**[METODE-METODE PENGEMBANGAN SISTEM INFORMASI](http://uchiuwik.blogspot.co.id/2014/11/metode-metode-pengembangan-sistem.html)**

**METODE-METODE PENGEMBANGAN SISTEM INFORMASI**

**a.**     **Metode System Development Life Cycle (SLDC)**

Metode ini adalah metode pengembangan sistem informasi yang pertama kali digunakan makanya disebut dengan metode tradisional. Metode ini prototype Adalah tahap-tahapan pekerjaan yang dilakukan oleh analis sistem dan programer dalam membangun sistem informasi.

***Adapun tahap-tahap tersebut yaitu:***

**1.   Melakukan survey dan menilai kelayakan proyek pengembangan sistem informasi.**

Tahap ini akan ditentukan ruang lingkup proyek bagi semua pemakai sistem informasi dan berbagai tingkat pertanggungjawaban, meneliti masalah dan berbagai kemungkinan adanya kendala, menentukan sasaran proyek dan menentukan solusinya.

Hasil dari survey adalah laporan kelayakan studi berisi temuan-temuan, rekomendasi, pertimbangan biaya dan manfaat. Temuan ini harus diketahui oleh komite pengawas

**2.   Mempelajari dan menganalisis sistem informasi yang sedang berjalan**

Mempelajari sistem yang sedang berjalan, mengetahui sebab dan kendala yang dihadapi. Hasil dari tahap ini adalah laporan yang mengungkapkan adanya berbagai permasalahan (problem statement)

**3.   Menentukan permintaan pemakai sistem informasi**

Hal terpenting dari sistem informasi adalah terlebih dahulu harus mendapat persetujuan dari para pemakai sistem (pemakai sistem dilibatkan). Hasil dari tahap ini adalah laporan permintaan dari pemakai sistem informasi yang akan dijadikan dasar untuk pembuatan keputusan.

**4.   Memilih solusi atau pemecahan masalah yang paling baik**

Dari berbagai solusi maka solusi dan pemecahan masalah terbaiklah yang akan dipilih yaitu berdasarkan hasil analisis permintaan pemakaian.

**5.   Menentukan perangkat keras dan perangkat lunak komputer**

Setelah proposal pengembangan sistem informasi disetujui maka ditentukan hardware dan software yang akan digunakan dan bagaimana cara mendapatkannya

**6.   Merancang sistem informasi baru**

Kegiatan perancangan sistem informasi baru umumnya meliputi: input, proses, output, bahan yang digunakan, metode dan prosedur serta pengendalian intern

**7.   Mengkomunikasikan dan mengimplementasikan sistem informasi baru**

Hasil penyusunan sistem informasi adalah sebuah software komputer yang siap pakai digunakan sesuai dengan kebutuhan users, selanjutnya analis harus memperkenalkan paket sistem informasi tersebut untuk dioperasikan (pelatihan users dll)

**8.   Memelihara dan melakukan perbaikan/peningkatan sistem informasi baru**

Pemeliharaan yang dilakukan adalah dengan melakukan perbaikan-perbaikan pada kesalahan atau kegagalan yang timbul dalam penggunaan sistem informasi.

**Kelebihan dan Kekurangan**

**1.**        **Kelebihan**

  Mudah diaplikasikan.

   Memberikan template tentang metode analisis, desain, pengkodean, pengujian, dan pemeliharaan.

**2.**        **Kekurangan**

  Jarang sekali proyek riil mengikuti aliran sekuensial yang dianjurkan model karena model ini bisa melakukan itersi tidak langsung.

  Pelanggan sulit untuk menyatakan kebutuhan secara eksplisit sehingga sulit untuk megakomodasi ketidakpastian pada saat awal proyek.

  Pelanggan harus bersikap sabar karena harus menunggu sampai akhir proyrk dilalui. Sebuah kesalahan jika tidak diketahui dari awal akan menjadi masalah besar karena harus mengulang dari awal.

  Pengembang sering malakukan penundaan yang tidak perlu karena anggota tim proyek harus menunggu tim lain untuk melengkapi tugas karena memiliki ketergantungan hal ini menyebabkan penggunaan waktu tidak efesien.

**b.**    **Model ‘Air** **Terjun’ (Waterfall)**

o  Sering juga disebut model Sequential Linier.

o  Metode pengembangan sistem yang paling tua dan

o  paling sederhana.

o  Cocok untuk pengembangan perangkat lunak dengan

o  spesifikasi yang tidak berubah-ubah.

 Model ini menyediakan pendekatan alur hidup perangkat

 lunak secara sequential atau terurut dimulai dari analisa,

 desain, pengkodean, pengujian dan tahap pendukung

***Tahap-tahap metode WATERFALL***

**1. Analisis Kebutuhan Perangkat Lunak**

Pengumpulan kebutuhan untuk menspesifikasikankebutuhan perangkat lunak sehingga dapat dipahami kebutuhan dari user.

**2. Desain**

Desain pembuatan program perangkat lunak termasuk struktur data, arsitektur perangkat lunak, representasi antar muka dan prosedur pengkodean.

**3. Pembuatan Kode Program**

Hasil tahap ini adalahprogram komputer sesuari dengan desain yang telah dibuat pada tahap desain.

**4. Pengujian**

Pengujian fokus pada perangkat lunak dari segi logik dan fungsional serta memastikan bahwa semua bagian sudah diuji sehingga keluaran yang dihasilkan sesuai dengan yang diinginkan.

**5. Pendukung atau Pemeliharaan**

Dikarenakan adanya perubahan ketika sudah dikirimkan ke user. Perubahan dapat terjadi karena adanya kesalahan

yang muncul dan tidak terdeteksi saat pengujian.

***Keunggulan dan Kelemahan Metode Waterfall***

Metode pengembangan *waterfall* mempunyai keunggulan dalam membangun dan mengembangkan suatu sistem, antara lain:

1.       Kualitas dari sistem yang dihasilkan akan baik. Ini dikarenakan oleh pelaksanaannya secara bertahap. Sehingga tidak terfokus pada tahapan tertentu.

2.       Dokumen 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.

Dalam proses membangun dan mengembangkan suatu sistem, metode *waterfall* mempunyai beberapa kelemahan, antara lain:

1.       Diperlukan majemen yang baik, karena proses pengembangan tidak dapat dilakukan secara berulang sebelum terjadinya suatu produk.*.*

2.       Kesalahan kecil akan menjadi masalah besar jika tidak diketahui sejak awal pengembangan.

3.       Pelanggan sulit menyatakan kebutuhan secara eksplisit sehingga tidak dapat mengakomodasi ketidakpastian pada saat awal pengembangan.

**c.**    **Model Prototyping**

Prototyping adalah proses iterative dalam pengembangan sistem dimana requirement diubah ke dalam sistem yang bekerja (working system) yang secara terus menerus diperbaiki melalui kerjasama antara user dan analis. Prototype juga bisa dibangun melalui beberapa tool pengembangan untuk menyederhanakan proses.

***Tahapan-tahapan Model Prototyping***

**1.**       **Pengumpulan Kebutuhan**

Pelanggan dan pengembang bersama-sama mendefinisikan format seluruh perangkat lunak, mengidentifikasikan semua kebutuhan, dan garis besar sistem yang akan dibuat.

**2.**       **Membangun Prototyping**

Membangun prototyping dengan membuat perancangan sementara yang berfokus pada penyajian kepada pelanggan (misalnya dengan membuat input dan format output).

**3.**       **Menggunakan Sistem**

Evaluasi ini dilakukan oleh pelanggan apakah prototyping yang sudah dibangun sudah sesuai dengan keinginann pelanggan.

**4.**       **Mengkodekan Sistem**

Dalam tahap ini prototyping yang sudah di sepakati diterjemahkan ke dalam bahasa pemrograman yang sesuai.

**5.**       **Menguji Sistem**

Setelah sistem sudah menjadi suatu perangkat lunak yang siap pakai, harus dites dahulu sebelum digunakan. Pengujian ini dilakukan dengan White Box, Black Box, Basis Path, pengujian arsitektur dan lain-lain.

**6.**       **Evaluasi Sistem**

Pelanggan mengevaluasi apakah sistem yang sudah jadi sudah sesuai dengan yang diharapkan.

**7.**       **Evaluasi Protoptyping**

Perangkat lunak yang telah diuji dan diterima pelanggan siap untuk digunakan.

**Kelebihan dan Kekurangan**

1.       Kelebihan

   Prototype melibatkan user dalam analisa dan desain.

   Punya kemampuan menangkap requirement secara konkret.

   Digunakan untuk memperluas SDLC.

2.       Kekurangan

 Proses analisis dan perancangan terlalu singkat.

 Mengesampingkan alternatif pemecahan masalah.

 Bisanya kurang fleksible dalam mengahdapi perubahan.

 Protitype yang dihasilkan tidak selamanya mudah dirubah dan cepat selesai.

**d.**    **Model RAD (Rapid Application Development)**

RAD adalah penggabungan beberapa metode atau teknik terstruktur. RAD menggunakan metode prototyping dan teknik terstruktur lainnya untuk menentukan kebutuhan user dan perancangan sistem informasiselain itu RAD menekankan siklus perkembangan dalam waktu yang singkat (60 sampai 90 hari) dengan pendekatan konstruksi berbasis komponen.

***Tahapan-tahapan Model RAD***

1.       **Bussiness Modelling**

Fase ini untuk mencari aliran informasi  seperti: informasi mengendalikan proses bisnis, di mana informasi digunakan,  siapa yang memprosenya, dan informasi apa yang dimunculkan.

2.       **Testing and Turnover**

Karena menggunakan kembali komponen yang telah ada, maka akan mengurangi waktu pengujian. Tetapi komponen baru harus diuji dan semua interface harus dilatih secara penuh..

3.       **Aplication Generation**

Selain menggunakan bahasa pemrograman generasi ketiga, RAD juga memakai komponen program yang telah ada atau menciptakan komponen yang bisa dipakai lagi. Alat-alat baantu bisa dipakai untuk memfasilitasi konstruksi perangkat lunak.

4.       **Process Modelling**

Aliran informasi pada fase data modelling ditransformasikan untuk mendapatkan aliran informasi yang diperlukan pada implementasi fungsi bisnis. Pemrosesan diciptakan untuk menambah, memodifikasi, menghapus, atu mendapatkan kembali objek data tertentu

5.       **Data Modelling**

Fase ini menjelaskan objek data yang dibutuhkan dalam proyek. Karakteristik (atribut) masing-masing data diidentifikasikan dan hubungan antar objek didefinisikan.

***Kelebihan dan Kekurangan***

**1.**       **Kelebihan**

   RAD mengikuti tahapan pengembangan sistem sepeti umumnya, tetapi mempunyai kemampuan untuk menggunakan kembali komponen yang ada *(reusable object).*

  Setiap fungsi dapat dimodulkan dalam waktu tertentu dan dapat dibicarakan oleh tim RAD yang terpisah dan kemudian diintegrasikan sehingga waktunya lebih efesien.

**2.**       **Kekurangan**

   Tidak cocok untuk proyek skala besar

  Proyek bisa gagal karena waktu yang disepakati tidak dipenuhi.

  Sistem yang tidak bisa dimodularisasi tidak cocok untuk model ini.

  Resiko teknis yang tinggi juga kurang cocok untuk model ini

**e.**     **Model Spiral**

Model spiral pada awalnya diusulkan oleh Boehm, adalah model proses perangkat lunak evolusioner yang merangkai sifat iteratif dari prototype dengan cara kontrol dan aspek sistematis model sequensial linier. Model iteratif ditandai dengan tingkah laku yang memungkinkan pengembang mengembangkan versi perangkat lunak yang lebih lengkap secara bertahap.

***Tahapan-tahapan Model Spiral***

**1.**       **Komunikasi Pelanggan**

Yaitu tugas-tugas untuk membangun komunikasi antara pelanggan dan kebutuhan- kebutuhan yang diinginkan oleh pelanggan.

**2.**       **Perencanaan**

Yaitu tugas-tugas untuk mendefinisikan sumber daya, ketepatan waktu, dan proyek informasi lain yg berhubungan.

**3.**       **Analisis Resiko**

Yaitu tugas-tugas yang dibutuhkan untuk menaksir resikomanajemen dan teknis.

**4.**       **Perekayasaan**

Yaitu tugas yang dibutuhkan untuk membangun satu atau lebih representasi dari apikasi tersebut.

**5.**       **Konstruksi dan Peluncuran**

Yaitu tugas-tugas yang dibutuhkan untuk mengkonstruksi, menguji, memasang, dan memberi pelayanan kepada pemakai.

**6.**       **Evaluasi Pelanggan**

Yaitu tugas-tugas untuk mendapatkan umpan balik dari pelanggan.

***Kelebihan dan Kekurangan***

**1.**       **Kelebihan**

  Dapat disesuaikan agar perangkat lunak bisa dipakai selama hidup perangkat lunak komputer.

  Lebih cocok untuk pengembangan sistem dan perangkat lunak skala besar

  Pengembang dan pemakai dapat lebih mudah memahami dan bereaksi terhadap resiko setiap tingkat evolusi karena perangkat lunak terus bekerja selama proses

  Menggunakan prototipe sebagai mekanisme pengurangan resiko dan pada setiap keadaan di dalam evolusi produk.

  Tetap mengikuti langkah-langkah dalam siklus kehidupan klasik dan memasukkannya ke dalam kerangka kerja iteratif .

  Membutuhkan pertimbangan langsung terhadp resiko teknis sehingga mengurangi resiko sebelum menjadi permaslahan yang serius.

**2.**       **Kekurangan**

  Sulit untuk menyakinkan pelanggan bahwa pendekatan evolusioner ini bisa dikontrol.

  Memerlukan penaksiran resiko yang masuk akal dan akan menjadi masalah yang serius jika resiko mayor tidak ditemukan dan diatur.

  Butuh waktu lama untuk menerapkan paradigma ini menuju kepastian yang absolute

**f.**       **Object Oriented Technology**

            Object Oriented Technology merupakan cara pengembangan perangkat lunak berdasarkan abstraksi objek-objek yang ada di dunia nyata. Dasar pembuatan adalah Objek, yang merupakan kombinasi antara struktur data dan perilaku dalam satu entitas. Filosofi Object Oriented sangat luar biasa sepanjang siklus pengenbangan perangkat lunak (perencanaan, analisis, perancangan dan implementasi) sehingga dapat diterapkan pada perancangan sistem secara umum: menyangkut perangkat lunak, perangkat keras dan system secara keseluruhan.

***Tahapan-Tahapan Object Oriented Technology***

Pada Object Oriented Technology ada beberapa metode yang digunakan dlam pengembagan sistem. Salah satu yang terkenal adalah **OMT***(Object Modelling Technique)*yang diciptakan oleh **Rambough.**Aktivitas-aktivitas yang dilakukan dalam OMT ini adalah:

o Model Objek

o Model Dinamis

o Model Fungsional

Dalam pengembangan sistem berbasis objek diperlukan tahapan proses analisis yang akan dilanjutkan dengan tahapan desain/perancangan sistem.

***Keunggulan dan Kelemahan Object Oriented Technology***

**a. Keunggulan OMT**

**1. Uniformity**

Pengembang cukup menggunakan satu metodelogi dari tahap analisis hingga perancangan. Dengan adanya perkembangan ke arah aplikasi GUI (*graphical User interface)*, OMT memungkinkan merancangn user interface secara terintegrasi bersama dengan perancangan perangkat lunak sekaligus dengan perancangan basis data.

**2. Understandability**

Kode-kode yang dihasilkan dapat diorganisasi ke dalam kelas-kels yang berhubungan dengan masalah sesungguhnya sehingga lebih mudah dipahami.

**3. Stability**

Kode program yang dihasilkan relatif stabil sebab mendekati permasalahn sesungguhnya dilapangan.

**4. Reusability**

             Dimungkinkan penggunaan kembali kode-kode sehingga akan mempercepat waktu pengembangan perangkat lunak.

**b. Kelemahan OMT**

Metode berorientasi objek merupakan konsep yang relatif baru sehingga belum ada standar yang diterima semua pihak dalam menentukan tool apa yang digunakan sebagai dasar analisi serat perancangan perangkat lunak.

**g.**    **Model Functional Decomposition**

Metodologi ini menekankan pada pemecahan dari sistem ke dalam subsistem-subsistem yang lebih kecil, sehingga akan lebih mudah untuk dipahami, dirancang dan ditetapkan.

Yang termasuk dalam kelompok metodologi ini adalah :

- HIPO (Hierarchy plus Input Process Output)

- Stepwise Refinement (SR) atau Iterative Stepwise Refinement (ISR)

- Information Hiding

**h.**    **Data Oriented Methodologies**

Metodologi ini menekankan pada karakteristik dari data yang akan diproses.

Dikelompokkan ke dalam dua kelas, yaitu :

**1. *Data flow oriented methodologies*,** sistem secara logika dapat digambarkan secara logika dari arus data dan hubungan antar fungsinya di dalam modul-modul di sistem. Yang termasuk dalam metodologi ini adalah :

- SADT (Structured Analysis and Design Techniques)

- Composite Design

- SSAD (Structured System Analysis and Design)

**2. *Data Structured oriented methodologies,***Metodologi ini menekankan struktur dari input dan output di sistem. Yang termasuk dalam metodologi ini adalah :

- JSD (Jackson’s System Development)

- W/O (Warnier/Orr)

1. **PrescriptiveMethodologies**

**Yang termasuk dalam metodologi ini adalah :** **ISDOS** (Information System Design dan Optimization System), merupakan perangkat lunak yang dikembangkan di University of Michigan. Kegunaan dari ISDOS adalah mengotomatisasi proses pengembangan system informasi. ISDOS mempunyai dua komponen, yaitu :

**1. PSL** (Program Statement Language), merupakan komponen utama dari ISDOS, yaitu suatu bahasa untuk mencatat kebutuhan pemakai dalam bentuk machine readable form. PSL dirancang sehingga output yang dihasilkannya dapat dianalisis oleh PSA. PSL merupakan bahasa untuk menggambarkan sistemnya dan bukan merupakan bahasa pemrograman prosedural.

**2. PSA** (Program Statement Analyzer) merupakan paket perangkat lunak yang mirip dengan kamus data (data dictionary) dan digunakan untuk mengecek data yang dimasukkan, disimpan, dianalisis dan yang dihasilkan sebagai output laporan.

**j.**         **Model V**

Model ini merupakan perluasan dari model waterfall. Disebut sebagai perluasan karena tahap-tahapnya mirip dengan yang terdapat dalam model waterfall. Jika dalam model waterfall proses dijalankan secara linear, maka dalam model V proses dilakukan bercabang.

***Tahapan-Tahapan Model V***

**1.**       **Requirement Analysis & Acceptance Testing**

Tahap Requirement Analysis sama seperti yang terdapat dalam model waterfall. Keluaran dari tahap ini adalah dokumentasi kebutuhan pengguna. Acceptance Testing merupakan tahap yang akan mengkaji apakah dokumentasi yang dihasilkan tersebut dapat diterima oleh para pengguna atau tidak

**2.**       **System Design & System Testing**

Dalam tahap ini analis sistem mulai merancang sistem dengan mengacu pada dokumentasi kebutuhan pengguna yang sudah dibuat pada tahap sebelumnya. Keluaran dari tahap ini adalah spesifikasi software yang meliputi organisasi sistem secara umum, struktur data, dan yang lain. Selain itu tahap ini juga menghasilkan contoh tampilan window dan juga dokumentasi teknik yang lain seperti Entity Diagram dan Data Dictionary.

**3.**       **Architecture Design & Integration Testing**

Sering juga disebut High Level Design. Dasar dari pemilihan arsitektur yang akan digunakan berdasar kepada beberapa hal seperti: pemakaian kembali tiap modul, ketergantungan tabel dalam basis data, hubungan antar interface, detail teknologi yang dipakai.

**4.**       **Module Design & Unit Testing**

Sering juga disebut sebagai Low Level Design. Perancangan dipecah menjadi modul-modul yang lebih kecil. Setiap modul tersebut diberi penjelasan yang cukup untuk memudahkan programmer melakukan coding. Tahap ini menghasilkan spesifikasi program seperti: fungsi dan logika tiap modul, pesan kesalahan, proses input-output untuk tiap modul, dan lain-lain.

**5.**       **Coding**

Dalam tahap ini dilakukan pemrograman terhadap setiap modul yang sudah dibentuk.

***Kelebihan dan Kekurangan***

**1.**       **Kelebihan**

  V Model sangat fleksibel. V Model mendukung *project tailoring* dan penambahan dan pengurangan*method* dan *tool* secara dinamik. Akibatnya sangat mudah untuk melakukan *tailoring* pada V Model agar sesuai dengan suatu proyek tertentu dan sangat mudah untuk menambahkan *method* dan *tool*baru atau menghilangkan *method* dan *tool* yang dianggap sudah *obsolete*.

  V Model dikembangkan dan di-*maintain* oleh publik. *User* dari V Model berpartisipasi dalam*change control board* yang memproses semua *change request* terhadap V Model.

**2.**       **Kekurangan**

  V Model adalah model yang *project oriented* sehingga hanya bisa digunakan sekali dalam suatu proyek.

  V Model adalah model yang *project oriented* sehingga hanya bisa digunakan sekali dalam suatu proyek.

**Penggunaan**

V Model digunakan dalam proyek teknologi informasi di negara Jerman. Hal ini berlaku terutama untuk proyek teknologi informasi pada pada sektor pertahanan negara Jerman. Selain itu, V Model juga digunakan oleh *software developer* negara Jerman untuk proyek teknologi informasi lain.

**k.**        **Metode End-user Development**

Disini pengembangan dilakukan langsung oleh end-user. Keterlibatan langsung end-user sangat menguntungkan, karena memahami benar bagaimana sistem bekerja. Artinya tahap analisis sistem dapat dilakukan lebih cepat. Kelemahan adalah pada pengendalian mutu dan kecenderungan tumbuhnya “*private*”sistem informasi. Integrasi dengan sistem yang lain menjadi sulit.

***Tahapan-tahapan EUD***

**1.**       **Tahap inisasi (*initiation*)**

Yaitu tahap dimana organisasi(perusahaan) mulai pertama kali mngenal teknologi  
informasi.

**2.**       **Tahap ketularan (*contagion*)**

Yaitu tahap diamana organisasi (perusahaan) sudah mulai banyak yang menggunakan  
teknologi informasi meskipun ini dilakukan atau tidak terlalu mempertimbangkan  
untung ruginya dari penggunaan teknologi informasi ini.

**3.**       **Tahap kendali (*control*)**

Pada tahap ini organisasi (perusahaa) sudah mulai selektif di dalam penggunaan  
teknologi informasi. Ada hal yang dijadikan pertimbangan sebelum memutuskan  
penggunaan teknolgi informasi seperti pertimbangan untung rugi.

**4.**       **Tahap matang (*mature*)**

Pada tahap ini organisasi (perusahaan) menggunakan teknologi informasi tidak hanya  
mempertimbangakan keuntungan (benfit) yang akan didapatkan serta berapa biaya  
(cost) yang harus dikeluarkan tetapi lebih dari itu bagaimana teknologi informasi  
yang digunakan dapat dijadikan sebagai alat keunggulan di dalam bersaing

***Kelebihan dan Kekurangan***

**1.**       **Kelebihan**

  Dapat menghindari permasalahan kemacetan di departemen sistem informasi.

  Kebutuhan pemakai sistem dapat lebih terpenuhi karena dapat dikembangkan sendiri  
oleh pemakai.

  Menambah atau meningkatkan partisifasi aktif pemakai dalam proses pengembangan  
sistemnya sehingga akan ada kepuasan sendiri dari pemakai sistem.

  Dapat menambah kualitas pemahaman pemakai terhadap aplikasi yang dikembangkan  
serta teknollogi yang digunakan dalam sistem.

**2.**       **Kekurangan**

  Karena pemakai sistem harus mengembangkan aplikasinya sendiri, maka dalam hal ini  
pemakai sekaligus pengembang sistem dituntut untuk memiliki pemahaman mengenai  
teknologi informasi (computer literacy) serta pemahaman tentang pengembangan sistem infomasi.

  End user computing memiliki resiko dapat menggangu bahkan merusak sistem  
informasi di luar yang dikembangkan oleh pemakai sistem.

  End user computing pasti akan berhadapan dengan maslah kemampuan teknis  
pemakai sekaligus pengembang sistem.

**l.**       [**Soft System Methodology**](http://onlyhadi.wordpress.com/2010/02/26/soft-system-methodology/)

Soft systems methodology (SSM) merupakan sebuah pendekatan untuk memecahkan situasi masalah kompleks yang tidak terstruktur berdasarkan analisis holistic dan berpikir system. SSM juga merupakan sebuah metodologi partisipatori yang dapat membantu para stakeholders yang berbeda untuk mengerti perspektif masing-masing stakeholders. Fokus SSM adalah untuk menciptakan system aktivitas dan hubungan manusia dalam sebuah organisasi atau grup dalam rangka mencapai tujuan bersama.

Berpikir system merupakan suatu bidang transdisiplin yang muncul sebagai respon terhadap keterbatasan dari pendekatan teknikal dalam proses reduksi untuk memecahkan masalah. SSM dibangun selama tahun 1980an oleh organisasi/lembaga yang menyadari bahwa pendekatan mekanikal secara *top down* dalam mengorganisasi manajemen tidak bekerja secara cepat untuk mengubah lingkungan sekitarnya. SSM digunakan untuk memfasilitasi proses perubahan di banyak sektor swasta dan organisasi publik.

Dasar SSM berangkat dari pemikiran bahwa jika partisipasi seseorang dalam suatu proses menemukan situasi masalah serta cara untuk memperbaikinya, maka orang tersebut akan lebih suka untuk mengerti perbaikan yang diharapkan, merasa memiliki permasalahan tersebut, dan berkomitman untuk merubahnya SSM adalah sebuah metodologi yang cocok untuk membantu suatu organisasi dalam menjelaskan tujuan mereka dan kemudian merancang *sistem aktivitas manusia* untuk mencapai tujuan tersebut.

SSM didiskripsikan sebagai **tujuh tahap**proses analisis yang menggunakan konsep human activity dalam memahami situasi di sekitarnya untuk menentukan aksi yang perlu diambil dalam rangka mengembangkan situasi yang ada. Ketujuh tahap SSM tersebut adalah :  
•Identifikasi situasi masalah yang tidak terstruktur.

•Situasi masalah digambarkan lebih terstruktur.

•Membuat definisi awal dari sistem yang bersangkutan.

•Membuat dan menguji model secara konseptual.

•Membandingkan model konseptual dengan kenyataan.

•Mengidentifikasikan perubahan kemungkinan dan keinginan elemen sistem.

•Memperbaiki situasi atau memecahkan masalah.

SUMBER :

<http://danylukman.blogspot.com/2012/10/metode-pengembangan-sistem-informasi.html>

<http://lytya-24.blogspot.com/2012/01/perbedaan-sistem-objek-oriented-dan.html>

<http://onlyhadi.wordpress.com/2010/02/26/soft-system-methodology/>

<http://mohammadbadrul.blogspot.com/2010/04/apa-itu-soft-system-methodology.html>