

## BAB II

### LANDASAN TEORI

#### 2.1 Website

*Website* atau dalam bahasa Indonesia disebut sebagai situs *web* merupakan kumpulan berbagai halaman *web* yang ditulis dengan bahasa HTML yang kemudian bisa dilihat menggunakan *software* yang disebut *web browser* (Zaki, 1999: 127). Halaman *web* bisa berisi file seperti gambar, video, dan sebagainya. Agar dapat diakses, halaman *web* harus diletakkan di *server web* untuk kemudian bisa diakses melalui peranti seperti internet, jaringan, dan sebagainya.

*Website* yang akan dibuat adalah sebuah *website* toko *online* dimana kebutuhan *website* toko *online* tidaklah sama dengan kebutuhan *website* - *website* lain seperti *website* profil perusahaan, *website* akademik, *website* jejaring sosial, menurut Balles, Louise (2012) untuk memenuhi kebutuhan *website* toko *online* diperlukan beberapa hal seperti:

##### 2.1.1 Web Host

Dalam rangka untuk mempublikasikan toko *online* di *internet*, harus dilakukan pendaftaran *account* sebuah toko *online* dengan penyedia *web-host*, yang juga akan menyediakan nama domain untuk sebuah toko *online*. Ketika memilih *webhost* mempertimbangkan biaya bulanan, jumlah kapasitas penyimpanan dan *bandwidth* (ukuran *download* dari pengunjung) disertakan dengan layanan.

##### 2.1.2 Keranjang Belanja

Toko *online* perlu keranjang belanja sebagai perangkat lunak untuk toko *online*.

Sistem keranjang belanja memungkinkan untuk membuat halaman produk

pelanggan dapat mengakses dan menambahkan item ke "keranjang belanja" virtual untuk pemesanan produk. Toko *online* harus mengintegrasikan informasi keranjang belanja ke *account* pelanggan.

### **2.1.3 Foto dan Deskripsi**

Setelah memiliki perangkat lunak keranjang belanja untuk toko *online*, toko *online* perlu untuk mengumpulkan dan meng-*upload* foto dan deskripsi produk. Pengambilan gambar produk bisa dilakukan sendiri atau bisa melalui tempat asal produk. Penulisan deskripsi produk dapat dilakukan sekitar 100 sampai 150 kata untuk menggambarkan setiap item yang akan menarik pengunjung untuk memesan agar tertarik memesan, termasuk informasi harga.

### **2.1.4 Kontak dan Halaman Privasi**

Ketika toko *online* sudah dioperasikan, pemilik harus meyakinkan pengunjung bahwa toko *online* adalah bisnis yang sah dan mapan. penting untuk memasukkan kontak dan halaman kebijakan privasi toko *online* di samping halaman toko *online*. Halaman kontak daftar alamat surat, nomor telepon, E-mail dan bentuk sosial media informasi kontak, sedangkan kebijakan privasi menginformasikan pengguna tentang bagaimana toko *online* berencana untuk mengelola informasi pelanggan.

## **2.2 Toko Online**

Toko *online* adalah proses dimana konsumen langsung memesan produk atau jasa dari seorang penjual secara *real-time*, tanpa perantara layanan, melalui *internet*. Ini adalah bentuk perdagangan elektronik. Sebuah toko *online* atau dikenal dengan istilah lain *e-shop*, *e-shop*, *internet shop*, *web shop*, *web store*,

*online shop*, atau *virtual store* membangkitkan analogi masyarakat bahwa untuk memesan suatu produk atau jasa di toko atau di pusat perbelanjaan tidak perlu datang ke tempat tersebut. Proses ini disebut *Business-to-Consumer* (B2C) toko *online*.(Josip, 2011).

Ketika mengunjungi sebuah toko *online* pelanggan dapat melihat item-item yang ditawarkan sebuah toko *online*, menambahkan item ke keranjang belanja dan kemudian membayar dengan kartu kredit atau transfer antar rekening bank.(Balles, Louise, 2012). Anda dapat memesan apa pun yang Anda inginkan secara *online*. Beberapa toko *online* tidak menerima *PayPal* sementara beberapa menolak kartu kredit.

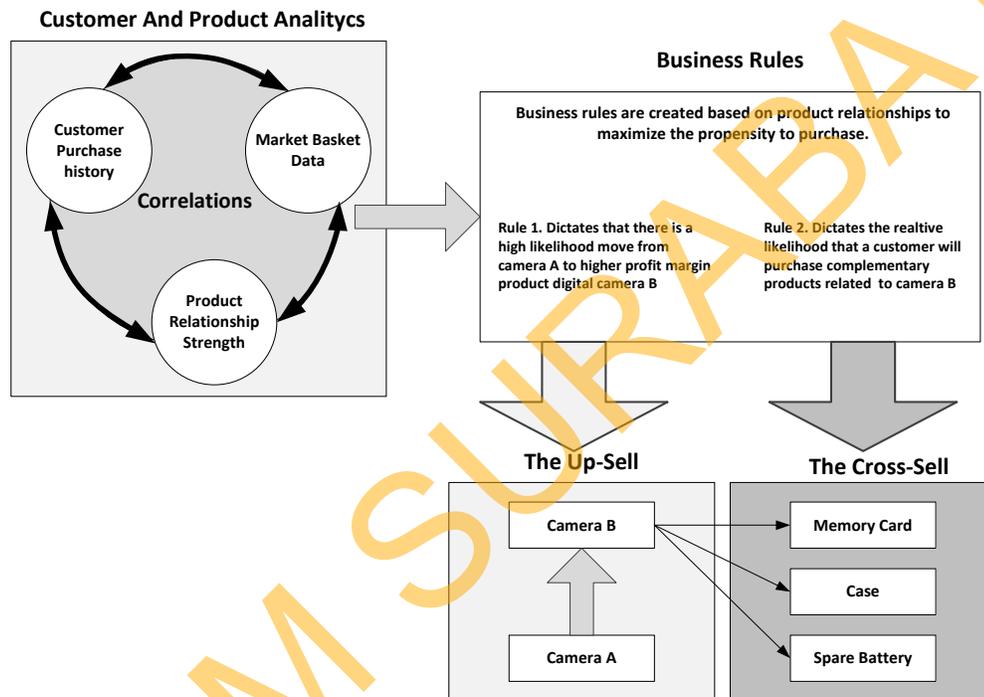
### **2.3 Cross selling**

Menurut Cohen , Kamakura, Ramaswami, Srivastava dalam Tama(2011:4) Strategi pemasaran *Cross selling* adalah sebuah istilah umum yang digunakan untuk menjelaskan penjualan *additional products* dan layanan kepada pelanggan yang telah memesan sesuatu dari perusahaan.

Menurut Tama(2011:11) Perusahaan dapat mengidentifikasi kebutuhan dan keinginan pelanggan untuk menetapkan *additional product* dari produk utamanya. *Additional product* yang disarankan dapat berupa *common-sense-based* (contohnya, jika pelanggan memesan sebuah kamera digital, maka *Cross selling* produknya biasanya adalah *memory card*, *case*, dan *spare-battery*) atau dapat pula berupa *data-driven* (*Cross selling* didapat dari data penjualan *history* pelanggan).

*Cross selling* biasanya digunakan oleh sebagian besar *online store* untuk menentukan rekomendasi produk apa yang seharusnya dijual. Sebagai contoh dapat dilihat salah satu *online bookstore* terkemuka seperti *Amazon.com* dapat

dengan jelas terlihat bahwa jika seorang pelanggan memesan buku secara *online* maka *website* akan memberikan pula rekomendasi mengenai *related books* yang direkomendasikan untuk dipesan. Menurut Tang, MacLennan dalam Tama(2011:4) Hal ini dapat dilakukan melalui analisis *cross selling* berdasarkan pola pemesanan pelanggan yang bertransaksi secara *online* melalui *website*.



Gambar 2.1 Business rules dengan cross selling.

## 2.4 Analisis Sistem

Didefinisikan secara resmi, analisis sistem merupakan kegiatan penelitian atau pembelajaran mengenai suatu sistem dan komponennya. Analisis sistem juga merupakan prasyarat untuk melakukan desain sistem, spesifikasi dari sistem yang baru dan lebih baik. Berdasarkan definisi klasiknya, dapat dilihat bahwa analisis sistem merupakan istilah yang secara kolektif menggambarkan fase awal dari pengembangan sistem.

Analisis sistem dibangun oleh pertimbangan bisnis dari pemilik sistem dan pengguna sistem. Sehingga analisis sistem dapat disimpulkan sebagai suatu teknik pemecahan masalah yang menguraikan suatu sistem menjadi bagian-bagian untuk tujuan dalam mempelajari seberapa baik bagian-bagian komponen bekerja dan berinteraksi untuk mencapai tujuan sistem yang sebenarnya.

Sedangkan analisis sistem informasi adalah tahap pengembangan dalam membangun *project* sistem informasi yang berfokus pada masalah bisnis dan kebutuhannya, terlepas dari teknologi yang dapat atau yang akan digunakan untuk solusi masalah tersebut. Analisis sistem mempunyai tahap-tahap yang harus dilakukan. Diantaranya tahap mendefinisikan ruang lingkup (*the scope definition phase*), tahap analisis masalah, tahap analisa kebutuhan, tahap analisis desain logikal(Whitten.Bentley, 2008).

#### **2.4.1 Ruang Lingkup**

Tahap pendefinisian ruang lingkup merupakan tahap awal dalam proses membangun suatu sistem. Dalam metodologi lain ini biasa disebut dengan tahap penyelidikan awal, tahap studi awal, tahap survey, atau tahap perencanaan. Tahap ini dapat menjawab pertanyaan “apakah sistem tersebut berguna untuk dikerjakan?” Untuk menjawab pertanyaan tersebut, harus mendefinisikan terlebih dahulu ruang lingkup sistem dan permasalahan yang ada meliputi peluang dan arahan yang memicu sistem tersebut dibangun. Selain meyakinkan bahwa sistem itu layak, pada tahap ini juga harus membangun perencanaan dalam hal skala, strategi pengembangan, menjadwalkan sumber daya dan anggaran.

**i. Analisis Masalah**

Tahap analisis masalah melakukan sistem analisis dengan memberikan pemahaman menyeluruh dari masalah, peluang, dan arahan yang memicu sistem tersebut dibangun. Dalam metodologi lain tahap ini biasa dikenal tahap studi, pemahaman sistem yang ada saat ini tahap investigasi secara mendetail, atau tahap analisis kemungkinan yang bisa digunakan. Tujuan dari tahap ini adalah mempelajari dan memahami area permasalahan dengan baik untuk dapat menganalisa masalah, peluang dan kendala.

**ii. Analisis Kebutuhan**

Tahap analisis kebutuhan merupakan tahap yang mendefinisikan kebutuhan bisnis untuk sebuah sistem baru. Tahap ini dapat menjelaskan tentang apa yang sebenarnya pengguna butuhkan dan yang diinginkan dari sistem baru yang akan dibangun sehingga dapat dikatakan bahwa tahap ini merupakan tahap kritis dalam kesuksesan suatu sistem informasi yang baru.

**iii. Desain Logis**

Sebuah desain logis dilanjutkan dengan suatu dokumen kebutuhan bisnis dengan menggunakan model sistem yang dapat menggambarkan struktur data, proses bisnis, aliran data, dan tampilan antarmuka.

**b. Desain Sistem**

Desain sistem adalah spesifikasi dari sebuah solusi detail yang berbasis komputer. Desain sistem informasi didefinisikan sebagai tugas yang berfokus pada spesifikasi dari solusi detail yang berbasis komputer. Hal tersebut dapat disebut *physical design*. Dengan demikian, dapat dikatakan analisis sistem menekankan masalah bisnis, sedangkan desain sistem berfokus pada masalah

teknis atau implementasi sistem. Desain sistem dibangun oleh pertimbangan teknis dari pendesain sistem (*System Designer*). Karena itu, desain sistem dibangun berdasarkan perspektif dari pendesain sistem. Suatu analisis sistem berfungsi sebagai fasilitator dari desain sistem.

Dari kesimpulan yang didapat pada tahap keputusan memicu desain sistem. Tujuan dari tahap desain yang utama ada dua. Pertama, analis berusaha untuk merancang suatu sistem yang memenuhi syarat kebutuhan dan dapat dengan mudah dimengerti oleh *end users*. Kedua, analis berusaha untuk menyajikan spesifikasi yang jelas dan lengkap untuk para *programmer* dan teknisi. Pada desain sistem ini terdapat desain arsitektur sistem, desain *database* sistem, desain *interface* (Whitten.Bentley, 2008).

Tahap desain sistem juga meliputi tahapan merancang pemodelan data yang dapat divisualisasikan melalui *Entity Relationship Diagram* (ERD), *Conceptual Data Model* (CDM), dan *Physical Data Model* (PDM). pemodelan proses yang dapat divisualisasikan melalui *Data Flow Diagram* (DFD) atau melalui *Unified Modelling Language* (UML). Dalam tahap ini juga mentransformasikan hasil dari analisis kebutuhan menjadi kebutuhan yang sudah lengkap yang difokuskan pada bagaimana memenuhi fungsi-fungsi yang dibutuhkan. Desain tersebut mencakup desain form dan laporan, desain antarmuka dan dialog, desain basis data dan *file (framework)*, dan desain proses atau desain struktur proses. (England, dkk, 2004 : 51-55)

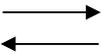
Sebagai dasar identifikasi titik-titik keputusan ini, dapat digunakan dokumen sistem bagan alir formulir (*paperworkflowchart* atau *form flowcharat*)

bila dokumentasi ini dimiliki oleh perusahaan. Berikut ini simbol–simbol dalam sistem maupun data *flow* diagram. (Jogiyanto, 2005 : 796-803)

### 1. Flowchart

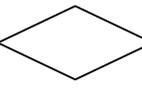
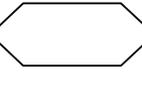
#### a. Flow Direction Symbols

Tabel 2.1 *Flow Direction Symbols*

	Simbol arus / <i>flow</i> , yaitu menyatakan jalanya arus suatu proses.
	Simbol <i>connector</i> , berfungsi menyatakan sambungan dari proses ke proses lainnya dalam halaman yang sama.
	Simbol <i>off-page connector</i> , menyatakan sambungan dari proses ke proses lainnya dalam halaman yang berbeda.

#### b. Processing Symbol

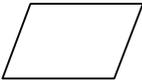
Tabel 2.2 *Processing Symbols*

	Simbol <i>process</i> , yaitu menyatakan suatu tindakan (proses) yang dilakukan oleh komputer.
	Simbol <i>manual</i> , yaitu menyatakan suatu tindakan (proses) yang tidak dilakukan oleh komputer.
	Simbol <i>decision</i> , yaitu menunjukkan suatu kondisi tertentu yang akan menghasilkan dua kemungkinan jawaban : ya / tidak.
	Simbol <i>preparation</i> , yaitu menyatakan penyediaan tempat penyimpanan suatu pengolahan untuk memberi harga awal.

	Simbol <i>terminal</i> , yaitu menyatakan permulaan atau akhir suatu program.
	Simbol <i>offline-storage</i> , menunjukkan bahwa data dalam simbol ini akan disimpan ke suatu media tertentu.
	Simbol <i>manual-input</i> , memasukkan data secara manual dengan menggunakan <i>online keyboard</i> .

c. *Input / Output Symbol*

Tabel 2.3 *Input / Output Symbol*

	Simbol <i>input-output</i> menyatakan proses <i>input</i> atau <i>output</i> tanpa tergantung jenis peralatanya.
	Simbol <i>storage</i> menyatakan <i>input</i> berasal dari disk atau <i>output</i> disimpan ke disk.
	Simbol <i>document</i> mencetak keluaran dalam bentuk dokumen (melalui printer).
	Simbol <i>display</i> mencetak keluaran dalam layar <i>monitor</i> .

2. *Data Flow Diagram*

*Data Flow Diagram* (DFD) fokus pada aliran data dari dan ke dalam sistem serta memproses data tersebut. (Kendall, 2003:231)

a. *External Entity*

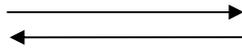
Suatu *External Entity* atau entitas merupakan orang, kelompok, departemen, atau sistem lain di luar sistem yang dibuat dapat menerima atau memberikan informasi atau data ke dalam sistem yang dibuat. Gambar 2.2 merupakan simbol entitas dalam DFD.



Gambar 2.2 Simbol *External Entity*

b. *Data Flow*

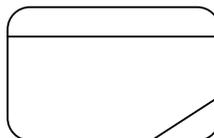
*Data Flow* atau aliran data disimbolkan dengan tanda panah. *Data Flow* menunjukkan arus data atau aliran data yang menghubungkan dua proses atau entitas dengan proses. Gambar 2.3 merupakan simbol *Data Flow*.



Gambar 2.3 Simbol *Data Flow*

c. *Process*

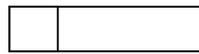
Suatu proses dimana beberapa tindakan atau sekelompok tindakan dijalankan. Gambar 2.4 merupakan simbol *Process*.



Gambar 2.4 Simbol *Process*

#### d. *Data Store*

*Data store* adalah simbol yang digunakan untuk melambangkan *process* penyimpanan data. Gambar 2.5 merupakan simbol *file* penyimpanan/*data store*.



Gambar 2.5 Simbol *Data Store*

### 3. *Entity Relationship Diagram*

*Attribute* adalah kolom di sebuah relasi. (Marlinda, 2004:28)

Macam-macam *attribute* yaitu :

#### a. *Simple Attribute*

*Attribute* ini merupakan *attribute* yang unik dan tidak dimiliki oleh *attribute* lainnya, misalnya *entity* mahasiswa yang *attribute*-nya NIM.

#### b. *Composite Attribute*

*Composite Attribute* adalah *attribute* yang memiliki dua nilai harga, misalnya nama besar (nama keluarga) dan nama kecil (nama asli).

#### c. *Single Value Attribute*

*Attribute* yang hanya memiliki satu nilai harga, misalnya *entity* mahasiswa dengan *attribute*-nya umur (tanggal lahir).

#### d. *Multi Value Attribute*

*Attribute* yang banyak memiliki nilai harga, misalnya *entity* mahasiswa dengan *attribute*-nya pendidikan (SD, SMP, SMA).

e. *Null Value Attribute*

*Attribute* yang tidak memiliki nilai harga, misalnya *entity* tukang becak dengan *attribute*-nya pendidikan (tanpa memiliki ijazah).

ERD ini diperlukan agar dapat menggambarkan hubungan antar *entity* dengan jelas, dapat menggambarkan batasan jumlah *entity* dan partisipasi antar *entity*, mudah dimengerti pemakai dan mudah disajikan oleh perancang database. (Kadir, 2008:46)

Untuk itu ERD dibagi menjadi 2 jenis model, yaitu :

a. *Conceptual Data Model (CDM)*

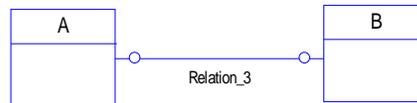
Merupakan jenis model data yang menggambarkan hubungan antar tabel secara konseptual.

b. *Physical Data Model (PDM)*

Merupakan jenis model data yang menggambarkan hubungan antar tabel secara fisikal.

ERD mempunyai 4 jenis hubungan antara lain :

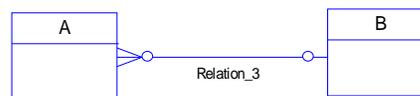
- a. Hubungan *one-to-one* ( 1:1 ) menyatakan bahwa setiap entitas pada tipe entitas A paling banyak berpasangan dengan satu entitas pada tipe entitas B. Begitu pula sebaliknya. Contoh :

Gambar 2.6 Hubungan *one-to-one*

b. Hubungan *one-to-many* ( 1:M ) menyatakan bahwa setiap entitas pada tipe entitas A bisa berpasangan dengan banyak entitas pada tipe entitas B, sedangkan setiap entitas pada B hanya bisa berpasangan dengan satu entitas pada tipe entitas B. Contoh :

Gambar 2.7 Hubungan *one-to-many*

c. Hubungan *many-to-one* ( M:1 ) menyatakan bahwa setiap entitas pada tipe entitas A paling banyak berpasangan dengan satu entitas pada tipe entitas B dan setiap entitas pada tipe entitas B bisa berpasangan dengan banyak entitas pada tipe entitas A. Contoh :

Gambar 2.8 Hubungan *many-to-one*

d. Hubungan *many-to-many* ( M:N ) Menyatakan bahwa setiap entitas pada suatu tipe entitas A bisa berpasangan dengan banyak entitas pada tipe entitas B dan begitu pula sebaliknya. Contoh :



Gambar 2.9 Hubungan *many-to-many*

Kardinalitas menggambar hubungan antara dua entitas dengan mengidentifikasi berapa banyak *instance* untuk setiap entitas yang nantinya dapat dihubungkan dengan setiap *instance* yang spesifik di entitas yang lain.

### 2.5.1 Desain Arsitektur Sistem

Tahap awal dari desain sistem adalah menentukan arsitektur *website* atau sistem. Arsitektur *website* mendefinisikan teknologi yang akan digunakan oleh satu, lebih, atau semua sistem informasi dalam hal data, proses, antarmuka, dan komponen jaringan.

### 2.5.2 Desain Database

Tahap selanjutnya dalam mendesain sistem adalah merancang spesifikasi *database* yang sesuai. *Database* merupakan sebuah sumber daya bersama. Beberapa *website* memungkinkan untuk menggunakan database yang sama. Perancang atau pendisain sistem juga harus menganalisa bagaimana *website* dapat mengakses data untuk meningkatkan performa.

### 2.5.3 Desain User Interface

Setelah proses desain *database*, desainer dapat bekerja sama dengan pengguna sistem untuk mengembangkan desain *input*, *output* dan spesifikasi dialog. Desain antarmuka penting dilakukan untuk menunjang pengguna dan manajer untuk dapat menggunakan *website* dengan mudah tanpa adanya kebingungan dalam melakukan *input* maupun membaca *output*.

**c. Black Box Testing**

*Black box testing* adalah cara pengujian yang terfokus pada apakah unit program memenuhi kebutuhan (*requirement*) yang disebutkan dalam spesifikasi (Al Fatta: 2007). Pada *black box testing*, cara pengujian hanya dilakukan dengan menjalankan atau mengeksekusi unit atau modul, kemudian diamati apakah hasil

STIKOM SURABAYA