

## **BAB II**

### **LANDASAN TEORI**

#### **2.1 Dosen**

Sulastri (2007,1) menguraikan dosen adalah pendidik profesional dan ilmuwan dengan tugas utama mentransformasikan, mengembangkan, dan menyebarkan ilmu pengetahuan, teknologi, dan seni melalui pendidikan, penelitian, dan pengabdian kepada masyarakat. Dalam bahasa Indonesia dosen adalah pengajar pada perguruan tinggi. Dosen adalah salah satu komponen manusiawi dalam proses belajar, yang ikut berperan dalam usaha pembentukan sumber daya manusia yang potensial di bidang pembangunan. Dengan keilmuan yang dimilikinya dia dapat menjadikan anak didik menjadi orang yang cerdas. Dosen adalah salah satu faktor pembangun generasi muda yang terpelajar, membantu memberikan bimbingan agar kelak banyak orang yang dapat memajukan bangsa lewat pendidikan.

#### **2.2 Data Mining**

Data mining adalah proses yang mempekerjakan satu atau lebih teknik pembelajaran komputer (machine learning) untuk menganalisis dan mengekstraksikan pengetahuan secara otomatis. Fajar Astuti Hermawati (2009,26)

Data mining adalah suatu istilah yang digunakan untuk menemukan pengetahuan yang tersembunyi di dalam database. Data mining merupakan proses semi otomatis yang menggunakan teknik statistik, matematika, kecerdasan buatan, dan machine learning untuk mengekstraksi dan mengidentifikasi informasi pengetahuan potensial dan berguna yang bermanfaat yang tersimpan di dalam database besar Turban (2005,98).

Data mining merupakan bidang dari beberapa keilmuan yang menyatukan teknik dari pembelajaran mesin, pengenalan pola, statistik, database, dan visualisasi untuk penanganan permasalahan pengambilan informasi dari database yang besar Larose (2006,189).

Kemajuan luar biasa yang terus berlanjut dalam bidang data mining didorong oleh beberapa faktor, antara lain.

1. Pertumbuhan yang cepat dalam kumpulan data.
2. Penyimpanan data dalam data warehouse, sehingga seluruh perusahaan memiliki akses ke dalam database yang baik.
3. Adanya peningkatan akses data melalui navigasi web dan intranet.
4. Tekanan kompetisi bisnis untuk meningkatkan penguasaan pasar dalam globalisasi ekonomi.
5. Perkembangan teknologi perangkat lunak untuk data mining (ketersediaan teknologi).
6. Perkembangan yang hebat dalam kemampuan komputasi dan pengembangan kapasitas media penyimpanan.

Berdasarkan definisi-definisi yang telah disampaikan, hal penting yang terkait dengan data mining adalah :

1. Data mining merupakan suatu proses otomatis terhadap data yang sudah ada.
2. Data yang akan diproses berupa data yang sangat besar.
3. Tujuan data mining adalah mendapatkan hubungan atau pola yang mungkin memberikan indikasi yang bermanfaat.

Hubungan yang dicari dalam data mining dapat berupa hubungan antara dua atau pernyataan tersebut menegaskan bahwa dalam data mining otomatisasi tidak menggantikan campur tangan manusia. Manusia harus ikut aktif dalam setiap fase dalam proses data mining. Kehebatan kemampuan algoritma data mining yang terdapat dalam perangkat lunak analisis yang terdapat saat ini memungkinkan terjadinya kesalahan penggunaan yang berakibat fatal. Pengguna mungkin menerapkan analisis yang tidak tepat terhadap kumpulan data dengan menggunakan pendekatan yang berbeda. Oleh karenanya, dibutuhkan pemahaman tentang statistik dan struktur model matematika yang mendasari kerja perangkat lunak, Larose (2006,189).

### 2.2.1 Pengelompokan Data Mining

Data mining dibagi menjadi beberapa kelompok berdasarkan tugas yang dapat dilakukan.

#### 1. Deskripsi

Terkadang peneliti dan analis secara sederhana ingin mencoba mencari cara untuk menggambarkan pola dan kecenderungan yang terdapat dalam data. Sebagai contoh, petugas pengumpulan suara mungkin tidak dapat menemukan keterangan atau fakta bahwa siapa yang tidak cukup profesional akan sedikit didukung dalam pemilihan presiden. Deskripsi dari pola dan kecenderungan sering memberikan kemungkinan penjelasan untuk suatu pola atau kecenderungan

#### 2. Estimasi

Estimasi hampir sama dengan klasifikasi, kecuali variabel target estimasi lebih ke arah numerik daripada ke arah kategori. Model dibangun menggunakan record lengkap yang menyediakan nilai dan variabel target sebagai nilai prediksi. Selanjutnya, pada peninjauan berikutnya estimasi nilai dari variabel target dibuat berdasarkan nilai variabel prediksi.

#### 3. Prediksi

Prediksi hampir sama dengan klasifikasi dan estimasi, kecuali bahwa dalam prediksi nilai dari hasil akan ada di masa mendatang.

Contoh prediksi dalam bisnis dan penelitian adalah :

- a) Prediksi harga beras dalam tiga bulan yang akan datang.
- b) Prediksi persentase kenaikan kecelakaan lalu lintas tahun depan jika batas bawah kecepatan dinaikan.

Beberapa metode dan teknik yang digunakan dalam klasifikasi dan estimasi dapat pula digunakan (untuk keadaan yang tepat) untuk prediksi.

#### 4. Klasifikasi

Dalam klasifikasi, terdapat target variabel kategori. Sebagai contoh penggolongan pendapatan dapat dipisahkan dalam tiga kategori, yaitu pendapatan tinggi, pendapatan sedang dan pendapatan rendah.

Contoh lain klasifikasi dalam bisnis dan penelitian adalah :

- a) Menentukan apakah suatu transaksi kartu kredit merupakan transaksi yang curang atau bukan.
- b) Memperkirakan apakah suatu pengajuan hipotek oleh nasabah merupakan suatu kredit yang baik atau buruk.
- c) Mendiagnosis penyakit seorang pasien untuk mendapatkan termasuk kategori penyakit apa.

#### 5. Pengklusteran

Pengklusteran merupakan pengelompokan record, pengamatan, atau memperhatikan dan membentuk kelas objek-objek yang memiliki kemiripan. Kluster adalah kumpulan record yang memiliki kemiripan satu dengan yang lainnya dan memiliki ketidakmiripan dengan record-record dalam kluster lain.

Pengklusteran berbeda dengan klasifikasi yaitu tidak adanya variabel target dalam pengklusteran. Pengklusteran tidak mencoba untuk melakukan klasifikasi, mengestimasi atau memprediksi nilai dari variabel target. Akan tetapi, algoritma pengklusteran mencoba untuk melakukan pembagian terhadap keseluruhan data menjadi kelompok-kelompok yang memiliki kemiripan (homogen), yang mana kemiripan record dalam satu kelompok akan bernilai maksimal, sedangkan kemiripan dengan record dalam kelompok lain akan bernilai minimal.

Contoh pengklusteran dalam bisnis dan penelitian adalah :

- a) Mendapatkan kelompok-kelompok konsumen untuk target pemasaran dari suatu produk bagi perusahaan yang tidak memiliki dana pemasaran yang besar.
- b) Untuk tujuan audit akuntansi, yaitu melakukan pemisahan terhadap perilaku finansial dalam baik dan mencurigakan.
- c) Melakukan pengklusteran terhadap ekspresi dari gen, untuk mendapatkan kemiripan perilaku dari gen dalam jumlah besar.

## 6. Asosiasi

Tugas asosiasi dalam data mining adalah menemukan atribut yang muncul dalam satu waktu. Dalam dunia bisnis lebih umum disebut analisis keranjang belanja.

Contoh asosiasi dalam bisnis dan penelitian adalah :

- a) Meneliti jumlah pelanggan dari perusahaan telekomunikasi seluler yang diharapkan untuk memberikan respons positif terhadap penawaran upgrade layanan yang diberikan.
- b) Menemukan barang dalam supermarket yang dibeli secara bersamaan dan barang yang tidak pernah dibeli secara bersamaan.

### **2.2.2 Proses *Data Mining***

Secara sistematis, ada 3 (tiga) langkah utama dalam data mining. Gorunescu (2011,66) yaitu :

#### 1. Eksplorasi atau pemrosesan awal data

Eksplorasi atau pemrosesan awal data terdiri dari pembersihan data, normalisasi data, transformasi data, penanganan data yang salah, reduksi dimensi, pemilihan subset fitur, dan sebagainya.

#### 2. Membangun model dan melakukan validasi terhadapnya

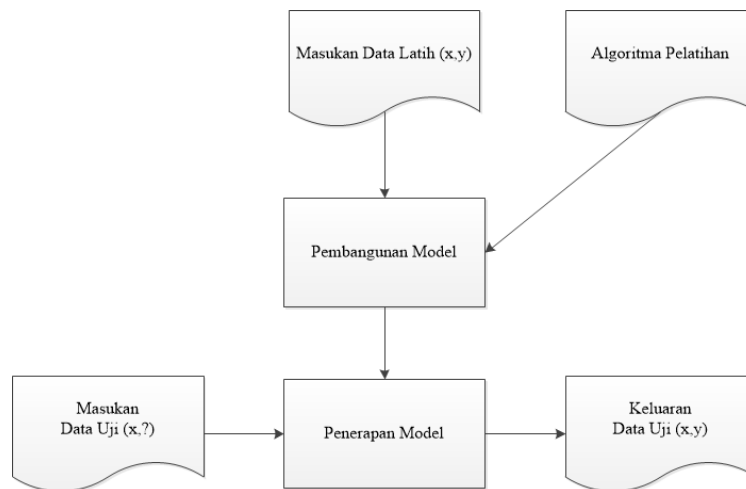
Membangun model dan melakukan validasi terhadapnya berarti melakukan analisis berbagai model dan memilih model dengan kinerja prediksi yang terbaik. Dalam langkah ini digunakan metode-metode seperti Klasifikasi, Regresi, Analisis Cluster, Deteksi Anomali, Asosiasi, Analisis Pola Sekuensial, dan sebagainya. Dalam beberapa referensi, Deteksi Anomali juga masuk dalam langkah eksplorasi. Akan tetapi, Deteksi Anomali juga dapat digunakan sebagai algoritma utama, terutama untuk mencari data yang spesial.

#### 3. Penerapan

Penerapan berarti menerapkan model pada data yang baru untuk menghasilkan perkiraan atau prediksi masalah yang diinvestigasi.

### 2.2.3 Konsep *Klasifikasi*

Menurut Prasetyo (2014,177) menyatakan bahwa klasifikasi dapat didefinisikan secara detail sebagai suatu pekerjaan yang melakukan pelatihan atau pembelajaran terhadap fungsi target  $f$  yang memetakan setiap vektor (sub fitur)  $x$  ke dalam satu dari sejumlah label kelas  $y$  yang tersedia. Pekerjaan pelatihan tersebut akan menghasilkan suatu model yang kemudian disimpan sebagai memori.



Gambar 2.1 Proses Pekerjaan Klasifikasi

Kerangka kerja klasifikasi ditunjukkan pada gambar 2.1. Pada gambar tersebut disediakan sejumlah data latih  $(x,y)$  untuk digunakan sebagai data pembangun model, kemudian menggunakan model tersebut untuk memprediksi kelas dari data uji  $(x,?)$  sehingga data uji  $(x,?)$  diketahui kelas  $y$  yang seharusnya.

Model yang sudah dibangun pada saat pelatihan kemudian dapat digunakan untuk memprediksi label kelas dari data baru yang belum diketahui label kelasnya. Dalam pembangunan model selama proses pelatihan tersebut diperlukan adanya suatu algoritma untuk membangunnya yang disebut sebagai algoritma pelatihan. Ada banyak algoritma pelatihan yang sudah dikembangkan oleh para peneliti seperti Decision Tree, K-Nearest Neighbor, Artificial Neural Network dan sebagainya. Setiap algoritma mempunyai prinsip yang sama yaitu melakukan suatu pelatihan sehingga di akhir

pelatihan model dapat memetakan (memprediksi) setiap vektor masukan ke label kelas keluaran yang benar.

#### **2.2.4 Pohon Keputusan**

Kusrini dan Luthfi (2009,188) pohon keputusan merupakan metode klasifikasi dan prediksi yang sangat kuat dan terkenal. Metode pohon keputusan mengubah fakta yang sangat besar menjadi pohon keputusan yang merepresentasikan aturan. Aturan dapat dengan mudah dipahami dengan bahasa alami. dan mereka juga dapat dieskpresikan dalam bentuk bahasa basis data seperti Structured Query Language untuk mencari record pada kategori tertentu.

Sebuah pohon keputusan mungkin dibangun dengan saksama secara manual atau dapat tumbuh secara otomatis dengan menerapkan salah satu atau beberapa algoritma pohon keputusan untuk memodelkan himpunan data yang belum terklasifikasi. Banyak algoritma yang dapat dipakai dalam pembentukan pohon keputusan, antara lain ID3, CART dan C4.5, Larose (2005,99).

Data dalam pohon keputusan biasanya dinyatakan dalam bentuk tabel dengan atribut dan record. Atribut menyatakan suatu parameter yang dibuat sebagai kriteria dalam pembentukan pohon. Misalkan untuk menentukan main tenis, kriteria yang diperhatikan adalah cuaca, angin dan temperatur. Proses pada pohon keputusan adalah mengubah bentuk data (tabel) menjadi model pohon, mengubah model pohon menjadi rule dan menyederhanakan rule.

#### **2.2.5 Algoritma C4.5**

Algoritma C4.5 merupakan algoritma yang digunakan untuk membentuk pohon keputusan. Menurut Kusrini dan Lutfhi (2009,51) menyatakan bahwa secara umum algoritma C4.5 untuk membangun pohon keputusan adalah sebagai berikut:

1. Pilih atribut sebagai akar.
2. Buat cabang untuk tiap-tiap nilai.

3. Bagi kasus dalam cabang.
4. Ulangi proses untuk setiap cabang sampai semua kasus pada cabang memiliki kelas yang sama.

Untuk memilih atribut sebagai akar, didasarkan pada nilai gain tertinggi dari atribut-atribut yang ada. Untuk menghitung gain digunakan rumus seperti tertera dalam persamaan berikut:

$$Gain_{S,A} = Entropy(S) - \sum_{i=1}^n \frac{|S_i|}{|S|} * Entropy(S_i)$$

Keterangan :

- S : himpunan kasus
- A : atribut
- n : jumlah partisi atribut A
- |S<sub>i</sub>| : jumlah kasus pada partisi ke-i
- |S| : jumlah kasus dalam S

Sementara itu, penghitungan nilai entropi dapat dilihat pada persamaan berikut :

$$Entropy(S) = - \sum_{i=1}^n p_i * \log_2 p_i$$

Keterangan :

- S : himpunan kasus
- A : atribut
- n : jumlah partisi S
- p<sub>i</sub> : proporsi dari S<sub>i</sub> terhadap S

sebagai contoh, untuk membentuk pohon keputusan dapat dilakukan dengan menentukan atribut sebagai akar lalu mencari nilai entropi dan gain.

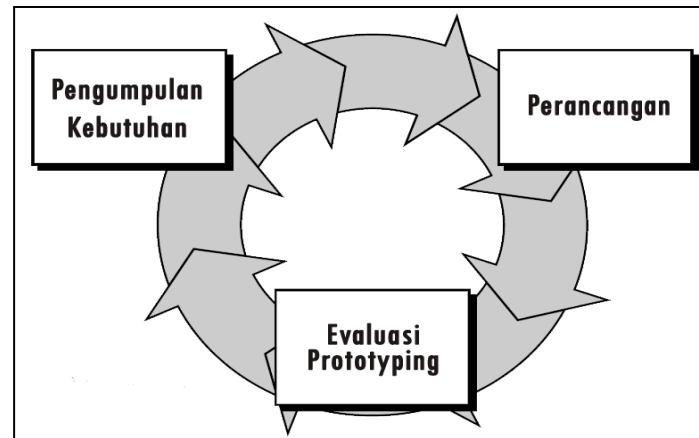
## 2.3 Metode Pengembangan Perangkat Lunak

### 2.3.1 Metode Prototipe (*prototyping model*)

A.S.Rosa (2016,31) menyatakan bahwa *Prototyping* paradigma dimulai dengan pengumpulan kebutuhan, pengembang bertemu dengan pengguna dan mengidentifikasi objektif keseluruhan dari perangkat lunak,



selanjutnya mengidentifikasi segala kebutuhan yang diketahui secara garis besar di mana definisi-definisi lebih jauh merupakan keharusan, kemudian dilakukan perancangan kilat, lalu diakhiri dengan evaluasi prototyping yang dapat dilihat pada gambar 2.4 berikut :



Gambar 2.2 *Prototype paradigm*

Tahap-tahap rekayasa software dalam prototype model pada gambar 2.4 di atas adalah sebagai berikut :

1) Pengumpulan kebutuhan

Developer dan klien bertemu untuk menentukan tujuan umum, kebutuhan yang diketahui dan gambaran bagian-bagian yang akan dibutuhkan berikutnya. Detail kebutuhan mungkin tidak dibicarakan disini, pada awal pengumpulan kebutuhan. Selanjutnya peneliti akan melakukan analisis terhadap data apa saja yang dibutuhkan, seperti analisis terhadap sistem yang berjalan, analisis kebutuhan perangkat lunak, analisis kebutuhan perangkat keras, dan analisis kebutuhan pelayanan darurat.

2) Perancangan

Perancangan dilakukan dengan cepat dan rancangan mewakili semua aspek software yang diketahui, dan rancangan ini menjadi dasar pembuatan prototype. Dalam tahap ini peneliti akan membangun sebuah versi prototype yang dirancang kembali dimana masalah-masalah tersebut diselesaikan

### 3) Evaluasi prototype

Pada tahap ini, calon pengguna mengevaluasi prototype yang dibuat dan digunakan untuk memperjelas kebutuhan software. Software yang sudah jadi dijalankan dan akan dilakukan perbaikan apabila kurang memuaskan. Perbaikan termasuk dalam memperbaiki kesalahan/ kerusakan yang tidak ditemukan pada langkah sebelumnya.

#### **2.3.2 Kelebihan Metode *Prototype***

Kelebihan *Prototype* adalah :

1. Adanya komunikasi yang baik antara pengembang dan pelanggan
2. Pengembangan dapat bekerja lebih baik dalam menentukan kebutuhan pelanggan.
3. Lebih menghemat waktu dalam pengembangan sistem.
4. Penerapan menjadi lebih mudah karena pemakaian mengetahui apa yang Diharapkannya.

#### **2.3.3 Kekurangan Metode *Prototype***

Kekurangan dari model *prototype* ini adalah sebagai berikut :

1. Resiko tinggi yaitu untuk masalah-masalah yang tidak terstruktur dengan baik, ada perubahan yang besar dari waktu ke waktu, dan adanya persyaratan data yang tidak menentu.
2. Interaksi pemakai penting. Sistem harus menyediakan dialog online antara pelanggan dan komputer.
3. Hubungan pelanggan dengan komputer yang disediakan mungkin tidak menceritakan teknik perancangan yang baik.

## **2.4 UML (*Unified Modeling Language*)**

*Unified Modelling Language* (UML) adalah suatu alat untuk memvisualisasikan dan mendokumentasikan hasil analisa dan desain yang berisi sintak dalam memodelkan sistem secara visual, juga merupakan satu kumpulan konvensi pemodelan yang digunakan untuk menentukan atau menggambarkan sebuah *system software* yang terkait dengan objek. Havaluddin (2014,1).

### **2.4.1 Tujuan Pemanfaatan UML**

Havilludin (2014,2) menguraikan bahwa tujuan utama dalam desain UML adalah :

1. Menyediakan bagi pengguna (analisis dan desain sistem) suatu
2. bahasa pemodelan visual yang ekspresif sehingga mereka dapat mengembangkan dan melakukan pertukaran model data yang bermakna.
3. Menyediakan mekanisme yang spesialisasi untuk memperluas konsep inti.
4. Karena merupakan bahasa pemodelan visual dalam proses pembangunannya maka UML bersifat independen terhadap bahasa pemrograman tertentu.
5. Memberikan dasar formal untuk pemahaman bahasa pemodelan.
6. Mendorong pertumbuhan pasar terhadap penggunaan alat desain sistem yang berorientasi objek (OO).
7. Mendukung konsep pembangunan tingkat yang lebih tinggi seperti kolaborasi, kerangka, pola dan komponen terhadap suatu sistem.
8. Memiliki integrasi praktik terbaik.

### **2.4.2 Komponen UML**

Havilludin (2014: 3) menguraikan bahwa komponen-komponen UML Sejauh ini para pakar merasa lebih mudah dalam menganalisa dan mendesain atau memodelkan suatu sistem karena UML memiliki seperangkat aturan dan notasi dalam bentuk grafis yang cukup spesifik. Komponen atau notasi UML diturunkan dari 3 (tiga) notasi yang telah ada sebelumnya yaitu Grady Booch, OOD (Object-Oriented Design), Jim Rumbaugh, OMT (Object Modelling Technique), dan Ivar Jacobson OOSE (Object-Oriented Software Engineering).

### **2.4.3 Class Diagram**

Havilludin (2014,3) menguraikan bahwa class diagram menggambarkan struktur statis dari kelas dalam sistem anda dan menggambarkan atribut,

operasi dan hubungan antara kelas. Class diagram membantu dalam memvisualisasikan struktur kelas-kelas dari suatu sistem dan merupakan tipe diagram yang paling banyak dipakai. Selama tahap desain, class diagram berperan dalam menangkap struktur dari semua kelas yang membentuk arsitektur sistem yang dibuat.

Class memiliki tiga area pokok :

1. Nama (dan stereotype)
2. Atribut
3. Metoda

#### 2.4.4 Use Case Diagram

Haviluddin (2014,6) menguraikan bahwa *Use case* adalah deskripsi fungsi dari sebuah system perspektif pengguna. *Use case* bekerja dengan cara mendeskripsikan tipikal interaksi antara user (pengguna) sebuah sistem dengan sistemnya sendiri melalui sebuah cerita bagaimana sebuah sistem dipakai. Urutan langkah-langkah yang menerangkan antara pengguna dan sistem disebut skenario. *Use case* merupakan awal yang sangat baik untuk setiap fase pengembangan berbasis objek, design, testing, dan dokumentasi yang menggambarkan kebutuhan sistem dari sudut pandang di luar sistem. Perlu diingat bahwa *use case* hanya menetapkan apa yang seharusnya dikerjakan oleh sistem, yaitu kebutuhan fungsional sistem dan tidak untuk menentukan kebutuhan non- fungsional, misalnya: sasaran kinerja, bahasa pemrograman dan lain sebagainya. berikut adalah symbol diagram *use case*:

Tabel 2.1 Simbol Diagram Use Case

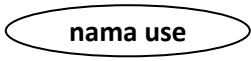

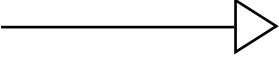
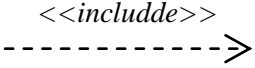
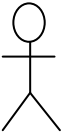
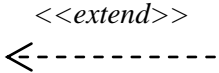
Simbol	Deskripsi
<i>Use case</i> 	Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor, biasanya dinyatakan dengan menggunakan kata kerja di awal <i>frase</i> nama <i>use case</i> .
<i>Asosiasi/association</i> 	Komunikasi antara aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan aktor.

Table 2.1 (Lanjutan)

<p>Generalisasi/<i>generalization</i></p> 	<p>Hubungan generalisasi dan spesialisasi (umum-khusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya.</p>
<p>Menggunakan <i>include</i></p> 	<p>Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan <i>use case</i> ini untuk menjalankan fungsinya atau sebagai syarat dijalankan <i>use case</i> ini ada dua sudut pandang yang cukup besar mengenai <i>include</i> di <i>use case</i> :</p> <ol style="list-style-type: none"> <li>1) <i>include</i> berarti <i>use case</i> yang ditambahkan akan selalu dipanggil saat <i>use case</i> tambahan dijalankan,</li> <li>2) <i>include</i> berarti <i>use case</i> yang tambahan akan selalu melakukan pengecekan apakah <i>use case</i> yang ditambahkan telah dijalankan sebelum <i>use case</i> tambahan dijalankan.</li> </ol>
<p>Aktor/<i>actor</i></p>  <p>nama aktor</p>	<p>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; biasanya dinyatakan menggunakan kata benda di awal frase nama aktor.</p>
<p>Ekstensi/<i>extend</i></p> 	<p>Relasi <i>use case</i> tambahan ke <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan itu; mirip dengan prinsip <i>inheritance</i> pada pemrograman berorientasi objek; biasanya <i>use case</i> tambahan memiliki nama depan yang sama dengan <i>use case</i> yang ditambahkan.</p>

### 2.4.5 Activity Diagram

Haviluddin (2014,4) menguraikan bahwa Activity Diagram adalah menggambarkan aktifitas-aktifitas, objek, state, transisi state dan event. Dengan kata lain kegiatan diagram alir kerja menggambarkan perilaku sistem untuk aktivitas.

S, Rosa & M (2016,97) menguraikan bahwa activity diagram menggambarkan workflow (alir kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Yang perlu

diperhatikan disini adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem.

Diagram aktivitas juga banyak digunakan untuk mendefinisikan hal-hal berikut :

1. Rancangan proses bisnis dimana setiap urutan aktivitas yang digambarkan merupakan proses bisnis sistem yang didefinisikan.
2. Urutan atau pengelompokan tampilan dari sistem/uses interface dimana setiap aktivitas dianggap memiliki sebuah rancangan antarmuka tampilan.
3. Rancangan pengujian dimana setiap aktivitas dianggap memerlukan sebuah pengujian yang perlu didefinisikan kasus ujinya.
4. Rancangan menu yang ditampilkan pada perangkat lunak.

Berikut adalah simbol *activity diagram*:

Tabel 2.2 Simbol *Activity Diagram*



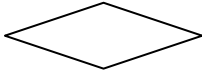


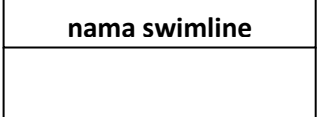
Simbol	Deskripsi
Status awal 	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah sistem awal.
Aktivitas 	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
Percabangan/ <i>decision</i> 	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu.
Penggabungan/ <i>join</i> 	Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu.
Status akhir 	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir.

Table 2.2 (Lanjutan)

<p>Swimlane</p> 	<p>Memisahkan organisasi bisnis yang bertanggungjawab terhadap aktivitas yang terjadi.</p>
---	--

### 2.4.6 Sequence Diagram

Havilludin (2014,5) menguraikan bahwa sequence diagram menjelaskan interaksi objek yang disusun berdasarkan urutan waktu. Secara mudahnya sequence diagram adalah gambaran tahap demi tahap, termasuk kronologi (urutan) perubahan secara logis yang seharusnya dilakukan untuk menghasilkan sesuatu sesuai dengan use case diagram.

berikut adalah symbol sequence diagram :

Tabel 2.3 Simbol Sequence Diagram


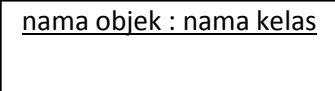

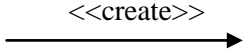

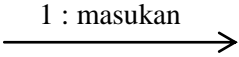
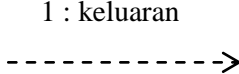
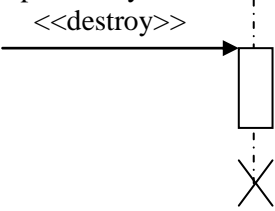
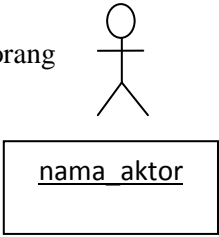
Simbol	Deskripsi
<p>Garis hidup/<i>lifeline</i></p> 	<p>Menyatakan kehidupan suatu objek.</p>
<p>Objek</p> 	<p>Menyatakan objek yang berinteraksi pesan.</p>
<p>Waktu aktif</p> 	<p>Menyatakan objek dalam keadaan aktif dan berinteraksi, semua yang terhubung dengan waktu aktif ini adalah sebuah tahapan yang dilakukan di dalamnya.</p>
<p>Pesan tipe create</p> 	<p>Menyatakan suatu objek membuat objek yang lain, arah panah mengarah pada objek yang dibuat.</p>
<p>Pesan tipe call 1 : nama_metode()</p> 	<p>Menyatakan suatu objek memanggil operasi/metode yang ada pada objek lain atau dirinya sendiri.</p>

Table 2.3 (Lanjutan)

Pesan tipe send 	Merupakan bahwa suatu objek mengirimkan data/masukan/informasi ke objek lainnya, arah panah mengarah pada objek yang dikirim.
Pesan tipe return 	Menyatakan bahwa suatu objek yang telah menjalankan suatu operasi atau metode menghasilkan suatu kembalian ke objek tertentu, arah panah mengarah pada objek yang menerima kembalian.
Pesan tipe destroy 	Menyatakan suatu objek mengakhiri hidup objek yang lain, arah panah mengarah pada objek yang diakhiri sebaliknya jika ada create maka ada destroy.
Aktor Atau orang 	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; biasanya dinyatakan menggunakan kata benda di awal frase nama aktor.

## 2.5 Aplikasi Web

Kadir (2009,67) aplikasi web adalah jenis aplikasi yang diakses melalui browser, misalnya Internet Explorer dan Mozilla Firefox. Aplikasi web yang paling dasar ditulis dengan menggunakan HTML. Sebagaimana diketahui, HTML (Hyper Text Markup Language) adalah bahasa standar untuk membuat halaman-halaman web.

### 2.5.1 PHP

Kadir (2009,72) menyatakan bahwa PHP adalah skrip yang dijalankan di server. Jadi, konsepnya berbeda dengan JavaScript yang dijalankan pada sisi klien. Keuntungan penggunaan PHP, kode yang menyusun program tidak perlu dibagikan ke pemakai yang berarti bahwa kerahasiaan kode dapat dilindungi. Hal menarik yang didukung oleh PHP tetapi tidak mungkin dilakukan oleh JavaScript adalah kenyataan bahwa PHP bisa digunakan untuk mengakses berbagai macam database, seperti Access, Oracle, MySQL, dan lain-lain. Seperti halnya JavaScript, kode PHP dapat disisipkan pada kode



HTML. Selain itu PHP juga bisa digunakan untuk menghasilkan kode-kode HTML. Skrip PHP diawali dengan `<?php` dan diakhiri dengan `?>`. Di dalam pasangan tanda tersebut terdapat pernyataan-pernyataan PHP. Seperti halnya pada JavaScript, antar pernyataan harus dipisahkan oleh tanda titik-koma (;).

### 2.5.2 HTML

Kadir (2009,84) menyatakan bahwa HTML adalah bahasa dengan tanda-tanda khusus yang digunakan di awal era web untuk menyajikan informasi. Kode HTML diawali dengan `<html>` dan diakhiri dengan `</html>`. Namun sebelum `<html>` diharuskan untuk diberikan deklarasi `<!DOCTYPE>`. Fungsi deklarasi ini adalah sebagai Public Text Identifier. Beberapa hal penting dalam kode HTML:

1. Tanda `<>` menyatakan sebuah tag.
2. Pada umumnya tag berpasangan. Contoh, `<html>` dengan `</html>`, `<head>` dengan `</head>` dan `<body>` dengan `</body>`.
3. Tag yang tidak berpasangan antara lain `<br/>` dan `<hr/>`
4. Pada tag yang berpasangan, tag yang berkedudukan sebagai tag penutup mempunyai bentuk berupa `</ >`.

### 2.5.3 Javascipt

Kadir (2009,108) menyatakan bahwa Javascript adalah bahasa skrip (bahasa yang kodenya ditulis menggunakan teks biasa) yang ditempelkan pada dokumen HTML dan diproses pada sisi klien. Dengan adanya bahasa ini, kemampuan dokumen HTML menjadi semakin luas. Sebagai contoh, dengan menggunakan JavaScript dimungkinkan untuk memvalidasi masukan-masukan pada formulir sebelum formulir dikirim ke server.

Beberapa contoh yang bisa dilakukan melalui JavaScript :

1. Memanipulasi jam lokal pada halaman web.
2. Mengatur warna latar belakang halaman web.
3. Mengganti gambar ketika pemakai menempatkan penunjuk mouse ke suatu gambar.
4. Memvalidasi keabsahan data yang dimasukkan oleh pemakai.

### 2.5.4 MySQL

Anhar (2010,141) mengatakan bahwa MySQL (My Structured Query Language) adalah sebuah program pembuat dan pengelola database atau yang sering disebut DBMS (Database Management System), sifat dari DBMS ini adalah Open Source dan ini didapatkan gratis pada alamat <http://www.mysql.com>.

### 2.6 Penelitian terdahulu

Tabel 2.4 berikut ini adalah beberapa penelitian terdahulu yang berkaitan dengan penerapan Algoritma C 4.5.

Tabel 2.4 Penelitian Terdahulu

No	Judul	Nama	Terbit/ Tahun	Keterangan
1	PERBANDINGAN KINERJA ALGORITME C4.5 DAN NAÏVE BAYES MENGLASIFIKASI PENYAKIT DIABETES	Hendra Marcos, Hengky S. U.	Jurnal Informatika , Vol.15, No.2, Bulan Desember 2015	Diabetes atau dapat disebut dengan kencing manis atau penyakit gula darah merupakan penyakit yang sukar disembuhkan namun kadar gula darah dapat dikontrol. Hal ini menyebabkan penderita penyakit diabetes semakin meningkat setiap tahunnya. Penelitian ini bertujuan untuk mengetahui algoritme mana yang memiliki tingkat akurasi klasifikasi paling.. baik, sehingga bisa digunakan untuk membantu dalam mengklasifikasi apakah seseorang terkena penyakit diabetes atau tidak. Data yang digunakan adalah Pima Indians Diabetesdataset yang diperoleh dari UCI machine learning. Pengolahan data mining dibagi menjadi dua tahap, yaitu tahap preprocessingdata dan seleksi fitur. Hasil

Tabel 2.4 (Lanjutan)

				penelitian yang telah dilakukan, algoritme C4.5 memiliki akurasi sebesar 73.82% dan meningkat menjadi 74,87% setelah dilakukannya seleksi atribut. Sedangkan naïve bayes memiliki tingkat akurasi sebesar 76,30% dan meningkat menjadi 77,47%. Hasil akhir dari penelitian ini adalah algoritme naïve bayes lebih baik dari pada algoritme C4.5 karena memiliki tingkat akurasi yang lebih baik
2	Penerapan Algoritma Decision Tree C4.5 Untuk Diagnosa Penyakit Stroke Dengan Klasifikasi Data Mining Pada Rumah Sakit Santa Maria Pemasang	Sigit Abdillah	Jurnal informatika Universitas Dian Nuswantoro, Jl. Nakula 1 No. 5-11 Semarang, 2014	<i>Data Mining</i> adalah proses ekstraksi sebelumnya tidak dikenal dan dipahami dari database berukuran besar dan digunakan untuk membuat keputusan bisnis yang penting. pasien penyakit saraf khususnya penyakit <i>Stroke</i> untuk dikelola menggunakan algoritma C4.5 dengan metode klasifikasi <i>data mining</i> .
3	Penerapan Algoritma ID3 Dan C45 Dalam Menemukan Hubungan Data Awal Masuk Mahasiswa Dengan Prestasi Akademik	Novyana Arista	Jurnal Informatika Fakultas Teknik UMRAH, 2014	Pemanfaatan data yang ada di dalam sistem informasi untuk menunjang kegiatan pengambilan keputusan, tidak cukup hanya mengandalkan data operasional saja, diperlukan suatu analisis data untuk menggali potensi - potensi informasi yang ada yaitu dengan menggunakan teknik data mining. Untuk itu penelitian ini menggunakan dua algoritma C4.5 dan ID3 dan

Tabel 2.4 (Lanjutan)

				data yang digunakan adalah data awal masuk mahasiswa dan data prestasi akademik mahasiswa Universitas Maritim Raja Ali Haji yaitu mahasiswa angkatan 2012,
4	Penerapan Algoritma C4.5 Dalam Pemilihan Bidang Peminatan Program Studi Sistem Informasi Di Stmik Potensi Utama Medan.	Fina Nasari	Jurnal Informatika STMIK Potensi Utama, 2014	Saat ini mayoritas mahasiswa memilih bidang peminatan mengikuti pilihan yang diambil mayoritas teman-teman satu kelas, tanpa mempertimbangkan faktor prestasi akademik mahasiswa. Hal ini berdampak pada ketidaksesuaian bidang peminatan dengan minat dan keterampilan mahasiswa tersebut, akibatnya banyak mahasiswa yang mengalami kesulitan ketika menyelesaikan tugas akhir.