

## BAB II

### TINJAUAN PUSTAKA

#### 2.1 Konsep Sistem Informasi

##### 2.1.1 Sistem

Sistem pada dasarnya adalah sekelompok unsur yang erat hubungannya satu dengan yang lain, yang berfungsi bersama-sama untuk mencapai tujuan tertentu. Dari definisi ini dapat dirinci lebih lanjut pengertian sistem secara umum, yaitu sebagai berikut: [2]

- a. Setiap sistem terdiri dari berbagai unsur. Unsur-unsur suatu sistem terdiri dari subsistem yang lebih kecil, yang terdiri pula dari kelompok-kelompok unsur yang membentuk subsistem tersebut.
- b. Unsur-unsur tersebut merupakan bagian yang tak terpisahkan dari sistem yang bersangkutan. Unsur-unsur sistem berhubungan erat satu sama lain dimana sifat serta kerja sama antarunsur dalam sistem tersebut mempunyai bentuk tertentu.
- c. Unsur-unsur didalam sistem tersebut bekerja sama untuk mencapai tujuan sistem. Setiap sistem mempunyai tujuan tertentu.
- d. Suatu sistem merupakan bagian dari sistem lain yang lebih besar.

Suatu sistem dibuat untuk menangani sesuatu yang berulang kali atau yang secara rutin terjadi. Pendekatan sistem merupakan suatu filsafat atau persepsi tentang struktur yang mengkoordinasikan kegiatan-kegiatan dan operasi-operasi dalam suatu organisasi dengan cara yang efisien dan yang paling baik. Suatu sistem dapat dirumuskan sebagai setiap kumpulan komponen atau subsistem yang dirancang untuk mencapai suatu tujuan.

Suatu sistem memiliki karakteristik atau sifat-sifat tertentu yaitu:

- a. Memiliki Komponen (*Component*)

Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, yang bekerja sama membentuk suatu kesatuan. Komponen-komponen sistem tersebut dapat berupa suatu bentuk subsistem. Setiap subsistem memiliki sifat-sifat sistem yang menjalankan suatu fungsi tertentu dan mempengaruhi proses sistem secara

keseluruhan. Suatu sistem dapat mempunyai sistem yang lebih besar yang disebut dengan supra sistem.

b. Memiliki Batasan (*Boundary*)

Ruang lingkup sistem merupakan daerah yang membatasi antara sistem dengan sistem lainnya atau sistem dengan lingkungan luarnya. Batasan sistem ini memungkinkan suatu sistem dipandang sebagai satu kesatuan yang tidak dapat dipisah-pisahkan.

c. Memiliki Lingkungan Luar Sistem (*Environment*)

Bentuk apapun yang ada diluar ruang lingkup atau batasan sistem yang mempengaruhi operasi sistem tersebut disebut dengan lingkungan luar sistem. Lingkungan luar sistem ini dapat menguntungkan dan dapat juga merugikan sistem tersebut. Lingkungan luar yang menguntungkan merupakan energi bagi sistem tersebut, yang dengan demikian lingkungan luar tersebut harus selalu dijaga dan dipelihara. Sedangkan lingkungan luar yang merugikan harus dikendalikan.

d. Memiliki Antar Muka (*Interface*)

Media yang menghubungkan sistem dengan subsistem yang lain disebut dengan penghubung sistem atau *interface*. Penghubung ini memungkinkan sumber-sumber daya mengalir dari satu subsistem ke subsistem yang lain. *Output* suatu subsistem akan menjadi masukan untuk subsistem yang lain dengan melewati penghubung. Dengan demikian terjadi suatu integrasi sistem yang membentuk satu kesatuan.

e. Memiliki Masukan (*Input*)

Energi yang dimasukkan ke dalam sistem dapat berupa pemeliharaan (*Maintenance Input*) dan sinyal (*Signal Input*). *Maintenance input* merupakan *input* yang dimasukkan agar suatu sistem dapat beroperasi. *Signal Input* merupakan *input* yang diproses untuk mendapatkan *output*.

f. Memiliki Keluaran (*Output*)

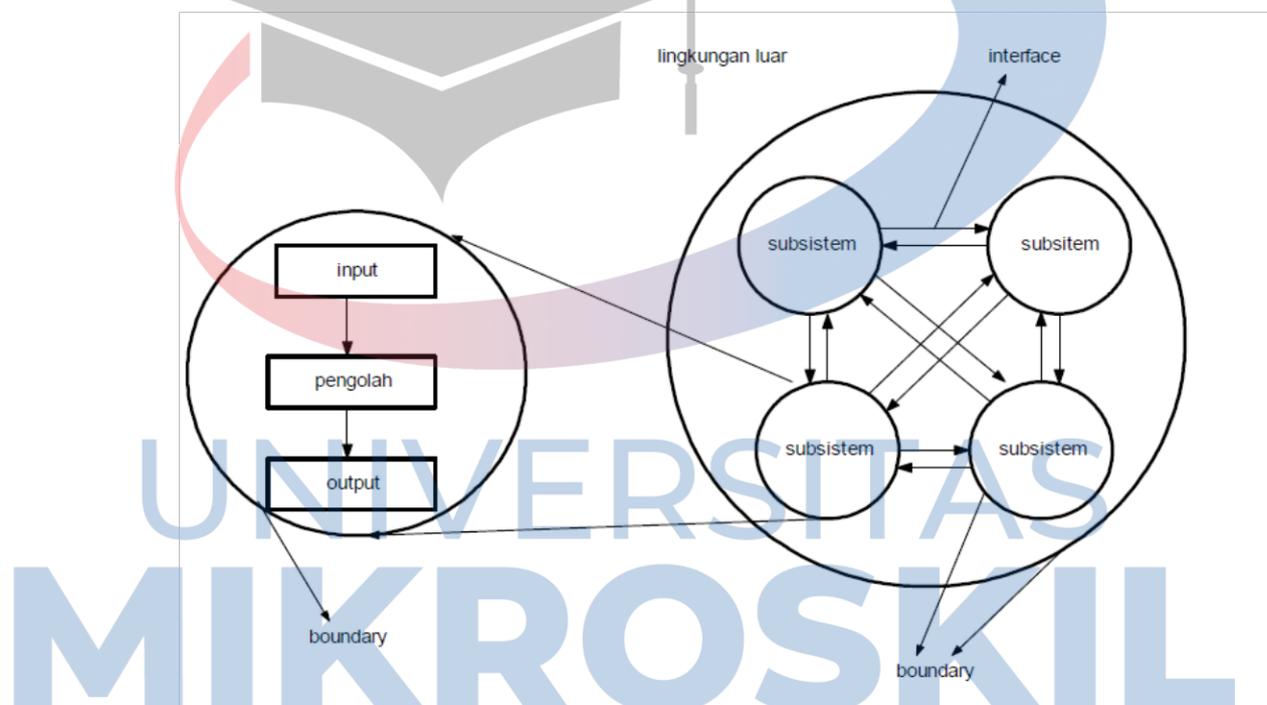
Hasil dari energi yang diolah dan diklasifikasikan menjadi *output* yang berguna. *Output* ini merupakan masukan bagi subsistem yang lain. Seperti contoh sistem informasi, *output* yang dihasilkan adalah informasi, dimana informasi ini dapat digunakan sebagai masukan untuk pengambilan keputusan atau hal-hal lain yang merupakan *input* bagi subsistem lainnya.

g. Memiliki Pengolah dan Pemrosesan Sistem (*Process*)

Suatu sistem dapat mempunyai suatu proses yang akan mengubah masukan menjadi *output*.

h. Memiliki Sasaran dan Tujuan Sistem (*Objective and Goal*)

Suatu sistem memiliki tujuan dan sasaran yang pasti dan bersifat deterministik. Kalau suatu sistem tidak memiliki sasaran, maka operasi sistem tidak ada gunanya. Suatu sistem dikatakan berhasil bila mengenai sasaran atau tujuan yang telah direncanakan.



Gambar 2.1 Karakteristik Sistem

### 2.1.2 Informasi

Informasi adalah data yang telah diklasifikasi atau diinterpretasi untuk digunakan dalam proses pengambilan keputusan. Sistem pengolahan informasi mengolah data menjadi informasi atau tepatnya mengolah data dari bentuk tak berguna menjadi berguna bagi penerimanya. Nilai informasi berhubungan dengan keputusan, maka informasi diperlukan, keputusan dapat berkisar dari keputusan berulang sederhana sampai keputusan strategis jangka panjang [3].

Informasi adalah sekumpulan data/fakta yang diorganisasi atau diolah dengan cara tertentu sehingga mempunyai arti bagi penerima. Data yang telah diolah menjadi sesuatu yang berguna bagi si penerima maksudnya yaitu dapat memberikan keterangan atau pengetahuan. Dengan demikian yang menjadi sumber informasi adalah data. Informasi dapat juga dikatakan sebuah pengetahuan yang diperoleh dari pembelajaran, pengalaman, atau instruksi [4].

Suatu informasi dapat dikatakan sebagai informasi yang berkualitas dengan karakteristik sebagai berikut:

a. Akurat

Informasi mencerminkan keadaan yang sebenarnya.

b. Lengkap

Informasi yang tidak lengkap tidak menggambarkan keseluruhan isi sehingga diperlukan informasi yang lengkap.

c. Relevan

Informasi yang diberikan harus sesuai dengan yang dibutuhkan.

d. Tepat Waktu

Informasi itu harus tersedia atau ada pada saat informasi tersebut dibutuhkan.

### 2.1.3 Sistem Informasi

Sistem informasi adalah suatu sistem didalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian yang mendukung fungsi operasi organisasi yang bersifat manajerial dengan kegiatan strategi dari suatu organisasi untuk dapat menyediakan kepada pihak luar tertentu dengan laporan-laporan yang diperlukan [3].

Sistem informasi adalah kumpulan komponen yang saling terorganisir untuk mengumpulkan, menyimpan, dan memproses data dengan tujuan untuk memberikan informasi yang dapat digunakan. Dalam bisnis dan organisasi lainnya, informasi adalah hal penting yang baik untuk kegiatan operasi dan operasi manajemen.

Sistem informasi terdiri dari komponen-komponen yang disebut blok bangunan, yang terdiri dari blok masukan, blok model, blok *output*, blok teknologi, blok basis data, dan blok kendali. Sebagai suatu sistem, keenam blok tersebut masing-masing

saling berinteraksi satu dengan yang lain membentuk suatu kesatuan untuk mencapai sasaran. Komponen tersebut terdiri dari: [3]

a. *Input*

*Input* adalah metode dan media untuk menangkap data yang akan dimasukkan, yang dapat berupa dokumen-dokumen dasar.

b. Model

Blok model terdiri dari kombinasi prosedur, logika, dan model matematik yang akan memanipulasi data *input* sehingga data yang tersimpan di basis data dapat diolah dengan cara tertentu untuk menghasilkan *output* yang diinginkan.

c. *Output*

Produk dari sistem informasi adalah *output* yang merupakan informasi yang berkualitas dan dokumentasi yang berguna untuk semua tingkatan manajemen serta semua pemakai sistem.

d. Teknologi

Teknologi digunakan untuk menerima *input*, menjalankan model, menyimpan dan mengakses data, menghasilkan dan mengirimkan *output*, dan membantu pengendalian dari sistem secara keseluruhan. Yang termasuk adalah perangkat keras (*hardware*), perangkat lunak (*software*), dan teknisi (*brainware*).

e. *Database*

*Database* memungkinkan kumpulan data saling berhubungan satu sama lainnya, tersimpan di perangkat keras komputer dan menggunakan perangkat lunak untuk memanipulasinya. Data perlu disimpan dalam basis data untuk keperluan penyediaan informasi lebih lanjut. Data didalam basis data perlu diorganisasikan sedemikian rupa agar informasi yang dihasilkan berkualitas. Organisasi basis data yang baik juga berguna untuk efisiensi kapasitas penyimpanannya. Basis data diakses atau dimanipulasi menggunakan perangkat lunak paket yang disebut *Database Management System (DBMS)*.

f. Kontrol

Banyak hal yang dapat merusak sistem informasi sehingga pengendalian perlu dirancang dan diterapkan untuk meyakinkan bahwa hal-hal yang dapat merusak

sistem dapat dicegah ataupun bila terlanjur terjadi kesalahan-kesalahan dapat langsung cepat diatasi.

## 2.2 Basis Data

Basis data (*database*) adalah suatu pengorganisasian sekumpulan data yang saling terkait sehingga memudahkan aktivitas perolehan informasi. Basis data dimaksudkan untuk mengatasi masalah pada sistem yang memakai pendekatan berbasis berkas [5]. Untuk mengelola dan mengorganisir *database* yang dibangun dalam sebuah SI, dibutuhkan suatu sistem pengelola *Database Management System* (DBMS).

*Database Management System* (DBMS) adalah suatu sistem atau *software* yang menentukan bagaimana data diorganisir, disimpan, diubah, diambil kembali, pengaturan mekanisme pengamanan data, mekanisme pemakaian data secara bersama, mekanisme pengolahan data dalam lingkungan SI *multi-user*, dan lain-lain.

Terdapat beberapa teknologi *database* yaitu:

### a. *Hierarchical Database*

*Hierarchical database* menggambarkan kumpulan *record* yang dihubungkan satu sama lain melalui hubungan berdasarkan *pointer* yang membentuk struktur pohon.

### b. *Network Database*

*Network database* merupakan *database* yang terdiri atas sekumpulan *record* yang dihubungkan melalui *pointer* yang membentuk hubungan antar *record* dalam bentuk *ring*.

### c. *Relational Database*

*Relational Database* berisi kumpulan tabel, dimana setiap tabel mempunyai nama dan struktur yang unik. Dalam setiap tabel, masing-masing *record* data diorganisir dalam struktur yang sama dan memiliki *field* kunci yang akan menjadi penghubung antar tabel yang ada dan berkaitan satu sama lain.

### d. *Object Oriented and Multimedia Database*

Teknologi pengelolaan *database* terus berkembang seiring dengan perkembangan pemodelan data dan teknik pemrograman. *Object Oriented Database* (OOD) merupakan tanggapan terhadap perkembangan teknik pemrograman berorientasi objek yang menekankan pada objek, atribut, dan metode.

e. *Web Database*

Pada sistem *web* yang statis, halaman *web* hanya berfungsi untuk menyajikan informasi-informasi. Sementara itu, penambahan fasilitas seperti video atau audio dapat membuat halaman *web* tampak dinamis. Pembangunan *web* yang interaktif dapat diupayakan dengan mengintegrasikan halaman *web* dengan *Database Management System* (DBMS).

f. *Data Warehouse*

*Data Warehouse* adalah sebuah *database* dengan peralatan pembuatan laporan dan *query* yang menyimpan data masa sekarang dan data masa lalu yang dipadukan dari berbagai sistem operasional. Data ini akan digunakan untuk menyajikan laporan dan melakukan analisis guna mendukung pengambilan keputusan manajerial.

Fungsi kedua dari DBMS adalah memproses *database*. Kegiatan proses tersebut kompleks, tetapi secara fundamental, DBMS memungkinkan suatu aplikasi melakukan 4 kegiatan operasi proses: baca, *insert*, modifikasi, dan *delete*. Operasi ini di-*request* saat DBMS dipanggil.

Dalam pengembangannya, struktur basis data dibagi antara basis data relasional (SQL) dan basis data non-relasional (NoSQL). Walaupun kedua basis data merupakan opsi yang tersedia, terdapat beberapa elemen penting dalam pemilihan struktur data basis data tersebut.

Perbedaan utama antara kedua database adalah sebagai berikut:

SQL	NoSQL
Relational Database Management System	Non-relational / Distributed Database System
Memiliki skema statis	Memiliki skema dinamis
Tidak cocok digunakan pada penyimpanan data yang bersifat hierarkis	Cocok digunakan pada penyimpanan data yang bersifat hierarkis
Cocok digunakan dengan <i>query</i> yang rumit	Tidak cocok digunakan dengan <i>query</i> yang rumit

### 2.3 Metode Pengembangan Aplikasi Cepat (PAC)

Pengembangan Aplikasi Cepat (PAC) adalah model proses perkembangan *software* sekuensial linier yang menekankan siklus perkembangan yang sangat singkat. Model PAC ini merupakan sebuah adaptasi kecepatan tinggi dari model sekuensial linier dimana perkembangan cepat dicapai dengan menggunakan pendekatan konstruksi berbasis pada komponen. Model proses ini mengacu pada pendekatan *prototyping* sistem untuk dikembangkan [6].

*Prototyping* adalah pengembangan/bagian awal dari suatu sistem yang berjalan yang dikembangkan dengan cepat dan relatif murah, dengan tujuan untuk memodifikasinya. Kapabilitas sistem awal *prototype* terbatas jika dibandingkan dengan tujuan dari sistem akhir. Pengembangan *prototype* terutama membantu dalam pengembangan tampilan depan dari suatu sistem. *Prototyping* diserahkan kepada *user*, yang mencoba dan memberikan masukan untuk dimodifikasi. Disaat proses pengembangan ini, baik *user* maupun pengembang dapat mendapati apa yang dibutuhkan oleh *user*.

Tujuan dari *prototyping* adalah untuk dapat mengembangkan *prototype* yang lebih kompleks sehingga terciptalah suatu spesifikasi fungsional terhadap apa yang menjadi tujuan dari sistem tersebut.

*Prototyping* dibagi ke beberapa jenis konsep, diantaranya:

#### a. *Prototype Patched-up*

*Prototyping* yang berkaitan dengan penyusunan sistem yang secara *patch* bersama-sama. Dalam bidang teknis, pendekatan ini disebut juga dengan *breadboarding*, yaitu menciptakan suatu *patch* sejalan dengan model kerja sirkuit yang terintegrasi.

#### b. *Prototype Non-operasional*

*Prototype* dari model skala yang menganggur yang disusun untuk menguji beberapa rancangan tertentu. Model skala yang menganggur dari suatu sistem informasi bisa diproduksi bila pengkodean yang diperlukan oleh aplikasi terlalu mahal untuk *prototype*, kecuali tersedianya gagasan yang dianggap sangat bermanfaat dari sistem bisa diperoleh melalui *prototyping input* dan *output* saja.

#### c. *Prototype First-of-Series*

Konsep penciptaan suatu model skala lengkap pertama dari sistem, yang disebut pilot. *Prototyping* semacam ini digunakan saat merencanakan beragam versi dari

sistem informasi yang sama. Model kerja skala lengkap ini memungkinkan *user* bisa mendapat pengalaman interaksi yang realistis dengan sistem baru, yang sebelumnya meminimalkan biaya untuk menanggulangi masalah yang ada.

d. *Prototype* Fitur Terpilih

Konsep yang berkaitan dengan pembangunan suatu model operasional yang mencakup beberapa, tetapi tidak semua fitur yang dimiliki sistem akhir. Bila jenis *prototyping* ini dilakukan, sistem akan tercipta sesuai modul sehingga bila fitur-fitur yang dibuat *prototype*-nya dianggap berhasil, maka akan diintegrasikan ke dalam sistem versi akhir yang besar.

Fase-fase dalam PAC mencakup, yaitu: [6]

a. Fase Perencanaan Kebutuhan

Fase perencanaan kebutuhan mengkombinasikan elemen-elemen dari perencanaan sistem dan analisis sistem. *User*, manajer, dan staf TI berdiskusi dan menentukan kebutuhan bisnis, cakupan proyek, batasan-batasan, dan kebutuhan sistem. Perencanaan kebutuhan berakhir ketika tim telah menyetujui isu yang dihadapi dan mendapatkan persetujuan manajemen untuk dilanjutkan.

b. Fase Desain *User*

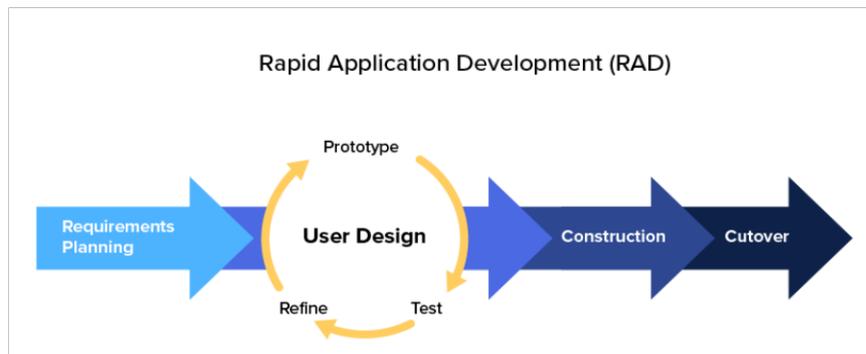
Dalam fase desain *user*, *user* berinteraksi dengan analis sistem dalam mengembangkan model dan *prototype* yang merepresentasikan semua proses sistem, *output*, dan *input*. Fase ini adalah fase berlanjut, proses interaktif yang memungkinkan *user* untuk mengerti, memodifikasi, dan akhirnya menyetujui model berjalan yang sesuai dengan kebutuhan.

c. Fase Konstruksi

Dalam fase ini, *user* tetap berpartisipasi dan menyarankan perubahan dan perbaikan sejalan dengan laporan yang dibuat.

d. Fase *Cutover*

Fase *cutover* adalah fase terakhir dimana implementasi, konversi data, testing, pergantian ke sistem baru, dan pelatihan *user* akan dilaksanakan.



Gambar 2.2 Fase-Fase Pengembangan Aplikasi Cepat (PAC)

## 2.4 Teknik Pengembangan Sistem

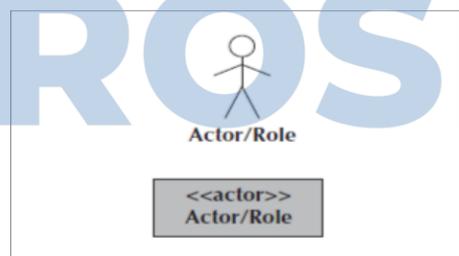
### 2.4.1 Model Use Case

*Use case diagram* menggambarkan interaksi-interaksi antar sistem dengan sistem eksternal dan *user*. *Use case* menjelaskan siapa yang akan menggunakan sistem dan bagaimana *user* dapat berinteraksi dengan sistem secara grafis. *Use case* digambarkan ketika menetapkan kebutuhan dari sistem [7].

Terdapat beberapa elemen dari *use case diagram*, yaitu: [7]

#### a. Aktor

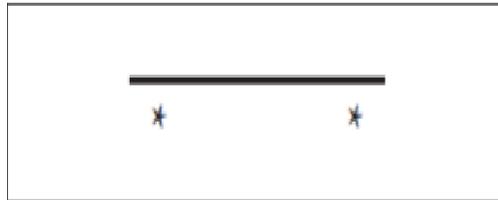
Aktor bukanlah *user* spesifik melainkan peran yang dapat digunakan oleh *user* ketika berinteraksi dengan sistem. Aktor juga dapat merepresentasikan sistem lain yang sedang diinteraksikan oleh sistem masa ini. Aktor dapat menyediakan *input*, menerima *output* dari sistem, ataupun keduanya.



Gambar 2.3 Aktor

#### b. Asosiasi

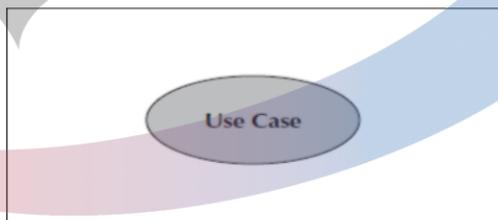
*Use case* dihubungkan ke aktor melalui asosiasi; relasi ini menunjukkan *use case* manakah yang berinteraksi dengan aktor. Sebuah garis yang dihubungkan dari aktor ke *use case* menggambarkan sebuah asosiasi. Asosiasi biasanya merepresentasikan komunikasi dua arah antara *use case* dan aktor.



Gambar 2.4 Asosiasi

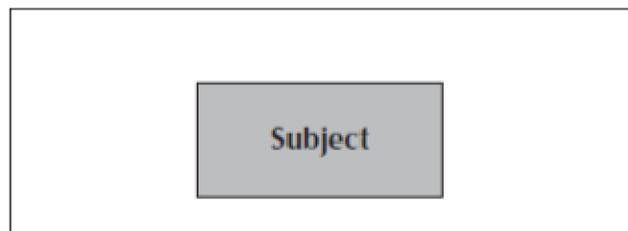
c. *Use case*

*Use case* adalah proses utama yang dijalankan oleh sistem yang menguntungkan aktor-aktor. *Use case* menggambarkan fungsi-fungsi dalam sistem dari sudut pandang *user* dengan cara yang mereka pahami. *Use case* merepresentasikan suatu tujuan dari sistem dan menggambarkan suatu rangkaian kegiatan dan interaksi *user* untuk mencapai suatu tujuan.

Gambar 2.5 *Use Case*

## d. Subjek

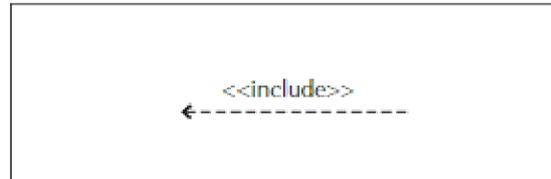
Subjek mendefinisikan cakupan dari sistem dan menggambarkan bagian-bagian diagram mana yang merupakan bagian eksternal dan internal. Subjek dapat digunakan untuk memisahkan suatu perangkat lunak dari lingkungan subsistem lainnya didalam perangkat lunak. Subjek juga dapat digunakan untuk memisahkan suatu sistem informasi, perangkat lunak, dan aktor internal dari lingkungannya.



Gambar 2.6 Subjek

e. *Include*

Merepresentasikan keadaan yang harus terpenuhi agar sebuah event dapat terjadi, dimana pada kondisi ini sebuah *use case* adalah bagian dari *use case* lainnya.



Gambar 2.7 *Include*

f. *Extend*

Keadaan yang hanya akan dijalankan dibawah kondisi tertentu.



Gambar 2.8 *Extend*

g. Generalisasi

Merepresentasikan *use case* khusus bagi sebuah *use case* umum.



Gambar 2.9 Generalisasi

*Use case narrative* adalah pendeskripsian fungsi dasar dari sebuah proses bisnis yang terdapat pada sebuah sistem. *Use case narrative* menyediakan sebuah cara agar dapat mendokumentasikan beberapa aspek dari *use case* individu. *Use case narrative* didasarkan pada identifikasi persyaratan, *use case diagram*, dan diagram aktivitas dari sebuah proses bisnis [7].

Ketika mengembangkan sebuah *use case narrative*, tim proyek haruslah bekerja sama dengan *user* untuk mendokumentasikan kebutuhan fungsional. Mengorganisir kebutuhan fungsional dan pengdokumentasiannya merupakan proses yang sederhana, tetapi memerlukan banyak latihan untuk memastikan bahwa narasinya lengkap untuk pemodelannya [7].

Terdapat 3 elemen dasar dari *use case narrative*, yaitu: [7]

a. *Overview Information*

*Overview Information* mengidentifikasi *use case* dan menyediakan latar belakang dasar dari informasi *use case* tersebut. Aktor utama seringkali merupakan pemicu dari sebuah *use case*. Tujuan utama dari *use case* adalah tercapainya tujuan dari aktor utama tersebut. Narasi singkat seringkali terdiri dari satu kalimat yang mendeskripsikan inti dari sebuah *use case*.

b. *Relationship*

Hubungan dari *use case* menjelaskan bagaimana sebuah *use case* berhubungan dengan *use case* dan *user* lainnya.

c. *Flow of Events*

Setiap tahapan dalam proses bisnis akan dinarasikan pada bagian ini. Ketiga kategori yang berbeda dapat didokumentasikan, yaitu: [7]

a) *Flow of Events* normal berisikan tahapan-tahapan yang biasanya dilakukan dalam sebuah *use case*. Tahapan tersebut disusun menurut urutan pengeksekusiannya.

b) *Subflows* digunakan ketika normal *flow of events* dipecah agar menjaganya tetap sederhana. *Subflows* bisa digantikan dengan *use case* yang baru jikalau *use case* tersebut dapat berdiri sendiri.

c) *Alternative or Exceptional flows* adalah suatu tahapan dimana suatu hal dapat terjadi tetapi bukan merupakan norma standar. Seperti *subflows*, tujuan utama untuk memisahkan hal ini adalah untuk menjaga *flow of events* tetap sederhana.

## 2.4.2 Prototyping

*Prototyping* adalah proses penciptaan sistem versi awal dengan cepat dari sistem informasi yang diajukan. Pembuatan *prototype* melibatkan proses analisa, desain, pemodelan, dan pengujian secara berulang-ulang. *Prototyping* adalah proses umum yang digunakan untuk desain mulai dari jaringan komputer hingga rumah [6].

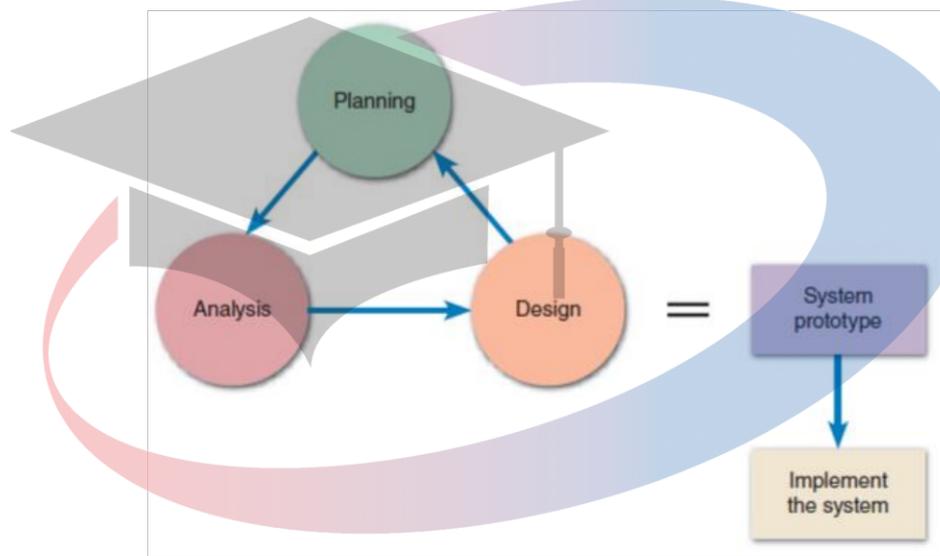
Masukan dari *user* adalah hal yang sangat penting pada setiap tahapan dari proses pengembangan sistem. *Prototyping* memungkinkan *user* untuk memeriksa model yang dapat merepresentasikan *output* sistem, *input*, tampilan antarmuka, dan proses-proses. *User* dapat menguji sistem secara aman dan bebas agar menyetujuinya

atau mengajukan perubahan. *Prototyping* juga dapat berkembang menjadi versi terakhir dari sebuah sistem informasi [6].

Sistem analis biasanya menggunakan 2 metode pembuatan *prototype*, yaitu: [6]

a. Pembuatan *Prototype* Sistem

*Prototyping* sistem menghasilkan model yang penuh dengan fitur yang berjalan dari sebuah sistem informasi. Sebuah *prototype* sistem yang memenuhi semua kebutuhan adalah sistem yang siap untuk diimplementasi.



Gambar 2.10 Sistem *Prototype*

b. Pembuatan *Prototype* Desain

Sistem analis juga menggunakan *prototype* untuk memverifikasi kebutuhan *user*, yang kemudian *prototype*-nya dibuang dan implementasi dilanjutkan. Pendekatan ini disebut dengan *prototype* desain, atau *prototype* buangan. Dalam kasus ini, tujuan pembuatan *prototype* lebih terbatas. Hasil akhir dari *prototype* desain adalah persetujuan model dari *user* yang mendokumentasikan fitur-fitur dari hasil akhir sistem. *Prototyping* desain memungkinkan untuk menangkap *input* dari *user* dan persetujuan seiring dengan pengembangan sistem dalam *framework*.

## 2.5 Situs *Web*

Situs *web* adalah layanan kumpulan halaman-halaman yang saling terhubung satu sama lainnya, dapat diakses secara publik dalam satu nama domain. Situs *web*

dapat diciptakan dan dipelihara oleh seorang individu, grup, bisnis ataupun organisasi yang bertujuan untuk berbagai tujuan. Semua situs *web* yang dapat diakses oleh publik didefinisikan sebagai *World Wide Web* (WWW).

Situs *web* merupakan kumpulan halaman *web* yang saling terhubung dan berkaitan. *Web* terdiri dari *page* atau halaman, dan kumpulan halaman yang dinamakan *homepage*. *Homepage* ini memiliki halaman-halaman yang terkait di posisi bawahnya. Pada umumnya, setiap halaman yang ada dibawah *homepage* disebut dengan *child page* yang berisi *hyperlink* ke halaman lain pada *web*.

Situs *Web* dapat dibagi menjadi 2, yaitu:

a. Situs *Web* statis

Situs *web* statis adalah situs *web* sederhana yang hanya terdiri dari HTML dan CSS, tanpa skrip. Hal yang interaktif dalam situs *web* statis hanyalah *hyperlink* yang berguna untuk navigasi situs *web* tersebut. Situs *web* statis dapat digunakan bila situs *web* yang ingin dibangun bersifat kecil.

b. Situs *Web* dinamis

Situs *Web* dinamis adalah situs *web* menggunakan skrip situs *web*, bukan hanya HTML dan CSS. Situs *web* dinamis berguna jikalau situs *web* tersebut bertujuan untuk digunakan, bukan hanya dibaca. Situs *web* ini menggunakan teknologi *database* dan server agar pembaharuan situs *web* dapat disimpan.

Sebelum merancang sebuah situs *web*, kita haruslah menentukan jenis situs *web* yang akan kita kembangkan. Kadangkala seseorang merancang sebuah situs *web* untuk tujuan spesifik. Apapun alasannya, jenis sebuah situs *web* baiknya dipikirkan terlebih dahulu. Beberapa jenis situs *web*, yaitu: [8]

a. *Personal*

Situs *web personal* menceritakan diri kita sendiri. Baik untuk mengekspresikan diri, menuangkan isi pikiran. Walaupun saat ini seseorang dapat menggunakan *social media* untuk hal ini, seseorang bisa saja menciptakan situs *web* untuk alasan pribadi.

b. *Resume*

Situs ini adalah situs *web personal* yang khusus digunakan untuk membangun karir seseorang. Daripada mencetak *resume* untuk diserahkan, mungkin lebih baik dengan cara mengirim *e-mail* berisi link ke situs *web* pencari kerja? Selain menjadi nilai tambah, pembuat situs juga dapat menambahkan detail yang lebih rinci.

c. *Topical*

Situs ini berfokus pada subjek tertentu yang diminati *user*, mulai dari musik, makanan, seni, buku, politik, dan lain-lain. Situs *web topical* dapat menjadi pilihan.

d. *Promotion*

Situs ini bertujuan untuk mempromosikan produk atau jasa agar dapat dikenal oleh orang-orang. Terkadang, situs *web* ini berkembang menjadi situs *web* bisnis kecil.

e. *Business*

Situs ini menunjukkan kesuksesan dari penggunaan situs *web* – menjual produk dan jasa.

Situs *web* haruslah dirancang sedemikian rupa agar dapat memenuhi tujuannya. Dalam perancangan sebuah situs *web* yang berkualitas, kiranya mengikuti prinsip-prinsip tertentu, yaitu:

a. Tujuan

Situs *web* yang berkualitas harus memenuhi kebutuhan *user*. Apakah *user* mencari informasi, edukasi, interaksi, bisnis? Setiap halaman situs *web* perlu memiliki tujuan yang jelas dan memenuhi kebutuhan spesifik *user* dengan seefektif mungkin.

b. Komunikasi

*User* cenderung ingin mendapatkan informasi secepat mungkin. Hal ini penting saat tahap desain sehingga informasi dapat dengan mudah didapat, dibaca dan dicerna.

c. Jenis

Secara umum, terdapat beberapa font yang lebih mudah dibaca, juga dengan mempertimbangkan ukuran font tulisan tersebut. Dalam pengembangan Situs *web*, penting untuk mengikuti konvensi-konvensi yang sudah ditetapkan agar situs *web* dapat diterima oleh khalayak ramai.

d. Warna

Warna yang tepat dapat meningkatkan kualitas penggunaan *user*. Warna yang saling mendukung dapat menciptakan keseimbangan yang membantu *user* untuk membaca tanpa menyebabkan mata letih ataupun lelah.

e. Gambar

Memilih gambar yang tepat untuk situs *web* dapat membantu situs *web* dalam memasarkan *brand* dan mencapai target calon *user*. Foto berkualitas tinggi dapat juga dibeli untuk memperindah tampilan situs *web*.

f. Navigasi

Navigasi sangatlah penting untuk kenyamanan *user*. Situs *web* dengan navigasi yang bagus dapat membantu *user* untuk berinteraksi dengan mudah untuk mendapatkan informasi. Salah satu strategi adalah ‘*Three-Click's rule*’, yaitu *user* haruslah dapat menemukan informasi yang dicari hanya dengan tiga klik.

g. *Layout* Berbasis *Grid*

Menempatkan konten-konten secara *random* dapat memperburuk pengalaman menggunakan *web* anda. *Layout* berbasis *grid* dapat memposisikan konten ke dalam bagian-bagian, kolom, kotak yang dapat memperindah tampilan situs *web*.

h. Waktu *Loading*

Situs *web* yang memakan waktu yang lama untuk dimuat akan mengalami kerugian. Situs *web* akan kehilangan *traffic* yang banyak karena *user* berasumsi situs *web* tidak berfungsi, dan akan berpindah ke situs *web* lain yang dapat dimuat dengan cepat.

i. *Mobile Friendly*

Situs *web* yang dapat diakses oleh telepon genggam / *smartphone* merupakan sebuah keharusan untuk meningkatkan *traffic*. *Traffic* dari *smartphone* tidak bisa disepelekan jumlahnya. Jikalau situs *web* yang diciptakan bisa diakses dengan *smartphone* dengan kualitas yang sama, tentu akan membantu situs *web* mendapatkan *user* lebih.

## 2.6 Teknologi Pengembangan *Web*

### 2.6.1 HTML dan CSS

Ketika kita melihat sebuah situs *web*, yang terjadi adalah *browser* menerima HTML dan CSS. Situs *web* juga menerima konten lainnya seperti gambar, audio, video, atau animasi. Hal lain yang dikirimkan adalah *Javascript* atau *Flash* ke *browser*. Situs *web* kecil biasanya hanya ditulis dengan HTML dan CSS. Situs *web* besar yang dinamis biasanya menggunakan teknologi yang lebih kompleks pada server *web* dan *database*. Sejak pertama kali *web* diciptakan, sudah terdapat beberapa versi HTML dan CSS. Saat ini versi yang digunakan adalah HTML5 dan CSS3 [9].

### 2.6.2 Javascript

*Javascript* pertama kali diperkenalkan pada tahun 1995 sebagai cara untuk menambahkan program ke dalam halaman *web* pada *browser Netscape Navigator*. Sejak saat itu, *javascript* telah diadopsi ke berbagai *browser* utama. *Javascript* memungkinkan kita berinteraksi dengan aplikasi *web* tanpa me-*refresh* halaman *web*.

Setelah diadopsi diluar *Netscape*, sebuah dokumen ditulis untuk mendeskripsikan bagaimana bahasa *Javascript* seharusnya beroperasi agar berbagai aplikasi yang mengklaim dapat mendukung *Javascript* memanglah sedang berbicara tentang *Javascript*. Dokumen ini disebut *ECMAScript*, setelah organisasi *Ecma International* mendeklarasikan standarnya. Dalam praktiknya, istilah *ECMAScript* dan *Javascript* dapat digunakan secara bergantian [10].

### 2.6.3 NodeJS

*NodeJS* pertama kali muncul ketika pengembang awal *javascript* mengembangkannya dari sebuah aplikasi yang hanya bisa dijalankan pada situs *web* agar dapat berjalan sebagai aplikasi tersendiri.

*NodeJS* adalah platform perangkat lunak pada tampilan *server* dan aplikasi jaringan. *NodeJS* dapat digunakan untuk mengembangkan aplikasi jaringan cepat dan dapat dikembangkan lebih lanjut.

### 2.6.4 MongoDB

*MongoDB* adalah sistem basis data berbasis dokument yang memiliki konsep NoSQL. Istilah ini dapat diartikan sebagai non-relasional karena berbeda dengan MySQL.

*MongoDB* pertama kali dikembangkan oleh MongoDB Inc pada Oktober 2007 sebagai produk dari PaaS (*Platform as a Service*) seperti *Windows Azure* dan *Google App Engine*. *MongoDB* dialihkan menjadi *open-source* pada tahun 2009.

## 2.7 E-Dagang

E-dagang adalah penggunaan internet, *World Wide Web*, dan aplikasi *mobile* dan *browser* yang terdapat pada perangkat selular untuk bertransaksi bisnis. Terdapat

beberapa argumen tentang pengertian e-dagang dan e-bisnis. E-dagang dan e-bisnis mengacu pada 2 hal yang berbeda.

E-bisnis memungkinkan transaksi dan proses dalam suatu perusahaan dilakukan secara digital, dimana tidak terjadi pertukaran nilai jual dari perdagangan dari bisnis lain atau konsumen. Aplikasi *e-bisnis* akan menjadi e-dagang ketika terjadi pertukaran nilai dari perdagangan.

E-dagang dikategorikan ke dalam pihak-pihak yang melakukan transaksi atau proses bisnis. Terdapat 5 kategori dasar dalam e-dagang, yaitu: [12]

a. *Business-to-consumer* (B2C)

Transaksi bisnis antar pelaku bisnis (produsen) kepada konsumen secara langsung dimana terjadi pertukaran produk atau jasa.

b. *Business-to-business* (B2B)

Transaksi bisnis antara pelaku bisnis yang menyediakan produk atau jasa kepada pelaku bisnis lainnya.

c. Proses Bisnis

Bisnis dan organisasi lainnya menjaga dan menggunakan informasi untuk mengidentifikasi dan mengevaluasi pelanggan, pemasok, dan pekerja.

d. *Consumer-to-consumer* (C2C)

Transaksi bisnis antara suatu individu dengan individu lainnya.

e. *Business-to-government* (B2G)

Pemasaran sektor publik yang mencakup pemasaran produk atau jasa untuk instansi pemerintah.

*Internet* telah mengubah cara masyarakat membeli, menjual, menyewa, dan mengorganisasi aktivitas bisnis dalam lebih banyak cara dengan cepat daripada teknologi lainnya dalam sejarah bisnis [12]. *Internet* pada awalnya hanya digunakan untuk riset akademis.

Kemunculan e-dagang muncul disaat kemunculan *WWW* pada tahun 1990an. Hal ini memungkinkan organisasi untuk memiliki eksistensi pada *internet*. Ketika *internet* dikomersialkan lebih lanjut, muncul lah istilah e-dagang. *Dot-coms*, *start-ups* merupakan produk awal dari e-dagang. Saat ini, semua organisasi pada negara berkembang sudah hadir dalam *web* [13]