



BAB II

TEKNOLOGI INFORMASI

UNIVERSITAS DARMA PERSADA

BAB II

LANDASAN TEORI

2.1 Kajian Penelitian Terdahulu

Berikut adalah kajian dari penelitian sebelumnya yang dijadikan referensi untuk peneliti melakukan penelitian ini. (Shafarina Aprilia and Jasri, 2024) dalam penelitiannya yang berjudul “*Clustering Loyalitas Pelanggan Menggunakan Algoritma K-Means Berbasis WEB*” dalam penelitian ini mengelompokkan loyalitas pelanggan dengan menggunakan metode *K-Means*. Dalam penelitian tersebut menghasilkan pembagian menjadi 3 kelompok diantaranya sangat loyal, cukup loyal, dan tidak loyal.

(Chandra, 2023) dalam penelitiannya yang berjudul “*Perancangan Aplikasi Segmentasi Pelanggan Menggunakan K-Means dengan model RFM*” dalam penelitian tersebut menyatakan bahwa *k-means clustering* dapat mengelompokan pelanggan menjadi 3 kelompok, yaitu kelompok sangat potensial, potensial, dan tidak potensial.

(Ardana et al., 2024) dalam penelitiannya yang berjudul “*Segmentasi Pelanggan Penjualan Online Menggunakan Metode K-Means Clustering*” dalam penelitian tersebut peneliti dapat mengetahui bahwa metode *k-means* dapat mengelompokan pelanggan. Pada penelitian tersebut pelanggan dibagi menjadi 2 kelompok, yaitu *cluster 0* dengan total transaksi yang tinggi, sedangkan *cluster 1* merupakan pelanggan yang sering melakukan transaksi.

(Basri et al., 2020) dalam penelitiannya yang berjudul “*Analisis Loyalitas Pelanggan Berbasis Model Recency, Frequency, dan Monetary (RFM) dan Decision Tree Pada PT.Solo*” penelitian tersebut mengaplikasikan model *RFM*

untuk klasterisasi data pelanggan berdasarkan jumlah transaksi, nominal transaksi, waktu transaksi, dan atribut *outlet* yang dikombinasikan dengan model *Decision Tree* untuk mengidentifikasi atribut *outlet* yang mempengaruhi tingkat akurasi.

(Amandari and Sanjaya, 2022) dalam penelitiannya yang berjudul “*Penerapan Algoritma Decision Tree dalam Segmentasi Customer*” dalam penelitian tersebut menggunakan metode *decision tree* untuk mengetahui Tingkat akurasi dari algoritma tersebut ketika digunakan untuk mengklasifikasikan pelanggan kedalam beberapa segmen.

(Wardani and Ariasih, 2019) dalam penelitiannya yang berjudul “*Analisa Komparasi Algoritma Decision Tree C4.5 dan Naïve Bayes untuk prediksi Churn Berdasarkan Kelas Pelanggan Retail*” dalam penelitian tersebut menggunakan model *RFM* digunakan untuk mengklasifikasikan pelanggan, dan masing-masing pelanggan dapat dianalisis atau diprediksi menggunakan model *Decision Tree C.45*.

2.2 Loyalitas Customer

Dilansir dari jurnal (Prasetya and Utari, 2022) Loyalitas dalam konteks ini berarti saat pelanggan mengeluarkan banyak uang untuk membeli produk dari perusahaan, dengan kata lain, pelanggan tidak mempermasalahkan besarnya biaya yang dikeluarkan. Maka dari itu loyalitas *customer* dalam suatu perusahaan itu sangat penting untuk diperhatikan.

Pada dasarnya, pelanggan yang setia membeli produk dari perusahaan akan secara tidak langsung membantu pemasaran dengan merekomendasikan produk tersebut terhadap orang-orang terdekatnya. Hal ini membuat produk perusahaan menjadi prioritas di hati pelanggan yang setia. Namun apabila suatu perusahaan

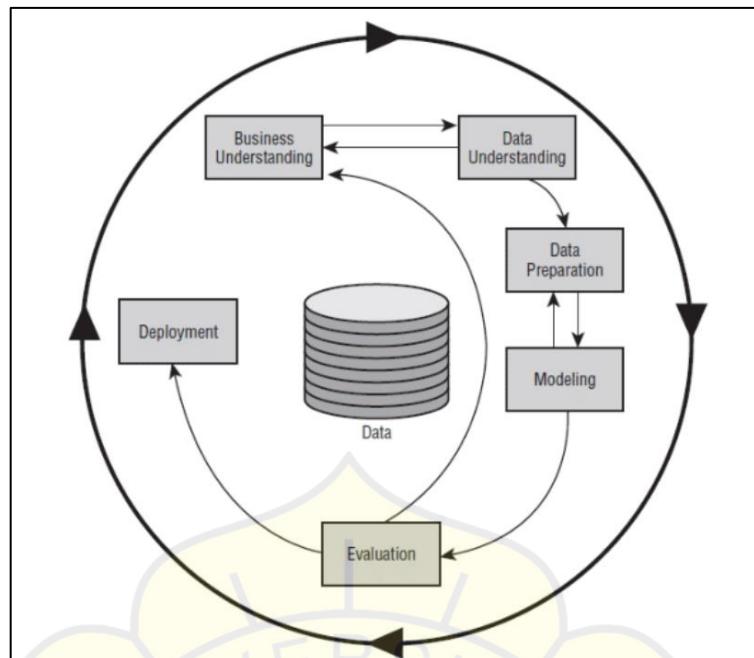
tidak dapat memperhatikan pelanggannya, besar kemungkinan pelanggan akan beralih hati ke perusahaan pesaing.

2.3 Data Mining

Dilansir dari jurnal (Gunia et al., 2024) *Data mining* ialah suatu metode yang digunakan untuk menganalisis sejumlah besar data dengan tujuan menemukan keterkaitan antar data dan menyajikannya dalam format yang mudah dipahami. Maka dapat disimpulkan data mining adalah suatu metode yang sangat baik digunakan untuk menganalisis suatu data pada suatu perusahaan, dan dapat pula membantu memberikan rekomendasi keputusan untuk strategi perusahaan agar dapat berkembang di masa mendatang.

2.3.1 Metode Pengolahan Data (*Cristp-DM*)

CRISP- DM (*Cross Industry Standard Process for Data Mining*) adalah pemrosesan data mining yang dirancang untuk memastikan data melewati setiap fase yang terstruktur, terdefinisi dengan jelas, dan efisien (Hasanah et al., 2021). Proses ini terdiri dari enam tahap, diantaranya adalah *Business Understanding*, *Data Understanding*, *Data Preparation*, *Modelling*, *Evaluation*, dan *Deployment*. Penjelasan lebih lanjut akan dijelaskan sebagai berikut.



Gambar 2. 1 CRISP DM (Ansori dan Wulandari, 2022)

Pada gambar 2.1 merupakan visualisasi dari proses *crisp-dm* yang meliputi *business understanding*, *data understanding*, *data preparation*, *modelling*, *evaluation* dan *deployment*.

1. *Business Understanding* (Pemahaman Bisnis)

Tahap ini melibatkan pemahaman kebutuhan dan tujuan dari perspektif bisnis, kemudian menerjemahkannya ke dalam definisi masalah dalam data mining, serta menentukan rencana dan strategi untuk mencapai tujuan data mining tersebut.

2. *Data Understanding* (Pemahaman Data)

Pada tahap ini diawali dengan pengumpulan data, pendeskripsi data dan mengevaluasi kualitas data.

3. *Data Preparation* (Persiapan Data)

Tahap ini melibatkan pembentukan dataset akhir dari dataset mentah. Langkah-langkah yang perlu dilakukan mencakup pembersihan data (*Data Cleaning*),

pemilihan data (*Data Selection*), serta transformasi data (*Data Transformaton*) untuk digunakan dalam tahap pemodelan.

4. *Modelling* (Pemodelan)

Tahap pemodelan ini secara langsung melibatkan *Machine Learning* untuk menentukan teknik *data mining*, alat bantu *data mining*, dan algoritma *data mining*.

5. *Evaluation* (Pengujian)

Tahapan ini mencakup analisis hasil *data mining* yang diperoleh dari proses pemodelan sebelumnya. Evaluasi dilakukan untuk memastikan bahwa model yang dipilih selaras dengan tujuan yang telah ditetapkan pada tahap awal.

6. *Deployment* (Penyebaran)

Tahap ini melibatkan implementasi solusi data mining yang sudah dikembangkan kedalam lingkungan operasional atau produksi. Pada tahap ini hasil dari data mining yang dikembangkan pada tahap sebelumnya akan di integrasikan ke dalam sistem. Tujuannya adalah agar hasil dapat digunakan secara efisien untuk mendukung pengambilan keputusan.

2.3.2 Perbandingan Metode *Clustering* dan Klasifikasi

Dilansir dari jurnal (Valentino et al., 2017) dari penelitian bertajuk “Komparasi Metode *Decision Tree* dan *K-Means Clustering* Dalam Mengatasi Masalah *Cold-start* Pengguna Baru” menjelaskan bahwa pada penelitian tersebut mengevaluasi efektivitas pengelompokan dalam sistem rekomendasi menggunakan dataset *Movielens 100k* dengan memanfaatkan algoritma *K-Means Clustering* dan *Decision Tree*. Penelitian ini juga menyimpulkan bahwa pengelompokan berdasarkan demografi usia pengguna efektif dalam mengatasi masalah *cold-start*

pada pengguna baru dan merekomendasikan penelitian lebih lanjut dengan metode pengelompokan lainnya.

(Pratiwi, 2016) meneliti tentang perbandingan *K-Means Clustering*, *Decision Tree*, dan *Naïve Bayes*. Ketiga metode tersebut dibandingkan untuk memperoleh pengelompokan siswa secara otomatis agar siswa dapat melakukan pembelajaran aktif yang memotivasi siswa agar belajar bersama dalam kelompok kecil.

(Saputra, 2021) meneliti tentang perbandingan metode *clustering* yaitu *Fuzzy C-Means* dan metode Klasifikasi *Naïve Bayes* yang mana keduanya dibandingkan dalam konteks penentuan pihak yang menerima manfaat berdasarkan data sosial ekonomi. Penelitian tersebut bertujuan untuk mengetahui hasil *clustering* dan hasil dari klasifikasi serta Tingkat akurasi penerapan kedua algoritma tersebut dalam penentuan penerima bantuan sosial di Kabupaten Buleleng. Perbandingan tersebut dilakukan berdasarkan nilai akurasi yang dihitung menggunakan *confusion matrix*.

2.3.3 K-Means Clustering

(Hidayat et al., 2021) *K-Means* ialah metode iterasi yang sederhana untuk membagi data kedalam sejumlah klaster yang diatur oleh pengguna. Tujuan dari klasterisasi data ini adalah untuk mengurangi fungsi objektif yang ditetapkan dalam proses pengelompokan, dengan upaya untuk mengurangi variasi dalam suatu klaster dan meningkatkan variasi antar klaster.

Algoritma *K-Means* menggunakan analisis pengelompokan dengan klaster non-hirarkis. Metode ini dimulai dengan menentukan klaster yang diinginkan, yang bisa terdiri dari dua, tiga, atau lebih agar terbentuk menjadi suatu klaster.

Menurut (Ardi et al., 2023) *K-Means* adalah algoritma *unsupervised learning* yang digunakan untuk mengelompokkan data dalam sejumlah *cluster* berdasarkan kemiripan antar data.

Untuk melakukan metode *K-Means* secara umum dilakukan menggunakan algoritma dasar sebagai berikut. (Sipayung. et al., 2020)

1. Menentukan jumlah klaster (k) pada dataset.
2. Menentukan nilai pusat (centroid). Untuk mencari nilai pusat di awal tahap, caranya adalah mencari nilai secara acak, rumus menentukan target awal *k-means*, rumus ini digunakan untuk menemukan nilai target data atau jarak antar klaster, yang berfungsi sebagai titik pusat awal dalam perhitungan algoritma *k-means* iterasi 0, seperti perumusan berikut.

$$\frac{\text{Jumlah data}}{\text{Jumlah class} + 1}$$

Keterangan :

Jumlah data = Jumlah keseluruhan data yang dipergunakan.

Jumlah class = Jumlah kelompok yang telah ditentukan sebelumnya seperti tinggi dan rendah.

Rumus rata-rata digunakan untuk mencari nilai iterasi. Perhitungan pada rumus tersebut bertujuan untuk menemukan nilai rata-rata seperti yang tertera pada persamaan berikut.

$$V_{ij} = \frac{1}{N_i} \sum_{k=0}^{N_i} X_{kj}$$

Keterangan :

V_{ij} = Centroid rata-rata klaster ke-i untuk variabel ke -j.

n_i = Jumlah anggota klaster ke – i.

i,k = Indeks dari kalster.

j = indeks dari variabel.

X_{kj} = Nilai data ke-k variabel ke-j untuk klaster tersebut.

3. Pada setiap record, hitunglah jarak terdekat dengan centroid. Rumus yang digunakan adalah *euclidean distance*, seperti persamaan berikut.

$$De = \sqrt{(xi - si)^2 + (yi - ti)^2}$$

Keterangan :

De = *Euclidean Distance*.

i= Banyaknya objek.

(x,y) = Koordinat objek.

(s,t) = Koordinat centroid.

4. Kelompokkan objek berdasarkan jarak ke centroid yang paling dekat.

Lakukan iterasi secara berulang dari langkah ke-2 sampai data tetap berada di klaster yang sama.

2.3.4 Decision Tree CART

(Bulkisah et al., 2024) *Decision tree* adalah algoritma yang dipergunakan untuk memprediksi suatu kelas. metode ini bekerja dengan membagi data menjadi kelompok berukuran lebih kecil. Dan algoritma *Decision Tree* dapat mengolah atribut yang bersifat diskrit ataupun numerik.

(Erliani et al., 2023) *Classification and regression tree (CART)* adalah suatu metode yang termasuk dalam teknik eksplorasi data, yaitu teknik pohon keputusan. *CART* bertujuan untuk mendapatkan suatu kelompok data yang akurat sebagai penciri dari suatu pengklasifikasian, selain itu *CART* digunakan untuk menjelaskan hubungan antar *variable response* dengan satu atau beberapa *variable predictor*.

(Oktavirahani and Maharesi, 2022) *CART* merupakan metode statistic nonparametric yang digunakan untuk klasifikasi dan regresi. Metode ini menghasilkan pohon keputusan yang menjelaskan kaitan antara variable respon (dependen) dengan satu atau lebih variable predictor (independen). Dalam konteks klasifikasi, *CART* menciptakan pohon klasifikasi jika *variable response* memiliki skala kategorik, dan menghasilkan pohon regresi jika *variable response* dalam bentuk data kontinu.

Berikut adalah tahapan dari *Decision Tree CART*. (Irmanita et al., 2021)

1. Menghitung *Gini Index* untuk setiap subset.

Hitung *gini index* di setiap subset dengan perumusan berikut.

$$Gini(D) = 1 - \sum_{i=1}^m P_i^2$$

Keterangan :

$G(D)$ = *Gini Index*.

k = Jumlah kelas.

P_i^2 = Probabilitas sebuah data dalam dataset D termasuk dalam kelas i terhadap *impurity*.

2. Menghitung *Gini Index* untuk setiap atribut.

Setelah mendapatkan probabilitas dari setiap partisi dari setiap atribut, kemudian hitung *gini index* untuk setiap atribut.

$$Gini_A(D) = \frac{|D_1|}{|D|} Gini(D_1) + \frac{|D_2|}{|D|} Gini(D_2)$$

Keterangan :

$\frac{|D_1|}{|D|}$ = Proporsi ukuran subset D1 terhadap total D.

$\frac{|D_2|}{|D|}$ = Proporsi ukuran subset D2 terhadap total D.

$G(D_1)$ dan $G(D_2) = Gini$ index dari masing masing subset setelah pemisahan.

$Gini_A(D) = Gini$ indeks dari suatu atribut.

3. Memilih Root Node

Setelah diketahui hasil dari *gini index* dari tiap atribut, pilih fitur dengan hasil terendah untuk dijadikan sebagai *root node*. Selanjutnya proses diulang untuk setiap sub-dataset yang dihasilkan dari pemisahan, hingga mencapai *leaf node* (semua sampel dalam node memiliki label yang sama atau tidak ada fitur yang tersisa untuk dipisahkan).

2.3.5 Confusion Matrix

(Antika et al., 2024) *Confusion matrix* adalah metode evaluasi yang digunakan untuk mengukur kinerja algoritma *Machine Learning*. Metode ini bekerja dengan cara membandingkan prediksi yang dibuat oleh algoritma dengan data aktual yang telah diketahui sebelumnya. Metode tersebut memberikan visualisasi yang jelas tentang sejauh mana algoritma dapat memprediksi setiap kelas atau label dalam dataset dengan tepat, serta membantu mengidentifikasi berbagai metrik evaluasi seperti akurasi.

Tabel 2. 1 Confusion Matrix

Kelas aktual	Kelas Prediksi	
	True	False

	<i>True</i>	TP	FP
	<i>False</i>	FN	TN

Tabel 2.1 merupakan metrik yang digunakan sebagai alat evaluasi dalam mengklasifikasikan suatu algoritma. Metrik tersebut terdiri dari empat elemen utama, diantaranya *True Positive* (TP), *True Negative* (TN), *False Positive* (FP), *False Negative* (FN). Berikut penjelasan dari empat elemen tersebut:

TP = Jumlah nilai positif yang terprediksi dengan benar sebagai positif.

FP = Jumlah nilai negative yang salah di prediksi sebagai positif.

TN = Jumlah data dengan nilai negative yang terprediksi dengan benar selaku nilai negative.

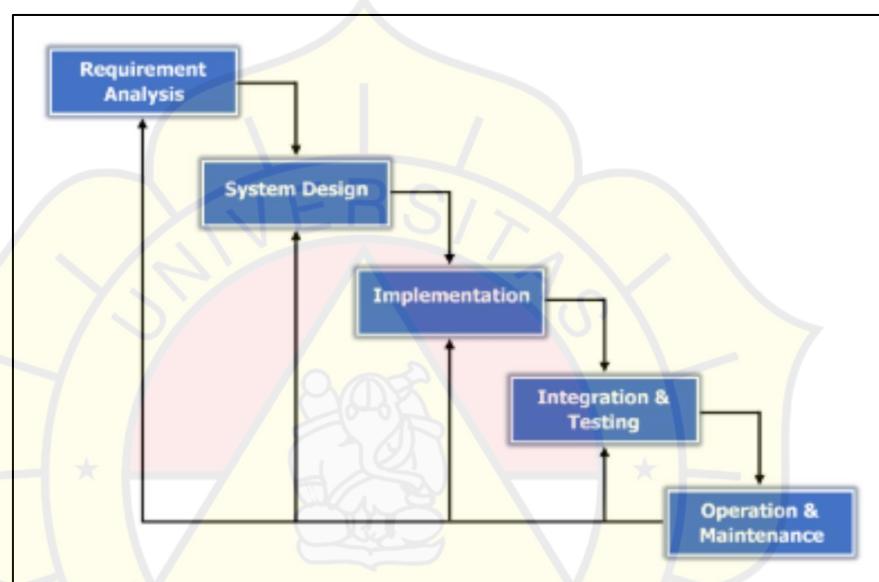
FN = jumlah nilai positif yang salah terprediksi sebagai nilai negatif.

Akurasi adalah ukuran yang menunjukkan seberapa dekat nilai prediksi sebuah model dengan nilai aktual atau sebenarnya dari data yang diamati. Ukuran ini mencerminkan kemampuan suatu model untuk mengidentifikasi kelas atau nilai yang tepat serta akurat dalam suatu dataset. Akurasi memberikan gambaran tentang tingkat keakuratan model dalam membuat prediksi dengan membandingkan prediksi yang dihasilkan dengan fakta yang sebenarnya dalam suatu data. Berikut rumusnya :

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

2.4 Metodologi Pengembangan Sistem (*Waterfall*)

Metode *waterfall* digunakan untuk merancang dan mengembangkan sistem analisis yang akan peneliti bangun. Metode ini merupakan pendekatan yang sistematis dan berurutan dalam pengembangan perangkat lunak (Ula, 2021). Pendekatan ini melibatkan lima tahap dalam pengembangan aplikasi sistem disposisi, yaitu *requirement definition, system and software design, implementation and unit testing, integration and system testing, dan operation and maintenance.*



Gambar 2. 2 Metode Perancangan Sistem

Pada gambar 2.2 merupakan visualisasi dari metode *waterfall* yang tahapannya meliputi *requirement analyst, system design, implementation, integration and unit testing, dan operation and maintenance.*

Metode *Waterfall* memiliki tahapan-tahapan sebagai berikut :

1. Requirements Analysis

Proses pengumpulan kebutuhan dilakukan secara matang untuk menyesuaikan kebutuhan perangkat lunak yang harus dianalisis agar dapat memahami jenis perangkat lunak yang dibutuhkan *user*.

2. System Designs

Dalam tahap ini, dimanfaatkan untuk merancang fitur pada sistem baru yang sedang dikembangkan.

3. Implementation

Pembuatan kode program merupakan tahap terjemah *program code* untuk desain sistem yang sudah dibuat kedalam perintah perintah yang dimengerti komputer.

4. Integration and system Testing

Pada pengujian program, seluruh unit yang dikembangkan dalam tahap implementasi digabungkan atau di integrasikan guna diuji kesesuaian program dengan kebutuhan dan tidak ada kesalahan.

5. Operation and maintenance

Tidak menutup kemungkinan sebuah program yang sudah selesai dibuat akan timbul kesalahan – kesalahan yang tidak ditemukan pada langkah sebelumnya. Oleh karena itu, setelah program berhasil dibuat perlu dilakukan pemeliharaan dengan baik untuk menanggulangi kesalahan – kesalahan yang tidak ditemukan tersebut.

2.5 Website

Menurut (Suheri et al., 2023) *Website* merupakan Kumpulan halaman-halaman yang diperuntukan untuk menampilkan informasi dalam bentuk teks, gambar, animasi, suara, atau kombinasi dari semuanya. Halaman-halaman ini dapat bersifat statis atau dinamis dan saling terhubung, membentuk suatu rangkaian struktur yang terkait melalui jaringan halaman.

2.6 Bahasa Pemrograman

Berikut beberapa bahasa pemrograman yang peneliti pakai.

2.6.1 HTML(Hypertext Markup Language)

Dilansir dari jurnal (Agung et al., 2022) HTML merupakan bahasa *markup* yang digunakan untuk membuat halaman web. Bahasa ini terdiri dari kombinasi teks dan simbol yang membentuk struktur sebuah *website* dan disimpan kedalam file. Pembuatan file HTML mengikuti standar internasional atau ASCII (*American Standard Code For Information Interchange*).

2.6.2 PHP (Perl Hypertext Preprocessor)

Menurut (Agung et al., 2022) PHP adalah Bahasa pemrograman yang mirip dengan *javascript* dan *python*. PHP biasanya digunakan untuk komunikasi sisi server, sedangkan *javascript* dapat digunakan untuk *frontend* dan *backend*. Sementara itu, *phyton* umumnya digunakan untuk *backend*.

2.6.3 CSS (Cascading Style Sheet)

(Malioy et al., 2023) CSS (*Cascading Style Sheet*) merupakan bahasa desain web yang digunakan untuk mengatur tampilan dokumen yang dibuat dengan *markup language*. CSS berguna untuk mengatur komponen pada halaman web, mengubah halaman menjadi standar yang lebih menarik dan estetis.

2.6.4 Javascript

Menurut (Fernando et al., 2023) *JavaScript* berfokus pada pengolahan data di sisi klien dan menyajikan komponen web yang lebih interaktif, menambah fungsionalitas dan kenyamanan halaman web. *JavaScript* merupakan bahasa skrip dinamis yang digunakan untuk membangun interaktivitas pada halaman HTML statis dengan menempatkan blok-blok kode *JavaScript* di hamper semua tempat pada halaman web.

2.7 Basis Data

Dilansir dari jurnal (Fernando et al., 2023) basis data atau *database* adalah media penyimpanan yang dapat menampung berbagai macam data, dengan sistem pendataan yang dikelola menggunakan manajemen data yang baik.

2.7.1 MySQL

Menurut (Siska et al., 2023) *MySQL* adalah sistem *database* relasional yang mengelompokkan informasi ke dalam tabel-tabel terkait. Setiap tabel memiliki bidang-bidang terpisah yang mewakili setiap bit informasi. *MySQL* menggunakan indeks untuk mempercepat pencarian baris informasi tertentu dan memerlukan setidaknya satu indeks per tabel. Biasanya, *primary key* atau pengenal unik digunakan untuk membantu penjejakkan data.

2.8 Alat Bantu Analisis Dan Perancangan Sistem

Berikut adalah beberapa alat bantu analisis dan perancangan sistem.

2.8.1 UML (*Unified Modelling Language*)

(Wira et al., 2019) UML merupakan salah satu standar bahasa yang banyak digunakan di industri untuk mendefinisikan *requirement*, melakukan analisis dan desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek adalah UML (*Unified Modelling Language*). Berikut tahapan dari pemodelan dalam UML.

1. *Usecase Diagram*

Usecase diagram adalah pemodelan untuk perilaku sistem informasi yang akan dibuat. *Usecase* mendeskripsikan interaksi tipikal antara pengguna sistem dan sistem itu sendiri melalui cerita tentang bagaimana sistem tersebut digunakan.

2. Sequence Diagram

Sequence diagram menjelaskan perilaku objek dalam *usecase* dengan mendeskripsikan waktu hidup objek serta pesan yang dikirim dan diterima antar objek. Diagram ini dibuat minimal sesuai jumlah *usecase* yang memiliki proses sendiri. Semua interaksi pesan dalam *usecase* harus dicakup dalam *sequence diagram*. Semakin banyak *usecase* yang didefinisikan, semakin banyak *sequence diagram* yang perlu dibuat.

3. Activity Diagram

Activity diagram adalah yang menunjukkan alur kerja atau aktivitas dari suatu sistem dalam perangkat lunak.