

BAB II

LANDASAN TEORI

2.1 Tinjauan Pustaka

Tinjauan pustaka salah satu bagian terpenting yang tidak bisa dipisahkan dari sebuah penelitian yang dapat digunakan untuk mendukung penelitian yang sedang dilakukan saat ini. Tinjauan pustaka ini memuat ulasan dan analisis terhadap berbagai literatur terkait yang telah dipublikasi sebelumnya. yang perlu diingat bahwa tinjauan pustaka bukan hanya daftar pustaka yang sekedar mendeskripsikan satu per satu publikasi atau penelitian yang sudah ada sebelumnya. Lebih dari itu tinjauan pustaka harus mampu memberikan ulasan kritis terhadap berbagai literatur tersebut sehingga dapat memberikan pemantapan dan penegasan tentang ciri khas penelitian yang hendak di kerjakan. Penulis telah mengumpulkan beberapa tinjauan pustaka yang dapat dilihat pada tabel 2.1

Tabel 2.1 Tinjauan Pustaka

No	Nama Peneliti	Keterangan
1	Rusnandi et al., 2020	<p>Judul: Penerapan Data Mining Untuk Analisis Market Basket Dengan AMKoritme FP-Growth Pada PD Pasar Tohaga</p> <p>Masalah Penelitian: Hasil dari data penjualan di 3 toko yang berbeda pada PD Pasar Tohaga yang selalu berupa buku transaksi hanya dilihat tanpa adanya tindak lanjut untuk menentukan keputusan di waktu yang akan datang</p> <p>Metode: <i>Frequent Pattern Growth, Association Rule</i></p>

No	Nama Peneliti	Keterangan
		<p><i>dan Algoritma Apriori</i></p> <p>Tools: <i>Software RapidMiner</i></p> <p>Hasil: Konsumen banyak membeli produk dengan merk converse dan cenderung akan membeli produk adidas. selain itu konsumen juga banyak menyukai produk dengan merek carvil dan cenderung akan membeli produk rafila</p>
2	Lestari, 2017	<p>Judul: Penerapan Data Mining Algoritma Apriori Dalam Sistem Informasi Penjualan</p> <p>Masalah Penelitian: Mempermudah kegiatan operasionalnya, baik dalam penginputan dan pengolahan data, hingga menyediakan laporan maupun informasi penting lainnya</p> <p>Metode: <i>Algoritma Apriori Dan Market Basket Analisis</i></p> <p>Tools: <i>PHP dan Python</i></p> <p>Hasil: Dengan adanya data mining dengan Algoritma apriori, dapat dilihat kecenderungan konsumen dalam membeli barang, dapat diketahui pola penjualan yang digunakan untuk menganalisa pangsa pasar</p>
3	Tana et al., 2018	<p>Judul:</p>

No	Nama Peneliti	Keterangan
		<p>Penerapan Metode Data Mining Market Basket Analisis Terhadap Data Penjualan Produk Pada Toko Oase Menggunakan Algoritma Apriori</p> <p>Masalah Penelitian: Pengumpulan data dan pemakaian data yang lama untuk menemukan keteraturan, pola atau hubungan dalam suatu data</p> <p>Metode: <i>Algoritma Apriori pada teknik Data Mining</i></p> <p>Tools: <i>Visual Studio</i></p> <p>Hasil: Sangat efisien dan dapat mempercepat proses pembentukan kecenderungan pola kombinasi itemset hasil penjualan Produk-produk barang di Toko OASE, yaitu dengan support dan confidence tertinggi adalah Rokok, kopi Snack, mie goreng, Nabati</p>
4	Rahmawati et al., 2017	<p>Judul: Aplikasi Data Mining Market Basket Analisis untuk Menemukan Pola Pembelian di Toko Metro Utama Balikpapan</p> <p>Masalah Penelitian: Market basket analisis dimanfaatkan oleh perusahaan retail seperti toko atau swalayan untuk mengembangkan strategi pemasaran dengan melihat item-item mana saja yang kemungkinan dibeli secara bersamaan oleh konsumen</p> <p>Metode: <i>Algoritma Apriori dan frequent pattern</i></p>

No	Nama Peneliti	Keterangan
		<p><i>growth (FP-growth).</i></p> <p>Tools: <i>Software R.2.11.1.</i></p> <p>Hasil: Dihasilkan pada 24 Desember 2015 sampai dengan 30 Desember 2015 dapat disimpulkan bahwa pada tanggal 24 Desember 2015 dari 213 transaksi, terdapat 1 hukum asosiasi, pada tanggal 25 Desember 2015 dari 222 transaksi, terdapat 4 hukum asosiasi, pada tanggal 26 Desember 2015 dari 194 transaksi, terdapat 1 hukum asosiasi, pada tanggal 27 Desember 2015 dari 213 transaksi, terdapat 4 hukum asosiasi, pada tanggal 28 Desember 2015 dari 229 transaksi, terdapat 6 hukum asosiasi, pada tanggal 29 Desember 2015 dari 221 transaksi, terdapat 9 hukum asosiasi dan pada tanggal 30 Desember 2015 dari 242 transaksi, terdapat 4 hukum asosiasi</p>
5	Islami et al., 2020	<p>Judul: Penerapan Metode Asosiasi Datamining Market Basket Analisis Menggunakan Algoritma Apriori Untuk Melakukan Pola Analisis Belanja Konsumen</p> <p>Masalah Penelitian: kurangnya persediaan stok barang hal inilah yang dapat menyebabkan kekecewaan bagi</p>

No	Nama Peneliti	Keterangan
		konsumen Metode: <i>Algoritma Apriori dan Market Basket Analisis (MBA)</i> Tools: Weka 3.6, Rapid Miner Hasil: Aplikasi Data Mining dirancang untuk dapat memudahkan dalam menyelesaikan tugas Data Mining dengan menggunakan Algoritma apriori association rules untuk mencari pola analisis belanja antar barang yang paling sering terjual pada Gerai Bukit Nusa Indah

2.2 Data Mining

Data Mining adalah proses untuk menggali sekumpulan data dan mengubahnya dalam bentuk informasi yang bermanfaat bagi pengguna diantaranya seperti clustering, classification, association, dan beberapa perkembangan teknik sesuai dengan perubahan kecendrungan data pada saat ini.

Data mining merupakan langkah dalam proses KDD yang terdiri dari penerapan analisis data dan Algoritma penemuan yang menghasilkan penghitungan pola atau model tertentu melalui data. Didalam jurnal Febriani et al., 2018 di dalam penelitiannya menjelaskan bahwa terdapat beberapa langkah di dalam proses KDD (Knowledge Discovery in Database) diantaranya secara berurutan selection, preprocessing, transformation, DATA MINING, dan Interpretation/evaluation. Tugas-tugas data mining dapat dikelompokkan ke dalam enam kelompok berikut ini:

1. Klasifikasi (classification): men-generalisasi struktur yang diketahui untuk diaplikasikan pada data-data baru. Misalkan, klasifikasi penyakit ke dalam sejumlah jenis, klasifikasi email ke dalam spam atau bukan.
2. Klasterisasi (clustering): mengelompokkan data, yang tidak diketahui label kelasnya, ke dalam sejumlah kelompok tertentu sesuai dengan ukuran kemiripannya.
3. Regresi (regression): menemukan suatu fungsi yang memodelkan data dengan galat (kesalahan prediksi) seminimal mungkin.
4. Deteksi anomali (anomaly detection): mengidentifikasi data yang tidak umum, bisa berupa outlier, perubahan atau deviasi yang mungkin sangat penting dan perlu investigasi lebih lanjut.
5. Pembelajaran aturan asosiasi (association rule mining) atau pemodelan kebergantungan (dependency modeling): mencari relasi antar variabel.
6. Perangkuman (summarization): menyediakan representasi data yang lebih sederhana, meliputi visualisasi dan pembuatan laporan.

Selain itu, beberapa pengaplikasian data mining dapat digunakan dalam berbagai bidang, diantaranya adalah:

1. Marketing dan Bisnis Perusahaan selalu memiliki data-data yang dapat dimanfaatkan dalam strategi marketing dan bisnis. Seperti melakukan pemilihan vendor yang tepat, penggunaan strategi penjualn produk, serta penggunaan aturan dalam kartu loyalti perusahaan. Berikut ini contoh aplikasi data mining dalam marketing dan bisnis:
 - a. Market Basket Analisis Analisis keranjang belanja yang memungkinkan dapat mengetahui kebiasaan belanja konsumen. MBA dikenal juga dengan

association rule (aturan asosiasi) yaitu salah satu konsep dalam data mining yang berusaha menemukan asosiasi atau keterkaitan data.

- b. Recommender System Merupakan sistem rekomendasi dari beberapa variabel sehingga dapat dilakukan pemilihan lebih tepat seperti dalam pemilihan rekomendasi supplier mana yang menunjukkan performansi baik. Aplikasi ini biasanya menggunakan teknik kalsterisasi ataupun klasifikasi.
 - c. Churn Prediction Merupakan analisis dari loyal apa tidaknya suatu pelanggan berdasarkan variabel-variabelnya. Sebagai contoh perusahaan telokomunikasi yang memiliki pelanggan hampir ratusan juta ingin melihat pelanggan apakah tetap loyal apa tidak, dengan menggunakan teknik data mining sehingga hal tersebut menjadi mudah dan cepat dilakukan. Teknik yang biasa digunakan adalah teknik klasifikasi dan kalsterisasi.
 - d. Fraud Detection Fraud detection digunakan dalam menemukan pelanggan yang mungkin melakukan kecurangan. Sejumlah data yang besar apabila dilakukan secara manual akan membutuhkan biaya dan waktu yang lama sehingga penggunaan teknik data mining dapat mempercepat dalam penemuan kecurangan di dalam suatu basis data pelanggan. Sistem ini dapat dibangun menggunakan teknik anomaly detection.
2. Sains dan Teknik Beberapa teknik data mining dapat digunakan dalam dunia sains dan teknik untuk menyelesaikan permasalahan yang kompleks, seperti genetika, medis, teknik elektro, dan sebagainya.
 3. Seni dan Hiburan Data mining juga dapat diaplikasikan ke dalam seni dan hiburan, seperti menentukan lagu kesukaan yang sering kali diputar ataupun merekomendasikan jenis lagu ataupun video yang memiliki kemiripan yang sama dengan lagu atau video favorit

2.2.1 Algoritma Apriori

Aturan Asosiasi atau disebut dengan association rule merupakan salah satu teknik data mining yang berguna dalam mencari aturan asosiatif dari suatu barang. Aturan Asosiasi dibentuk dengan cara menganalisis pola data yang sering muncul (frequent pattern) dan dengan menggunakan parameter support dan confidence untuk mengidentifikasi hubungan yang paling penting. Support adalah indikasi yang menunjukkan seberapa sering suatu item muncul di dalam database. Sedangkan Confidence menunjukkan berapa kali pernyataan tersebut tersebut benar berdasarkan jurnal (Edusainstek et al., 2018).

1. Support Support adalah suatu ukuran yang menunjukkan seberapa besar tingkat dominasi suatu item/item set dari keseluruhan transaksi. Ukuran ini akan menentukan apakah suatu item/item set layak untuk dicari confidece selanjutnya.

$$Support = P(X \cap Y) = \frac{\sum \text{transaksi yang mengandung } X \text{ dan } Y}{\text{Jumlah transaksi}}$$

2. Confidence Confidence adalah suatu ukuran yang menunjukkan hubungan antar 2 item secara conditional seperti seberapa sering item Y dibeli jika orang membeli X.

$$Confidence = P = \left(\frac{Y}{X}\right) = \frac{P(X \cap Y)}{\sum \text{Transaksi yang mengandung } X}$$

3. Lift Ratio Lift Ratio adalah salah salah satu cara yang baik untuk melihat kuat atau tidaknya aturan asosiasi yang terbentuk. Lift Ratio juga merupakan nilai yang menunjukkan kevalidan proses transaksi dan memberikan informasi apakah benar produk X dibeli bersamaan dengan produk Y. Nilai Lift mengukur kemungkinan X dan Y terjadi bersama-sama dibagi dengan kemungkinan X dan Y terjadi jika mereka adalah peristiwa independen.

$$Left\ Ratio = \frac{Support(X \cap Y)}{Support(X) \cdot Support(Y)}$$

Aturan asosiasi (Association rule) biasanya diperlukan untuk memenuhi minimum support yang ditentukan user dan minimum confidence yang ditentukan pengguna pada saat yang sama. Pembuatan aturan asosiasi biasanya dibagi menjadi dua langkah terpisah (Dhanalakshmi & Porkodi, 2017):

1. Ambang batas minimum support diterapkan untuk menemukan semua frequent itemset di dalam database.
2. Batasan minimum confidence diterapkan pada frequent itemset digunakan untuk membentuk aturan.

Di dalam (Dhanalakshmi & Porkodi, 2017) menuliskan ada dua langkah di dalam menggunakan Algoritma Apriori, yaitu langkah join dan prune.

1. Join Step: Untuk mencari suatu itemset yang baru (L_k), beberapa kandidat dari kitemset dibentuk oleh penggabungan itemset sebelumnya (L_{k-1}). Pada proses ini setiap item dikombinasikan dengan item yang lainnya sampai tidak terbentuk kombinasi lagi.
2. Prune Step : Proses ini adalah proses pemangkasan itemset yang telah dibentuk sebelumnya berdasarkan dengan minimum support yang ditentukan. Sehingga menghasilkan suatu itemset yang sesuai dengan keinginan oleh user Sedangkan di dalam (Dhanalakshmi & Porkodi, 2017) Algoritma Apriori memiliki beberapa langkah dalam pencarian aturan asosiasi. Langkah ini hampir sama dengan sebelumnya, hanya saja pendefinisian dalam langkah ini yang berbeda:
 - a. Pembentukan Kandidat Itemset : pemangkasan kandidat k-itemset yang subsetnya berisi k-1 item. Dimana kandidat itemset dibentuk dari kombinasi ($k-1$)-itemset yang berasal dari iterasi sebelumnya.

- b. Perhitungan support dari tiap kandidat k-itemset.
- b. Menetapkan pola frekuensi tinggi

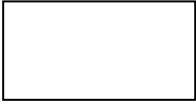

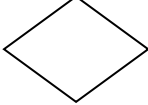



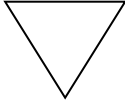




Bila tidak didapat pola frekuensi tinggi maka seluruh proses dihentikan..



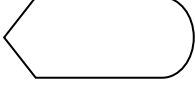

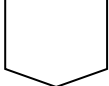
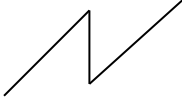

2.3 Bagan Alir Dokumen

Bagan Alir adalah salah satu alat dan teknik yang digunakan dalam pengembangan sistem informasi, bagan alir dokumen merupakan bagan alir yang menunjukkan arus dari laporan dan formulir termasuk tembusannya (Shalahuddin, 2016).

Simbol yang digunakan dapat dilihat pada tabel 2.2 sebagai berikut:

Tabel 2.2 Simbol-simbol dalam Bagan Alir

Simbol	Keterangan
	Simbol ini digunakan untuk menyatakan sambungan dari satu proses ke proses lainnya dalam halaman/lembar yang berbeda.
	Simbol ini digunakan untuk menyatakan suatu tindakan (proses) yang tidak dilakukan oleh komputer.
	Simbol ini digunakan untuk menunjukkan suatu kondisi tertentu yang menghasilkan dua kemungkinan jawaban, ya/tidak.
	Simbol ini digunakan untuk menyatakan penyediaan tempat penyimpanan suatu pengolahan untuk memberi harga awal.
	Simbol ini digunakan untuk menyatakan permulaan atau akhir suatu program.
	Simbol ini digunakan untuk menyatakan segala jenis operasi yang diproses dengan menggunakan suatu mesin yang mempunyai <i>keyboard</i> .
	Simbol ini digunakan untuk menunjukkan bahwa data dalam simbol ini akan disimpan ke suatu media tertentu.
	Simbol ini digunakan untuk memasukkan data secara manual dengan menggunakan online <i>keyboard</i> .
	Simbol ini digunakan untuk menyatakan proses input dan <i>output</i> tanpa tergantung dengan jenis peralatanya.
	Simbol ini digunakan untuk menyatakan input berasal dari kartu atau <i>output</i> ditulis ke kartu
	Simbol ini digunakan untuk menyatakan <i>input</i> berasal dari pita <i>magnetic</i> atau <i>output</i> disimpan ke pita <i>magnetic</i> .

	Simbol ini digunakan untuk menyatakan <i>input</i> berasal dari <i>disk</i> atau <i>output</i> disimpan ke <i>disk</i> .
	Simbol ini digunakan untuk mencetak laporan ke printer.
	Simbol ini digunakan untuk menyatakan peralatan <i>output</i> yang digunakan berupa layar (video, komputer).
	Simbol ini digunakan untuk menyatakan sambungan dari satu proses ke proses lainnya dalam halaman/lembar yang sama.
	Simbol ini digunakan untuk menyatakan sambungan dari satu proses ke proses lainnya dalam halaman/lembar yang berbeda.
	Simbol ini digunakan untuk menyatakan bahwa adanya transisi suatu data/informasi dari satu lokasi ke lokasi lainnya.
	Simbol ini digunakan untuk menyatakan jalannya arus suatu proses.

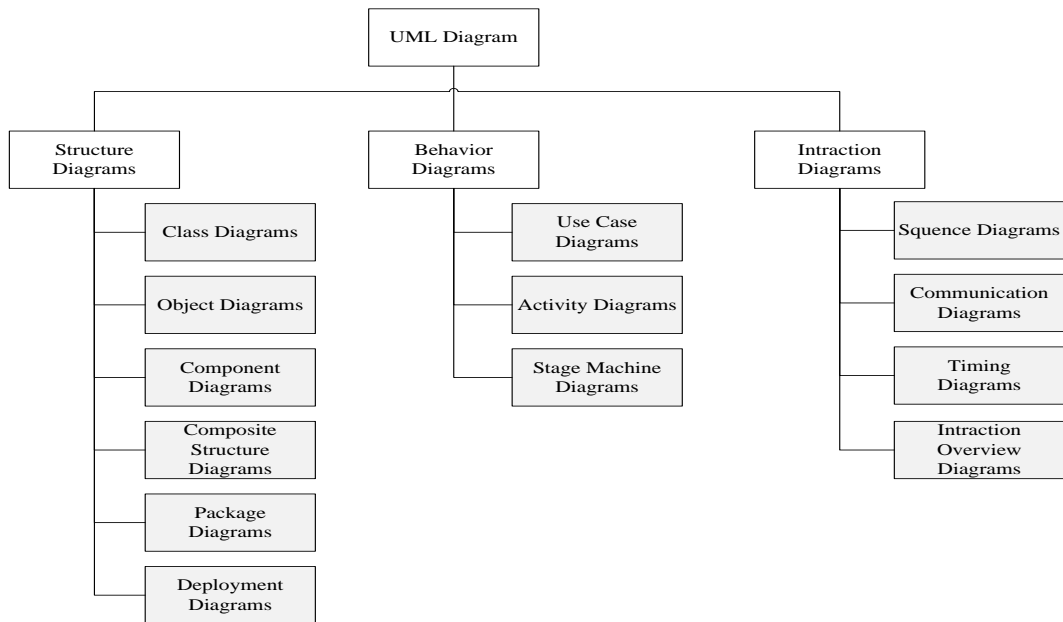
Sumber : Rosa A.S dan M. Shalahuddin (2016)

2.4 Perancangan *Unified Modelling Language* (UML)

Menurut (Shalahuddin, 2016) menyatakan bahwa:

“Bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks pendukung”.

Berikut ini merupakan penjelasan tentang masing-masing diagram yang ada pada *Unified Modelling Language* (UML), macam-macam diagram yang dikelompokkan dalam 3 kategori. Pembagian kategori dan macam-macam diagram tersebut dapat dilihat pada gambar 2.1. sebagai berikut:



Gambar 2.1 Diagram UML

Sumber: Shalahuddin, (2016)


2.4.1 Use Case Diagram

Menurut (Shalahuddin, 2016) menyatakan bahwa:

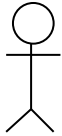

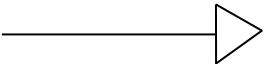


“Use case diagram atau diagram use case merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat”.

Use case mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat.

Tabel 2.3 Simbol Use Case Diagram

No.	Simbol	Keterangan
1.		Use Case Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antara unit atau aktor, biasanya menggunakan kata kerja diawal di awa frase nama use case.

Tabel 2.3 Simbol *Use Case Diagram* (Lanjutan)

No.	Simbol	Keterangan
2.		<p>Aktor</p> <p>Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor belum tentu merupakan orang, biasanya dinyatakan menggunakan kata benda diawal frase nama aktor</p>
3.		<p>Asosiasi/association</p> <p>Merupakan komunikasi antara aktor dan use case yang berpartisipasi pada use case atau use case memiliki interaksi dengan aktor.</p>
4.		<p>Generalisasi (<i>generalization</i>)</p> <p>merupakan hupromosin (umum – khusus) antara dua buah use case dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya.</p>
5.	<p><< Include >></p> 	<p>Include berarti use case yang ditambahkan akan dipanggil saat use case tambahan dijalankan.</p>
6.	<p><<Extend>></p> 	<p>Ekstensi (<i>extend</i>) merupakan use case tambahan ke sebuah use case yang ditambahkan dapat berdiri sendiri walau tanpa use case tambahan itu.</p>



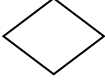

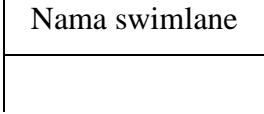

Sumber: Shalahuddin, (2016)

2.4.2 Activity Diagram

Menurut (Shalahuddin, 2016) menyatakan bahwa:

“*Activity Diagram* menggambarkan aliran kerja atau aktivitas dari sebuah sistem atau proses bisnis atau menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem”.

Tabel 2.4 Simbol Activity Diagram

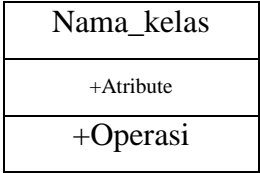


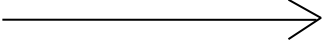
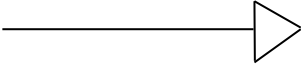
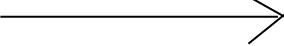
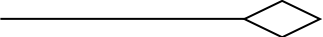
No.	Simbol	Keterangan
1.		Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal.
2.		Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
3.		Percabangan (Decision) merupakan asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu.
4.		Penggabungan (Join) merupakan asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu.
5.		Swimlane Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas.
6.		Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir.

2.4.3 Class Diagram

Menurut (Shalahuddin, 2016) menyatakan bahwa:

“*Class diagram* mengembangkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem”.

Tabel 2.5 Simbol *Class Diagram*

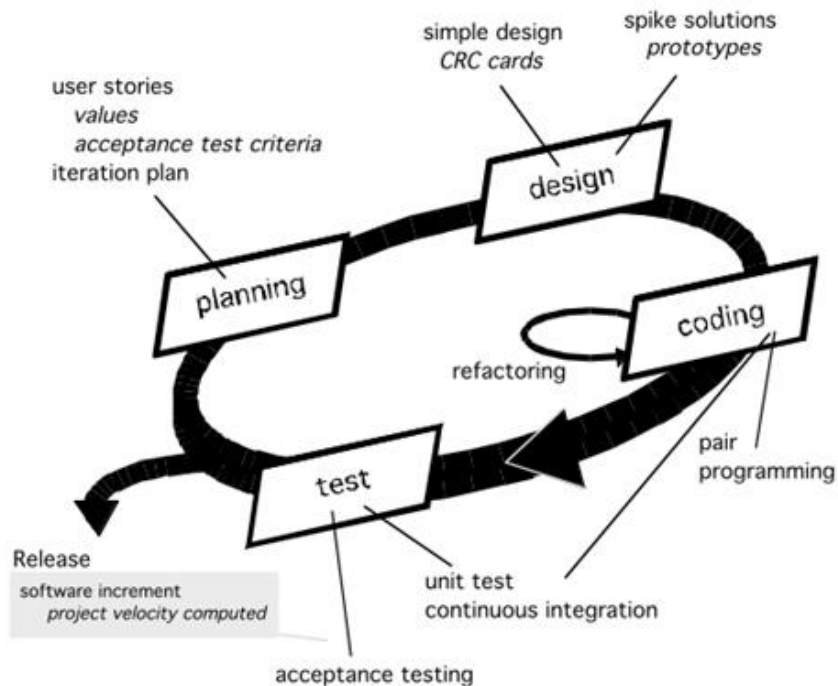
No.	Simbol	Keterangan
1.		Kelas pada struktur sistem.
2.	<p>Antar Muka/Interface</p>  <p>Nama_Interface</p>	Sama dengan konsep interface dalam pemrograman berorientasi objek.
3.	<p>Asosiasi / Asociation</p> 	Relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan multiplicity.
4.	<p>Asosiasi Berarah / Directed Association</p> 	Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan multiplicity.
5.	<p>Generalisasi</p> 	Relasi antar kelas dengan makna generalisasi-spesialisasi (umum khusus)
6.	<p>Ketergantungan / dependency</p> 	Relasi antar kelas dengan makna ketergantungan antar kelas.
7.	<p>Agregasi / aggregation</p> 	Relasi antar kelas dengan maksna semua bagian (whole-part)

Sumber: Shalahuddin, (2016)

2.5 Metode Pengembangan Sistem

Menurut (Abrahamsson, 2018), *Extreme Programming* (XP) adalah metode pengembangan perangkat lunak yang sederhana dan mencakup salah satu metode tangkas yang dipelopori oleh Kent Beck, Ron Jeffries, dan Ward Cunningham. XP adalah salah satu metode tangkas yang paling banyak digunakan dan menjadi pendekatan yang sangat terkenal. Tujuan XP adalah tim yang terbentuk antara kursus

berukuran kecil hingga menengah, tidak perlu menggunakan tim besar. Hal ini dimaksudkan untuk mengatasi persyaratan yang tidak jelas dan perubahan persyaratan dengan sangat cepat. Untuk gambar extreme programming ditampilkan pada gambar 2.2 Sebagai berikut:



Gambar 2.2. *Extreme Programming*

Adapun tahapan pembangunan aplikasi web seleksi siswa berprestasi dengan XP adalah sebagai berikut:

1. *Planning* (Perencanaan) Tahapan ini dimulai dengan mendengarkan kumpulan kebutuhan aktifitas suatu sistem yang memungkinkan pengguna memahami proses bisnis untuk sistem dan mendapatkan gambaran yang jelas mengenai fitur utama, fungsionalitas dan keluaran yang diinginkan. Dalam pembangunan data mining pada tahapan ini dimulai dari mengidentifikasi permasalahan yang timbul pada sistem yang sedang berjalan, kemudian dilakukan analisa kebutuhan pengguna terhadap sistem yang akan dibangun.

2. *Design* (Perancangan) Pada tahapan perancangan dilakukan pembuatan pemodelan sistem berdasarkan hasil analisa kebutuhan yang didapatkan. Selain itu dibuatkan juga pemodelan basis data untuk menggambarkan hubungan antar data. Pemodelan sistem yang digunakan yaitu Unified Modelling Language (UML) yang terdiri dari beberapa diagram antara lain *Use Case Diagram*, *Activity Diagram*, *Component Diagram* dan *Deployment Diagram*. Sedangkan untuk pemodelan basis data menggunakan *Entity Relationship Diagram* (ERD) dan *Logical Record Structure* (LRS).
3. *Coding* (Pengkodean) Tahapan ini merupakan implementasi dari perancangan model sistem yang telah dibuat kedalam kode program yang menghasilkan prototipe dari perangkat lunak. Dalam pembangunan data mining menggunakan bahasa pemrograman PHP yang dikombinasikan dengan HTML, CSS dan Javascript. Untuk implementasi basis data, Database Management System yang digunakan adalah MySQL.
4. *Testing* (Pengujian) Tahapan ini merupakan tahapan pengujian terhadap aplikasi yang sudah dibangun, pada tahapan ini ditentukan oleh pengguna sistem dan berfokus pada fitur dan fungsionalitas dari keseluruhan sistem kemudian ditinjau oleh pengguna sistem. Metode yang digunakan dalam melakukan pengujian terhadap data mining adalah ISO 25010 Testing dengan melakukan pengujian terhadap masukan dan keluaran yang dihasilkan sistem.
5. *Software Increment* (Peningkatan Perangkat Lunak). Tahapan ini merupakan tahap pengembangan sistem yang sudah dibuat secara bertahap yang dilakukan setelah sistem diterapkan dalam organisasi dengan menambahkan layanan atau konten yang mengakibatkan bertambahnya kemampuan fungsionalitas dari sistem.

2.6 ISO 25010

Model ini merupakan bagian dari *Software product Quality Requirements and Evaluation* (SQuaRE), dimana model ini berkaitan dengan model kualitas perangkat lunak yang merupakan pengembangan dari model sebelumnya. Pada model ini terdapat beberapa sub-karakteristik tambahan dan beberapa sub-karakteristik yang dipindahkan ke karakteristik lain. Berikut ini merupakan karakteristi atau faktor kualitas internal dan eksternal yang terdapat pada model ISO-25010 (Lamada et al., 2020), dapat dilihat pada Tabel 2.6.

Tabel 2.6 Faktor dan Sub-faktor pada Model ISO-25010

No	Faktor Sub-faktor	Faktor Sub-faktor
1	<i>Functional Suitability</i>	<ol style="list-style-type: none"> 1. <i>Appropriateness</i> 2. <i>Accuracy</i> 3. <i>Functional Suitability Compliance</i>
2	<i>Reliability</i>	<ol style="list-style-type: none"> 1. <i>Availability</i> 2. <i>Fault Tolerance</i> 3. <i>Recoverability</i> 4. <i>Reliability Compliance</i>
3	<i>Performance Efficiency</i>	<ol style="list-style-type: none"> 1. <i>Time-behaviour</i> 2. <i>Resource-utilisation</i> 3. <i>Performance Efficiency Compliance</i>
4	<i>Operability</i>	<ol style="list-style-type: none"> 1. <i>Appropriateness Recognisability</i> 2. <i>Learnability</i> 3. <i>Ease of Use</i> 4. <i>Helpfulness</i> 5. <i>Attractiveness</i> 6. <i>Technical Accessibility</i> 7. <i>Operability Compliance</i>
5	<i>Security</i>	<ol style="list-style-type: none"> 1. <i>Confidentiality</i> 2. <i>Integrity</i>

No	Faktor Sub-faktor	Faktor Sub-faktor
		<ul style="list-style-type: none"> 3. <i>Non-repudiation</i> 4. <i>Accountability</i> 5. <i>Authenticity</i> 6. <i>Security Compliance</i>
6	<i>Compatibility</i>	<ul style="list-style-type: none"> 1. <i>Replaceability</i> 2. <i>Co-existence</i> 3. <i>Interoperability</i> 4. <i>Compatibility Compliance</i>
7	<i>Maintainability</i>	<ul style="list-style-type: none"> 1. <i>Modularity</i> 2. <i>Reusability</i> 3. <i>Analyzability</i> 4. <i>Changeability</i> 5. <i>Modification Stability</i> 6. <i>Testability</i> 7. <i>Maintainability Compliance</i>
8	<i>Transferability</i>	<ul style="list-style-type: none"> 1. <i>Portability</i> 2. <i>Adaptability</i> 3. <i>Installability</i> 4. <i>Transferability Compliance</i>

Berikut ini merupakan pengertian dari masing-masing faktor dan sub-faktor yang terdapat pada model ISO-25010, antara lain:

1. *Functional suitability* merupakan tingkat dimana perangkat lunak dapat menyediakan fungsionalitas yang dibutuhkan ketika perangkat lunak digunakan pada kondisi yang spesifik.
 - a. *Appropriateness* merupakan tingkat dimana perangkat lunak dapat memenuhi kelayakan dari sebuah fungsi untuk melakukan pekerjaan yang spesifik bagi pengguna tertentu.
 - b. *Accuracy* merupakan tingkat dimana perangkat lunak dapat memberikan hasil yang tepat dan ketelitian terhadap tingkat kebutuhan.
 - c. *Functional Suitability Compliance* merupakan tingkat dimana perangkat lunak memenuhi standar *functional suitability* yang terdapat pada perangkat lunak lainnya yang sejenis.
2. *Reliability* merupakan tingkatan dimana perangkat lunak dapat bertahan pada tingkatan tertentu ketika digunakan oleh pengguna pada kondisi yang spesifik.
 - a. *Availability* merupakan tingkatan dimana komponen perangkat lunak dapat beroperasi dan siap ketika dibutuhkan untuk digunakan.
 - b. *Fault Tolerance* merupakan tingkat dimana sebuah perangkat lunak dapat bertahan pada tingkat kemampuan tertentu terhadap kegagalan atau kesalahan yang terdapat pada perangkat lunak.
 - c. *Recoverability* merupakan tingkat dimana perangkat lunak dapat kembali pada tingkat kemampuan tertentu dan melakukan pengembalian data secara langsung yang disebabkan oleh kegagalan atau kesalahan yang terjadi pada perangkat lunak.

- d. *Reliability Compliance* merupakan tingkat dimana perangkat lunak dapat memenuhi standar ketahanan yang dimiliki oleh perangkat lunak lain sejenis
3. *Performance Efficiency* merupakan tingkat dimana perangkat lunak dapat memberikan kinerja yang tepat terhadap sejumlah sumber daya yang digunakan pada kondisi tertentu.
 - a. *Time-behaviour* merupakan tingkat dimana perangkat lunak dapat memberikan reaksi dan waktu yang dibutuhkan ketika melakukan aksi dari sebuah fungsi pada kondisi tertentu.
 - b. *Resource-utilisation* merupakan tingkat dimana sebuah perangkat lunak menggunakan sejumlah dan beberapa sumber daya ketika perangkat lunak melakukan aksi dari sebuah fungsi pada kondisi tertentu.
 - c. *Performance Efficiency Compliance* merupakan tingkatan dimana perangkat lunak memenuhi standar yang berhubungan dengan efisiensi kinerja perangkat lunak.
 4. *Operability* merupakan tingkat dimana perangkat lunak dapat dimengerti, dipelajari, digunakan, dan menarik perhatian pengguna ketika digunakan pada kondisi tertentu.
 - a. *Appropriateness Recognisability* merupakan tingkat dimana perangkat lunak memungkinkan untuk dianalisis oleh pengguna apakah perangkat lunak sudah dapat memenuhi kebutuhan mereka.
 - b. *Learnability* merupakan tingkat dimana perangkat lunak mudah untuk dipelajari penggunaannya oleh para pengguna.
 - c. *Ease of Use* merupakan tingkat dimana sebuah perangkat lunak dapat digunakan dan dioperasikan oleh para penggunanya.
 - d. *Helpfulness* merupakan tingkat dimana perangkat lunak dapat memberikan bantuan ketika pengguna membutuhkan panduan.

- e. *Attractiveness* merupakan tingkat dimana perangkat lunak dapat menarik perhatian bagi para penggunanya.
- f. *Technical Accessibility* merupakan tingkat dimana penggunaan perangkat lunak untuk memenuhi kebutuhan pengguna yang memiliki keterbatasan
- g. *Operability Compliance* merupakan tingkat dimana perangkat lunak dapat memenuhi standarisasi dan panduan yang berhubungan dengan penggunaan perangkat lunak.

5. *Security*

Security merupakan perlindungan terhadap perangkat lunak dari berbagai ancaman, akses atau penggunaan dari pengguna yang tidak dikenal.

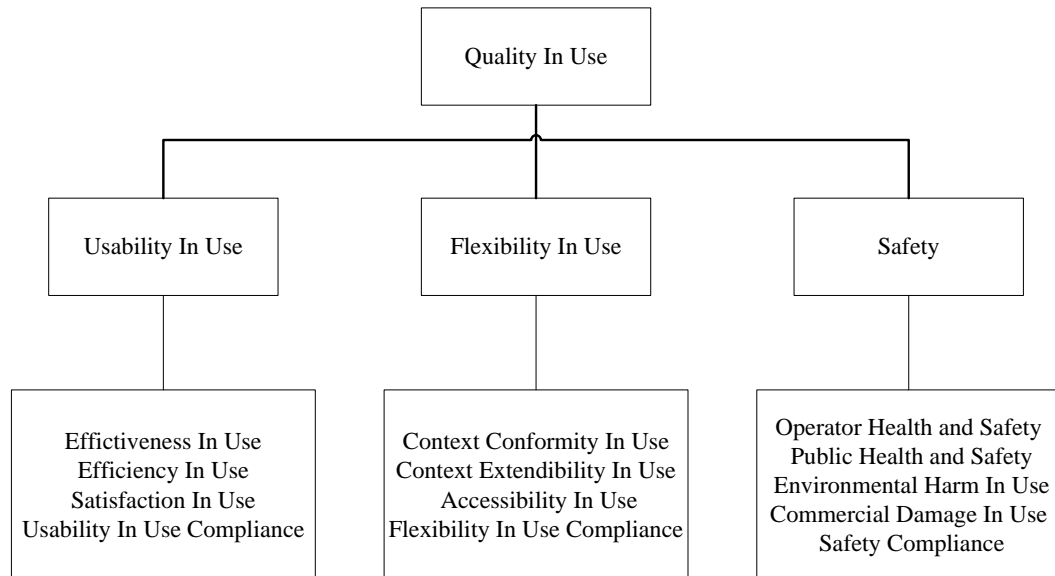
- a. *Confidentially* merupakan tingkat dimana perangkat lunak memiliki perlindungan terhadap data atau informasi dari pengguna, apakah sebagai ancaman atau kejengajaan.
 - b. *Integrity* merupakan tingkat dimana kelengkapan dan ketepatan dari sejumlah asset telah dijaga.
 - c. *Non-repudiation* merupakan tingkatan dimana aksi atau tindakan yang dilakukan telah terbukti sehingga hal tersebut tidak dapat ditolak.
 - d. *Accountability* merupakan tingkat dimana aksi dari sebuah entitas dapat ditelusuri keunikannya terhadap entitas.
 - e. *Authenticity* merupakan tingkat dimana identitas dari sebuah subjek atau sumber dapat diterima.
 - f. *Security Compliance* merupakan tingkat dimana perangkat lunak dapat memenuhi standarisasi yang berhubungan dengan keamanan perangkat lunak.
6. *Compatibility* Faktor ini merupakan kemampuan dari dua atau lebih komponen perangkat lunak untuk dapat melakukan pertukaran informasi dan melakukan fungsi

yang dibutuhkan ketika digunakan pada hardware atau lingkungan perangkat lunak yang sama.

- a. *Replaceability* merupakan tingkat dimana perangkat lunak dapat digunakan pada perangkat lunak lain dengan tujuan yang spesifik pada lingkungan yang sama.
 - b. *Co-existence* merupakan tingkat dimana perangkat lunak dapat berdampingan dengan perangkat lunak lain dan berbagi sumber daya pada lingkungan yang sama. tanpa dampak yang merugikan.
 - c. *Interoperability* merupakan tingkat dimana perangkat lunak dapat beroperasi secara kooperatif dengan satu atau lebih perangkat lunak lain.
 - d. *Compatibility Compliance* merupakan tingkat dimana perangkat lunak dapat memenuhi standarisasi terkait compatibility yang terdapat pada perangkat lunak sejenis lainnya.
7. *Maintainability* merupakan tingkat dimana sebuah perangkat lunak dapat dimodifikasi. Modifikasi ini termasuk perbaikan, perubahan atau penyesuaian perangkat lunak untuk dapat berubah pada lingkungan, kebutuhan, dan fungsionalitas yang spesifik.
- a. *Modularity* merupakan tingkatan dimana sistem atau program terdiri dari komponen yang berlainan sehingga perubahan terhadap satu komponen minimal memiliki pengaruh terhadap komponen lain.
 - b. *Reusability* merupakan tingkat dimana sebuah asset dapat digunakan pada lebih dari satu sistem perangkat lunak atau pada pembangunan asset lainnya.
 - c. *Analyzability* merupakan tingkat dimana perangkat lunak dapat dianalisis untuk mengetahui apa yang menyebabkan kegagalan pada perangkat lunak atau untuk mengidentifikasi bagian yang dapat dimodifikasi.

- d. *Changeability* merupakan tingkat dimana perangkat lunak memungkinkan sebuah modifikasi yang spesifik untuk dapat di implementasikan. Hal ini memudahkan perangkat lunak dapat untuk dimodifikasi.
 - e. *Modification Stability* merupakan tingkat dimana perangkat lunak dapat menghindari efek yang tidak diharapkan dari modifikasi yang dilakukan terhadap perangkat lunak.
 - f. *Testability* merupakan tingkat dimana perangkat lunak memungkinkan modifikasi perangkat lunak untuk dilakukan validasi.
 - g. *Maintainability Compliance* merupakan tingkatan dimana sebuah perangkat lunak dapat memenuhi standarisasi yang berhubungan dengan maintainability pada perangkat lunak lain.
8. *Transferability* merupakan tingkat dimana perangkat lunak dapat berpindah dari lingkungan yang satu ke lingkungan yang lain.
- a. *Portability* merupakan kemudahan dimana sistem atau komponen dapat berpindah dari lingkungan hardware atau perangkat yang satu ke *hardware* atau perangkat lunak lain.
 - b. *Adaptability* merupakan tingkat dimana perangkat lunak dapat beradaptasi pada spesifikasi lingkungan yang berbeda tanpa menerapkan aksi atau cara lain dari pada memberikan tujuan tertentu terhadap perangkat lunak yang telah ada.
 - c. *Installability* merupakan tingkat dimana perangkat lunak dapat dipasang dan dihapus pada lingkungan yang spesifik.
 - d. *Transferability Compliance* merupakan tingkat dimana sebuah perangkat lunak dapat memenuhi standar yang berhubungan dengan kemampuan perangkat lunak untuk dapat berpindah dari satu lingkungan ke lingkungan lain.

Selain faktor internal dan eksternal yang terdapat pada model ISO-25010, terdapat pula faktor quality in use seperti pada model sebelumnya, hanya saja terdapat beberapa perubahan pada model ini yang dapat dilihat pada Gambar 2.3.



Gambar 2.3. Model *Quality in Use* pada ISO-25010

1. *Usability in Use* merupakan tingkatan dimana pengguna dapat mencapai tujuannya dengan penggunaan yang efektif, efisien, dan kepuasan pada penggunaannya.
 - a. *Effectiveness in Use* merupakan tingkat dimana pengguna dapat mencapai tujuannya dengan ketepatan dan kesesuaian penggunaan.
 - b. *Efficiency in use* merupakan tingkat dimana pengguna dapat mengurangi sejumlah sumber daya untuk efektifitas penggunaan.
 - c. *Satisfaction in use* merupakan tingkatan dimana seorang pengguna mendapatkan kepuasan saat menggunakan perangkat lunak.
 - d. *Usability in Use Compliance* berhubungan dengan kesesuaian penggunaan perangkat lunak dengan standarisasi penggunaan yang ada.
2. *Flexibility in Use* merupakan tingkat dimana perangkat lunak dapat digunakan pada berbagai penggunaan.

- a. *Context Conformity in Use* merupakan tingkat sejauh mana perangkat lunak digunakan dalam hal penggunaannya.
 - b. *Context Extendibility in Use* merupakan tingkat sejauh mana perangkat lunak digunakan dalam hal diluar penggunaannya.
 - c. *Accessibility in Use* merupakan tingkat penggunaan untuk pengguna dengan keterbatasan kemampuan.
 - d. *Flexibility in Use Compliance* berhubungan dengan kesesuaian dengan standarisasi terkait *flexibility*.
3. *Safety* berhubungan dengan tingkatan penerimaan resiko dari kegagalan terhadap data, perangkat lunak, atau lingkungan yang termasuk dalam konteks penggunaan.
- a. *Operator Health and Safety* merupakan tingkat penerimaan dari resiko kegagalan terhadap pengguna pada konteks penggunaan.
 - b. *Public Health and Safety* merupakan tingkat penerimaan dari resiko kegagalan terhadap orang lain pada konteks penggunaan.
 - c. *Environmental Harm in Use* merupakan tingkat penerimaan dari resiko kegagalan yang dialami lingkungan dalam konteks penggunaan.
 - d. *Commercial Damage in Use* merupakan tingkat penerimaan dari resiko kegagalan terhadap kesalahan yang dapat mengarah kepada kerusakan reputasi pada konteks penggunaan.
 - e. *Safety Compliance* berhubungan dengan tingkatan dimana perangkat lunak dapat memenuhi standar keamanan terhadap penggunaannya