

BAB II

TINJAUAN PUSTAKA

II.1. Perancangan

Perancangan adalah: Tahap setelah analisis sistem dalam pengembangan sistem, Pendefinisian kebutuhan fungsional, Persiapan untuk rancang bangun implementasi, Menggambarkan bagaimana sistem dibentuk (Abdur Rochman, 2016; 39).

II.2. Sistem

Sistem adalah “kumpulan atau himpunan dari unsur atau variabel-variabel yang saling terkait, saling berinteraksi, dan saling tergantung satu sama lain untuk mencapai tujuan (Abdur Rochman, 2016; 39).

Sistem merupakan suatu kesatuan yang terdiri dari dua atau lebih komponen atau subsistem yang saling berinteraksi untuk mencapai suatu tujuan (Andri Sukmaindrayana, 2015 : 22).

II.2.1. Klarifikasi Sistem

Sistem dapat diklasifikasikan dari beberapa sudut pandang, diantaranya adalah sebagai berikut :

1. Sistem abstrak (*abstract system*) dan sistem fisik (*physical system*).

Sistem abstrak adalah sistem yang berupa pemikiran atau ide-ide yang

tidak tampak secara fisik. Sistem fisik adalah sistem yang ada secara fisik.

2. Sistem alamiah (*natural system*) dan sistem buatan manusia (*human made system*). Sistem alamiah adalah sistem yang terjadi melalui proses alam, tidak dibuat manusia. Sistem buatan manusia yang melibatkan interaksi antara manusia dengan mesin disebut dengan *man-machine system*.
3. Sistem tertentu (*deterministic system*) dan sistem tak tentu (*probabilistic system*). Sistem tertentu beroperasi dengan tingkah laku yang sudah dapat diprediksi. Sistem tak tentu adalah sistem yang kondisi masa depannya tidak dapat diprediksi karena mengandung unsur probabilitas.
4. Sistem tertutup (*closed system*) dan sistem terbuka (*open system*). Sistem tertutup adalah sistem yang tidak berhubungan dan tidak terpengaruh dengan lingkungan luarnya. Sistem terbuka adalah sistem yang berhubungan dan terpengaruh dengan lingkungan luarnya (Reza Pratama, 2017 ; 5).

II.2.2. Konsep Dasar Informasi

Menurut Jogiyanto informasi ibarat darah yang mengalir dalam tubuh suatu organisasi. Suatu sistem yang kurang mendapatkan informasi akan menjadi lusuh, kerdil, dan akhirnya berakhir (Reza Pratama, 2017;5).

sistem informasi adalah kombinasi antara prosedur kerja, informasi, orang, dan teknologi informasi yang diorganisasikan untuk mencapai tujuan dalam sebuah organisasi (Andri Sukmaindrayana, 2015 : 22)

II.2.3. Informasi

Menurut Raymond Mcleod informasi adalah data yang telah diolah menjadi bentuk yang memiliki arti bagi si penerima dan bermanfaat bagi pengambilan keputusan saat ini atau mendatang. Secara umum informasi dapat di definisikan sebagai hasil pengolahan data dalam bentuk yang lebih berguna dan lebih berarti bagi penerimannya (Reza Pratama, 2017;5).

II.2.3. Data

Data adalah fakta atau bagian dari fakta yang digambarkan dengan simbol-simbol, gambar-gambar, nilai-nilai, bilangan-bilangan, uraian karakter yang mempunyai arti pada suatu karakter tertentu.

Data merupakan bahan baku yang diolah menjadi informasi. Data juga merupakan kenyataan yang menggambarkan suatu kejadian dan kesatuannya. Pengertian kejadian itu sendiri adalah sesuatu yang terjadi pada saat tertentu (Andri Sukmaindrayana, 2015; 23).

II.2.4. Sistem Informasi

Sistem informasi dapat berupa gabungan dari beberapa elemen teknologi berbasis komputer yang saling berinteraksi dan bekerjasama berdasarkan prosedur kerja yang telah ditetapkan, dimana data diproses dan diolah menjadi informasi

yang dapat digunakan dalam mendukung keputusan (Andri Sukmaindrayana, 2015; 23).

II.3. Perizinan Pemasangan Iklan

Menurut Peraturan Walikota Nomor 70 Tahun 2010, Reklame adalah benda, alat, perbuatan, atau media yang bentuk dan corak ragamnya dirancang untuk tujuan komersial memperkenalkan, menganjurkan, mempromosikan, atau untuk menarik perhatian umum terhadap barang, jasa, orang, atau badan, yang dapat dilihat, dibaca, didengar, dirasakan, dan/atau dinikmati oleh umum. Ada beberapa macam jenis reklame antara lain :

a. Reklame Permanen

Reklame papan dengan luas bidang 8 m² (delapan meter persegi) kebawah yang diselenggarakan di persil atau reklame berjalan.

b. Reklame Terbatas

Reklame papan dengan luas bidang lebih dari 8 m² (delapan meter persegi) yang diselenggarakan di lokasi persil dan reklame papan yang diselenggarakan di lokasi bukan persil.

c. Reklame Insidental

Reklame baliho, kain, reklame peragaan, reklame selebaran, reklame melekat, reklame film, reklame udara, reklame apung dan reklame suara (Nurcahyawati, 2013 : 31).

II.4. Microsoft Visual Studio 2010

Microsoft Visual Studio adalah sebuah *Integrated Development Environment* buatan *Microsoft Corporation*. *Microsoft Visual Studio* dapat digunakan untuk mengembangkan aplikasi dalam *native code* (dalam bentuk bahasa mesin yang berjalan di atas *Windows*) ataupun *managed code* (dalam bentuk *Microsoft Intermediate Language* di atas *.NET Framework*). Selain itu, *Visual Studio* juga dapat digunakan untuk mengembangkan aplikasi *Silverlight*, aplikasi *Windows Mobile* (yang berjalan di atas *.NET Compact Framework*). *Visual Basic* mencakup sebuah kode editor yang didukung oleh fitur *intellisense* atau yang disebut dengan *code refactoring*. *Debugger* telah terintegrasi bekerja pada *level source level debugger* dan *level debugger mesin*. *Toll built in* mencakup *form desainer* untuk membangun sebuah aplikasi *GUI*, *webdesainer*, *class desainer* dan *database schema desainer*. *Microsoft Visual Studio* didukung bahasa pemrograman yang berbeda. Adapun bahasa pemrograman yang didukung oleh *Visual Basic Studio* adalah bahasa pemrograman *C++*, *Visual Basic*, *Visual C#*. *Visual Studio* juga dapat mendukung bahasa pemrograman lain seperti *M*, *python* dan *ruby* yang semuanya itu terdapat pada *pack extra* yang terpisah dari *visual studio* (Nency Extise Putri, 2017; 206].

II.4. Basis Data (*Database*)

Menurut Asrianda dalam Urva, Gellysa (2008) *Database* adalah sekumpulan tabel-tabel yang saling berelasi, relasi tersebut bisa ditunjukkan

dengan kunci dari tiap tabel yang ada. Satu *database* menunjukkan satu lingkup perusahaan atau instansi.

Database juga merupakan kumpulan data yang umumnya menggambarkan aktifitas-aktifitas dan pelakunya dalam suatu organisasi. Sistem *database* merupakan sistem komputer yang digunakan untuk menyimpan dan mengelola data tersebut (Ade Hendini, 2016; 107)

II.5. SQL Server 2008

Microsoft SQL Server merupakan produk RDBMS (*Relational Database Management Sistem*) yang dibuat oleh Microsoft. Orang sering menyebutnya dengan SQL Server saja. Microsoft SQL Server juga mendukung SQL sebagai bahasa untuk memproses *query* ke dalam database. Microsoft SQL Server banyak digunakan pada dunia bisnis, pendidikan atau juga pemerintahan sebagai solusi database atau penyimpanan data.

Pada tahun 2000 Microsoft mengeluarkan SQL Server 2000 yang merupakan versi yang banyak digunakan. Berikut ini adalah beberapa fitur yang dari sekian banyak fitur yang ada pada SQL Server 2000.

- a. *XML Support*. Dengan fitur ini, Anda bisa menyimpan dokumen XML dalam suatu tabel, meng-*query* data ke dalam format XML melalui *Transact-SQL* dan lain sebagainya.
- b. *Multi-Instance Support*. Fitur ini memungkinkan Anda untuk menjalankan beberapa *database engine* SQL Server pada mesin yang sama.

- c. *Data Warehousing and Business Intelligence (BI) Improvements*. SQL Server dilengkapi dengan fungsi-fungsi untuk keperluan *Business Intelligence* melalui *Analysis Services*. Selain itu, SQL Server 2000 juga ditambahi dengan tools untuk keperluan data mining.
- d. *Performance and Scalability Improvements*. SQL Server menerapkan *distributed partitioned views* yang memungkinkan untuk membagi *workload* ke beberapa server sekaligus. Peningkatan lainnya juga dicapai di sisi DBCC, *indexed view*, dan *index reorganization*.
- e. *Query Analyzer Improvements*. Fitur yang dihadirkan antara lain: *integrated debugger*, *object browser*, dan fasilitas *object search*.
- f. *DTS Enhancement*. Fasilitas ini sekarang sudah mampu untuk memperhatikan *primary key* dan *foreign key constraints*. Ini berguna pada saat migrasi tabel dari RDBMS lain.
- g. *Transact-SQL Enhancements*. Salah satu peningkatan disini adalah T-SQL sudah mendukung UDF (*User-Definable Function*). Ini memungkinkan Anda untuk menyimpan rutin-rutin ke dalam *database engine* (M. Nuryana, 2014; 2).

II.6. *Unified Model Language(UML)*

Menurut Windu Gata, Grace (2013:4), *Unified Modeling Language (UML)* adalah bahasa spesifikasi standar yang dipergunakan untuk mendokumentasikan, menspesifikasikan dan membangun perangkat lunak. UML merupakan metodologi dalam mengembangkan sistem berorientasi objek dan juga merupakan alat untuk mendukung pengembangan system (Ade Hendini, 2016; 108).

II.6.1. Diagram-Diagram *Unified Modeling Language* (UML)




Adapun jenis-jenis dari diagram UML adalah sebagai berikut :


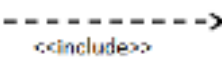
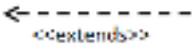
1. *Use Case* Diagram

Use case diagram merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. *Use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut (Ade Hendini, 2016; 108).

Simbol-simbol yang digunakan dalam *Use Case Diagram* yaitu:

Tabel II.1. Simbol – Simbol *Use Case Diagram*

Gambar	Keterangan
	<p><i>Use Case</i> menggambarkan fungsionalitas yang disediakan sistem sebagai unit-unit yang bertukar pesan antar unit dengan aktor, yang dinyatakan dengan menggunakan kata kerja</p>
	<p><i>Actor</i> atau Aktor adalah <i>Abstraction</i> dari orang atau sistem yang lain yang mengaktifkan fungsi dari target sistem. Untuk mengidentifikasi aktor, harus ditentukan pembagian tenaga kerja dan tugas-tugas yang berkaitan dengan peran pada konteks target sistem. Orang atau sistem bisa muncul dalam beberapa peran. Perlu dicatat bahwa aktor berinteraksi dengan <i>Use Case</i>, tetapi tidak memiliki kontrol terhadap <i>use case</i></p>
	<p>Asosiasi antara aktor dan <i>use case</i>, digambarkan dengan garis tanpa panah yang mengindikasikan siapa atau apa yang meminta interaksi secara langsung dan bukannya mengindikasikan data.</p>


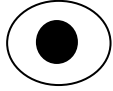

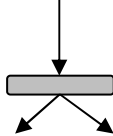
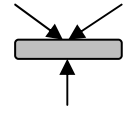
	Asosiasi antara aktor dan <i>use case</i> yang menggunakan panah terbuka untuk mengindikasikan bila aktor berinteraksi secara pasif dengan sistem
	<i>Include</i> , merupakan di dalam <i>use case</i> lain (<i>required</i>) atau pemanggilan <i>use case</i> oleh <i>use case</i> lain, contohnya adalah pemanggilan sebuah fungsi program
	<i>Extend</i> , merupakan perluasan dari <i>use case</i> lain jika kondisi atau syarat terpenuhi

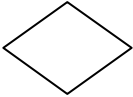
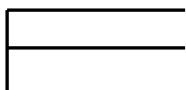
Sumber : (Ade Hendini, 2016)

2. Diagram Aktivitas (*Activity Diagram*)

Activity Diagram menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis. Simbol-simbol yang digunakan dalam *activity Diagram* (Ade Hendini; 2016 : 109).

Tabel II.2. Simbol – Simbol *Activity Diagram*

Gambar	Keterangan
	<i>Start Point</i> , diletakkan pada pojok kiri atas dan merupakan awal aktivitas
	<i>End Point</i> , akhir aktivitas
	<i>Activities</i> , menggambar kan suatu proses/kegiatan bisnis
	<i>Fork/</i> percabangan, digunakan untuk menunjukkan kegiatan yang dilakukan secara paralel atau untuk menggabungkan dua kegiatan paralel menjadi satu
	<i>Join</i> (penggabungan) atau <i>rake</i> , digunakan untuk menunjukkan adanya dekomposisi




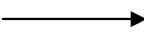

	<i>Decision Points</i> , menggambar kan pilihan untuk pengambilan keputusan, <i>true</i> atau <i>false</i>
	<i>Swimlane</i> , pembagian <i>activity diagram</i> untuk menunjukkan siapa melakukan apa



Sumber : (Ade Hendini, 2016)

3. *Sequence Diagram*

Sequence diagram menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan pesan yang dikirimkan dan diterima antar objek. Simbol-simbol yang digunakan dalam *Sequence Diagram* (Ade Hendini; 2016 : 110).

Tabel II.3. Simbol – Simbol *Sequence Diagram*

Gambar	Keterangan
	<i>Entity Class</i> , merupakan bagian dari sistem yang berisi kumpulan kelas berupa entitas-entitas yang membentuk gambaran awal system dan menjadi landasan untuk menyusun basis data
	<i>Boundary Class</i> , berisi kumpulan kelas yang menjadi <i>interfaces</i> atau interaksi antara satu atau lebih actor dengan sistem, seperti tampilan form entry dan form cetak
	<i>Control class</i> , suatu objek yang berisi logika aplikasi yang tidak memiliki tanggung jawab kepada entitas, contohnya adalah kalkulasi dan aturan bisnis yang melibatkan berbagai objek
	<i>Message</i> , symbol mengirim pesan antar <i>class</i>
	<i>Recursive</i> , menggambarkan pengiriman pesan yang dikirim untuk dirinya sendiri

	<i>Activation</i> , mewakili sebuah eksekusi operasi dari objek, panjang kotak ini berbanding lurus dengan durasi aktivasi sebuah operasi
	<i>Lifeline</i> , garis titik-titik yang terhubung dengan objek, sepanjang <i>life line</i> terdapat <i>activation</i>

(Sumber : Ade Hendini, 2016)

4. Diagram Kelas (*Class Diagram*)

Diagram Kelas (*Class Diagram*) merupakan hubungan antar kelas dan penjelasan detail tiap-tiap kelas di dalam model desain dari suatu sistem, juga memperlihatkan aturan-aturan dan tanggung jawab entitas yang menentukan perilaku sistem. *Class Diagram* juga menunjukkan atribut-atribut dan operasi-operasi dari sebuah kelas dan *constraint* yang berhubungan dengan objek yang dikoneksikan. *Class Diagram* secara khas meliputi : Kelas (*Class*), Relasi *Assosiations*, *Generalitation* dan *Aggregation*, atribut (*Attributes*), operasi (*operation/method*) dan *visibility*, tingkat akses objek eksternal kepada suatu operasi atau atribut. Hubungan antar kelas mempunyai keterangan yang disebut dengan *Multiplicity* atau *Cardinality* (Ade Hendini; 2016 : 114).

Tabel II.4. Simbol-Simbol Class Diagram

<i>Multiplicity Class Diagram</i> <i>Multiplicity</i>	Penjelasan
1	Satu dan hanya satu
0..*	Boleh tidak ada atau 1 atau lebih
1..*	1 atau lebih
0..1	Boleh tidak ada, maksimal 1
n..n	Batasan Antara. Contoh 2..4 mempunyai arti minimal 2 maksimal 4

(Sumber : Ade Hendini, 2016)