

BAB 2

LANDASAN TEORI

Bab ini berisi mengenai teori-teori, konsep, pendapat para ahli yang mendukung teori-teori yang berhubungan dalam penulisan ini, literatur review, dan kerangka pikir.

2.1 Teori Umum

2.1.1 Definisi Data

Menurut (Coronel dan Morris 2017), data terdiri dari fakta mentah. Kata mentah disini mengindikasikan bahwa fakta belum diolah untuk kemudian didapat manfaatnya.

2.1.2 Definisi Informasi

Menurut (Coronel dan Morris 2017), informasi merupakan hasil dari pengolahan data mentah, sehingga bisa diketahui arti atau makna sebenarnya. Pengolahan data ini bisa berbentuk sangat sederhana seperti mengorganisasikan data untuk menangkap pola, dan bisa berbentuk kompleks seperti pembuatan *forecast* atau penggambaran simpulan menggunakan model *statistikal*. Untuk menangkap maksud sesungguhnya, informasi membutuhkan konteks. Misalnya rata rata temperatur menunjukkan 106 *derajat*. Hal ini tidak berarti banyak jika tidak diketahui konteksnya, seperti apakah *derajat Celcius* atau *Fahrenheit* ? apakah temperatur tersebut adalah suhu badan, suhu ruangan, atau temperatur udara diluar?

Informasi ini merupakan fondasi utama dari pengambilan keputusan. Misalnya data *summary* untuk fakultas mampu menyediakan informasi untuk badan akreditasi apakah diperlukan perbaruan akreditasi untuk universitas tersebut atau tidak.

2.1.3 Definisi Sistem Informasi

Menurut (Satzinger et al., 2012), sistem informasi merupakan sekumpulan komponen-komponen komputer yang saling berhubungan yang mengumpulkan, memproses, menyimpan, dan menyediakan sebagai keluaran informasi yang dibutuhkan untuk menyelesaikan tugas-tugas bisnis.

Berdasarkan pemaparan diatas dapat disimpulkan bahwa sistem informasi adalah pemanfaatan sekumpulan komponen-komponen perangkat keras dan perangkat (sumber daya komputer) untuk mengumpulkan, memproses, menyimpan, dan menyediakan informasi yang berguna.

2.1.4. Analisis Sistem

Analisis sistem terdiri dari kegiatan-kegiatan yang memungkinkan seseorang untuk memahami dan menentukan apa yang harus dicapai oleh sistem baru (Satzinger et al., 2012).

Kesimpulannya bahwa kegiatan analisis sistem ditujukan untuk menggambarkan apa yang harus sistem lakukan demi mencapai kepuasan suatu kebutuhan atau apa yang harus sistem lakukan untuk memecahkan suatu masalah.

2.1.5. Desain system

Desain sistem terdiri dari kegiatan-kegiatan yang memungkinkan seseorang untuk menguraikan secara rinci dalam memecahkan suatu kebutuhan system. (Satzinger et al., 2012). dengan kata lain bahwa desain sistem menggambarkan tentang bagaimana cara sistem bekerja agar dapat memberikan solusi sesuai dengan yang diharapkan.

2.1.6. Tahapan System Development Life Cycle (SDLC)

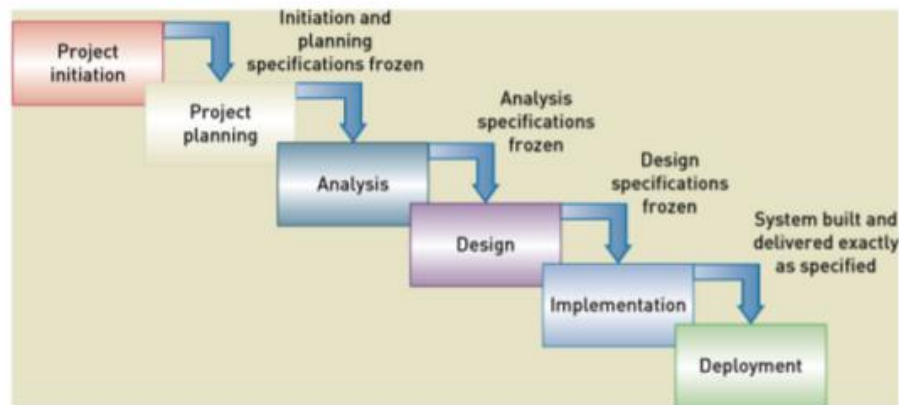
Menurut (Satzinger fifth., edition), *System Development Life Cycle (SDLC)* merupakan seluruh proses yang terdiri dari semua kegiatan yang diperlukan untuk membangun, meluncurkan, dan memelihara sistem informasi.

System Development Life Cycle (SDLC) terdiri atas 6 tahap utama, yaitu:

- 1) **Initiation**, mengidentifikasi permasalahan atau kebutuhan dan mendapatkan persetujuan untuk proses lebih lanjut
- 2) **Planning**, perencanaan dan pemantauan proyek. Pada tahap ini menentukan apa yang akan dilakukan, bagaimana caranya, dan siapa yang akan melakukan.
- 3) **Analysis**, mencari dan memahami detail permasalahan dan kebutuhan pengguna.
- 4) **Design**, perancangan komponen sistem yang mampu memecahkan masalah dan memenuhi kebutuhan pengguna.

- 5) **Implementation**, membangun sistem, melakukan testing dan mengintegrasikan komponen sistem.
- 6) **Deployment**, menyelesaikan pengujian sistem dan memberikan solusi.

Berdasarkan pemaparan diatas dapat disimpulkan bahwa *System Development Life Cycle* adalah tahapan-tahapan yang dibutuhkan dalam pengembangan sebuah sistem, meliputi: *Initiation, Planning, Analysis (Discover & understand), Design, Implementation (Build, testing, integration)* dan *Deployment & support*. Tahapan lainnya adalah *support* yang meliputi aktifitas *upgrade* dan *maintain* sistem setelah *deployment*.



Gambar 2.1 SDLC

(Sumber : Satzinger et al., 2012)

2.1.7 Analisis dan Desain Sistem Informasi

2.1.7.1 Analisis Sistem Informasi

Analysis Sistem Informasi mencari dan memahami detail permasalahan dan kebutuhan pengguna. Analisis sistem terdiri dari kegiatan-kegiatan yang memungkinkan seseorang untuk memahami dan menentukan apa yang harus dicapai oleh sistem baru. (Satzinger et al., 2012), Kegiatan - kegiatan pada tahap analisis sistem terdiri dari mengumpulkan informasi secara detail, mendefinisikan kebutuhan, memprioritaskan kebutuhan, membangun user interface dialog, dan mengevaluasi kebutuhan dengan pengguna.

2.1.7.2 Desain/Perancangan Sistem Informasi

Desain sistem terdiri dari kegiatan-kegiatan yang memungkinkan seseorang untuk menguraikan secara rinci dalam memecahkan suatu kebutuhan system. (Satzinger et al., 2012).

Dengan kata lain bahwa desain sistem menggambarkan tentang bagaimana cara sistem bekerja agar dapat memberikan solusi sesuai dengan yang diharapkan

2.1.8 Object-Oriented Analysis and Design (OOAD)

Menurut Satzinger, Jackson and Burd (2012, p.241), *object-oriented analysis (OOA)*, mendefinisikan semua jenis objek yang melakukan pekerjaan dalam suatu sistem dan menunjukkan interaksi dari pengguna yang dibutuhkan untuk menyelesaikan tugas.

Menurut Satzinger, Jakson and Burd (2012, p.241), *object-oriented design (OOD)*, mendefinisikan semua tipe objek dari objek yang diperlukan untuk dapat berkomunikasi dengan orang dan perangkat yang ada di dalam sistem, menggambarkan bagaimana objek dapat berinteraksi untuk menyelesaikan tugas, menyesuaikan dan menyempurnakan masing-masing jenis objek sehingga dapat di implementasikan dengan bahasa tertentu.

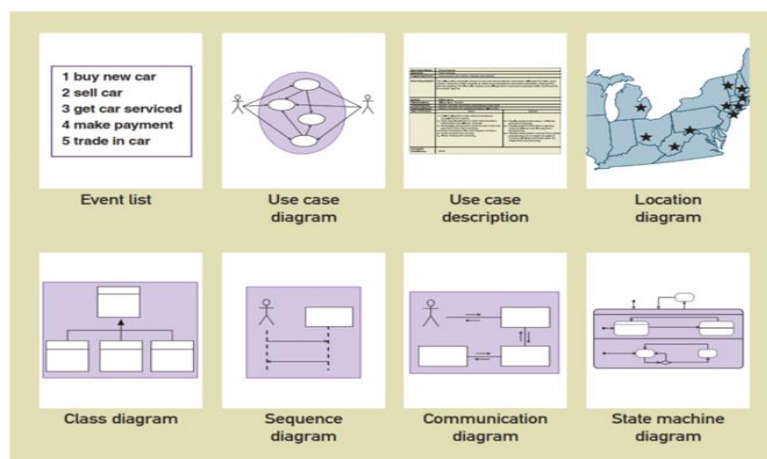
2.1.9 Unified Manipulation Language (UML)

UML merupakan metode permodelan yang digunakan untuk menentukan, memvisualisasikan, menganalisa, dan mendokumentasikan rancangan dari sebuah sistem (Satzinger, J. W. Jackson, R. B. and Burd, 2012). Banyak model grafis yang digunakan dalam pengembangan sistem digambarkan dengan notasi dari *UML*, seperti :

- a) *Use Case Diagram*
- b) *Activity Diagram*
- c) *Package Diagram*
- d) *State Machine Diagram*
- e) *Sequence Diagram*
- f) *Class Diagram*
- g) *Communication Diagram*
- h) *Composite Structure Diagram*
- i) *Obeject Diagram*

- j) *Timing Diagram*
- k) *Component Diagram*
- l) *Deployment Diagram*
- m) *Interaction Overview Diagram*

Dari sekian banyak jenis *Unified Modeling Language (UML)* diagram yang ada, jenis diagram yang digunakan dalam skripsi ini adalah *Use Case Diagram* beserta *Fully Use Case Descriptive*, *Activity Diagram*, *Class Diagram*, dan *Sequence Diagram*.



Gambar 2.2 Analisis dan desain model *unified modelling language (UML)*

(Sumber :Satzinger et al., 2012)

2.1.10 Use Case Diagram

Menurut (Satzinger et al., 2012), *use case* adalah suatu kegiatan yang dilakukan sistem, biasanya sebagai tanggapan atas permintaan oleh pengguna. Menurut (Munawar, 2018), *use case* adalah serangkaian *scenario* yang digabungkan bersama-sama oleh tujuan umum pengguna.

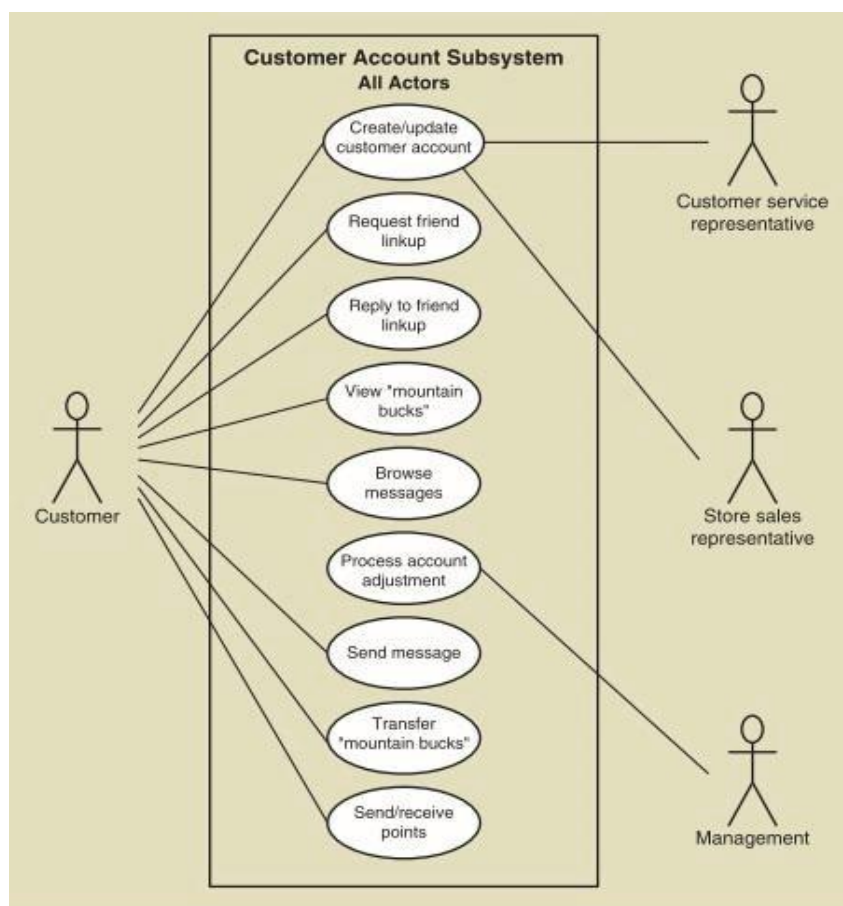
Berkenaan dengan *use case*, pengguna biasanya disebut dengan *actor*. *Actor* adalah sebuah peran yang bisa dimainkan oleh pengguna dalam interaksinya dengan sistem. Sedangkan *Scenario* adalah urutan langkah-langkah yang menerangkan antara pengguna dengan sistem.

Berikut adalah komponen – komponen dari *Use Case Diagram*:

- 1) *Actor* berbentuk orang. Seorang *actor* selalu di luar batas otomatisasi (*automation boundary*) sistem tetapi dapat menjadi bagian dari bagian manual dari sistem. Terkadang, *actor* untuk *use case* bukan orang, namun bisa berupa

sistem atau perangkat lain yang menerima layanan dari sistem. Gambar simple stick digunakan untuk merepresentasikan actor. Sosok tongkat diberi nama yang menjadi ciri khas peran yang dimainkan actor.

- 2) *Use case* itu sendiri diwakili oleh oval dengan nama use case di dalamnya. *Use case* menggambarkan apa yang dilakukan actor di dalam sistem.
- 3) *Connecting line* atau garis penghubung antara aktor dan use case menunjukkan bahwa aktor terlibat dengan use case tersebut.
- 4) *Automation boundary* (Batas otomatisasi), mendefinisikan batas antara bagian yang terkomputerisasi dari aplikasi dan orang yang mengoperasikan aplikasi, notasi ini ditampilkan dengan bentuk sebuah persegi panjang yang berisi use case.



Gambar 2.3 Contoh *Use Case Diagram*

(Sumber :Satzinger et al., 2012)

2.1.10.1 Use Case Description

Menurut (Satzinger et al., 2012), *use case description* merupakan sebuah model tekstual yang berisi daftar dan mendeskripsikan rincian proses untuk setiap *use case*.

Jadi setiap detail informasi tentang setiap use case di deskripsikan dengan *use case description*. Menurut (Munawar, 2018), *Use case* dibuat berdasarkan kebutuhan actor. *Use case* harus merupakan “apa” yang dikerjakan software aplikasi, bukan “bagaimana” software aplikasi mengerjakannya. Setiap *use case* harus diberi nama yang menyatakan apa hal yang dicapai dari hasil iteraksinya dengan actor

Use case name:	Create customer account.	
Scenario:	Create online customer account.	
Triggering event:	New customer wants to set up account online.	
Brief description:	Online customer creates customer account by entering basic information and then following up with one or more addresses and a credit or debit card.	
Actors:	Customer.	
Related use cases:	Might be invoked by the <i>Check out shopping cart</i> use case.	
Stakeholders:	Accounting, Marketing, Sales.	
Preconditions:	Customer Account subsystem must be available. Credit/debit authorization services must be available.	
Postconditions:	Customer must be created and saved. One or more Addresses must be created and saved. Credit/debit card information must be validated. Account must be created and saved. Address and Account must be associated with Customer.	
Flow of activities:	Actor	System
	1. Customer indicates desire to create customer account and enters basic customer information.	1.1 System creates a new customer. 1.2 System prompts for customer addresses.
	2. Customer enters one or more addresses.	2.1 System creates addresses. 2.2 System prompts for credit/debit card.
	3. Customer enters credit/debit card information.	3.1 System creates account. 3.2 System verifies authorization for credit/debit card. 3.3 System associates customer, address, and account. 3.4 System returns valid customer account details.
Exception conditions:	1.1 Basic customer data are incomplete. 2.1 The address isn't valid. 3.2 Credit/debit information isn't valid.	

Gambar 2.4 Contoh *Use Case Description* sisi Admin

(Sumber :Satzinger et al., 2012)

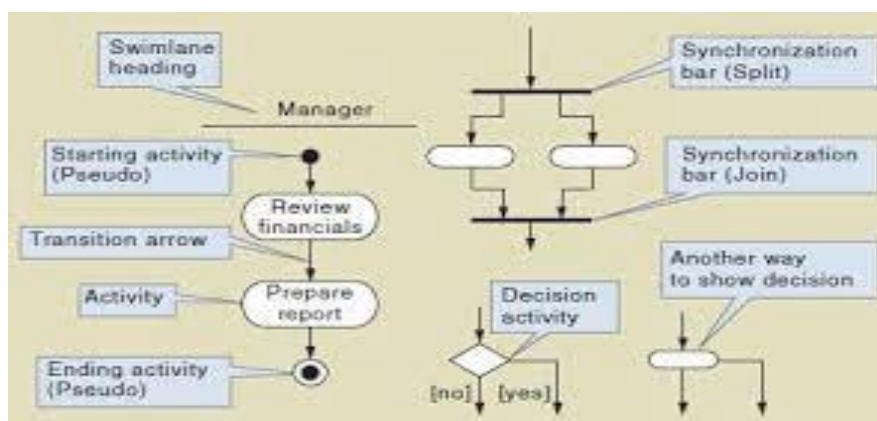
2.1.10.2 Activity Digram

Menurut (Satzinger et al., 2012), Suatu activity diagram merupakan gambaran berbagai pengguna (atau sistem) kegiatan, orang yang melakukan setiap kegiatan, dan aliran sekuensial dari kegiatan tersebut.

Activity diagram merupakan salah satu diagram yang dapat digunakan untuk mendeskripsikan sistem baru, menangani transaksi bisnis atau requirement pengguna.

Simbol - simbol yang digunakan Activity diagram diantaranya:

- 1) *Oval* Merepresentasikan aktivitas individu dalam alur kerja.
- 2) *Connecting arrow* (Bentuk Panah penghubung) Merepresentasikan urutan antara aktivitas.
- 3) *Black circle denote* (Bentuk Lingkaran hitam) Menunjukkan awal dan akhir alur kerja.
- 4) *Diamond* (Bentuk Berlian) merupakan, titik keputusan di mana aliran proses akan mengikuti satu jalur atau lainnya.
- 5) *Syschronization Bar* (Bentuk Garis tух tebal) Merupakan bilah sinkronisasi (pemisah), yang membagi jalan menjadi beberapa jalur konkuren atau menggabungkan kembali jalur konkuren.
- 6) *Swimlane heading* (Judul *swimlane*) Merepresentasikan agen yang melakukan kegiatan. Karena dalam alur kerja memiliki agen yang berbeda (mis., Orang) melakukan langkah - langkah berbeda dari proses alur kerja, symbol swimlane membagi aktivitas alur kerja menjadi kelompok yang menunjukkan agen mana yang melakukan aktivitas mana.



Gambar 2.5 Gambar simbol Activity Diagram

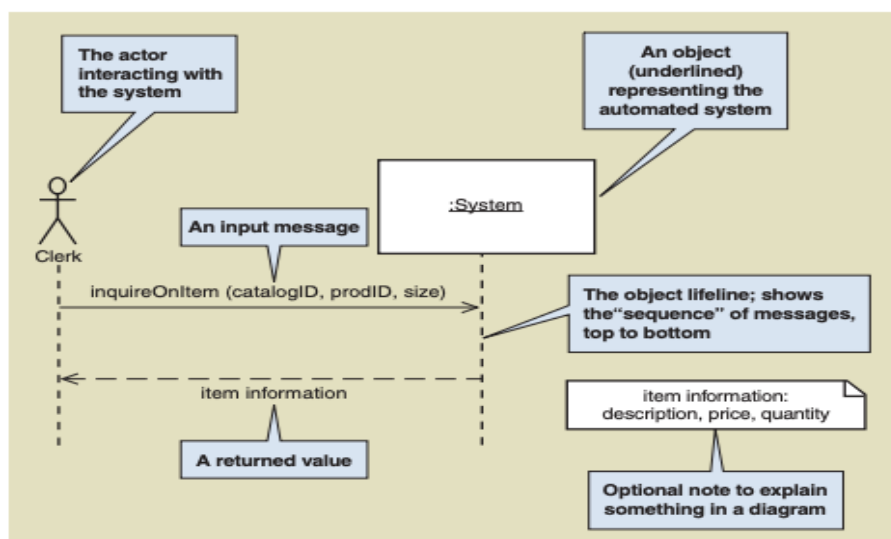
(Sumber :Satzinger et al., 2012)

2.1.10.3 System Sequence Diagram

Menurut (Satzinger et al., 2012), dalam pendekatan berorientasi objek, aliran informasi dicapai melalui mengirim pesan baik ke dan dari aktor atau bolak-balik antara internal objek. *Sistem Sequence Diagram (SSD)* digunakan untuk menggambarkan aliran informasi masuk dan keluar dari sistem otomatis.

Seperti halnya diagram use case, gambar stick mewakili aktor — seseorang (atau peran) yang berinteraksi dengan sistem. dalam *Use Case Diagram*, aktor “menggunakan” sistem, tetapi penekanan pada *SSD* adalah pada bagaimana aktor “berinteraksi” dengan sistem dengan memasukkan data input dan penerimaan data output. Kotak berlabel: Sistem adalah objek yang merepresentasikan keseluruhan sistem otomatis. Dalam *SSD* dan semua diagram interaksi lainnya, analisis menggunakan objek notasi alih-alih notasi kelas. dalam notasi objek, kotak mengacu pada objek individual, bukan kelas dari semua objek serupa. Notasi hanyalah sebuah persegi panjang dengan nama objek yang digarisbawahi. Titik dua sebelum digarisbawahi nama kelas adalah bagian notasi objek yang sering digunakan tetapi opsional. didalam diagram interaksi, pesan dikirim dan diterima oleh individu objek, bukan oleh kelas. dalam sebuah *SSD*, satu-satunya objek yang disertakan adalah yang mewakili seluruh sistem.

Jadi *Sequence diagram* digunakan untuk menggambarkan perilaku pada sebuah skenario. Diagram ini menunjukkan sejumlah contoh objek dan message (pesan) yang diletakkan diantara objek - objek ini didalam use case.



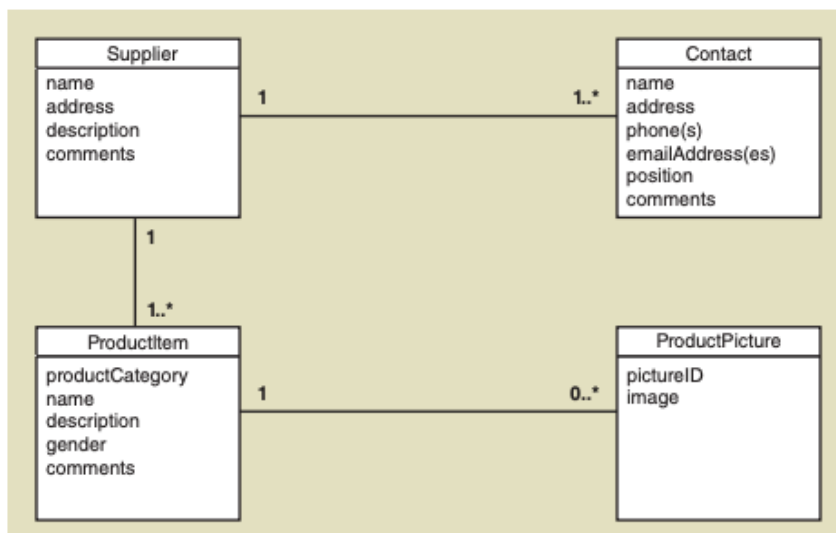
Gambar 2.6 Contoh System Sequence Diagram

(Sumber :Satzinger et al., 2012)

2.1.10.4 Domain Class Diagram

Domain class diagram merupakan diagram yang sering digunakan untuk memahami dan mendokumentasikan persyaratan informasi dari suatu sistem.

Domain class diagram dibagi dalam 2 pengertian yaitu *class diagram* dan *domain modeling*. *Class Diagram* merupakan dasar untuk tahapan desain basis data. Dimana ini merupakan inti dari *UML* untuk menggambarkan sebuah sistem. Sedangkan *domain modeling* merupakan tahapan identifikasi kebutuhan objek untuk memenuhi kandidat kelas pada *class diagram*.



Gambar 2.7 Contoh Domain Class Diagram

(Sumber :Satzinger et al., 2012)

2.1.11 User Interface

Menurut Satzinger, Jackson, dan Burd (Satzinger, J. W. Jackson, R. B. and Burd, 2012), *User Interface* adalah pandangan pengguna akhir terhadap sistem yang dikemukakan. Input dan output yang ada langsung melibatkan pengguna akhir. Desain dari user interface bisa bermacam-macam variasinya dan tergantung pada faktor-faktor seperti tujuan, karakteristik pengguna, karakteristik sistem, dan banyak lagi.

2.1.12 Proses System Development Life Cycle (SDLC)

Menurut Satzinger (2012), *System Development Life Cycle* (SDLC) adalah kerangka kerja yang mengidentifikasi semua kegiatan yang diperlukan untuk meneliti, membangun, menyebarkan, dan memelihara sistem informasi (Satzinger, J. W. Jackson, R. B. and Burd, 2012). Ada beberapa pendekatan didalam *SDLC* termasuk

variasi khusus untuk jenis proyek tertentu, dan setiap *SDLC* terdapat proses utama didalamnya, diantaranya:

- 1) *Identify the Problem and Obtain Approval*
Mengidentifikasi masalah atau kebutuhan dan mendapatkan persetujuan untuk diproses didalam proyek.
- 2) *Plan and Monitor the Project*
Merencanakan dan memantau proyek , termasuk apa yang harus dilakukan, bagaimana melakukannya, dan siapa yang melakukannya.
- 3) *Discover and Understand Details*
Menemukan dan memahami secara detail dari masalah atau kebutuhan , dan apa saja yang dibutuhkan .
- 4) *Design System Component*
Desain komponen sistem merupakan pemecahan masalah atau pemenuhan kebutuhan serta bagaimana cara kerja sistem .
- 5) *Build, Test and Integrate System Component*
Membangun, menguji, dan mengintegrasikan komponen sistem dari banyak bahasa pemrograman serta integrasi komponen.
- 6) *Complete System test and deploy solution*
Menyelesaikan pengujian sistem dan kemudian menggunakan solusi yang telah dibangun.

2.1.13 Hypertext Markup Language (HTML) dan Extensible Hypertext Markup Language (XHTML)

HTML (Hypertext Markup Language) dan *XHTML (Extensible Hypertext Markup Language)* merupakan bahasa sederhana yang diperlukan untuk menjelaskan struktur halaman berbagai web. Pada halaman web *HTML* dan *XHTML* bertugas menjelaskan struktur kata - kata diantaranya heading, awal dan akhir paragraphs, dan bullets points. *HTML & XHTML* juga menentukan tautan antara halaman web yang berbeda dan di mana gambar akan muncul. (Duckett, J. 2010).

Ada beberapa versi *HTML* sejak Web dimulai, dan pengembangan bahasa diawasi oleh organisasi bernama *W3C (World Wide Web Consortium)*. Versi terakhir *HTML* adalah *HTML 4.01* pada Desember 1999. Pada Januari 2000, beberapa aturan

yang lebih ketat ditambahkan ke *HTML 4.01*, menciptakan apa yang dikenal sebagai XHTML (Extensible Hypertext Markup Language). (Duckett, J. 2010).

Menurut (Duckett, J. 2010) Struktur dokumen Web menggunakan XHTML terdiri dari:

- 1) *Elemen <head>*: Sering disebut sebagai kepala halaman, ini berisi informasi tentang halaman (ini bukan konten utama halaman). Misalnya, mungkin berisi judul dan deskripsi halaman, atau instruksi di mana browser dapat menemukan aturan CSS yang menjelaskan bagaimana tampilan dokumen. Ini terdiri dari tag pembuka *<head>*, tag *</head>* penutup, dan semuanya di antaranya.
- 2) *Elemen <body>*: Sering disebut sebagai badan laman, ini berisi informasi yang sebenarnya Anda lihat di jendela browser utama. Ini terdiri dari tag pembuka *<body>*, tag *</body>* penutup, dan semua yang ada di antaranya.

2.1.14 Cascading Style Sheets (CSS)

Cascading Style Sheets (CSS) untuk mengendalikan gaya halaman anda, termasuk warna dan ukuran font, lebar dan warna garis, dan jumlah ruang antara item pada halaman. CSS bekerja dengan memungkinkan anda menentukan aturan yang menyatakan bagaimana konten elemen dalam dokumen anda akan muncul. Misalnya, Anda dapat menentukan bahwa latar belakang halaman adalah warna krem, isi semua *elemen <p>* harus ditampilkan dalam warna abu-abu menggunakan jenis huruf Arial, dan bahwa semua elemen *<h1>* harus berwarna merah menggunakan Times New. Jenis huruf Romawi. (Duckett, J. 2010).

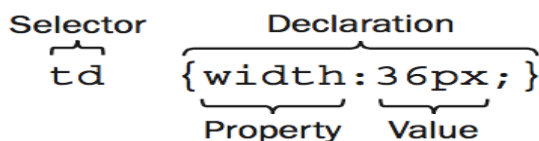
(Duckett, J. 2010) menyebutkan bahwa aturan penulisan CSS terdiri dari 2 bagian diantaranya:

- 1) *The Selector*, menunjukkan elemen tag HTML atau elemen – elemen deklarasi diterapkan (jika diterapkan untuk lebih dari satu elemen, Anda dapat memiliki daftar beberapa elemen yang dipisahkan koma)
- 2) *The Declaration*, Sekumpulan yang menunjukkan selector mana yang ingin diberi stye.

Menurut (Duckett, J. 2010) Declaration juga dibagi menjadi dua bagian, dipisahkan oleh tanda titik dua:

- 1) *Property*, yang merupakan properti dari elemen yang dipilih yang ingin Anda pengaruhi, dalam hal ini properti *width*.

- 2) *Value*, yang merupakan spesifikasi untuk *property*. dalam hal ini sel tabel harus selebar 36 piksel.



Gambar 2.8 Contoh CSS
Sumber (Duckett, J. 2010)

2.1.15 Java Script

Merupakan bahasa pemrograman yang dapat menambahkan interaktivitas ke halaman web. *JavaScript* itu sendiri adalah topik yang sangat besar, sehingga tidak tercakup dalam kedalaman yang sama dengan *XHTML* dan *CSS*. (Duckett, J. 2010).

(Duckett, J. 2010) menyebutkan bahwa *JavaScript* memberi *developer web* sebuah bahasa pemrograman untuk digunakan di halaman *web* yang memungkinkan mereka untuk melakukan tugas-tugas seperti berikut ini:

- 1) Baca elemen dari dokumen dan tulis elemen dan teks baru ke dalam dokumen.
- 2) Memanipulasi atau memindahkan teks.
- 3) Melakukan perhitungan matematis pada data.
- 4) Bereaksi terhadap beberapa hal, seperti pengguna mengklik tombol.
- 5) Mengambil tanggal dan waktu terkini dari komputer pengguna atau terakhir kali dokumen diubah.
- 6) Tentukan ukuran layar pengguna, versi *browser*, atau resolusi layar.
- 7) Melakukan tindakan berdasarkan kondisi seperti memperingatkan pengguna jika mereka memasukkan informasi yang salah ke dalam formulir.

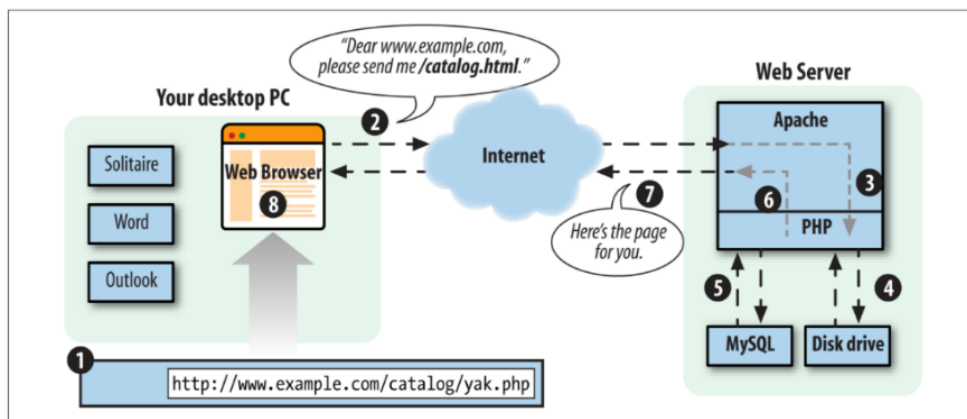
JavaScript dapat disematkan di halaman atau ditempatkan di file skrip eksternal (lebih seperti *CSS*). Tetapi agar dapat bekerja di browser, browser harus mengaktifkan *JavaScript*. (Browser utama memungkinkan pengguna untuk menonaktifkan *JavaScript*, meskipun sangat sedikit yang melakukannya.) (Duckett, J. 2010).

2.1.16 Hypertext Preprocessor (PHP)

PHP adalah bahasa pemrograman yang sebagian besar digunakan untuk membangun situs web. dari pada berjalan di komputer desktop untuk penggunaan perorangan, biasanya dijalankan di server web dan diakses oleh banyak orang yang menggunakan browser web di komputernya. (Sklar, 2016)

Program yang anda tulis dalam *PHP* (bahasa pemrograman) menyebabkan mesin *PHP* melakukan hal-hal seperti berbicara ke database, menghasilkan halaman web yang dipersonalisasi, atau menampilkan gambar. (Sklar, 2016)

Menurut (Sklar, 2016) *PHP* disebut bahasa pemrograman *Server - side* karena bekerja di *server web*. Berikut ilustrasinya:



Gambar 2.9 Komunikasi Klient & Server dengan PHP

The PHP engine runs the commands between <?php (the PHP start tag) and ?> (the PHP end tag). PHP pages typically live in files whose names end in .php. Example 1-1 shows a page with one PHP command. (Sklar, 2016). Menurut (Sklar, 2016) bahwa mesin *PHP* berjalan diantara perintah *tags <?php* (sebagai awalan *PHP*) dan *?>* (sebagai akhiran *tags PHP*).

2.1.17 Basis Data

2.1.17.1 Data Base

Menurut (Kusuma & Publication, 2011), *database* bila diterjemahkan ke dalam bahasa Indonesia secara harafiah berarti pangkalan data. Sama seperti pangkalan militer yang memuat banyak personil militer dan alat-alat militer maka pangkalan data

atau selanjutnya anda sebut *database* saja, memuat banyak data dan *tool-tool* yang terkait dengan operasi data.

2.1.17.2 Mysql

MySql merupakan sebuah perangkat lunak sistem manajemen basis data *SQL* (bahasa Inggris: database management system) atau *DBMS* yang multialur, multipengguna, dengan sekitar 6 juta instalasi di seluruh dunia. *MySQL AB* membuat *MySQL* tersedia sebagai perangkat lunak gratis di bawah lisensi *GNU General Public License (GPL)*, tetapi mereka juga menjual di bawah lisensi komersial untuk kasus-kasus di mana penggunaannya tidak cocok dengan penggunaan *GPL*. Kelebihan *MySql* sebagai berikut:

- 1) Merupakan salah satu software yang *portable*
- 2) *MySQL* merupakan salah satu *DBMS* yang *opensource*
- 3) *Multi-User*
- 4) Memiliki tipe data yang bervariasi
- 5) Memiliki fitur keamanan yang baik
- 6) Administrative tools yang lengkap
- 7) Struktur tabel yang lebih fleksibel
- 8) Dapat diintegrasikan dengan berbagai bahasa pemrograman
- 9) Tidak membutuhkan spesifikasi hardware yang tinggi
- 10) *RAM* Kecil dapat menggunakannya

2.1.17.3 Perintah SQL

Perintah *SQL* merupakan perintah untuk mengolah, menampilkan dan memanipulasi data yang tersimpan dalam sebuah *RDBMS*. Berikut adalah perintah *SQL* yang dibagi berdasarkan fungsinya, yaitu:

1) *Data Definition Language*

Merupakan perintah untuk membuat atau mengubah struktur objek database, yaitu:

- a. *CREATE* untuk membuat struktur objek *database*.
- b. *ALTER* untuk mengubah struktur objek *database*.
- c. *DROP* untuk menghapus objek *database*.
- d. *TRUNCATE* untuk menghapus data dan *reclaim space*.
- e. *RENAME* untuk mengganti nama tabel.

2) *Data Control Language*

Merupakan perintah untuk mengatur hak akses *user* dalam *database*, yaitu:

- a. *GRANT* untuk memberikan izin tertentu ke *user*
- b. *REVOKE* untuk menarik izin tertentu ke *user*

3) *Data Manipulation Language*

Merupakan perintah untuk melakukan manipulasi data seperti menambah, menghapus, dan mengubah data dalam tabel, yaitu:

- a. *INSERT* untuk menambah baris ke table.
- b. *DELETE* untuk menghapus baris dari table.
- c. *UPDATE* untuk mengubah data di table.
- d. *MERGE* untuk menggabungkan data dari dua table.

4) Perintah *Query*

Merupakan perintah untuk menampilkan data dari tabel, yaitu:

- a. *SELECT* untuk menampilkan data dari table

5) *Transaction Control Language*

Merupakan perintah untuk melakukan penyimpanan atau perubahan data secara permanen dan melakukan pembatalan terhadap perubahan data yang disimpan, yaitu:

- a. *COMMIT* untuk melakukan penyimpanan atau perubahan data secara permanen.
- b. *ROLLBACK* untuk membatalkan perubahan data yang disimpan.
- c. *SAVEPOINT* untuk menentukan tahapan-tahapan dalam melakukan perubahan data sehingga jika ada kesalahan data bisa dikembalikan ke tahapan tertentu.

2.1.17.4 Aturan Penulisan Perintah SQL

Terdapat beberapa aturan yang harus diperhatikan pada saat menulis perintah *SQL*, yaitu sebagai berikut:

- a) Perintah *SQL* tidak *case sensitive*, artinya tidak dibedakan antara penulisan huruf kecil dan huruf besar.
- b) Perintah *SQL* dapat terdiri dari lebih dari satu baris.
- c) Kata kunci pada perintah *SQL* tidak bisa disingkat atau dipisah di baris yang berbeda.
- d) Klausa biasanya ditempatkan pada baris yang berbeda.

- e) Gunakan indentasi untuk setiap klausa untuk mempermudah pembacaan perintah.
- f) Setiap perintah *SQL* diakhiri dengan tanda titik koma “;”.
- g) Penulisan komentar pada perintah *SQL* menggunakan tanda minus dua kali “—
“ untuk komentar satu baris, sedangkan komentar lebih dari satu baris harus diawali dengan tanda /* dan diakhiri tanda */.

2.1.18 Pengujian

Merupakan tahapan *SDLC* yang dilakukan setelah tahapan *development*. Hal ini bertujuan untuk melakukan pengujian terkait sistem yang sudah dibuat sebelumnya dan memastikan tidak ada error dan memenuhi permintaan pengguna. Menurut Pressman (2010), Ketika kode sebuah aplikasi telah dibuat, aplikasi harus diuji untuk mencari serta mengatasi permasalahan sebanyak-banyaknya sebelum aplikasi didistribusikan kepada pengguna. Untuk itu dibutuhkan sebuah teknik pengujian perangkat lunak untuk memberikan panduan yang sistematis dalam menguji logika, tampilan, input dan output untuk menemukan kesalahan dalam fungsi, tingkah laku, serta performa aplikasi.

2.1.18.1 White Box Testing

White Box Testing menurut Pressman (2010), adalah sebuah metode pengujian dimana alur pengujainnya (*test-case*) mengikuti pada program (*control structure*) dan diuji oleh pengembang sendiri. dengan menggunakan uji *white-box* kita dapat memperoleh sebuah *test-case* dengan setiap jalur pada setiap modul, setiap putusan logika, perulangan, dan keabsahan data yang telah teruji.

2.1.18.2 Black Box Testing

Black Box Testing menurut Pressman (2010), adalah pengujian yang fokus pada kebutuhan fungsional pada sebuah perangkat lunak. Dengan melakukan pengujian *blackbox* dapat diperoleh kumpulan input yang akan menguji seluruh kebutuhan fungsional sebuah aplikasi. Pengujian *black-box* bukan alternatif untuk pengujian *white-box*, namun bersifat sebagai pendekatan lain yang digunakan untuk melihat permasalahan yang berbeda dengan metode pengujian *whitebox*. Jadi ketika pembuatan sistem telah selesai, maka akan dilakukan pengujian terkait setiap fungsi

yang ada pada sistem, apakah fungsi yang dibuat sesuai dengan validitas yang ditentukan atau terdapat *error*, sehingga validitas sistem tidak dijalankan

2.2 Teori Khusus

2.2.1 Ecommerce

E-commerce adalah penggunaan *Internet*, *Web*, dan aplikasi untuk transaksi bisnis. lebih formal lagi, transaksi komersial yang dimungkinkan secara digital di antara dan diantara organisasi dan individu. *E-business* adalah digital yang memungkinkan transaksi dan proses dalam suatu perusahaan, mencakup system informasi di bawah kontrol perusahaan. Internet adalah jaringan *worldwide* dari jaringan computer, dan *Web* adalah satu dari layanan paling populer Internet, menyediakan akses ke milyaran laman *Web*. *App* (singkatan dari Aplikasi) adalah aplikasi perangkat lunak. Istilah tersebut secara khusus digunakan ketika mengacu pada aplikasi mobile, meskipun itu juga seringkali digunakan untuk mengacu pada aplikasi computer desktop.

2.2.2 Perbedaan E-commerce dan E-business

Adapun Perbedaan antara *e-commerce* dan *e-business* Beberapa argumentasi menjelaskan bahwa *e-commerce* mencakup seluruh dunia dari aktivitas organisasi yang didasarkan secara elektronik yang mendukung pertukaran pasar perusahaan – mencakup infrastruktur system informasi seluruh perusahaan (Rayport dan Jaworski, 2003). Penjelasan lainnya, pada acara yang lain, bahwa *e-business* mencakup seluruh dunia internal dan eksternal aktivitas didasarkan secara elektronik, mencakup *e-commerce* (Kalakota and Robinson, 2003).

E-Commerce mencakup secara digital transaksi komersial yang dimungkinkan di antara dan antara organisasi dan individu. Transaksi yang dimungkinkan secara digital mencakup seluruh yang dimediasi dengan teknologi digital, berarti, untuk sebagian besar bagian, transaksi yang terjadi melalui *Internet*, *Web*, dan/atau mobile apps. Transaksi komersial mencakup pertukaran nilai (misalnya uang) antara organisasi dan individu yang membatasi dalam pengembalian produk atau layanan.

E-Business mengacu terutama pada digital yang memungkinkan transaksi dan proses dalam perusahaan, mencakup system informasi di bawah control perusahaan. Untuk bagian paling umum, *e-business* tidak mencakup transaksi komersial antar organisasi yang membatasi dimana nilai ditukar.

Identifikasi dan Uraian Fitur khusus dari Teknologi *e-commerce* dan mengemukakan Bisnis signifikannya Terdapat delapan fitur teknologi *e-commerce* yang unit terhadap media ini:

- 1) *Ubiquity* : tersedia dimana saja, pada seluruh waktu, membuatnya memungkinkan untuk membeli dari desktop anda, di rumah, di tempat kerja, atau mungkin dari kendaraan anda.
- 2) *Global Reach* : memungkinkan transaksi komersial untuk melintasi batasan budaya dan nasional jauh lebih secara konvensional dan biaya-yang murah daripada yang benar dalam perdagangan tradisional.
- 3) *Universal Standard* : di-share dengan seluruh bangsa di seluruh dunia, sebaliknya untuk sebagian besar teknologi perdagangan tradisional, yang membedakan dari satu bangsa dengan selanjutnya.
- 4) *Richness* : memungkinkan suatu online merchant mengirimkan pesan pemasaran dalam suatu cara yang tidak memungkinkan dengan teknologi perdagangan tradisional.
- 5) *Interactivity* : memungkinkan untuk komunikasi dua-arah di antara merchant dan consumer dan memungkinkan merchant untuk menghubungkan consumer dalam cara serupa dengan pengalaman tatap-muka, tetapi pada suatu skala global, yang lebih massive.
- 6) *Information density* : adalah jumlah total dan kualitas informasi yang tersedia untuk seluruh partisipan pasar. Internet mengurangi kumpulan informasi, penyimpanan, pemrosesan, dan biaya komunikasi sementara meningkatkan nilai mata uang, keakuratan, dan timeline dari informasi.
- 7) *Personalization and Customization* : peningkatan dalam densitas informasi memungkinkan merchant menargetkan pesan pemasarannya terhadap individu khusus dan hasil dalam suatu tingkat personalisasi dan kustomisasi yang tidak dapat dipikirkan dengan teknologi perdagangan yang ada sebelumnya.
- 8) *Social Technology* : menyediakan suatu modil banyak-ke-banyak pada komunikasi massa. Jutaan pengguna dapat menghasilkan konten yang dipakai oleh jutaan pengguna lainnya. Hasilnya adalah formasi jaringan social pada skala lebar dan agregasi dari audiens yang besar pada platform jaringan social.

2.2.3 Jenis - Jenis Ecommerce

Terdapat lima jenis Utama e-commerce :

- 1) *B2C e-commerce* : mencakup bisnis penjualan ke pelanggan dan merupakan jenis ecommer yang sebagian besar pelanggan yang mungkin menghadapi.
- 2) *B2B E-commerce*: mencakup bisnis penjualan ke bisnis yang lain dan merupakan bentuk terbesar dari e-commerce.
- 3) *C2C e-commerce* : adalah suatu cara untuk pelanggan menjual antara satu dengan yang lain. dalam *C2C e-commerce*, pelanggan menyiapkan produk untuk pasar, menempatkan produk untuk lelang atau sale, dan berada pada pembuat pasar untuk menyediakan katalog, mesin pencari, dan kemampuan membuka transaksi sehingga produk dapat dengan mudah ditayangkan, diperoleh, dan dibayar.
- 4) *Social e-commerce* : adalah e-commerce yang dimungkinkan dengan social netwrk dan hubungan online social.
- 5) *M-Commerce* : mencakup penggunaan peralatan digital tanpa kabel untuk memungkinkan transaksi online.
- 6) *Local e-commerce* : adalah bentuk dari e-commerce yang difokuskan pada pertautan consumer didasarkan pada lokasi geografinya saat ini.

2.2.4 History E-commerce

Histori singkat memahami evolusi *e-commerce* dari tahun awal sampai saat ini *e-commerce* telah melalui tiga tahap : *Inovasi, Konsolidasi, dan Reinvention*. Tahun-tahun awal dari e-commerce adalah suatu periode pertumbuhan eksplosif, dimulai tahun 1995 dengan penggunaan secara luas pertama pada *Web* untuk mengiklankan produk dan berakhir tahun 2000 dengan kolapsnya nilai pasar saham untuk perusahaan *dot-com*.

- a) Tahun-tahun awal dari *e-commerce* adalah suatu keberhasilan teknologi, dengan infrastruktur digital yang dibuat selama periode cukup solid untuk mempertahankan pertumbuhan yang signifikan dalam *e-commerce* selama dekade selanjutnya, dan suatu sukses bisnis yang dicampur, dengan pertumbuhan keuntungan signifikan dan kegunaan customer, tetapi margin *profit* yang rendah.
- b) *E-commerce* selama tahun-tahun awal tidak memenuhi visi eknomois dari perdagangan gesekan-bebas yang tepat, atau memenuhi visi kewirausahaan

dan modal perusahaan untuk manfaat berpindah-pertama, akuisisi kustomer rendah dan biaya mempertahankan, serta biaya rendah dari bisnis yang dilakukan.

- c) *E-commerce* memasuki periode konsolidasi mulai tahun 2001 dan berakhir tahun 2006.
- d) *E-commerce* dimasukkan suatu periode reinvention dalam tahun 2007 dengan menggabungkan platform digital mobile, social network, dan aplikasi Web 2.0 yang menarik audiens besar dalam rentang waktu yang sangat pendek.

2.2.5 Identifikasi E-commerce

Mengidentifikasi faktor - faktor yang akan menentukan masa depan dari *e-commerce*, adapun Faktor-faktor yang akan menentukan masa depan *e-commerce* mencakup berikut ini :

- 1) Teknologi *e-commerce* (*Internet, Web, dan platform mobile*) akan melanjutkan mendorong melalui seluruh aktivitas komersial, dengan seluruh keuntungan dari e-commerce dan jumlah produk dan layanan yang menjual seluruh yang naik.
- 2) Perusahaan *Fortune 500* yang berpengalaman dan yang diberikan-dengan baik tradisional akan berlanjut memainkan suatu peran dominan.
- 3) Jumlah perusahaan online asal yang berhasil akan berlanjut menurun, dan sebagian besar perusahaan *e-commerce* yang berhali akan mengadopsi suatu strategi yang diintegrasikan, multi-channel brick-and-click.
- 4) Peraturan di Internet dan *e-commerce* oleh pemerintah akan tumbuh keduanya di Amerika dan dunia.

2.2.6 Mengorganisasi E-commerce

E-commerce mencakup tiga tema *board* yang saling berkaitan:

- 1) Teknologi : Untuk memahami *e-commerce*, anda perlu suatu pemahaman dasar dari teknologi informasi padanya itu dibangun, mencakup Internet, *Web* dan platform mobile, dan induk dari teknologi pelengkap – *cloud computing*, computer pribadi, telepon pintar, computer tablet, jaringan *LAN*, komputasi *Client/server*, komunikasi packet-switched, protocol seperti *TCP/IP*, *web server*, *HTML*, dan database relasional dan non-relasional, di antaranya.

- 2) *Business* :Sementara teknologi menyediakan infrastruktur, itu merupakan aplikasi bisnis – potensial untuk mengembalikan investasi yang luar biasa – yang membua menarik dan kegembiraan di *e-commerce*. Selanjutnya, anda juga perlu memahami beberapa kunci konsep bisnis seperti pasar elektronik, barang informasi, model bisnis, perusahaan dan rantai nilai industri, struktur industry, dan perilaku consumer dalam pasar digital.
- 3) *Society* : Memahami tekanan bahwa *e-commerce* global menempatkan pada masyarakat kontemporer yang kritis untuk menjadi berhasil dalam pasar e-commerce. Permasalahan masyarakat Utama adalah kekayaan intelektual, privacy individu dan kebijakan public.

2.2.7 Pendekatan E-commerce

Mengidentifikasi disiplin akademik utama yang berkontribusi terhadap *E-commerce*. Terdapat dua pendekatan utama terhadap *e-commerce*: teknik dan perilaku. Masing-masing dari pendekatan tersebut ditunjukkan dengan beberapa disiplin akademik.

1) Pada sisi teknis:

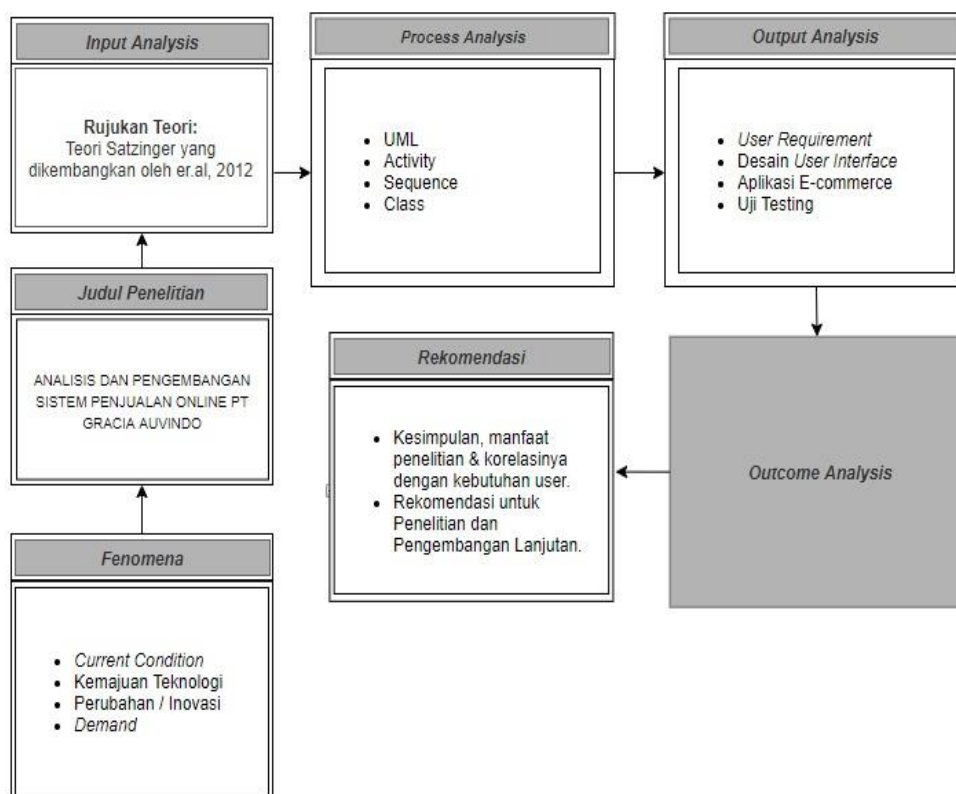
- Ilmuwan computer tertarik dalam e-commerce sebagai suatu aplikasi teknologi Internet.
- Ilmuwan manajemen terutama tertarik dalam pembangunan model matematika pada proses bisnis dan mengoptimisasinya untuk mempelajari bagaimana bisnis dapat mengeksploitasi Internet untuk menyempurnakan operasi bisnisnya.
- Profesional Sistem Informasi tertarik dalam e-commerce karena implikasi untuk perusahaan dan rantai nilai industry, struktur industry, dan strategi perusahaan.
- Orang-orang ekonomis telah difokuskan pada perilaku consumer online dan pada figure dari pasar elektronik digital.

2) Pada sisi perilaku:

- Sosiolog memfokuskan pada studi pada penggunaan Internet, peran ketidaksamaan social dalam menentukan manfaat Internet, dan penggunaan Web sebagai piranti komunikasi kelompok dan personal.

- Sarjana keuangan dan akuntansi telah memfokuskan pada valuasi perusahaan ecommerce dan praktek akuntansi.
- Sarjana manajemen memfokuskan pada perilaku kewirausahaan dan tantangan yang dihadapi oleh perusahaan mudah yang diperlukan untuk mengembangkan struktur organisasi dalam rentang waktu pendek. –
- Sarjana marketing telah memfokuskan pada respon consumer terhadap pemasaran online dan pengiklanan, dan kemampuan perusahaan terhadap merek, segmen pasar, audiens sasaran, dan posisi produk untuk mencapai hasil yang lebih tinggi pada investasi.

2.2.8 Kerangka Pemikiran Penelitian



Gambar 2.10 Kerangka Pemikiran Penelitian

Kesimpulan & Rekomendasi

- Kesimpulan, manfaat penelitian & korelasinya dengan kebutuhan user.
- Rekomendasi u/Penelitian dan Pengembangan Lanjutan.

