetes, aktivitas fisik, berat badan, tinggi badan, dan riwayat keluarga stroke.

b. *Proses*

Proses dimulai dengan melakukan pendeteksian resiko penyakit stroke sesuai dengan kondisi pasien. Selanjutnya menentukan nilai *belief* dan mencari nilai *plausibility*. Setelah densitas terbentuk, maka selanjutnya menentukan *frame of discernment*. Selanjutnya melakukan kombinasi antar faktor resiko yang diketahui. Nilai kombinasi yang tertinggi akan menjadi hasil akhir.

c. *Output*

Keluaran berupa tingkat resiko penyakit stroke (tinggi, sedang, rendah) sesuai dengan keadaan pasien.

### 3.5. Implementasi Sistem

Implementasi merupakan tahap pembangunan sistem. Implementasi sistem menggunakan bahasa pemrograman PHP, database MySQL, dan *tools* pendukung lainnya serta menerapkan metode *Dempster Shafer* dalam perhitungan yang dilakukan. Implementasi sistem didasarkan pada perancangan sistem yang telah dibuat. Implementasi sistem tersebut meliputi :

- a. Pembuatan antarmuka sistem
- b. Memasukkan data faktor resiko penyakit stroke ke dalam database MySQL untuk diolah menjadi data oleh sistem.
- c. Menerapkan metode *Dempster Shafer* dalam sistem dengan menggunakan bahasa pemrograman PHP.
- d. Hasil berupa hasil deteksi dini penyakit stroke sesuai dengan masukan user.

### 3.6. Pengujian SPK

Pengujian bertujuan untuk mengetahui apakah sistem telah berjalan sesuai dengan yang diharapkan. Pengujian pada penelitian SPK deteksi dini penyakit stroke menggunakan metode *Dempster Shafer* ini dilakukan dengan beberapa tahapan.

Pengujian pertama yang dilakukan adalah pengujian fungsionalitas dari aplikasi dan pengujian kedua adalah pengujian akurasi pada aplikasi. Pengujian ini dilakukan untuk mengetahui apakah sistem berjalan dengan baik atau tidak.

Pengujian yang pertama dilakukan dengan menggunakan metode *blackbox testing* untuk mengetahui fungsionalitas aplikasi apakah telah berjalan sesuai dengan kriteria yang telah ditentukan. Pengujian ini dilakukan secara internal yang dilakukan oleh peneliti dan juga eksternal yang dilakukan oleh calon pengguna. Pengujian ini tidak difokuskan terhadap alur jalannya algoritma program namun lebih ditekankan untuk menemukan kesesuaian antara kinerja sistem dengan daftar kebutuhan. Pada setiap kebutuhan dilakukan proses pengujian dengan kasus uji masing-masing untuk mengetahui kesesuaian antara kebutuhan dengan kinerja sistem.

Pengujian kedua merupakan pengujian akurasi sistem yang bertujuan untuk membandingkan keakuratan keluaran sistem dengan perhitungan secara manual. Perhitungan manual merupakan hasil diagnosa dari dokter spesialis saraf yang digunakan sebagai acuan dalam penelitian. Pengujian akurasi dilakukan untuk menguji kehandalan metode *Dempster Shafer* dalam mendeteksi dini penyakit stroke berdasarkan kondisi pasien yang ada. Pengujian akurasi dalam penelitian ini dilakukan untuk mengetahui kemampuan sistem dalam memberikan kesimpulan. Perhitungan akurasi dapat menggunakan rumus seperti pada persamaan berikut :

$$Akurasi = \frac{\Sigma \text{ data uji benar}}{\Sigma \text{ data uji}} \times 100\% \dots \dots \dots (3-1)$$

### 3.7. Pengambilan Kesimpulan

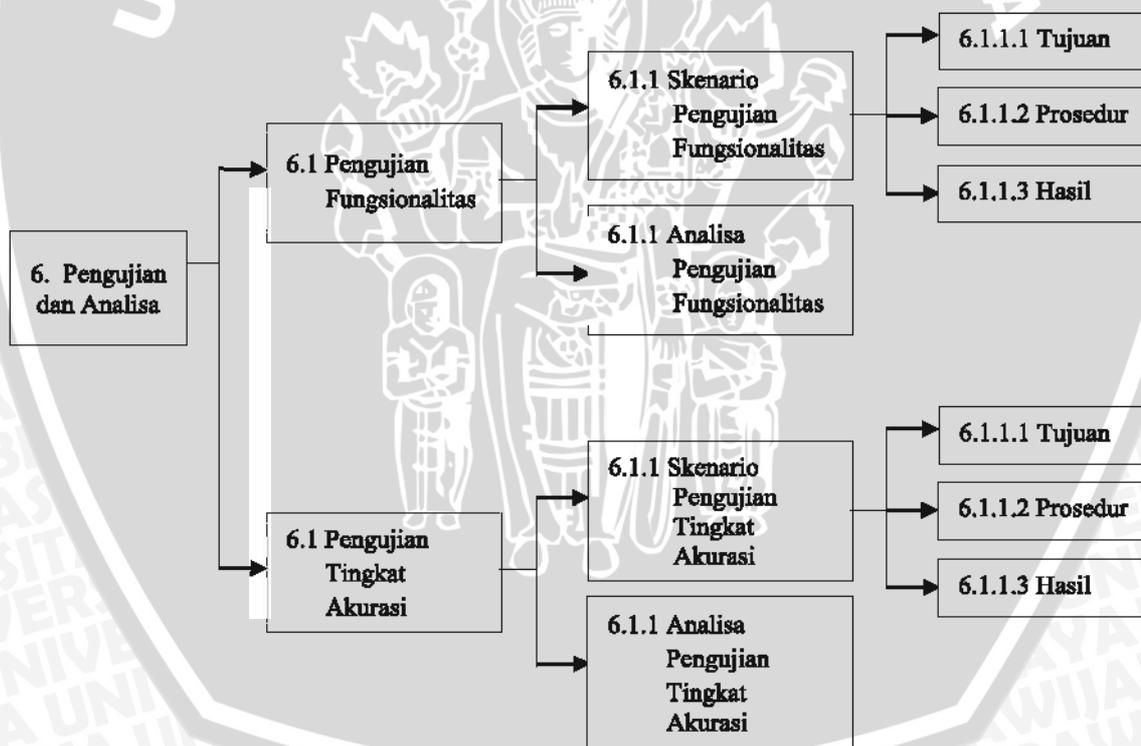
Pengambilan keputusan merupakan tahap akhir dalam penelitian yang dilakukan, pengambilan keputusan ketika perancangan, implementasi, dan pengujian sistem selesai dilakukan. Pengambilan kesimpulan bertujuan untuk mengevaluasi hasil dari pembuatan aplikasi. Kesimpulan dibuat berdasarkan hasil dari analisa dan pengujian sistem dari penggunaan metode *Dempster Shafer*.

Selain pengambilan kesimpulan, dilakukan juga penelitian saran yang bertujuan untuk memberikan pertimbangan dalam penggunaan metode *Dempster Shafer* sehingga dapat memperbaiki kekurangan yang ada.



## BAB IV PENGUJIAN DAN ANALISA

Bab ini membahas mengenai tahapan pengujian dan analisis SPK Deteksi Dini Penyakit Stroke Menggunakan Metode *Dempster Shafer*. Proses pengujian dilakukan melalui dua tahapan yaitu pengujian validasi dan pengujian akurasi. Pada pengujian validasi akan digunakan teknik pengujian *blackbox (blackbox testing)*. Pengujian akurasi digunakan untuk menguji tingkat akurasi dari sistem. Pengujian akurasi digunakan untuk membandingkan hasil dari keluaran sistem dan hasil diagnosa dari dokter spesialis saraf (konsultan). Gambar 6.1 merupakan pohon pengujian dan analisis.



Gambar 6.1 Pohon Pengujian dan Analisa  
Sumber : [Pengujian]

## 6.1. Pengujian Fungsionalitas

Pada bagian ini akan dijelaskan tentang skenario pengujian pertama yaitu pengujian fungsionalitas sistem berdasarkan daftar kebutuhan sistem menggunakan *blackbox testing*. Pengujian *blackbox* merupakan pengujian yang dilakukan terhadap sistem untuk mengetahui apakah sistem yang dibangun telah sesuai dengan daftar kebutuhan sistem yang telah ditentukan. Daftar kebutuhan yang digunakan dalam proses pengujian fungsionalitas ditunjukkan pada tabel 4.2 Pada tabel tersebut terdapat 15 kebutuhan yang harus terdapat pada sistem yang dibangun, 15 daftar kebutuhan yang telah diimplementasikan pada sistem tersebut nantinya akan diuji dengan pengujian fungsionalitas untuk mengetahui tingkat kesesuaian antara kinerja sistem dengan daftar kerja sistem yang telah disebutkan. Pengujian *Blackbox* tidak menekankan pada jalannya algoritma sistem, namun lebih kepada menemukan kesesuaian antara hasil kinerja dari sistem yang telah dibangun dengan daftar kebutuhan pengguna.

### 6.1.1. Skenario Pengujian Fungsionalitas

Sub bab berikut akan menjelaskan tentang tujuan, prosedur, serta hasil akhir yang didapatkan dari skenario pengujian pertama yang merupakan skenario pengujian fungsionalitas atau pengujian *blackbox*. Pada pengujian *blackbox* ini akan dilakukan pengamatan terhadap cara kerja SPK deteksi dini penyakit stroke menggunakan metode *Dempster-Shafer* secara eksternal.

#### 6.1.1.1. Tujuan

Pengujian ini akan dilakukan untuk mengetahui apakah sistem yang dibangun sudah sesuai dengan daftar kebutuhan sistem yang telah ditentukan. Dalam pengujian ini juga dapat diketahui apakah fitur-fitur yang terdapat pada sistem sudah dapat berjalan dengan baik dan tidak *error*.

### 6.1.1.2. Prosedur

Prosedur pengujian fungsionalitas dilakukan dengan cara membuat kasus uji untuk setiap daftar kebutuhan sistem yang telah ditentukan pada tabel 4.2. Berdasarkan daftar kebutuhan yang ditunjukkan pada tabel 4.2 terdapat 15 daftar kebutuhan sistem yang nantinya akan diuji. Setiap kasus uji daftar kebutuhan sistem akan berisi tentang nama kasus uji yang dilakukan, tujuan pengujian, prosedur pengujian, dan hasil yang diharapkan. Adapun kasus uji yang digunakan untuk pengujian fungsionalitas adalah sebagai berikut :

#### a. Kasus Uji Login

Kasus uji login menjelaskan tentang pengujian fungsionalitas proses login seperti yang ditunjukkan pada tabel 6.1.

Tabel 6.1 Penjelasan Kasus Uji untuk Pengujian Proses Login

<b>Nama Kasus Uji</b>	<b>Kasus Uji Login</b>
<b>Tujuan Pengujian</b>	Pengujian dilakukan untuk memastikan bahwa sistem dapat memenuhi kebutuhan fungsional untuk kebutuhan masuk ke dalam sistem dengan tujuan agar pengguna dapat masuk atau menggunakan SPK deteksi dini penyakit stroke sesuai dengan hak akses.
<b>Prosedur Uji</b>	<ol style="list-style-type: none"> <li>1. Sistem dijalankan ketika program dieksekusi</li> <li>2. Pengguna akan langsung dengan antarmuka login</li> <li>3. Pengguna mengisikan username dan password ke dalam kolom yang telah disediakan</li> <li>4. Pengguna menekan tombol login</li> </ol>
<b>Hasil yang diharapkan</b>	<ol style="list-style-type: none"> <li>1. Sistem dapat mengakses database sistem</li> <li>2. Sistem dapat melakukan pemeriksaan data login yang telah dimasukkan oleh pengguna</li> <li>3. Sistem dapat menampilkan pesan peringatan ketika data login yang dimasukkan tidak sesuai dengan data login yang tersimpan dalam database sistem.</li> <li>4. Sistem dapat menampilkan halaman utama</li> </ol>

Sumber : [Pengujian]

Tabel 6.2 merupakan tabel kasus pengujian fungsionalitas untuk kasus uji login pengguna.

Tabel 6.2 Skenario Pengujian Fungsionalitas Proses Login

<b>No.</b>	<b>Skenario</b>	<b>Test Case</b>	<b>Hasil yang</b>	<b>Hasil</b>	<b>Status</b>
------------	-----------------	------------------	-------------------	--------------	---------------

	Pengujian		Diharapkan	Pengujian	
1.	Mengosongkan semua field, atau salah satu field lalu menekan tombol Login	Username: - Password: -	Sistem akan menolak akses login dan menampilkan "Isi data Anda!"	Sistem akan menolak akses login dan menampilkan "Isi data Anda!"	Valid
2.	Memasukkan dengan salah satu data benar dan salah satu data salah, kemudian menekan tombol Login	Username: (benar) Password: (salah)	Sistem akan menolak akses login dan menampilkan pesan "Username dan password Anda salah!"	Sistem akan menolak akses login dan menampilkan pesan "Username atau password Anda salah!"	Valid
3.	Memasukkan data dengan benar lalu menekan tombol Login	Username: (benar) Password: (benar)	Sistem akan menerima akses login dan menampilkan halaman sesuai hak akses pengguna	Sistem akan menerima akses login dan menampilkan halaman sesuai hak akses pengguna	Valid

Sumber : [Pengujian]

**b. Kasus Uji Tambah Data Pengguna**

Kasus uji tambah pengguna menjelaskan tentang pengujian fungsionalitas proses menambah data pengguna pada sistem yang hanya dapat dilakukan oleh pengguna dengan hak akses admin seperti yang ditunjukkan pada tabel 6.3 berikut :

Tabel 6.3 Penjelasan Kasus Uji untuk Tambah Data Pengguna

Nama Kasus Uji	Kasus Uji Tambah Data Pengguna
<b>Tujuan Pengujian</b>	Pengujian dilakukan untuk memastikan bahwa sistem dapat memenuhi kebutuhan fungsional untuk menambah data pengguna.
<b>Prosedur Uji</b>	<ol style="list-style-type: none"> <li>1. Pengguna dengan hak akses admin masuk ke dalam sistem melalui proses login</li> <li>2. Pengguna memilih menu data pengguna</li> <li>3. Pengguna mengisikan data pengguna baru</li> <li>4. Pengguna menekan tombol simpan</li> </ol>
<b>Hasil yang diharapkan</b>	<ol style="list-style-type: none"> <li>1. Sistem dapat menampilkan halaman tambah data pengguna</li> <li>2. Sistem dapat menyimpan data pengguna yang baru ke dalam <i>database</i></li> </ol>

Sumber : [Pengujian]



Tabel 6.4 merupakan tabel kasus pengujian fungsionalitas untuk kasus uji tambah data pengguna.

Tabel 6.4 Skenario Pengujian Fungsionalitas Proses Tambah Data Pengguna

No.	Skenario Pengujian	Test Case	Hasil yang Diharapkan	Hasil Pengujian	Status
1.	Mengosongkan semua field, atau salah satu field lalu menekan tombol simpan	Nama: - Jenis Kelamin: - Tanggal lahir: - Pekerjaan: - Username: - Password: - Hak akses: -	Sistem akan menolak akses proses dan menampilkan "Field harus diisi"	Sistem akan menolak akses login dan menampilkan "Field harus diisi"	Valid
2.	Mengosongkan salah satu isi field lalu menekan tombol simpan	Nama: - Jenis Kelamin: (isi) Tanggal lahir: (isi) Pekerjaan: (isi) Username: (isi) Password: (isi) Hak akses: (isi)	Sistem akan menolak proses dan menampilkan pesan "Field harus diisi"	Sistem akan menolak akses login dan menampilkan pesan "Field harus diisi"	Valid
3.	Memasukkan data dengan benar pada semua field lalu menekan tombol simpan	Nama: (isi) Jenis Kelamin: (isi) Tanggal lahir: (isi) Pekerjaan: (isi) Username: (isi) Password: (isi) Hak akses: (isi)	Sistem akan menerima proses tambah data pengguna dan menampilkan pesan "Simpan berhasil"	Sistem akan menerima proses tambah data pengguna dan menampilkan pesan "Simpan berhasil"	Valid

Sumber : [Pengujian]

**c. Kasus Uji Ubah Data Pengguna**

Kasus uji ubah pengguna menjelaskan tentang pengujian fungsionalitas proses mengubah data pengguna pada sistem yang hanya dapat dilakukan oleh pengguna dengan hak akses admin seperti yang ditunjukkan pada tabel 6.5 berikut :

Tabel 6.5 Penjelasan Kasus Uji untuk Ubah Data Pengguna

<b>Nama Kasus Uji</b>	Kasus Uji Ubah Data Pengguna
-----------------------	------------------------------



<b>Tujuan Pengujian</b>	Pengujian dilakukan untuk memastikan bahwa sistem dapat memenuhi kebutuhan fungsional untuk mengubah data pengguna.
<b>Prosedur Uji</b>	<ol style="list-style-type: none"> <li>1. Pengguna dengan hak akses admin masuk ke dalam sistem melalui proses login</li> <li>2. Pengguna memilih menu data pengguna dan memilih data pengguna yang ingin diubah kemudian memilih ubah untuk mengubah data pengguna yang telah dipilih</li> <li>3. Pengguna mengisikan data pengguna baru ke form yang telah disediakan</li> <li>4. Pengguna menekan tombol simpan</li> </ol>
<b>Hasil yang diharapkan</b>	<ol style="list-style-type: none"> <li>1. Sistem dapat menampilkan halaman ubah data pengguna</li> <li>2. Sistem dapat menyimpan data pengguna yang baru ke dalam <i>database</i></li> </ol>

Sumber : [Pengujian]

Tabel 6.6 merupakan tabel kasus pengujian fungsionalitas untuk kasus uji ubah data pengguna.

Tabel 6.6 Skenario Pengujian Fungsionalitas Proses Ubah Data Pengguna

No.	Skenario Pengujian	Test Case	Hasil yang Diharapkan	Hasil Pengujian	Status
1.	Mengosongkan semua field, atau salah satu field lalu menekan tombol ubah	Nama: - Jenis Kelamin: - Tanggal lahir: - Pekerjaan: - Username: - Password: - Hak akses: -	Sistem akan menolak akses proses dan menampilkan "Field harus diisi"	Sistem akan menolak akses login dan menampilkan "Field harus diisi"	Valid
2.	Mengosongkan salah satu isi field lalu menekan tombol ubah	Nama: - Jenis Kelamin: (isi) Tanggal lahir: (isi) Pekerjaan: (isi) Username: (isi) Password: (isi) Hak akses: (isi)	Sistem akan menolak proses dan menampilkan pesan "Email atau password Anda salah!"	Sistem akan menolak akses login dan menampilkan pesan "Email atau password Anda salah!"	Valid
3.	Memasukkan data dengan benar pada semua field lalu menekan tombol ubah	Nama: (isi) Jenis Kelamin: (isi) Tanggal lahir: (isi) Pekerjaan: (isi) Username: (isi) Password: (isi)	Sistem akan menerima proses ubah data pengguna dan menampilkan pesan "Update berhasil"	Sistem akan menerima proses ubah data pengguna dan menampilkan pesan "Update berhasil"	Valid

		Hak akses: (isi)			
--	--	------------------	--	--	--

Sumber : [Penguujian]

**d. Kasus Uji Hapus Data Pengguna**

Kasus uji hapus data pengguna menjelaskan tentang pengujian fungsionalitas proses menghapus data pengguna pada sistem yang hanya dapat dilakukan oleh pengguna dengan hak akses admin seperti yang ditunjukkan pada tabel 6.7 berikut:

Tabel 6.7 Penjelasan Kasus Uji untuk Hapus Data Pengguna

<b>Nama Kasus Uji</b>	Kasus Uji Hapus Data Pengguna
<b>Tujuan Pengujian</b>	Pengujian dilakukan untuk memastikan bahwa sistem dapat memenuhi kebutuhan fungsional untuk menghapus data pengguna.
<b>Prosedur Uji</b>	<ol style="list-style-type: none"> <li>1. Pengguna dengan hak akses admin masuk ke dalam sistem melalui proses login</li> <li>2. Pengguna memilih menu data pengguna dan memilih data pengguna yang ingin diubah kemudian memilih hapus untuk menghapus data pengguna yang telah dipilih</li> </ol>
<b>Hasil yang diharapkan</b>	<ol style="list-style-type: none"> <li>3. Sistem dapat menampilkan halaman hapus data pengguna</li> <li>4. Sistem dapat menampilkan data pengguna terbaru pada halaman admin</li> </ol>

Sumber : [Penguujian]

Tabel 6.8 merupakan tabel kasus pengujian fungsionalitas untuk kasus uji hapus data pengguna.

Tabel 6.8 Skenario Pengujian Fungsionalitas Proses Hapus Data Pengguna

No.	Skenario Pengujian	Test Case	Hasil yang Diharapkan	Hasil Pengujian	Status
1.	Memilih data pengguna yang akan dihapus lalu menekan tombol hapus	Memilih data pengguna yang akan dihapus	Sistem akan menerima proses hapus data pengguna dan menampilkan pesan “Anda yakin akan menghapus	Sistem akan menerima proses hapus data pengguna dan menampilkan pesan “Anda yakin akan menghapus	Valid



			pengguna?" dan pengguna menekan tombol OK.	pengguna?" dan pengguna menekan tombol OK.	
--	--	--	---	---	--

Sumber : [Penguujian]

**e. Kasus Uji Lihat Data Faktor Resiko**

Kasus uji lihat faktor resiko menjelaskan tentang penguujian fungsionalitas proses melihat data faktor resiko pada sistem yang hanya dapat dilakukan oleh pengguna dengan hak akses admin, dokter spesialis saraf (konsultan), dan dokter umum seperti yang ditunjukkan pada tabel 6.9 berikut :

Tabel 6.9 Penjelasan Kasus Uji untuk Lihat Data Faktor Resiko

<b>Nama Kasus Uji</b>	Kasus Uji Lihat Data Faktor Resiko
<b>Tujuan Penguujian</b>	Penguujian dilakukan untuk memastikan bahwa sistem dapat memenuhi kebutuhan fungsional untuk melihat data faktor resiko.
<b>Prosedur Uji</b>	<ol style="list-style-type: none"> <li>1. Pengguna dengan hak akses admin, dokter spesialis saraf (konsultan), dokter umum masuk ke dalam sistem melalui proses login</li> <li>2. Pengguna memilih menu data faktor resiko</li> <li>3. Pengguna melihat data faktor resiko</li> </ol>
<b>Hasil yang diharapkan</b>	<ol style="list-style-type: none"> <li>1. Sistem dapat menampilkan halaman lihat data faktor resiko</li> </ol>

Sumber : [Penguujian]

Tabel 6.10 merupakan tabel kasus penguujian fungsionalitas untuk kasus uji lihat data faktor resiko.

Tabel 6.10 Skenario Penguujian Fungsionalitas Proses Lihat Data Faktor Resiko

No.	Skenario Penguujian	Test Case	Hasil yang Diharapkan	Hasil Penguujian	Status
1.	Memilih data faktor resiko dan melihat data faktor resiko	Melihat data faktor resiko	Sistem akan menerima menampilkan data faktor resiko	Sistem akan menerima menampilkan data faktor resiko	Valid

Sumber : [Penguujian]

**f. Kasus Uji Lihat Data Tingkat Resiko Stroke**



Kasus uji lihat tingkat resiko stroke menjelaskan tentang pengujian fungsionalitas proses melihat data resiko stroke pada sistem yang hanya dapat dilakukan oleh pengguna dengan hak akses admin, dokter spesialis saraf (konsultan), dan dokter umum seperti yang ditunjukkan pada tabel 6.13 berikut :

Tabel 6.11 Penjelasan Kasus Uji untuk Lihat Data Tingkat Resiko Stroke

<b>Nama Kasus Uji</b>	Kasus Uji Tambah Data Tingkat Resiko Stroke
<b>Tujuan Pengujian</b>	Pengujian dilakukan untuk memastikan bahwa sistem dapat memenuhi kebutuhan fungsional untuk melihat data tingkat resiko stroke.
<b>Prosedur Uji</b>	<ol style="list-style-type: none"> <li>1. Pengguna dengan hak akses admin, dokter spesialis saraf (konsultan), dokter umum masuk ke dalam sistem melalui proses login</li> <li>2. Pengguna memilih menu data resiko stroke</li> <li>3. Pengguna melihat data resiko stroke</li> </ol>
<b>Hasil yang diharapkan</b>	<ol style="list-style-type: none"> <li>1. Sistem dapat menampilkan halaman lihat data resiko stroke</li> </ol>

Sumber : [Pengujian]

Tabel 6.12 merupakan tabel kasus pengujian fungsionalitas untuk kasus uji lihat data tingkat resiko stroke.

Tabel 6.12 Skenario Pengujian Fungsionalitas Proses Lihat Data Resiko Stroke

No.	Skenario Pengujian	Test Case	Hasil yang Diharapkan	Hasil Pengujian	Status
1.	Memilih data resiko stroke dan melihat data resiko stroke	Melihat data resiko stroke	Sistem akan menerima menampilkan data resiko stroke	Sistem akan menerima menampilkan data resiko stroke	Valid

Sumber : [Pengujian]

#### g. Kasus Uji Tambah Data Aturan

Kasus uji tambah aturan menjelaskan tentang pengujian fungsionalitas proses menambah data aturan pada sistem yang hanya dapat dilakukan oleh pengguna dengan hak akses admin dan dokter spesialis saraf (konsultan) seperti yang ditunjukkan pada tabel 6.13 berikut :

Tabel 6.13 Penjelasan Kasus Uji untuk Tambah Data Aturan

<b>Nama Kasus Uji</b>	Kasus Uji Tambah Data Aturan
<b>Tujuan Pengujian</b>	Pengujian dilakukan untuk memastikan bahwa sistem dapat memenuhi kebutuhan fungsional untuk menambah data aturan.
<b>Prosedur Uji</b>	<ol style="list-style-type: none"> <li>1. Pengguna dengan hak akses admin atau dokter spesialis saraf (konsultan) masuk ke dalam sistem melalui proses login</li> <li>2. Pengguna memilih menu data aturan</li> <li>3. Pengguna mengisikan data aturan baru</li> <li>4. Pengguna menekan tombol simpan</li> </ol>
<b>Hasil yang diharapkan</b>	<ol style="list-style-type: none"> <li>1. Sistem dapat menampilkan halaman tambah data aturan</li> <li>2. Sistem dapat menyimpan data aturan yang baru ke dalam <i>database</i></li> </ol>

Sumber : [Pengujian]

Tabel 6.14 merupakan tabel kasus pengujian fungsionalitas untuk kasus uji tambah data aturan.

Tabel 6.14 Skenario Pengujian Fungsionalitas Proses Tambah Data Aturan

No.	Skenario Pengujian	Test Case	Hasil yang Diharapkan	Hasil Pengujian	Status
1.	Mengosongkan semua field, atau salah satu field lalu menekan tombol simpan	Faktor resiko: - Resiko stroke: - Nilai: -	Sistem akan menolak akses proses dan menampilkan "Field harus diisi"	Sistem akan menolak akses proses dan menampilkan "Field harus diisi"	Valid
2.	Mengosongkan salah satu isi field lalu menekan tombol simpan	Faktor resiko: (isi) Resiko stroke: - Nilai: (isi)	Sistem akan menolak akses proses dan menampilkan "Field harus diisi"	Sistem akan menolak akses proses dan menampilkan "Field harus diisi"	Valid
3.	Memasukkan data dengan benar pada semua field lalu menekan tombol simpan	Faktor resiko: (isi) Resiko stroke: (isi) Nilai: (isi)	Sistem akan menerima proses tambah data aturan dan menampilkan pesan "Simpan berhasil"	Sistem akan menerima proses tambah data aturan dan menampilkan pesan "Simpan berhasil"	Valid

Sumber : [Pengujian]

#### h. Kasus Uji Ubah Data Aturan

Kasus uji ubah faktor resiko menjelaskan tentang pengujian fungsionalitas proses mengubah data aturan pada sistem yang hanya dapat dilakukan oleh pengguna dengan hak akses admin atau dokter spesialis saraf (konsultan) seperti yang ditunjukkan pada tabel 6.15 berikut :

Tabel 6.15 Penjelasan Kasus Uji untuk Ubah Data Aturan

<b>Nama Kasus Uji</b>	Kasus Uji Ubah Data Aturan
<b>Tujuan Pengujian</b>	Pengujian dilakukan untuk memastikan bahwa sistem dapat memenuhi kebutuhan fungsional untuk mengubah data aturan.
<b>Prosedur Uji</b>	<ol style="list-style-type: none"> <li>1. Pengguna dengan hak akses admin atau dokter spesialis saraf (konsultan) masuk ke dalam sistem melalui proses login</li> <li>2. Pengguna memilih menu data aturan dan memilih data aturan yang ingin diubah kemudian memilih ubah untuk mengubah data aturan yang telah dipilih</li> <li>3. Pengguna mengisikan data aturan baru ke form yang telah disediakan</li> <li>4. Pengguna menekan tombol simpan</li> </ol>
<b>Hasil yang diharapkan</b>	<ol style="list-style-type: none"> <li>1. Sistem dapat menampilkan halaman ubah data aturan</li> <li>2. Sistem dapat menyimpan data aturan yang baru ke dalam <i>database</i></li> </ol>

Sumber : [Pengujian]

Tabel 6.16 merupakan tabel kasus pengujian fungsionalitas untuk kasus uji ubah data aturan

Tabel 6.16 Skenario Pengujian Fungsionalitas Proses Ubah Data Aturan

No.	Skenario Pengujian	Test Case	Hasil yang Diharapkan	Hasil Pengujian	Status
1.	Mengosongkan field lalu menekan tombol simpan	Nilai: -	Sistem akan menolak akses proses dan menampilkan "Field harus diisi"	Sistem akan menolak akses proses dan menampilkan "Field harus diisi"	Valid
2.	Memasukkan data dengan benar pada semua field lalu menekan tombol simpan	Nilai: (isi)	Sistem akan menerima proses tambah data aturan dan menampilkan pesan "Update	Sistem akan menerima proses tambah data aturan dan menampilkan pesan "Update	Valid

			berhasil"	berhasil"	
--	--	--	-----------	-----------	--

Sumber : [Penguujian]

**i. Kasus Uji Hapus Data Aturan**

Kasus uji hapus data aturan menjelaskan tentang pengujian fungsionalitas proses menghapus data aturan pada sistem yang hanya dapat dilakukan oleh pengguna dengan hak akses admin atau dokter spesialis saraf (konsultan) seperti yang ditunjukkan pada tabel 6.17 berikut:

Tabel 6.17 Penjelasan Kasus Uji untuk Hapus Data Aturan

<b>Nama Kasus Uji</b>	Kasus Uji Hapus Data Aturan
<b>Tujuan Pengujian</b>	Pengujian dilakukan untuk memastikan bahwa sistem dapat memenuhi kebutuhan fungsional untuk menghapus data aturan.
<b>Prosedur Uji</b>	<ol style="list-style-type: none"> <li>1. Pengguna dengan hak akses admin atau dokter spesialis saraf (konsultan) masuk ke dalam sistem melalui proses login</li> <li>2. Pengguna memilih menu data aturan dan memilih data aturan yang ingin diubah kemudian memilih hapus untuk menghapus data aturan yang telah dipilih</li> </ol>
<b>Hasil yang diharapkan</b>	<ol style="list-style-type: none"> <li>1. Sistem dapat menampilkan halaman hapus data aturan</li> <li>2. Sistem dapat menampilkan data aturan terbaru pada halaman admin atau dokter spesialis saraf (konsultan)</li> </ol>

Sumber : [Penguujian]

Tabel 6.18 merupakan tabel kasus pengujian fungsionalitas untuk kasus uji hapus data aturan.

Tabel 6.18 Skenario Pengujian Fungsionalitas Proses Hapus Data Aturan

No.	Skenario Pengujian	Test Case	Hasil yang Diharapkan	Hasil Pengujian	Status
1.	Memilih data aturan yang akan dihapus lalu menekan tombol hapus	Memilih data aturan yang akan dihapus	Sistem akan menerima proses hapus data aturan dan menampilkan pesan "Anda yakin akan menghapus?" lalu klik OK	Sistem akan menerima proses hapus data aturan dan menampilkan pesan "Anda yakin akan menghapus?" lalu klik OK	Valid

Sumber : [Penguujian]



**j. Kasus Uji Lihat Data Aturan**

Kasus uji lihat aturan menjelaskan tentang pengujian fungsionalitas proses melihat data aturan pada sistem yang dapat dilakukan oleh pengguna dengan hak akses dokter umum seperti yang ditunjukkan pada tabel 6.19 berikut :

Tabel 6.19 Penjelasan Kasus Uji untuk Lihat Data Aturan

<b>Nama Kasus Uji</b>	Kasus Uji Tambah Data Aturan
<b>Tujuan Pengujian</b>	Pengujian dilakukan untuk memastikan bahwa sistem dapat memenuhi kebutuhan fungsional untuk melihat data aturan.
<b>Prosedur Uji</b>	<ol style="list-style-type: none"> <li>1. Pengguna dengan hak akses dokter umum masuk ke dalam sistem melalui proses login</li> <li>2. Pengguna memilih menu data aturan</li> <li>3. Pengguna melihat data aturan</li> </ol>
<b>Hasil yang diharapkan</b>	1. Sistem dapat menampilkan halaman lihat data aturan

Sumber : [Pengujian]

Tabel 6.20 merupakan tabel kasus pengujian fungsionalitas untuk kasus uji lihat data aturan.

Tabel 6.20 Skenario Pengujian Fungsionalitas Proses Lihat Data Aturan

No.	Skenario Pengujian	<i>Test Case</i>	Hasil yang Diharapkan	Hasil Pengujian	Status
1.	Memilih data aturan dan melihat data aturan	Melihat data aturan	Sistem akan menerima menampilkan data aturan	Sistem akan menerima menampilkan data aturan	Valid

Sumber : [Pengujian]

**k. Kasus Uji Proses Deteksi Dini Penyakit Stroke**

Kasus uji deteksi dini penyakit stroke menjelaskan proses fungsionalitas proses deteksi dini penyakit stroke pada pasien yang beresiko stroke yang hanya dapat dilakukan oleh pengguna dengan hak akses dokter spesialis saraf (konsultan) atau dokter umum seperti yang ditunjukkan pada tabel 6.21 berikut:

Tabel 6.21 Penjelasan Kasus Uji Proses Deteksi Dini Penyakit Stroke

<b>Nama Kasus Uji</b>	Kasus Uji Deteksi Dini Penyakit Stroke
<b>Tujuan Pengujian</b>	Pengujian dilakukan untuk memastikan bahwa sistem dapat memenuhi kebutuhan fungsional untuk mendeteksi dini

	penyakit stroke.
<b>Prosedur Uji</b>	<ol style="list-style-type: none"> <li>1. Pengguna dengan hak akses dokter spesialis saraf (konsultan) atau dokter umum masuk ke dalam sistem melalui proses login</li> <li>2. Pengguna memilih menu data deteksi dan memasukkan data faktor resiko sesuai dengan kondisi pasien beresiko stroke</li> </ol>
<b>Hasil yang diharapkan</b>	<ol style="list-style-type: none"> <li>1. Sistem dapat mendeteksi tingkat resiko stroke yang diderita pasien yang beresiko stroke</li> <li>2. Sistem dapat menampilkan hasil deteksi dini penyakit stroke</li> </ol>

Sumber : [Pengujian]

Tabel 6.22 merupakan tabel kasus pengujian fungsionalitas untuk kasus uji deteksi dini penyakit stroke

Tabel 6.22 Skenario Pengujian Fungsionalitas Proses Deteksi Dini Penyakit Stroke

No.	Skenario Pengujian	Test Case	Hasil yang Diharapkan	Hasil Pengujian	Status
1.	Memasukkan data faktor resiko pasien sesuai dengan kondisi pasien beresiko stroke	Memasukkan data faktor resiko	Sistem akan menampilkan hasil deteksi tingkat resiko penyakit stroke	Sistem akan menampilkan hasil deteksi tingkat resiko penyakit stroke	Valid

Sumber : [Pengujian]

**1. Kasus Uji Lihat Hasil Deteksi**

Kasus uji lihat aturan menjelaskan tentang pengujian fungsionalitas proses melihat hasil deteksi pada sistem yang dapat dilakukan oleh pengguna dengan hak akses dokter spesialis dan dokter umum seperti yang ditunjukkan pada tabel 6.23 berikut :

Tabel 6.23 Penjelasan Kasus Uji untuk Lihat Hasil Deteksi

<b>Nama Kasus Uji</b>	Kasus Uji Lihat Hasil Deteksi
<b>Tujuan Pengujian</b>	Pengujian dilakukan untuk memastikan bahwa sistem dapat memenuhi kebutuhan fungsional untuk melihat hasil deteksi.
<b>Prosedur Uji</b>	<ol style="list-style-type: none"> <li>1. Pengguna dengan hak akses dokter spesialis dan dokter umum masuk ke dalam sistem melalui proses login</li> <li>2. Setelah melakukan pengisian form deteksi, setelah menekan tombol Deteksi, maka halaman hasil deteksi akan muncul</li> <li>3. Pengguna melihat hasil deteksi</li> </ol>



<b>Hasil yang diharapkan</b>	1. Sistem dapat menampilkan halaman hasil deteksi
------------------------------	---

Sumber : [Pengujian]

Tabel 6.24 merupakan tabel kasus pengujian fungsionalitas untuk kasus lihat hasil deteksi.

Tabel 6.24 Skenario Pengujian Fungsionalitas Proses Lihat Hasil Deteksi

No.	Skenario Pengujian	Test Case	Hasil yang Diharapkan	Hasil Pengujian	Status
1.	Mengisi form deteksi dan menampilkan hasil deteksi	Melihat hasil deteksi	Sistem akan menerima menampilkan hasil deteksi	Sistem akan menerima menampilkan hasil deteksi	Valid

Sumber : [Pengujian]

**m. Kasus Uji Menampilkan Riwayat Deteksi**

Kasus uji menampilkan riwayat deteksi menjelaskan proses menampilkan riwayat deteksi pasien yang beresiko stroke yang hanya dapat dilakukan oleh pengguna dengan hak akses dokter spesialis saraf (konsultan) atau dokter umum seperti yang ditunjukkan pada tabel 6.25 berikut:

Tabel 6.25 Penjelasan Kasus Uji Menampilkan Riwayat Deteksi

Nama Kasus Uji	Kasus Uji Menampilkan Riwayat Deteksi
<b>Tujuan Pengujian</b>	Pengujian dilakukan untuk memastikan bahwa sistem dapat memenuhi kebutuhan fungsional untuk menampilkan riwayat deteksi.
<b>Prosedur Uji</b>	<ol style="list-style-type: none"> <li>1. Pengguna dengan hak akses dokter spesialis saraf (konsultan) atau dokter umum masuk ke dalam sistem melalui proses login</li> <li>2. Pengguna memilih menu data riwayat</li> </ol>
<b>Hasil yang diharapkan</b>	1. Sistem dapat menampilkan riwayat deteksi pasien yang beresiko stroke

Sumber : [Pengujian]

Tabel 6.26 merupakan tabel kasus pengujian fungsionalitas untuk kasus uji menampilkan riwayat pasien beresiko stroke

Tabel 6.26 Skenario Pengujian Fungsionalitas Proses Menampilkan Riwayat

No.	Skenario Pengujian	Test Case	Hasil yang Diharapkan	Hasil Pengujian	Status
1.	Memilih menu riwayat	Memilih menu riwayat	Sistem akan menampilkan data riwayat deteksi	Sistem akan menampilkan data riwayat deteksi	Valid

Sumber : [Pengujian]

#### n. Kasus Uji Menampilkan Detail Riwayat Deteksi

Kasus uji menampilkan detail riwayat deteksi menjelaskan proses menampilkan riwayat deteksi pasien yang beresiko stroke yang hanya dapat dilakukan oleh pengguna dengan hak akses dokter spesialis saraf (konsultan) atau dokter umum seperti yang ditunjukkan pada tabel 6.27 berikut:

Tabel 6.27 Penjelasan Kasus Uji Menampilkan Detail Riwayat Deteksi

<b>Nama Kasus Uji</b>	Kasus Uji Menampilkan Detail Riwayat Deteksi
<b>Tujuan Pengujian</b>	Pengujian dilakukan untuk memastikan bahwa sistem dapat memenuhi kebutuhan fungsional untuk menampilkan detail riwayat deteksi.
<b>Prosedur Uji</b>	<ol style="list-style-type: none"> <li>1. Pengguna dengan hak akses dokter spesialis saraf (konsultan) atau dokter umum masuk ke dalam sistem melalui proses login</li> <li>2. Pengguna memilih menu data riwayat</li> <li>3. Memilih detail pada riwayat deteksi</li> </ol>
<b>Hasil yang diharapkan</b>	1. Sistem dapat menampilkan detail riwayat deteksi pasien yang beresiko stroke

Sumber : [Pengujian]

Tabel 6.28 merupakan tabel kasus pengujian fungsionalitas untuk kasus uji menampilkan riwayat pasien beresiko stroke

Tabel 6.28 Skenario Pengujian Fungsionalitas Proses Menampilkan Riwayat

No.	Skenario Pengujian	Test Case	Hasil yang Diharapkan	Hasil Pengujian	Status
1.	Memilih menu riwayat	Memilih menu riwayat	Sistem akan menampilkan data riwayat deteksi	Sistem akan menampilkan data riwayat deteksi	Valid

Sumber : [Pengujian]

#### o. Kasus Uji Logout

Kasus uji logout menjelaskan proses fungsionalitas proses logout seperti pada tabel 6.29 berikut:

Tabel 6.29 Penjelasan Kasus Uji Logout

<b>Nama Kasus Uji</b>	Kasus Uji Logout
<b>Tujuan Pengujian</b>	Pengujian dilakukan untuk memastikan bahwa sistem dapat memenuhi kebutuhan fungsional untuk logout.
<b>Prosedur Uji</b>	<ol style="list-style-type: none"> <li>1. Pengguna sudah masuk ke dalam sistem sesuai dengan hak akses</li> <li>2. Pengguna memilih menu logout</li> </ol>
<b>Hasil yang diharapkan</b>	<ol style="list-style-type: none"> <li>1. Sistem dapat menampilkan pesan peringatan bahwa pengguna keluar dari sistem</li> <li>2. Sistem dapat menampilkan halaman login</li> </ol>

Sumber : [Pengujian]

Tabel 6.30 merupakan tabel kasus pengujian fungsionalitas untuk kasus uji logout

Tabel 6.30 Skenario Pengujian Fungsionalitas Proses Deteksi Dini Penyakit Stroke

No.	Skenario Pengujian	Test Case	Hasil yang Diharapkan	Hasil Pengujian	Status
1.	Memilih menu logout dan pengguna dapat keluar dari sistem	Memilih menu logout	Sistem akan keluar dan akan menampilkan pesan:	Sistem akan keluar dan akan menampilkan pesan:	Valid

Sumber : [Pengujian]

#### 6.1.1.3 Hasil

Berdasarkan pengujian fungsionalitas menggunakan *blackbox testing* yang telah dilakukan, terbukti bahwa seluruh kebutuhan fungsional yang telah disusun sebelumnya telah berjalan sesuai dengan hasil yang diharapkan. Hal ini membuktikan bahwa sistem dapat berjalan dengan baik sesuai dengan kebutuhan fungsional yang ada.

#### 6.1.2 Analisis Pengujian Fungsionalitas

Pengujian fungsionalitas yang menggunakan *blackbox testing* yang telah dilakukan membuktikan bahwa pada tahapan implementasi, fungsionalitas sistem

dapat berjalan sesuai dengan daftar kebutuhan fungsional yang telah dibuat sebelumnya. Proses analisa yang dilakukan adalah dengan melihat kesesuaian antara hasil yang diharapkan dan hasil pengujian (hasil yang diperoleh), kesesuaian keduanya memiliki presentase sebesar 100%.

## **6.2 Pengujian Akurasi**

Pengujian tingkat akurasi dilakukan untuk mengetahui performa dari SPK deteksi dini penyakit stroke menggunakan metode *Dempster-Shafer*. Pengujian akurasi dilakukan dengan cara membandingkan data hasil deteksi pada sistem dengan hasil deteksi pakar.

### **6.2.1. Skenario Pengujian Akurasi**

Pengujian akurasi dilakukan menggunakan 30 data uji yang diperoleh dari data kasus resiko penyakit stroke. Dari 30 data uji yang ada kemudian dilakukan analisa kesesuaian antara hasil deteksi sistem dengan hasil deteksi yang dilakukan pakar. Pengujian ini akan menghasilkan akurasi sistem sebagai ukuran performa SPK yang telah dibuat.

#### **6.2.1.1 Tujuan**

Tujuan dari pengujian akurasi adalah untuk mengetahui seberapa banyak kecocokan data antara hasil deteksi sistem dengan hasil deteksi pakar. Pakar menetapkan 30 kasus beserta deteksi penyakit yang nantinya hasil deteksi tersebut akan dievaluasi dengan hasil keputusan sistem menggunakan metode *Dempster-Shafer*.

#### **6.2.1.2 Prosedur**

Prosedur pengujian tingkat akurasi dilakukan dengan cara mencocokkan hasil deteksi pakar dengan hasil deteksi sistem. Dalam pengujian ini terdapat 30 data kasus penyakit stroke beserta dengan deteksi pakar. Nilai yang digunakan sebagai acuan diperoleh dari wawancara pakar yang terdapat pada lampiran.

**6.2.1.3 Hasil**

Hasil pengujian akurasi SPK dari 30 data kasus yang telah diuji ditunjukkan pada Tabel 6.31.

Tabel 6.31 Tabel Pengujian Akurasi Hasil Deteksi Sistem dengan Pakar

Kasus	Data Faktor Resiko Pasien	Hasil Deteksi Pakar	Nilai Densitas Sistem	Hasil Deteksi Sistem	Akurasi Sistem
1	<ol style="list-style-type: none"> <li>1. Tekanan Darah : 80/60</li> <li>2. Riwayat Fibrilasi Atrium : Detak jantung beraturan</li> <li>3. Merokok : Tidak merokok</li> <li>4. Kolesterol : 100</li> <li>5. Diabetes : 180</li> <li>6. Aktivitas Fisik : Rutin olahraga</li> <li>7. Tinggi Badan : 163</li> <li>8. Berat Badan : 54</li> <li>9. Riwayat Keluarga : Memiliki riwayat keluarga</li> </ol>	Stroke Tinggi	0,98	Stroke Tinggi	1
2	<ol style="list-style-type: none"> <li>1. Tekanan Darah : 150/90</li> <li>2. Riwayat Fibrilasi Atrium : Detak jantung tidak beraturan</li> <li>3. Merokok : Perokok</li> <li>4. Kolesterol : 300</li> <li>5. Diabetes : 250</li> <li>6. Aktivitas Fisik : Tidak pernah olahraga</li> <li>7. Tinggi Badan : 165</li> <li>8. Berat Badan : 89</li> <li>9. Riwayat Keluarga : Tidak memiliki riwayat keluarga stroke</li> </ol>	Stroke Tinggi	0,97479	Stroke Tinggi	1
3	<ol style="list-style-type: none"> <li>1. Tekanan Darah :</li> </ol>	Stroke	0,97492	Stroke	1



	<p>160/100</p> <p>2. Riwayat Fibrilasi Atrium : Detak jantung tidak beraturan</p> <p>3. Merokok : Perokok</p> <p>4. Kolesterol : 310</p> <p>5. Diabetes : 300</p> <p>6. Aktivitas Fisik : Tidak pernah olahraga</p> <p>7. Tinggi Badan : 156</p> <p>8. Berat Badan : 80</p> <p>9. Riwayat Keluarga : Tidak yakin memiliki riwayat keluarga stroke</p>	Tinggi		Tinggi	
4	<p>1. Tekanan Darah : 80/100</p> <p>2. Riwayat Fibrilasi Atrium : Detak jantung tidak beraturan</p> <p>3. Merokok : Perokok</p> <p>4. Kolesterol : 250</p> <p>5. Diabetes : 250</p> <p>6. Aktivitas Fisik : Tidak pernah olahraga</p> <p>7. Tinggi Badan : 159</p> <p>8. Berat Badan : 59</p> <p>9. Riwayat Keluarga : Tidak yakin memiliki riwayat keluarga stroke</p>	Stroke Tinggi	0,9749	Stroke Tinggi	1
5	<p>1. Tekanan Darah : 90/100</p> <p>2. Riwayat Fibrilasi Atrium : Detak jantung tidak tahu</p> <p>3. Merokok : Perokok</p> <p>4. Kolesterol : 280</p> <p>5. Diabetes : 205</p> <p>6. Aktivitas Fisik : Jarang olahraga</p> <p>7. Tinggi Badan : 165</p> <p>8. Berat Badan : 71</p> <p>9. Riwayat Keluarga : Tidak yakin</p>	Stroke Tinggi	0,55375	Stroke Tinggi	1

	memiliki riwayat keluarga stroke				
6	<ol style="list-style-type: none"> <li>1. Tekanan Darah : 130/100</li> <li>2. Riwayat Fibrilasi Atrium : Detak jantung tidak tahu</li> <li>3. Merokok : Perokok</li> <li>4. Kolesterol : 246</li> <li>5. Diabetes : 210</li> <li>6. Aktivitas Fisik : Rutin olahraga</li> <li>7. Tinggi Badan : 153</li> <li>8. Berat Badan : 59</li> <li>9. Riwayat Keluarga : Tidak memiliki riwayat keluarga stroke</li> </ol>	Stroke Tinggi	0,54831	Stroke Tinggi	1
7	<ol style="list-style-type: none"> <li>1. Tekanan Darah : 130/100</li> <li>2. Riwayat Fibrilasi Atrium : Detak jantung tidak tahu</li> <li>3. Merokok : Perokok</li> <li>4. Kolesterol : 210</li> <li>5. Diabetes : 280</li> <li>6. Aktivitas Fisik : Jarang olahraga</li> <li>7. Tinggi Badan : 163</li> <li>8. Berat Badan : 74</li> <li>9. Riwayat Keluarga : Tidak yakin memiliki riwayat keluarga stroke</li> </ol>	Stroke Tinggi	0,69624	Stroke Sedang	0
8	<ol style="list-style-type: none"> <li>1. Tekanan Darah : 160/90</li> <li>2. Riwayat Fibrilasi Atrium : Detak jantung tidak beraturan</li> <li>3. Merokok : Perokok</li> <li>4. Kolesterol : 150</li> <li>5. Diabetes : 240</li> <li>6. Aktivitas Fisik : Olahraga teratur</li> <li>7. Tinggi Badan : 163</li> <li>8. Berat Badan : 80</li> <li>9. Riwayat Keluarga :</li> </ol>	Stroke Tinggi	0,70884	Stroke Tinggi	1

	Tidak memiliki riwayat keluarga stroke				
9	<ol style="list-style-type: none"> <li>1. Tekanan Darah : 120/80</li> <li>2. Riwayat Fibrilasi Atrium : Detak jantung tidak beraturan</li> <li>3. Merokok : Perokok</li> <li>4. Kolesterol : 260</li> <li>5. Diabetes : 220</li> <li>6. Aktivitas Fisik : Tidak pernah olahraga</li> <li>7. Tinggi Badan : 170</li> <li>8. Berat Badan : 90</li> <li>9. Riwayat Keluarga : Tidak yakin memiliki riwayat keluarga stroke</li> </ol>	Stroke Sedang	0,81044	Stroke Sedang	1
10	<ol style="list-style-type: none"> <li>1. Tekanan Darah : 110/90</li> <li>2. Riwayat Fibrilasi Atrium : Detak jantung tidak tahu</li> <li>3. Merokok : Kadang merokok</li> <li>4. Kolesterol : 220</li> <li>5. Diabetes : 180</li> <li>6. Aktivitas Fisik : Jarang olahraga</li> <li>7. Tinggi Badan : 164</li> <li>8. Berat Badan : 90</li> <li>9. Riwayat Keluarga : Tidak yakin memiliki riwayat keluarga stroke</li> </ol>	Stroke Sedang	0,99191	Stroke Sedang	1
11	<ol style="list-style-type: none"> <li>1. Tekanan Darah : 100/90</li> <li>2. Riwayat Fibrilasi Atrium : Detak jantung tidak beraturan</li> <li>3. Merokok : Kadang merokok</li> <li>4. Kolesterol : 360</li> <li>5. Diabetes : 190</li> </ol>	Stroke Sedang	0,98603	Stroke Sedang	1

	6. Aktivitas Fisik : Tidak pernah olahraga 7. Tinggi Badan : 160 8. Berat Badan : 86 9. Riwayat Keluarga : Tidak yakin memiliki riwayat keluarga stroke				
12	1. Tekanan Darah : 80/60 2. Riwayat Fibrilasi Atrium : Detak jantung beraturan 3. Merokok : Tidak merokok 4. Kolesterol :145 5. Diabetes : 210 6. Aktivitas Fisik : Tidak pernah olahraga 7. Tinggi Badan : 164 8. Berat Badan : 88 9. Riwayat Keluarga : Tidak memiliki riwayat keluarga stroke	Stroke Sedang	0,52999	Stroke Rendah	0
13	1. Tekanan Darah : 120/80 2. Riwayat Fibrilasi Atrium : Detak jantung tidak tahu 3. Kolesterol : 210 4. Merokok : Perokok 5. Diabetes : 206 6. Aktivitas Fisik : Jarang olahraga 7. Tinggi Badan : 163 8. Berat Badan : 65.3 9. Riwayat Keluarga : Tidak yakin memiliki riwayat keluarga stroke	Stroke Sedang	0,87903	Stroke Sedang	1
14	1. Tekanan Darah : 100/90 2. Riwayat Fibrilasi Atrium : Detak jantung tidak tahu	Stroke Sedang	0,9417	Stroke Sedang	1

	<ol style="list-style-type: none"> <li>3. Merokok : Kadang merokok</li> <li>4. Kolesterol : 330</li> <li>5. Diabetes : 180</li> <li>6. Aktivitas Fisik : Tidak pernah olahraga</li> <li>7. Tinggi Badan : 159</li> <li>8. Berat Badan : 62.4</li> <li>9. Riwayat Keluarga : Tidak yakin memiliki riwayat keluarga stroke</li> </ol>				
15	<ol style="list-style-type: none"> <li>1. Tekanan Darah : 100/60</li> <li>2. Riwayat Fibrilasi Atrium : Detak jantung beraturan</li> <li>3. Merokok : Kadang merokok</li> <li>4. Kolesterol : 216</li> <li>5. Diabetes : 190</li> <li>6. Aktivitas Fisik : Tidak pernah olahraga</li> <li>7. Tinggi Badan : 162</li> <li>8. Berat Badan : 79</li> <li>9. Riwayat Keluarga : Tidak yakin memiliki riwayat keluarga stroke</li> </ol>	Stroke Sedang	0,93774	Stroke Sedang	1

Sumber : [Pengujian]

Hasil akurasi bernilai 1 berarti keluaran dari perhitungan sistem sama dengan hasil deteksi pakar, sebaliknya jika hasil akurasi bernilai 0 artinya keluaran dari deteksi sistem tidak sama dengan deteksi pakar. Berdasarkan tabel diatas dilakukan perhitungan akurasi menggunakan persamaan 3.1 dan menghasilkan nilai akurasi sebagai berikut.

$$\text{Nilai akurasi} = \frac{27}{30} \times 100\% = 90\%$$

Dapat disimpulkan bahwa akurasi SPK berdasarkan 30 data yang diuji adalah 90% menunjukkan bahwa SPK ini dapat berfungsi dengan baik yaitu deteksi sistem sesuai dengan deteksi pakar.

### 6.2.2 Analisis Pengujian Akurasi

Akurasi sistem berdasarkan 30 data uji adalah sebesar 90%. Nilai presentase diperoleh dari pembagian data yang benar sebesar 27 dari 30 kasus sebenarnya. Nilai akurasi tersebut diperoleh dikarenakan pada kasus 7,12,18 terjadi perbedaan hasil diagnosa dimana deteksi pakar pada kasus 7 adalah stroke tinggi namun pada sistem mendeteksi stroke sedang. Pada kasus 12 deteksi pakar adalah stroke sedang namun pada sistem mendeteksi stroke rendah. Pada kasus 18 deteksi pakar adalah stroke sedang namun pada sistem mendeteksi stroke rendah. Ketidakakurasian SPK ini dapat disebabkan oleh subyektifitas dalam memberikan nilai densitas pada setiap gejala penyakit. Karakteristik metode *Dempster-Shafer* yang terlihat pada penelitian ini yaitu apabila semakin sedikit dan umum gejala fakta yang dimasukkan maka hasil deteksinya dapat menyebar ke banyak tingkat resiko. Tetapi sebaliknya jika semakin banyak dan spesifik gejala fakta yang dimasukkan maka hasil deteksinya akan lebih spesifik ke suatu tingkat resiko tertentu.



## BAB V

### PENUTUP

Pada bab ini dibahas mengenai kesimpulan dan saran dari penelitian yang telah dilakukan berdasarkan hasil perancangan, implementasi, dan pengujian sistem.

#### 7.1 Kesimpulan

Berdasarkan pada hasil perancangan, implementasi, dan pengujian yang telah dilakukan maka kesimpulan yang didapat adalah sebagai berikut :

- a. SPK deteksi dini penyakit stroke dengan metode *Dempster Shafer* dapat digunakan sebagai salah satu cara untuk mendeteksi tingkat resiko penyakit stroke. Sistem diimplementasikan dengan menggunakan faktor resiko pasien beresiko stroke yang diberikan nilai densitas.
- b. SPK deteksi dini penyakit stroke menggunakan metode *Dempster Shafer* ini dapat digunakan untuk mendeteksi tingkat resiko penyakit stroke dengan 8 faktor resiko yang telah ditentukan yaitu tekanan darah, diabetes, riwayat keluarga, merokok, kolesterol, diet, aktivitas fisik, dan riwayat fibrilasi atrium.
- c. Hasil pengujian fungsionalitas SPK deteksi dini penyakit stroke menggunakan metode *Dempster Shafer* memiliki tingkat presentase 100%. Sedangkan hasil pengujian akurasi SPK deteksi dini penyakit stroke menggunakan metode *Dempster Shafer* memiliki tingkat kesesuaian presentase sebesar 90%.

#### 7.2 Saran

Saran yang diberikan untuk pengembangan sistem dalam penelitian selanjutnya adalah :

- a. Dapat dilakukan penelitian lebih lanjut dalam nilai densitas tiap tingkat faktor resiko penyakit stroke sehingga akurasi sistem menjadi lebih optimum.
- b. Sistem dapat dikembangkan lebih lanjut dengan menggunakan metode lain dalam proses perhitungan untuk membandingkan hasil akurasi mana yang lebih optimum.



## DAFTAR PUSTAKA

- [1] M. Irfan, *Fisioterapi Bagi Insan Stroke*, Yogyakarta: Graha ilmu, 2012.
- [2] L. Ginsberg, *Lectures Notes : Neurologi*, Jakarta: Erlangga, 2008.
- [3] C. A. Dinata, Y. Safrita and S. Sastri, "Gambaran Faktor Resiko dan Tipe Stroke pada Pasien Rawat Inap di Bagian Penyakit Dalam RSUD Kabupaten Solok Selatan Periode 1 Januari 2010 - 31 Juni 2012," *Jurnal Kesehatan Andalas*, pp. 57-61, 2013.
- [4] Kusriani, *Konsep dan Aplikasi Sistem Pendukung Keputusan*, Yogyakarta: Andi, 2007.
- [5] A. F. Sholeh, A. Djunaidy and W. Anggraeni, "Aplikasi Pendukung Keputusan Untuk Deteksi Dini Risiko Penyakit Stroke Menggunakan Logika Fuzzy Mamdani : Studi Kasus Di Rs XYZ," *Teknik POMITS*, vol. 1, pp. 1-6, 2012.
- [6] S. D. B. Mau, "Sistem Pendukung Keputusan Seleksi Beasiswa Menggunakan Teorema Bayes dan Dempster Shafer," *Pekommas*, vol. 17, pp. 23-32, 2014.
- [7] O. A. Wahab, H. Otok and A. Mourad, "A Cooperative Watchdog Model Based On Dempster-Shafer For Detecting Misbehaving Vehicles," 2014.
- [8] M. Prihastuti, A. A. Soebroto and R. Regasari, "Aplikasi Sistem Pakar Pendeteksian Dan Penanganan Dini Pada Penyakit Sapi Dengan Metode Dempster-Shafer Berbasis Web," 2013.
- [9] E. G. Wahyuni and W. Prijodiprojo, "Prototipe Sistem Pakar untuk Mendeteksi Tingkat Resiko Penyakit Jantung Koroner dengan Metode Dempster-Shafer," *IJCCS*, vol. 7, pp. 133-144, 2013.
- [10] M. K. Mukhlis, "Diagnosa Kemungkinan Pasien Terkena Stroke dengan Menggunakan Metode Naive Bayes dan Jaringan Syaraf Tiruan Berbasis Web," 2010.
- [11] P. M. Prihatini, "Metode Ketidakpastian Dan Kesamaran Dalam Sistem Pakar," 2011.
- [12] A. H. Pratama, A. A. Soebroto and R. Regasari, "Aplikasi Sistem Pakar Untuk Identifikasi Hama Dan Penyakit Tanaman Tebu Dengan Metode Naive Bayes Berbasis Web," 2014.
- [13] I. Ardhiatma, A. A. Soebroto and R. Regasari, "Aplikasi Sistem Pakar Untuk Mendiagnosa Dan Penanganan Dini Gangguan Autisme Pada Anak Dengan Metode Certainty Factor Berbasis Web," 2014.
- [14] E. Prasetyo, *Data Mining-Konsep dan Aplikasi Menggunakan MATLAB*, Yogyakarta: Andi, 2012.
- [15] E. Mulyanto, V. Suhartono and T. Sutojo, *Kecerdasan Buatan*, Yogyakarta: Andi, 2010.
- [16] S. Kusumadewi, *Artificial Intelligence*, Yogyakarta: Graha Ilmu, 2003.
- [17] I. N. Hidayati, "Pemanfaatan Teori Bukti Dempster Shafer Untuk Optimalisasi

- Penggunaan Lahan Berdasarkan Data Spasial dan Citra Multisumber," *Embryo*, vol. 7, pp. 53-66, 2010.
- [18] R. Paul, T. Groza, J. Hunter and A. Zankl, "Decision Support Methods for Finding Phenotype — Disorder Associations in the Bone Dysplasia Domain," *PlosOne*, 2014.
- [19] R. Hedwig, *Teori Sistem*, Jakarta: Universitas Bina Nusantara, 2010, pp. 1-5.
- [20] T. Sutabri, *Konsep Sistem Informasi*, Yogyakarta: Andi, 2012.
- [21] E. Turban, J. E. Aronson and T.-P. Liang, *Decision Support Systems and Intelligent Systems*, Yogyakarta: Andi, 2005.
- [22] E. Turban, *Decision Support Systems and Intelligent Systems*, Yogyakarta: Andi, 2005.
- [23] L. Sustrani, *Stroke*, Jakarta: Gramedia Pustaka Utama, 2013.
- [24] N. S. Association, "Stroke," National Stroke Association, United States, 2009.
- [25] T. A. Ariani, *Sistem Neurobehaviour*, Jakarta: Salemba Medika, 2012.
- [26] W. H. Organization, "BMI Classification," World Health Organization, Switzerland.
- [27] J. L. Whitten, L. D. Bentley and K. C. Dittman, *Metode Desain dan Analisis Sistem*, Yogyakarta: Andi, 2004.
- [28] Parno, *Sistem Informasi Data Flow Diagram*, 2010.
- [29] R. A.S. and M. Shalahuddin, *Rekayasa Perangkat Lunak*, Bandung: Informatika, 2013.
- [30] I. Sommerville, *Software Engineering*, 9th Edition, New York: Addison-Wesley, 2011.
- [31] A. Taroun and J. B. Yang, "Dempster-Shafer Theory of Evidence: Potential usage for decision making and risk analysis in construction project management," *The Built & Human Environment Review*, vol. 4, no. 1, pp. 155-166, 2011.

