

## **BAB II**

### **KAJIAN PUSTAKA**

#### **A. Dasar-dasar Teori**

##### **1. Rancang Bangun**

Perancangan, menurut (Iskandar & Eka Susilawati, 2025). adalah serangkaian prosedur yang digunakan untuk menginterpretasikan atau menjelaskan hasil analisis suatu sistem ke dalam bahasa pemrograman. Proses ini mendeskripsikan secara rinci bagaimana komponen-komponen sistem dapat diimplementasikan dengan efektif. Sementara itu, pembangunan sistem merujuk pada kegiatan menciptakan sistem baru yang menggantikan sistem lama atau yang sudah ada, baik secara keseluruhan maupun sebagian.

Perancangan sistem melibatkan penentuan cara sistem yang akan dibangun dapat menyelesaikan berbagai masalah yang ada. Pada tahap ini, termasuk di dalamnya adalah konfigurasi perangkat lunak, perangkat keras, dan komponen-komponen lain dalam sistem, sehingga setelah diimplementasikan, sistem yang dibangun dapat memenuhi kebutuhan yang telah ditetapkan dalam analisis sistem sebelumnya (Aji Sujatmiko et al., 2024).

Dalam konteks penelitian ini, fokus perancangan dan pembangunan sistem adalah pada pengembangan sistem informasi inventaris berbasis web, yang diharapkan dapat menggantikan metode pencatatan manual yang selama ini digunakan.

##### **2. Sistem**

###### **a. Pengertian Sistem**

Sistem adalah sekumpulan komponen atau elemen yang saling berinteraksi satu sama lain, membentuk suatu kesatuan untuk mencapai tujuan tertentu (Mulyanto et al., 2008). Menurut Kamus Besar Bahasa Indonesia (KBBI), sistem didefinisikan sebagai perangkat unsur yang saling terkait secara teratur sehingga membentuk suatu totalitas. Definisi ini juga mencakup susunan yang teratur dari pandangan, teori, atau prinsip.

Dari penjelasan di atas, dapat disimpulkan bahwa sistem merupakan kumpulan elemen atau komponen yang saling berhubungan dan terorganisasi dengan baik untuk mencapai tujuan tertentu. Definisi ini mencakup interaksi antara komponen secara keseluruhan, baik dalam bentuk unsur fisik maupun konsep, yang tersusun secara sistematis.

b. Karakteristik Sistem

Pada dasarnya, sistem memiliki model dasar yang terdiri dari input, proses, dan output. Model ini merupakan konsep sederhana dari sebuah sistem, di mana suatu sistem dapat memiliki beberapa masukan dan keluaran secara bersamaan. Setiap sistem memiliki sifat atau karakteristik tertentu, yaitu (Sutabri, 2012):

1) Komponen Sistem (*System Components*)

Dalam suatu sistem, terdapat sejumlah komponen yang sering disebut sebagai subsistem, yang saling berhubungan dan berinteraksi satu sama lain. Ini berarti bahwa komponen-komponen tersebut bekerja sama untuk membentuk satu kesatuan. Beberapa komponen dalam sistem dapat dibagi menjadi bagian-bagian atau subsistem, di mana masing-masing subsistem memiliki karakteristik yang mendukung fungsi tertentu yang dapat mempengaruhi proses sistem secara keseluruhan.

2) Batas Sistem (*Boundary*)

Batas sistem adalah area yang memisahkan suatu sistem dari lingkungan luar, termasuk sistem lainnya. Batasan ini dapat direpresentasikan sebagai ruang lingkup (*scope*) dari sistem itu sendiri.

3) Lingkungan Luar Sistem (*Environments*)

Lingkungan luar sistem mencakup segala sesuatu yang berada di luar batas sistem yang dapat mempengaruhi operasional sistem tersebut. Beberapa elemen dalam lingkungan luar dapat memberikan keuntungan bagi sistem, sementara yang lain dapat merugikan.

4) Penghubung Sistem (*Interface*)

Penghubung di sini merujuk pada media yang menghubungkan satu subsistem dengan subsistem lainnya. Dengan adanya penghubung ini, sistem dapat mengalirkan sumber daya dari satu subsistem ke subsistem lainnya.

5) Masukan Sistem (*Input*)

Masukan sistem atau input adalah energi yang dimasukkan ke dalam sistem, yang dapat berupa masukan perawatan dan masukan sinyal. Masukan perawatan adalah energi yang diperlukan agar sistem dapat beroperasi dengan baik, sedangkan masukan sinyal adalah energi yang akan diproses untuk menghasilkan keluaran dari sistem.

6) Keluaran Sistem (*Output*)

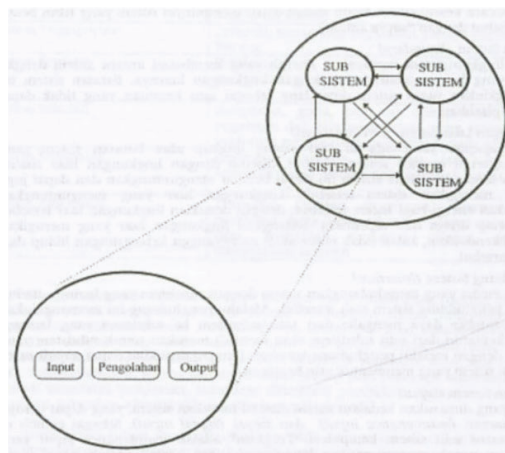
Keluaran sistem atau output adalah hasil dari energi masukan yang telah diproses, yang diklasifikasikan menjadi keluaran yang berguna serta sisa pembuangan dari proses tersebut.

7) Pengolah Sistem (*Procces*)

Pengolah sistem atau proses adalah bagian yang bertanggung jawab untuk mengubah input menjadi output.

8) Sasaran Sistem (*Objective*)

Setiap sistem memiliki sasaran atau tujuan yang mendasari penciptaannya. Jika suatu sistem tidak memiliki sasaran, maka operasional sistem tersebut tidak akan memberikan manfaat.



Gambar 2. 1 Karakteristik Sistem

### c. Klasifikasi Sistem

Dari berbagai sudut pandang, sebuah sistem dapat diklasifikasikan ke dalam beberapa kategori, antara lain (Sutabri, 2012):

#### 1) Sistem Abstrak dan Sistem Fisik

Sistem abstrak adalah sistem yang tidak dapat dilihat secara fisik dan biasanya berupa ide atau pemikiran. Contohnya adalah sistem yang berkaitan dengan hubungan antara manusia dan Tuhan, yang sering disebut sebagai sistem teologi. Di sisi lain, sistem fisik adalah sistem yang dapat dilihat dan dirasakan secara nyata, seperti sistem produksi, sistem akuntansi, sistem penjualan, sistem komputer, dan sistem inventaris.

#### 2) Sistem Alamiah dan Sistem Buatan Manusia

Sistem alamiah adalah sistem yang terbentuk melalui proses alami dan tidak diciptakan oleh manusia, seperti sistem perputaran bumi, sistem tata surya, pergantian musim, dan siklus siang dan malam. Sebaliknya, sistem buatan manusia adalah sistem yang melibatkan interaksi antara manusia dan mesin, contohnya adalah sistem informasi, di mana terdapat interaksi antara pengguna dan sistem (komputer).

#### 3) Sistem Deterministik dan Probabilistik

Sistem deterministik adalah sistem yang perilakunya dapat diprediksi karena telah diprogram oleh pembuatnya, sehingga hasilnya dapat diperkirakan. Sementara itu, sistem probabilistik adalah sistem yang perilakunya tidak dapat diprediksi di masa depan karena mengandung unsur ketidakpastian.

#### 4) Sistem Terbuka dan Tertutup

Sistem terbuka adalah sistem yang menerima masukan dari lingkungan luar dan menghasilkan keluaran untuk subsistem lainnya. Sebaliknya, sistem tertutup, yang juga dikenal sebagai sistem otomatis, adalah sistem yang tidak memiliki interaksi dengan lingkungan luar dan tidak dipengaruhi oleh faktor eksternal.

### 3. Informasi

#### a. Pengertian Data

Data adalah bentuk informasi mentah yang belum dapat memberikan pemahaman yang jelas tentang suatu hal, sehingga memerlukan pengolahan lebih lanjut (Sutabri, 2012). Data juga dapat dianggap sebagai representasi dari dunia nyata yang mewakili objek, peristiwa, atau kejadian yang direkam dalam berbagai bentuk, seperti huruf, angka, simbol, suara, gambar, atau kombinasi dari beberapa bentuk tersebut (Phkhoverishvili et al., 2019). Dengan demikian, dapat disimpulkan bahwa data adalah entitas nyata yang menggambarkan atau mewakili sesuatu. Data berfungsi sebagai bahan mentah yang belum memiliki makna tertentu atau dampak langsung bagi pengguna, sehingga perlu diolah lebih lanjut untuk menghasilkan informasi yang lebih berguna dan bermakna.

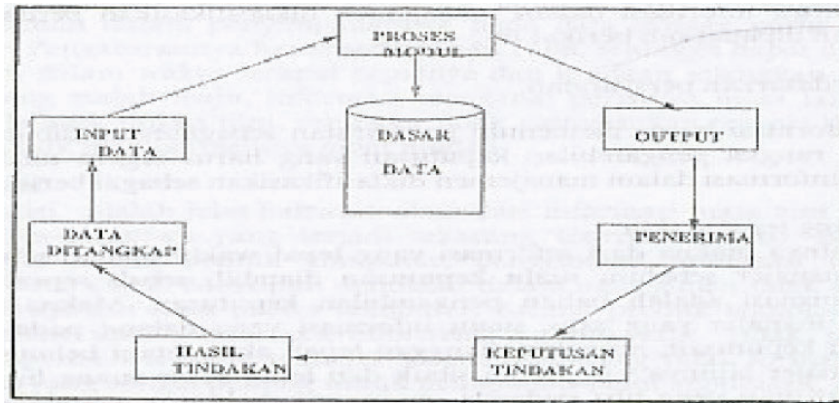
#### b. Pengertian Informasi

Informasi adalah data atau kumpulan data yang telah diinterpretasikan atau diolah sehingga dapat digunakan oleh pengguna dalam proses pengambilan keputusan (Sutabri, 2012). Sistem informasi berfungsi untuk mengolah data yang diterimanya dari masukan yang awalnya tidak berguna atau tidak memiliki makna, menjadi informasi yang bermanfaat bagi penerimanya. Nilai suatu informasi sangat terkait dengan keputusan yang diambil; jika tidak ada keputusan yang dihasilkan, maka informasi tersebut menjadi tidak relevan.

#### c. Siklus Informasi

Agar satu atau beberapa data dapat menjadi informasi, diperlukan suatu model atau siklus yang menjelaskan bagaimana proses tersebut berlangsung (Sutabri, 2012). Data diolah menggunakan model tertentu untuk menghasilkan informasi. Setelah itu, penerima informasi akan membuat keputusan berdasarkan informasi yang diterima, dan selanjutnya melakukan tindakan berdasarkan keputusan tersebut. Tindakan ini akan menghasilkan data baru, yang kemudian dimasukkan ke dalam sistem dan diolah kembali menggunakan model tertentu, sehingga membentuk siklus yang berkelanjutan.

Siklus ini dikenal sebagai siklus informasi atau siklus pengolahan data. Berikut adalah penjelasan mengenai siklus informasi tersebut:



Gambar 2. 2 Siklus Informasi

d. Kualitas Informasi

Informasi memiliki kualitas yang ditentukan oleh tiga aspek utama, yaitu keakuratan, ketepatan waktu, dan relevansi (Sutabri, 2012).

1) Akurat (*Accurate*)

Akurasi berarti bahwa informasi harus dengan jelas mencerminkan maksud yang ingin disampaikan. Informasi juga harus bebas dari kesalahan fakta dan tidak menyesatkan. Selama proses penyampaian dari sumber informasi ke penerima, informasi dapat mengalami gangguan (*noise*) yang dapat mengubah makna aslinya.

2) Tepat Waktu (*Timelines*)

Informasi yang dihasilkan harus sampai kepada penerima dalam waktu yang tepat dan tidak boleh terlambat. Informasi yang datang terlambat akan kehilangan nilai atau bahkan menjadi tidak berguna, karena informasi merupakan dasar dalam pengambilan keputusan. Keterlambatan dalam pengambilan keputusan dapat berdampak fatal bagi organisasi. Oleh karena itu, informasi memiliki nilai tinggi dan harus dikirim serta diterima dengan cepat, yang memerlukan

teknologi untuk memperoleh, mengolah, dan mengirimkannya tepat waktu.

### 3) Relevan (*Relevance*)

Informasi harus memberikan manfaat bagi penerimanya. Relevansi informasi dapat berbeda untuk setiap individu. Misalnya, menyampaikan informasi tentang keuntungan penjualan kepada bagian gudang perusahaan mungkin kurang relevan. Sebaliknya, informasi tersebut akan lebih relevan jika disampaikan kepada manajer atau pemimpin perusahaan.

## 4. Sistem Informasi

### a. Pengertian Sistem Informasi

Sistem informasi adalah suatu sistem dalam sebuah organisasi yang berfungsi untuk menghubungkan kebutuhan pengolahan transaksi dengan fungsi operasional organisasi, serta mendukung kegiatan strategis agar dapat menghasilkan laporan yang diperlukan baik oleh organisasi itu sendiri maupun oleh pihak luar (Sutabri, 2012). Dari pernyataan tersebut, dapat disimpulkan bahwa sistem informasi adalah sistem yang secara efektif mengolah data menjadi informasi yang berguna bagi organisasi tertentu.

### b. Komponen Sistem Informasi

Komponen sistem informasi dapat disebut sebagai "building blocks" (blok bangunan) yang terdiri dari blok masukan, blok model, blok keluaran, blok teknologi, blok basis data, dan blok kendali (Sutabri, 2012). Blok-blok ini saling berhubungan dan berinteraksi satu sama lain untuk membentuk suatu sistem yang bertujuan mencapai hasil tertentu. Berikut adalah penjelasan mengenai setiap blok:

#### 1) Blok Masukan (*Input Block*)

Blok masukan merujuk pada data yang masuk ke dalam sistem informasi. Input di sini adalah media atau metode yang digunakan untuk mengumpulkan data yang akan dimasukkan ke dalam sistem, yang dapat berupa dokumen-dokumen dasar.

2) Blok Model (*Model Block*)

Blok model adalah prosedur yang dilakukan untuk memanipulasi data dengan cara yang telah ditentukan sebelumnya, sehingga dapat dihasilkan keluaran yang diinginkan.

3) Blok Keluaran (*Output Block*)

Keluaran adalah produk hasil olahan dari suatu sistem, yang biasanya berupa informasi yang akan digunakan oleh penerimanya.

4) Blok Teknologi (*Technology Block*)

Teknologi berfungsi sebagai alat atau "toolbox" dalam sistem informasi. Terdiri dari tiga bagian utama: teknisi (brainware), perangkat keras (hardware), dan perangkat lunak (software).

5) Blok Basis Data (*Database Block*)

Basis data adalah kumpulan data yang saling berhubungan dan berkaitan satu sama lain. Perangkat lunak digunakan untuk memanipulasi data yang tersimpan di perangkat keras. Data perlu disimpan dalam basis data yang terorganisir dengan baik untuk menghasilkan informasi yang lebih berkualitas.

6) Blok Kendali (*Control Block*)

Pengendalian dalam sistem informasi sangat penting karena banyak faktor yang dapat merusak sistem, seperti bencana alam atau tindakan manusia yang merugikan, seperti peretasan atau manipulasi data. Oleh karena itu, pengendalian diperlukan untuk menentukan siapa yang dapat mengakses sistem dan untuk meningkatkan keamanan agar tidak dapat diakses oleh pihak luar. Selain itu, pengendalian juga mencakup langkah-langkah pemulihan jika terjadi kerusakan atau kesalahan, sehingga masalah dapat diselesaikan dengan cepat.

## 5. *Inventory*

### a. Pengertian *Inventory*

Inventaris (persediaan) adalah bahan atau barang yang disimpan di suatu tempat untuk tujuan tertentu, seperti untuk proses produksi yang mengubah bahan mentah menjadi produk jadi. Jika bahan atau barang yang

disimpan berupa komponen, maka barang tersebut akan dijual kembali sebagai barang dagangan (Rapidah & Khoirina, 2025). Menurut (Mufida et al., 2019), inventaris adalah sejumlah sumber daya, baik dalam bentuk bahan mentah maupun barang jadi, yang disediakan oleh perusahaan untuk memenuhi permintaan konsumen.

Dalam konteks penelitian ini, inventaris yang akan dikelola mencakup barang-barang elektronik dan non-elektronik, yang merupakan barang jadi yang langsung dijual kepada pelanggan, seperti handphone, aksesoris handphone, charger, kabel charger, dan produk lainnya yang disimpan untuk dijual kepada pelanggan.

b. Fungsi *Inventory*

Keberhasilan operasi produksi suatu perusahaan dapat ditingkatkan melalui fungsi-fungsi yang diberikan oleh inventaris. Beberapa fungsi inventaris antara lain (Abdulrohim & Marlina, 2022):

1) Memenuhi Antisipasi Permintaan Pelanggan

Inventaris membantu dalam manajemen stok untuk memenuhi antisipasi permintaan pelanggan, sehingga dapat menjaga kepuasan pelanggan.

2) Memisahkan Komponen Barang

Inventaris berfungsi untuk memisahkan komponen-komponen barang dalam proses produksi.

3) Mengatasi Fluktuasi Permintaan

Inventaris membantu memisahkan operasi perusahaan dari fluktuasi tingkat permintaan pelanggan, dengan menyediakan stok barang yang memungkinkan pendekatan yang lebih baik kepada pelanggan.

4) Memperlancar Operasi Produksi

Inventaris mendukung kelancaran kebutuhan yang digunakan dalam operasi produksi, terutama dalam menghadapi musim tertentu.

5) Mendapatkan Potongan Harga Kuantitas

Inventaris dapat berfungsi sebagai alat untuk mendapatkan potongan harga melalui pembelian bahan dalam jumlah besar, yang dapat mengurangi biaya pembelian.

6) Menjaga Kelangsungan Operasi Produksi

Inventaris berfungsi untuk memisahkan operasi produksi dari kejadian tertentu, berfungsi sebagai penyangga yang menjaga kelangsungan operasi dan mencegah kerusakan peralatan yang dapat mengakibatkan penghentian sementara produksi.

7) Menghindari Kekurangan Stok

Inventaris membantu perusahaan menghindari kekurangan stok akibat keterlambatan pemesanan, tidak memperhitungkan durasi pengiriman, dan fluktuasi permintaan pelanggan yang dapat menyebabkan kekurangan barang.

8) Melindungi dari Perubahan Harga dan Inflasi

Inventaris berfungsi untuk melindungi perusahaan dari dampak perubahan harga yang meningkat dan inflasi.

9) Memanfaatkan Keuntungan dari Siklus Pemesanan

Inventaris juga berfungsi untuk memanfaatkan keuntungan dari siklus pemesanan dengan cara mengontrol pembelian dan biaya persediaan.

10) Memungkinkan Penambahan Barang Secara Efisien

Inventaris memungkinkan perusahaan untuk beroperasi dengan baik saat ada penambahan barang, seperti memanfaatkan barang yang sedang dalam proses.

c. Jenis-jenis *Inventory*

Secara umum, persediaan atau inventaris dapat dibedakan menjadi lima jenis, yaitu (Siagian, 2005):

1) Persediaan bahan baku (*raw material*)

Ini adalah bahan atau barang yang akan diproses lebih lanjut menjadi barang setengah jadi atau barang jadi.

2) Persediaan barang dalam proses

Jenis ini mencakup persediaan yang telah mengalami perubahan tetapi belum selesai diproduksi.

3) *Supplies Inventory*

Ini adalah persediaan yang berfungsi untuk mendukung proses operasi atau produksi agar dapat berjalan dengan lancar.

4) Persediaan barang dagangan

Ini adalah persediaan yang menyimpan barang untuk dijual kembali sebagai barang dagangan.

5) Persediaan barang jadi

Ini adalah persediaan yang dihasilkan dari operasi atau produksi perusahaan yang telah selesai, dan barang tersebut masih disimpan di gudang perusahaan.

## 6. Sistem *Inventory*

Sistem inventaris adalah pengendalian dan serangkaian kebijakan yang digunakan untuk memantau dan menjaga tingkat persediaan, termasuk kapan stok harus ditambah dan berapa banyak barang yang perlu dipesan (Abdulrohman & Marlina, 2022). Sistem inventaris merupakan pendekatan manajerial dalam mengendalikan persediaan barang, yang mencakup pemantauan, pengendalian, dan perencanaan persediaan guna memastikan ketersediaan bahan atau produk yang optimal sesuai dengan kebutuhan organisasi. Tujuan utama dari sistem ini adalah untuk menjaga keseimbangan antara ketersediaan barang dan biaya penyimpanan, sehingga organisasi dapat menghindari kekurangan atau kelebihan stok yang berpotensi merugikan (Fitria et al., 2023).

Sistem inventaris memberikan keuntungan bagi struktur organisasi serta kebijakan operasi produksi dengan cara mengawasi dan menjaga stok barang. Dengan adanya sistem ini, diharapkan manajemen dapat bertanggung jawab atas pemesanan dan penerimaan barang yang telah dipesan. Hal ini mencakup pengawasan terhadap waktu pemesanan, serta pemantauan proses pemesanan, termasuk dari siapa barang dipesan dan berapa banyak yang dipesan (Assauri, 2016).

Sistem inventaris dapat dikategorikan ke dalam beberapa jenis, antara lain:

- 1) Sistem Inventaris Periodik – Pengecekan stok dilakukan pada interval waktu tertentu.

- 2) Sistem Inventaris Perpetual – Pemantauan stok dilakukan secara terus-menerus dengan teknologi seperti barcode atau RFID.
- 3) Just-in-Time (JIT) – Pendekatan yang mengurangi persediaan dengan hanya memesan barang saat dibutuhkan.
- 4) Material Requirements Planning (MRP) – Sistem berbasis perencanaan yang menentukan kebutuhan inventaris berdasarkan jadwal produksi.

## **7. Sistem Informasi *Inventory***

Sistem informasi inventaris adalah sistem yang dirancang untuk mengelola dan memantau persediaan barang dalam suatu organisasi. Menurut (Nugrahaningsih et al., 2024), sistem informasi inventaris dapat membantu perusahaan dalam mengoptimalkan pengelolaan persediaan, meningkatkan efisiensi operasional, dan menyediakan informasi yang akurat untuk pengambilan keputusan. Fitur umum dari sistem informasi inventaris mencakup pencatatan transaksi, pelacakan stok, dan pembuatan laporan analisis.

Dalam konteks ini, penelitian oleh (Harfizar et al., 2022) menunjukkan bahwa penerapan sistem informasi inventaris berbasis web dapat meningkatkan transparansi dan akuntabilitas dalam pengelolaan persediaan. Mereka menemukan bahwa sistem yang terintegrasi memungkinkan akses informasi secara real-time, yang sangat penting untuk pengambilan keputusan yang cepat dan tepat.

## **8. *Rapid Application Development (RAD)***

*Rapid Application Development (RAD)* adalah pendekatan berorientasi objek terhadap suatu metode pengembangan sistem serta perangkat-perangkat lunak. RAD memiliki tujuan untuk mempersingkat waktu yang biasanya diperlukan dalam siklus hidup pengembangan sistem tradisional antara penerapan dan perancangan suatu sistem informasi (Sikumbang et al., 2020). Pada akhirnya, RAD juga berusaha untuk memenuhi syarat-syarat bisnis yang berubah secara cepat. Peneliti menggunakan model RAD ini dikarenakan penghematan waktu dan biaya yang ditawarkannya, dikarenakan RAD

menerapkan metode berulang dimana model bekerja sistem dibangun sedemikian rupa di awal tahap pengembangan untuk menetapkan kebutuhan-kebutuhan pengguna. Dalam menentukan kebutuhan pengguna cara yang digunakan adalah bekerja sama dengan pengguna dalam membangun sistem sehingga hasil akhir yang didapat akan sesuai dan cepat. Gagasan-gagasan RAD adalah (Whitten, 2004):

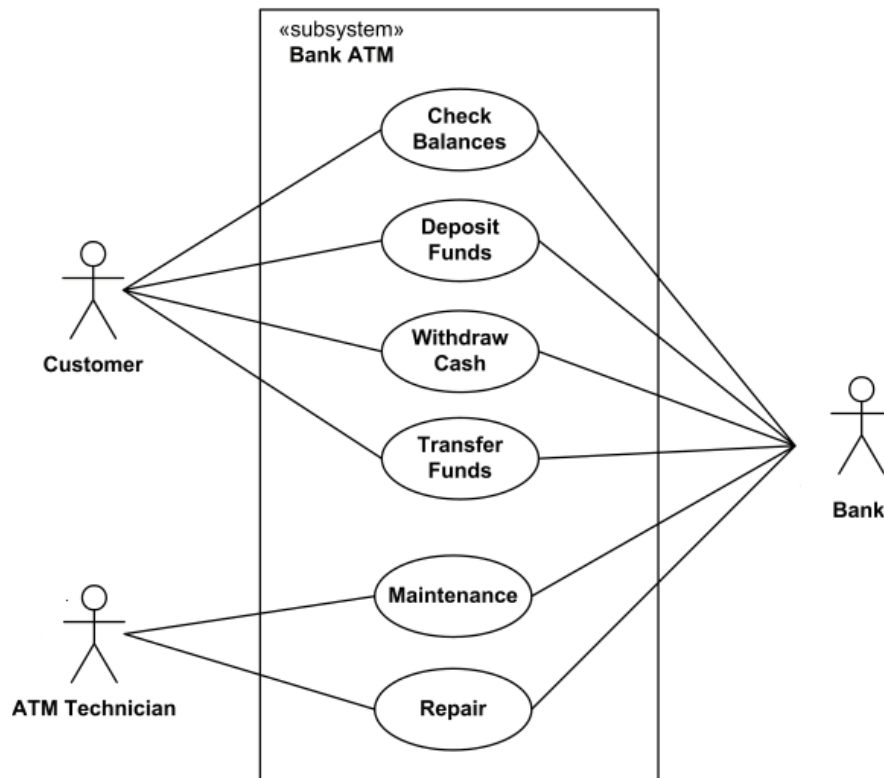
- a. Para pengguna sistem dilibatkan dalam aktifitas analisis, desain, dan konstruksi sehingga dinilai lebih aktif.
- b. Mengadakan serangkaian seminar yang intensif dalam mengorganisasikan pengembangan sistem dan berfokus dengan para pemilik, pengguna penganalisis, perancang, dan pembangunan sistem.
- c. Dengan pendekatan konstruksi berulang RAD dapat mengakselerasi fase-fase analisis dan desain persyaratan.
- d. Mempersingkat waktu yang diperlukan sebelum para pengguna mulai melihat sebuah sistem yang bekerja.



Gambar 2. 3 Siklus *Rapid Application Development*

## 9. *Unified Modeling Language (UML)*

Pemodelan adalah proses merancang perangkat lunak sebelum tahap pengkodean (coding) dilakukan. Model perangkat lunak dapat dianalogikan sebagai pembuatan rancangan atau cetak biru (blueprint) sebelum pembangunan fisik dilakukan (Hughes et al., 2022). Sistem yang kompleks dapat menyulitkan pemahaman secara menyeluruh, sehingga penting untuk membuat model dari sistem tersebut. Semakin kompleks sistem yang ingin dibangun, semakin penting pula penggunaan teknik pemodelan yang efektif. Unified Modeling Language (UML) adalah bahasa grafis atau bahasa pemodelan yang kompleks dan kaya fitur (Acharya, 2024). UML telah menjadi standar dalam industri untuk perancangan, pendokumentasian, dan visualisasi sistem perangkat lunak. Dengan UML, tersedia model yang telah diakui sebagai standar untuk perancangan sistem.





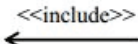
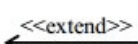


Gambar 2. 4 Contoh uml *use case diagram* atm

Sumber: (dicoding, 2021) <https://www.dicoding.com/blog/apa-itu-uml/>

## 10. Use case Diagram

*Diagram Use case* merupakan sebuah model untuk menggambarkan kelakuan (behavior) dari suatu sistem informasi yang akan dibuat (Acharya, 2024). Sebuah interaksi antara aktor dengan sistem yang akan dibuat dimana aktor dapat berupa satu aktor atau lebih akan dideskripsikan di *diagram Use case*. Sederhananya *Use case* menjelaskan fungsi-fungsi apa saja yang ada di dalam sistem dan siapa saja yang dapat menjalankan fungsi tersebut. Dalam memberikan nama untuk *Use case*, nama *Use case* tersebut harus didefinisikan sesederhana mungkin, menggunakan kata kerja dan mudah dipahami. Berikut merupakan simbol-simbol yang digunakan dalam *Use case* antara lain:

Simbol	Keterangan
	Aktor : Mewakili peran orang, sistem yang lain, atau alat ketika berkomunikasi dengan <i>use case</i>
	<i>Use case</i> : Abstraksi dan interaksi antara sistem dan aktor
	<i>Association</i> : Abstraksi dari penghubung antara aktor dengan <i>use case</i>
	<i>Generalisasi</i> : Menunjukkan spesialisasi aktor untuk dapat berpartisipasi dengan <i>use case</i>
	Menunjukkan bahwa suatu <i>use case</i> seluruhnya merupakan fungsionalitas dari <i>use case</i> lainnya
	Menunjukkan bahwa suatu <i>use case</i> merupakan tambahan fungsional dari <i>use case</i> lainnya jika suatu kondisi terpenuhi

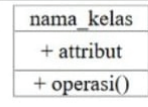



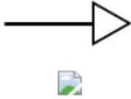


Gambar 2. 5 Simbol-simbol *Use case Diagram*

## 11. Class Diagram

*Class diagram* merupakan gambaran terstruktur sistem dari segi pendefinisian kelas kelas yang akan dibentuk untuk membangun suatu sistem (Acharya, 2024). *Class* memiliki atribut dan operasi atau metode sebagai berikut:

- Suatu *class* memiliki atribut yang merupakan variabel-variabel
- Atribut mendeskripsikan properti dengan sebaris teks di dalam kotak *class* tersebut.
- Operasi atau metode merupakan fungsi-fungsi yang dimiliki oleh suatu *class*.

*Class diagram* memiliki fungsi untuk mendeskripsikan jenis-jenis objek yang berada dalam sistem yang akan dibuat dan berbagai hubungan statis yang terdapat di antara mereka. *Class diagram* juga menjelaskan properti dan

Simbol	Nama	Keterangan
	Kelas	Kelas pada struktur sistem.
	Interface	Sama dengan konsep interface dalam pemrograman berorientasi objek.
	Association	Relasi antarclass dengan arti umum, asosiasi biasanya juga disertai dengan Multiplicity.
	Directed Association	Relasi antarkelas dengan makna kelas yang atau digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan multiplicity.
	Generalisasi	Relasi antarkelas dengan makna generalisasi-spesialisasi (umum khusus).
	Dependency	Relasi antarkelas dengan makna kebergantungan antarkelas'
	Aggregation	Relasi antarkelas dengan makna







Gambar 2. 6 Simbol-simbol *Class Diagram*

operasi sebuah dari sebuah *class* dan batasan-batasan apa saja yang terdapat dalam hubungan-hubungan antar objek tersebut. *Class diagram* menggambarkan struktur *class* (kelas), *package* (paket), dan objek kemudian dideskripsikan beserta hubungan-hubungannya seperti *containment*, asosiasi, pewarisan, dan lainnya. Adapun simbol-simbol yang digunakan dalam *class diagram* adalah sebagai berikut:

## 12. Sequence Diagram

*Sequence diagram* atau *diagram* sekuens merupakan sebuah *diagram* yang menggambarkan kelakuan (behavior) objek yang ada pada *Use case* dengan mendefinisikan waktu hidup (lifetime) dari objek tersebut dan pesan yang diterima dan dikirimkan antar objek (Acharya, 2024). Maka dari itu, dalam membuat *diagram* sekuens sebelumnya pembuat harus mengetahui objek-objek apa saja yang terlibat dalam sebuah *use case* beserta metode atau operasi apa yang dimiliki *class* yang diinstansiasi menjadi objek itu.

Penggambaran *diagram* sekuens sebanding dengan banyaknya pendefinisian *Use case* yang mempunyai proses sendiri atau semua *Use case* yang sudah didefinisikan interaksinya atau jalannya pesan. Dapat disimpulkan jika semakin banyak *Use case* yang didefinisikan, maka semakin banyak pula *diagram* sekuens yang harus dibuat. Simbol-simbol yang digunakan dalam *sequence diagram* adalah:







NO	GAMBAR	NAMA	KETERANGAN
1		<i>Actor</i>	Menggambarkan orang yang sedang berinteraksi dengan sistem.
2		<i>Entity Class</i>	Menggambarkan hubungan yang akan dilakukan
3		<i>Boundary Class</i>	Menggambarkan sebuah gambaran dari foem
4		<i>Control Class</i>	Menggambarkan penghubung antara boundary dengan tabel
5		<i>A focus of Control &amp; A Life Line</i>	Menggambarkan tempat mulai dan berakhirnya message
6		<i>A message</i>	Menggambarkan Pengiriman Pesan

Gambar 2. 7 Simbol-simbol *Sequence Diagram*

### 13. Activity diagram

*Activity diagram* atau *diagram* aktifitas merupakan gambar yang menjelaskan aktivitas atau aliran kerja dari sebuah sistem atau proses bisnis (Acharya, 2024). Ada hal yang harus diperhatikan disini yaitu bahwa *diagram* aktivitas menggambarkan aktivitas dari sistemnya bukan apa saja yang dilakukan aktor. *Diagram* aktivitas dapat mendukung perilaku paralel. *Activity diagram* menggambarkan aliran-aliran aktivitas di dalam sistem yang sedang dirancang yang menjelaskan bagaimana aliran-aliran berawal, keputusan yang mungkin terjadi, dan bagaimana aliran-aliran tersebut berakhir.

*Activity diagram* dalam UML merupakan *state diagram* khusus, dimana sebagian besar dari *state* tersebut adalah aksi dan sebagian besar perpindahannya dipicu oleh selesainya *state* sebelumnya. Oleh karena itu, *Activity diagram* tidak menggambarkan kelakuan internal suatu sistem dan interaksi antara satu subsistem dengan subsistem lainnya secara eksak, namun lebih menggambarkan jalur dan proses-proses aktifitas dari level atas secara umum. Adapun simbol-simbol yang digunakan dalam penggambaran *Activity diagram* adalah sebagai berikut:

Simbol	Nama	Keterangan
	Status awal	Sebuah diagram aktivitas memiliki sebuah status awal.
	Aktivitas	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
	Percabangan / Decision	Percabangan dimana ada pilihan aktivitas yang lebih dari satu.
	Penggabungan / Join	Penggabungan dimana yang mana lebih dari satu aktivitas lalu digabungkan jadi satu.
	Status Akhir	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir
	Swimlane	Swimlane memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi

Gambar 2. 8 Simbol-simbol *Activity Diagram*

## 14. Webiste

### a. Pengertian Website

Website adalah sebuah domain yang terdiri dari kumpulan halaman web yang mengandung berbagai informasi (Lin et al., 2020). Sebuah website dibangun dengan banyak halaman yang saling terhubung. Hubungan antar halaman ini disebut hyperlink, sedangkan teks yang berfungsi sebagai penghubung disebut hypertext. Halaman awal dari sebuah website dikenal sebagai homepage.

Domain adalah nama unik yang dimiliki oleh sebuah organisasi atau individu, yang memungkinkan akses melalui internet. Untuk mendapatkan domain yang diinginkan, biasanya diperlukan pendaftaran melalui penyedia jasa domain yang tersedia di berbagai situs web.

### b. Jenis-jenis Website

Dalam pengelompokan jenis-jenis website, biasanya dibedakan berdasarkan sifat, tujuan, dan bahasa pemrograman yang digunakan(Lin et al., 2020).

Jenis-jenis Website Berdasarkan sifatnya adalah:

#### 1) Website statis

Website ini memiliki konten yang tidak pernah atau sangat jarang diubah.

#### 2) Website dinamis

Website ini memiliki konten yang dapat diubah sesuai keinginan pemiliknya, sehingga kontennya sering diperbarui.

Jenis-jenis *website* berdasarkan tujuannya adalah sebagai berikut:

#### 1) *Personal* web

Website yang berisi informasi pribadi seseorang.

#### 2) *Corporate* web

Website yang dimiliki oleh sebuah perusahaan.

#### 3) *Portal* web

Website yang menyediakan berbagai layanan, seperti email, berita, dan jasa lainnya.

#### 4) *Forum* Web

Website yang dibuat sebagai media untuk diskusi.

Ditinjau dari segi bahasa pemrograman yang digunakan, website dapat dibedakan menjadi:

1) *Server Side*

Website ini menggunakan bahasa pemrograman yang bergantung pada ketersediaan server, seperti PHP (*Hypertext Preprocessor*), ASP (*Active Server Pages*), dan lainnya. Jika tidak ada server, website yang dibangun dengan bahasa pemrograman ini tidak akan berfungsi.

2) *Client side*

Website ini tidak memerlukan server untuk menjalankannya; cukup diakses melalui browser saja.

## 15. PHP: *Hypertext Preprocessor*

PHP adalah bahasa pemrograman yang dirancang untuk pengembangan aplikasi berbasis web (Kar et al., 2020). Meskipun PHP bukan satu-satunya bahasa pemrograman untuk web, ia termasuk salah satu yang paling populer. PHP memungkinkan penggunanya untuk membuat aplikasi web yang dinamis, di mana perubahan pada data dapat mengubah tampilan halaman web tanpa perlu mengubah kode (*script*) yang menyusun halaman tersebut.

PHP memiliki banyak kelebihan yang tidak dimiliki oleh bahasa pemrograman sejenis. PHP fokus pada pembuatan *script server-side*, yang dapat melakukan semua fungsi yang dapat dilakukan oleh CGI (*Common Gateway Interface*). PHP dapat dijalankan di berbagai sistem operasi, termasuk Linux, Windows, UNIX, MAC OS, dan RISC OS. Selain itu, PHP mendukung banyak web server, seperti Apache, PWS, MIIS, dan lainnya.

PHP tidak terbatas pada keluaran HTML saja; ia juga memiliki kemampuan untuk mengolah gambar, file movie flash, dan PDF (*Portable Document Format*). Selain itu, PHP dapat menghasilkan teks dalam format seperti XHTML dan file XML lainnya. Salah satu fitur andalan PHP adalah dukungannya terhadap berbagai database, termasuk Direct MS-SQL, Oracle, MySQL, ODBC, dan lainnya.

## 16. *Framework*

*Framework* adalah sekumpulan alat, pustaka, dan konvensi yang dirancang untuk memudahkan pengembangan perangkat lunak dengan menyediakan struktur dan fungsionalitas yang telah ditentukan sebelumnya. Dalam konteks pengembangan aplikasi, *framework* membantu pengembang untuk mengorganisir kode, mengurangi waktu pengembangan, dan meningkatkan kualitas perangkat lunak. Beberapa karakteristik dan manfaat dari penggunaan *framework* dalam bahasa pemrograman meliputi (Vyas, 2022):

### a. Struktur yang Terorganisir

*Framework* menyediakan struktur yang jelas untuk pengembangan aplikasi, memisahkan logika bisnis, antarmuka pengguna, dan pengelolaan data. Ini membantu pengembang untuk menjaga kode tetap terorganisir dan mudah dipahami.

### b. Pengurangan Waktu Pengembangan

Dengan menyediakan pustaka dan alat yang siap pakai, *framework* memungkinkan pengembang untuk fokus pada logika aplikasi daripada menulis kode dari awal. Ini mempercepat proses pengembangan dan memungkinkan pengembang untuk menyelesaikan proyek lebih cepat.

### c. Standarisasi

*Framework* sering kali mengikuti konvensi dan praktik terbaik yang telah terbukti efektif. Ini membantu dalam menciptakan kode yang konsisten dan mudah dipelihara, serta memudahkan kolaborasi antar pengembang.

### d. Keamanan

Banyak *framework* dilengkapi dengan fitur keamanan bawaan yang membantu melindungi aplikasi dari berbagai ancaman, seperti serangan SQL Injection, *Cross-Site Scripting* (XSS), dan *Cross-Site Request Forgery* (CSRF).

### e. Dukungan Komunitas

*Framework* yang populer biasanya memiliki komunitas yang besar dan aktif, yang menyediakan dokumentasi, tutorial, dan paket tambahan. Ini

memudahkan pengembang untuk mendapatkan bantuan dan sumber daya saat mengembangkan aplikasi.

## 17. Laravel

### a. Pengertian Laravel

Laravel adalah sebuah *framework* open-source untuk pengembangan aplikasi web berbasis PHP yang dirancang untuk memudahkan proses pengembangan dengan menyediakan berbagai fitur dan alat yang membantu pengembang dalam membangun aplikasi yang berskala besar dan kompleks. Laravel pertama kali diperkenalkan oleh Taylor Otwell pada tahun 2011 dan sejak saat itu telah menjadi salah satu *framework* PHP yang paling banyak digunakan di dunia (Khan & Khanam, 2023).

### b. Fitur Utama Laravel

Laravel menawarkan berbagai fitur yang membuatnya menjadi pilihan populer di kalangan pengembang. Beberapa fitur utama Laravel meliputi:

- *Routing* yang Sederhana: Laravel menyediakan sistem routing yang intuitif dan mudah digunakan, memungkinkan pengembang untuk mendefinisikan rute aplikasi dengan cepat dan jelas.
- *Eloquent ORM*: Laravel dilengkapi dengan Eloquent, sebuah *Object-Relational Mapping* (ORM) yang memudahkan pengembang dalam berinteraksi dengan database. *Eloquent* memungkinkan penggunaan model untuk mengelola data dan relasi antar tabel dengan sintaks yang sederhana.
- *Blade Templating Engine*: Laravel menggunakan Blade sebagai mesin templating yang memungkinkan pengembang untuk membuat tampilan yang dinamis dengan sintaks yang bersih dan mudah dibaca.
- *Middleware*: Laravel mendukung penggunaan *middleware*, yang memungkinkan pengembang untuk menyisipkan logika tertentu di antara permintaan HTTP dan respons. Ini berguna untuk autentikasi, logging, dan pengolahan permintaan.

- Artisan CLI: Laravel menyediakan *command-line interface* (CLI) yang disebut Artisan, yang memungkinkan pengembang untuk menjalankan berbagai perintah dan tugas otomatis, seperti migrasi database, pembuatan model, dan pengujian.
- Keamanan: Laravel memiliki fitur keamanan yang kuat, termasuk perlindungan terhadap serangan CSRF (*Cross-Site Request Forgery*), XSS (*Cross-Site Scripting*), dan SQL Injection.

#### c. Arsitektur Laravel

Laravel mengikuti arsitektur *Model-View-Controller* (MVC), yang memisahkan logika aplikasi menjadi tiga komponen utama:

- *Model*: Dapat bertanggung jawab untuk mengelola data dan juga logika bisnis. *Model* berinteraksi dengan database dan menyediakan data yang diperlukan untuk tampilan.
- *View*: Merupakan antarmuka pengguna yang menampilkan data kepada pengguna. *View* bertanggung jawab untuk menyajikan informasi dengan menggunakan Blade templating engine.
- *Controller*: Menghubungkan *model* dan *view*. *Controller* menerima permintaan dari pengguna, memproses data melalui *model*, dan mengembalikan tampilan yang sesuai.

#### d. Kelebihan Laravel

Laravel memiliki sejumlah kelebihan yang menjadikannya pilihan utama bagi banyak pengembang (Khan & Khanam, 2023):

- Kemudahan Penggunaan: Laravel dirancang untuk memudahkan pengembang, dengan dokumentasi yang lengkap dan komunitas yang aktif. Ini membuatnya lebih mudah untuk dipelajari dan digunakan, bahkan bagi pemula.
- Fleksibilitas dan Skalabilitas: Laravel dapat digunakan untuk membangun berbagai jenis aplikasi, mulai dari aplikasi kecil hingga aplikasi besar dan kompleks. *Framework* ini mendukung pengembangan yang fleksibel dan dapat diskalakan.

- Dukungan Komunitas: Laravel memiliki komunitas yang besar dan aktif, yang menyediakan berbagai paket dan plugin yang dapat digunakan untuk memperluas fungsionalitas aplikasi.
- Pengembangan yang Cepat: Dengan fitur-fitur seperti Eloquent ORM, Artisan CLI, dan routing yang sederhana, pengembang dapat membangun aplikasi dengan lebih cepat dan efisien.

## 18. Database

### a. Pengertian Database

Sistem basis data adalah sistem terkomputasi yang bertujuan untuk memelihara dan menjaga data, baik yang belum diolah maupun yang sudah diolah (informasi), sehingga informasi tersebut dapat tersedia saat dibutuhkan (Ajah & Nweke, 2019). Database berfungsi sebagai media untuk menyimpan dan mengorganisir data agar dapat diakses dengan mudah dan cepat.

Database merupakan kumpulan data yang terintegrasi dan diatur sedemikian rupa sehingga data tersebut dapat dicari, diambil, dan dimanipulasi dengan mudah dan cepat (Ali, 2020). Perangkat lunak yang digunakan untuk mengelola dan menjalankan kueri pada basis data disebut sistem manajemen basis data (Database Management System), yang sering disingkat DBMS. Beberapa perangkat lunak basis data yang banyak digunakan dalam pemrograman antara lain Oracle, MySQL, Microsoft SQL Server, Microsoft Access, Paradox, FoxPro, Firebird, dan masih banyak lagi.

### b. MYSQL

MySQL adalah perangkat lunak yang termasuk dalam kategori DBMS (Database Management System) dan bersifat Open Source. Open Source berarti perangkat lunak ini dilengkapi dengan kode sumber yang dapat diakses dan dimodifikasi. MySQL awalnya dikembangkan oleh perusahaan konsultan bernama TcX yang berlokasi di Swedia.

Menurut (Raharjo, 2015), MySQL adalah perangkat lunak DBMS yang mampu mengelola database dengan sangat cepat. MySQL memiliki

kemampuan untuk menampung data dalam jumlah yang sangat besar, dapat diakses oleh banyak pengguna (multiuser), dan mendukung proses yang dilakukan secara bersamaan (multi-threaded).

Fitur-fitur yang terdapat pada MySQL adalah sebagai berikut:

1) *Multiplatform*

MySQL tersedia di berbagai platform, termasuk Windows, Linux, Unix, MacOS, dan lainnya.

2) Andal, cepat dan mudah digunakan

MySQL tergolong sebagai database server yang andal. Ia mampu menangani basis data berukuran besar dengan kecepatan tinggi dalam aksesnya. MySQL juga mendukung banyak fungsi untuk mengakses database dan sangat mudah digunakan.

3) Jaminan keamanan akses

MySQL menyediakan pengamanan database dengan berbagai kriteria untuk aksesnya.

4) Dukungan SQL

MySQL mendukung perintah SQL (*Structured Query Language*), yang merupakan standar untuk mengakses database relasional.

c. *Mapping Cardinality*

*Mapping Cardinality* adalah hubungan antar entitas yang mencakup suatu relasi atau relationship (Ajah & Nweke, 2019). Jenis-jenis *Mapping Cardinality* dibagi menjadi tiga kategori, yaitu:

1) *One to Many* (1-M)

One to Many adalah perbandingan satu entitas dengan banyak entitas lainnya.

2) *One to One* (1-1)

One to One adalah perbandingan satu entitas dengan satu entitas lainnya.

3) *Many to Many* (M-M)

Many to Many adalah perbandingan antara satu entitas dengan banyak entitas lainnya.

## 19. *User interface*

*User interface* atau antarmuka pengguna adalah cara di mana suatu program berinteraksi dengan pengguna (Indrayanti et al., 2021). *Human Computer Interaction* (HCI) adalah istilah yang setara dengan *user interface*, mencakup semua aspek interaksi antara pengguna dan komputer. Semua yang terlihat di layar, aktivitas membaca atau memanipulasi informasi menggunakan keyboard, serta proses menerjemahkan informasi atau menghubungkan pengguna dengan sistem operasi, memungkinkan komputer untuk digunakan.

*Wireframe* adalah representasi visual awal dari suatu halaman atau aplikasi yang menunjukkan struktur dasar elemen-elemen UI tanpa detail desain yang kompleks. *Wireframe* digunakan dalam tahap perancangan untuk mendefinisikan tata letak, navigasi, dan hierarki informasi sebelum pengkodean atau pembuatan prototipe visual yang lebih lanjut.

Menurut Setiyawati, (Setiyawati et al., 2022) dalam jurnal *User Experience Design on Visualization of Mobile-Based Land Monitoring System Using a User-Centered Design Approach*, *wireframe* adalah alat yang efektif dalam menyampaikan kebutuhan pengguna dalam perancangan aplikasi berbasis seluler dengan pendekatan User-Centered Design.

Penelitian lain oleh (Alao et al., 2022) dalam makalah *User-centered/user experience Uc/Ux design thinking approach for designing a university information management system* menyoroti bagaimana *wireframe* membantu meningkatkan pemahaman tim pengembang terhadap permasalahan pengguna dan solusi desain yang sesuai.

## 20. **Black Box Testing**

Black box testing merupakan pengujian yang berfokus pada spesifikasi fungsional dari perangkat lunak, seorang tester dapat mendefinisikan kumpulan kondisi input dan melakukan pengetesan pada spesifikasi fungsional program. Black box testing merupakan pengujian yang memungkinkan seorang software engineer dan Tester mendapatkan serangkaian kondisi input yang sepenuhnya menggunakan semua persyaratan fungsional untuk suatu program. Black box merupakan salah satu metode pengujian yang tidak perlu

melihat dan menguji source code program. Black box testing bekerja dengan mengabaikan struktural internal pada software sehingga perhatiannya berfokus pada interface saja atau input dan output pada software. Blackbox testing merupakan pengujian yang bertujuan melihat program tersebut sama dengan tugas program tersebut tanpa harus mengetahui kode program yang di pakai (Dika Pratama & Noviansyah Dadaprawira, 2023).

## 21. Software Ideas Modeler

Software Ideas Modeler adalah alat CASE yang ringan dan canggih. Alat ini membantu Anda membuat berbagai diagram, seperti diagram UML dan ERD. ERD dapat diekspor ke SQL untuk membuat basis data. Alat ini juga menyediakan dukungan rekayasa balik basis data. Pilih tabel basis data atau berkas kode sumber, dan alat ini akan mengubah data ke bentuk yang lebih mudah dibaca – diagram.

Software Ideas Modeler menyediakan berbagai fitur untuk memodelkan sistem perangkat lunak menggunakan beragam diagram seperti UML, ERD, BPMN, dan diagram lainnya. Tool ini mendukung pembuatan, pengelolaan, serta ekspor berbagai jenis diagram yang umum digunakan dalam rekayasa perangkat lunak, dan dapat membantu proses desain serta dokumentasi perangkat lunak secara visual (Rodina, n.d.).

## B. Penelitian Relevan

Berikut adalah beberapa literatur yang telah dijadikan sebagai penelitian sebelumnya dan relevan:

### 1. Rancang Bangun Sistem Informasi *Inventory* Pada Salon Kecantikan

Jurnal dengan judul "Rancang Bangun Sistem Informasi *Inventory* pada Salon Kecantikan" membahas pengembangan sistem informasi *Inventory* berbasis web yang dirancang untuk mengatasi kendala pengelolaan stok barang secara manual, seperti human error, ketidaksesuaian data, dan lambatnya proses pengolahan informasi. Penelitian ini menggunakan metode pengembangan sistem Waterfall, yang meliputi tahapan analisis kebutuhan, perancangan sistem, implementasi, pengujian, dan penerapan. Pada tahap

perancangan, digunakan alat bantu seperti Unified Modeling Language (UML) untuk menggambarkan *Use case diagram*, *class diagram*, *sequence diagram*, dan *Activity diagram*, serta Entity Relationship Diagram (ERD) untuk mendesain hubungan antar entitas dalam database. Sistem ini dibangun dengan bahasa pemrograman PHP menggunakan *framework* CodeIgniter, database MySQL, dan server lokal berbasis XAMPP.

Hasilnya, sistem informasi *Inventory* yang dihasilkan mampu memberikan informasi stok barang secara cepat dan akurat, menghasilkan laporan stok otomatis, serta mempermudah proses monitoring stok barang masuk dan keluar (Mufida et al., 2019). Penelitian ini relevan dengan pengembangan sistem informasi *Inventory* berbasis web untuk Konter +62 Luragung Kuningan karena menawarkan pendekatan metodologi, pemilihan teknologi, dan fitur utama yang serupa, seperti efisiensi dalam pengelolaan stok dan pengurangan kesalahan manual.

## **2. Rancang Bangun Sistem Informasi *Inventory* Barang Berbasis Web (Studi Kasus Toko Dina Beauty Care)**

Jurnal berjudul "Rancang Bangun Sistem Informasi *Inventory* Barang Berbasis Web (Studi Kasus Toko Dina Beauty Care)" membahas pengembangan sistem berbasis web untuk mengelola inventaris produk kecantikan di Toko Dina Beauty Care. Penelitian ini menggunakan metode pengembangan perangkat lunak Waterfall, yang mencakup tahapan pengumpulan kebutuhan, perancangan sistem dengan UML (Unified Modeling Language), implementasi menggunakan *framework* CodeIgniter dan database MySQL, pengujian melalui *Blackbox Testing*, serta pemeliharaan sistem. Sistem yang dihasilkan memiliki fitur utama seperti pelacakan produk, manajemen stok, dan pembuatan laporan otomatis yang terintegrasi.

Hasil penelitian menunjukkan bahwa sistem ini berhasil meningkatkan akurasi data inventaris, mengurangi risiko kesalahan manual, dan mempermudah pengambilan keputusan berbasis laporan yang komprehensif (Rura & Ardiansyah, 2024). Penelitian ini relevan dengan pengembangan sistem informasi *Inventory* barang berbasis web untuk Konter +62 Luragung Kuningan, karena menawarkan pendekatan metodologi dan teknologi yang

dapat diadaptasi untuk kebutuhan usaha ritel, serta fokus pada peningkatan efisiensi operasional dan kepuasan pelanggan.

### **3. Rancang Bangun Sistem Informasi Persediaan Barang Berbasis Web Dengan Metode Fast (*Framework for The Applications*)**

Jurnal berjudul “Rancang Bangun Sistem Informasi Persediaan Barang Berbasis Web Dengan Metode Fast (*Framework For The Applications*)” membahas pengembangan sistem informasi persediaan barang berbasis website pada perusahaan NetZap layanan provider dari PT. Solusi Aksesindo Pratama perusahaan yang bergerak di bidang jasa internet dan penyedia sparepart jaringan yang dimana pada pt tersebut untuk sistem pengelolaan dan persediaan barang masih dilakukan secara manual mulai dari pencatatan barang masuk, permintaan barang dari bagian sales, proses barang keluar oleh bagian gudang, sampai kepada pembuatan laporan, sehingga memungkinkan pada saat proses berlangsung terjadi kesalahan-kesalahan dalam pencarian data-data yang diperlukan. Penelitian ini menggunakan metode pengembangan sistem yaitu FAST (*Framework for the Application System Thinking*) terdiri dari fase-fase Scope Definition, Problem Analysis, Requirements Analysis, Logical Design dan Physical Design dan perancangan informasinya menggunakan PHP dan HTML serta MySQL sebagai databasenya.

Hasil penelitian menunjukkan bahwa sistem ini berhasil menyelesaikan permasalahan-permasalahan sistem persediaan barang secara manual dan sekarang sudah tekomputerisasi dengan ketentuan hasil perancangan ini didukung partisipasi aktif dari pemakai sistem, terutama kedisiplinan para pelaksana yang menangani secara langsung pada sistem yang dirancang. Dengan sistem yang terkomputasi ini pengolahan data untuk keluar masuk barang lebih efektif dan efisien, pencarian data dapat lebih efisien karena data sudah terorganisir dengan baik sesuai dengan level akses admin, sales dan gudang, data atau stok barang dapat terpantau dengan baik, file data barang masuk dan keluar tersimpan dengan baik dan dapat diakses dengan mudah ketika dibutuhkan, dan proses pelaporan menjadi lebih baik, karena dapat diakses dan dicetak langsung (Oktarini Sari & Nuari, 2019). Penelitian ini relevan dengan pengembangan sistem informasi *Inventory* berbasis web untuk

Konter +62 Luragung Kuningan karena menawarkan pemilihan teknologi, dan fitur utama yang serupa, seperti efisiensi dalam pengelolaan stok dan pengurangan kesalahan manual.

#### **4. Rancang Bangun Penerapan Model Prototype Dalam Perancangan Sistem Informasi Pencatatan Persediaan Barang Berbasis Web**

Jurnal berjudul “Rancang Bangun Penerapan Model Prototype Dalam Perancangan Sistem Informasi Pencatatan Persediaan Barang Berbasis Web” oleh Jehan Saptia Kurnia dan Fitria Risyda membahas pentingnya sistem informasi dalam pengelolaan persediaan barang di perusahaan, khususnya di PT. ABC, yang saat ini masih menggunakan Microsoft Excel untuk pencatatan. Penelitian ini mengidentifikasi berbagai masalah, seperti data redundansi dan ketidaksesuaian data, serta menawarkan solusi melalui pengembangan sistem informasi berbasis web dengan menggunakan metode prototype. Metode ini meliputi pengumpulan kebutuhan, pembangunan prototype, evaluasi, pengkodean, pengujian, dan evaluasi sistem, yang memungkinkan interaksi yang lebih baik antara pengembang dan pengguna. Hasil yang diharapkan dari penelitian ini adalah peningkatan efisiensi dan akurasi dalam pengelolaan data persediaan barang. Jurnal ini juga membahas berbagai jenis persediaan dan pentingnya sistem informasi dalam pengelolaan tersebut, serta kelebihan dan kekurangan dari metode prototype yang digunakan. Meskipun jurnal ini memberikan kontribusi signifikan dalam bidang sistem informasi, terdapat kekurangan, seperti kurangnya data empiris atau studi kasus yang mendukung klaim tentang peningkatan efisiensi dan akurasi. Oleh karena itu, disarankan agar penulis menyertakan data pengujian sistem dan analisis yang lebih mendalam mengenai dampak implementasi sistem terhadap kinerja perusahaan (Saptia Kurnia & Risyda, 2019).

Penelitian ini relevan dengan pengembangan sistem informasi *Inventory* barang pada konter +62 Luragung Kuningan karena Keduanya berfokus pada pengembangan sistem informasi untuk pengelolaan *Inventory* atau persediaan barang, namun bedanya menggunakan metode prototype yang memungkinkan pengujian dan umpan balik dari pengguna. Namun jurnal ini serupa mengembangkan sistem berbasis web, mengikuti tren teknologi terkini. Jurnal

ini juga mengidentifikasi masalah yang mungkin relevan dengan konteks di Konter +62 Luragung Kuningan, seperti data redundansi dan ketidaksesuaian data, yang dapat membantu merancang solusi yang lebih efektif. Dengan demikian, hasil dari penelitian ini dapat memberikan dampak positif yang serupa dalam konteks yang berbeda, menjadikan jurnal ini sebagai sumber yang berharga untuk mendukung argumen dan analisis dalam skripsi ini.

#### **5. Rancang Bangun Sistem Informasi *Inventory* Barang (Sinbar) Berbasis Website**

Jurnal yang berjudul "Rancang Bangun Sistem Informasi *Inventory* Barang (SINBAR) Berbasis Website" oleh Agyztia Premana membahas pengembangan sistem informasi untuk manajemen *Inventory* barang di UPT Pusat Teknologi Informasi dan Komunikasi Universitas Muhadi Setiabudi. Penelitian ini dilatarbelakangi oleh permasalahan yang dihadapi dalam pengelolaan data barang yang masih dilakukan secara manual, yang mengakibatkan kesulitan dalam perhitungan dan ketidakakuratan informasi ketersediaan barang. Dengan menggunakan metode prototype dan pemodelan Data Flow *Diagram* (DFD), penelitian ini bertujuan untuk merancang dan membangun sistem informasi yang dapat mengelola barang masuk dan keluar, serta stok barang secara lebih efisien.

Hasil dari penelitian ini menunjukkan bahwa sistem yang dibangun dapat meningkatkan kecepatan dan akurasi dalam pengelolaan data *Inventory*, serta memberikan kemudahan akses informasi bagi pengguna. Jurnal ini juga menyoroti pentingnya teknologi informasi dalam meningkatkan kualitas layanan dan efisiensi operasional di institusi pendidikan. Meskipun penelitian ini memberikan kontribusi yang signifikan dalam pengembangan sistem informasi *Inventory*, terdapat beberapa kekurangan, seperti kurangnya analisis mendalam mengenai dampak implementasi sistem terhadap kinerja operasional secara keseluruhan (Premana, 2019). Penelitian ini relevan dengan pengembangan sistem informasi *Inventory* berbasis web untuk Konter +62 Luragung Kuningan karena menawarkan pentingnya teknologi informasi dalam meningkatkan kualitas layanan dan efisiensi operasional di sebuah institusi ataupun perusahaan, dan fitur utama yang serupa.

Dari penelitian terdahulu yang telah dijelaskan maka dapat disimpulkan persamaan dan perbedaan dengan pengembangan yang akan pengembang lakukan yaitu:

**Persamaan:**

1. Tujuan Umum: Baik penelitian yang akan dilakukan maupun penelitian sebelumnya memiliki tujuan umum yang serupa, yaitu mengintegrasikan teknologi informasi dalam *management Inventory* barang, lalu meningkatkan efektivitas dan efisiensi dalam pengelolaan barang masuk dan keluar, hingga pelaporan secara otomatis dan terintegrasi untuk meminimalisir kesalahan pengelolaan secara manual.
2. Alat Bantu: Kedua penelitian diatas menggunakan alat bantu perancangan yaitu UML untuk memvisualisasikan, mendefinisikan, membangun, dan mendokumentasikan sistem pengembangan perangkat lunak berbasis berorientasi objek.

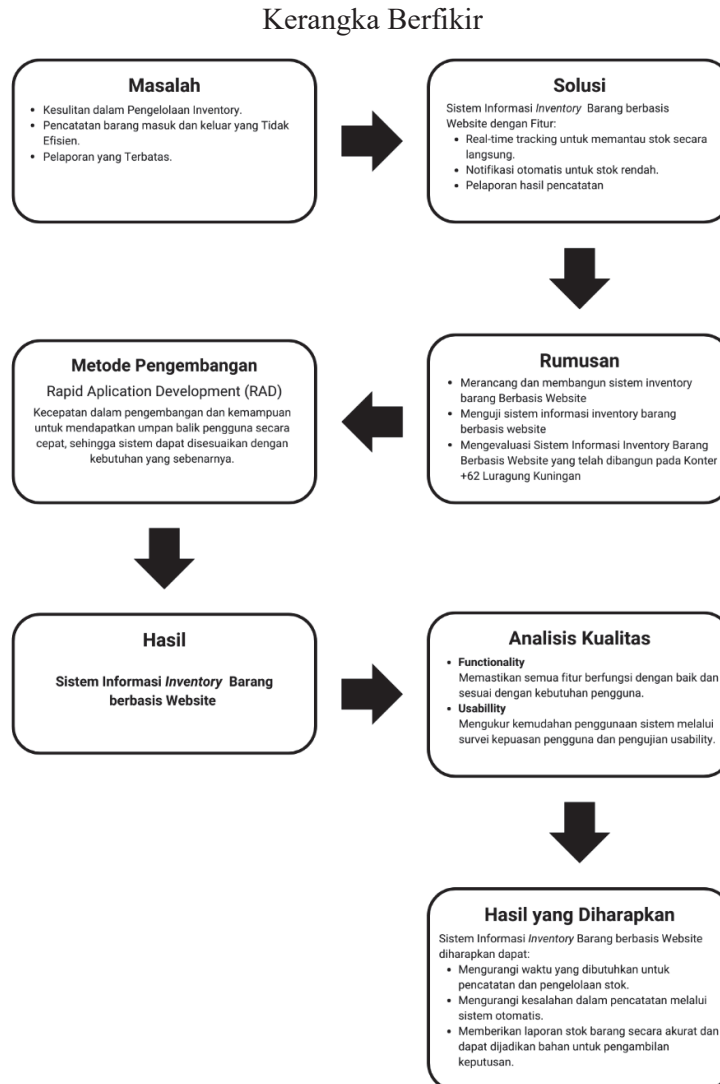
**Perbedaan:**

1. Teknologi yang digunakan: Penelitian sebelumnya menggunakan PHP, MySQL, dan *framework* codeigniter, sedangkan penelitian yang akan dilakukan juga mencakup PHP dan MySQL namun dengan pembaruan teknologi yang digunakan yaitu *framework* Laravel 11.
2. Objek Penelitian: Objek dari penelitian sebelumnya dengan yang akan pengembang kembangkan berbeda. Penelitian sebelumnya melibatkan perusahaan salon kecantikan, body care, PT. Solusi Aksesindo, PT. ABC, UPT Pusat Teknologi Informasi dan Komunikasi Universitas Muhadi Setiabudi. Sementara pengembangan yang akan dilakukan fokus pada Konter +62 Luragung Kuningan saja.
3. Tingkat Kelayakan: Penelitian sebelumnya menunjukkan tingkat kelayakan yang tinggi, sementara pengembangan yang akan dilakukan belum memberikan hasil kelayakan karena masih dalam tahap pelaksanaan.

Dengan demikian, meskipun ada beberapa persamaan dalam tujuan dan metode, terdapat perbedaan signifikan dalam teknologi yang digunakan, objek penelitian, dan tingkat kematangan pengembangan antara kedua penelitian tersebut.

### C. Kerangka Berfikir

Kerangka berpikir dalam penelitian ini bertujuan untuk memberikan pemahaman yang jelas mengenai pengembangan sistem informasi *Inventory* berbasis web. Kerangka berpikir ini terdiri dari beberapa komponen yang saling terkait, sebagai berikut:



Gambar 2. 9 Peta Konsep Kerangka Berfikir

Bagan kerangka berpikir ini menggambarkan proses pengembangan sistem informasi *Inventory* berbasis website. Dimulai dengan identifikasi masalah menggunakan lembar observasi dan angket wawancara, dengan hasil adanya kesulitan dalam pengelolaan *Inventory*, pencatatan barang yang tidak efisien, dan

terbatasnya pelaporan. Untuk mengatasi masalah ini, solusi yang diusulkan adalah pengembangan sistem informasi *Inventory* yang dapat melakukan real-time tracking, memberikan notifikasi otomatis untuk stok rendah, dan menghasilkan laporan stok barang yang akurat.

Metode yang digunakan dalam pengembangan sistem ini adalah *Rapid Application Development* (RAD), dengan tahapan *requirements planning*, *workshop design*, dan *implementation* yang memungkinkan pengembang untuk mendapatkan umpan balik pengguna dengan cepat, sehingga sistem dapat disesuaikan dengan kebutuhan yang sebenarnya.

Analisis kualitas dilakukan untuk memastikan bahwa semua fitur berfungsi dengan baik dan sesuai dengan kebutuhan pengguna menggunakan instrumen uji *functionality* oleh ahli di bidang TIK, serta mengukur kemudahan penggunaan sistem melalui survei kepuasan pengguna menggunakan angket *Usability* yang diisi oleh pengguna. Hasil yang diharapkan dari sistem ini adalah pengurangan waktu dalam pencatatan, pengurangan kesalahan, dan penyediaan laporan stok yang akurat untuk pengambilan keputusan.