



BAB II

TINJAUAN PUSTAKA

2.1 Teori Umum

2.1.1 Pengertian Aplikasi

Menurut Jogiyanto aplikasi merupakan penggunaan dalam suatu komputer, instruksi (instruction) atau pernyataan (statement) yang disusun sedemikian rupa sehingga komputer dapat memproses input menjadi output.

Menurut Kamus Besar Bahasa Indonesia (KBBI) aplikasi adalah penerapan dari rancang sistem untuk mengolah data yang menggunakan aturan atau ketentuan bahasa pemrograman tertentu. Aplikasi adalah suatu program komputer yang dibuat untuk mengerjakan dan melaksanakan tugas khusus dari user (pengguna).

Wikipedia mengatakan Aplikasi adalah suatu subkelas perangkat lunak komputer yang memanfaatkan kemampuan komputer langsung untuk melakukan suatu tugas yang diinginkan pengguna. (www.sepengetahuan.com/2016/06/10-pengertian-aplikasi-menurut-para-ahli-lengkap.html)

2.1.2 Pengertian Website

Menurut Jhonsen (2004:5), *Website (Situs Web)* merupakan kumpulan dari halaman-halaman *web* yang berhubungan dengan file-file lain yang terkait. Dalam sebuah website terdapat suatu halaman yang dikenal dengan sebutan *home page*. *Home page* adalah sebuah halaman yang pertama kali dilihat ketika seseorang mengunjungi *website*. Dari *home page*, pengunjung dapat mengklik hyperlink untuk pindah kehalaman lain yang terdapat dalam *website* tersebut.



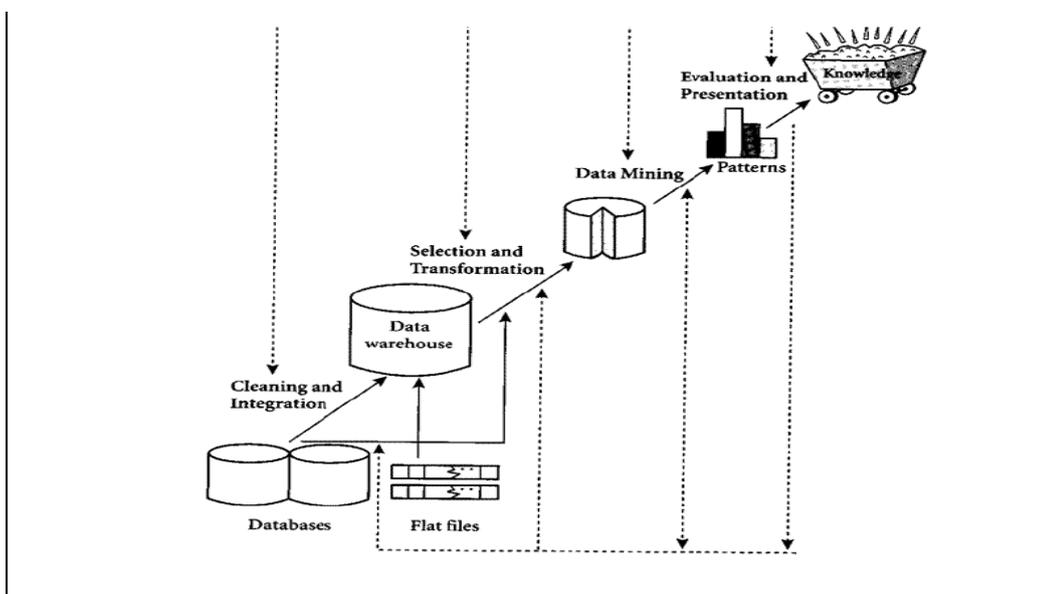
2.2 Teori Judul

2.2.1 Data Mining

Menurut Hermawati (2013:3), *Data Mining* merupakan proses interatif dan interaktif untuk menemukan pola atau model baru yang dapat digeneralisasi untuk masa yang akan datang, bermanfaat dan dapat dimengerti dalam suatu *database* yang sangat besar (*massive database*). *Data Mining* berisi pencarian *trend* atau pola yang diinginkan dalam *database* besar untuk membantu pengambilan keputusan di waktu yang akan datang. Pola-pola ini dikenali oleh perangkat tertentu yang dapat memberikan suatu analisa data yang berguna dan berwawasan yang kemudian dapat dipelajari dengan lebih teliti, yang mungkin saja menggunakan perangkat pendukung keputusan lainnya.

2.2.2 Tahapan Data Mining

Menurut Al-Fatah, 2007 dalam Sutrisno (2013:4). *Data mining* sebenarnya merupakan bagian dari proses *Knowledge Discovery in Database* (KDD), bukan sebagai teknologi yang utuh dan mandiri. *Data mining* merupakan suatu bagian langkah yang penting dalam proses KDD. Tahapan dan proses KDD diilustrasikan pada Gambar 2.1



Gambar 2.1 Tahapan pada proses *knowledge discovery*



1. *Data cleaning,*

Untuk menghilangkan data *noise* (data yang tidak relevan/berhubungan langsung dengan tujuan akhir proses *data mining*, misal: *data mining* yang bertujuan untuk menganalisa hasil penjualan, maka data-data dalam kumpulan seperti "nama pegawai", "umur", dan sebagainya dapat di-*ignore*) dan tidak konsisten.

2. *Data integration,*

Untuk menggabungkan suatu data *multiple data source*.

3. *Data selection,*

Untuk mengambil data sesuai kebutuhan untuk keperluan analisa.

4. *Data transformation,*

Untuk mentransformasikan data ke dalam bentuk yang lebih sesuai untuk di *mining*. *Data mining* Proses terpenting dimana metode tertentu diterapkan untuk menghasilkan *data pattern*.

5. *Pattern evaluation,*

Untuk mengidentifikasi apakah pola yang menarik yang didapatkan sudah cukup mewakili *knowledge* berdasarkan perhitungan tertentu.

6. *Knowledge presentation,*

Untuk mempresentasikan sebuah *knowledge* yang sudah didapatkan.

2.2.3 Clustering

Menurut Agusta, *Clustering* adalah metode penganalisaan data, yang sering dimasukkan sebagai salah satu metode *data mining*, yang tujuannya adalah untuk mengelompokkan data dengan karakteristik yang sama ke suatu 'wilayah' yang sama dan data dengan karakteristik yang berbeda ke 'wilayah' yang lain.

Ada beberapa pendekatan yang digunakan dalam mengembangkan metode clustering. Dua pendekatan utama adalah clustering dengan pendekatan partisi dan clustering dengan pendekatan hirarki. *Clustering* dengan pendekatan partisi atau sering disebut dengan *partition-based clustering* mengelompokkan data dengan memilah-milah data yang dianalisa ke dalam cluster-cluster yang ada. *Clustering* dengan pendekatan hirarki atau sering disebut dengan *hierarchical clustering*



mengelompokkan data dengan membuat suatu hirarki berupa dendogram dimana data yang mirip akan ditempatkan pada hirarki yang berdekatan dan yang tidak pada hirarki yang berjauhan. (<https://yudiagusta.wordpress.com/clustering/>)

2.3 Teori Program

2.3.1 PHP

Menurut Sunarfrihantono (2002:9). PHP adalah bahasa *server-side scripting* yang menyatu dengan HTML untuk membuat halaman web yang dinamis. Maksud dari *server-side scripting* adalah sintaks dan perintah-perintah yang diberikan akan sepenuhnya akan dijalankan diserver tetapi disertakan pada dokumen HTML. Pembuatan web ini merupakan kombinasi antara php sendiri sebagai bahasa pemrograman dan HTML sebagai pembangun halaman web.

2.3.2 MySQL

Menurut Sunarfrihantono (2002:13). MySQL adalah *multiuser database* yang menggunakan bahasa *Structured Query Language (SQL)*. MySQL dalam operasi *client server* melibatkan *server daemon* MySQL disisi *server* dan berbagai macam program serta *library* yang berjalan disisi *client*. MySQL mampu menangani data yang cukup besar. Perusahaan yang mengembangkan MySQL yaitu TEX, mengaku mampu menyimpan data lebih dari 40 database, 10.000 tabel, dan sekitar 7.000.000 baristotalnya kurang lebih 100 Gigabyte data.

2.3.3 Appserv

Menurut Fhadeli, Pengertian Appserv, Appserv adalah aplikasi yang memiliki fungsi untuk menginstal beberapa program yaitu Apache, PHP, MySQL dalam sekejap. Nah banyak orang memiliki masalah saat Install Apache, PHP, MySQL karena memang membutuhkan waktu yang agak lama untuk mengkonfigurasi dan terkadang dapat menyebabkan pusing 7 keliling. Dengan adanya Appserv orang sangat didipermudah. Beberapa Fitur Appserv adalah Apache, PHP, MySQL, phpMyAdmin.



Yang membedakan Appserv dengan yang lain adalah dia hanya menginstal tools yang benar - benar digunakan, sehingga dalam folder direktori itu sedikit sehingga tidak memakan space terlalu banyak.

(<http://www.maniacms.web.id/2012/01/pengertian-appserv.html>).

2.3.5 Adobe Dreamweaver CS6

Menurut Sadeli (2013:2), Adobe Dreamweaver CS6 adalah suatu perangkat lunak web editor keluaran Adobe System yang digunakan untuk membangun dan mendesain suatu website dengan fitur - fitur yang menarik dan kemudahan dalam penggunaannya.

2.3.5 UML (*Unified Modelling Language*)

Menurut Nugroho (2010:6), UML (*Unified Modeling Language*) adalah ‘bahasa’ pemodelan untuk sistem atau perangkat lunak yang berparadigma ‘berorientasi objek’. Pemodelan (modeling) sesungguhnya digunakan untuk penyederhanaan permasalahan-permasalahan yang kompleks sedemikian rupa sehingga lebih mudah dipelajari dan dipahami (Nugroho, *Pengertian Unified Modeling Language (UML) dan Modelnya Menurut Pakar dan Ahli*. <http://www.bangpahmi.com/2015/04/pengertian-unified-modelling-language-uml-dan-modelnya-menurut-pakar.html>).

1. *Use case Diagram*

Menurut Tantra, (Rudy Tantara ; 2012 : 152) *Usecase* adalah sekumpulan urutan tindakan yang dilakukan oleh system dan menghasilkan nilai yang dapat diamati, kepada actor tertentu. Digunakan untuk melakukan strukturisasi aturan benda-benda dalam model.

Syarat penamaan pada *usecase* adalah nama didefinisikan sesimpel mungkin dan dapat dipahami. Ada dua hal utama pada *usecase* yaitu pendefinisian apa yang disebut aktor dan *usecase*.

- a. Aktor merupakan orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu

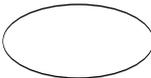
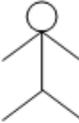


sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang.

- b. *Usecase* merupakan fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor.

Simbol-simbol yang digunakan pada *use case* diagram ditunjukkan pada Tabel 2.1.

Tabel 2.1 Simbol Pada Use Case diagram

Nama	Simbol	Diskripsi
Usecase		Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor, biasanya dinyatakan dengan menggunakan kata kerja di awal di frase nama Use Case
Aktor		Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri. Aktor hanya memberikan informasi ke sistem, aktor hanya menerima informasi dari sistem, aktor memberikan dan menerima informasi ke sistem dan dari sistem.
Asosiasi		Komunikasi antara aktor dan use case yang berpartisipasi pada use case





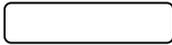
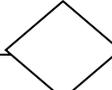
		atau use case memiliki interaksi dengan aktor. Asosiasi merupakan hubungan statis antar elemen yang menggambarkan elemen yang memiliki atribut berupa elemen lain, atau elemen yang harus mengetahui ekstensi elemen lain.
--	--	--

(Sumber : <http://www.materidosen.com/2017/04/use-case-diagram-lengkap-studi-kasus.html>)

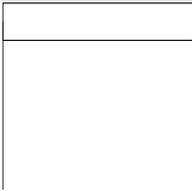
2. Activity Diagram

Activity diagram merupakan diagram yang bersifat dinamis. *Activity diagram* adalah tipe khusus dari diagram state yang memperlihatkan aliran dari suatu aktifitas ke aktifitas lainnya dalam suatu sistem dan berfungsi untuk menganalisa proses (Murad; 2013; <https://widuri.raharja.info/index.php/KP1222473399>).

Tabel 2.2 Simbol-Simbol Pada Activity Diagram

Nama	Simbol	Deskripsi
Status Awal		Status awal aktivitas sistem, sebuah aktivitas memiliki sebuah status awal
Aktivitas		Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
<i>Decision</i>		Asosiasi percabangan dimana jika ada pilihan aktivitas lebih



		dari satu.
Status Akhir		Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir
Swimlane		Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi.

(Sumber : <http://bopungmn.blogspot.co.id/2012/03/activity-diagram-salagsatu-cara-untuk.html>)

3. Sequence Diagram

Menurut Wijayanto, *Sequence* diagram dibuat berdasarkan *activity* diagram dan *class diagram* yang telah dibuat, maka digambarkan *sequencediagram* yang menggambarkan aliran pesan yang terjadi antar kelas dengan menggunakan operasi tersebut (Wijayanto ; 2013; <https://widuri.raharja.info/index.php/KP1222473399>).

Tabel 2.3 Simbol Sequence Diagram

Nama	Simbol	Deskripsi
Aktor		Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi



		yang akan dibuat itu sendiri.
<i>Lifeline</i>		Menyatakan kehidupan suatu objek, untuk menggambarkan kelas dan objek
Objek		Menyatakan Objek berinteraksi (pesan)
Waktu Aktif		Menyatakan objek dalam keadaan aktif dan berinteraksi, semua yang terhubung dengan waktu aktif.
Pesan Tipe	<code><<create>>()</code> 	Menyatakan suatu objek membuat objek yang lain, arah panah mengarah pada objek yang dibuat.

(Sumber : <http://kosemisme.blogspot.com/2015/04/sequence-diagram.html>)

4. Class Diagram

Sukanto dan Shalahuddin (2013:141), “Diagram kelas atau class diagram menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem”. Kelas memiliki apa yang disebut atribut dan metode atau operasi.

1. Atribut merupakan variabel-variabel yang dimiliki oleh suatu kelas.
2. Operasi atau metode adalah fungsi-fungsi yang dimiliki oleh suatu kelas.

Susunan struktur kelas yang baik pada diagram kelas sebaiknya memiliki jenis-jenis kelas berikut:

1. Kelas main

Kelas yang memiliki fungsi awal dieksekusi ketika sistem dijalankan.



2. Kelas yang menangani tampilan sistem (*view*)

Kelas yang mendefinisikan dan mengatur tampilan ke pemakai.

3. Kelas yang diambil dari pendefinisian use case (*controller*)

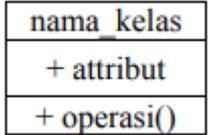
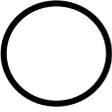
Kelas yang menangani fungsi-fungsi yang harus ada diambil dari pendefinisian use case, kelas ini biasanya disebut dengan kelas proses yang menangani proses bisnis pada perangkat lunak.

4. Kelas yang diambil dari pendefinisian data (*model*)

Kelas yang digunakan untuk memegang atau membungkus data menjadi sebuah kesatuan yang diambil maupun akan disimpan ke basis data.

Berikut adalah simbol-simbol yang ada pada diagram kelas menurut Rosa dan Shalahuddin (2013:146):

Tabel 2.4 Simbol-simbol *class diagram*

No.	Simbol	Keterangan
1.		Kelas pada struktur sistem
2.	 nama_interface	Sama dengan konsep interface dalam pemrograman berorientasi objek
3.		Relasi antarkelas dengan makna umum, asosiasi biasanya juga disertai dengan multiplicity
4.	Asosiasi berarah / <i>directed association</i>	Relasi antarkelas dengan makna kelas yang satu digunakan oleh



		kelas yang lain, asosiasi biasanya juga disertai dengan multiplicity
5.	Agregasi / aggregation 	Relasi antarkelas dengan makna semua- bagian (whole-part)
6.	Generalisasi 	Relasi antarkelas dengan makna generalisasi-spesialisasi (umum khusus)
7.	Kebergatungan / <i>dependency</i> 	Relasi antarkelas dengan makna kebergantungan antarkelas

Sumber : Rosa A.S dan M. Shalahudin (2014:146)