

METODOLOGI PENGEMBANGAN SISTEM DAN SDLC (*Systems Development Life Cycle*)

1. Pengertian

Metodologi pengembangan sistem adalah suatu proses yang digunakan untuk mengembangkan sistem informasi. Metodologi meliputi (1) aktivitas yang dilakukan pada setiap langkah (fase) (2) peranan individu dan kelompok dalam setiap aktivitas (3) *deliverables* dan *standard* kualitas bagi setiap aktivitas dan (4) peralatan (“tools”) dan teknik yang digunakan untuk setiap aktivitas .

Mengapa pengembangan sistem menggunakan metodologi, karena:

- Memastikan pendekatan yang konsisten dalam setiap proyek.
- Mengurangi risiko yang berkaitan dengan kesalan dan menghindari jalan pintas.
- Menghasilkan dokumentasi yang konsisten dan lengkap untuk setiap proyek .

2. Prinsip Pengembangan Sistem

1. Melibatkan pengguna dan pemilik sistem (*Get the owners and users involved*).

Sangat penting dalam kesuksesan pengembangan sistem untuk melibatkan pengguna dan pemilik sistem. Luangkan waktu secukupnya kepada mereka dan pastikan bahwa mereka benar-benar terlibat dan mintalah persetujuan dalam semua keputusan yang melibatkan mereka.

2. Gunakan pendekatan penyelesaian masalah (*Use a problem-solving approach*).

Teliti dan mengerti masalah dan konteks sistem. Tentukan kebutuhan bagi setiap penyelesaian masalah. Ketahui alternatif pemecahannya dan coba pilih yang terbaik. Kemudian desain dan implementasikan alternatif penyelesaian masalah tersebut. Dan yang terakhir adalah meneliti nilai atau kesan penyelesaian masalah tersebut dan coba untuk memperbaikinya.

3. Tentukan fase dan aktivitas (*Establish phases and activities*).

Secara umum terdapat 5 fase dasar dalam pengembangan sistem, yaitu:

- Tinjauan sistem
- Analisis sistem
- Desain sistem
- Implementasi sistem
- Penunjang sistem

4. Tentukan Standar bagi Pembangunan dan Dokumentasi yang Konsisten (*Establish standards for consistent development and documentation*).

Standar pengembangan sistem biasanya menerangkan hal-hal sebagai berikut :

- Aktivitas
- Tanggungjawab
- Garis panduan @ kebutuhan dokumentasi
- Kriteria pengukuran kualitas

5. Lihat sistem sebagai investasi penting (*Justify systems as capital investments*).

Sistem informasi merupakan investasi penting, oleh karena itu terdapat dua isu yang harus dipertimbangkan, yaitu pemecahan masalah dan feasibilitas terutama dari segi signifikan biaya. Signifikan biaya adalah hasil perbandingan di antara biaya pengembangan yang dijalankan dengan manfaat sistem.

6. Jangan takut untuk membatalkan dan membetulkan ruang lingkup (Don't be afraid to cancel).

Adakalanya setiap proyek yang dibatalkan tersebut memiliki lebih rendah biayanya di bandingkan dengan apabila proyek tersebut diteruskan dengan memiliki banyak masalah. Kemudian menentukan "checkpoints" feasibilitas proyek dengan mempertimbangkan :

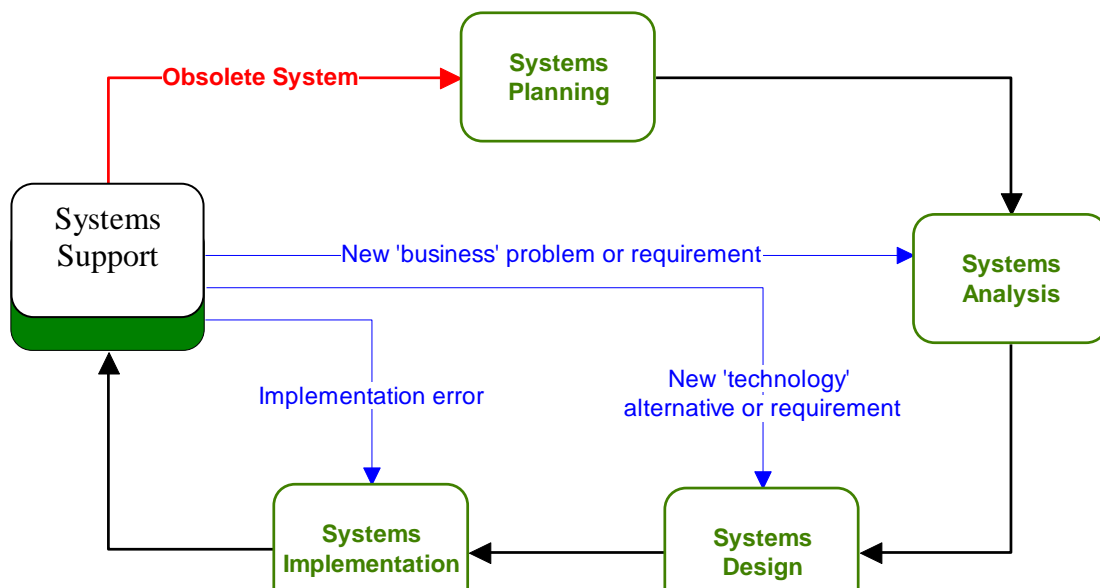
- Pembatalan proyek jika proyek tersebut tidak feasibel.
- Penilaian semula biaya dan jadwal jika ruang lingkup proyek dibesarkan.
- Kecilkan ruang lingkup jika anggaran dan jadwal dibekukan dan tidak mencukupi untuk menampung semua tujuan proyek .

7. Divide and Conquer

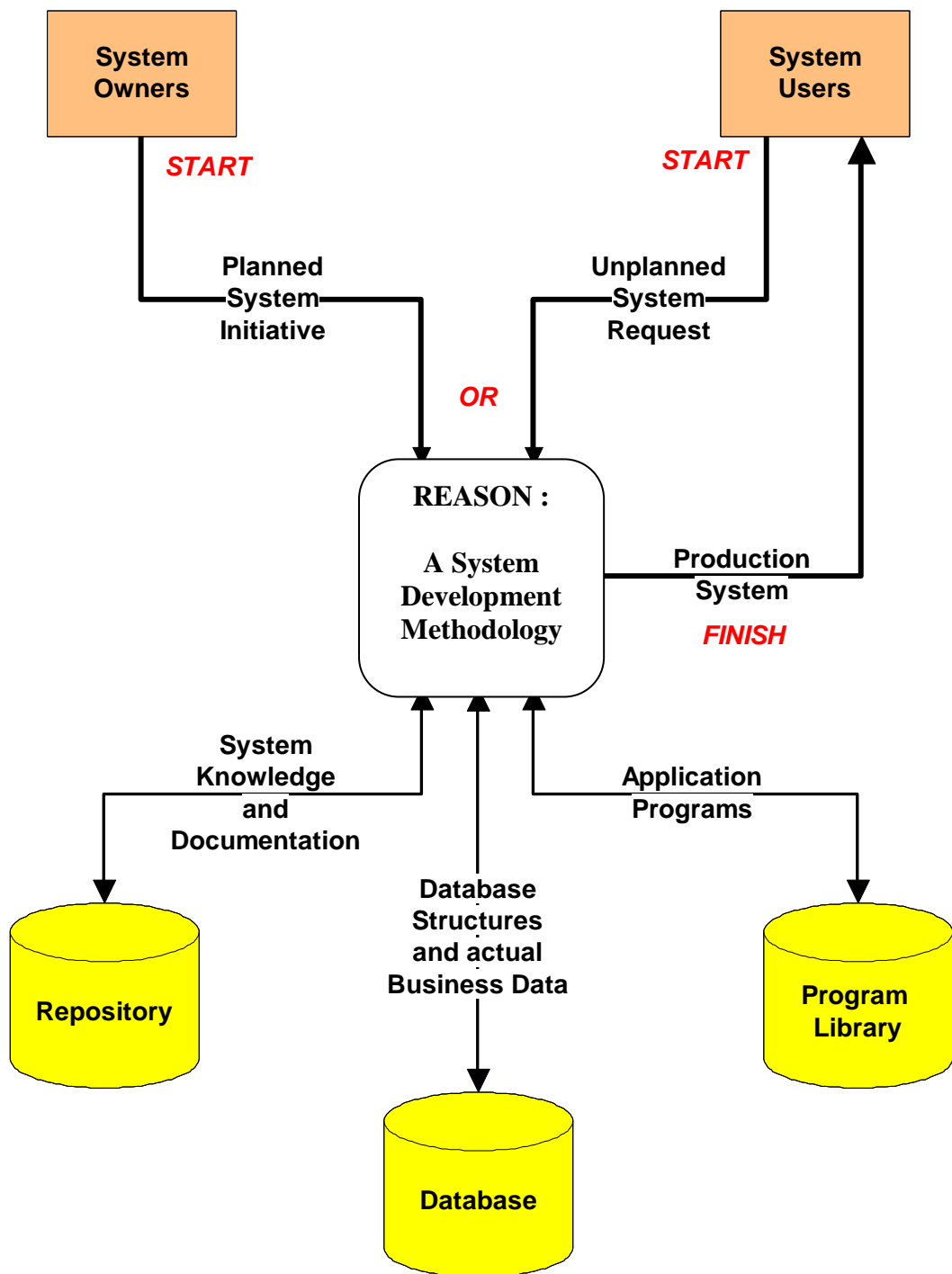
Setiap sistem merupakan sebahagian dari sistem yang lebih besar (super-systems). Dan memiliki sistem yang lebih kecil (subsystems). Pecahkan sistem kepada sub sistem supaya mudah untuk mengontrol masalah dan mengembangkan sistem yang lebih besar.

8. Design Systems for Growth and Change

Kebanyakan sistem analisis mengembangkan sistem mengikuti perkembangan yang secara langsung dirasakan pada saat ini. Entropy yaitu kerusakan yang biasa berlaku & tidak bisa dihindari oleh sistem. Dalam fase penunjang sistem biasanya biaya pelaksanaan melebihi biaya permulaan, keadaan ini menyebabkan sistem menjadi “obsolete”. Gambar 2 di bawah ini menerangkan fase-fase penunjang sistem dan Gambar 3 menggambarkan diagram konteks pengembangan sistem.



Gambar 2 : Systems Support and Entropy



Gambar 3 : The Context of System Development

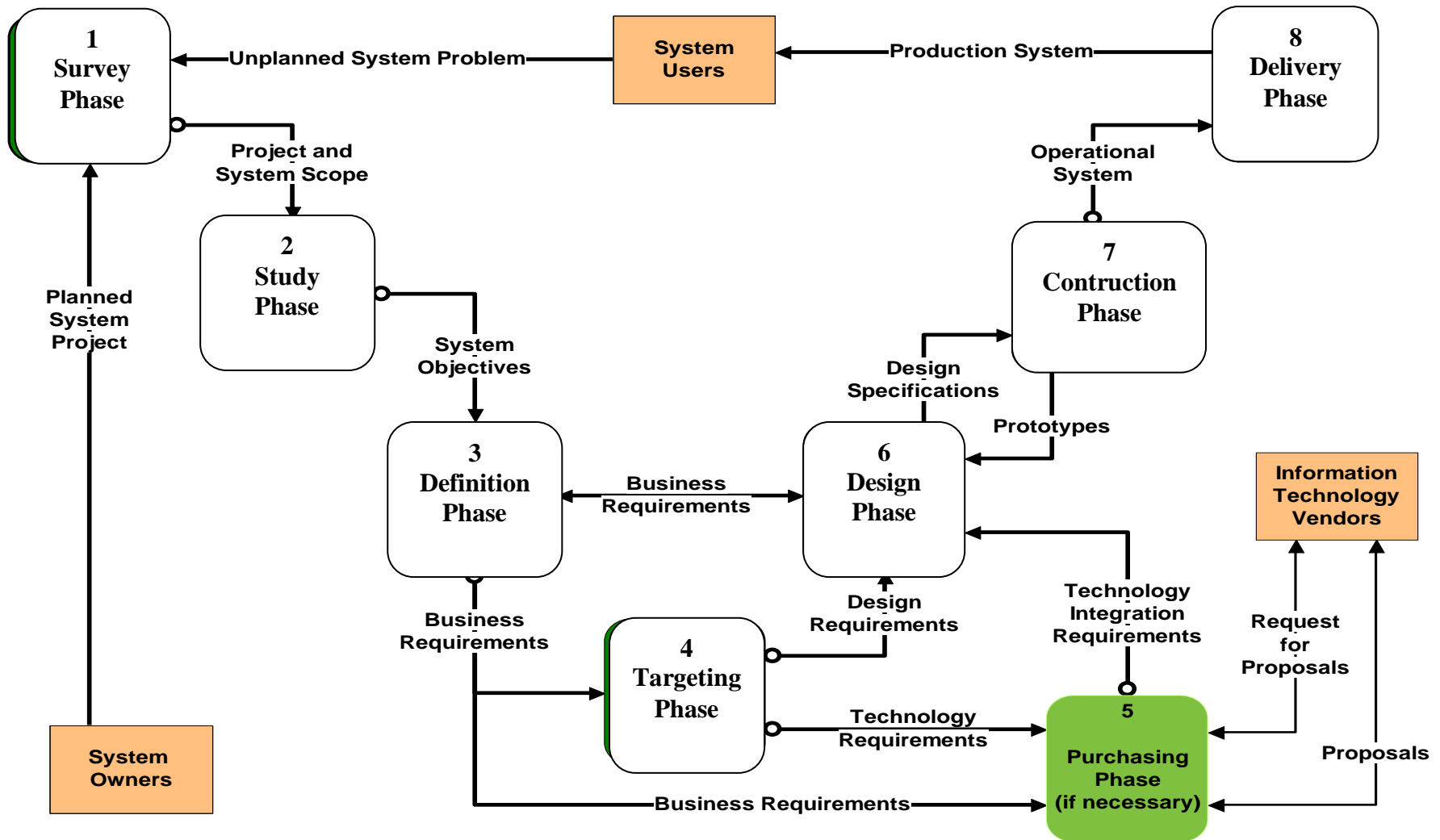
3. Metodologi Pengembangan Sistem : FAST (*Forecasting and Assessment in Science and Technology*)

Keterangan *FAST Life Cycle and Methodology*, terhadap simbol-simbol yang digunakan:

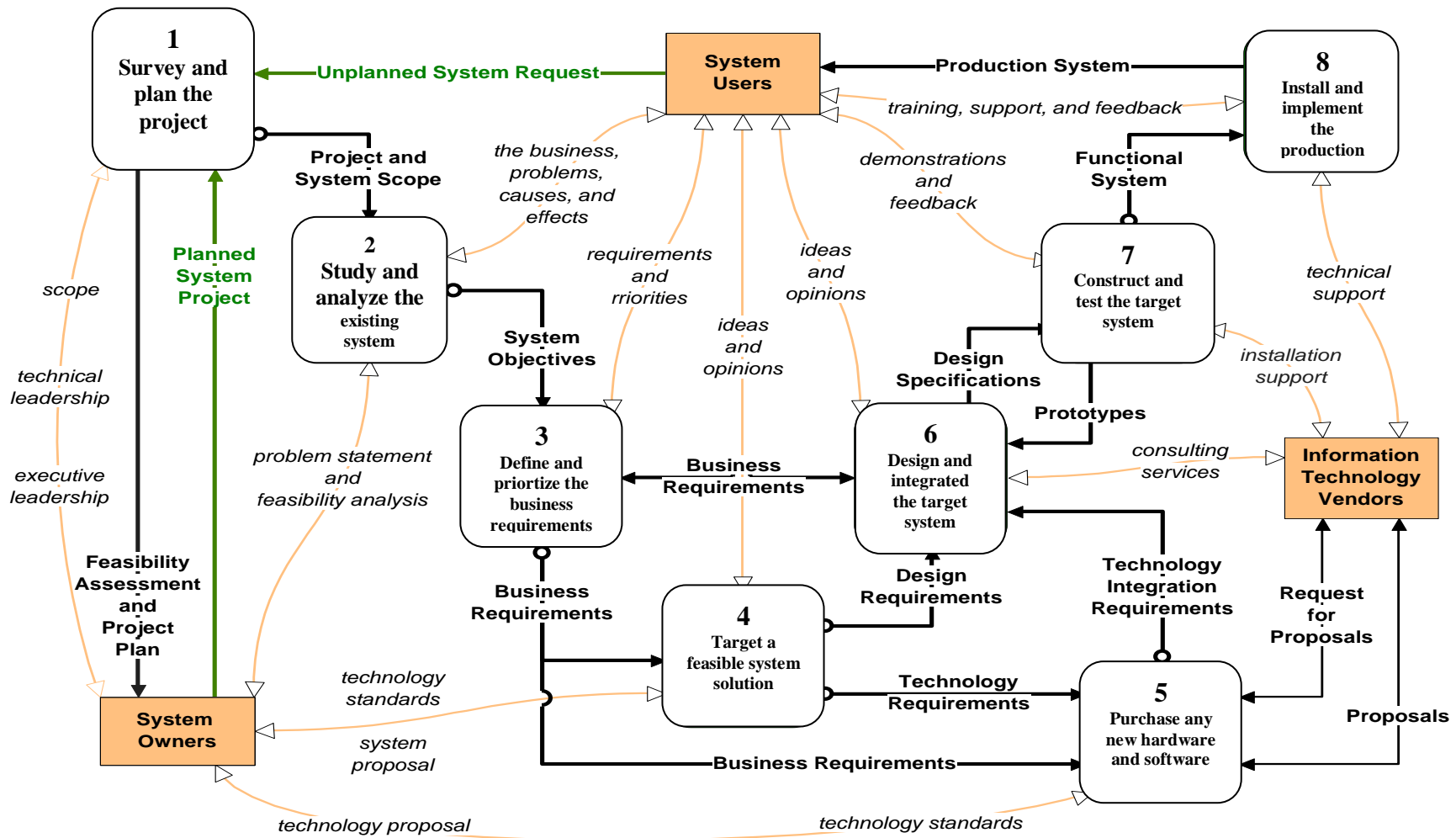
- Segiempat bulat: fase pengembangan sistem.
- Anak panah hijau tebal: aliran informasi yang memulai proyek *FAST*.
- Anak panah hitam tebal: deliverables (outputs) utama setiap fase. Setiap deliverable mempunyai dokumentasi dan/atau spesifikasi penting. Deliverable sebuah fase bisa menjadi input kepada fase yang lain.
- Anak panah hitam, tipis, dua arah: aliran informasi sekunder dan komunikasi. Aliran terdiri dari pembicaraan, musyawarah, surat, memo, laporan, dll.
- Segiempat: individu/organisasi yang berinteraksi dengan sistem analisis.
- Bulatan hitam: checkpoints di mana peserta proyek perlu menilai lagi feasibilitas dan/atau ruang lingkup proyek .

Gambar 4 , 5 dan 6 menerangkan tentang metodologi *FAST* yang terdiri dari 8 fase yaitu :

1. Fase Tinjauan : Penentuan konteks, ruang lingkup, anggaran, staf dan jadwal proyek .
2. Fase Kajian : Mengetahui dan menganalisis domain masalah bisnis dan teknikal untuk setiap masalah, sebab dan kesan tertentu.
3. Fase Definisi : Mengetahui dan menganalisis kebutuhan bisnis yang bisa digunakan untuk penyelesaian teknikal suatu masalah.
4. Fase Konfigurasi : Mengetahui dan menganalisis beberapa penyelesaian teknikal yang bisa mengatasi masalah dan memenuhi kebutuhan bisnis. Output fase ini adalah penyelesaian target dan feasibilitas.
5. Fase Pembelian : Mengetahui dan menganalisis produk perangkat keras dan perangkat lunak yang akan dibeli untuk kebutuhan proyek.
6. Fase Desain : Menentukan kebutuhan teknikal untuk penyelesaian target.
7. Fase Pembangunan : Mengembangkan dan menguji penyelesaian sistem (prototaip)
8. Fase Penghantaran : Menggunakan sistem dalam produksi sehari-hari.



Gambar 4 : System Development Phases



Gambar 5 : The FAST Phase Diagram

3.1 Fase Tinjauan

Fase ini menerangkan sistem dan proyek dari perspektif pemilik sistem. Contoh peranan pemilik sistem adalah:

- Sponsor eksekutif : pengurus paling atas yang membiayai proyek .
- Pelaksanas Proyek: Ketua proyek . Menentukan staf, anggaran dan jadwal.

Tujuan fase tinjauan:

- Menjawab soalan “Adakah proyek ini baik untuk dilihat?”
- Menakrif skop, masalah, peluang proyek .
- Menentukan peserta dan kumpulan, belanjawan dan jadual proyek .

Peserta dan Peranan : Fasilitator fase ini ialah bagian analisis sistem.

Input : Permintaan sistem

Aktivitas :

- Menentukan ruang lingkup atau ukuran proyek .
- Menjawab “Adakah proyek ini baik untuk ditinjau?”
- Dengan mengandaikan sistem baik untuk ditinjau, ketua proyek perlu merancang proyek. Ini termasuk membuat jadwal dan anggaran dasar dan menentukan staf pembangun sistem.

Output : Project charter yaitu menunjukkan hasil, saran dan perancangan staf kepada sponsor eksekutif.

3.2 Fase Kajian

Definisi proyek dan ruang lingkup sistem ada jika proyek telah setuju untuk diteruskan. Penilaian feasibilitas dan perencanaan proyek diteliti oleh pemilik sistem. Ada beberapa alternatif yang bisa diambil dari fase kajian ini, yaitu:

- Proyek disambung
- Menukar ruang lingkup proyek dan disambung

- Proyek ditolak
- Proyek ditangguhkan.

Tujuan Kajian :

- Kelompok proyek memahami masalah bisnis sepenuhnya.
- Menjawab “Adakah masalah (peluang dan sasaran) baik untuk diselesaikan”?
- Menentukan apakah sistem layak untuk dikembangkan.

Peserta dan peranan : Fasilitator fase ini ialah bagian analisis sistem yang bertugas untuk menerangkan sistem dan proyek dari perspektif pengguna sistem.

Input : Pernyataan ruang lingkup sistem dan proyek.

Aktivitas :

- Mempelajari istilah, sejarah dan budaya sistem
- Meneliti sebab dan kesan masalah, peluang dan sasaran.

Output : Pernyataan masalah dan hasil analisis feasibilitas (*laporan kajian terperinci*) yang diteliti oleh pemilik sistem.

3.3 Fase Definisi

Fase membedakan “apa” dengan “bagaimana” menakrif dan mengutamakan kebutuhan yang benar-benar diperlukan. Pengumpulan dan perbincangan kebutuhan dan prioritas dari pengguna sistem yang disahkan oleh pemilik sistem.

Tujuan : Mengenalpasti data, proses, interface dan kebutuhan database bagi pengguna sistem baru (Tidak melibatkan aspek teknikal)

Peserta dan Peranan :

- Fasilitator - analisis sistem.
- Pengguna sistem menentukan, menjelaskan dan mendokumentasikan kebutuhan bisnis.

Aktivitas:

- Mengetahui dan mengesahkan kebutuhan bisnis.
- Pendekatan – permodelan (melukis perwakilan gambar sistem)
- Pendekatan lain – prototaip (membuat model berskala kecil yang mencerminkan kebutuhan pengguna untuk disetujui)
- Menentukan prioritas kebutuhan (dikelaskan sebagai ‘wajib’, ‘dikehendaki’ atau ‘pilihan’)
- Deliverables (model akhir dan prototaip disusun dalam pernyataan kebutuhan bisnis)

3.4 Fase Konfigurasi

Fase ini merupakan fase untuk menspesifikasi kebutuhan bisnis yang lengkap agar bisa menampung idea dan pandangan semua kelas pengguna sistem. Dan juga untuk mengetahui perkembangan teknologi dari pemilik sistem.

Tujuan :

- Membuat beberapa penyelesaian terhadap kebutuhan bisnis.
- Mengetahui alternatif penyelesaian, menganalisis dan menyarankan satu sistem sasaran yang akan didesain dan dilaksanakan.

Peserta dan Peranan:

- Fasilitator - systems analyst.
- Semua peserta proyek termasuk pemilik, pengguna dan pedesain sistem.

Aktiviti :

- Mendefinisikan alternatif penyelesaian (beberapa pilihan teknikal dikhususkan oleh arsitektur teknologi yang disahkan oleh ketua proyek sistem)
- Setiap alternatif dinilai mengikut kriteria berikut :
 - Feasibilitas Teknikal : adakah penyelesaian praktikal secara teknikal? Apakah para pekerja mempunyai keahlian dalam mendesain dan mengembangkan sistem ?

- Feasibilitas Pengoperasian : apakah proyek memenuhi kebutuhan pengguna? bagaimana cara kerja para pekerja? apa pandangan pengguna?
 - Feasibilitas Ekonomi : apakah biaya yang dikeluarkan berjalan dengan efektif?
 - Feasibilitas Jadwal : apakah proyek bisa dijalankan dalam waktu yang relatif wajar ?
- Menyarankan pilihan yang feasibilitas sebagai sistem sasaran.
 - Deliverables
 - Menyarankan proyek sistem kepada pemilik sistem (dipresentasikan dan biasanya didiskusikan dengan pemilik sistem untuk keputusan bisnis dan keuangan yang pasti)
 - Jika keputusan untuk membeli kebutuhan (hardware dan software aplikasi) telah diputuskan maka kebutuhan tersebut diteruskan ke *fase perolehan*.
 - Kebutuhan desain harus disediakan untuk *fase desain*.
 - Postrequisites and Feasibility Checkpoints
 - Pemilik sistem bisa memilih apakah proyek tersebut disetujui atau tidak.
 - Berdasarkan keputusan, mulai *fase perolehan* atau batalkan fase desain (yang mungkin sudah dimulai) atau ubah skop/sasaran.
 - Analisis impak
 - fase ini tidak semestinya dijalankan jika ada *arkitektur aplikasi*. (Arkitektur aplikasi mendefinisikan set teknologi yang telah disetujui untuk digunakan sewaktu mengembangkan sistem baru).

3.5 Fase Perolehan

Tujuan : Untuk mengkaji pasaran IT, saran vendor dan menyarankan (kepada pengelola) saran terbaik yang memenuhi kebutuhan bisnis dan teknologi.

Peserta dan peranan :

- Fasilitator - systems analyst.
- Vendor IT (yang menjual hardware atau software).

- Pengguna (di luar dan di dalam proyek).
- Pemilik sistem.
- Agen pembelian dan badan hukum yang terlibat dalam persetujuan kontrak.

Input : Kebutuhan bisnis dari fase definisi dan kebutuhan teknologi dari Fase konfigurasi. Kemampuan teknologi (dari pengelola sistem) mesti dipatuhi.

Aktivitas :

- Mengkaji teknologi dan pasaran.
- Mengatur kebutuhan bisnis, teknologi dan menentukan mekanisme yang akan digunakan untuk menilai pilihan teknikal (kebutuhan dan mekanisme ini dinyatakan kepada vendor untuk mendapatkan penawaran).
- Vendor membalas dengan memberikan penawaran formal.
- Menilai proposal dan merujuk bbrp bhgn untuk menentukan (1) yang mana memenuhi kebutuhan dan spesifikasi, dan (2) yang paling cost effective.
- Juru Analis membuat cadangan kepada pemilik sistem (dan pengurus sistem)
- Agen berdaftar bisnis membuat pesanan, kontrak, lesen dan perjanjian servis yang muktamad.
- Deliverables
- proposal teknologi kepada pemilik sistem untuk memperolehi perkaksan/perisian tertentu.
- Jika diluluskan, kebutuhan integrasi teknologi dihantar kepada fase desain .
- Postrequisites and Feasibility Checkpoints
- ia diikuti oleh fase desain setelah perisian yang dibeli memenuhi kebutuhan bisnis dan teknologi proyek sepenuhnya.
- Jika tiada keputusan dibuat, projek diteruskan untuk didesain dan dibina sbg in-house.
- Analisis impak
- fase pilihan - berdasarkan buat- dan -beli dalam fase konfigurasi.

3.6 Fase Desain

Tujuan : menterjemah kebutuhan bisnis dari *fase definisi* kepada “blueprint” desain teknikal untuk dikembangkan.

Peserta dan Peranan :

- Facilitator - systems analyst.
- Database specialists : medesain /mengesahkan desain pangkalan data baru/yang dikembangkan.
- Network specialists : medesain /mengesahkan struktur rangkaian komputer.
- Microcomputer specialists : membantu medesain komponen software berdasarkan stasiun kerja.
- Human interface specialists : membantu medesain interface pengguna.
- Pengguna Sistem : Menilai fasilitas ; mempelajari, menggunakan dan kesesuaian sistem baru dengan kebutuhan bisnis.

Input :

- Kebutuhan bisnis dari fase definisi.
- Kebutuhan desain dari fase konfigurasi.
- Bagi proyek yang membeli hardware/software, fase ini juga menerima kebutuhan integrasi teknologi dari fase perolehan.
- Pengguna sistem memberi *idea dan pandangan* tentang desain sistem.

Activitas :

FAST menggabungkan fase desain dan pengembangan untuk menghasilkan pendekatan “*rapid application development*” (RAD) berdasarkan prototaip berulang. Strategi ini medesain dan mengembangkan sistem sebagai satu seri prototaip yang memperkirakan unpan balik pengguna.

Proses prototaip:

- Jelaskan ruang lingkup dasar versi sistem pertama (atau seterusnya).
- Jelaskan, desain, kembangkan dan buat pangkalan data.

- Jelaskan, desain, dan kembangkan input. Demo prototaip kepada pengguna sistem. (*Ulang langkah 3 sehingga pengguna merasa puas hati. Jika perlu, ulang langkah 1 untuk menambah kebutuhan baru kepada desain pangkalan data*).
- Jelaskan, desain dan kembangkan output. Demo prototaip kepada pengguna sistem. (*Ulang langkah 4 sehingga pengguna merasa puas hati. Jika perlu, ulang langkah 1 untuk menambah kebutuhan baru kepada desain pangkalan data, atau langkah 2 untuk menambah kebutuhan input baru*).
- Jelaskan, desain, dan kembangkan interface. Demo prototaip kepada pengguna sistem. (*Ulang langkah 5 sehingga pengguna merasa puas hati. Jika perlu, ulang langkah 1,2,3 untuk menambah kebutuhan baru pangkalan data, input atau output*).
- Medesain dan mengembangkan kontrol sistem yang tertinggal seperti sistem keamanan, backup, bongkar pasang, dll.
- Laksanakan versi sistem ini.
- Mulai langkah 1 untuk memulai lingkaran RAD bagi versi sistem seterusnya.

Ada beberapa bentuk spesifikasi desain, yang biasa untuk membuat permodelan, yaitu:

- Struktur pangkalan data.
- Struktur aplikasi keseluruhan.
- Interface pengguna keseluruhan 'look and feel'.
- Struktur rangkaian komputer.
- Struktur desain software kompleks yang akan ditulis.

3.7 Fase Pembangunan

Tujuan :

- Mengembangkan dan menguji satu sistem fungsi yang memenuhi kebutuhan bisnis dan desain
- Melaksanakan interface di antara sistem baru dan sistem langsung

Peserta dan Peranan :

- Facilitator - systems analyst.
- Juru analisa sebagai kontraktor untuk mengerjakan yang dilakukan oleh pakar teknikal atau subkontraktor.
- Tanggung jawab pengguna sistem terbatas kepada memberi unpan balik terhadap menggunakan fasilitas dan mempelajari fungsi sistem.

Input :

- spesifikasi desain (am atau terperinci).
- Vendor IT mungkin menyediakan *installation support* bg pakej perisian atau peralatan pembangunan perisian.

Aktivitas :

- Mengembangkan pangkalan data & rangkaian (menyediakan infrastruktur sistem) (*kecuali jika pangkalan data telah ada*).
- Memasang dan menguji paket software baru.
- Mengembangkan dan menguji pemrograman baru.
- Menguji pemrograman aplikasi; unit dan sistem.
 - Pengujian unit memastikan pemrograman aplikasi berfungsi sewaktu diuji secara berpisah dari pemrograman aplikasi lain.
 - Pengujian Sistem 'System tests' memastikan pemrograman aplikasi yang ditulis secara berpisah berfungsi dengan lancar sewaktu digabungkan dengan keseluruhan sistem lain.
- Deliverables
- Sistem fungsi.
- Strategi RAD dalam *FAST* menghasilkan beberapa prototaip.
- Postrequisites, Feasibility Checkpoints, and Impact Analysis
- Projek biasanya tidak dibatalkan pada peringkat ini karena fase ini "optional".

3.8 Fase Penghantaran

Tujuan : Memasang, menambah dan meletakkan sistem baru dalam operasi/produksi.

Peserta dan peranan :

- Facilitator - systems analyst.
- S.A. paling penting - mengemukakan masalah dan isu pelaksanaan di antara pengguna, pendesain dan pengembang sistem.
- Semua anggota projek terlibat secara aktif.
- Pemilik dan pengguna sistem menjadi ketua pada sistem baru.

Input : sistem fungsi.

- Pengguna Sistem memberi unpan balik secara terus menerus karena masalah dan isu sistem baru biasanya timbul.
- Bagi IT baru (hardware dan software), vendor IT memberikan *technical support* yang diperlukan.

Aktivitas :

- Melatih pengguna sistem.
- Penulisan manual.
- Memuatkan fail dan pangkalan data.
- Deliverables
- Sistem produksi bagi pengguna sistem.
- Senunjang dan latihan.
- Postrequisites, Feasibility Checkpoints, and Impact Analysis
- Projek tamat!.
- Mungkin penilaian sistem mulai lagi, metodologi dan anggota.

4. Metodologi *Life Cycle* Proyek

Selepas pengembangan sistem selesai perlu dibuat penunjang sistem, yaitu : pemeliharaan secara terus menerus terhadap sistem termasuk di dalamnya adalah program penyelenggaraan dan peningkatan sistem.

Aktivitas :

- Membetulkan kesalahan sistem.
- Membantu pengguna.
- Menyesuaikan sistem kepada kebutuhan baru.

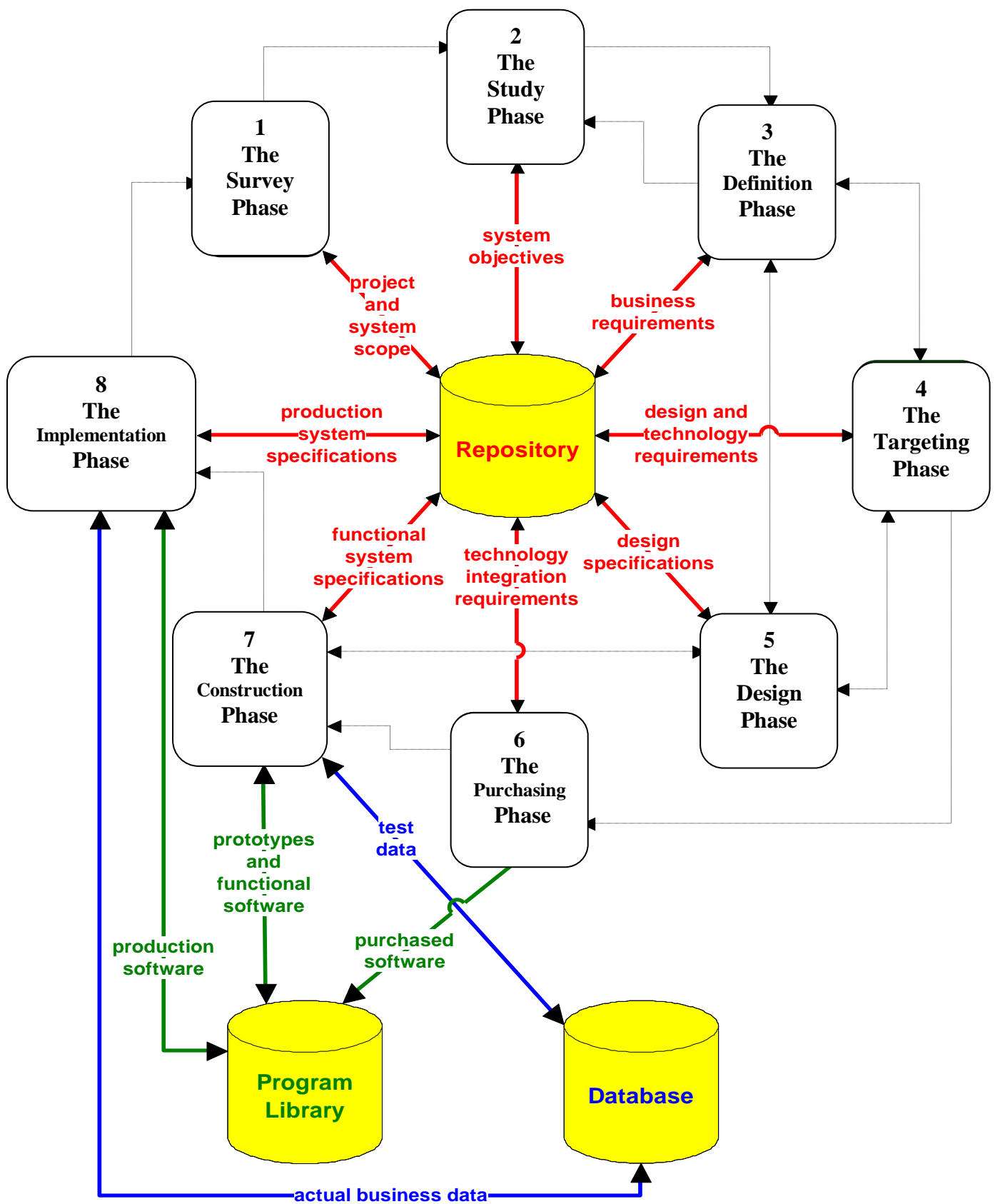
Secara umum aktivitas yang sering dijalankan dalam kebanyakan/semua metodologi adalah :

- Pencarian fakta : pengumpulan informasi; pengumpulan data, proses formal yang menggunakan kajian, wawancara, diskusi, angket, percontohan dan teknik-teknik lain untuk mengumpul informasi tentang kebutuhan dan keutamaan sistem.
- Dokumentasi dan presentasi : kemahiran berkomunikasi penting dalam mensukseskan proyek.
 - Dokumentasi; mengumpulkan fakta dan spesifikasi bagi kebutuhan sistem. Diantara dokumentasi adalah: versi sistem langsung, versi produksi sistem langsung, versi sistem yang melalui aktivitas pengembangan dan pengujian dan versi sistem di sekitar *life cycle* (untuk menghasilkan versi baru)
 - Presentasi; mempresentasikan dokumentasi untuk dinilai oleh pemilik dan pengguna secara tertulis ataupun lisan.
- Anggaran dan pengukuran : untuk melihat kualitas dan produktivitas sistem.
 - Anggaran : menganggarkan waktu, tugas, biaya dan manfaat membangunkan sistem.
 - Pengukuran : mengukur dan menganalisis produktivitas dan kualitas, biaya pembangun sistem.

Terdapat dua pendekatan yang bisa digunakan di dalam menganggarkan proyek, yaitu:

- Mengelakan penganggaran takut, tidak pasti atau kurang keyakinan

- Menggunakan pengalaman dan data (own + others) dari proyek sebelumnya dan memperbaiki anggaran secara terus menerus.
- Analisis feasibilitas : penilaian terhadap manfaat pembangunan sistem informasi kepada organisasi atau perusahaan.
- Pengelolaan proyek dan proses :
 - Pengelolaan proyek : aktivitas terus menerus di mana bagian analisis merancang, mendistribusikan, mengarahkan dan mengontrol proyek untuk membangun sistem mengikuti biaya dan waktu yang ditetapkan. Kebanyakan proyek gagal dikarenakan kepemimpinan dan pengelolaan yang lemah.
 - Pengelolaan proses : aktivitas yang terus menerus dalam menetapkan kebutuhan bagi aktivitas proses, manfaat, peralatan dan output yang akan dihasilkan. Tujuan pengelolaan proses adalah untuk menyelaraskan pendekatan dan output proyek .



Gambar 6 : The FAST Phase Diagram

5. Metodologi CASE (*Computer-Aided Systems Engineering*)

CASE merupakan aplikasi teknologi informasi untuk aktivitas, teknik, metodologi pengembangan sistem.

Peralatan CASE adalah pemrograman (software) yang mengotomatiskan atau mendukung fase-fase SDLC(Systems Development Life Cycle).

Tujuan CASE adalah untuk mempercepat proses pengembangan sistem dan meningkatkan kualitas sistem yang dihasilkan.

Sebetulnya CASE bukan satu metodologi atau alternatif kepada metodologi, namun CASE lebih berkecenderungan untuk mendukung strategi, teknik dan output metodologi.