

## BAB II TINJAUAN PUSTAKA

Bagian ini mencakup sumber-sumber yang digunakan oleh penulis sebagai pedoman dalam perancangan dan pengembangan aplikasi berbasis *web* untuk Surat Keterangan Bebas Temuan menggunakan *Algoritma Decision Tree*.

### 2.1 Pencapaian Terdahulu

Penelitian sebelumnya memainkan peran krusial sebagai dasar referensi yang memperkuat argumen sekaligus menjadi panduan dalam penelitian ini. Bab ini juga berfungsi untuk mencegah terjadinya pengulangan penelitian yang serupa serta memastikan adanya hubungan yang relevan antara fenomena yang dikaji dengan algoritma yang akan diterapkan. Tabel 2.1 berisi referensi terdahulu yang mencakup publikasi ilmiah/jurnal terkait dengan penelitian ini.

Tabel 2.1 Algoritma Decision Tree Untuk Deteksi Aktifitas Malware

Nama Penulis	(Agus Syamsul Arifin et al., 2024)
Judul	Deteksi Aktifitas Malware pada Internet of Things menggunakan Algoritma Decision Tree dan Random Forest.
Hasil	Hasil nya adalah akurasi yang sangat tinggi. Kedua algoritma ini mencatatkan akurasi 100% pada data latih, sedangkan pada data uji, Random Forest memperoleh akurasi 99,94%, sedikit lebih tinggi dibandingkan Decision Tree yang mencatatkan 99,90%. Meskipun keduanya menunjukkan hasil yang hampir sama pada data latih, Random Forest lebih efektif dalam mendeteksi kelas Backdoor_Malware pada data uji. Nilai precision, recall, dan F1-Score yang diperoleh oleh Random Forest masing-masing adalah 99%, 64%, dan 78%, sementara Decision Tree mencatatkan precision 71%, recall 72%, dan F1-Score 72%. Hasil dari cross-validation 10-fold menunjukkan bahwa model yang dikembangkan tidak menunjukkan tanda-tanda overfitting, yang mengindikasikan bahwa model ini memiliki kemampuan generalisasi yang baik. Temuan ini membuktikan bahwa Random Forest lebih unggul dibandingkan Decision Tree dalam mendeteksi serangan malware pada jaringan IoT, terutama dalam mendeteksi kelas Backdoor_Malware. Penelitian ini diharapkan dapat memberikan kontribusi dalam pengembangan sistem deteksi malware yang lebih efektif dan andal untuk jaringan IoT.

Tabel 2.2 Algoritma Decision Tree Untuk Klasifikasi Tingkat Penjualan

Nama Penulis	(Priatmojo et al., 2023)
Judul	Perbandingan Klasifikasi Tingkat Penjualan Buah di Supermarket dengan Pendekatan Algoritma Decision Tree, Naive Bayes dan K-Nearest Neighbor.
Hasil	Penelitian ini membandingkan tiga algoritma klasifikasi untuk memprediksi tingkat penjualan buah di supermarket, yaitu Decision Tree, Naive Bayes, dan K-Nearest Neighbor. Hasilnya menunjukkan bahwa Decision Tree memiliki performa terbaik dengan akurasi 92,31% dan AUC 93,8%, yang masuk dalam kategori "Excellent". K-Nearest Neighbor juga memberikan hasil yang cukup baik dengan akurasi 88,46% dan AUC 99,4%, meskipun

	sedikit lebih rendah dibandingkan Decision Tree. Di sisi lain, Naive Bayes memiliki performa terendah dengan akurasi sebesar 73,08% dan AUC 79,4%, yang tergolong dalam kategori "Fair". Berdasarkan hasil tersebut, dapat disimpulkan bahwa Decision Tree merupakan algoritma paling efektif untuk mengolah data penjualan buah di Foodmart Supermarket.
--	---

Tabel 2.3 Algoritma Decision Tree Untuk Prediksi Persediaan Obat

Nama Penulis	(Pratiwi et al., 2024)
Judul	Prediksi Persediaan Obat Pada Apotek Menggunakan Algoritma Decision Tree.
Hasil	Hasil dari penelitian ini. Dengan data perbandingan 80:20, tingkat akurasi yang dicapai adalah 98,71%, dengan nilai matrix yaitu 0,9872, 0,9872, dan 0,9867. Pada data perbandingan 70:30, meskipun akurasi sedikit lebih rendah yaitu 98,28%, perhitungan evaluasi model tetap tinggi, yaitu sebesar 0,9832, 0,9828, dan 0,9804. Hasil menunjukkan bahwa prediksi model persediaan obat menggunakan Decision Tree memberikan performa yang konsisten dan dapat diandalkan. Penelitian ini berkontribusi dalam mengembangkan model prediksi persediaan obat yang dirancang untuk membantu apotek mengelola stok secara lebih efisien dan efektif. Implementasi model ini dalam bentuk aplikasi diharapkan dapat meningkatkan efisiensi operasional dan kualitas pelayanan apotek kepada pelanggan.

Tabel 2.4 Algoritma Decision Tree Untuk Prediksi Mahasiswa Non-Aktif

Nama Penulis	(Khatib Sulaiman et al., n.d.)
Judul	Prediksi Mahasiswa Berpotensi Non-Aktif Menggunakan Algoritma <i>Decision Tree Classifier</i>
Hasil	Penelitian ini mengungkapkan bahwa pengembangan model prediksi ini memberikan dampak yang besar dalam mengidentifikasi mahasiswa non aktif secara akademik di Universitas Dinamika Bangsa. Model ini memiliki tingkat akurasi sebesar 95,63%, yang menunjukkan kemampuan tinggi dalam mengklasifikasikan status akademik mahasiswa. Variabel yang paling berpengaruh dalam prediksi mahasiswa tidak aktif meliputi indeks prestasi semester 3 dan 2, dan usia saat disetujui. Selain itu, pengujian menunjukkan bahwa penambahan atribut nilai mata kuliah semester 3 meningkatkan akurasi model hingga mencapai 95,63%. Temuan ini sangat bermanfaat bagi institusi pendidikan.

Tabel 2.5 Penerapan Decision Tree untuk Klasifikasi Status Kesehatan

Nama Penulis	(Budi Prasetyo et al., 2024).
Judul	Penerapan Decision Tree untuk Klasifikasi Status Kesehatan dengan perbandingan KNN dan Naive Bayes
Hasil	Hasil dan bukti dari penelitian ini didasarkan pada analisis yang dilakukan setelah pelatihan model. Berikut adalah beberapa hasil dan bukti yang mungkin ditemukan dalam penelitian ini: Akurasi Prediksi: Penelitian ini mungkin menunjukkan akurasi prediksi dari masing-masing algoritma dalam klasifikasi status kesehatan berdasarkan variabel usia dan intensitas olahraga. Algoritma yang memberikan akurasi tertinggi dapat diidentifikasi sebagai algoritma yang paling optimal dalam memprediksi status kesehatan individu.

Tabel 2.6 Optimasi Algoritma Decision Tree untuk Prediksi Penyakit

Nama Penulis	Danita Suryani (2024).
Judul	Optimasi Algoritma Decision Tree untuk Prediksi Penyakit
Hasil	Dalam penelitian ini, hasil dan bukti ditemukan meliputi: Akurasi Prediksi: Model Decision Tree yang telah dioptimalkan mampu mencapai tingkat akurasi prediksi hingga 90%, menunjukkan peningkatan signifikan dalam kemampuan model untuk memprediksi penyakit berdasarkan data medis pasien. Kinerja Model yang Ditingkatkan: Setelah optimasi parameter, model Decision Tree yang dioptimalkan menunjukkan kinerja yang lebih baik dalam mengklasifikasikan keberadaan atau ketiadaan penyakit pada dataset pengujian. Optimasi parameter berhasil meningkatkan akurasi, presisi, recall, dan F1-score model secara signifikan. Implikasi Praktis: Hasil penelitian ini memiliki manfaat nyata dalam bidang kesehatan, di mana model yang ditingkatkan dapat membantu praktisi medis dalam diagnosis dini, peramalan penyakit, serta perencanaan pengobatan yang lebih efektif. Kontribusi Signifikan: Penelitian ini memberikan kontribusi penting dalam pengembangan model prediksi penyakit yang dapat mendukung pengambilan keputusan diagnostik yang lebih akurat dan informatif. Selain itu, temuan ini membuka peluang untuk pengembangan lebih lanjut melalui penerapan teknik optimasi dan algoritma machine learning lainnya dalam berbagai aplikasi medis. Dengan hasil dan bukti yang menunjukkan peningkatan kinerja model Decision Tree setelah optimasi parameter.

Tabel 2.7 Algoritma Decision Tree Untuk Pengukuran Capaian Pembelajaran Mahasiswa

Nama Penulis	(Ramadhan et al., 2024)
Judul	Pengembangan Aplikasi Pengukuran Capaian Pembelajaran Mahasiswa Untuk Rekomendasi Mata Kuliah Peminatan Menggunakan Algoritma Decision Tree
Hasil	Aplikasi pengukur yang berbasis algoritma Decision Tree ini menghasilkan beberapa temuan penting. Tingkat Akurasi: Pengujian menunjukkan bahwa akurasi aplikasi meningkat seiring dengan bertambahnya data. Sebagai contoh, ketika data yang digunakan sebesar 20%, akurasi mencapai 75%, dan meningkat menjadi 87,5% saat data diperluas hingga 40%. Aplikasi memiliki keunggulan untuk memberikan rekomendasi peminatan secara akurat. Kemudahan Penggunaan: Aplikasi ini dirancang untuk mempermudah mahasiswa dan pihak kampus dalam mengakses informasi capaian pembelajaran dan memperoleh rekomendasi secara otomatis, menjadikan proses pengolahan data lebih sederhana dan efisien. Tingkat Kesalahan Rendah: Selama proses pengujian, aplikasi beroperasi dengan stabil tanpa adanya bug atau kesalahan, membuktikan sistemnya yang andal dan dapat diandalkan. Rekomendasi yang Tepat: Berdasarkan analisis nilai mata kuliah, aplikasi mampu memberikan rekomendasi peminatan yang sesuai dengan kebutuhan dan preferensi mahasiswa, sehingga membantu mereka memilih mata kuliah dengan peluang kelulusan yang lebih tinggi. Dengan temuan-temuan ini, dapat disimpulkan bahwa aplikasi ini tidak hanya mempermudah pengukuran capaian pembelajaran mahasiswa, tetapi juga memberikan rekomendasi peminatan yang relevan dan mendukung keberhasilan akademik mahasiswa.

Tabel 2.8 Algoritma Decision Tree untuk Seleksi Penerima Beasiswa

Nama Penulis	(Hartanto, 2023)
Judul	Penerapan Algoritma Decision Tree untuk Seleksi Penerima Beasiswa (Studi Kasus: Smpn 1 Soreang)
Hasil	Penelitian ini digunakan untuk mendukung proses seleksi penerima beasiswa di SMPN 1 Soreang. Hasil penelitian menunjukkan bahwa algoritma ini mampu menghasilkan keputusan seleksi dengan tingkat akurasi mencapai 92%. Model dengan kriteria Gini Index menunjukkan akurasi sebesar 0,9684 pada data latih dan 0,9367 pada data uji. Sementara itu, model dengan kriteria Entropi mencatat akurasi 0,9620 pada data latih dan 0,9241 pada data uji. Penelitian juga mengungkap bahwa atribut yang paling memengaruhi proses seleksi adalah jumlah nilai siswa dan status klasifikasinya. Berdasarkan hasil tersebut, algoritma Decision Tree dengan kriteria Gini Index dan Entropi terbukti efektif dalam mendukung proses seleksi penerima beasiswa di SMPN 1 Soreang.

Tabel 2.9 Algoritma Decision Tree Untuk Klasifikasi Keluarga Miskin

Nama Penulis	(Fernanda et al., 2023)
Judul	Perancangan Aplikasi Data Mining Untuk Klasifikasi Keluarga Miskin Menggunakan Metode Decision Tree
Hasil	Penelitian ini menghasilkan pengembangan aplikasi data mining untuk klasifikasi keluarga miskin menggunakan Metode Decision Tree C4.5. Aplikasi ini dirancang dengan mempertimbangkan beberapa kriteria, seperti pekerjaan, Pajak Bumi Bangunan (PBB), jenis lantai, jenis dinding, sumber air, daya listrik, dan penghasilan. Tujuan utama dari aplikasi ini adalah membantu pihak Kecamatan Kerkap, Kabupaten Bengkulu Utara, dalam menganalisis dan mengidentifikasi keluarga miskin yang berhak menerima bantuan. Pengujian menunjukkan bahwa aplikasi ini berfungsi dengan baik dan mampu menyajikan hasil analisis klasifikasi keluarga miskin secara efektif menggunakan metode decision tree. Oleh karena itu, aplikasi ini diharapkan dapat menjadi alat yang andal untuk mendukung proses identifikasi keluarga miskin dan menentukan bantuan yang tepat sasaran.

Tabel 2.10 Algoritma Decision Tree Untuk Klasifikasi Peserta Menjadi 2 Jenis

Nama Penulis	(Prayudi & Yapis Dompnu, 2023)
Judul	Pengembangan Web Forum Ikatan Guru Indonesia Kabupaten Dompnu
Hasil	Hasil pengembangan Web Forum Ikatan Guru Indonesia Kabupaten Dompnu menggunakan metode Waterfall mencakup pengujian program oleh dua validator, yaitu ahli media dan ahli website. Hasil pengujian menunjukkan bahwa ahli media memberikan penilaian dengan persentase 85%, sementara ahli website memberikan 90%. Secara keseluruhan, rata-rata nilai persentasenya mencapai 87,5%, yang masuk dalam kategori sangat baik. Hal ini menunjukkan bahwa Web Forum yang dikembangkan menerima respons positif dari validator dan dinilai baik dalam hal fungsionalitas serta tampilan. Sebagai saran, peneliti selanjutnya disarankan untuk mengganti tema atau tampilan website dengan desain yang lebih menarik dan elegan, agar pengguna merasa lebih nyaman dan tidak bosan menggunakan forum tersebut. Saran ini diharapkan dapat meningkatkan pengalaman pengguna (user experience) dan keterlibatan pengguna dalam forum online yang telah dikembangkan.

## **2.2 Tinjauan Teoritis**

Tinjauan teoritis dalam penelitian ini berfungsi sebagai referensi dan pendukung informasi dalam membuktikan penelitian, adapun penjelasan detail mengenai tinjauan teoritis yang diterapkan dalam penelitian ini adalah sebagai berikut.

### **2.2.1 Dashboard**

Beberapa bagan, grafik, laporan, indikator visual, dan sistem peringatan ditampilkan di dasbor, antarmuka komputer yang akan digabungkan untuk menciptakan platform informasi yang dinamis dan relevan. Dasbor berfungsi sebagai konsol manajemen waktu nyata untuk proyek-proyek perusahaan (Triyanto et al., 2023).

### **2.2.2 Analisis Studi literatur**

Domain permasalahan ini berfokus pada pengembangan aplikasi berbasis web untuk pengelolaan data dan pengambilan keputusan menggunakan algoritma Decision Tree. Berdasarkan studi terdahulu, beberapa poin penting yang relevan adalah:

1. Efektivitas dan Akurasi: Decision Tree telah terbukti memberikan hasil akurasi yang tinggi dalam berbagai domain, seperti kesehatan, pendidikan, dan pengelolaan data logistik.
2. Pemilihan Variabel: Keberhasilan model sangat bergantung pada pemilihan dan pengolahan variabel input yang relevan, termasuk fitur-fitur yang memiliki korelasi kuat dengan output.
3. Optimasi Model: Optimasi parameter, seperti max depth, pruning, dan split criteria, dapat meningkatkan akurasi model serta kemampuan generalisasi.
4. Aplikasi Berbasis Web: Integrasi algoritma Decision Tree dalam aplikasi web memerlukan antarmuka yang user-friendly dan fitur pendukung, seperti pengunduhan data hasil prediksi dalam format yang mudah diakses.

### 2.2.2 *Algoritma Decision Tree*

Algoritma Decision Tree berfungsi sebagai model prediksi keputusan yang disusun dalam struktur hierarki. Setiap pohon memiliki cabang, di mana setiap cabang merepresentasikan atribut yang harus terpenuhi sebelum melanjutkan ke cabang berikutnya hingga mencapai daun sebagai hasil akhir. (Yulian Pamuji et al., 2021).

Aplikasi ini dirancang untuk mendukung penerbitan Surat Keterangan Bebas Temuan dengan memanfaatkan *Algoritma Decision Tree*. Proses dimulai dengan pengguna meninput data kedalam database berupa kolom-kolom yang di perlukan kedalam database. Dataset ini diverifikasi untuk memastikan kelengkapan dan validitasnya, termasuk melengkapi data kosong dan encoding data kategorikal. Data yang valid selanjutnya dipisah menjadi pelatihan dan pengujian. Model *Algoritma Decision Tree* dilatih menggunakan fitur seperti "jenis temuan" dan "aspek temuan" untuk memprediksi kelayakan data temuan. Model dievaluasi menggunakan metrik seperti akurasi, confusion matrix, precision, dan recall untuk memastikan keakuratan dan keandalan prediksi yang dihasilkan.

*Confusion matrix* memberikan gambaran performa model, dengan menghitung jumlah prediksi yang benar dan salah untuk masing-masing kelas, serta metrik seperti *precision* dan *recall* untuk setiap kelas. Hasil klasifikasi ditampilkan dalam bentuk tabel yang mencakup data asli, prediksi model, dan atribut input, sehingga memudahkan pengguna untuk menganalisis hasil klasifikasi. Berikut adalah rumus dari *gini index*, *precision*, *recall*, *accuracy*, penjelasan visualisasi pohon keputusan dan tabel klasifikasi.

*Gini Index* digunakan untuk mengukur *impurity* (ketidakmurnian) pada setiap node dalam pohon keputusan. *Gini Index* di aplikasi ini digunakan sebagai metrik utama dalam *Algoritma Decision Tree* untuk membangun model klasifikasi yang memprediksi rekomendasi SKBT berdasarkan data temuan administratif. Rumus ini secara otomatis diterapkan melalui library *sklearn* selama proses training model. Berikut adalah penjelasannya.

: Proporsi data di kelas  $i$ .

: Jumlah kelas.

*Precision* digunakan untuk mengukur proporsi prediksi positif yang benar. *Precision* merupakan metrik yang penting dalam aplikasi ini untuk memastikan model memberikan prediksi yang relevan dan akurat, khususnya untuk data dengan label "Layak". *Precision* membantu mendukung keputusan administratif yang lebih tepat dan bertanggung jawab, sehingga meningkatkan keandalan sistem dalam menghasilkan rekomendasi SKBT. Berikut adalah penjelasannya.

*True Positives (TP)*: Prediksi "Layak" yang benar.

*False Positives (FP)*: Prediksi "Layak" yang salah.

Contoh Perhitungan (Kelas Layak):

TP = 41, FP = 9.

*Precision*:

Recall digunakan untuk mengukur proporsi data positif yang terdeteksi dengan benar. Recall dalam aplikasi ini digunakan untuk mengukur kemampuan model dalam mendeteksi data positif, yaitu pegawai yang layak mendapatkan rekomendasi SKBT. Nilai recall yang tinggi, seperti 95.35% pada contoh, mengindikasikan bahwa model dapat mengenali sebagian besar data "Layak" secara tepat, sehingga meminimalkan risiko kesalahan deteksi pegawai layak (False Negative). Berikut adalah rumus dan penjelasannya.

*False Negatives (FN)*: Data "Layak" yang salah diprediksi sebagai "Tidak Layak."

Contoh Perhitungan (Kelas Layak):

TP = 41, FN = 2.

Accuracy digunakan untuk mengukur proporsi prediksi benar dari keseluruhan data. Dalam contoh, accuracy sebesar 81.03% menunjukkan bahwa model berhasil memprediksi dengan benar sebagian besar data "Layak" dan "Tidak Layak". Metrik ini memberikan gambaran umum performa model, namun perlu dilengkapi dengan Precision dan Recall untuk evaluasi yang lebih komprehensif. Berikut adalah rumus dan penjelasan dari accuracy.

Contoh Perhitungan:

Total data = 58.

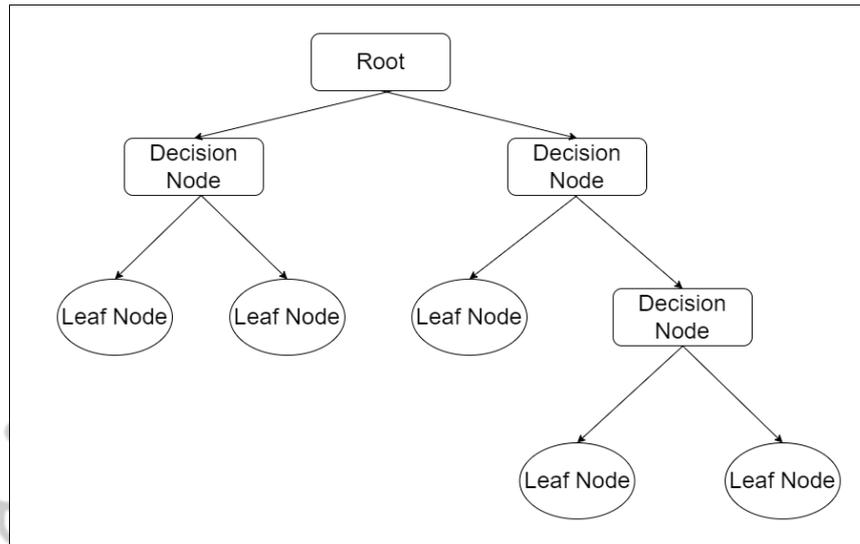
Prediksi benar: TP (Layak) + True Negatives (Tidak Layak) = 41 + 6 = 47.

Accuracy:

Berikut adalah penjelasan dari tabel hasil klasifikasi.

Tabel ini mencocokkan data aktual dengan prediksi. Misalnya:

1. Baris ke-1: "Jenis Temuan" = 5, "Aspek Temuan" = 1, Prediksi = "Layak", dan nilai aktual juga "Layak" (Prediksi Benar).
2. Keseluruhan hasil membantu pengguna memvalidasi model.



Gambar 2.1 Struktur Decision Tree

Gambar 2.1 merupakan struktur pohon keputusan (decision tree) yang terdiri dari beberapa node dan cabang. Dalam konteks klasifikasi data temuan, pohon keputusan dapat digunakan untuk mengidentifikasi apakah suatu entitas atau transaksi bebas temuan atau memiliki temuan berdasarkan kriteria tertentu, seperti kepatuhan terhadap regulasi, kesesuaian laporan keuangan, dan indikator relevan lainnya. Misalnya:

- 1) Root Node: Kepatuhan terhadap regulasi.
- 2) Decision Node: Kesesuaian laporan keuangan.
- 3) Leaf Node: Bebas temuan atau memiliki temuan.

Dengan mengikuti langkah-langkah di atas, *Algoritma Decision Tree* akan membantu mengklasifikasikan data audit secara efektif.

### 2.3.3 Flask

Flask adalah sebuah micro-framework berbasis Python yang dirancang untuk kesederhanaan dan fleksibilitas dalam membangun aplikasi web. Flask memungkinkan pengembang untuk memulai dengan cepat tanpa membawa terlalu banyak fitur bawaan, sehingga memberikan kendali penuh pada pengembang untuk menentukan kebutuhan aplikasi. Secara default, Flask tidak menyertakan komponen umum seperti validasi formulir, pengelolaan basis data, atau fitur lainnya. Namun, salah satu keunggulan Flask adalah dukungannya terhadap berbagai ekstensi yang memungkinkan penambahan fungsionalitas tersebut ke dalam aplikasi yang dikembangkan (Tamariska Bota & Setiyawati, 2022). Dengan pendekatan ini, Flask menjadi pilihan ideal bagi pengembang yang membutuhkan kerangka kerja ringan namun tetap dapat diperluas sesuai kebutuhan.

### 2.3.4 HTML

HTML (*HyperText Markup Language*), adalah bahasa yang digunakan untuk membuat halaman *website*. Kita bisa membangun sebuah *website* menggunakan berbagai tag dalam kodenya. Sederhananya, HTML dapat diibaratkan sebagai kerangka utama dari sebuah *website*. (Dody Firmansyah, 2023). Dalam pengembangan ini HTML digunakan dalam membuat struktur halaman *website*, elemen antarmuka pengguna, tata letak dan tampilan dari aplikasi SKBT.

### 2.3.7 Bootstrap

Kerangka kerja CSS yang disebut Bootstrap digunakan oleh para pengembang untuk merampingkan, mempercepat, dan menyempurnakan tampilan situs web mereka. Bootstrap adalah pustaka yang sangat membantu untuk pemrograman front-end, atau sisi klien (Dody Firmansyah, 2023).

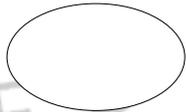
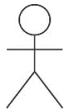
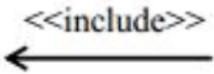
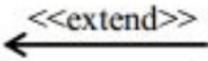
### 2.3.8 Unified Modelling Language (UML)

Pengembang perangkat lunak menggunakan UML sebagai metode pemodelan untuk mendefinisikan, membuat, juga mengatur sistem *software*. UML bertugas menjelaskan arsitektur dan struktur sistem kepada para pemangku kepentingan, tim pengembang, dan peserta proyek lainnya.

### 1) Use case Diagram

Teknik untuk mendokumentasikan dan menggambarkan fungsi sistem dari sudut pandang pengguna adalah diagram kasus penggunaan. Berikut adalah penjelasan simbol yang digunakan dalam diagram kasus penggunaan.

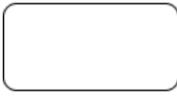
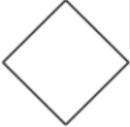
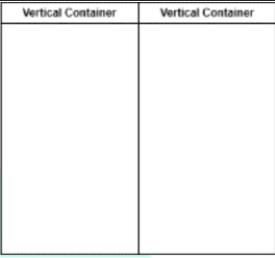
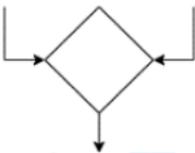
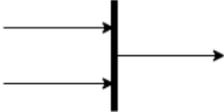
Tabel 2.11 Simbol Usecase Diagram

Nama	Simbol	Keterangan
Use Case		Kasus penggunaan: Abstraksi sistem dan interaksi aktor-sistem
Actor		Aktor: ketika berinteraksi dengan sebuah use case, mereka memainkan peran sebagai individu, sistem lain, atau alat.
Association		Abstraksi hubungan antara usecase dan aktor disebut asosiasi.
Generalisasi		Generalisasi: Mendemonstrasikan bidang keahlian aktor agar mereka dapat terlibat dengan kasus penggunaan.
Include		Mengindikasikan bahwa fungsi suatu use case sepenuhnya bergantung pada fungsi yang lain.
Extend		Mengindikasikan sebuah use case dapat memperluas fungsi dari yang lain jika suatu kondisi terpenuhi.

### 2) Activity Diagram

Tujuannya adalah memaparkan alur kerja proses. Untuk tujuan pemodelan dan pencatatan alur kerja, proses bisnis, perangkat lunak, atau operasi sistem lainnya, diagram ini sangat membantu. Penjelasan mengenai simbol-simbol pada diagram aktivitas dapat ditemukan di bawah ini.

Tabel 2.12 Simbol Activity Diagram

Nama	Simbol	Keterangan
<i>Start Node</i>		Tanda berfungsi sebagai titik awal diagram aktivitas.
<i>Activity</i>		Mengidentifikasi tugas yang sedang dilakukan.
<i>Decision</i>		Titik yang menunjukkan prasyarat untuk melakukan suatu aktivitas.
<i>Swimline</i>		Dirancang untuk membagi tugas ke dalam kolom-kolom di mana objek-objek tersebut bertanggung jawab untuk melaksanakan tugas.
<i>Merge Event</i>		Gabungkan aliran yang dibagi berdasarkan pilihan.
<i>Fork</i>		Membantu mengintegrasikan perilaku ke dalam tugas-tugas yang bersamaan.
<i>Action Flow</i>		Digunakan untuk menunjukkan aktivitas berikutnya sebagai transisi dari satu aktivitas ke simpul terakhir sebagai indikasi bahwa aktivitas tertentu akan segera berakhir.
<i>Final Mode</i>		Digunakan untuk menunjukkan aktivitas berikutnya sebagai transisi dari satu aktivitas ke simpul terakhir sebagai indikasi bahwa aktivitas tertentu akan segera berakhir.