

BAB 2

TINJAUAN PUSTAKA

2.1 Tinjauan Pustaka

Penerapan data mining dalam mengevaluasi kelayakan pemberian kredit saat ini telah banyak digunakan. Beberapa penelitian yang relevan selama 5 (lima) tahun terakhir yang menjadi referensi terkait dalam penelitian. Beberapa dari penelitian tersebut telah menerapkan metode data mining untuk mengevaluasi kelayakan pemberian kredit dengan pemakaian sejumlah atribut dalam penyelesaiannya. Berikut beberapa penelitian yang terkait dengan penelitian penulis antara lain penelitian yang dilakukan oleh Hera Wasiati dan Wijayanti [10]. Setiap perusahaan atau instansi pada umumnya telah menggunakan aplikasi yang terkomputerisasi sebagai penunjang untuk memudahkan dan mempercepat pengolahan data. Kenyataan dalam penyeleksian Tenaga Kerja Indonesia dinilai kurang siap karena sistem yang digunakan masih bersifat manual mengakibatkan proses penyeleksian calon tenaga kerja kurang efektif. Untuk menyelesaikan masalah tersebut, peneliti mengaplikasikan algoritma Naïve Bayes dalam menentukan kelayakan calon tenaga kerja Indonesia dengan kategori layak diterima atau ditolak. Dalam penelitian ini menggunakan data hasil pendaftaran calon tenaga kerja Indonesia sedangkan atribut yang digunakan adalah usia, pendidikan, tinggi badan, berat badan, dan hasil tes. Dari hasil pengujian melibatkan data sebanyak 542 record dengan 362 digunakan sebagai data *training* dan 180 digunakan sebagai data *testing*. Pengolahan data tersebut menghasilkan pola akurasi sebesar 73.89% dan errornya 26.11% dengan rincian data yang diprediksi benar sebesar 133 dan data yang diprediksi salah sebesar 47.

Penelitian yang dilakukan oleh Bustami [11], menjadi salah satu bahan rujukan karena penjelasan dalam penggunaan algoritma Naïve Bayes yang dilihat dari perhitungan manual diketahui sangat mudah dimengerti sehingga pada proses algoritma Naïve Bayes dapat diketahui menggunakan data kategorikal. Permasalahan yang diangkat dalam penelitian ini adalah meningkatnya jumlah

nasabah perusahaan asuransi yang sering menunggak pembayaran premi asuransi. Metode yang digunakan pada penelitian ini adalah Naïve Bayes untuk mengklasifikasi data nasabah yang dikategorikan ke dalam tiga jenis yaitu lancar, kurang lancar, dan tidak lancar. Adapun atribut yang digunakan adalah ID, nama, jenis kelamin, status, usia, pekerjaan, penghasilan per tahun, masa asuransi, dan cara pembayaran. Penelitian ini menghasilkan sebuah aplikasi data mining yang dapat dipakai oleh perusahaan asuransi dalam pengambilan keputusan untuk menentukan kelayakan penerimaan nasabah baru berdasarkan data historis nasabah lama yang sudah melalui proses sebelumnya dalam hal pembayaran premi asuransi.

Penelitian selanjutnya yang menjadi bahan rujukan adalah penelitian yang dilakukan oleh Diasrina Dahri, Fahrul Agus, dan Dyna Marisa Kairina [12]. Berawal dari permasalahan yang dipaparkan yakni diketahui dari jumlah 900 pendaftar beasiswa bidikmisi di salah satu Universitas setiap tahunnya mengakibatkan lambannya proses seleksi yang terjadi dan berpotensi tidak konsisten. Ketidakkonsistenan pada sistem penentuan penerima beasiswa bidikmisi ini menyebabkan tujuan penyelenggaraan beasiswa menjadi tidak terarah, tidak transparan dan tidak tepat sasaran. Oleh karena itu peneliti bertujuan untuk membantu bagian proses seleksi dengan mengembangkan aplikasi perangkat lunak sistem pendukung keputusan menggunakan metode Naïve Bayes. Penentuan penerima beasiswa menggunakan beberapa atribut antara lain: pekerjaan orang tua, penghasilan orang tua, jumlah tanggungan, daya listrik (*watt*), dan nilai ujian nasional. Hasil dari penelitian ini adalah aplikasi sistem pendukung keputusan dengan tingkat akurasi sebesar 85.56% yang artinya hasil tersebut tergolong akurat.

Adapun penelitian dari Deny Cahya Mahendra dan Achmad Wachid Kurniawan [13] yaitu diawali dengan permasalahan yang sering dihadapi perbankan adalah banyaknya nasabah yang menunggak dalam pembayaran kredit. Dari pemaparan permasalahan tersebut, peneliti menggunakan algoritma Naïve Bayes untuk memprediksi kelayakan pemberian kredit agar menghindari resiko kredit macet. Data yang akan digunakan adalah sebanyak 489 record dan menggunakan 2 jenis

atribut yaitu atribut kategorikal (jenis kelamin, status, dan pekerjaan) dan atribut numerik (umur, jumlah penghasilan, jangka waktu kredit, angsuran, suku bunga dan keterangan macet atau lancar yang merupakan label target). Hasil penelitian algoritma Naïve Bayes ini, dapat diterapkan dalam menilai kelayakan kredit pada akurasi data awal sebesar 79,84% yang dapat dikategorikan dalam *Fair Classification*, sedangkan data yang sudah melalui tahap pengolahan data yaitu sebesar 88,61% sudah termasuk dalam *Good Classification*, dan terakhir data yang sudah melalui pengolahan data dan konversi data memiliki akurasi paling tinggi yaitu 90,28% merupakan rentang akurasi *Excellent Classification*, data yang diolah dengan *pre-processing* lebih unggul dibandingkan data yang belum diolah, sekalipun algoritma Naïve Bayes mampu menangani data yang hilang atau tidak lengkap. Untuk mengembangkan penelitian ini juga dapat dilakukan penambahan *record* data atau atribut lebih banyak untuk mendapatkan nilai akurasi yang lebih baik.

Tabel 2.1 Penelitian Terkait

No	Nama Peneliti dan Tahun	Masalah	Metode	Hasil
1.	Hera Wasiati dan Dwi Wijayanti (2016)	Proses penyeleksian calon tenaga kerja Indonesia dinilai kurang efektif karena masih bersifat manual	Klasifikasi dengan algoritma Naïve Bayes	Menghasilkan pola akurasi sebesar 73.89% dan errornya 26.11% dengan rincian data yang diprediksi benar sejumlah 133 dan data yang diprediksi salah sejumlah 47
2.	Bustami (2014)	Meningkatnya jumlah nasabah yang sering menunggak pembayaran premi asuransi	Klasifikasi dengan algoritma Naïve Bayes	Hasil klasifikasi yang terbagi menjadi 3 kategori yaitu lancar, kurang lancar, dan tidak lancar. Klasifikasi ini dapat digunakan untuk pengambilan keputusan dalam penentuan nasabah yang layak untuk diterima atau ditolak
3.	Diasrina Dahri, Fahrul Agus, dan Dyna Marisa Kairina (2016)	Ketidakkonsistenan pada sistem penentuan penerima beasiswa bidikmisi menyebabkan tujuan penyelenggaraan beasiswa tidak terarah, tidak	Klasifikasi dengan algoritma Naïve Bayes	Penelitian ini menghasilkan sistem pendukung keputusan untuk menentukan kelayakan penerima beasiswa bidikmisi dengan tingkat akurasi sebesar 85.56%

No	Nama Peneliti dan Tahun	Masalah	Metode	Hasil
		transparan dan tidak tepat sasaran		
4.	Deny Cahya Mahendra dan Achmad Wahid Kurniawan (2015)	Banyaknya jumlah nasabah tidak tepat waktu atau menunggak dalam pembayaran kredit	Klasifikasi dengan algoritma Naïve Bayes	Hasil pada akurasi data awal sebesar 79.84%, dan data yang sudah melalui tahap pengolahan data sebesar 88.61%, sedangkan data yang sudah melalui pengolahan data dan konversi data memiliki akurasi paling tinggi yaitu sebesar 90.28%

2.2 Landasan Teori

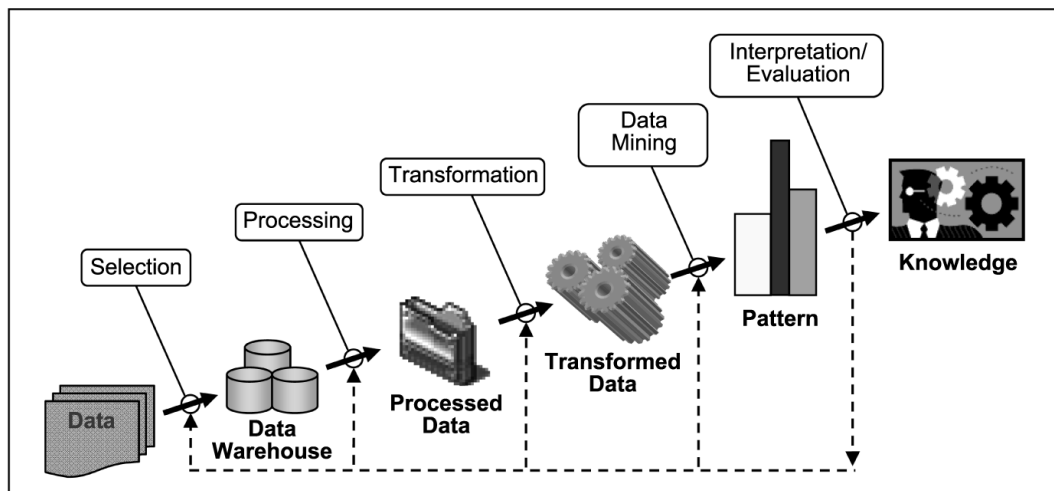
Dalam mendukung proses penelitian ini perlu adanya landasan teori yang sesuai dengan tujuan penelitian. Landasan teori berfungsi untuk mendukung proses penelitian ini yang diambil dari buku, jurnal, paper, maupun prosiding.

2.2.1 Data Mining

Data mining merupakan suatu bidang ilmu yang digunakan untuk mengeksplorasi terkait dengan sejumlah pengetahuan yang tersembunyi dalam suatu *database*. Data mining merupakan suatu proses yang di dalamnya melibatkan teknik matematik, statistik, *machine learning*, dan kecerdasan buatan untuk mengidentifikasi berbagai macam informasi berupa *knowledge* yang bersumber dari *database* dalam ukuran yang sangat besar [14].

Data mining juga sering disebut dengan istilah Knowledge Discovery in Database (KDD) yang artinya menggali, mengumpulkan, menemukan, menambang

database dalam jumlah yang besar. Kenyataannya data mining dan KDD merupakan dua konsep yang berbeda, namun keduanya memiliki keterkaitan satu dengan yang lain. Keterkaitan tersebut dapat dibuktikan dalam tahapan KDD yaitu adanya proses data mining.



Gambar 2.1 Tahapan *Knowledge Discovery in Database* [14]

Berdasarkan gambar 2.1 dapat dijelaskan tahapan-tahapan yang dimiliki KDD sebagai berikut:

1. *Data Selection*

Memasuki tahapan KDD yang pertama yaitu penyeleksian data dari sekumpulan data operasional. Kemudian data yang telah terseleksi akan ditempatkan ke dalam suatu berkas berbeda dari data awal yang akan dibutuhkan pada proses data mining.

2. *Pre-processing*

Sebelum melalui proses data mining, perlunya dilakukan proses *pre-processing* atau *cleaning* pada data yang sudah ditargetkan dalam KDD. Proses *cleaning* itu sendiri yaitu pemeriksaan data yang tidak konsisten, pembuangan data yang terduplikasi, perbaikan data yang salah seperti salah ketik (*tipografi*). Hal ini dilakukan agar dapat menciptakan hasil data atau informasi yang cukup relevan dalam tahapan KDD.

3. Transformation

Data kategorikal dan data numerik adalah inputan yang dapat ditampung oleh proses data mining. Data-data kemudian akan ditransformasikan ke dalam bentuk yang sesuai dengan mekanisme yang dibutuhkan dalam mempermudah proses data mining. Hal ini ditujukan untuk menghasilkan sebuah pola informasi sesuai yang direncanakan.

4. Data Mining

Data mining adalah proses mencari pola, mengekstrak pengetahuan, atau menggali informasi menggunakan metode tertentu. Teknik, metode dan algoritma yang dimiliki data mining sangat bervariasi dan data yang digunakan juga harus memenuhi standar. Pemakaian data berjumlah banyak sangat dianjurkan dalam teknik data mining agar tidak terjadi *invalid data* atau kesalahan lainnya.

5. Interpretation (Evaluation)

Informasi atau pola yang dihasilkan dari proses data mining harus digambarkan dengan cara yang mudah dipahami. Hal ini menjadi tahapan terakhir dalam KDD yang disebut *interpretation* atau *evaluation*. Tahapan ini mencakup evaluasi pola atau informasi yang dihasilkan apakah bertentangan dengan hipotesis sebelumnya.

2.2.2 Klasifikasi

Dalam data mining terdapat beberapa teknik yang memiliki fungsi berbeda-beda yaitu prediksi, estimasi, deskripsi, klasifikasi, pengklasteran dan asosiasi. Klasifikasi merupakan metode yang digunakan untuk menemukan model atau fungsi yang digambarkan dengan perbedaan kelas data atau konsep yang berfungsi untuk memprediksi kelas dari objek yang label kelasnya tidak diketahui [11]. Klasifikasi akan terjadi 2 proses, proses pertama adalah *learning* (fase *training*), dimana algoritma klasifikasi diciptakan untuk menganalisa data *training* kemudian direpresentasikan ke dalam bentuk *rule* klasifikasi. Proses kedua adalah klasifikasi, dimana data *testing* akan dipakai untuk memperkirakan akurasi dari

rule klasifikasi [15]. Menurut *Gonurescu* (2011), terdapat komponen yang mendasari proses klasifikasi sebagai berikut [16]:

1. *Class*

Variabel dependen berupa data kategorikal yang merepresentasikan suatu label yang melekat pada objek. Hal ini dapat dicontohkan seperti: jenis bunga, jenis bencana, resiko kredit macet, dan resiko kanker.

2. *Predictor*

Variabel independen yang direpresentasikan oleh (data) atribut tertentu. Hal ini dapat dicontohkan seperti: penghasilan per bulan, jaminan, aset, dan umur.

3. *Training dataset*

Sekumpulan data yang mengandung nilai dari kedua komponen sebelumnya yang dimanfaatkan untuk pemilihan kelas yang tepat berdasarkan *predictor*.

4. *Testing dataset*

Sekumpulan data baru yang akan melalui proses klasifikasi menggunakan model yang telah ditentukan. Hasil dari klasifikasi tersebut yang kemudian akan dievaluasi keakuratannya.

2.2.3 Algoritma Naive Bayes Classifier

Naive Bayes merupakan suatu istilah algoritma dalam data mining dengan mendasari metode klasifikasi [11]. Metode pengklasifikasian dalam Naive Bayes ini dikenal dengan metode probabilitas dan statistik yang dikemukakan oleh ilmuwan asal Inggris Thomas Bayes, yaitu memprediksi peluang di masa mendatang berdasarkan kejadian di masa sebelumnya sehingga dikenal dengan *Teorema Bayes*. Naive Bayes diketahui sebagai algoritma yang paling sederhana di antara algoritma klasifikasi. Oleh karena itu, algoritma ini diasumsikan dapat dengan mudah menangani *dataset* dengan banyak atribut sekalipun [8]. Formulasi Teorema Bayes sebagai metode klasifikasi sebagai berikut:

$$P(H|X) = \frac{P(X|H) \cdot P(H)}{P(X)} \quad (2.1)$$

Tabel 2.2 Keterangan Formulasi Teorema Bayes

X	Data dengan kelas (<i>class</i>) yang belum diketahui
H	Hipotesis data X yang merupakan suatu class yang spesifik
$P(H X)$	Probabilitas H berdasarkan data X (<i>posteriori</i>)
$P(X H)$	Probabilitas X berdasarkan kondisi H
$P(H)$	Probabilitas H (<i>prior</i>)
$P(X)$	Probabilitas X

2.2.4 Alat Ukur Evaluasi dan Validasi

Dalam penelitian ini untuk mengukur tingkat akurasi metode klasifikasi menggunakan algoritma Naive Bayes akan melibatkan metode sebagai alat ukur yaitu *Confusion Matrix* dan Kurva ROC (*Receiver Operating Characteristic*).

1. *Confusion Matrix*

Metode ini menggunakan tabel matriks seperti pada Tabel 2.3 jika suatu dataset memiliki 2 kelas, kelas yang satu dianggap kelas positif dan yang lainnya merupakan kelas negatif [15].

Tabel 2.3 Model *Confusion Matrix*

Klasifikasi	Diklasifikasikan sebagai	
	<i>Class = Yes</i>	<i>Class = No</i>
<i>Class = Yes</i>	<i>True positives</i>	<i>False negatives</i>
<i>Class = No</i>	<i>False positives</i>	<i>True negatives</i>

Keterangan dari tabel *Confusion Matrix* tersebut sebagai berikut:

- True Positives (TP) : proporsi benar dalam dataset yang dikategorikan benar.
- False Negatives (FN) : proporsi salah dalam dataset yang dikategorikan salah.
- False Positives (FP) : proporsi salah dalam dataset yang dikategorikan benar.

- d. True Negative (TN) : proporsi benar dalam dataset yang dikategorikan salah.

Berdasarkan pada keterangan tabel diatas, berikut merupakan persamaan dari model Confusion Matrix:

1. Nilai Akurasi (*acc*) merupakan proporsi dari jumlah prediksi yang benar. Berikut ini merupakan persamaannya:

$$acc = \frac{TP + TN}{TP + TN + FP + FN} \quad (2.2)$$

2. *Recall* atau *Sensitivity* yang digunakan untuk memberikan perbandingan antara proporsi TP terhadap proporsi yang benar dengan persamaan sebagai berikut:

$$Sensitivity = \frac{TP}{TP + FN} \quad (2.3)$$

3. *Specificity* biasanya digunakan sebagai perbandingan antara proporsi TN terhadap TP yang salah, berikut ini merupakan persamaannya:

$$Specificity = \frac{TN}{TN + FP} \quad (2.4)$$

4. *Precision* atau *Positive Predictive Value* (PPV) merupakan proporsi suatu kejadian dengan hasil diagnosa benar. Persamaannya sebagai berikut:

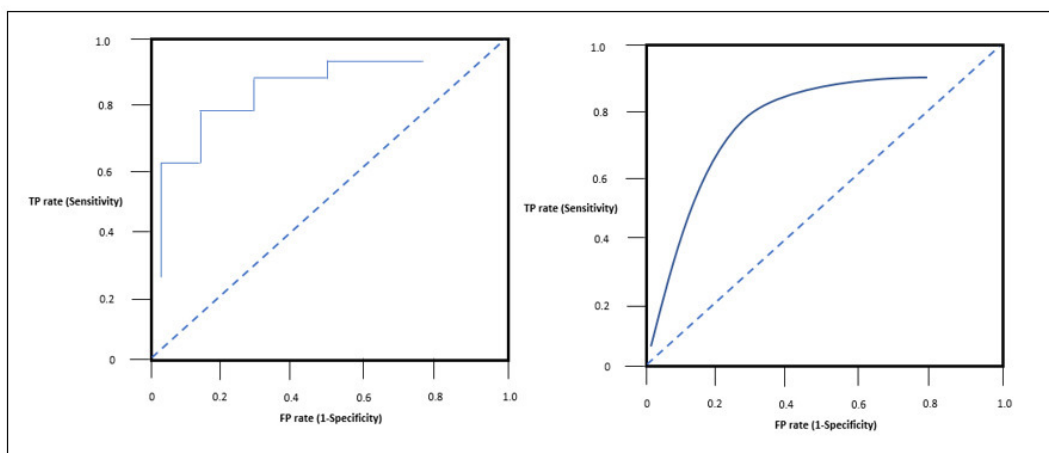
$$PPV = \frac{TP}{TP + FP} \quad (2.5)$$

5. *Negative Predictive Value* (NPV) merupakan proporsi suatu kejadian dengan hasil diagnosa salah. Persamaannya sebagai berikut:

$$NPV = \frac{TN}{TN + FN} \quad (2.6)$$

2. Kurva ROC (*Receiver Operating Characteristic*)

Kurva ROC sering digunakan untuk mengevaluasi dan menentukan hasil dari suatu prediksi atau suatu ramalan [16]. Dalam kasus klasifikasi yang melibatkan dua kelas keputusan (klasifikasi biner), masing-masing objek dipetakan ke dalam salah satu elemen dari himpunan pasangan (P,N) yaitu positif dan negatif. Beberapa model klasifikasi (pohon keputusan) yang banyak digunakan akan menghasilkan label kelas diskrit dimana hanya mengacu pada kelas yang diprediksi objeknya. Contoh pengklasifikasian lain adalah jaringan saraf dan Niave Bayes yang menghasilkan kelas berupa output kontinu.



Gambar 2.2 Kurva ROC Diskrit dan Kontinu [16]

Gambar diatas menunjukkan dua jenis kurva ROC (diskrit dan kontinu). Kurva ROC juga dikenal dengan grafik ROC yang memiliki dua dimensi grafik dimana tingkat TP diplot pada sumbu Y dan tingkat FP diplot pada sumbu X. Oleh karena itu, grafik ROC dapat digunakan sebagai gambaran *trade-off* antara mafaat relatif (*True Positive*) dan biaya (*False Positive*). Kategori tingkat akurasi dalam klasifikasi untuk pengujian prediksi adalah sebagai berikut [16]:

- a. Kategori *Excellent Classification* dengan nilai akurasi sebesar (0.90-1.00)
- b. Kategori *Good Classification* dengan nilai akurasi sebesar (0.80-0.90)
- c. Kategori *Fair Classification* dengan nilai akurasi sebesar (0.70-0.80)
- d. Kategori *Poor Classification* dengan nilai akurasi sebesar (0.60-0.70)
- e. Kategori *Failure* dengan nilai akurasi sebesar (0.50-0.60)

2.2.5 RapidMiner

RapidMiner merupakan salah satu *tools* yang bersifat *open source*, cukup populer dengan fungsi untuk memberikan solusi dalam data mining [17]. RapidMiner memberikan kemudahan untuk diimplementasikan ke dalam metode dan pendekatan baru dengan maksud akan dibandingkan dengan metode dan pendekatan lainnya. RapidMiner dapat digunakan oleh berbagai kalangan dari disiplin ilmu yang berbeda-beda seperti dalam bidang statistik, komputer, matematika, dan lain sebagainya yang berkesinambungan dengan data mining, *machine learning*, dan metode statistik. Tidak sedikit yang memerlukan alat untuk memudahkan dalam menganalisis yang fleksibel, RapidMiner menyediakan hal tersebut karena berbagai metode dari evaluasi statistik sederhana seperti analisis korelasi regresi hingga prosedur klasifikasi dan klustering dapat digunakan. Metode tersebut dapat digunakan untuk berbagai aplikasi domain seperti teks, audio, gambar dan analisis *time series*. Hasil dari analisis menggunakan RapidMiner dapat divisualisasikan dengan berbagai cara secara otomatis.

2.2.6 Hypertext PreProcessors (PHP)

PHP (*Hypertext PreProcessors*) pada mulanya merupakan singkatan dari *Personal Home Page* yang diperkenalkan oleh Rasmus Lerdorf pada 1995 adalah *script* yang dapat disisipkan dan diimplementasikan ke dalam HTML. PHP merupakan bahasa pemrograman interpreter yang bersifat *open source* dan banyak didukung oleh penyedia *web server* selain itu PHP dapat digunakan di sistem operasi BSD, Linux, dan Windows. Umumnya PHP diintegrasikan dengan aplikasi *open source database management system* seperti XAMPP yang termasuk didalamnya MySQL, Apache, FileZilla, Tomcat dan Mercury yang merupakan aplikasi *database*.

2.2.7 Kredit

Kredit merupakan suatu kapabilitas yang memungkinkan seseorang untuk memperoleh uang atau melakukan suatu pinjaman uang dengan syarat adanya perjanjian untuk membayar uang tersebut pada batas waktu yang telah disepakati

[1]. Menurut UU No. 10 tahun 1998 tentang perbankan menjelaskan bahwa kredit adalah penyediaan uang atau tagihan yang dapat dipersamakan dengan itu berdasarkan persetujuan atau kesepakatan pinjam meminjam antara bank dengan pihak lain yang mewajibkan pihak peminjam untuk melunasi hutangnya setelah jangka waktu tertentu dengan pemberian bunga [2].

2.2.8 UML


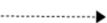




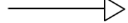
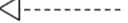
Unified Modeling Language atau disingkat UML merupakan salah satu alat bantu dalam pengembangan sistem berorientasi objek [18]. UML memungkinkan pengembang sistem untuk mencetak blueprint secara baku, mudah dipahami, serta dilengkapi dengan mekanisme efektif dalam berbagi dan berkomunikasi mengenai rancangan dengan yang lainnya. Ada 9 diagram pada UML yaitu sebagai berikut:

1. *Use-case Diagram*
2. *Class Diagram*
3. *State Diagram*
4. *Object Diagram*
5. *Activity Diagram*
6. *Sequence Diagram*
7. *Component Diagram*
8. *Collaboration Diagram*
9. *Deployment Diagram*

2.2.8.1 Use Case Diagram

Use Case Diagram adalah deskripsi dari suatu fungsi sebuah sistem yang dilihat dari sisi pengguna. Cara kerja *Use Case Diagram* adalah mendeskripsikan interaksi yang terjadi antara pengguna suatu sistem dengan sistem itu sendiri dalam bentuk sebuah skenario bagaimana sistem tersebut digunakan.



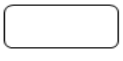

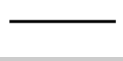
Tabel 2.4 Notasi *Use Case Diagram*

Gambar Notasi	Nama Notasi	Keterangan
	<i>Actor</i>	Spesifik peran yang pengguna gunakan pada saat berinteraksi dengan <i>use case</i> .
	<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada elemen <i>independent</i> akan mempengaruhi elemen yang bergantung pada elemen <i>independent</i> .
	<i>Generalization</i>	Hubungan dimana <i>descendent</i> berbagi struktur data dan perilaku dari objek yang ada di atasnya <i>ancestor</i> .
	<i>Association</i>	Menghubungkan antara satu objek dengan objek yang lainnya.
	<i>Use case</i>	Mendeskripsikan urutan kegiatan-kegiatan yang ditampilkan sistem dalam menghasilkan suatu hasil yang terukur bagi aktor.
	<i>Collaboration</i>	Interaksi antara elemen dan aturan-aturan lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah elemen-elemennya.
	<i>Include</i>	Mendeskripsikan <i>use case</i> yang menjadi sumber secara eksplisit.
	<i>Extend</i>	Mendeskripsikan <i>use case</i> yang menjadi target dalam memperluas perilaku dari <i>use case</i> yang menjadi sumber pada suatu titik.

2.2.8.2 Activity Diagram

Activity Diagram merupakan suatu teknik untuk mendeskripsikan prosedur, aliran kerja, dan proses bisnis dalam beberapa kasus. *Activity Diagram* disajikan dalam bentuk *flowchart* yang dapat mendukung perilaku aktor secara paralel.

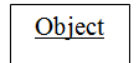

Tabel 2.5 Notasi *Activity Diagram*



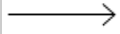
Gambar Notasi	Nama Notasi	Keterangan
	<i>Initial</i>	Titik awal dalam memulai aktivitas pada <i>activity diagram</i>
	<i>Final</i>	Titik akhir dalam mengakhiri aktivitas.
	<i>Activity</i>	Menandakan suatu aktivitas pada <i>activity diagram</i> .
	<i>Decision</i>	Pilihan dalam pengambilan suatu keputusan.
	<i>Join/Fork</i>	Menggabungkan dua kegiatan secara paralel.

2.2.8.3 *Sequence Diagram*

Sequence diagram merupakan gambaran dari suatu interaksi antar objek yang berada di dalam maupun di sekitar sistem berupa pesan yang digambarkan terhadap waktu. *Sequence diagram* memiliki 2 (dua) dimensi yaitu dimensi vertikal (waktu) dan dimensi horizontal (objek-objek yang terkait). Tujuan dari *sequence diagram* adalah untuk menggambarkan rangkaian langkah-langkah sebagai respon dari sebuah kegiatan untuk menghasilkan suatu *output* tertentu.

Tabel 2.6 Notasi *Sequence Diagram*

Gambar Notasi	Nama Notasi	Keterangan
	<i>Object</i>	<i>Instance</i> dari suatu <i>class</i> yang digambarkan secara horizontal.
	<i>Actor</i>	Peran yang berkomunikasi dengan objek.

Gambar Notasi	Nama Notasi	Keterangan
	<i>Lifeline</i>	Mengindikasikan keberadaan objek dalam basis waktu.
	<i>Activation</i>	Mengindikasikan objek yang melakukan suatu aksi.
	<i>Message</i>	Mengindikasikan komunikasi antar objek.

2.2.9 Prototype

Prototype atau prototipe merupakan pengembangan suatu sistem atau perangkat lunak (*software*) yang menggambarkan secara rinci penampilan dan karakteristik *user interface* dari suatu sistem atau *software* [19]. Salah satu keuntungan menggunakan *prototype* adalah *user* dapat berinteraksi dengan pengembang sehingga pengembang dapat mengembangkan *prototype* yang sesuai dengan kebutuhan *user* dan membutuhkan waktu yang lebih sedikit karena pengembang sudah memahami apa yang dibutuhkan *user*.

Terdapat empat langkah dalam proses pembuatan *prototype*, langkah tersebut terdiri atas :

1. Identifikasi kebutuhan dasar *user*;
2. Pengembangan awal *prototype*;
3. Penggunaan *prototype*;
4. Mengevaluasi dan memperbaiki *prototype*.

Dalam pembuatan *prototype*, keempat langkah tersebut dilakukan terus menerus hingga pelanggan atau *user* merasa puas dengan *prototype* yang dibangun. Dalam pengembangan *prototype* dikenal istilah *prototype model* atau model prototipe merupakan acuan pengembangan *prototype*, berikut ini merupakan penjelasan dari model prototipe :

1. *Communication*

Komunikasi dilakukan dengan tujuan mengidentifikasi kebutuhan yang diperlukan oleh *user*, sehingga *developer* dapat mengetahui sistem seperti apa yang diinginkan oleh *user*.

2. *Quick Plan*

Merupakan perencanaan cepat yang dilakukan dalam waktu singkat seputar bagaimana sistem yang akan dibangun, setelah dilakukan komunikasi antara *developer* dan *user*.

3. *Modelling Quick Design*

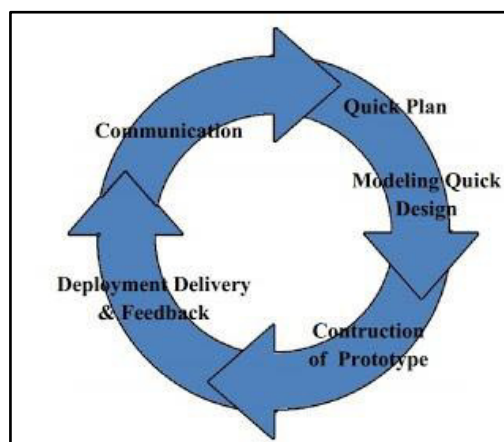
Setelah membuat perencanaan seputar sistem, maka dilakukan pembuatan rancangan permodelan sistem yang berpusat pada gambaran *software* yang dibutuhkan oleh *user*.

4. *Constuction of Prototype*

Setelah dilakukan perancangan model sistem, maka mulai dilakukan pembangunan sistem sesuai dengan rencana dan gambaran dari sistem yang telah dibuat sebelumnya.

5. *Deployment*

Setelah keempat tahapan sebelumnya dilakukan maka akan dihasilkan *software* yang kemudian akan diserahkan kepada *user* yang nantinya akan digunakan dan dievaluasi oleh *user*. Kemudian akan diperoleh *feedback* dari *user* seputar *software* yang telah dibangun dan digunakan.



Gambar 2.3 Model Prototipe