

# Software

Disiapkan oleh Umi Proboyekti, S.Kom, MLIS

## Pengantar

Software atau computer software atau perangkat lunak komputer adalah suatu program yang membuat komputer mengerjakan tugas tertentu. Software terdiri dari urutan perintah untuk mengubah status hardware. Perintah-perintah tersebut menjadi penghubung antara data dan perangkat keras.

Software dibangun melalui pemrograman, yaitu proses menyusun sekumpulan perintah untuk komputer mengerjakan sesuatu. Kumpulan perintah-perintah tersebut disebut bahasa pemrograman. Perintah yang dapat diterima komputer berupa nilai biner 0 atau 1 saja. Perintah ini tidak mudah untuk disusun oleh orang, karena itu dibuatlah bahasa-bahasa pemrograman yang lebih mudah dimengerti oleh orang.

## Bahasa-bahasa Pemrograman

Bahasa-bahasa tersebut punya tingkatan berdasarkan kemudahannya dimengerti oleh manusia atau kedekatannya dengan bahasa alami yang dimengerti manusia.

1. **lowest level/tingkat 1**: bahasa mesin, berupa bilangan hexadecimal (0, 1, 2, 3, 4, 5,6,7,8,9,A,B,C,D,E,F) kemudian bilangan tersebut diterjemahkan menjadi bilangan biner: 0 dan 1.
2. **second level/tingkat 2**: bahasa assembly yang diwakili dengan kode-kode. Contoh:

```
Code_seg      Segment
               Assume      cs:Code_seg
               Org          100h

Start:
    mov  AH,2      ; register AH, diisi dengan 2, service 2
    mov  DL, 41h   ; register DL, diisi dengan 41h =65=A
    int  21h       ; perintah cetak

    int  20h       ; kembali ke kondisi awal (prompt)
Code_seg      Ends
               End start
```

Hasil tampilan : A

Dengan bahasa assembly penulisan program menjadi lebih cepat dan relatif lebih mudah dimengerti oleh manusia dari pada bahasa mesin. Pemrogram yang bekerja langsung dengan CPU akan menggunakan bahasa ini. Bahasa ini digunakan dalam membangun program untuk sistem tertanam atau embedded system. Misalnya program yang tertanam pada mesin cuci atau AC.

3. **High Level language/third level language/bahasa tingkat tinggi/bahasa tingkat 3**: kumpulan perintah dalam bahasa pemrograman tidak langsung untuk memprogram CPU tapi lebih mudah dimengerti oleh pemrogram. Contoh bahasa pemrograman pada tingkat ini adalah C, Pascal, COBOL, BASIC, Fortran yang

merupakan bahasa tingkat tinggi generasi awal. Sedangkan contoh yang lebih modern adalah: Java dan C++. Java dan C++ adalah bahasa yang menganut teknologi berorientasi objek. Java memang diciptakan untuk pemrograman berorientasi objek (OOP= Object Oriented Programming), sedangkan C++ masih dapat digunakan untuk pemrograman terstruktur dan pemrograman berorientasi objek.

Contoh program dalam bahasa C:

```
main()
{
    int a, b, sum;

    a=15;
    b=20;
    sum = a+b;
    if (sum !=0)
    {
        printf("%d\n", sum);
    }
}
```

Contoh program dalam bahasa Pascal:

```
User crt;
Var
    sum, a, b: integer;
Begin
    a:=15;
    b:=20;
    Sum:=a+b;
    Writeln(sum);
End;
```

Contoh bahasa Java:

```
class Matematika {
    private int a;
    private int b;

    public int getA() { return this.a; }
    public int getB() { return this.a; }
    public void setA(int nil) { this.a = nil;}
    public void setB(int nil) { this.b = nil;}

    public Matematika() {
        this.a = 0;
        this.b = 0;
    }

    public Matematika(int nil1, int nil2) {
        this.a = nil1;
        this.b = nil2;
    }

    public int tambahan() {
        return this.a + this.b;
    }
}
```

```
public int kurang(){
    return this.a - this.b;
}

public int kali(){
    return this.a * this.b;
}
public int modulus() {
    return this.a % this.b;
}
}

public class TestMat {
    public static void main(String[] args) {
        Matematika t = new Matematika();
        t.setA(10);
        t.setB(20);
        System.out.println("Tambah : " + t.tambahan());
        System.out.println("Modulus : " + t.modulus());
    }
}
```

4. **4GL/Fourth Generation Language:** terdiri dari berbagai pendekatan dalam pemrograman, namun memiliki karakteristik yang sama yaitu perintah-perintahnya mirip dengan kata-kata dalam bahasa Inggris, dan pemrograman dengan visualisasi yang lebih menekankan cara "drag and drop" daripada menuliskan perintah.

Bahasa pemrograman dalam generasi ini adalah:

- **bahasa-bahasa database:** SQL (Structured Query Language) untuk menampilkan data-data dalam susunan tertentu yang memberikan informasi. Misalnya dari kumpulan data mahasiswa dan kumpulan data matakuliah dan kumpulan data dosen dihasilkan KRS (Kartu Rencana Studi) untuk tiap mahasiswa.
- **bahasa-bahasa dengan pendekatan visual atau form-based language:** membuat program dengan mengisi membuat form dan mengisi form, menentukan menu dan komponen-komponen lain. Contoh adalah Visual Basic, Visual Foxpro, Visual C++.
- **bahasa markup:** digunakan untuk mendeskripsikan dokumen dan isinya. "Ibu" dari bahasa markup adalah SGML (Standard Generalised Markup Language). Contoh bahasa yang diturunkan dari SGML adalah HTML (HyperText Markup Language) dan XML (eXtensible Markup Language). SGML disebut juga meta-language, karena dari SGML dibentuk bahasa lain yaitu HTML dan XML. Perintah dalam SGML biasanya disebut tag.

Contoh SQL:

```
SELECT *
INTO #temp_titles
FROM titles
WHERE NOT EXISTS
    (SELECT * FROM sales WHERE title_ID = titles.title_ID)
```



Dalam proses pembuatan software aplikasi, ada proses yang disebut DEBUGGING. Proses ini adalah proses menguji perintah-perintah yang dituliskan untuk mencari kesalahan, dan kemudian memperbaiki kesalahan tersebut.

### **Tipe Software**

Bahasa-bahasa pemrograman di atas digunakan untuk membangun software lain. Kalau begitu software untuk membangun software? Kenyataannya begitu. Agar tidak membingungkan, secara prakteknya software digolongkan pada beberapa tipe:

1. **software sistem/system software:** berfungsi untuk mengendalikan dan mendukung sistem komputer dan aktifitas pengolahan informasi. Software sistem memfasilitasi pemrograman, pengujian dan, pencarian-perbaikan kesalahan dari program komputer. Contoh: sistem operasi (Microsoft Window 95-98, Microsoft XP, Novell, Linux, Unix, Macintosh), driver (VGA driver, USB driver, Printer Driver).
2. **software pemrograman/ bahasa pemrograman:** menyediakan alat untuk membuat software aplikasi. Software pemrograman terdiri dari editor tempat mengetikkan perintah-perintah bahasa pemrograman, compiler, interpreter, debugger dan fasilitas lain yang berbeda dari satu software pemrograman dan yang lainnya.
3. **software aplikasi/application software:** software yang dibangun untuk menyelesaikan tugas khusus. Aplikasi biasanya digunakan untuk otomasi pada industri, untuk keperluan bisnis, untuk keperluan pendidikan, untuk keperluan kesehatan, untuk pengolahan data pada database dan untuk hiburan. Penggunaan software aplikasi terbesar adalah pada bidang bisnis. Hal lain yang diotomasi pada umumnya menggunakan software aplikasi.

Semua tipe software dibangun dari bahasa pemrograman untuk tujuan yang berbeda, namun demikian setiap software harus dijalankan pada komputer dan didukung oleh software sistem yaitu sistem operasi. Tanpa sistem operasi software lain tidak dapat menjalankan fungsinya karena software sistem menjembatani antara hardware dan software lainnya yang dijalankan pada komputer.

Komputer tidak dapat berfungsi jika tidak ada software yang memberinya instruksi untuk melakukan tugas tertentu. Karena itulah saat ini software komputer berkembang sangat cepat untuk berbagai keperluan manusia.

### **Operasi Software dalam Komputer**

Software dimasukkan ke dalam tempat penyimpanan yang disebut RAM. Kemudian software dijalankan oleh komputer dengan mengirimkan perintah-perintah dari software aplikasi ke hardware melalui software sistem yang menerjemahkan perintah-perintah tadi menjadi bahasa mesin yang dimengerti oleh hardware. Setiap perintah menyebabkan

komputer menjalankan perintah seperti memindahkan data, menghitung, menampilkan gambar dan lain-lain.

Perintah-perintah tadi mungkin saja dijalankan secara berurutan, atau bergantung kondisi nilai atau beberapa perintah dilakukan berulang-ulang. Dalam setiap bahasa pemrograman jenis-jenis perintah tersebut ada dengan berbagai macam istilah namun serupa fungsinya.

### **Pembangunan Software**

Software berbeda dari produk lain yang kita kenal misalnya: Sepeda motor, baju, sepatu dan mobil. Software nampak mata hasil produknya, tetapi tidak dapat dipegang. Pembuatannya sesuai kebutuhan dan software tidak dapat kadaluarsa, tapi mungkin saja menjadi usang karena kondisi lingkungan. Ketika ada kerusakan, dan ada ketidak sesuaian software dapat diubah sehingga memenuhi kondisi yang ada.

Software dibangun berdasarkan permintaan atau kebutuhan penggunanya. Ini sangat jelas pada software aplikasi. Misalnya software yang digunakan untuk registrasi matakuliah di universitas, software untuk pencarian informasi buku di perpustakaan dan software untuk mencari informasi di Internet.

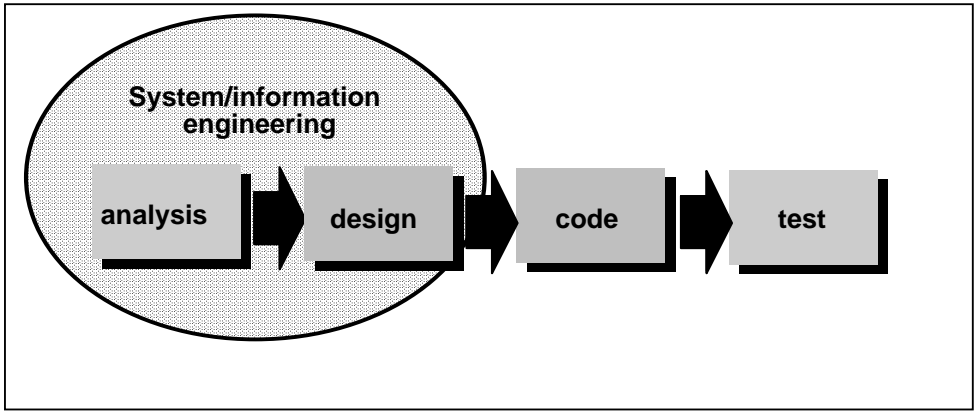
Pembangunan software, sering disebut REKAYASA PERANGKAT LUNAK atau Software Engineering. Dalam pembangunannya, ada beberapa tahapan yang pasti dilakukan oleh tim pembangun software:

1. tahap menentukan kebutuhan
2. tahap merancang
3. tahap menerapkan rancangan atau menuliskan perintah-perintah sesuai dengan rancangan
4. tahap penerapan atau penggunaan
5. tahap menguji hasil penulisan
6. tahap pemeliharaan ketika ada kesalahan, atau ketika ada penambahan kebutuhan

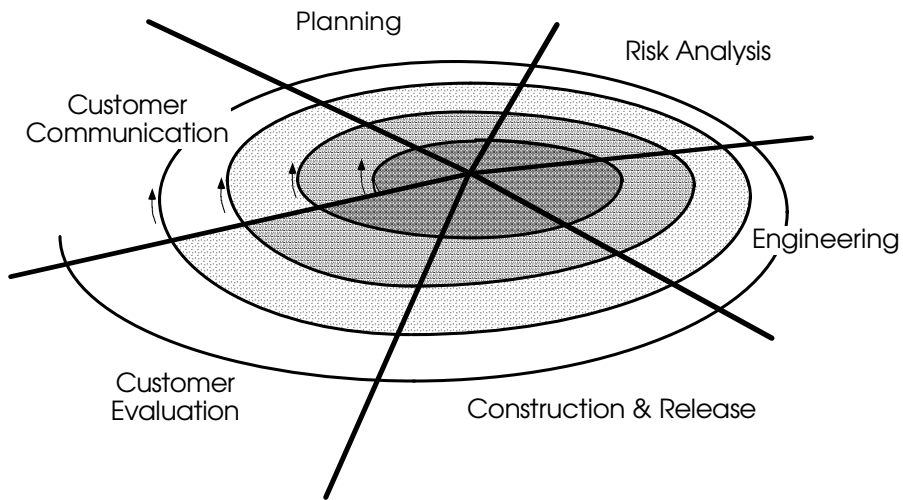
software yang berukuran besar: memenuhi banyak kebutuhan, dan baris perintahnya sangat banyak (ratusan ribu atau bahkan jutaan) tidak dikerjakan seorang diri, melainkan dikerjakan di dalam tim yang dikelola secara baik.

Dalam pembangunan perangkat lunak, ada beberapa model pembangunan yang dirancang untuk membantu pembangunan perangkat lunak. Beberapa model yang dikenal adalah: Waterfall model, Spiral model, dan Increment model. Berikut ini beberapa bagan yang menunjukkan tahapan-tahapan pembangunan perangkat lunak berdasarkan contoh-contoh model tersebut.

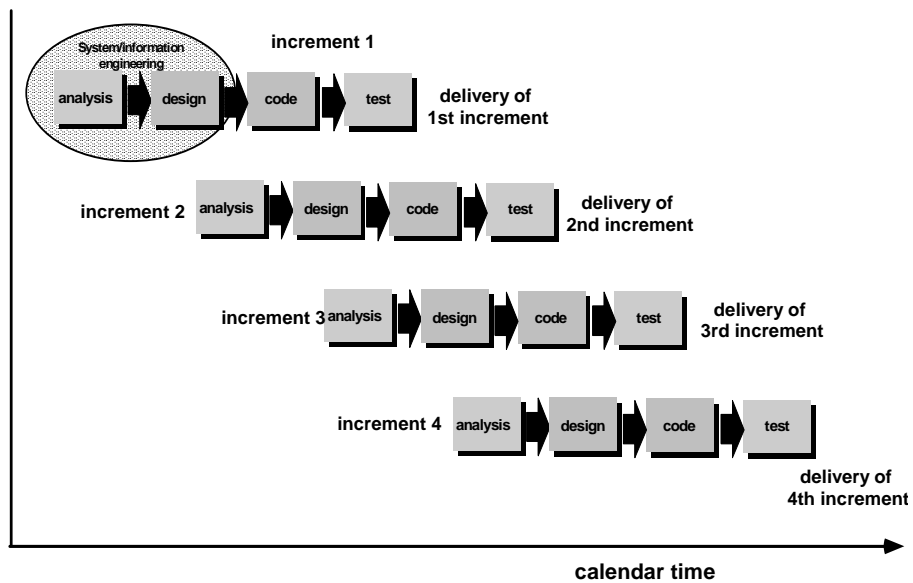
### Waterfall Model



### Spiral Model



### Increment Model



## Jenis Software Menurut Lisensi

Dengan cara pembangunan yang rumit, tentu saja software berharga mahal. Ada beberapa macam lisensi penggunaan software. Berdasarkan lisensi ini, software dapat dikelompokkan menjadi:

1. software komersial (off-the-shelf, proprietary software)/Non-free software  
software dalam jenis ini dijual bebas di pasaran. Pengguna software ini harus membeli ijin untuk menggunakan software dan tidak diijinkan untuk mengubah, menggandakan dan mendistribusikan.

Selain itu ada software yang termasuk software komersial tetapi pengguna dapat menggunakan tanpa biaya yaitu **freeware**. Freeware bukan free software. Freeware lebih mengacu pada bebas biaya, tetap free software berarti bebas untuk digunakan diubah dipelajari dan didistribusikan. Freeware tidak dapat diubah-ubah, dan hanya pembuatnya yang berhak untuk mengubah dan mengembangkannya.

Berbeda dari freeware, **shareware** adalah jenis software yang berlisensi seperti software komersial, tetapi mengizinkan pengguna menggunakan software tersebut dalam jangka waktu tertentu untuk mencoba, dan setelah waktu yang ditetapkan habis, pengguna harus membeli untuk menggunakannya lagi.

2. free software  
software dalam jenis ini dikenal sebagai open source software atau free software karena penggunaannya berdasarkan lisensi (ijin pemakaian) open source license atau free software license. Lisensi jenis ini dikeluarkan pertama kali oleh Free Software Foundation (FSF). Nama lisensi yang dikeluarkan adalah GNU GENERAL PUBLIC LICENSE atau sering disingkat GNU GPL.  
Pengguna software yang berlisensi GNU GPL mempunyai hak untuk menggunakan, mengubah dan mendistribusikan secara bebas. Bebas bukan selalu berarti tanpa uang. Jikalau membayar, biasanya membayar untuk jasa atau pemaketan. Mereka yang mendapatkan secara bebas, harus juga memberikan secara bebas kepada orang lain. Beberapa software berlisensi GNU GPL selain merupakan free software juga freeware.

Untuk lebih jelas tentang macam-macam lisensi, informasi yang lebih lengkap dan mudah untuk dipahami secara bertahap dapat diakses pada free encyclopedia wikipedia di URL: <http://en.wikipedia.com/>.

Selain itu untuk menambah pengetahuan tentang sejarah perkembangan software, informasi pada History of Software yang dapat diakses di <http://www.softwarehistory.org/index.html>.

Sumber:

Pressman, Roger.S. "Software Engineering : A Practioner's Approach." 4th . McGrawHill. 1997

Sommerville, Ian. "Software Engineering". 6th. Addison Wesley. 2001.

Turban, Efraim. Rainer, R. Kelly. Richard E.Potter. *Introduction to Information Technology*. Edisi ke 3. John Wiley and Sons. 2005.

Walters, E. Garrison. *The Essential Guide to Computing*. Prentice Hall PTR. NJ, 2001.

*Wikipedia: The Free Encyclopedia*. <http://en.wikipedia.com/>.