

BAB II LANDASAN TEORI

2.1 Kecerdasan Buatan

Turban dalam Kusrini [1,1] mendefinisikan kecerdasan buatan sebagai ilmu yang mempelajari cara membuat komputer dapat bertindak dan memiliki kecerdasan seperti manusia.

2.1.1 Lingkup Utama Kecerdasan Buatan

Menurut Kusumadewi [2,7], lingkup utama dalam kecerdasan buatan adalah:

- a) Sistem Pakar (*Contohpert System*). Di sini komputer digunakan sebagai sarana untuk menyimpan pengetahuan para pakar.
- b) Pengelohan Bahasa Alami (*Natural Language Processing*). Dengan pengolahan bahasa alami ini diharapkan user dapat berkomunikasi dengan komputer dengan menggunakan bahasa sehari-hari.
- c) Pengenalan Ucapan (*Speech Recognition*). Melalui pengenalan ucapan diharapkan manusia dapat berkomunikasi dengan komputer menggunakan suara.

2.1.2 Konsep Kecerdasan Buatan

Untuk melakukan aplikasi kecerdasan buatan ada dua bagian utama yang sangat dibutuhkan, yaitu:

- a) Basis pengetahuan (*knowledge base*), berisi fakta-fakta, teori, pemikiran dan hubungan antara satu dengan lainnya.
- b) Motor inferensi (*inference engine*), yaitu kemampuan menarik kesimpulan berdasarkan pengalaman [2, 3].



Gambar 2.1.2 Konsep Kecerdasan Buatan

2.2 Pengertian Sistem Pakar

Sistem pakar adalah suatu program komputer yang dirancang untuk memodelkan kemampuan penyelesaian masalah yang dilakukan oleh seorang pakar.

Adapun ciri-ciri sistem pakar diantaranya adalah :

- Bekerja secara sistematis berdasarkan pengetahuan dan mekanisme tertentu.
- Dikembangkan secara bertahap dan terbatas pada bidang keahlian tertentu saja.
- Outputnya berupa saran atau anjuran.

2.2.1 Kelebihan Sistem Pakar

1. Efisiensi waktu, namun sistem atau orang biasa/awam yang terlibat di dalamnya bekerja layaknya sang pakar.
2. Penyimpanan data-data pengetahuan ke dalam database dengan lengkap dan terpercaya menyebabkan informasi yang dibutuhkan bisa diakses dalam jangka waktu yang cukup lama.
3. Suatu aplikasi sistem pakar dapat diperbanyak dan disebarluaskan dengan mudah dan cepat.

2.2.2 Kelemahan Sistem Pakar

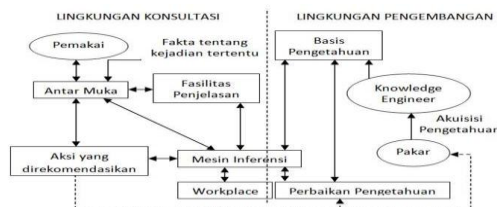
1. Daya kerja dan produktifitas manusia menjadi berkurang karena semuanya dilakukan secara otomatis oleh system.
2. Pengembangan perangkat lunak sistem pakar lebih sulit dibandingkan dengan perangkat lunak konvensional.

2.2.3 Struktur Detail Sistem Pakar

Menurut Turban dalam Arhami [3,13], sistem pakar disusun oleh duabagian utama, yaitu lingkungan pengembang (*development environment*) dan lingkungan konsultasi (*consultation environment*). Lingkungan pengembangan sistem pakar digunakan untuk memasukkan pengetahuan pakar ke dalam lingkungan sistem pakar, sedangkan lingkungan konsultasi digunakan oleh pengguna yang bukan pakar guna memperoleh pengetahuan pakar. Komponen-komponen sistem pakar dalam kedua bagian tersebut dapat dilihat dalam gambar berikut ini:

Terdiri atas 2 bagian :

- *Development Environment* (Lingkungan Pengembangan)
- *Consultation Environment* (Lingkungan Konsultasi)



Gambar 2.2.3 Struktur Detail Sistem Pakar

Komponen-komponen yang ada pada sistem pakar adalah :

- a) Subsistem penambahan pengetahuan. Bagian ini digunakan untuk memasukkan pengetahuan, mengkonstruksi atau memperluas pengetahuan dalam basis pengetahuan.
- b) Basispengetahuan. Berisi pengetahuan-pengetahuan yang dibutuhkan untuk memahami, memformulasikan dan menyelesaikan masalah.
- c) Motor inferensi. Program yang berisi metodologi yang digunakan untuk melakukan penalaran terhadap informasi-informasi dalam basis

- pengetahuan dan blackboard, serta digunakan untuk memformulasikan konklusi. Ada 3 elemen utama dalam motor inferensia, yaitu :
- ✓ *Interpreter* : mengeksekusi item-item agenda yang terpilih dengan menggunakan aturan-aturan dalam basis pengetahuan yang sesuai.
 - ✓ *Schedular* : akan mengontrol agenda.
 - ✓ *Consistency enforcer* : akan berusaha memelihara konsistensi dalam mempresentasikan solusi yang bersifat darurat.
- d) *Blackboard*. Merupakan area dalam memori yang digunakan untuk merekam kejadian yang sedang berlangsung termasuk keputusan sementara. Ada 3 tipe keputusan yang dapat direkam, yaitu :
- ✓ Rencana : bagaimana menghadapi masalah.
 - ✓ Agenda : aksi-aksi yang potensial yang sedang menunggu untuk dieksekusi.
 - ✓ Solusi : calon aksi yang akan dibangkitkan.
- e) Antarmuka. Digunakan untuk media komunikasi antara user dan program.
- f) Subsistem penjelasan. Digunakan untuk melacak respond dan memberikan penjelasan tentang kelakuan sistem pakar secara interaktif melalui pernyataan :
- ✓ Mengapa suatu pernyataan ditanyakan oleh sistem pakar ?
 - ✓ Bagaimana konklusi dicapai ?
 - ✓ Mengapa ada alternative dibatalkan ?
 - ✓ Rencana apa yang digunakan untuk mendapatkan solusi ?
- g) Sistem penyaring. Sistem ini digunakan untuk mengevaluasi kinerja sistem pakar itu sendiri untuk melihat apakah pengetahuan-pengetahuan yang ada masih cocok untuk digunakan di masa mendatang.

2.3 Representasi Pengetahuan

Agar pengetahuan dapat digunakan dalam sistem, pengetahuan harus direpresentasikan dalam format tertentu yang kemudian dihimpun dalam suatu basis pengetahuan. Cara sistem pakar merepresentasikan pengetahuan akan mempengaruhi perkembangan, efisiensi, dan perbaikan sistem [1,23].

2.3.1 Definisi

Kusrini [1,24] menjelaskan, representasi pengetahuan merupakan metode yang digunakan untuk mengkodekan pengetahuan dalam sebuah system pakar yang berbasis pengetahuan. Perepresentasian dimaksudkan untuk menangkap sifat-sifat penting problema dan membuat informasi itu dapat diakses oleh prosedur pemecahan problema.

2.4 Basis Pengetahuan

Kusumadewi [2,115] menjelaskan, basis pengetahuan berisi pengetahuan-pengetahuan dalam penyelesaian masalah, tentu di dalam domain tertentu. Ada dua bentuk pendekatan basis pengetahuan yang sangat umum digunakan, yaitu:

a) Penalaran Berbasis Aturan (*Rule-Based Reasoning*)

Pada penalaran berbasis aturan, pengetahuan direpresentasikan dengan menggunakan aturan berbentuk *if-then*. Bentuk ini digunakan apabila kita memiliki sejumlah pengetahuan pakar pada suatu permasalahan tertentu, dan si pakar dapat menyelesaikan masalah tersebut secara berurutan.

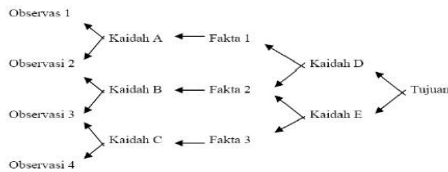
b) Penalaran Berbasis Kasus (*Case-Based Reasoning*).

Pada penalaran berbasis kasus, basis pengetahuan akan berisi solusi-solusi yang telah dicapai sebelumnya, kemudian akan diturunkan suatu solusi untuk keadaan yang terjadi sekarang (fakta yang ada). Bentuk ini akan digunakan apabila user menginginkan untuk tahu lebih banyak lagi pada kasus-kasus yang hampir sama (mirip).

2.5 Mesin Inferensi

Menurut Kusriani [1,35], inferensi merupakan proses untuk menghasilkan informasi dari fakta yang diketahui atau diasumsikan. Inferensi adalah konklusi logis (*logical conclusion*) atau implikasi berdasarkan informasi yang tersedia. Dalam sistem pakar proses inferensi dilakukan dalam suatu modul yang disebut *inference engine* (mesin inferensi). Ada dua metode inferensi yang penting dalam sistem pakar, yakni runut maju (*forward chaining*) dan runut balik (*backward chaining*).

Gambar berikut menunjukkan proses *forward chaining*

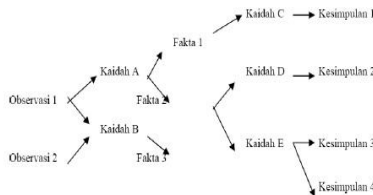


Gambar 2.5.1 Proses Forward Chaining

Sedangkan pelacakan ke depan,

Menurut Arhami [3,19], adalah pendekatan yang dimotori data (*data-driven*). Dalam pendekatan ini pelacakan dimulai dari informasi masukan, dan selanjutnya mencoba menggambarkan kesimpulan. Pelacakan ke depan mencari fakta yang sesuai dengan bagian IF dari aturan IF-THEN.

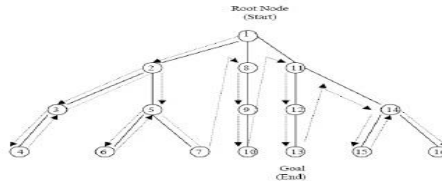
Gambar berikut menunjukkan proses *backward chaining*:



Gambar 2.5.2 Proses Backward Chaining

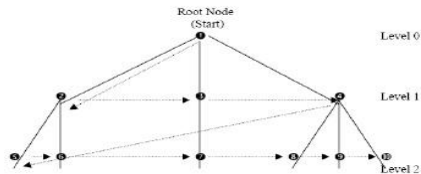
Menurut Arhami [3, 20], kedua metode inferensi tersebut dipengaruhi oleh tiga macam penelusuran, yaitu *Depth-first search*, *Breadth-first search* dan *Best-first search*.

1. *Depth-first search*, melakukan penelusuran kaidah secara mendalam dari simpul akar bergerak menurun ke tingkat dalam yang berurutan.



Gambar 2.5.3 Diagram Alir Teknik Penelusuran *Depth First Search*

2. *Breadth-first search*, bergerak dari simpul akar, simpul yang ada pada setiap tingkat diuji sebelum pindah ke tingkat selanjutnya.



Gambar 2.5.4 Diagram Alir Teknik Penelusuran *Breadth First Search*

2.6 Hormon

2.6.1 Pengertian Hormon

Hormon berasal dari kata *Hormaein* yang artinya memacu atau mengaktifkan atau merangsang. Dibutuhkan oleh tubuh dalam jumlah yang tidak terlalu banyak (sedikit), tetapi jika kekurangan atau berlebihan akan mengakibatkan hal yang tidak baik (kelainan seperti penyakit) sehingga dapat mengganggu pertumbuhan dan perkembangan serta proses metabolisme tubuh. Dengan adanya hormon dalam tubuh maka organ akan berfungsi menjadi lebih baik.

Hormonologi : yaitu ilmu yang mempelajari mengenai seluk beluk hormon.

Kelenjar dalam tubuh manusia dibedakan menjadi 2 bagian yaitu :

1. Kelenjar eksokrin yaitu kelenjar yang mempunyai saluran khusus dalam penyaluran hasil sekretnya/getahnya.
contoh : kelenjar-kelenjar pencernaan.
2. Kelenjar endokrin yaitu kelenjar yang tidak mempunyai saluran khusus dalam penyaluran hasil sekretnya/getahnya.
contoh : kelenjar hipofisis, thyroid, thymus dll.

2.6.2 Fungsi Hormon

Hormon berfungsi :

- a. Memacu pertumbuhan dan metabolisme tubuh.
- b. Memacu reproduksi.
- c. Mengatur keseimbangan cairan tubuh/homeostasis.
- d. Mengatur tingkah laku.

Macam kelenjar : A. Berdasarkan aktivitasnya :

1. Kelenjar yang bekerja sepanjang masa.

Kelenjar golongan ini akan bekerja terus menerus sepanjang kehidupan manusia dan akan berhenti jika sudah tidak ada kehidupan pada manusia tersebut. Sehingga tidak terbatas pada usia.

contoh : Hormon metabolisme.

2. Kelenjar yang bekerjanya mulai masa tertentu.

Hormon golongan ini tidak akan dapat berfungsi jika belum mencapai proses perkembangan dalam diri manusia atau proses pendewasaan sel yang terjadi dalam tubuh manusia. Kedewasaan sel akan terjadi pada saat usia tertentu seperti pada saat usia pubertas.

Contoh : Hormon kelamin.

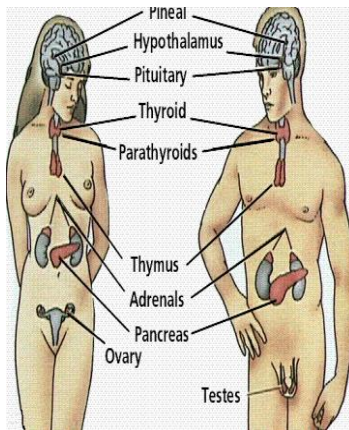
3. Kelenjar yang bekerja sampai pada masa tertentu.

Hormon golongan ini bekerja pada saat manusia itu dilahirkan sampai pada usia tertentu. Hormon ini akan berhenti dihasilkan pada saat tubuh mulai memperlambat atau menghentikan proses pertumbuhan. Biasanya hormon ini bekerja pada kisaran usia 0 hari sampai 17 tahun (masa pertumbuhan).

Contoh : Hormon pertumbuhan, kelenjar thyms.

B. Berdasarkan letaknya :

1. Kelenjar hipophysis/pituitary di dasar cerebrum, dibawah hypothalamus.
2. Kelenjar pineal/epiphysis di cerebrum.
3. Kelenjar thyroid di daerah leher.
4. Kelenjar parathyroid di dekat kelenjar thyroid.
5. Kelenjar thymus di rongga dada.
6. Kelenjar adrenal/suprarenalis di atas ren.
7. Kelenjar pulau langerhans/pankreas di rongga perut.
8. Kelenjar Usus dan lambung di rongga perut.
9. Kelenjar kelamin :
 - a. Ovarium di rongga perut.
 - b. Testis di rongga perut bawah.

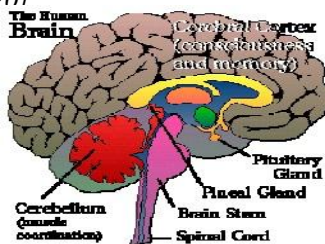


Gambar 2.6.2 letak Kelenjar endokrin

2.7 Macam – Macam Kelenjar Endokrin

1. Kelenjar PINEAL

- ✓ Hormon melatonin : warna/pigmen kulit melanin. Hormon ini dapat jugamengatur rasa kantuk pada diri seseorang. Pada remaja hormon ini dihasilkan lebih banyak bila dibandingkan dengan orang dewasa.
- ✓ Hormon vasotocin (Mammalia) : mirip fungsinya dengan vasopresin dan oksitosin [5,20]



Gambar 2.7.1 Letak Kelenjar Pineal

2. Kelenjar HIPOFISIS/PITUITARY/MASTER OF GLANDS

- ✓ **LOBUS ANTERIOR/ADENOHYPHYSIS :**
 Hormon yang dihasilkan oleh lobus anterior lebih di dominasi oleh hormon yang mengatur mengenai pertumbuhan, reproduksi dan masalah stress [6,37].
 Macam hormon yang dihasilkan :
 1. STH(Somatotrof Hormone)/GH (Growth Hormon)/Somatotropin :
 Hormon ini berfungsi :
 - a. Memacu pertumbuhan terutama pada peristiwa osifikasi, pada cakraepifise.
 - b. Mengatur metabolisme lipid dan karbohidrat.

✓ Hipersekresi :

Bila kelebihan hormon ini terjadi pada masa pertumbuhan akan mengakibatkan pertumbuhan yang tidak terkendali/menjadi lebih cepat. Pertumbuhan yang seperti ini dikenal dengan gigantisme. Sedangkan bila kelebihan hormon ini terjadi pada masa dewasa akan mengakibatkan pertumbuhan yang tidak normal pada beberapa bagian organ tubuh. Pertumbuhan akromegali biasanya terjadi diatas usia 25 tahun.

✓ Hiposekresi :

Bila penghasilan hormon ini kurang akan menyebabkan pertumbuhan kretinisme/dwarfisme, yaitu pertumbuhan yang terhambat. Pada pertumbuhan ini pertumbuhan berjalan normal, hanya saja pertumbuhan tulang sangat terhambat.

2. TSH (Thyroid Stimulating Hormone)/TRETROP/Thyrotropin :

Hormon ini berfungsi :

a. Merangsang sekresi kelenjar thyroid.

3. ACTH(Adrenocorticotropic

Hormone)/ADRENOTROPIN/Corticotropin Hormon ini berfungsi :

a. Merangsang kerja kelenjar adrenal.

4. GONADOTROPIC/HORMON KELAMIN :

✓ FSH/Folicle Stimulating Hormone : memengaruhi pembentukan folikel sel ovum dan proses spermatogenesis.

✓ LH (Luteinizing Hormone)atau ICSH (Interstitial Cell Stimulating Hormone) : Berfungsi untuk memacu sekresi hormon testosteron pada sel Leydig dan proses ovulasi sel ovum.

✓ LOBUS INTERMEDIA

5. MSH (Melanotropin Stimulating Hormone) atau INTERMEDIN:

Hormon ini berfungsi :

a. Memacu pembentukan pigmen melanin kulit.

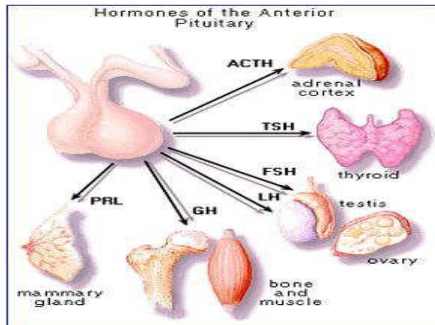
b. Mengatur penyebaran pigmen melanin.

✓ LOBUS POSTERIOR/NEUROHIPOPHYISIS

6. ADH :

Hormon ini berfungsi :

a. Mengatur pengeluaran urine. Mengatur reabsorpsi air dari tubulusren.



Gambar 2.7.2 Kelenjar Pituitary bagian anterior beserta organ yang dipengaruhinya.

3. Kelenjar THYROID

Kelenjar ini merupakan kelenjar yang kaya akan pembuluh darah dan merupakan sepasang kelenjar yang terletak berdampingan di sekitar leher [7].

Macam hormon yang dihasilkan :

1. Hormon Tiroksin (T4) dan Triiodotironin (T3)

Hormon ini berfungsi :

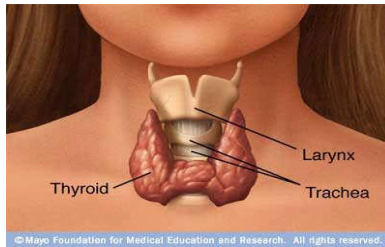
- Mengatur metabolisme karbohidrat.
- Memengaruhi perkembangan mental.
- Memengaruhi pertumbuhan, perkembangan dan diferensiasi sel.
- Memengaruhi kegiatan sistem saraf.

2. Hormon Calsitonin.

Hormon ini berfungsi :

- Menurunkan kadar Ca (Calsium) darah.
- Mengatur absorpsi Calcium oleh tulang.

Pembengkakan kelenjar Thyroid dikenal dengan istilah GOITER. Hal ini dapat disebabkan karena menurunnya hormon yang dihasilkan sehingga menyebabkan stimulasi produksi TSH berlebihan. Resiko terkena penyakit ini lebih banyak dialami oleh wanita dengan perbandingan wanita : pria adalah 5 : 1. Kisaran wanita yang terkena penyakit ini adalah anatar 40 – 60 tahun. Biasanya banyak dialami oleh penduduk daerah marjinal yang sulit mendapatkan garam beryodium. Dengan mineral Yodium/Iodium dapat mengatur pengeluaran hormon yang dihasilkan oleh kelenjar ini sehingga tubuh tidak akan kekurangan hormon dari kelenjar Thyroid.



Gambar 2.7.3 Struktur morfologi kelenjar Thyroid.

4. Kelenjar PARATHYROID

Kelenjar ini merupakan kelenjar yang menempel pada kelenjar Thyroid. Setiap kelenjar Thyroid mempunyai sepasang kelenjar Parathyroid, sehingga semuanya berjumlah 4 buah kelenjar parathyroid [8,39].

Hormon yang dihasilkan Hormon PTH (Parathormon).

Berfungsi :

- a. Mengatur metabolisme Ca^{2+} (Calcium) dan PO_4^{3-} (phosphat).
- b. Mengendalikan pembentukan tulang.

✓ Hipersekresi :

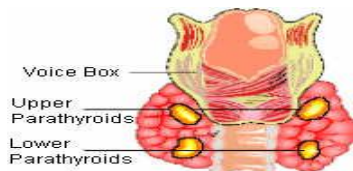
Bila terjadi kelebihan dalam penghasilan hormon ini akan menyebabkan pertumbuhan :

- a. Kretinisme bila terjadi pada masa pertumbuhan.
- b. Miksodema bila terjadi pada masa dewasa.
- c. Batu ginjal dalam pelvis renalis/rongga ginjal.

✓ Hiposekresi :

Bila terjadi kelebihan dalam penghasilan hormon ini akan menyebabkan :

- a. Pertumbuhan Morbus basedowi.
- b. Kejang otot/tetani.



Gambar 2.7.4 Struktur morfologi kelenjar Parathyroid

5. Kelenjar THYMUS

Merupakan penimbunan dari hormon somatotrof dalam tubuh. Hormon ini dihasilkan selama masa pertumbuhan sampai dengan masa pubertas, setelah melewati mas pubertas, secara perlahan hormon ini akan berkurang sedikit demi sedikit [9].

Hormon ini berfungsi :

- a. Mengatur proses pertumbuhan.

- b. Kekebalan tubuh/imunitas setelah kelahiran.
- c. Memacu pertumbuhan dan pematangan sel Limfosit yang menghasilkan Lymphocyte cell/T Cell.

Bila kekurangan atau kelebihan, gejalanya hampir mirip dengan hormone tiroksin.

6. Kelenjar ADRENAL/SUPRARENALIS

✓ BAGIAN KORTKONTOH

1. Hormon Cortison atau antiadison

Berfungsi : sebagai anti peradangan dan membantu pembentukan formasi karbohidrat.

✓ Hiposekresi :

Bila kekurangan hormon ini akan menyebabkan penyakit Adison.

Gejalanya :

- a) Kulit memerah/timbulnya ruam pada kulit.
- b) Dapat menimbulkan kematian.
- c) Tekanan darah rendah.
- d) Nafsu makan hilang.
- e) Pengendapan pigmen melanin yang banyak..

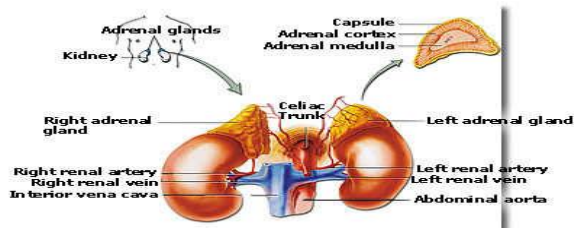
2. Hormon Corticosterone

Berfungsi :

- a. Mempengaruhi metabolisme karbohidrat, protein dan lipid.
- b. Meningkatkan respon imunitas tubuh.
- c. Hormon Mineralokortikoid
- d. Mengatur keseimbangan air dan elektrolit dalam tubuh.
- e. Merangsang reabsorpsi Na^+ dan Cl^- dalam tubulus ginjal.

✓ Hiposekresi :

Bila kekurangan hormon ini akan menyebabkan penyakit Adison.



Gambar 2.7.5 Letak kelenjar adrenal

✓ BAGIAN MEDULLA

1. Hormon Adrenalin/Epinefrin

Hormon ini secara umum berfungsi :

- a. Memicu reaksi terhadap tekanan dan kecepatan gerak tubuh.
- b. memicu reaksi terhadap efek lingkungan, seperti suara yang tinggi, intensitas cahaya dll.

Secara khusus hormon ini berfungsi :

- a) Memacu aktivitas cor/jantung
- b) Menaikkan tekanan darah
- c) Mempercepat glikolisis
- d) Pengeluaran keringat dingin
- e) Rasa keterkejutan/shock
- f) Memengaruhi otak yang akan mengakibatkan : Indera perasa menjadi kebal terhadap rasa sakit.
- g) Melindungi dari penyakit Alzheimer, penyakit jantung, kanker payudara, kanker ovarium dan osteoporosis.

✓ Hiposekresi :

Bila terjadi kekurangan penghasilan hormon adrenalin/epinefrin akan menyebabkan penyakit Addison. Gejalanya dapat dilihat pada hiposekresi Hormon Mineralokortikoid dan Hormon Cortison.

2. Hormon Androgen

Berfungsi :

- a. Menentukan sifat kelamin sekunder pada pria dan wanita.

✓ Hipersekresi : Bila terjadi kelebihan hormon ini akan menyebabkan penyakit Cushing Syndrome/sindrom Cushing serta penyakit kelainan ciri kelamin sekunder pada laki-laki dan perempuan.

Gejala Cushing syndrome :

- a) Membulatnya wajah/muka.
- b) Penimbunan lemak di daerah leher.
- c) Pengecilan pada daerah lengan dan kaki.
- d) Terhentinya atau terganggunya periode menstruasi.
- e) Penurunan daya seksualitas.
- f) Kenaikan tekanan darah dan kadar gula darah.
- g) Masalah rambut pada wanita.

7. Kelenjar VENTRICULUS

Dihasilkan Hormon Gastrin

Hormon ini berfungsi :

- a. Memacu pengeluaran sekret/getah lambung.
- b. Membantu dalam proses pencernaan.

8. Kelenjar USUS

1. Hormon Sekretin

Berfungsi memacu sekresi getah usus dan pankreas.

2. Hormon Kolesistokinin

Berfungsi memacu sekresi getah empedu dan pankreas.

9. Kelenjar LANGERHANS/PANKREAS

1. Hormon Insulin

Bersifat antagonis dengan hormon adrenalin [10,26].

Hormon ini berfungsi :

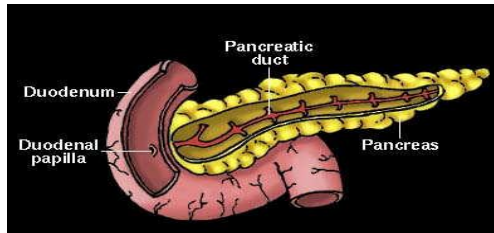
- a. Mengatur kadar glukosa dalam darah.
 - b. Membantu perubahan glukosa menjadi glikogen dalam hepar dan otot.
- ✓ Hiposekresi :
- Bila kekurangan dalam penghasilan hormon ini akan menyebabkan penyakit diabetes mellitus/penyakit kencing manis.
- Gejala penyakit diabetes mellitus :
- a. Kenaikan jumlah gula dalam darah.
 - b. Badan menjadi lems.
 - c. Sering merasa haus/banyak minum.
 - d. Banyak melakukan urinasi (pembuangan urine).
 - e. Energy berkurang.
 - f. Merasa selalu lapar.

2. Hormon Glukagon

Hormon ini mempunyai sifat kerja yang sinergis dengan hormon adrenalin.

Hormon ini berfungsi :

- a. Meningkatkan kadar gula dalam darah.
- b. Mengubah glikogen menjadi glukosa dalam peristiwa glikolisis.



Gambar 2.7.6 Struktur morfologi dan letak kelenjar Pankreas

10. Kelenjar KELAMIN/GONAD

Menghasilkan hormon dan sel kelamin [11,9].

Macamnya ada 2 sel kelamin :

1. Sel Testis

Menghasilkan Hormon Androgen, Contoh : Hormon Testosteron, merupakan satu hormon yang terpenting dalam pembentukan sel spermatozoa.

Fungsi Hormon Testosteron :

- a. Mengatur ciri kelamin sekunder.
- b. Mempertahankan proses spermatogenesis.

2. Sel Ovarium

Menghasilkan 3 hormon penting dalam seorang wanita :

- a. Hormon Estrogen

Hormon ini berfungsi untuk : memperlihatkan ciri-ciri kelamin sekunder wanita.

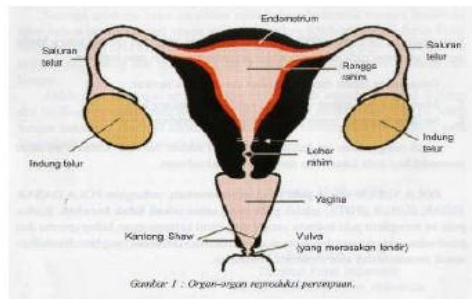
b. Hormon Progesteron

Hormon ini berfungsi :

- a. Mempersiapkan masa kehamilan dengan menebalkan dinding uterus.
- b. Menjaga kelenjar susu dalam menghasilkan air susu.

c. Hormon Relaksin

Hormon ini berfungsi untuk membantu proses persalinan dalam kontraksi otot.



Gambar 1 : Organ-organ reproduksi perempuan.

Gambar 2.7.7 Struktur morfologi kelenjar ovarium

2.8 Jenis – Jenis Penyakit Pada Sistem Hormon

Jumlah keseluruhan jenis penyakit pada sistem hormon ada 28 jenis penyakit, jenis penyakit pada system hormon ini menjelaskan definisi , penyebab , gejala serta diagnosa dari penyakit tersebut.

1. Akromegali

Akromegali adalah gangguan hormonal yang terjadi akibat kelenjar hipofise menghasilkan terlalu banyak *hormon pertumbuhan* pada orang dewasa. Akromegali biasanya terjadi pada usia dewasa muda, Gejala umum akromegali termasuk pertumbuhan abnormal tangan dan kaki, pertumbuhan tulang pada wajah yang mengarah ke rahang menonjol rendah dan kening dan tulang hidung membesar, nyeri sendi, tebal, kasar, kulit berminyak, dan bibir membesar, hidung, dan lidah.

Akromegali disebabkan oleh adanya produksi hormon pertumbuhan (growth hormone) yang berlebihan secara terus menerus oleh kelenjar hipofise. Setelah hormon pertumbuhan dilepaskan ke aliran darah, hormon ini akan memicu hati untuk menghasilkan hormon yang

disebut insulin-like growth factor I (IGF-I). IGF-I akan menstimulasi pertumbuhan tulang dan jaringan lain.

2. Cushing Syndrome

Tanda-tanda khas dan gejala sindrom Cushing termasuk obesitas tubuh bagian atas, wajah bulat, kulit yang memar dengan mudah dan menyembuhkan buruk, lemah tulang, kelebihan pertumbuhan rambut tubuh dan ketidakteraturan menstruasi pada wanita, dan penurunan kesuburan pada pria. Sindrom Cushing disebabkan oleh paparan glukokortikoid, yang digunakan untuk mengobati penyakit inflamasi, atau oleh kelebihan tubuh kortisol, paling sering disebabkan oleh tumor pada kelenjar hipofisis atau paru-paru.

3. Diabetes Insipidus

Diabetes Insipidus adalah suatu keadaan dimana terjadi kekurangan *hormon antidiuretik* yang menyebabkan rasa haus yang berlebihan (*polidipsi*) dan pengeluaran sejumlah besar air kemih (*poliuri*).

Diabetes insipidus terjadi akibat penurunan pembentukan hormon antidiuretik (*vasopresin*), yaitu hormon yang secara alami mencegah pembentukan air kemih yang terlalu banyak. Hormon ini dibuat di *hipotalamus*, kemudian disimpan dan dilepaskan ke dalam aliran darah oleh *hipofisa posterior*.

Diagnosis ditegakkan berdasarkan gejalanya. Untuk menyingkirkan diabetes melitus (kencing manis) dilakukan pemeriksaan gula pada air kemih. Selain itu, pemeriksaan darah menunjukkan berbagai kadar elektrolit yang abnormal

4. Diabetes Melitus

Diabetes Mellitus adalah suatu penyakit dimana kadar glukosa (gula sederhana) di dalam darah tinggi karena tubuh tidak dapat melepaskan atau menggunakan insulin secara kuat.

Kadar gula darah bervariasi sepanjang hari. Gula darah akan meningkat setelah makan dan kembali normal dalam waktu 2 jam. Kadar gula darah yang normal pada pagi hari setelah berpuasa malam sebelumnya adalah 70-110 mg/dL. Kadar gula darah biasanya kurang dari 120-140 mg/dL pada 2 jam setelah makan atau minum cairan yang mengandung gula maupun karbohidrat lainnya.

Gejala awalnya berhubungan dengan efek langsung dari kadar gula darah yang tinggi. Jika kadar gula darah melebihi 160-180 mg/dL, maka glukosa akan sampai ke air kemih. Jika kadarnya lebih tinggi lagi, maka ginjal akan membuang air lebih banyak untuk mengencerkan sejumlah besar glukosa yang hilang. Karena ginjal

menghasilkan air kemih dalam jumlah yang berlebihan, maka penderita menjadi lebih sering berkemih dengan air kemih yang lebih banyak (poliuri).

5. Feokromositoma

Feokromositoma adalah suatu tumor kelenjar adrenal yang jarang. Tumor ini berasal dari *sel-sel kromafin* kelenjar adrenal yang menyebabkan pembentukan *katekolamin* yang berlebihan. Katekolamin merupakan hormon yang mengontrol detak jantung, tekanan darah, dan metabolisme.

Gejala yang paling menonjol adalah tekanan darah tinggi, yang bisa sangat berat. Pada sekitar 50% penderita, tekanan darah tinggi ini bersifat menetap. Pada 50% penderita lainnya, tekanan darah tinggi dan gejala lainnya terjadi hilang-timbul, kadang dipicu oleh adanya perangsangan pada tumor, misalnya dengan pemijatan, pengobatan (terutama obat bius dan beta bloker) dan trauma emosional.

6. Galaktore

Galaktore adalah keluarnya air susu dari payudara yang tidak berhubungan dengan masa menyusui setelah kehamilan. Kondisi ini biasanya terjadi pada wanita, namun juga dapat terjadi pada pria. Cairan putih ini dapat keluar dari salah satu atau kedua payudara. Payudara dapat mengeluarkan cairan ini dengan atau tanpa stimulasi.

Gejala utama adalah keluarnya susu dari puting susu yang tidak terkait dengan menyusui. Debit dapat berasal dari salah satu atau kedua payudara.

7. Hipertiroidisme

Pada hipertiroidisme, kelenjar tiroid bekerja terlalu aktif dan menghasilkan hormon tiroid yang sangat banyak melebihi jumlah yang dibutuhkan oleh tubuh. Wanita biasanya lebih sering terkena hipertiroidisme dibandingkan pria. Hipertiroidisme juga lebih banyak ditemukan pada usia muda dibandingkan pada usia tua.

Hipertiroidisme paling sering disebabkan oleh penyakit Graves, gangguan autoimun. Penyebab lainnya adalah nodul tiroid, tiroiditis, mengkonsumsi terlalu banyak yodium, dan overmedicating dengan hormon tiroid sintetis ada hipertiroidisme.

Untuk menilai fungsi tiroid dilakukan pemeriksaan:

- TSH serum (biasanya rendah)
- T3, T4 (biasanya tinggi)

8. Hipoglikemia

Hipoglikemia adalah suatu keadaan dimana kadar gula darah (*glukosa*) secara abnormal rendah. Dalam keadaan normal, tubuh mempertahankan kadar gula darah antara 70-110 mg/dL.

Kadar gula darah yang rendah dapat menyebabkan gangguan fungsi berbagai sistem organ tubuh.

Gejala hipoglikemia jarang terjadi sebelum kadar gula darah mencapai 50 mg/dL. Diagnosis hipoglikemia ditegakkan berdasarkan gejala-gejalanya dan hasil pemeriksaan kadar gula darah. Pengobatan perlu diberikan jika gula darah mencapai dibawah 70 mg/dL.

9. Hipopituitarisme

Hipopituitarisme adalah suatu kondisi dimana kelenjar pituitary (atau disebut juga kelenjar hipofise) tidak dapat memproduksi satu atau lebih hormon dalam jumlah yang cukup.

Gejala hipopituitarisme bervariasi tergantung kepada jenis hormon apa yang kurang. Pada beberapa kasus, kelenjar pituitary hanya mengalami penurunan produksi hanya dari satu jenis hormon. Namun lebih sering terjadi penurunan produksi beberapa hormon dalam waktu bersamaan (panhipopituitarisme).

10. Hipotiroidisme

Hipotiroidisme adalah suatu keadaan dimana *kelenjar tiroid* kurang aktif dan menghasilkan terlalu sedikit *hormon tiroid*.

Penyebab hipotiroidisme yang paling sering adalah tiroiditis Hashimoto, yaitu gangguan kelenjar tiroid yang disebabkan oleh kelainan sistem imunitas tubuh yang menyerang kelenjar tiroid. Pada tiroiditis Hashimoto, kelenjar tiroid seringkali membesar dan hipotiroidisme terjadi beberapa bulan kemudian akibat rusaknya kelenjar tiroid. Penyebab hipotiroidisme tersering kedua adalah akibat efek pengobatan hipertiroidisme, baik dengan yodium radioaktif, pasca operasi pengangkatan kelenjar tiroid, atau pasca radiasi kelenjar tiroid.

Ekspresi wajah menjadi tumpul, suara menjadi serak dan berbicara menjadi lambat, kelopak mata menutup dan mata serta wajah menjadi bengkak. Banyak penderita yang mengalami penambahan berat badan, sembelit dan tidak tahan terhadap cuaca dingin. Rambut menjadi tipis, kasar dan kering; kulit menjadi kasar, kering, bersisik dan menebal.

11. Hiperaldosteronisme

Hiperaldosteronisme merupakan suatu keadaan dimana terjadi produksi hormon aldosteron yang berlebihan sehingga menyebabkan penahanan cairan di dalam tubuh, peningkatan tekanan darah, kelemahan, dan pada kasus yang jarang dapat juga terjadi periode kelumpuhan.

Kadar aldosteron yang tinggi dapat menyebabkan penurunan kadar kalium. Kadar kalium yang rendah bisa tidak bergejala, tetapi dapat menyebabkan timbulnya rasa lemas, kesemutan, kram otot, dan

periode kelumpuhan sementara. Beberapa penderita merasa sangat haus dan sering berkemih.

Pemeriksaan kelenjar adrenal akan adanya tumor yang bukan kanker (adenoma) dapat dilakukan jika terdapat produksi aldosteron yang berlebihan. Pemeriksaan dapat dilakukan dengan CT scan atau MRI, tetapi kadangkala contoh darah dari setiap kelenjar adrenal perlu diambil dan diperiksa untuk menentukan asal hormon [22].

12. Kanker tiroid

Kanker tiroid adalah suatu keganasan pada tiroid yang memiliki 4 tipe, yaitu papiler, folikuler, anaplastik atau meduler. Kanker jarang menyebabkan pembesaran kelenjar, lebih sering menyebabkan pertumbuhan kecil (*nodul*) di dalam kelenjar. Sebagian besar nodul tiroid bersifat jinak dan biasanya kanker tiroid bisa disembuhkan.

Pertanda awal dari kanker tiroid biasanya adalah benjolan yang tidak terasa nyeri di leher. Skening tiroid bisa menentukan apakah nodulnya berfungsi atau tidak, karena nodul yang tidak berfungsi cenderung bersifat ganas.

13. Kelenjar adrenal kurang aktif

Tubuh memiliki 2 kelenjar adrenal, masing-masing terletak di puncak ginjal. Kelenjar adrenal terdiri dari dua bagian, yaitu bagian korteks (luar) dan medulla (dalam).

Bagian dalam dari kelenjar adrenal (medula) menghasilkan katekolamin. Pada lapisan ini terdapat sel-sel kromafin, yaitu sel E yang menghasilkan epinefrin dan sel NE yang menghasilkan norepinefrin. Kedua hormon ini merupakan katekolamin yang bekerja mempengaruhi tekanan darah, denyut jantung, berkeringat dan aktivitas lainnya yang juga diatur oleh sistem saraf simpatis.

14. Penyakit Addison

Penyakit Addison adalah kelainan yang terjadi akibat kelenjar adrenal yang tidak dapat menghasilkan hormon-hormon dalam jumlah yang cukup. Kekurangan aldosteron menyebabkan tubuh mengeluarkan natrium yang banyak dan menahan kalium. Hal ini menyebabkan kadar natrium yang rendah dan kadar kalium yang tinggi di darah. Hal ini membuat penderita kencing secara berlebihan dan dapat terjadi dehidrasi. Dehidrasi hebat dan kadar sodium yang rendah juga dapat menyebabkan penurunan volume darah dan bisa menyebabkan shock.

Sekitar 80% penyakit Addison disebabkan oleh kelainan autoimun. Insufisiensi adrenal terjadi ketika kerusakan sudah mengenai minimal 90% korteks adrenal. Hal ini mengakibatkan, penurunan produksi kortisol dan aldosteron. Terkadang kelainan

autoimun ini hanya mengenai kelenjar adrenal saja, namun pada kasus lain, kelainan autoimun ini juga mengenai kelenjar-kelenjar endokrin lain, sehingga terjadi sindroma defisiensi poliendokrin.

15. Sindroma defisiensi poliglanduler

Sindroma Defisiensi Poliglanduler adalah keadaan dimana beberapa kelenjar endokrin menjadi kurang aktif dan menghasilkan hormon dalam jumlah yang kurang dari normal.

Kerusakan kelenjar endokrin dapat disebabkan oleh berbagai hal, seperti infeksi, berkurangnya aliran darah yang menuju ke kelenjar, dan tumor. Namun penyebab paling sering terjadinya sindroma ini adalah akibat reaksi autoimun yang menyebabkan peradangan dan menghancurkan sebagian atau seluruh kelenjar. Reaksi autoimun dapat dipicu oleh adanya infeksi virus atau antigen lain dari lingkungan. Adanya faktor genetik meningkatkan kemungkinan seseorang untuk terkena sindroma ini.

16. Sindroma sakit eutiroid

Sindroma sakit eutiroid (Euthyroid sick syndrome) merupakan suatu keadaan dimana kadar hormon tiroid di dalam tubuh rendah pada pasien-pasien dengan fungsi kelenjar tiroid yang normal (eutiroid) dan tanpa adanya penyakit tiroid.

Penyebab penyakit ini belum diketahui tetapi kemungkinan disebabkan oleh adanya gangguan konversi T₄ menjadi T₃, penurunan penurunan buangan rT₃ yang dihasilkan dari T₄, dan penurunan ikatan hormon tiroid ke TBG (thyroxine binding globulin). Sitokin peradangan seperti TNF alfa dan interleukin-1 mungkin berperan dalam perubahan ini.

17. Sindroma empty sella

Sella turcica adalah suatu struktur tulang berbentuk pelana yang terletak pada dasar tengkorak di tulang sfenoid, dimana terletak kelenjar hipofise (disebut juga kelenjar pituitary).

Sindroma empty sella (empty sella syndrome) merupakan suatu keadaan dimana sella turcica tidak sepenuhnya terisi oleh kelenjar hipofise sehingga sella turcica terisi sebagian atau seluruhnya oleh cairan serebrospinal. Keadaan ini dapat dilihat dengan bantuan pemeriksaan CT scan atau MRI dari sella tursika.

Untuk itu, pada kasus terjadinya empty sella diperlukan evaluasi yang meliputi pemeriksaan radiologis otak, pemeriksaan lapangan pandang, dan pemeriksaan apakah terdapat abnormalitas hormon.

18. Sindroma neoplasia endokrin multiple

Sindroma Neoplasia Endokrin Multipel adalah suatu penyakit keturunan yang jarang terjadi, dimana tumor jinak maupun tumor ganas tumbuh di beberapa kelenjar endokrin. Tumor pada penyakit ini

bisa timbul pada masa bayi maupun pada usia tua. Kelainan akibat neoplasia endokrin multipel terutama terjadi karena pembentukan hormon yang berlebihan oleh tumor.

Penyakit neoplasia endokrin multipel terjadi akibat adanya mutasi genetik, yaitu mutasi pada gen MEN1 pada neoplasia endokrin multipel (NEM) tipe I dan mutasi pada gen RET pada NEM tipe II. Gen MEN1 berfungsi dalam pengaturan produksi protein yang disebut menin. Menin bekerja sebagai supresor tumor, sehingga sel-sel tubuh dijaga untuk tetap dapat membelah dan tumbuh dengan normal dan terkendali.

19. Tiroiditis granulomatosa subakut

Tiroiditis Granulomatosa Subakut (Subacute granulomatous thyroiditis) atau disebut juga deQuervain's thyroiditis merupakan peradangan kelenjar tiroid yang biasanya terjadi setelah infeksi saluran pernafasan bagian atas.

Tiroiditis subakut biasanya terjadi paling sering pada wanita usia pertengahan dengan riwayat baru terkena infeksi virus pada saluran nafas. Gejala awal yang paling sering terjadi adalah timbulnya rasa nyeri pada kelenjar tiroid. Kelenjar tiroid menjadi lunak dan biasanya timbul demam yang tidak terlalu tinggi (37,2-38,3 oC).

Peradangan biasanya menyebabkan kelenjar tiroid menghasilkan terlalu banyak hormon tiroid, sehingga akan terjadi hipertiroidisme, yang hampir selalu diikuti oleh hipotiroidisme sementara. Banyak penderita merasakan kelelahan yang luar biasa. Kesembuhan total akan terjadi dalam waktu beberapa minggu atau bulan.

20. Tiroiditis Hashimoto

Tiroiditis Hashimoto (Tiroiditis autoimun) adalah peradangan kelenjar tiroid yang disebabkan oleh penyakit autoimun, dimana sistem imun tubuh menyerang kelenjar tiroid. Tiroiditis Hashimoto menyebabkan terjadinya hipotiroidisme, karena kelenjar tiroid tidak dapat cukup membuat hormon tiroid sesuai kebutuhan tubuh.

Penyebab mengapa terjadi reaksi autoimun sehingga muncul tiroiditis Hashimoto belum diketahui secara pasti. Sekitar 20% penderita tiroiditis Hashimoto memiliki kelenjar tiroid yang kurang aktif, sisanya memiliki kelenjar yang berfungsi normal. Banyak penderita yang juga memiliki kelainan endokrin lainnya (seperti diabetes (kencing manis), kelenjar adrenal yang kurang aktif atau kelenjar paratiroid yang kurang aktif) dan penyakit autoimun lainnya (misalnya anemia pernisiiosa, artritis reumatoid, sindroma Sjogren atau lupus eritematosus sistemik).

21. Tiroiditis limfositik laten

Tiroiditis Limfositik Laten atau biasa disebut juga Silent Lymphocytic Thyroiditis adalah peradangan pada kelenjar tiroid, yang disertai oleh penyusupan (infiltrasi) limfosit. Kata silent disini mengarah pada peradangan yang tidak disertai rasa nyeri.

Penyakit ini bersifat self-limited, atau akan hilang dengan sendirinya. Gangguan yang bersifat subakut paling sering terjadi pada wanita pada periode setelah melahirkan. Gejala-gejala yang muncul pada awalnya adalah gejala hipertiroidisme, kemudian hipotiroidisme, dan kemudian menyembuh ke keadaan eutiroid.

Penyakit ini dimulai pada masa postpartum (setelah melahirkan), biasanya dalam waktu 12-16 minggu. Tiroiditis limfositik laten memiliki gejala khas yaitu adanya pembesaran kelenjar tiroid yang tidak nyeri dengan adanya fase hipertiroidisme selama beberapa minggu, diikuti dengan fase hipotiroidisme akibat habisnya hormon tiroid, tetapi biasanya akan menyembuh ke keadaan eutiroid. Fase hipertiroidisme akan pulih kembali, dapat berlangsung singkat dan mungkin dapat terabaikan. Banyak penderita baru terdiagnosa saat mereka mengalami fase hipotiroidisme, yang kadang-kadang dapat bersifat permanen.

22. Tumor karsinoid

Tumor Karsinoid adalah suatu jenis kanker neuroendokrin yang tumbuh dengan lambat dan dapat terjadi pada berbagai tempat pada tubuh. Tumor karsinoid biasanya terjadi pada saluran pencernaan (lambung, usus halus, usus buntu, kolon, rektum) atau pada paru-paru.

Penyebab tumor karsinoid masih belum jelas. Secara umum, kanker terjadi karena adanya mutasi genetik pada sel sehingga terjadi pembelahan dan pertumbuhan sel-sel yang abnormal. Tumor karsinoid biasanya berasal dari sel-sel pembentuk hormon yang melapisi usus halus (sel-sel neuroendokrin) atau sel-sel lainnya pada saluran pencernaan, pankreas, buah zakar, indung telur atau paru-paru.

23. Virilisasi

Virilization (virilisasi) adalah terbentuknya sifat-sifat maskulin yang berlebihan, biasanya pada wanita, seringkali sebagai akibat kelenjar adrenal yang memproduksi androgen secara berlebihan (testosteron dan hormon-hormon sejenis).

Penyebab paling sering terjadinya virilisasi adalah pembesaran atau hiperplasi korteks adrenal. Terkadang virilisasi disebabkan oleh tumor (adenoma atau kanker) dari kelenjar adrenal. Pada kasus tertentu, virilisasi disebabkan oleh kanker di luar kelenjar adrenal yang memproduksi androgen. Penyebab virilisasi dari luar tubuh yaitu pada atlet-atlet yang memakai androgen dalam jumlah besar untuk

meningkatkan massa otot, juga dapat terjadi virilisasi. Pembesaran indung telur (ovarium) akibat tumor atau kista juga dapat menyebabkan terjadinya virilisasi, tetapi biasanya bersifat ringan.

2.9 JAVA

Java merupakan salah satu bahasa pemrograman yang berorientasi objek [35,6]. Dalam memecahkan masalah, java membagi program menjadi objek-objek, kemudian memodelkan sifat dan tingkah laku masing-masing. Selanjutnya, java menentukan dan mengatur interaksi antara objek yang satu dengan yang lainnya.

Java juga bersifat *multithreaded*, artinya dapat mengerjakan beberapa proses dalam waktu yang hampir bersamaan. Selain itu, java menggantikan konsep pewarisan lebih dari satu (*multiple inheritance*) dengan interface, menghilangkan konsep pointer yang sering membingungkan, otomatisasi sistem alokasi memory, dan sebagainya [35, 8]. Ini membuat java menjadi relatif sederhana dan mudah untuk dipelajari dibandingkan bahasa pemrograman lainnya.

2.9.1 Sejarah Java

Java adalah bahasa pemrograman yang disusun oleh James Gosling yang dibantu oleh rekan-rekannya seperti Patrick Naughton, Chris Warth, Ed Rank, dan Mike Sheridan di suatu perusahaan perangkat lunak yang bernama Sun Microsystems pada tahun 1991. Bahasa pemrograman ini mula-mula diinisialisasi dengan nama "Oak", namun pada tahun 1995 diganti namanya menjadi "Java" [35, 9].

2.9.2 Arsitektur Java

Secara arsitektur, Java tidak berubah sedikit pun semenjak awal mula bahasa tersebut dirilis. Kompiler Java (yang disebut dengan **javac** atau *Java Compiler*) akan mentransformasikan kode-kode dalam bahasa Java ke dalam suatu *bytecode*. *Bytecode* adalah sekumpulan perintah hasil kompilasi yang kemudian dapat dieksekusi melalui sebuah mesin komputer abstrak, yang disebut dengan JVM (*Java Virtual Machine*). JVM juga sering dinamakan sebagai *interpreter*, karena sifatnya yang selalu menerjemahkan kode-kode yang tersimpan dalam *bytecode* dengan cara baris demi baris [35, 11].

2.9.3 Teknologi Java

Java adalah suatu teknologi di dunia *software* komputer, yang merupakan suatu bahasa pemrograman, dan sekaligus suatu *platform*. Java mudah dipelajari, terutama bagi programmer yang telah mengenal C/C++. Sebagai bahasa pemrograman Java dirancang menjadi handal dan aman [35, 12]. Java juga dirancang agar dapat dijalankan di semua *platform*, dan juga dirancang untuk menghasilkan aplikasi-aplikasi dengan performansi yang terbaik, seperti aplikasi *database* Oracle 8i/9i yang *core*-nya dibangun menggunakan bahasa pemrograman Java. Sebagai sebuah *platform*, Java terdiri atas dua bagian utama, yaitu:

1. *Java Virtual Machine (JVM)*
2. *Java Application Programming Interface (Java API)*

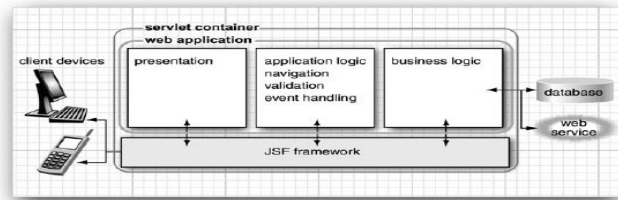
Sun membagi arsitektur Java menjadi tiga bagian, yaitu:

1. *J2EE (Java 2 Enterprise Edition)*
2. *J2SE (Java 2 Second Edition)*
3. *J2ME (Java 2 Micro Edition)*

2.10 JavaServer Faces

JavaServer Faces atau biasa disingkat JSF adalah *user interface framework* dalam bahasa Java untuk membangun aplikasi web [36, 6]. JSF merupakan salah satu bagian dari teknologi yang membangun *platform* Java EE. JSF diciptakan pada tahun 2002 melalui *Java Specification Request (JSR)* 127. JSR 127 kemudian mengalami banyak perubahan dan pada bulan Maret 2004 diluncurkan versi akhirnya [36, 8]. Salah satu kelebihan utama dari JSF adalah teknologi ini menawarkan pembagian yang jelas antara layer presentasi dan bisnis.

2.10.1 Arsitektur *JavaServer Faces*



Pada gambar 2.10.1 terlihat bahwa JSF bertanggung jawab dalam menangani interaksi klien dan aplikasi, menghubungkan bagian presentasi, logika aplikasi, dan bisnis logika menjadi suatu aplikasi web. Subsistem lainnya, seperti layanan EJB atau basis data dapat diintegrasikan dengan mudah walaupun bukan merupakan bagian dari JSF. JSF mengikuti arsitektur *Model-View-Controller (MVC)* dengan pembagian sebagai berikut:

1. *Model* adalah logika bisnis dan data yang bisa berupa EJB, basis data, atau yang lainnya.
2. *View* adalah layer presentasi yang berinteraksi langsung dengan pengguna. *View* bisa berupa JSP, atau teknologi *display* lainnya.
3. *Controller* adalah kode aplikasi yang menangani *events* dan menghubungkan *model* dan *view*. Dalam JSF, *servlet* berperan sebagai *controller*.

2.10.2 Elemen Pembentuk *Java Server Faces*

Seperti *framework* lainnya, JSF mempunyai elemen-elemen penyusun yang menyediakan rangkaian fitur yang dapat digunakan oleh pengembang aplikasi web [36, 10].

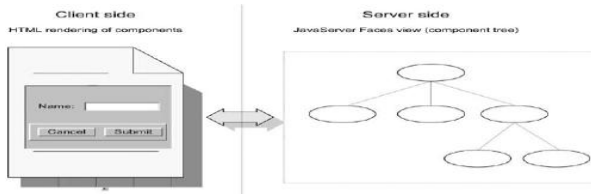
Pada dasarnya, terdapat delapan elemen utama yang perlu diketahui, yaitu:

1. **Komponen *User Interface (UI)*.**

Komponen UI adalah objek yang menangani interaksi dengan *end user*.

Komponen tersebut mirip dengan kontrol pada Swing yang dibangun di atas *JavaBeans*. Walaupun sama-sama memiliki properti, *method*, dan *events*, komponen UI didesain dengan batasan yang unik untuk aplikasi web, dan hidup di server, bukan di klien. Komponen pada JSF dapat mengingat nilai antara *request* yang satu dengan yang lain dikarenakan JSF menyimpan representasi pohon keterhubungan dari komponen-komponen pada halaman web [36, 10]. Pohon komponen ini disebut *view* dan merupakan representasi internal dari JSF terhadap suatu halaman web.

Pohon itu menggambarkan hubungan bapak-anak dari komponen. Sebagai contoh, sebuah *form* mengandung label, *tcontoh field*, dan *panel* dengan dua tombol seperti pada gambar 2.10.2. Dalam hal ini, komponen *form* merupakan bapak dari komponen label, *tcontoh field*, dan *panel*.



Gambar 2.10.2 *Komponen UI dikelola oleh server dalam bentuk pohon*

Semua komponen pada JSF merupakan turunan dari kelas *UIComponentBase* yang mendefinisikan status *default* dan perilaku dari komponen tersebut. Beberapa komponen baku telah disediakan oleh JSF antara lain *tcontoh tbox*, *panel*, *label*, *data grid*, *graphic*, *listbox*, *radio button*, *checkbox*, dsb. Selain komponen-komponen baku tersebut, masih ada komponen lainnya seperti *toolbar*, *menu*, *file upload*, *tabbed panel*, dsb. Komponen dapat dideklarasikan pada halaman *markup* sebagai berikut:

```
<h:inputTcontoh id="helloInput" value="default" required="true">
```

Komponen tersebut dapat dimanipulasi pada kode Java di server dengan cara sebagai berikut:

```
...
HtmlInputTcontoh input = (HtmlInputTcontoh)event.getSender();
input.setDisabled(true);
input.setStyle("color: blue");
...
```

2. Renderer.

Renderer bertanggung jawab dalam melakukan *encoding* dan *decoding*

komponen. *Encoding* adalah proses menampilkan komponen pada halaman web sehingga dapat dimengerti dan dilihat oleh pengguna, sedangkan *decoding* mengekstraksi parameter *request* yang benar dan kemudian mengeset *value* dari komponen berdasarkan parameter tersebut [36, 11]. Sebagai contoh, kode berikut mendefinisikan sebuah komponen Html Input Tcontoh.

```
<h:inputTcontoh id="inputTcontoh" size="20" maxlength="30"/>
```

Ketika di-*encode*, kode tersebut akan diubah menjadi kode HTML yang sesuai dengan teknologi *view* yang digunakan oleh pengguna.

```
<input id="myForm:inputTcontoh" type="tcontoh"
name="myForm:inputTcontoh" maxlength="30" size="20" />
```

Ketika pengguna memasukkan teks "foo" ke dalam komponen ini, maka proses *decoding* akan mengambil parameter *form* yang dikirim melalui HTTP *response* dan mengeset nilai dari instans komponen Html InputT contoh di server ke "foo". *Renderer* diorganisir ke dalam *render kits* yang umumnya dibedakan berdasarkan tipe keluarannya. JSF telah menyertakan *render kit* standar untuk HTML 4.01.

3. *Backing beans*.

Backing beans merupakan objek yang menangani interaksi antara *view* dan *model* pada arsitektur MVC. *Backing beans* umumnya mengandung property atau masukan dari user dan juga *event listener methods* yang memproses properti tersebut. JSF memungkinkan suatu *backing beans* memiliki asosiasi dengan komponen UI menggunakan JSF *contohpression language* (EL).

JSF EL cukup serupa dengan JSP 2.0 ataupun JSTLcontohpression language [36, 11]. Programmer dapat menghubungkan suatu properti *backing beans* dengan *value* dari suatu komponen seperti pada contoh di bawah.

```
<h:outputTcontoh id="helloBeanOutput"
value="#{helloBean.numControls}"/>
```

Kode di atas menunjukkan bahwa *programmer* mengasosiasikan (*bind*) properti *numControls* dari objek *helloBean* ke nilai komponen *outputTcontoh*. Apabila nilai pada komponen berubah, maka properti dari objek yang berasosiasi pun ikut berubah.

4. *Validators*.

Validators melakukan verifikasi terhadap nilai suatu komponen. Sebuahkomponen UI dapat berasosiasi dengan satu atau banyak jenis *validator*. JSF sendiri menangani validasi dengan tiga cara, yaitu pada level komponen UI, level *backing beans* melalui *validator method*, atau lewat kelas validator.

Komponen UI biasanya hanya menangani validasi yang sederhana seperti mengecek suatu nilai harus ada atau tidak. *Validator method* berguna apabila dibutuhkan validasi pada satu atau banyak *field* dalam suatu *form*. Contoh *validator* berguna untuk kasus yang generik misalnya panjang maksimal masukan atau kisaran angka masukan. *Validator* jenis ini bersifat *pluggable*, dapat diasosiasikan dengan banyak komponen. JSF menyertakan berbagai macam validator standar untuk mengecek kisaran atau panjang. Ketika ditemukan kesalahan, misalnya *string* yang terlalu panjang, maka *validator* akan menambahkan *error message* pada daftar *message*. Ini akan sangat memudahkan apabila pesan *error* ingin ditampilkan [36, 12].

```
<h:inputTcontoht>
  <f:validateLength minimum="2" maximum="10" />
</h:inputTcontoht>
```

Kode di atas memperlihatkan suatu komponen *HtmlInputTcontoht* yang berasosiasi dengan *Length validator* yang mengecek masukan dari pengguna agar panjangnya kurang dari dua dan tidak lebih dari sepuluh karakter.

5. Converters.

Untuk melakukan ini, *renderers* harus mempunyai pengetahuan mengenai komponen yang ditampilkan. Hal ini menjadi masalah karena komponen dapat berasosiasi dengan properti *backing beans* yang dapat memiliki beragam tipe, seperti *String*, *Date*, atau tipe bentukan lainnya.. *Converters* hadir untuk menangani masalah ini. *Converters* bertanggung jawab dalam mengubah nilai dari komponen ke bentuk *String* untuk ditampilkan, dan dari masukan *String* kembali ke objek semula. Selain itu, *converters* juga dapat menangani *formatting* dan *localization*. Sebagai contoh, *DateTime converter* dapat melakukan format pada objek *Date* dalam bentuk *short*, *medium*, *long*, atau *full style*. Untuk setiap bentuk, *converters* akan menampilkannya sesuai dengan *user's locale* [36, 13]. Berikut cara untuk me-*register* suatu *converter* ke dalam komponen.

```
<h:outputTcontoht
  value="#{user.dateOfBirth}">
  <f:convert_datetime type="both" dateStyle="full"/>
</h:outputTcontoht>
```

Misalkan tanggal lahir dari pengguna adalah 4 Mei 1942, maka komponen *HtmlOutputTcontoht* akan menampilkan *string* "05/04/42" apabila pengguna

berasal dari Inggris atau menampilkan “04/05/42” jika berasal dari Canada.

4. *Events dan listeners.*

Event merepresentasikan interaksi yang pengguna lakukan terhadap komponen UI. Suatu *event* dapat berupa memencet/mengklik suatu komponen, menyorot komponen, atau mengeksekusi suatu perintah yang spesifik. JSF menggunakan model *JavaBeans events* seperti *Swing*. Suatu objek dapat menciptakan banyak *events* yang kemudian akan ditangkap oleh *listeners*. *Listener* dapat diimplementasikan pada *backing beans method* atau kelas *listener* sendiri.

Contoh:

```
<h:commandButton type="submit"
value="Login"
action="#{loginForm.login}"/>
```

Kode di atas menunjukkan suatu komponen *CommandButton* yang meregister *action method* “login”. Ketika tombol diklik, maka *action event* akan dipanggil, dan fungsi “login” akan dieksekusi.

7. *Messages.*

JSF mempunyai *built-in support* untuk menampilkan pesan pada aplikasi. Pesan pada JSF dapat dihasilkan baik oleh *validator*, *converter*, komponen UI, *kode aplikasi*, ataupun *event listener*. Selain itu, pesan dapat diasosiasikan dengan komponen pada halaman web [36, 14].

Contoh:

```
<h:message id="errors" for="helloInput" style="color: red"/>
```

Tag pesan ini akan menampilkan semua error yang dihasilkan oleh komponen dengan id “helloInput”.

8. *Navigation.*

Navigation adalah proses perpindahan dari salah satu halaman web ke halaman lainnya. JSF mempunyai sistem navigasi yang elegan yang bersifat deklaratif. *Programmer* dapat mendefinisikan perpindahan ke suatu halaman melalui keluaran yang dihasilkan oleh suatu *action method* [36, 14].

```

<navigation-rule>
<from-view-id>/login.jsp</from-view-id>
<navigation-case>
<from-outcome>success</from-outcome>
<to-view-id>/mainmenu.jsp</to-view-id>
</navigation-case>
<navigation-case>
<from-outcome>failure</from-outcome>
<to-view-id>/login.jsp</to-view-id>
</navigation-case>
</navigation-rule>

```

Contoh di atas memperlihatkan dua kasus pada halaman *login.jsp*, yaitu untuk keluaran “*success*” dan “*failure*”. Untuk setiap halaman yang diberikan, *navigation rule* mendefinisikan keluaran dan halaman yang dipanggil sesuai dengan keluaran tersebut. Untuk satu hubungan antara keluaran dan halamannya didefinisikan dalam *navigation case*. Jika *action method* pada *login.jsp* menghasilkan string “*success*” maka navigasi akan memanggil halaman *mainmenu.jsp*, sebaliknya jika “*failure*” maka navigasi tetap di halaman *login.jsp*.

2.11 Primefaces

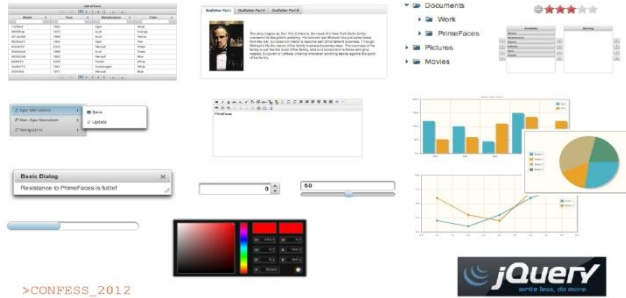
PrimeFaces adalah salah satu seperti sumber komponen terbuka yang berisi lebih dari ninety komponen, yang mencakup orang-orang seperti editor HTML, lightbox, Grid, Charts, Kalender, Penjadwal dan Panel Kontrol. PrimeFaces komponen yang mudah digunakan dan karena itu memberikan keuntungan yang sempurna untuk perusahaan pengembangan Java outsourcing untuk membuat aplikasi secara efisien dalam timeline berkomitmen [37, 20].

2.11.1 Fitur dan Manfaat PrimeFaces

Berbagai komponen PrimeFaces juga dibuat tersedia melalui topik grafis readymade yang membantu java pengembang untuk dengan mudah menerapkannya dalam sebuah proyek pengembangan perangkat lunak java tertentu. PrimeFaces memanfaatkan lengkap dari teknologi Ajax, yang berarti bahwa seseorang dapat bekerja dan tidak perlu kembali halaman tersebut. Bila dibandingkan dengan perpustakaan lain, PrimeFaces Ajaxowa bisa jauh lebih mudah terintegrasi. Alasannya karena tidak digunakan dalam ekstensi JSF, yang meliputi parser HTML, servlets, filter dan manajer negara.

2.11.2 Komponen User Interfaces Prmefaces

Primefaces menyediakan 100 lebih komponen user interfaces yang siap pakai hanya dengan menambahkan jaringan yang berukuran 1.7mb saja [38].



Gambar 2.11.2.2 Komponen Form Primefaces

2.11.3 Tema Primefaces

PrimeFaces menggunakan jQuery ThemeRoller kerangka tema CSS, dan datang dengan 30 + tema pra-desain yang dapat di download dan diterapkan dalam project aplikasi [38].



Gambar 2.11.3 Tema Primefaces

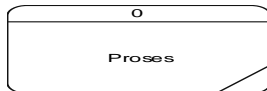
2.12 Teknik Data Flow Diagram

DFD merupakan suatu diagram aliran data yang berfungsi untuk menggambar aliran data serta proses yang mengubah dan mentransformasikan data melalui sistem. Sehingga dari sini kita dapat mengetahui jalannya suatu sistem informasi. Elemen-elemen lingkungan berada diluar batas sistem. Elemen-elemen ini menyediakan bagi sistem input data dan menerima output data sistem pada DFD, tidak dibuat perbedaan antara data dan informasi.

Semua arus dipandang sebagai data. Nama entitir digunakan untuk menggambarkan elemen-elemen lingkungan yang menandai titik-titik berakhir sistem. Entity digunakan dalam DFD dengan segi empat [39, 10].

2.12.1 Proses

Proses adalah sesuatu yang mengubah input menjadi output. Proses dapat digambarkan dengan lingkungan, segi empat horizontal tegak lurus dengan sudut-sudut yang membulat. Tiap simbol proses didefinisikan dengan label. Teknik pembuatan label yang paling umum adalah dengan menggunakan kata kerja dan obyek tetapi kita juga dapat menggunakan nama sistem atau program computer.

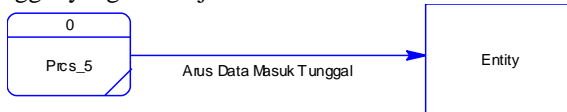


Gambar 2.12.1 Proses

2.12.2 Arus data

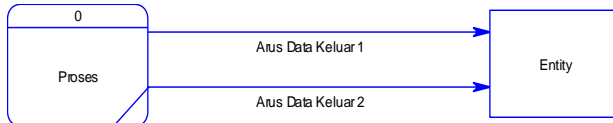
Arus data terdiri dari sekelompok elemen data yang saling berhubungan secara logis, yang bergerak dari satu titik proses lain. Tanda panah digunakan untuk menggambarkan arus tersebut. Arus data terdiri dari satu beberapa struktur data. Struktur data adalah sekelompok elemen data yang menggambarkan suatu hal atau transaksi tertentu. Adapun jenis arus data diantaranya adalah sebagai berikut :

1. Arus data tunggal yang menunjukkan ke suatu lokasi dari elemen tunggal.



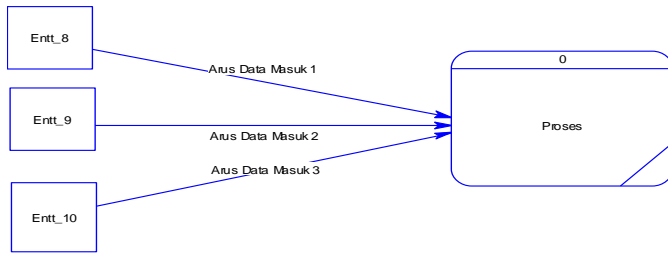
Gambar 2.12.2.1 Arus Data Masuk

2. Arus data dapat bercabang (*diverge*) ketika data yang sama bergerak ke beberapa lokasi dalam sistem.



Gambar 2.12.2.2 Arus Data Keluar

3. Arus data dapat juga memusat (*konvergen*) untuk menggambarkan beberapa arus data yang sama bergerak ke satu lokasi.



Gambar 2.12.2.3 Arus Data Konvergen

- Arus data dengan sistem dua arah untuk menggambarkan status pemesanan data dan sistem jawaban informasi yang diminta.

2.12.3 Penyimpanan Data

Jika data perlu dipertahankan karena suatu sebab, maka digunakan penyimpanan data (*data store*). Dalam istilah DFD, penyimpanan data digambarkan sebagai satu set garis parallel, segi empat terbuka atau bentuk lonjong. Dengan dilakukannya penyimpanan data ini diharapkan pada saat akhir suatu proses pembuatan laporan data-data yang dibutuhkan akan mudah untuk didapatkan sebagai upaya untuk pertanggungjawaban pada atasan.



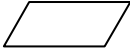
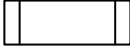
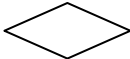


Gambar 2.12.3 Penyimpanan Data

2.13 Flowchart Program

Flowchart adalah gambaran dalam bentuk diagram alir dari algoritma-algoritma dalam suatu program, yang menyatakan arah alur program tersebut [39,16]. Berikut adalah beberapa simbol yang digunakan dalam menggambar suatu flowchart :

Tabel 2.13 – Flowchart

SIMBOL	NAMA	FUNGSI
	TERMINATOR	Permulaan/akhir program
	GARIS ALIR (FLOW LINE)	Arah aliran program
	PREPARATION	Proses inisialisasi/pemberian harga awal
	PROSES	Proses perhitungan/proses pengolahan data

	INPUT/OUTPUT DATA	Proses input/output data, parameter, informasi
	PREDEFINED PROCESS (SUB PROGRAM)	Permulaan sub program/proses menjalankan sub program
	DECISION	Perbandingan pernyataan, penyeleksian data yang memberikan pilihan untuk langkah selanjutnya
	ON PAGE CONNECTOR	Penghubung bagian-bagian flowchart yang berada pada satu halaman
	OFF PAGE CONNECTOR	Penghubung bagian-bagian flowchart yang berada pada halaman berbeda

2.14 Database

Salah satu komponen penting dalam sistem adalah database, karena merupakan dasar dalam menyediakan informasi. Dengan database informasi dapat disampaikan lebih akurat, tepat pada waktunya dan relevan. Selain itu juga dapat mengurangi duplikasi data, meningkatkan hubungan data dan mengurangi pemborosan tempat simpanan luar [40, 8].

2.14.1 Pengertian Database

Database adalah sekumpulan data yang terdiri dari suatu atau lebih table yang saling berhubungan. User mempunyai wewenang untuk mengakses data tersebut, baik untuk menambah, mengubah atau menghapus data yang ada dalam tabel tersebut [40, 8].

Database digunakan untuk menampung beberapa tabel atau query yang dijadikan media untuk menyimpan data sebagai sumber pengolahan data.

Menurut Wahyono [40, 9], database merupakan kumpulan data yang terorganisasi dalam file-file terstruktur yang khusus digunakan untuk menampung data.

2.14.2 Teknik Perancangan Basis Data

Pokok pemikiran dalam merancang *database* adalah bagaimana merancang database sehingga dapat memenuhi kebutuhan saat ini dan

kemudahannya untuk dikembangkan dimasa yang akan datang. Perancangan model konseptual perlu dilakukan disamping perancangan secara fisik. Pada perancangan konseptual, digunakan beberapa konsep pendekatan relasional namun tidak berarti konsep ini harus diimplementasikan ke model relasional saja tetapi juga apat dengan model Hirarchi dan model Network.

Tugas merancang database adalah bagian dari tugas database administrator Model konseptual mengkombinasikan beberapa cara untuk memproses data dan untuk beberapa aplikasi. Model konseptual tidak tergantung aplikasi tertentu dan tidak tergantung DBMS, Hardware yang digunakan.

Pada perancangan model konseptual tinjauan dilakukan pada struktur data dan relasi antar file menggunakan model dan relasional. [40, 24].

Terdapat dua teknik dalam merancang database yaitu :

- Teknik Normalisasi
- Teknik Entity Relationship

2.14.3 Teknik Normalisasi

Cara ini dimulai dari dokumen dasar yang sudah ada pada sistem atau sudah dipakai sistem tersebut, data-data pada dokumen dasar tersebut dipisahkan menjadi file-file yang tiap field pada file tersebut bergantung penuh pada kunci utama (field kunci) yang biasanya dikenal dengan bentuk normal ketiga.

Kemudian setiap file dalam database tersebut ditentukan hubungannya dengan file-file yang lain dengan cara memasang field tamu pada file-file anak atau file konektor [40, 25].

2.14.4 Teknik Entity Relationship

Langkah ini sering digunakan pada rancangan sistem, dimulai dengan pembuatan diagram arus data yang menghasilkan kamus data yang merupakan daftar semua elemen/field yang dibutuhkan dalam sistem tersebut. Dari field-field tersebut dipilih field kunci yang bersifat unik artinya keseluruhan record tersebut yang mana elemen/field dalam field tersebut bergantung penuh dengan field kunci tersebut. Setelah membuat table baru ditentukan relasi dari tiap table tersebut seperti halnya teknik normalisasi [40, 25].

2.15 MySQL

MySQL merupakan software sistem manajemen database (Relational Database Management System - RDBMS) yang sangat populer di kalangan programmer web, terutama di lingkungan linux dengan menggunakan script PHP dan perl. Software database ini kini telah tersedia juga pada platform system operasi Windows [41, 2].

Menurut Haris Saputro [42, 1], MySQL merupakan database server dimana pemrosesan data terjadi di server. Dan client hanya mengirim data serta meminta data. Oleh karena pemrosesan terjadi di server sehingga pengaksesan data tidak terbatas. Pengaksesan dapat dilakukan dimana saja oleh siapa saja dengan catatan komputer telah terhubung ke server.

2.16 Certainty Factor

Metode yang digunakan pada pembuatan sistem pakar penyakit gangguan sistem hormon manusia ini adalah metode certainty factor yaitu menghitung tingkat derajat keyakinan terhadap suatu fakta atau aturan. Berikut konsep dari metode certainty factor. Penggunaan *certainty factor* dilakukan untuk:

- menentukan nilai keyakinan atas fakta awal yang diberikan oleh User.
- menentukan nilai keyakinan atas konklusi yang diperoleh dari aturan, nilai ini ditentukan oleh pakar ke dalam aturan.
- menentukan nilai keyakinan atas fakta-fakta dan *goal* yang diperoleh selama proses penalaran dari hasil eksekusi aturan.
- menyesuaikan nilai keyakinan atas fakta atau *goal* yang diperoleh dari aturan yang berbeda tetapi menghasilkan konklusi yang sama.

Rumus Certainty factor :

$$CF[h,e] = MB[h,e] - MD[h,e]$$

Dengan:

CF[h,e] : faktor kepastian

MB[h,e] : Measure Belief / ukuran kepercayaan terhadap hipotesis h

MD[h,e] : Measure Disbelief / ukuran ketidakpercayaan terhadap evidence h

Nilai ukuran kepercayaan dari MB/MD

- 0 - 0.2, Tidak Tahu/Tidak Ada
- 0.4, Mungkin
- 0.6, Kemungkinan Besar
- 0.8, Hampir Pasti
- 1.0, Pasti

2.16.1 Sejarah Metode Certainty Factor

Faktor kepastian (certainty factor) diperkenalkan oleh Shortliffe Buchanan dalam pembuatan MYCIN pada tahun 1975 untuk mengakomodasi ketidakpastian pemikiran (incomplete reasoning) seorang pakar. Teori ini berkembang bersamaan dengan pembuatan sistem pakar MYCIN. Tim pengembang MYCIN mencatat bahwa dokter sering kali menganalisa informasi yang ada dengan ungkapan seperti misalnya: mungkin, kemungkinan besar, hampir pasti. Untuk mengakomodasi hal ini tim MYCIN menggunakan certainty factor (CF) guna menggambarkan tingkat keyakinan pakar terhadap permasalahan yang sedang dihadapi. Secara umum, rule direpresentasikan dalam bentuk sebagai berikut (John Durkin, 1994) [43, 18] :

$$\text{IF } E1 \text{ [AND / OR] } E2 \text{ [AND / OR] } \dots E_n \\ \text{THEN } H \text{ (CF = CF}_i\text{)}$$

dimana:

E1 ... E_n : fakta – fakta (evidence) yang ada.

H : hipotesa atau konklusi yang dihasilkan.

CF : tingkat keyakinan (Certainty Factor) terjadinya hipotesa H akibat adanya fakta – fakta E1 s/d En .

2.16.2 Aturan Certainty Factor

Definisi menurut Kusri Certainty Factor adalah suatu metode untuk membuktikan apakah suatu fakta itu pasti ataukah tidak pasti yang berbentuk metric yang biasanya digunakan dalam sistem pakar. Metode ini sangat cocok untuk sistem pakar yang mendiagnosis sesuatu yang belum pasti [43, 20].

Aturan metode Certainty Factors:

1. Kusri menggambarkan aturan untuk menambahkan dua faktor Certainty positif adalah:
 2. Aturan untuk menambahkan dua Certainty yang negatif adalah:
 3. Aturan untuk menambahkan Certainty Factors positif dan Certainty Factors negatif lebih kompleks:
- 3 aturan ini menyediakan suatu skala interval untuk Certainty Factors.

Contoh untuk fakta yang positif:

Strong suggestive (CFa): 0.8
Suggestive (CFb) : 0.6
CFcombine (CFa CFb) = 0.8 + 0.6 (1-0.8) = 0.92

Contoh untuk fakta yang negatif:

Strong suggestive (CFc): -0.8
Suggestive (CFd) : -0.6
CFcombine (CFc CFd) = -0.8 + -0.6 + -0.8 * -0.6 = -0.92

Contoh untuk fakta yang positif dan negatif:

Certainty factor adalah 0.88 (CFe)
Certainty factor against adalah 0.90 (CFf)

Metode certainty factors ini hanya bisa mengolah 2 bobot dalam sekali perhitungan. Untuk bobot yang lebih dari 2 banyaknya, untuk melakukan perhitungan tidak terjadi masalah apabila bobot yang dihitung teracak, artinya tidak ada aturan untuk mengkombinasikan bobotnya, karena untuk kombinasi seperti apapun hasilnya akan tetap sama.

Untuk mengetahui apakah seorang pasien tersebut menderita penyakit jantung atau tidak, itu dilihat dari hasil perhitungan bobot setelah semua keluhan-keluhan diinputkan dan semua bobot dihitung dengan menggunakan metode certainty factors. Pasien yang divonis mengidap penyakit jantung adalah pasien yang memiliki bobot mendekati +1 dengan keluhan-keluhan yang dimiliki mengarah kepada penyakit jantung. Sedangkan pasien yang

mempunyai bobot mendekati -1 adalah pasien yang dianggap tidak mengidap penyakit jantung, serta pasien yang memiliki bobot sama dengan 0 diagnosisnya tidak diketahui atau unknown atau bisa disebut dengan netral.

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

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

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

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

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

factor sebuah aturan JIKA E MAKA H adalah seperti ditunjukkan oleh persamaan 2 berikut:

Dimana

CF(E,e) : certainty factor evidence E yang dipengaruhi oleh evidence e

CF(H,E) : certainty factor hipotesis dengan asumsi evidence diketahui dengan pasti, yaitu ketika $CF(E, e) = 1$

CF(H,e) : certainty factor hipotesis yang dipengaruhi oleh evidence e Jika semua evidence pada antecedent diketahui dengan pasti maka persamaannya akan menjadi:

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

Dalam aplikasinya, CF(H,E) merupakan nilai kepastian yang diberikan oleh pakar terhadap suatu aturan, sedangkan CF(E,e) merupakan nilai kepercayaan yang diberikan oleh pengguna terhadap gejala yang dialaminya. Sebagai contoh, berikut ini adalah sebuah aturan dengan CF yang diberikan oleh seorang pakar:

**JIKA batuk
DAN demam
DAN sakit kepala
DAN bersin-bersin
MAKA influenza, CF: 0,7**

2.16.3 Kelebihan dan Kekurangan Metode Certainty Factors

a. Kelebihan metode Certainty Factors adalah:

1. Metode ini cocok dipakai dalam sistem pakar untuk mengukur sesuatu apakah pasti atau tidak pasti dalam mendiagnosis penyakit sebagai salah satu contohnya.
2. Perhitungan dengan menggunakan metode ini dalam sekali hitung hanya dapat mengolah dua data saja sehingga keakuratan data dapat terjaga.

b. Kekurangan metode Certainty Factors adalah:

Ide umum dari pemodelan ketidakpastian manusia dengan menggunakan numerik metode certainty factors biasanya diperdebatkan. Sebagian orang akan membantah pendapat bahwa formula untuk metode certainty factors diatas memiliki sedikit kebenaran. Metode ini hanya dapat mengolah ketidakpastian/kepastian hanya 2 data saja. Perlu dilakukan beberapa kali pengolahan data untuk data yang lebih dari 2 buah [43, 24].

Halaman ini sengaja di kosongkan