



### **Hak cipta dan penggunaan kembali:**

Lisensi ini mengizinkan setiap orang untuk menggubah, memperbaiki, dan membuat ciptaan turunan bukan untuk kepentingan komersial, selama anda mencantumkan nama penulis dan melisensikan ciptaan turunan dengan syarat yang serupa dengan ciptaan asli.

### **Copyright and reuse:**

This license lets you remix, tweak, and build upon work non-commercially, as long as you credit the origin creator and license it on your new creations under the identical terms.

## BAB II

### LANDASAN TEORI

#### 2.1 American Society of Anesthesiologists Physical Status Classification

Klasifikasi American Society of Anesthesiologists (ASA) Physical Status (PS) dikembangkan pada tahun 1941 oleh tiga dokter, yaitu Meyer Saklad, Emery Rovenstine, dan Ivan Taylor (Daabiss, 2011).

Tujuan awal dari klasifikasi ini adalah untuk mengumpulkan dan membandingkan data statistik anestesi. Saat ini, klasifikasi ASA PS digunakan untuk pengelompokan menurut keparahan penyakit dan kesehatan secara keseluruhan yang secara rutin dilakukan saat evaluasi pra-operasi di seluruh dunia (Aronson dkk, 2003).

Selain itu klasifikasi ASA PS juga digunakan untuk pembuatan kebijakan, evaluasi kinerja sebagai alat yang mudah untuk audit, alokasi sumber daya, dan penggantian pelayanan anestesi (Daabiss, 2011).

Tabel 2.1 merupakan definisi dari 6 klasifikasi ASA PS ([www.asahq.org](http://www.asahq.org)).

Tabel 2.1 Definisi Klasifikasi ASA PS

Klasifikasi ASA PS	Definisi
ASA I	Seorang pasien normal dan sehat
ASA II	Seorang pasien dengan penyakit sistemik ringan
ASA III	Seorang pasien dengan penyakit sistemik berat
ASA IV	Seorang pasien dengan penyakit sistemik berat yang merupakan ancaman konstan untuk hidup
ASA IV	Seorang pasien dengan penyakit sistemik berat yang merupakan ancaman konstan untuk hidup

Tabel 2.1 Definisi Klasifikasi ASA PS (Lanjutan)

Klasifikasi ASA PS	Definisi
ASA V	Seorang pasien sekarat yang tidak diharapkan untuk bertahan hidup tanpa operasi
ASA VI	Seorang pasien yang dinyatakan mati otak yang organnya sedang dilepas untuk tujuan donor

## 2.2 Sistem Pakar

Sistem pakar adalah sistem berbasis komputer yang menggunakan pengetahuan, fakta, dan teknik penalaran dalam memecahkan masalah yang biasanya hanya dapat dipecahkan oleh seorang pakar dalam bidang tersebut (Martin dan Oxman, 1998).

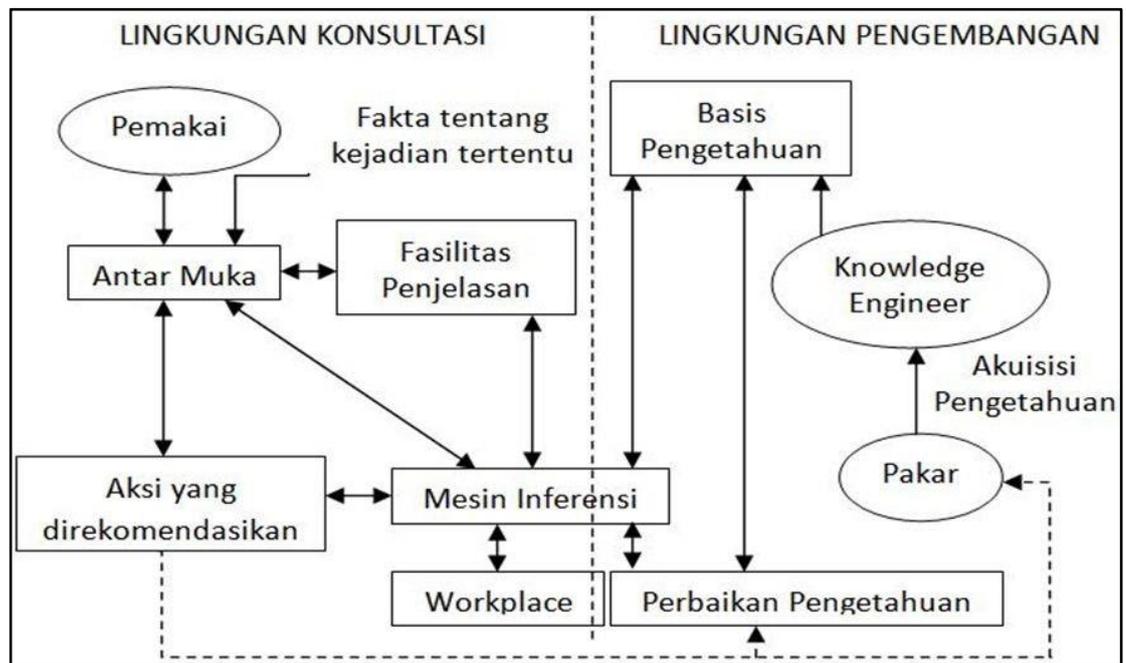
Pada dasarnya sistem pakar diterapkan untuk mendukung aktivitas pemecahan masalah. Beberapa aktivitas pemecahan yang dimaksud adalah pembuatan keputusan, pemaduan pengetahuan, pembuatan desain, perencanaan, prakiraan, pengaturan, pengendalian, diagnosis, perumusan, penjelasan, pemberi nasihat, dan pelatihan. Selain itu sistem pakar juga dapat berfungsi sebagai asisten yang pandai dari seorang pakar (Martin dan Oxman, 1998).

Ada beberapa manfaat dari sistem pakar, antara lain (Kusumadewi, 2003).

1. Memungkinkan orang awam bisa mengerjakan pekerjaan para ahli.
2. Menyimpan pengetahuan dan keahlian para pakar.
3. Meningkatkan output dan produktivitas.
4. Meningkatkan kualitas.
5. Mampu mengambil dan melestarikan keahlian para pakar.
6. Mampu beroperasi dalam lingkungan berbahaya.
7. Memiliki kemampuan untuk mengakses pengetahuan.

8. Memiliki realibilitas.
9. Meningkatkan kapabilitas sistem computer.
10. Sebagai media pelengkap dalam pelatihan.
11. Meningkatkan kapabilitas dalam penyelesaian masalah.
12. Menghemat waktu dalam pengambilan keputusan.

Gambar 2.1 merupakan arsitektur sistem pakar menurut Turban (1995).



Gambar 2.1 Arsitektur Sistem Pakar (Turban, 1995)

Berikut ini adalah penjelasan dari setiap komponen yang terdapat pada Gambar 2.1 menurut Yuwono (2010).

1. Akuisisi pengetahuan

Akuisisi pengetahuan adalah akumulasi, transfer, dan transformasi keahlian dalam menyelesaikan masalah dari sumber pengetahuan ke program computer.

Pada tahap ini, *knowledge engineer* berusaha menyerap pengetahuan untuk selanjutnya di-*transfer* ke dalam basis pengetahuan.

2. Basis pengetahuan

Berisi pengetahuan-pengetahuan yang dibutuhkan untuk memahami, memformulasikan dan menyelesaikan masalah.

3. Mesin Inferensi

Program yang berisi metodologi yang digunakan untuk melakukan penalaran terhadap informasi-informasi dalam basis pengetahuan dan blackboard, serta digunakan untuk memformulasikan konklusi.

4. Workplace

Merupakan area dari sekumpulan memori kerja.

5. Antarmuka

Digunakan untuk media komunikasi antara user dan program.

6. Fasilitas penjelasan

Fasilitas penjelasan memungkinkan pengguna untuk mendapatkan penjelasan dari hasil konsultasi.

7. Perbaikan pengetahuan

Sistem ini digunakan untuk mengevaluasi kinerja pakar itu sendiri untuk melihat apakah pengetahuan-pengetahuan yang ada masih cocok untuk digunakan di masa mendatang.

### 2.3 Certainty Factor

Certainty Factor (CF) merupakan parameter klinis yang diberikan MYCIN untuk menunjukkan besarnya kepercayaan. CF menunjukkan ukuran kepastian terhadap suatu fakta dan aturan (Kusumadewi, 2003).

Konsep dari Certainty Factor adalah untuk mengakomodasi ketidakpastian dari seorang pakar yang sering menganalisa informasi dengan frasa seperti

mungkin, mungkin sekali, hampir pasti, dsb. Notasi Certainty Factor adalah sebagai berikut (Munandar dkk, 2012).

$$CF[h, e] = MB[h, e] - MD[h, e] \quad \dots \text{Rumus(2.1)}$$

Keterangan rumus:

CF[h,e] = faktor kepastian

MB[h,e] = *measure of belief*, kepercayaan hipotesis h, jika diberikan *evidence* e antara 0 dan 1

MD[h,e] = *measure of disbelief*, ketidakpercayaan hipotesis h, jika diberikan *evidence* e antara 0 dan 1

Terdapat situasi dimana lebih dari satu *evidence* menghasilkan CF untuk fakta yang sama. Pada situasi seperti ini, CF harus dikombinasikan. Jika 2 *rule* mendukung hipotesa yang sama, maka akan menambahkan nilai *belief* dalam hipotesa tersebut. Rumus dari kombinasi CF adalah sebagai berikut (Grosan dan Abraham, 2011).

$$CF(H, e1 \wedge e2) = \begin{cases} CF(e1) + CF(e2)(1 - CF(e1)), & \text{jika } CF(e1), CF(e2) > 0 \\ CF(e1) + CF(e2)(1 + CF(e1)), & \text{jika } CF(e1), CF(e2) < 0 \\ \frac{CF(e1) + CF(e2)}{1 - \min\{|CF(e1)|, |CF(e2)|\}}, & \text{jika tanda}(CF(e1)) \neq \text{tanda}(CF(e2)) \end{cases} \dots \text{Rumus(2.2)}$$

Nilai CF direpresentasikan pada Tabel 2.2.

Tabel 2.2 Representasi Nilai CF (Puspitasari, 2013)

Uncertainty Term	CF
Pasti tidak	-1.0
Hampir pasti tidak	-0.8
Kemungkinan besar tidak	-0.6
Mungkin tidak	-0.4
Tidak tahu	-0.2 sampai 0.2
Mungkin	0.4
Kemungkinan besar	0.6
Hampir pasti	0.8
Pasti	1.0

Berikut ini adalah contoh kasus Certainty Factor (Munandar, 2012).

Berdasarkan gejala yang diderita pasien, diketahui bobot nilai gejala 1 ( $CF(e1) = 0.35$ ), bobot nilai gejala 2 ( $CF(e2) = 0.36$ ), dan bobot nilai gejala 3 ( $CF(e3) = 0.24$ ) untuk penyakit anemia.

Karena semua nilai CF positif, maka digunakan persamaan pertama dari Rumus 2. Kombinasi dari nilai-nilai CF yang diperoleh adalah sebagai berikut.

$$\begin{aligned}CF(\text{comb}) &= CF(e1) + CF(e2)(1-CF(e1)) \\ &= 0.35 + 0.36(1-0.35) \\ &= 0.58\end{aligned}$$

$$\begin{aligned}CF(\text{comNew}) &= CF(\text{comb}) + CF(e3)(1-CF(\text{comb})) \\ &= 0.58 + 0.24(1-0.58) \\ &= 0.68\end{aligned}$$

Dari nilai CF yang didapat dari perhitungan tersebut, dapat disimpulkan kemungkinan pasien terkena penyakit anemia sebesar 0.68. Jika dihitung menjadi persentase adalah  $0.68 * 100\% = 68\%$ . Diketahui nilai CF pasien terkena penyakit gagal jantung untuk gejala yang sama adalah 90%. Nilai CF terbesar mengindikasikan bahwa penyakit tersebut lebih dekat dengan hasil keyakinan yang terbaik berdasarkan gejala yang dirasakan pengguna. Oleh karena itu, dapat disimpulkan bahwa pasien tersebut terkena penyakit gagal jantung.

## 2.4 Android

Android merupakan sebuah sistem operasi yang berbasis pada linux yang telah dimodifikasi. Android dikembangkan pertama kali pada tahun 2003 oleh sebuah perusahaan bernama Android Inc (Safaat, 2012). Pada tahun 2005, Google

membeli Android dan mengambil alih pengembangan aplikasinya beserta tim pengembangannya sebagai bagian dari strategi untuk memasuki pasar mobile (Lee, 2012).

Menurut Safaat (2012: hal 3), Android merupakan platform yang lengkap, terbuka, dan bebas. Penjelasanannya adalah sebagai berikut.

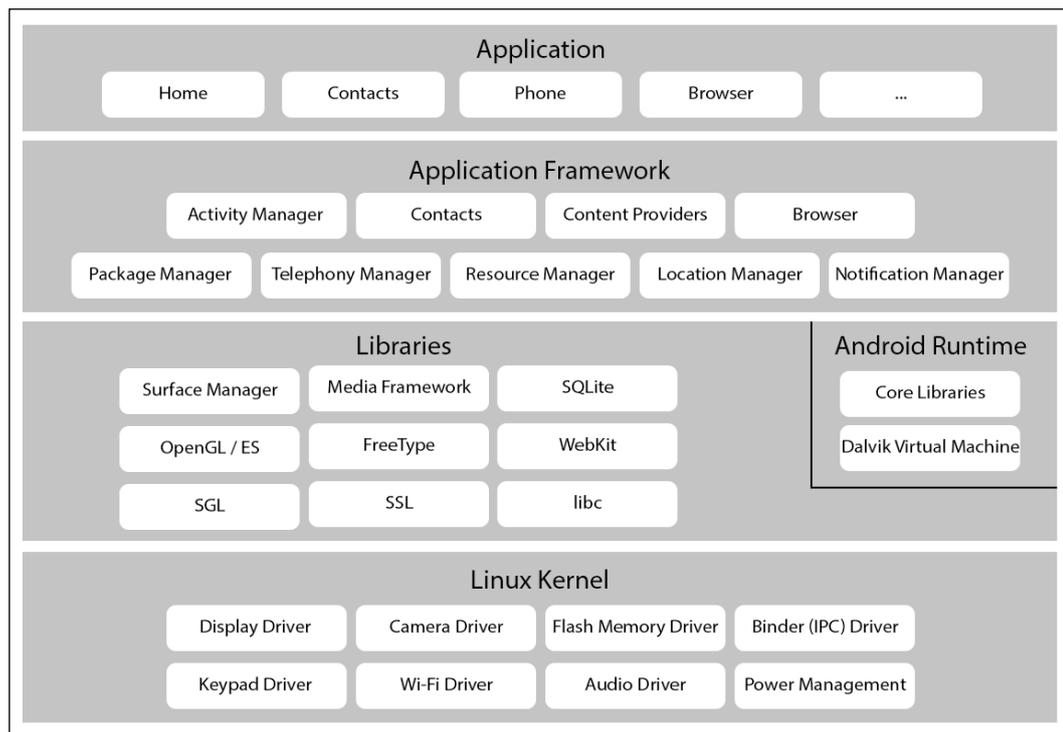
1. Lengkap: Para desainer dapat melakukan pendekatan yang komprehensif ketika mereka sedang mengembangkan *platform* Android. Android merupakan sistem operasi yang aman dan banyak menyediakan *tools* dalam membangun software dan memungkinkan untuk peluang pengembangan aplikasi.
2. Terbuka: *Platform* Android disediakan melalui lisensi *open source*.
3. Free: Android adalah *platform* yang bebas untuk develop. Tidak ada lisensi atau biaya royalti untuk dikembangkan pada *platform* Android. Tidak ada biaya keanggotaan diperlukan, tidak diperlukan biaya pengujian, tidak ada kontrak yang diperlukan. Aplikasi untuk Android dapat didistribusikan dan diperdagangkan dalam bentuk apa pun.

Ada beberapa fitur dari Android, antara lain sebagai berikut (Lee, 2012).

1. Penyimpanan, menggunakan SQLite, sebuah basis data ringan untuk penyimpanan data.
2. Konektivitas, mendukung GSM/EDGE, IDEN, CDMA, EV-DO, UMTS, Bluetooth, Wi-Fi, LTE, WiMAX
3. Pengiriman pesan, mendukung SMS dan MMS.
4. Web Browser, berbasis WebKit bersama dengan JavaScript V8 Chrome Engine.

5. Media support, mendukung media berikut: H.263, H.264, MPEG-4 SP, AMR, AMR-WB, AAC, HE-AAC, MP3, MIDI, Ogg Vorbis, WAV, JPEG, PNG, GIF, dan BMP.
6. Hardware support, mendukung sensor accelerometer, kamera, kompas digital, sensor proximity, dan GPS.
7. Multi touch pada layar sentuh.
8. Multi tasking untuk aplikasi.
9. Tethering, mendukung pembagian sambungan internet sebagai hotspot dengan atau tanpa kabel.

Gambar 2.2 merupakan berbagai layer yang membentuk sistem operasi Android (Lee,2012).



Gambar 2.2 Layer dalam Sistem Operasi Android

Sistem operasi Android dibagi menjadi 5 bagian dalam 4 layer, yaitu.

1. Linux Kernel — Layer ini adalah kernel yang menjadi basis dari Android. Layer ini berisi semua perangkat *low-level driver* untuk berbagai komponen hardware dari sebuah perangkat Android.
2. Libraries — Layer ini berisi semua code yang menyediakan fitur utama dari sistem operasi Android. Contohnya adalah SQLite *library* yang menyediakan dukungan basis data sehingga aplikasi dapat menggunakannya untuk penyimpanan data.
3. Android Runtime — Berada di layer yang sama dengan *libraries*, Android Runtime menyediakan set inti *library* yang memungkinkan *developers* untuk menulis aplikasi Android menggunakan bahasa pemrograman Java.
4. Application Framework — Mengekspos berbagai kemampuan dari sistem operasi Android untuk aplikasi *developers* sehingga mereka dapat menggunakannya di aplikasi mereka.
5. Application — Pada layer paling atas ini, terdapat aplikasi yang sudah ada sebelumnya dari perangkat Android (Phone, Contact, Browser, dll) maupun aplikasi yang diunduh dan dipasang dari Android Market.