



## MODUL PERKULIAHAN

# Pengembangan Sistem Pendukung Keputusan

SDLC, Prototyping, Manajemen Perubahan, Perangkat dan tingkat teknologi, Platform pengembangan SPK

Fakultas

Ilmu Komputer

Program Studi

Sistem Informasi

Tatap Muka

**05**

Kode MK

18032

Disusun Oleh

Indrajani, S.Kom., MM.

### Abstract

Membahas SDLC, prototyping, manajemen perubahan, platform spk

### Kompetensi

Mahasiswa mampu memahami tahapan SDLC, prototyping  
Mahasiswa mampu menjelaskan manajemen perubahan  
Mahasiswa mampu memahami platform SPK  
Mahasiswa mampu menjawab dan menyelesaikan tugas dengan tepat.

## 1. Siklus hidup SDLC traditional

SDLC (Systems Development Life Cycle, Siklus Hidup Pengembangan Sistem) atau Systems Life Cycle (Siklus Hidup Sistem), dalam rekayasa sistem dan rekayasa perangkat lunak, adalah proses pembuatan dan perubahan sistem serta model dan metodologi yang digunakan untuk mengembangkan sistem-sistem tersebut.

Konsep ini umumnya merujuk pada sistem komputer atau informasi. SDLC juga merupakan pola yang diambil untuk mengembangkan sistem perangkat lunak, yang terdiri dari tahap-tahap:

### a) rencana(planning)

Perencanaan pengembangan sistem informasi bertujuan untuk mengidentifikasi dan memprioritaskan sistem informasi apa yang akan

dikembangkan, sasaran-sasaran yang ingin dicapai, jangka waktu pelaksanaan serta mempertimbangkan dana yang tersedia dan siapa yang akan melaksanakan.

Perencanaan sistem dapat mencakup keseluruhan unit business maupun secara departemen dengan memperhatikan misi dari usaha bisnis tersebut.

### b) analisis (analysis)

### c) desain (design)

### d) implementasi (implementation)

### e) uji coba (testing)

### f) pengelolaan (maintenance)

Dalam rekayasa perangkat lunak, konsep SDLC mendasari berbagai jenis metodologi pengembangan perangkat lunak. Metodologi-metodologi ini membentuk suatu kerangka kerja untuk perencanaan dan pengendalian pembuatan sistem informasi, yaitu proses pengembangan perangkat lunak.

Terdapat 3 jenis metode siklus hidup sistem yang paling banyak digunakan, yakni:

### a) siklus hidup sistem tradisional (traditional system life cycle)

### b) siklus hidup menggunakan prototyping (life cycle using prototyping)

### c) siklus hidup sistem orientasi objek (object-oriented system life cycle)

Langkah-langkah SDLC adalah sebagai berikut:

### a) Melakukan survei dan menilai kelayakan proyek pengembangan sistem informasi

### b) Mempelajari dan menganalisis sistem informasi yang sedang berjalan

### c) Menentukan permintaan pemakai sistem informasi

### d) Memilih solusi atau pemecahan masalah yang paling baik

### e) Menentukan perangkat keras (hardware) dan perangkat lunak (software)

### f) Merancang sistem informasi baru

- g) Membangun sistem informasi baru
- h) Mengkomunikasikan dan mengimplementasikan sistem informasi baru
- i) Memelihara dan melakukan perbaikan/peningkatan sistem informasi baru bila diperlukan

Beberapa model SDLC adalah:

a) Waterfall

Metode *Waterfall* adalah metode yang menyarankan sebuah pendekatan yang sistematis dan sekuensial melalui tahapan-tahapan yang ada pada SDLC untuk membangun sebuah perangkat lunak. Metode *Waterfall* menekankan pada sebuah keterurutan dalam proses pengembangan perangkat lunak. Metode ini adalah sebuah metode yang tepat untuk membangun sebuah perangkat lunak yang tidak terlalu besar dan sumber daya manusia yang terlibat dalam jumlah yang terbatas.

Keuntungan Metode Waterfall adalah:

- ✓ Kualitas dari sistem yang dihasilkan akan baik. Ini dikarenakan oleh pelaksanaannya secara bertahap. Sehingga tidak terfokus pada tahapan tertentu.
- ✓ Document pengembangan sistem sangat terorganisir, karena setiap fase harus terselesaikan dengan lengkap sebelum melangkah ke fase berikutnya. Jadi setiap fase atau tahapan akan mempunyai dokumen tertentu.

Kelemahan metode waterfall adalah:

- ✓ Diperlukan majemen yang baik, karena proses pengembangan tidak dapat dilakukan secara berulang sebelum terjadinya suatu produk.
- ✓ Kesalahan kecil akan menjadi masalah besar jika tidak diketahui sejak awal pengembangan.
- ✓ Pelanggan sulit menyatakan kebutuhan secara eksplisit sehingga tidak dapat mengakomodasi ketidakpastian pada saat awal pengembangan.

b) Fountain

Model Fountain adalah peningkatan logis untuk model waterfall. Langkah-langkahnya sama, dalam urutan yang sama, tetapi pada setiap langkah bisa ada

fallback ke langkah sebelumnya. Bergerak melalui sejumlah langkah dan jatuh kembali satu atau lebih langkah, dilakukan berulang kali, jauh lebih fleksibel dibandingkan dengan model Waterfall.

Langkah-langkah dasar meliputi:

- Analisis persyaratan pengguna
- Persyaratan Pengguna Spesifikasi
- spesifikasi persyaratan software
- Sistem / desain, desain logis yang luas
- Program / desain, detail desain fisik
- Implementasi / coding
- pengujian program: unit
- pengujian program: Sistem
- penggunaan program
- Software maintenance

Keuntungan dari metode fountain adalah:

- ✓ Tidak harus membekukan persyaratan terlalu cepat
- ✓ Lebih banyak interaksi antara desain dan persyaratan
- ✓ Mampu untuk memulai coding sebelumnya
- ✓ Mengurangi trauma karena perubahan system
- ✓ Memaksimalkan pengembalian modal investasi konsumen

#### c) Spiral

Model spiral (spiral model) adalah model proses software yang evolusioner yang merangkai sifat iteratif dari prototipe dengan cara kontrol dan aspek sistematis dari model sekuensial linier.

Kelebihan Model Spiral adalah:

- Lebih ditekankan pada pencairan alternatif, dan pemaksaan penggunaan kembali Software yang telah ada
- Analisa resiko
- Adanya prototype memudahkan komunikasi dengan konsumen

Kekurangan Spiral Model adalah:

- Biasanya pihak pengembang dan perusahaan berada pada satu pihak yang sama

- Tahapan analisa resiko sewaktu-waktu dapat membatalkan proses rekayasa, jika pihak pengembang adalah pihak di luar perusahaan, maka timbulah masalah hukum

#### d) Rapid Application Development

Rapid application development (RAD) atau rapid prototyping adalah model proses pembangunan perangkat lunak yang tergolong dalam teknik incremental (bertingkat). RAD menekankan pada siklus pembangunan pendek, singkat, dan cepat. Waktu yang singkat adalah batasan yang penting untuk model ini.

Keuntungan metode RAD adalah:

- ✓ Membeli sistem yang baru memungkinkan untuk lebih menghemat biaya  
Proses pengiriman menjadi lebih mudah
- ✓ Mudah untuk diamati karena menggunakan model prototype
- ✓ Lebih fleksibel
- ✓ Bisa mengurangi penulisan kode yang kompleks
- ✓ Keterlibatan user semakin meningkat
- ✓ Mampu meminimalkan kesalahan-kesalahan dengan menggunakan CASE tools  
Mempercepat waktu pengembangan sistem secara keseluruhan
- ✓ Tampilan yang lebih standar dan nyaman dengan bantuan software-software pendukung.

Kelemahan metode RAD adalah:

- ✓ Sistem yang tidak bisa dimodularisasi tidak cocok untuk model ini.
- ✓ Penghalusan dan penggabungan dari beberapa tim di akhir proses sangat diperlukan dan ini memerlukan kerja keras.
- ✓ Proyek bisa gagal karena waktu yang disepakati tidak dipenuhi
- ✓ Risiko teknis yang tinggi juga kurang cocok untuk model ini.

#### e) Prototyping

Metode ini secara keseluruhan akan mengacu kepada kepuasan user. Bisa dikatakan bahwa metode ini merupakan metode waterfall yang dilakukan secara berulang-ulang.

Keunggulan metode Prototyping adalah:

- ✓ Adanya komunikasi baik antara pengembang dengan pelanggan.

- ✓ Pengembang dapat bekerja lebih baik untuk memenuhi kebutuhan pelanggan.
- ✓ Pelanggan berperan aktif dalam pengembangan sistem.
- ✓ Menghemat waktu dalam pengembangannya.
- ✓ Penerapan lebih mudah karena pemakai akan mengetahui apa yang diharapkan.

Kelemahan metode Prototyping adalah:

- ✓ Kualitas sistem kurang baik karena hanya mengedepankan aspek kenyamanan *user*.
- ✓ Pengembang kadang-kadang menggunakan implementasi yang sembarangan.
- ✓ Tidak mencerminkan proses perancangan yang baik.

f) Incremental

Model pengembangan sistem pada software engineering berdasarkan requirement software yang dipecah menjadi beberapa fungsi atau bagian sehingga model pengembangannya secara bertahap.

g) Build and fix

Model Build dan Fix yang juga diketahui sebagai pembangunan Ad-Hoc adalah struktur yang paling lemah dari metodologi putaran hidup pembangunan sistem ( System Development Life Cycle ). Model ini sangat mengandalkan keahlian individu dari anggota tim, kebanyakan tugas-tugasnya dikerjakan oleh satu orang. Dengan metode Build dan Fix, pembangun /developers menulis beberapa kode, kemudian berlanjut untuk memodifikasinya sampai kode tersebut dapat bekerja dan konsumen puas. Dengan perencanaan yang kurang, strategi ini dapat beresiko, dan jika dijalankan seenaknya dapat menyusahkan pembangunan dalam melakukan maintenance / pemeliharaan terhadap software tersebut.

h) Synchronize and stabilize

Memungkinkan tim untuk bekerja efisien secara paralel pada berbagai modul aplikasi individu. Hal ini dilakukan sambil sering sinkronisasi pekerjaan sebagai individu dan itu sebagai anggota tim paralel dan berkala menstabilkan dan / atau debugging kode seluruh proses pembangunan.

## 2. Prototyping metodologi pengembangan

### Tahapan Metode Prototyping

1. **Pemilihan Fungsi.** Mengacu pada pemilihan fungsi yang harus ditampilkan oleh prototyping. Pemilihan harus selalu dilakukan berdasarkan pada tugas-tugas yang relevan yang sesuai dengan contoh
2. **Evaluasi.**

### 3. Penggunaan selanjutnya.

Jenis Jenis Prototyping terdiri dari:

1. **Feasibility prototyping.** Digunakan untuk menguji kelayakan dari teknologi yang akan digunakan untuk system informasi yang akan disusun.
2. **Requirement prototyping.** Digunakan untuk mengetahui kebutuhan aktivitas bisnis user. Misalnya dalam sebuah perusahaan terdapat *user* direktur, manajer, dan karyawan. Maka penggunaan sistem dapat dibedakan berdasarkan *user* tersebut sesuai dengan kebutuhannya.
3. **Desain Prototyping.** Digunakan untuk mendorong perancangan system informasi yang akan digunakan.
4. **Implementation prototyping.** Merupakan lanjutan dari rancangan protipe, prototype ini langsung disusun sebagai suatu system informasi yang akan digunakan.

### 3. Manajemen perubahan

Manajemen Perubahan adalah upaya yang dilakukan untuk mengelola akibat-akibat yang ditimbulkan karena terjadinya perubahan dalam organisasi. Perubahan dapat terjadi karena sebab-sebab yang berasal dari dalam maupun dari luar organisasi tersebut. Manajemen perubahan merupakan suatu hal yang penting dalam suatu organisasi. Dengan adanya manajemen perubahan, suatu organisasi dapat menjadi lebih dinamis dalam menghadapi perkembangan jaman dan kemajuan teknologi.

Perubahan terjadi melalui revolusi, reformasi, evolusi, dan inovasi. Setiap orang tentu berbeda-beda dalam menanggapi perubahan. Beberapa tanggapan tersebut antara lain menolak, cuek, belum siap, dan siap. Perbedaan tanggapan tersebut membuat masing-masing orang mendapatkan pilihan yang berbeda-beda dari perubahan. Pilihan tersebut adalah:

#### 1. Menjadi motor perubahan

Disini, kita memiliki posisi di garda terdepan terhadap proses perubahan yang terjadi. Kita dituntut untuk memiliki pengetahuan tentang konsep dan alasan perlunya sebuah perubahan harus dilakukan. Dengan demikian kita bisa mempengaruhi serta meyakinkan pihak lain bahwa kondisi yang ada pada saat ini perlu diubah,

#### 2. Mendiamkan perubahan

Posisi ini merupakan posisi yang paling banyak dipilih oleh mereka yang ingin mengambil posisi aman terhadap kondisi yang ada. Mereka tidak berada di posisi sebagai pelopor perubahan, namun juga tidak menolak atas perubahan yang terjadi. Biasanya, orang-orang seperti ini tergolong sebagai kaum oportunis. Dimana ketika perubahan itu akan membawa keuntungan bagi mereka, maka perubahan itu akan mereka dukung. Sebaliknya jika mereka

melihat perbaan itu tidak membawa keuntungan serta proses perubahan tersebut cenderung gagal, mereka memilih posisi aman dengan diam pada posisi yang ada pada saat ini.

3. Melawan perubahan

Posisi ini biasanya dilakukan oleh pihak yang sudah merasa nyaman dan memiliki keuntungan atas sebuah kondisi yang ada. Sehingga, mereka akan berusaha menolak semua usaha yang bertujuan untuk menggantikan posisi yang sudah ada sebelumnya. Biasanya, penolakan ini dilakukan karena pertimbangan materi dan kedudukan.

4. Dirubah oleh arus perubahan

Posisi diambil oleh mereka yang melihat bahwa perubahan yang terjadi membawa sebuah perbaikan. Sehingga mereka merasa perlu untuk mengikuti perubahan yang terjadi tersebut secara rasional, dan bukan atas dasar keinginan untuk mendapatkan keuntungan. Namun lebih pada kesadaran bahwa perubahan tersebut memang perlu dilakukan serta membawa ke arah kebaikan.

Teori perubahan Lewin-Schein merupakan panduan yang sangat bagus dan praktis untuk mempertimbangkan isu-isu yang kompleks. Tiga langkah dasar adalah:

1. *Unfreezing* : menciptakan kesadaran akan kebutuhan dan perubahan dan sebuah iklim yang menerima perubahan
2. *Moving*: mengubah tekanan dan atau arah kekuatan yang menentukan kebutuhan atau perubahan awal dengan mengembangkan metode-metode baru dan mempelajari sikap dan perilaku baru
3. *Refreezing*: memperkuat perubahan yang diinginkan yang telah terjadi dan membangun sebuah ekuilibrium yang dapat dipelihara dan stabil.

Perubahan dalam organisasi terdiri dari:

1. Perubahan Operasional: Perubahan-perubahan kecil yang bersifat parsial dan umumnya tidak menimbulkan dampak yang luar biasa bagi yang lainnya.
2. Perubahan Strategis : Perubahan yang berdampak luas dan memerlukan koordinasi dan dukungan dari yang lainnya. Jenis perubahan ini adalah : perubahan budaya, perubahan fokus, dan perubahan cara kerja.

Hambatan dan Tantangan dalam Perubahan

Tantangan yang dihadapi dalam perubahan, muncul ketika individu maupun organisasi memiliki:



- a) Rasa takut terhadap perubahan
- b) Resiko terhadap penolakan, kegagalan, dan kerugian
- c) Kesulitan mendapatkan apa yang diperlukan untuk memutuskan dan mencoba perubahan

#### 4. Perangkat dan tingkat teknologi

Pengembangan SPK berdasarkan perangkatnya terdapat 3 tingkatan perangkat keras / perangkat lunak, ketiga tingkatan tersebut adalah:

##### 1. **SPK spesifik (specific DSS)**

SPK spesifik merupakan spk siap digunakan secara langsung untuk menyelesaikan pekerjaan sistem, ini meliputi sistem informasi terapan. Tetapi dengan karakteristik yang sangat berbeda (karakteristik tertentu) dengan pemrosesan data biasanya. **SPK Spesifik adalah:** perangkat keras/ perangkat lunak yang memungkinkan pembuat keputusan menyelesaikan kelompok masalah yang saling berhubungan.

##### 2. **Pembangkit SPK (DSS Generator)**

Pembangkit SPK adalah: suatu paket perangkat keras / lunak yang mempunyai kemampuan untuk mengembangkan SPK khusus secara cepat dan mudah.

Contoh pada pembangkit SPK yaitu:

- Geodata Analisis and Display Sistem (GADS) dari IBM dan interactive
- Financial Planning system (FPS) dari executive system.

Pembangkit SPK di antaranya: Fasilitas penyiapan laporan, bahasa simulasi, tampilan grafik

##### 3. **Peralatan SPK (DSS tools).**

Peralatan spk adalah: tingkatan paling dasar dalam tingkatan spk. Peralatan spk adalah: elemen2 perangkat keras / perangkat lunak yang digunakan untuk mengembangkan spk spesifik maupun pembangkit spk, yang termasuk kategori2 perangkat / teknologi, ini adalah bahasa pemrograman (c, pascal, dbase) dan sebagainya.

#### 5. Platform pengembangan SPK

Membangun sebuah SPK, apalagi yang besar, merupakan proses yang rumit.

- Melibatkan hal-hal: teknis (hardware, jaringan) dan perilaku (interaksi manusia-mesin, dampak SPK pada individu).
- Agar lebih mudah membangun SPK bisa digunakan bahasa khusus (misal CASE TOOLS).

#### **Strategi Pengembangan**

- a. Tulis SPK dengan bahasa pemrograman umum: Pascal, Delphi, C, C++, C#, Java, dan lainnya.
- b. Menggunakan 4GL: data-oriented language, spreadsheets, dan financial-oriented language.
- c. Menggunakan Generator SPK: Excel, Lotus 1-2-3, Quattro Pro, Express. Generator lebih efisien dari 4GL tapi ini tergantung juga pada batasannya.
- d. Menggunakan Generator SPK khusus (domain specific): Commander FDC untuk budgeting & financial analysis, EFPM untuk kalangan perguruan tinggi.
- e. Mengembangkan SPK dengan metodologi CASE. Memiliki jaminan kualitas yang memadai.
- f. Untuk SPK yang kompleks, bisa mengintegrasikan pendekatan-pendekatan di atas.

#### 6. Memilih perangkat pengembangan SPK

Berbagai pertanyaan yang harus dijawab oleh suatu organisasi yang akan menggunakan Generator SPK:

- (1) generator seperti apa yang akan digunakan
- (2) hardware seperti apakah yang dipakai untuk menjalankannya
- (3) sistem operasi seperti apa yang akan digunakan
- (4) jaringan seperti apakah yang akan dipakai untuk menjalankannya

Dengan kemampuan PC yang luar biasa sekarang ini, software DSS lebih banyak ditemui pada jenis komputer mikro. Kemudian dengan adanya program-program berbasis Windows, membuat SPK menjadi lebih disukai karena kemudahan penggunaannya.

#### **Pemilihan Software.**

Tool software dasar yang patut dipertimbangkan adalah:

- ✓ Fasilitas database relasional dengan fasilitas pembuatan laporan yang baik dan fasilitas pemilihan data setiap saat.
- ✓ Bahasa penghasil grafis.
- ✓ Bahasa pemodelan.
- ✓ Bahasa analisis data statistik umum.
- ✓ Bahasa khusus yang lain (misal: untuk membangun simulasi).
- ✓ Bahasa pemrograman (generasi ketiga).
- ✓ Tool pemrograman berorientasi objek.
- ✓ Tool pembangun ES.
- ✓ Jaringan.
- ✓ CASE tools.

Pemilihan tool dan/atau generator adalah merupakan proses yang rumit dengan alasan:

- a. Pada saat pemilihan, informasi SPK yang dibutuhkan dan outputnya tidak diketahui secara lengkap.
- b. Terdapat ratusan paket software di pasaran.
- c. Paket software selalu berganti dengan cepat.
- d. Harga seringkali berubah.
- e. Beberapa orang terlibat dalam tim evaluasi.
- f. Suatu bahasa digunakan dalam membangun berbagai SPK. Dari sini kemampuan yang dibutuhkan dari tool berubah dari satu aplikasi ke aplikasi lain.
- g. Pemilihan keputusan melibatkan banyak kriteria yang diterapkan pada bermacam-macam paket yang dibandingkan. Beberapa kriteria tak bisa diukur, yang lain memiliki konflik langsung dengan yang lain lagi.
- h. Semua masalah teknis, fungsional, end-user, dan manajerial harus dipertimbangkan masak-masak.
- i. Evaluasi komersial yang tersedia, misalnya oleh Data Decisions, Data Pro, dan Software Digest, Inc.; dan jurnal panduan user, misalnya PC Week dan Infosystems, semuanya bersifat subjektif dan seringkali dangkal ulasannya. Mereka hanyalah merupakan salah satu sumber informasi saja, yang setara dengan sumber-sumber informasi lain.

#### 7. SPK yang dikembangkan oleh tim atau individu

Pengembangan DSS pada tahun 1970 dan 1980-an melibatkan skala yang besar, sistemnya kompleks, dan didesain utama untuk mendukung organisasi. Sistem ini didesain oleh tim yang terdiri dari user, penghubung (intermediaries), DSS builder, tenaga ahli, dan berbagai tool. Berbagai individu dalam setiap kategori tadi, sehingga ukuran tim menjadi besar dan komposisinya sering berubah seiring berjalannya waktu. Intinya dengan berbasis tim, maka pembangunan DSS menjadi kompleks, lama, dan prosesnya memakan biaya.

- ✓ Pendekatan lainnya adalah membangun DSS berbasiskan user. Dimulai mulai tahun 1980-an, seiring pesatnya perkembangan di bidang PC (Personal Computer), jaringan komunikasi komputer, berkurangnya biaya hardware maupun software.
- ✓ Enterprise-wide computing serta kemudahan akses data dan pemodelan berarsitektur client/server juga mendukung pengembangan DSS berbasis user.
- ✓ Tentu saja kedua pendekatan ini bisa dikombinasikan, untuk mendapat kinerja yang diinginkan.

### **Pengembangan SPK berbasis tim**

- Sistem ini didesain oleh tim yang terdiri dari user, penghubung, SPK builder, tenaga ahli, dan berbagai tools
- Secara organisasi, penempatan group SPK bisa dimana-mana, umumnya pada lokasi:
  - a) Dalam departemen IS (Information Services)
  - b) Executive Staff Group
  - c) Dalam Wilayah keuangan ataupun fungsi lainnya
  - d) Dalam departemen rekayasa industri
  - e) Dalam kelompok manajemen pengetahuan (Managemen Science Group)
  - f) Dalam kelompok pusat informasi (Information Center Group)
- Dengan berbasis Tim, maka pembangunan SPK menjadi kompleks, lama dan prosesnya memakan biaya.

### **Pengembangan SPK berbasis Pengguna**

- PC telah tersebar di seluruh organisasi, komunikasi dengan server data telah meningkat, dan piranti perangkat lunak telah meningkat (kemampuan, user friendly, harga dan kualitas). Sehingga para pengguna memerlukan piranti untuk mengembangkan SPK/BI mereka sendiri, bahkan sistem berbasis web.
- Keuntungan penting dari SPK berbasis Pengguna:
  - a) Waktu penyerahan pendek. Tidak perlu menunggu tim pengembangan SI untuk menjadwalkan dan menyelesaikan penyelesaian
  - b) Prasyarat dari spesifikasi persyaratan pengguna ekstensif dan formal dapat disesuaikan.
  - c) Berkurangnya beberapa masalah implementasi
  - d) Biaya pada umumnya rendah

### **Risiko pengembangan SPK berbasis pengguna:**

- a. Kualitas buruk, karena kurangnya pengalaman design SPK formal dan kecenderungan pengguna akhir untuk mengabaikan kendali konvensional, prosedural pengujian, dan standar dokumentasi dapat mendorong kepada sistem berkualitas rendah.
- b. Risiko keamanan meningkat karena pengguna kurang mengenali ukuran keamanan

- c. Tidak adanya prosedur dokumentasi dan pemeliharaan mungkin menyebabkan permasalahan, terutama jika pengembang meninggalkan organisasi

## 8. Mengembangkan SPK bersama

Tool-tool pengembangan meningkatkan produktivitas builder dan membantu mereka menghasilkan SPK yang sesuai dengan kebutuhan user pada biaya yang moderat.

Hal mendasar pada pengembangan tool dan generator ini adalah konsep:

- 1) penggunaan tool otomatis skala tinggi di keseluruhan proses pengembangan, dan
- 2) penggunaan bagian-bagian pra-fabrikasi dalam proses manufaktur keseluruhan sistem. Sistem pengembangan SPK bisa dibayangkan seperti bengkel dengan berbagai tool dan komponen. Sistem ini melibatkan komponen-komponen utama seperti:

- Penanganan permintaan/query (mendapatkan informasi dari database).
- Fasilitas analisis dan desain sistem (pengeditan, penginterpretasian, dll.).
- Sistem manajemen dialog (antarmuka user).
- Generator laporan (memformat laporan output).
- Generator grafis.
- Manajer kode sumber (menyimpan dan mengakses model built-in dan model yang dikembangkan user/user developed).
- Sistem manajemen berbasis model.
- Sistem manajemen knowledge.

# Daftar Pustaka

1. Turban, Efraim & Jay E.Aronson, "Decision Support Systems and Intelligent Systems 7th edition", Prentice Hall, 2005.
2. Marakas, George M., "Decision Support Systems in The 21st Century", Prentice Hall, 2003.
3. Mallach, Efraim G., "Decision Support and Data Warehouse Systems", McGraw-Hill International Editions, 2000.
4. Taylor, Bernard W, "Introduction to Management Science 7th edition", Prentice Hall, 2001.