

BAB II

KAJIAN PUSTAKA

2.1 Teori-teori terkait bahasan penelitian (Relevan Theories)

Bab ini menjelaskan teori-teori yang mendukung penelitian, hingga konsep-konsep pendukung yang relevan. Selain itu, teori-teori ini juga akan dihubungkan dengan konteks permasalahan yang diteliti, sehingga dapat memberikan landasan yang kuat untuk analisis data. Dengan memahami teori-teori tersebut, peneliti dapat mengembangkan perspektif yang komprehensif dan sistematis, yang pada akhirnya akan memudahkan dalam menjawab pertanyaan penelitian serta memberikan kontribusi terhadap perkembangan ilmu pengetahuan di bidang ini.

2.1.1 Konsep Dasar Sistem dan Sistem Informasi

Menurut Suhendi & Ali (2020), sistem merupakan sebuah kesatuan yang terdiri dari kumpulan beberapa komponen yang diatur sedemikian rupa agar saling terintegrasi. Sinergi antar elemen ini bertujuan untuk mencapai sasaran tertentu sehingga seluruh bagian dapat berfungsi secara efektif dalam satu alur kerja yang terorganisir. Melalui pengorganisasian elemen yang tepat, suatu sistem mampu mentransformasikan berbagai masukan menjadi keluaran yang konsisten dan terukur sesuai dengan standar yang ditetapkan.

Menurut Frisdayanti (2019), sistem informasi didefinisikan sebagai sebuah kesatuan komponen yang saling terintegrasi untuk menjalankan fungsi pengumpulan, pemrosesan, penyimpanan, hingga pendistribusian informasi. Keberadaan sistem ini bertujuan untuk menjadi instrumen pendukung dalam pengambilan keputusan serta sebagai sarana pengendalian operasional di dalam suatu organisasi. Hal ini memungkinkan terciptanya efisiensi kerja melalui penyediaan informasi yang akurat dan tepat waktu bagi seluruh pihak yang membutuhkan di dalam lingkup manajemen.

Merujuk pada penelitian Bratha (2022) mengenai arsitektur teknologi, sistem informasi terdiri dari unsur-unsur yang diintegrasikan dengan baik untuk menghasilkan informasi berkualitas. Komponen-komponen tersebut meliputi:

- a. *Hardware*: Peralatan fisik yang digunakan dalam proses pengumpulan, pemasukan, penyimpanan, dan pengeluaran hasil pengolahan data menjadi informasi.
- b. *Software*: Kumpulan program yang digunakan untuk menjalankan komputer atau aplikasi tertentu pada sebuah perangkat.
- c. *Brainware*: Bagian terpenting atau utama yang berperan sebagai pengelola dan pengguna dalam sistem informasi manajemen.
- d. *Prosedur*: Rangkaian aktivitas atau kegiatan yang dilakukan secara berulang-ulang dengan metode yang sama untuk memproses data.
- e. *Basis Data (Database)*: Pengorganisasian sejumlah data yang saling terkait sehingga memudahkan proses pencarian informasi.
- f. *Jaringan Komputer dan Komunikasi Data*: Infrastruktur yang menghubungkan berbagai komponen agar data dapat saling terintegrasi dan didistribusikan.

Dari pengertian-pengertian di atas, dapat disimpulkan bahwa sistem informasi merupakan suatu kesatuan komponen teknis dan manajerial yang meliputi *hardware*, *software*, *brainware*, *prosedur*, *basis data*, serta jaringan yang diintegrasikan untuk mentransformasikan data mentah menjadi informasi yang akurat, relevan, dan tepat waktu. Dalam konteks sistem informasi manajemen, keterpaduan antar elemen ini tidak hanya berfungsi sebagai alat pengolah data, melainkan sebagai instrumen strategis untuk mengendalikan operasional organisasi. Oleh karena itu, efektivitas suatu sistem informasi sangat bergantung pada sejauh mana seluruh komponen tersebut mampu bersinergi dalam

menghasilkan informasi yang dapat diandalkan bagi pengambilan keputusan manajerial.

2.1.2 Konsep dan Karakteristik Aplikasi yang Dikembangkan

Domain sistem yang diteliti dalam penelitian ini adalah sistem informasi manajemen rantai pasok (*Supply Chain Management*) yang menerapkan metode pengendalian persediaan *Safety Stock*. Sistem informasi SCM digunakan untuk mengelola data yang berkaitan dengan aktivitas pengadaan bahan baku, meliputi data pemasok (*supplier*), manajemen kategori produk heterogen, hingga pemantauan sisa stok secara *real-time*.

Aplikasi yang dikembangkan tidak hanya berfungsi untuk mencatat mutasi stok, tetapi juga untuk melakukan pengolahan data persediaan guna mengantisipasi ketidakpastian permintaan. Pengolahan data tersebut digunakan untuk menentukan jumlah stok pengaman (*safety stock*) yang harus tersedia di gudang. Hal ini bertujuan sebagai dasar pertimbangan bagi pihak manajemen dalam menjaga stabilitas ketersediaan bahan baku agar proses produksi tetap berjalan lancar tanpa terhambat oleh risiko kekosongan barang (*stockout*).

Berdasarkan domain sistem tersebut, konsep dan karakteristik aplikasi yang dikembangkan mengacu pada beberapa teori, yaitu:

A. *Supply Chain Management* (SCM)

Supply Chain Management (SCM) dapat didefinisikan sebagai sebuah jaringan kolaboratif antar organisasi yang bekerja secara sinergis untuk menciptakan sekaligus mendistribusikan produk hingga ke tangan konsumen akhir. SCM juga berperan sebagai instrumen integrasi bagi berbagai proses bisnis utama, mulai dari tingkat pemasok hingga pengguna akhir, guna memberikan nilai

tambah berupa produk, layanan, serta informasi bagi pelanggan maupun para pemangku kepentingan (Monalisa & Apsyarin, 2021).

Dalam operasionalnya, SCM merupakan mekanisme strategis yang bertujuan untuk mendongkrak produktivitas organisasi melalui optimasi waktu, ketepatan lokasi, serta pengaturan alur kuantitas material. Khusus pada sektor industri dan manufaktur, implementasi SCM menuntut kemampuan perusahaan untuk menjamin kepuasan pelanggan melalui pengembangan produk yang tepat waktu, serta kemampuan menekan biaya operasional pada aspek persediaan dan distribusi dengan manajemen yang fleksibel namun tetap terukur.

1) Konsep *Supply Chain Management* (SCM)

Implementasi *Supply Chain Management* didasarkan pada beberapa konsep utama yang bertujuan untuk menjamin efektivitas operasional perusahaan, yaitu:

- a. Integrasi dan Sinkronisasi Alur Kerja: Mengintegrasikan peran pemasok, produsen, hingga distributor guna menjamin produk yang dihasilkan telah sesuai dengan standar kualitas, kuantitas, serta waktu distribusi yang direncanakan. Hal ini bertujuan untuk mengefisienkan biaya operasional sekaligus meningkatkan kualitas pelayanan kepada pelanggan.
- b. Rasionalisasi Jumlah Pemasok: Menyederhanakan jaringan pemasok untuk meminimalkan variabilitas produk, menekan biaya administrasi, serta mempercepat proses negosiasi dan pemantauan. Strategi ini bergeser dari penggunaan banyak pemasok (*multiple suppliers*) menuju kolaborasi dengan pemasok tunggal (*single supplier*) yang memiliki kualifikasi terbaik.

- c. Pengembangan Kemitraan Strategis: Membangun hubungan kerja sama jangka panjang dengan pemasok (*supplier partnership*) untuk menjamin stabilitas pasokan barang. Fokus utamanya adalah melakukan perbaikan berkelanjutan pada aspek efisiensi biaya dan standar mutu material.
- d. Optimalisasi Pengadaan Langsung: Menerapkan mekanisme pengadaan barang langsung dari produsen tanpa melalui pihak ketiga (perantara). Dalam konsep ini, pemasok idealnya bertindak sebagai produsen utama guna menghindari beban biaya tambahan dan menjaga keaslian serta kualitas produk.

2) Aktivitas *Supply Chain Management* (SCM)

Aktivitas SCM dibagi menjadi tiga rantai utama :

- a. Rantai Suplai Hulu (*Upstream Supply Chain*):
Fase ini mencakup seluruh interaksi antara perusahaan dengan pemasok eksternal. Fokus utamanya adalah pada aktivitas pengadaan (*procurement*) serta memastikan kelancaran alokasi bahan baku atau material yang dibutuhkan untuk proses produksi.
- b. Rantai Suplai Internal (*Internal Supply Chain*):
Fase ini berfokus pada proses transformasi di dalam organisasi. Aktivitas intinya meliputi manajemen produksi, teknis pabrikasi, serta pengendalian inventori secara menyeluruh di dalam gudang untuk menjaga keseimbangan stok.
- c. Rantai Suplai Hilir (*Downstream Supply Chain*):
Fase ini melibatkan proses distribusi produk jadi agar sampai ke tangan konsumen akhir. Aktivitas ini mencakup manajemen pergudangan, pengaturan

transportasi, hingga teknis penyerahan barang yang efektif kepada pelanggan.

3) Prinsip *Supply Chain Management* (SCM)

Prinsip dasar yang harus dipenuhi dalam implementasi SCM meliputi:

- a. Integrasi: Seluruh unit yang terlibat membentuk suatu kesatuan sistem yang solid dan memiliki kesadaran kolektif terhadap adanya keterkaitan fungsional antar elemen.
- b. Jejaring: Terwujudnya pola hubungan serta koordinasi kerja yang harmonis dan selaras di antara semua pihak dalam rantai pasok.
- c. Ujung ke Ujung (*End-to-End*): Cakupan operasional yang komprehensif, dimulai dari pengelolaan pemasok bahan baku paling awal hingga produk diterima oleh konsumen akhir.
- d. Saling Tergantung: Adanya bentuk sinergi yang saling memberikan profitabilitas guna memperkuat posisi dan daya saing seluruh pihak secara kolektif.
- e. Komunikasi: Proses transmisi data yang presisi dan transparan untuk memastikan kelancaran arus distribusi barang tanpa hambatan informasi.
- f. Kemitraan: Komitmen untuk saling bertukar informasi dan membangun kepercayaan bersama dengan memprioritaskan standar mutu serta ketepatan waktu.
- g. Dukungan: Adanya komitmen penuh dari pihak manajemen serta keterlibatan aktif fungsi operasional dalam setiap tahapan koordinasi maupun pengendalian sistem.

4) **Manfaat *Supply Chain Management* (SCM)**

Penerapan SCM secara efektif memberikan keuntungan strategis bagi perusahaan, khususnya dalam pengelolaan operasional, yaitu:

- a. **Optimalisasi Inventori:** Mampu meminimalkan penumpukan aset dengan menekan tingkat persediaan melalui sistem pengendalian yang ketat dan pertukaran informasi yang intensif. Hal ini bertujuan untuk mencapai keseimbangan stok yang ideal, sehingga perusahaan terhindar dari risiko kelebihan barang (*overstock*) maupun kekurangan barang (*stockout*).
- b. **Reduksi *Lead Time*:** Mempersingkat durasi waktu tunggu melalui koordinasi data yang presisi di setiap lini. Dengan alur informasi yang tepat, proses mulai dari pengadaan bahan baku, tahap produksi, hingga distribusi ke pelanggan dapat dilakukan secara lebih cepat dan efisien.

5) **Fungsi *Supply Chain Management* (SCM)**

Secara fundamental, SCM menjalankan dua fungsi utama yang saling berkaitan dalam menjamin kelangsungan bisnis perusahaan :

- a. **Fungsi Fisik:** Berperan dalam proses transformasi teknis, yaitu mengolah bahan mentah menjadi produk jadi dan mendistribusikannya hingga ke tangan pengguna akhir. Fungsi ini berfokus pada efisiensi biaya-biaya riil yang timbul selama proses berlangsung, seperti biaya pengadaan material, ongkos produksi, biaya penyimpanan di gudang, hingga biaya transportasi.

- b. Fungsi Mediasi Pasar: Berperan dalam menyelaraskan antara ketersediaan produk dengan kebutuhan serta ekspektasi pelanggan. Fungsi ini sangat vital untuk menjaga stabilitas bisnis dengan meminimalkan risiko kerugian yang disebabkan oleh barang yang tidak terjual habis (*markdown*) maupun potensi kehilangan penjualan akibat ketiadaan stok di gudang (*stockout cost*).

6) Tujuan *Supply Chain Management* (SCM)

Tujuan utama SCM adalah menekan total biaya operasional secara signifikan tanpa mengurangi standar pelayanan yang diberikan kepada pelanggan. Dalam perkembangannya, pencapaian tujuan ini sangat bergantung pada pemanfaatan teknologi informasi untuk menyatukan berbagai proses bisnis yang terpisah. Melalui sistem yang terintegrasi, pertukaran data antar pihak yang terlibat dapat berjalan secara *real-time*, sehingga tercipta koordinasi yang lebih akurat dan efisien dalam rantai pasok.

Dari pengertian di atas, dapat disimpulkan bahwa *Supply Chain Management* (SCM) merupakan mekanisme strategis yang mengintegrasikan seluruh proses bisnis mulai dari pengadaan bahan baku hingga distribusi produk akhir ke dalam satu kesatuan sistem yang kolaboratif. Esensi dari SCM adalah terciptanya sinergi antara fungsi fisik (transformasi material) dan fungsi mediasi pasar (penyesuaian terhadap permintaan) guna mencapai efisiensi operasional dan optimalisasi inventori. Dengan memanfaatkan integrasi data yang presisi, SCM mampu memitigasi risiko ketidakpastian melalui reduksi *lead time* dan menjaga keseimbangan stok yang ideal. Oleh karena itu, SCM menjadi kerangka kerja fundamental dalam penelitian ini untuk membangun sistem yang mampu

menyeimbangkan ketersediaan material dengan kebutuhan produksi secara efisien dan berkelanjutan.

B. Pengelolaan Produk Heterogen (Buket dan Kue)

Menurut Kamus Besar Bahasa Indonesia (KBBI), pengelolaan adalah cara atau proses mengelola atau mengendalikan. Dalam konteks UMKM, pengelolaan produk melibatkan perencanaan stok agar ketersediaan barang tetap terjaga (Iskandar et al., 2025). UMKM Nawang Moments mengelola produk yang bersifat heterogen, yaitu penggabungan antara produk kreatif (buket) dan produk kuliner (kue). Persediaan pada UMKM yang memiliki variasi produk berbeda memerlukan strategi kategorisasi bahan baku berdasarkan karakteristik dan masa simpannya (Fa'iza & Ramadhani, 2025). Pada aplikasi yang dikembangkan, manajemen produk heterogen ini ditransformasikan ke dalam bentuk entitas pengelompokan (kategori) pada basis data.

Dalam operasionalnya, perbedaan karakteristik ini menuntut perlakuan rantai pasok yang spesifik menurut Iskandar et al. (2025), yaitu:

- 1) Produk *Perishable* (Mudah Rusak): Bahan baku kue memiliki masa kedaluwarsa singkat, sehingga memerlukan pemantauan stok yang sangat ketat dan frekuensi pemesanan yang lebih sering guna menghindari kerugian akibat bahan baku yang busuk atau kualitas yang menurun. Karakteristik ini mendasari perlunya visualisasi tingkat urgensi ketersediaan stok pada antarmuka sistem.
- 2) Produk *Non-Perishable* (Tahan Lama): Bahan baku bouquet (kertas *wrap*, pita, bunga artifisial) memiliki masa simpan lebih lama, namun memerlukan manajemen variasi yang baik agar tidak terjadi penumpukan stok mati (*dead stock*) yang dapat menghambat perputaran modal usaha.

Dari pengertian di atas, dapat disimpulkan bahwa pengelolaan produk heterogen merupakan strategi pengendalian persediaan yang adaptif, dengan penekanan pada kategorisasi bahan baku berdasarkan karakteristik masa simpan. Pemisahan entitas antara produk *perishable* (mudah rusak) dan *non-perishable* (tahan lama) merupakan instrumen krusial dalam manajemen inventori untuk menentukan perlakuan rantai pasok yang spesifik. Pendekatan ini memungkinkan pengelolaan risiko secara presisi untuk memitigasi kerugian akibat penurunan kualitas produk *perishable*, sekaligus mengoptimalkan perputaran modal dengan mencegah akumulasi stok mati pada produk *non-perishable*. Dengan demikian, kategorisasi yang tepat merupakan prasyarat fundamental dalam menjaga stabilitas ketersediaan barang dan efisiensi operasional suatu organisasi.

C. Metode *Safety Stock*

Safety Stock merupakan metode persediaan pengaman yang digunakan untuk mengantisipasi adanya ketidakpastian permintaan pelanggan serta fluktuasi dalam pemakaian bahan baku harian (Iskandar et al., 2025). Pengendalian persediaan yang tepat melalui metode ini sangat krusial untuk menghindari risiko kekurangan stok (*stockout*) maupun kelebihan stok (*overstock*) yang dapat mengganggu efisiensi operasional perusahaan. Penerapan metode ini menjadi instrumen kritis, khususnya bagi unit usaha yang mengelola bahan baku bersifat *perishable* (mudah rusak), guna meminimalisir risiko penurunan kualitas barang dan kerugian finansial akibat kedaluwarsa.

Secara literatur manajemen operasi, kuantitas pengaman dihitung berdasarkan pencatatan deviasi waktu tunggu dan volume penggunaan. Persamaan matematis standar yang digunakan untuk menentukan nilai *Safety Stock* adalah sebagai berikut:

$$SS = (D_{\max} \times L_{\max}) - (D_{\text{avg}} \times L_{\text{avg}})$$

Keterangan :

- 1) SS (*Safety Stock*) = Jumlah stok pengaman unit persediaan.
- 2) Dmax (*Maximal Demand*) = Kuantitas pemakaian harian maksimum bahan baku dalam satu periode observasi.
- 3) Lmax (*Maximal Lead Time*) = Durasi waktu tunggu (*lead time*) terlama dalam siklus pemesanan bahan baku dari pemasok.
- 4) Davg (*Average Demand*) = Rata-rata kuantitas pemakaian bahan baku per hari dalam rentang periode waktu tertentu.
- 5) Lavg (*Average Lead Time*) = Rata-rata durasi waktu tunggu penerimaan bahan baku dalam kondisi operasional normal.

Metode ini dipilih karena efektivitasnya dalam memitigasi risiko operasional pada manajemen rantai pasok yang memiliki pola pemakaian bahan baku cenderung stabil, namun memiliki ketergantungan tinggi pada aspek ketepatan waktu pengiriman oleh pihak pemasok (*supplier*).

D. Website

Website adalah sebuah kumpulan halaman yang berisi informasi dalam bentuk teks, gambar, atau multimedia yang dapat diakses melalui peramban web (*web browser*) dengan menggunakan koneksi internet. Menurut Monalisa & Apsyarin (2021), sistem informasi berbasis web pada manajemen rantai pasok dirancang untuk menciptakan alur kerja yang lebih terukur, di mana data dapat diakses secara transparan oleh pihak-pihak yang berkepentingan.

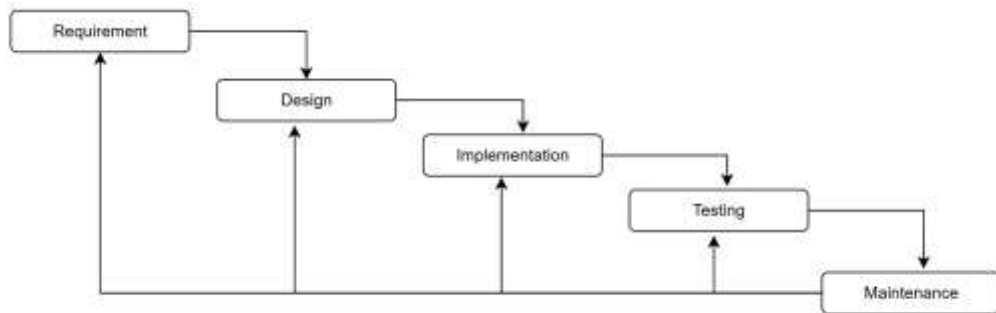
Penggunaan teknologi website dalam pengembangan sistem SCM di Nawang Moments dipilih karena sifatnya yang lintas platform. Sejalan dengan penelitian Dzaki et al. (2025), karakteristik sistem berbasis *website* memungkinkan pengguna untuk melakukan pemantauan data secara *real-time* tanpa harus terikat pada satu perangkat keras tertentu, sehingga memberikan fleksibilitas tinggi bagi pihak manajemen dalam memantau pergerakan operasional secara efisien.

Dari pengertian di atas, dapat disimpulkan bahwa *website* merupakan media penyampai informasi berbasis *online* yang memiliki fleksibilitas tinggi dalam integrasi sistem informasi manajemen. Sebagai sebuah infrastruktur berbasis *web*, teknologi ini mampu mentransformasikan proses operasional menjadi alur kerja yang transparan, terukur, dan dapat diakses secara *real-time* lintas platform. Oleh karena itu, penggunaan *website* dalam pengembangan sistem manajemen rantai pasok merupakan solusi yang tepat untuk meningkatkan efisiensi pemantauan data serta mempermudah pengambilan keputusan manajerial yang cepat dan akurat, tanpa dibatasi oleh ketergantungan pada perangkat keras tertentu.

2.1.3 Metode Pengembangan Sistem

Software Development Life Cycle (SDLC) dengan model *waterfall* atau sering disebut sebagai *classic life cycle* merupakan model pengembangan perangkat lunak yang dilakukan secara sistematis dan berurutan (*sequential*). Menurut Pressman sebagaimana dikutip dalam (Sofyan et al., 2024), model ini mempresentasikan fase-fase proses yang berbeda seperti spesifikasi persyaratan, perancangan, implementasi, hingga pengujian. Karakteristik utama dari metode ini adalah alur kerjanya yang linear, di mana setiap tahapan harus diselesaikan secara menyeluruh sebelum melangkah ke tahap berikutnya.

Penggunaan metode ini memberikan keuntungan dalam hal manajemen proyek karena setiap fase memiliki target serta kontrol kualitas yang jelas dan terdokumentasi dengan baik. Hal ini bertujuan agar setiap proses pengembangan sistem informasi SCM pada UMKM Nawang Moments berjalan secara terorganisir sesuai dengan metodologi yang telah ditetapkan. Melalui pendekatan ini, risiko terjadinya tumpang tindih (*overlapping*) antar tahapan dapat diperkecil sehingga menghasilkan sistem yang lebih stabil, konsisten, dan terstruktur.



Gambar 2. 1 Tahapan metode waterfall

Sumber : Pressman (2010), seperti dikutip dalam Sofyan et al. (2024)

A. Tahapan *Waterfall*

Menurut Pressman (Sofyan et al. 2024), model *waterfall* (sering disebut sebagai *classic life cycle*) merupakan pendekatan sistematis dan sekuensial dalam pengembangan perangkat lunak. Tahapan-tahapan dalam model ini adalah sebagai berikut:

1) *Requirement*

Tahap ini merupakan fase pengumpulan kebutuhan secara menyeluruh untuk memahami tujuan, spesifikasi, dan batasan perangkat lunak yang akan dibangun. Proses ini dilakukan melalui teknik pengumpulan data guna mendapatkan spesifikasi kebutuhan fungsional (fitur sistem) dan non-fungsional (performa dan keamanan) yang menjadi dasar utama seluruh proses pengembangan.

2) *Design*

Pada tahap ini, spesifikasi kebutuhan dari tahap sebelumnya diterjemahkan ke dalam bentuk cetak biru teknis (*blueprint*). Fokus utama tahap ini meliputi perancangan arsitektur perangkat lunak, desain antarmuka (*user interface*), struktur data, dan representasi algoritma. Alat bantu yang digunakan dalam pemodelan sistem umumnya berupa diagram *Unified Modeling Language* (UML) seperti *Use Case Diagram*, *Activity Diagram*, *Class Diagram*, dan *Sequence Diagram*. Selain itu, digunakan

pula *Entity Relationship Diagram* (ERD) untuk merancang struktur basis data (*database*) secara konseptual, memetakan hubungan antar entitas, serta menentukan *primary key* dan *foreign key* guna menjamin integritas data.

3) *Implementation*

Tahap ini merupakan proses realisasi dari hasil rancangan teknis ke dalam bentuk kode program (*coding*). Pengembang memanfaatkan bahasa pemrograman, *framework* pilihan, dan sistem pengelola basis data (*Database Management System*) untuk membangun komponen-komponen serta modul aplikasi sesuai dengan spesifikasi cetak biru yang telah disepakati.

4) *Testing*

Tahap ini dilakukan untuk memastikan bahwa perangkat lunak yang dibangun telah memenuhi spesifikasi kebutuhan dan berfungsi dengan benar. Fokus utama pada tahap ini adalah untuk menemukan kesalahan (*bug*) atau kegagalan logika melalui berbagai metode pengujian, seperti *Black-box* dan *White-box testing*, sebelum sistem diimplementasikan secara nyata.

5) *Maintenance*

Tahap ini dilakukan setelah perangkat lunak dioperasikan oleh pengguna. Fokus utamanya adalah menjaga agar sistem tetap berfungsi secara optimal melalui perbaikan kesalahan yang baru ditemukan, penyesuaian sistem terhadap perubahan lingkungan teknologi, serta peningkatan fungsionalitas berdasarkan kebutuhan operasional yang berkembang di masa depan.

B. Kelebihan Metode *Waterfall*

Menurut Wahyu et al. (2024), penggunaan model *Waterfall* dalam pengembangan sistem memiliki karakteristik yang mencakup sejumlah keunggulan serta keterbatasan tertentu, yaitu:

- 1) *Kualitas Sistem yang Optimal*: Prosedur pengembangan yang dilakukan secara bertahap dan terencana menghasilkan perangkat lunak dengan standar kualitas yang baik.
- 2) *Minimalisir Kesalahan Teknis*: Proses pengerjaan yang dilakukan secara berurutan pada setiap fase memungkinkan pengembang untuk lebih fokus, sehingga potensi kekeliruan dapat diminimalisir.
- 3) *Dokumentasi yang Terstruktur*: Setiap fase pengembangan menghasilkan dokumentasi yang sangat terorganisir, karena penyelesaian satu tahap merupakan syarat mutlak sebelum melanjutkan ke tahap berikutnya.

C. Kekurangan Metode *Waterfall*

Menurut Wahyu et al. (2024), selain kelebihan yang dimiliki, metode ini juga mempunyai beberapa keterbatasan, yaitu:

- 1) *Alokasi Waktu dan Biaya*: Implementasi metode ini cenderung membutuhkan durasi pengerjaan yang cukup panjang serta estimasi biaya yang relatif tinggi.
- 2) *Kekakuan dalam Manajemen*: Alur kerja bersifat linear dan kaku, sehingga menyulitkan pengembang jika ingin melakukan perubahan atau pengulangan sebelum produk akhir selesai sepenuhnya.
- 3) *Risiko Akumulasi Masalah*: Kekeliruan kecil pada fase awal yang tidak terdeteksi dapat berkembang menjadi kendala besar yang berdampak pada keberhasilan tahapan selanjutnya.
- 4) *Ketidaksesuaian Praktik Teoritis*: Dalam penerapannya di lapangan, proses pengembangan sering kali sulit untuk benar-benar mematuhi urutan *waterfall* yang kaku sebagaimana digambarkan dalam teori.

Dari pengertian di atas, dapat disimpulkan bahwa metode *Waterfall* merupakan pendekatan pengembangan perangkat lunak yang sistematis

dengan alur kerja linear dan sekuensial. Melalui tahapan yang terstruktur mulai dari analisis kebutuhan hingga pemeliharaan metode ini memberikan kontrol kualitas yang ketat dan dokumentasi yang terdokumentasi dengan baik di setiap fasenya. Meskipun memiliki keterbatasan dalam hal fleksibilitas perubahan di tengah jalan, keunggulan metode ini dalam meminimalisir kesalahan teknis menjadikannya pilihan yang tepat bagi pengembangan sistem informasi yang membutuhkan stabilitas dan konsistensi tinggi. Oleh karena itu, penerapan metode *Waterfall* menjadi landasan metodologis yang ideal untuk memastikan bahwa sistem yang dibangun dapat memenuhi spesifikasi kebutuhan secara akurat, terukur, dan terorganisir.

2.1.4 Pemodelan dan Perancangan Sistem

A. *Unified Modeling Language* (UML)

Unified Modeling Language (UML) merupakan bahasa standar yang digunakan untuk memvisualisasikan, menspesifikasikan, membangun, dan mendokumentasikan artefak dari sistem perangkat lunak. Menurut Dzaki et al. (2025), pemodelan sistem menggunakan UML sangat krusial dalam memetakan aliran manajemen rantai pasok (SCM) agar proses bisnis yang kompleks—terutama pada pengelolaan produk heterogen—dapat diilustrasikan secara transparan dan terukur.

UML membantu pengembang sistem dalam memahami hubungan antara aktor (pengguna) dengan fungsionalitas sistem, serta bagaimana data diolah di dalam *database*. Dalam pengembangan sistem SCM Nawang Moments, penggunaan UML berfungsi untuk meminimalkan kesalahan logika pada tahap desain sebelum masuk ke tahap pengkodean. Hal ini penting untuk memastikan fitur seperti perhitungan *safety stock* berjalan akurat sesuai kebutuhan operasional (Alfin, 2025).

1) *Use Case Diagram*

Use case diagram adalah model grafis yang menggambarkan sekumpulan *use case* dan aktor serta hubungan di antara mereka. Diagram ini memberikan gambaran fungsionalitas sistem dari sudut pandang pengguna. Menurut Sukri & Pathiassana (2022), identifikasi aktor sangat penting dalam SCM untuk membedakan hak akses antara pemilik usaha dan bagian produksi dalam mengelola aliran barang dan informasi.

Tabel 2. 1 Simbol Use Case Diagram

Sumber : Rosa dan Shalahuddin (2019), seperti dikutip dalam Nurlita & Anggraini (2023)

Simbol	Nama	Keterangan
	Aktor	Seseorang atau siapa saja yang berhubungan dengan sistem yang sedang dibangun.
	Use Case	Bagian fungsionalitas tingkat tinggi yang disediakan oleh sistem. Use case digambarkan sebagai suatu kegiatan yang dilakukan aktor
	Association	Komunikasi antara aktor dan use case yang berpartisipasi dengan aktor

<code><<extend>></code> 	Ekstensi/extend	Menunjukkan bahwa suatu use case merupakan tambahan fungsional dari use case lainnya jika suatu kondisi terpenuhi
<code><<include>></code> 	Include	Menunjukkan bahwa suatu use case seluruhnya merupakan fungsionalitas dari use case lainnya

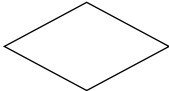
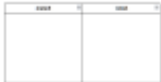
2) *Activity Diagram*

Activity diagram digunakan untuk memodelkan proses bisnis dan aliran kerja (*workflow*) secara berurutan di dalam sistem. Diagram ini sangat membantu dalam memvisualisasikan alur pengelolaan stok produk heterogen, mulai dari input bahan baku masuk hingga laporan stok sisa. Hal ini selaras dengan kebutuhan pemantauan bahan *perishable* yang harus memiliki alur penanganan cepat guna menghindari kerusakan bahan.

Tabel 2. 2 Simbol Activity Diagram

Sumber : Rosa dan Shalahuddin (2019), seperti dikutip dalam Nurlita & Anggraini (2023)

Simbol	Nama	Keterangan
	Status Awal	Status awal aktivitas sistem, sebuah

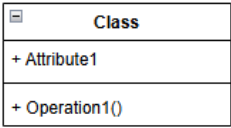
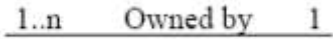
		diagram aktivitas memiliki sebuah status awal
	Aktivitas	Aktivitas adalah suatu kegiatan yang dilakukan didalam sistem, biasanya diawali dengan kata kerja.
	Percabangan / Decision	Percabangan adalah suatu kegiatan dimana terdapat pilihan kegiatan didalamnya.
	Penggabungan / Join	Penggabungan dimana yang mana lebih dari satu aktivitas lalu digabungkan jadi satu
	Status Akhir	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir
	Swimlane	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas terjadi.

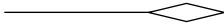
3) *Class Diagram*

Class diagram memberikan gambaran statis mengenai struktur sistem dengan menunjukkan kelas-kelas yang ada, atributnya, metodenya, serta hubungan antar kelas. *Class diagram* menjadi fondasi dalam pembuatan database karena mendefinisikan entitas seperti data produk (kue/buket), data pemasok (*supplier*), dan data persediaan (*inventory*).

Tabel 2. 3 Simbol Class Diagram

Sumber : Yusman (2016)

Simbol	Nama	Keterangan
	Kelas / <i>Class</i>	Kelas merupakan gambaran dari struktur sistem. Atribut adalah penggambaran tentang keadaan dari suatu objek. Operasi adalah penggambaran tentang fungsi.
	Association	Asosiasi merupakan sebuah hubungan antara 2 class dan dilambangkan oleh sebuah garis. Garis ini merupakan tipe-tipe hubungan. One-to-one, one-to-many, many-to-many.

0..1, 1, 0..*, 1..*, n, 0..n, 1..n	Multiplicity	Menunjukkan jumlah minimum dan maksimum objek yang dapat terlibat dalam suatu relasi antar kelas.
	Composition	Jika sebuah class tidak bisa berdiri sendiri dan merupakan bagian dari class lain, maka class tersebut memiliki relasi komposisi terhadap class tempat dia bergantung.
	Generalisasi	Generalisasi adalah hubungan pewarisan (inheritance) antara kelas induk (superclass) dan kelas turunan (subclass), di mana subclass mewarisi atribut dan metode dari superclass.
	Aggregation	Hubungan keseluruhan dan bagian (whole-part relationship), di mana suatu kelas terdiri dari kelas lain, tetapi bagian tersebut masih dapat berdiri sendiri tanpa keseluruhan.
	Directed association	Relasi antara kelas dengan makna kelas

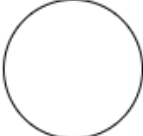
		yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertakan dengan <i>multiplicity</i>
--	--	--

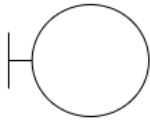
4) *Sequence Diagram*

Sequence diagram adalah diagram yang menggambarkan kolaborasi objek berdasarkan urutan waktu. Diagram ini menekankan pada pengiriman pesan (*message*) dalam rentang waktu tertentu untuk menjalankan suatu fungsi sistem. Menurut Dzaki et al. (2025), penggunaan *sequence diagram* pada sistem SCM memastikan bahwa setiap transaksi data, mulai dari pengadaan hingga stok keluar, tercatat secara kronologis di dalam basis data.

Tabel 2. 4 Simbol *Sequence Diagram*

Sumber : Nurlita & Anggraini (2023)

Gambar	Nama	Keterangan
	Aktor	Menggambarkan orang yang sedang berinteraksi dengan sistem.
	Entity Class	Menggambarkan hubungan kegiatan yang dilakukan

	Boundary Class	Menggambarkan antarmuka sebuah penggambaran dari halaman atau form sistem.
	Control Class	Menggambarkan penghubung antara boundary dengan tabel.
	Message	Menggambarkan spesifikasi dari komunikasi antara objek yang memuat informasi-informasi tentang aktifitas yang terjadi
	Activation	Mewakili proses durasi aktivasi sebuah operasi

Dari pengertian di atas, dapat disimpulkan bahwa *Unified Modeling Language* (UML) merupakan bahasa pemodelan visual yang krusial dalam proses perancangan sistem untuk menerjemahkan kebutuhan bisnis yang kompleks menjadi cetak biru teknis yang terstruktur. Melalui berbagai jenis diagram seperti *Use Case*, *Activity*, *Class*, dan *Sequence Diagram*, UML berfungsi sebagai instrumen untuk meminimalkan ambiguitas, mencegah kesalahan logika, serta menjamin integritas data dalam sistem. Dengan memetakan alur kerja dan struktur basis data secara sistematis, UML berperan sebagai fondasi utama dalam pengembangan perangkat lunak yang presisi, sehingga fungsionalitas system termasuk perhitungan logis seperti *Safety*

Stock dapat dirancang dengan akurat sebelum memasuki tahapan implementasi.

2.1.5 Konsep Basis Data

Basis data (*database*) adalah kumpulan data yang terorganisir secara sistematis sehingga dapat diakses, dikelola, dan diperbarui dengan mudah. Dalam sistem SCM UMKM Nawang Moments, basis data berfungsi sebagai pusat penyimpanan seluruh informasi operasional. Teori dasar yang digunakan meliputi:

A. Tabel :

Dalam sistem basis data, tabel menjadi komponen utama yang digunakan untuk menyimpan kumpulan data, di mana setiap tabel dibentuk oleh kolom (*field*) untuk klasifikasi jenis data dan baris (*record*) sebagai wadah dari data aktualnya (Mardiono et al., 2019). Dalam sistem ini, setiap entitas seperti produk, pemasok, dan transaksi direpresentasikan dalam bentuk tabel.

B. Relasi :

Menurut Putri & Khairunnisa (2025), relasi dalam *Entity Relationship Diagram* (ERD) merupakan representasi hubungan antar entitas yang secara visual disimbolkan dengan bentuk belah ketupat, di mana hubungan tersebut diklasifikasikan ke dalam beberapa derajat kardinalitas. Klasifikasi tersebut meliputi *One to One* yang menghubungkan satu anggota entitas secara eksklusif dengan satu anggota entitas lain, *One to Many* yang memungkinkan satu anggota entitas berelasi dengan berbagai anggota pada entitas lainnya, serta *Many to Many* yang menggambarkan keterhubungan antara beberapa anggota entitas dengan banyak anggota pada entitas yang berbeda.

C. Primary Key

Primary Key adalah atribut atau sekumpulan atribut dalam suatu tabel yang digunakan untuk mengidentifikasi setiap record secara unik. Nilai dari Primary Key tidak boleh sama antar baris dan tidak

diperkenankan bernilai *null*, sehingga setiap data yang tersimpan memiliki identitas yang jelas dan tidak terjadi duplikasi.

Primary Key berperan penting dalam menjaga integritas data karena menjadi penanda utama suatu entitas dalam basis data relasional. Selain itu, Primary Key juga menjadi dasar dalam pembentukan relasi antar tabel, karena atribut ini akan direferensikan oleh tabel lain melalui penggunaan *foreign key*. Dalam sistem Nawang Moments, contoh Primary Key adalah *id_produk*, *id_supplier*, dan *id_transaksi*, yang masing-masing digunakan untuk membedakan setiap data produk, pemasok, dan transaksi secara unik.

D. Foreign Key :

Foreign Key adalah atribut dalam suatu tabel yang berfungsi sebagai penghubung dengan Primary Key pada tabel lain. Atribut ini digunakan untuk membentuk relasi antar tabel dalam basis data relasional serta memastikan bahwa data yang saling berhubungan tetap konsisten. Dengan adanya Foreign Key, sistem dapat mengetahui keterkaitan antar data yang tersimpan di tabel yang berbeda.

Foreign Key juga berperan dalam menjaga integritas referensial (*referential integrity*), yaitu memastikan bahwa setiap nilai yang dimasukkan memiliki referensi yang valid pada tabel induk. Sebagai contoh dalam sistem Nawang Moments, atribut *id_transaksi* pada tabel *detail_transaksi* berfungsi sebagai Foreign Key yang merujuk pada tabel *transaksi*. Hal ini memastikan bahwa setiap detail transaksi selalu terhubung dengan transaksi yang sah dan terdaftar dalam sistem.

E. Normalisasi :

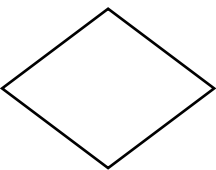
Normalisasi adalah prosedur mendekomposisikan tabel yang memiliki anomali agar menjadi struktur yang lebih sederhana dan terorganisir untuk menghindari duplikasi data dalam basis data. Implementasi teknik ini bertujuan untuk memastikan bahwa setiap manipulasi baris data, baik berupa penambahan, penghapusan, maupun pembaruan, dapat dilakukan oleh pengguna tanpa menyebabkan

terjadinya ketidakselarasan data (Efendy, 2018). Normalisasi diterapkan pada basis data Nawang Moments untuk memastikan data stok bahan baku dan data pesanan tidak mengalami duplikasi yang dapat menyebabkan kesalahan perhitungan *safety stock*.

F. Manajemen Basis Data (*Database Management System, DBMS*) :

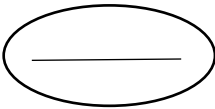
Menurut Hesananda et al. (2017), *Database Management System* (DBMS) merupakan serangkaian perangkat lunak yang berfungsi untuk mendefinisikan, mengelola, serta mengolah basis data, sementara basis data itu sendiri berperan sebagai instrumen dalam menyusun struktur data tersebut. Dalam penelitian ini, DBMS yang digunakan adalah MySQL, karena bersifat *open-source*, mendukung sistem basis data relasional, serta mampu mengelola data dalam jumlah besar secara efisien.

Tabel 2. 5 Simbol Entity Relationship Diagram

Simbol	Nama	Fungsi
	<i>Entity/Entitas</i>	Entitas adalah Objek data prinsip tentang informasi yang dikumpulkan.
	<i>Relationship/Relasi</i>	Relasi merupakan hubungan atau keterkaitan antara dua tabel atau lebih dalam suatu basis data. Relasi menunjukkan bagaimana data pada satu entitas berhubungan dengan data pada entitas lainnya. Jenis relasi

		dapat dijelaskan sebagai berikut: 1) Relasi satu ke satu (one to one relationship) : Relasi satu ke satu adalah hubungan di mana satu entitas hanya memiliki satu hubungan dengan satu entitas lainnya. Relasi ini dinyatakan dengan notasi 1:1. 2) Relasi satu ke banyak (one to many relationship) : Relasi satu ke banyak merupakan hubungan di mana satu entitas dapat memiliki banyak hubungan
--	--	--

		dengan entitas lainnya. Relasi ini dinyatakan dengan notasi 1:M. 3) Relasi banyak ke satu (many to one relationship) : Relasi banyak ke satu adalah hubungan di mana banyak entitas dapat memiliki satu hubungan dengan satu entitas lainnya. Relasi ini dinyatakan dengan notasi M:1. 4) Relasi banyak ke banyak (many to many relationship) : Relasi banyak ke banyak merupakan hubungan di mana setiap
--	--	--

		entitas dapat memiliki banyak hubungan dengan entitas lainnya, dan sebaliknya. Relasi ini dinyatakan dengan notasi M:M.
	Atribut	Atribut merupakan ciri atau karakteristik yang dimiliki oleh suatu entitas dalam data, yang berfungsi untuk menggambarkan sifat atau karakter dari entitas tersebut.
	Primary Key	Merupakan atribut yang digunakan untuk membedakan antara satu data dengan data lain secara unik dalam sebuah entitas.
<<FK>>	Foreign Key	Atribut dalam suatu tabel yang merujuk pada Primary Key di tabel lain untuk

		membentuk relasi antar tabel.
	Garis	Garis digunakan untuk menghubungkan entitas dengan relasi atau menghubungkan entitas dengan atribut

Dari pengertian di atas, dapat disimpulkan bahwa basis data merupakan fondasi penyimpanan informasi yang terorganisir secara sistematis untuk mendukung efektivitas pengelolaan data operasional. Melalui penerapan teknik normalisasi, penentuan *Primary Key* dan *Foreign Key* yang tepat, serta pemanfaatan DBMS yang handal, basis data mampu menjamin integritas, konsistensi, dan efisiensi akses data dalam skala yang besar. Dalam konteks sistem manajemen rantai pasok, arsitektur basis data yang relasional menjadi instrumen krusial untuk memetakan hubungan antar entitas secara akurat. Oleh karena itu, perancangan struktur basis data yang solid merupakan prasyarat mutlak untuk menghasilkan informasi yang berkualitas, yang menjadi basis data pendukung dalam proses pengambilan keputusan manajerial terkait ketersediaan stok dan kelancaran operasional usaha.

2.1.6 Konsep Pengujian Sistem

Pengujian sistem merupakan tahapan dalam pengembangan perangkat lunak yang dilakukan untuk memastikan sistem telah berjalan sesuai dengan kebutuhan fungsional yang ditetapkan. Pengujian ini bertujuan untuk menilai kualitas sistem, memastikan fungsi berjalan dengan benar, serta meminimalkan kesalahan sebelum sistem digunakan secara nyata.

Dalam penelitian ini, pengujian sistem dilakukan menggunakan dua pendekatan, yaitu pengujian Black Box dan pengujian White Box.

A. *White Box Testing*

White Box Testing merupakan pendekatan pengujian perangkat lunak yang menitikberatkan pada analisis struktur internal serta logika prosedural aplikasi dengan memanfaatkan deskripsi kontrol pada perancangan tingkat komponen sebagai dasar penyusunan skenario uji (*test cases*). Dalam praktiknya, metode ini menerapkan berbagai teknik evaluasi seperti *Data Flow Testing*, *Control Flow Testing*, *Basis Path/Path Testing*, hingga *Loop Testing* yang mengharuskan penguji memiliki pemahaman mendalam terhadap kode sumber (*source code*) guna mendeteksi celah kesalahan implementasi secara akurat. Melalui pemeriksaan logika yang komprehensif ini, setiap jalur fungsional di dalam program dapat divalidasi untuk memastikan bahwa seluruh instruksi berjalan sesuai dengan alur sistem yang telah direncanakan (Idris et al., 2025).

Berdasarkan struktur *flow graph*, dilakukan perhitungan *Cyclomatic Complexity* (CC) untuk menentukan kuantitas jalur independen di dalam kode program. Sejalan dengan pendapat (Pratiwi & Widiarti, 2025), nilai *Cyclomatic Complexity* ini menjadi parameter dalam menentukan jumlah minimal skenario pengujian yang diperlukan guna menjamin

reliabilitas sistem. Formula yang digunakan untuk menghitung nilai tersebut adalah:

$$V(G) = E - N + 2$$

Keterangan:

$V(G)$: Nilai *Cyclomatic Complexity* (kompleksitas struktural).

E : Jumlah *edge* (sisi penghubung) pada *flow graph*.

N : Jumlah *node* (simpul instruksi) pada *flow graph*.

Setiap jalur independen yang dihasilkan dari perhitungan ini wajib diuji melalui skenario tertentu untuk memastikan bahwa seluruh percabangan logika dalam program telah tervalidasi. Dengan demikian, penerapan *white box testing* berbasis *basis path* tidak hanya berfungsi sebagai alat ukur kerumitan kode, tetapi juga sebagai mekanisme penjaminan mutu bahwa struktur internal sistem telah berjalan sesuai dengan spesifikasi rancangan yang telah ditetapkan. Dalam penelitian ini, *White Box Testing* akan difokuskan pada logika program dalam modul perhitungan *safety stock* untuk memastikan jalur keputusan (*decision path*) dalam menentukan ambang batas stok sudah akurat secara kode.

Dari pengertian di atas, dapat disimpulkan bahwa *White Box Testing* merupakan metode pengujian perangkat lunak yang berfokus pada validasi struktur internal dan logika prosedural melalui analisis mendalam terhadap kode sumber (*source code*). Dengan menerapkan teknik seperti *Basis Path Testing* dan penggunaan parameter *Cyclomatic Complexity* ($V(G) = E - N + 2$), metode ini mampu mengukur kompleksitas jalur logis serta menjamin bahwa setiap percabangan dalam sistem telah tervalidasi secara akurat. Dalam pengembangan sistem informasi, pengujian ini berperan sebagai mekanisme penjaminan mutu yang krusial untuk memastikan bahwa alur instruksi program tidak hanya berfungsi secara fungsional, tetapi juga memiliki reliabilitas tinggi sesuai dengan spesifikasi rancangan teknis yang telah ditetapkan.

B. *Black Box Testing*

Black Box Testing merupakan teknik pengujian perangkat lunak yang berfokus pada evaluasi nilai keluaran (*output*) berdasarkan masukan (*input*) tertentu tanpa perlu mendalami detail struktur kode atau logika internal program. Melalui simulasi pengisian data pada setiap formulir yang tersedia, pengujian ini dapat memastikan bahwa seluruh fungsi sistem telah beroperasi secara stabil dan mampu menghasilkan *output* yang selaras dengan spesifikasi kebutuhan pengguna. Dalam penelitian ini, *Black Box Testing* digunakan untuk memverifikasi fungsionalitas fitur manajemen rantai pasok, seperti keakuratan notifikasi *safety stock* dan keberhasilan transaksi data pesanan kepada pemasok, guna memastikan sistem telah sesuai dengan kebutuhan operasional UMKM Nawang Moments (Idris et al., 2025).

Implementasi pengujian *black box* dilakukan secara sistematis guna memvalidasi bahwa seluruh fungsionalitas aplikasi telah beroperasi sesuai dengan dokumentasi kebutuhan pengguna. Merujuk pada teori yang dikemukakan oleh Dalle et al. (2024), proses pengerjaan pengujian ini terbagi ke dalam 4 (empat) tahapan utama, sebagai berikut:

- 1) Identifikasi Masukan (*Input*): Tahap awal yang dilakukan adalah menentukan unit fungsional atau fitur spesifik di dalam aplikasi yang akan menjadi objek pengujian.
- 2) Penyusunan Skenario Uji: Membuat rancangan prosedur pengujian yang komprehensif, lengkap dengan hasil yang diharapkan (*expected result*) sebagai parameter tolak ukur kebenaran sistem.
- 3) Eksekusi Pengujian: Melakukan pengetesan secara langsung dengan menjalankan skenario yang telah disusun pada aplikasi untuk memverifikasi respon sistem dalam kondisi nyata.
- 4) Evaluasi dan Penarikan Kesimpulan: Menetapkan status keberhasilan setiap fitur melalui perbandingan antara hasil

aktual yang diperoleh dengan hasil yang telah direncanakan sebelumnya.

Dari pengertian di atas, dapat disimpulkan bahwa *Black Box Testing* merupakan metode pengujian perangkat lunak yang berfokus pada validasi fungsionalitas sistem berdasarkan input dan output tanpa mempertimbangkan struktur logika internal. Melalui tahapan sistematis mulai dari identifikasi input hingga evaluasi hasil, metode ini berfungsi sebagai instrumen untuk memastikan bahwa seluruh fitur aplikasi telah beroperasi sesuai dengan spesifikasi kebutuhan pengguna. Dalam konteks pengembangan sistem, pengujian ini menjadi langkah krusial untuk menjamin kualitas antarmuka (*user interface*) dan keandalan fungsionalitas operasional, sehingga sistem yang dibangun dapat memberikan hasil yang akurat, stabil, dan relevan dengan tujuan penggunaan akhir.

C. UAT

Menurut (Aprilia & Dermawan, 2024), *User Acceptance Test* (UAT) merupakan fase pengujian sistem yang dilakukan secara langsung oleh pengguna akhir (*end-user*) untuk memastikan bahwa perangkat lunak yang dibangun telah sesuai dengan kebutuhan operasional. Output dari proses ini adalah dokumen formal penerimaan sistem serta kesepakatan tertulis yang menyatakan bahwa aplikasi telah memenuhi seluruh kriteria kebutuhan pengguna.

Dari pengertian di atas, dapat disimpulkan bahwa *User Acceptance Test* (UAT) merupakan tahap validasi krusial yang menempatkan pengguna akhir sebagai penentu keberterimaan sistem. Melalui pengujian ini, keselarasan antara fitur yang dibangun dengan kebutuhan operasional nyata di lapangan dapat dipastikan, sehingga sistem yang dihasilkan tidak hanya layak secara teknis, tetapi juga tepat guna dalam mendukung alur kerja bisnis secara efektif dan berkelanjutan.

2.1.7 Teknologi Pendukung Sistem (Opsional dan Kontekstual)

Teknologi pendukung sistem merupakan kumpulan konsep teknologi yang digunakan untuk menunjang proses operasional dan analisis pada sistem informasi yang dikembangkan. Teknologi ini berperan dalam memastikan sistem dapat berjalan sesuai kebutuhan pengguna. Adapun teknologi pendukung yang digunakan dalam sistem informasi *Supply Chain Management* Nawang Moments adalah sebagai berikut:

A. Sistem Berbasis Web

Website atau *World Wide Web (WWW)* merupakan infrastruktur jaringan internet yang berfungsi mengintegrasikan dokumen secara lokal maupun global melalui media *hypertext*. Dokumen-dokumen digital ini, yang dikenal sebagai *web page*, memungkinkan pengguna melakukan navigasi dan berpindah antarhalaman secara berkesinambungan meskipun data tersimpan pada peladen (*server*) yang berbeda di seluruh dunia (Zulfa & Arifudin, 2025). Dalam konteks SCM Nawang Moments, teknologi berbasis web berfungsi sebagai platform utama yang memberikan aksesibilitas jarak jauh bagi pengguna. Hal ini memungkinkan pemantauan stok bahan baku, manajemen pemasok, dan pengawasan distribusi dilakukan secara *real-time* melalui penjelajah web tanpa terbatas oleh perangkat tertentu.

B. Framework MVC (Laravel)

Menurut Sofyan et al. (2024), *framework* merupakan kerangka kerja yang berfungsi menyediakan fungsi dasar guna membangun aplikasi secara lebih terorganisir melalui penggunaan arsitektur *Model-View-Controller* (MVC). Dalam pengembangan sistem SCM Nawang Moments, teknologi ini berperan sebagai pengatur alur kerja yang memisahkan antara logika pengolahan data persediaan (*Model*), antarmuka pengguna website (*View*), serta penghubung instruksi operasional (*Controller*). Implementasi kerangka kerja ini bertujuan untuk menjamin keamanan data transaksi, meningkatkan efisiensi pemrosesan data stok, serta mempermudah pemeliharaan sistem agar

dapat berjalan secara stabil dan adaptif terhadap kebutuhan operasional UMKM.

C. Xampp

XAMPP merupakan paket perangkat lunak bebas lintas platform yang mengintegrasikan berbagai program pendukung dalam satu kompilasi untuk menjalankan fungsi *server* lokal (*localhost*). Di dalam paket ini, terdapat komponen utama seperti Apache HTTP Server, basis data MySQL, serta interpreter untuk bahasa pemrograman PHP dan Perl yang dapat digunakan untuk mendukung pengembangan sistem secara mandiri. XAMPP dapat dioperasikan sebagai perangkat lunak sumber terbuka yang menyediakan paket distribusi Apache, MariaDB, PHP, dan Perl untuk kebutuhan peladen lokal (*local server*). Layanan *server* lokal ini dapat berjalan secara mandiri pada perangkat komputer atau laptop guna memfasilitasi tahap pengujian aplikasi sebelum situs web diunggah ke server web jarak jauh yang sebenarnya. (Idris et al., 2025).

D. PHP

PHP merupakan bahasa pemrograman *server-side* yang terintegrasi dengan HTML untuk membangun halaman web dinamis melalui kemampuan dalam menerima, mengolah, serta menyajikan data. Proses kerja bahasa ini dapat dilakukan dengan mengolah data pada *database server* untuk kemudian hasilnya dapat ditampilkan kembali pada lapisan *browser* di sebuah situs. PHP dapat didefinisikan sebagai bahasa pemrograman *server-side scripting* yang menjalankan seluruh proses pengolahan datanya pada sisi server sebelum mengirimkan hasilnya kepada pengguna. Sebagai bahasa berbasis kode skrip, PHP dapat mengolah data secara dinamis dan mengonversinya kembali menjadi kode HTML agar dapat ditampilkan melalui *web browser* klien (Fauzan et al., n.d.).

E. Database Management System (MySQL)

MySQL merupakan sebuah sistem manajemen basis data relasional (*Relational Database Management System* - RDBMS) yang bersifat

sumber terbuka (*open source*) dan menggunakan bahasa standar *Structured Query Language* (SQL). MySQL berfungsi sebagai media penyimpanan data yang efisien, berkinerja tinggi, dan memiliki tingkat keamanan yang baik untuk aplikasi berbasis web (Firnando et al., 2023).

F. Draw.io

Draw.io merupakan sebuah Aplikasi yang digunakan untuk membuat rancangan diagram tanpa diperlukan instalasi aplikasi, cukup dengan sambungan internet. *Draw.io* merupakan aplikasi yang bersifat fleksibel, datanya dapat disimpan dimana saja dan dapat menggunakan media penyimpanan sendiri. *Draw.io* platform berbasis web yang dirancang khusus untuk menyusun diagram UML dengan antarmuka responsif yang dapat diakses secara daring. Layanan ini dapat diintegrasikan secara langsung dengan Google Drive untuk mempermudah penyimpanan dokumen, sehingga menjadikannya solusi alternatif yang efektif dalam mempercepat proses pembuatan berbagai diagram teknis (Marthiawati et al., 2024).

G. HTML

HTML atau *Hypertext Markup Language* merupakan bahasa dasar yang digunakan dalam pembuatan halaman website. HTML berfungsi untuk menyusun dan menampilkan berbagai elemen pada halaman web, seperti teks, gambar, maupun multimedia agar dapat ditampilkan dengan baik melalui browser. Dalam penggunaannya, HTML tersusun dari kumpulan tag yang ditandai dengan simbol “<” dan “>” untuk mendefinisikan struktur serta isi halaman web.

Seiring perkembangan teknologi web, HTML terus mengalami pembaruan guna mendukung pengembangan website yang lebih interaktif dan fleksibel. Selain digunakan untuk menampilkan konten, HTML juga dapat dikombinasikan dengan bahasa pemrograman lain untuk meningkatkan kemampuan dan interaktivitas halaman web. Saat ini, pengembangan HTML telah mencapai versi HTML5 yang

menyediakan berbagai fitur baru dalam mendukung pembuatan aplikasi dan website moder (Christian & Voutama, 2024).

H. Bootstrap

Bootstrap merupakan framework front-end yang digunakan untuk membantu proses pengembangan website dengan memanfaatkan HTML, CSS, dan JavaScript. Framework ini dirancang untuk mempermudah serta mempercepat pembuatan tampilan website yang responsif, sehingga tampilan website dapat menyesuaikan ukuran layar pada berbagai perangkat, baik desktop, tablet, maupun mobile device. Selain itu, Bootstrap juga mendukung kompatibilitas pada berbagai browser populer seperti Chrome, Firefox, Opera, dan Internet Explorer.

Bootstrap menyediakan kumpulan file CSS, JavaScript, serta library pendukung yang telah dilengkapi berbagai class dan komponen siap pakai. Komponen tersebut meliputi navigasi, tombol, form, card, pagination, progress bar, dan berbagai elemen antarmuka lainnya yang dapat digunakan untuk membangun tampilan website secara lebih terstruktur, rapi, dan menarik. Dengan adanya sistem layout dan fitur responsif yang dimiliki Bootstrap, pengembang dapat lebih mudah dalam mengembangkan website tanpa harus membuat desain antarmuka dari awal (Suprayogi & Rahmanesa, 2019).

I. Javascript

JavaScript merupakan bahasa pemrograman yang digunakan dalam pengembangan website dan termasuk ke dalam kategori client-side programming language. Bahasa ini pertama kali diperkenalkan oleh Netscape pada tahun 1995 dengan nama “LiveScript” dan digunakan pada browser Netscape Navigator 2. JavaScript terdiri dari kumpulan kode program yang dijalankan pada dokumen HTML untuk mendukung proses pengembangan web.

Dalam perkembangannya, JavaScript menjadi salah satu bahasa skrip utama yang digunakan untuk menciptakan halaman web yang interaktif dan dinamis. Eksekusi kode JavaScript dilakukan melalui web

browser pada sisi pengguna, sehingga website dapat memberikan respons secara langsung terhadap berbagai interaksi pengguna. Selain itu, JavaScript juga mendukung penggunaan event atau peristiwa tertentu yang dapat meningkatkan kemampuan HTML dalam menampilkan fitur yang lebih fleksibel dan interaktif pada aplikasi web (Christian & Voutama, 2024).

J. Rich Picture

Rich Picture merupakan suatu bentuk representasi visual yang digunakan untuk menggambarkan kondisi suatu sistem secara menyeluruh dan sederhana agar lebih mudah dipahami. Penyajian Rich Picture dilakukan melalui gambar atau ilustrasi yang memuat berbagai unsur dalam sistem, seperti pihak yang terlibat, proses, struktur, objek, serta permasalahan yang terjadi dalam suatu kasus atau proses bisnis.

Rich Picture juga dapat digunakan sebagai media pencatatan yang memiliki fungsi serupa dengan narasi, namun disampaikan dalam bentuk visual sehingga membantu peneliti dalam memahami situasi masalah secara lebih jelas dan terstruktur. Dengan menggunakan ilustrasi gambar, peneliti dapat melihat hubungan antar komponen dalam sistem serta mengidentifikasi permasalahan yang terjadi secara lebih mudah (Kevin & Widjaja, 2024).

K. Fishbone

Fishbone Diagram atau diagram tulang ikan merupakan salah satu metode yang digunakan untuk mengidentifikasi serta menganalisis penyebab terjadinya suatu permasalahan. Diagram ini dikenal juga dengan istilah *cause and effect diagram* atau diagram sebab-akibat karena digunakan untuk menggambarkan hubungan antara suatu masalah dengan faktor-faktor penyebabnya. Metode ini pertama kali diperkenalkan oleh Dr. Kaoru Ishikawa, seorang ilmuwan asal Jepang pada tahun 1960-an, sehingga Fishbone Diagram juga sering disebut sebagai Diagram Ishikawa.

Pada awal penggunaannya, Fishbone Diagram banyak diterapkan dalam manajemen kualitas untuk membantu proses pengendalian mutu dengan menggunakan data yang bersifat kualitatif atau verbal. Diagram ini membantu pengguna dalam mengelompokkan berbagai faktor penyebab masalah ke dalam beberapa kategori sehingga proses analisis menjadi lebih sistematis dan mudah dipahami. Selain memperkenalkan Fishbone Diagram, Dr. Kaoru Ishikawa juga dikenal sebagai pelopor tujuh alat pengendalian kualitas (*7 tools of quality control*), seperti control chart, histogram, scatter diagram, pareto chart, run chart, dan flowchart (Kurnia & Nasarudin, 2023).

L. Sistem Notifikasi Persediaan (*Real-time Notification System*)

Menurut Monalisa & Apsyarin (2021), sistem notifikasi dalam manajemen rantai pasok berfungsi sebagai instrumen komunikasi otomatis yang memberikan informasi status ketersediaan barang secara *real-time*. Dalam pengembangan sistem SCM Nawang Moments, teknologi ini digunakan sebagai sarana pendukung keputusan (*decision support*) bagi Admin. Sistem akan memicu pesan peringatan visual pada antarmuka website saat saldo bahan baku menyentuh titik minimum berdasarkan perhitungan *Safety Stock*. Fungsi utamanya adalah memastikan pengadaan kembali (*restock*) dilakukan tepat waktu untuk menghindari risiko kekosongan bahan baku (*stockout*) yang dapat menghambat produksi kue dan buket.

M. Visualisasi Data (Dashboard)

Visualisasi data didefinisikan sebagai metode pembelajaran berbasis visual yang bertujuan untuk menerjemahkan informasi melalui penggabungan elemen grafis seperti diagram dan bagan. Menurut Irfanullah et al. (2023), penyajian data dalam format visual sangat efektif untuk memudahkan pengguna dalam memahami informasi yang kompleks. Dalam sistem SCM Nawang Moments, konsep ini diterapkan melalui *dashboard* interaktif yang merangkum berbagai data mentah

menjadi pengetahuan yang mudah dicerna. Hal ini memungkinkan pemilik usaha untuk memantau indikator kinerja utama dan tren persediaan secara cepat guna mendukung pengambilan keputusan bisnis yang lebih tepat sasaran.

Dari pemaparan teknologi di atas, dapat disimpulkan bahwa pemilihan teknologi pendukung yang tepat merupakan elemen fundamental dalam arsitektur sistem informasi manajemen rantai pasok. Integrasi antara platform berbasis *web*, arsitektur *MVC*, bahasa pemrograman dinamis, serta sistem manajemen basis data yang relasional, membentuk ekosistem aplikasi yang stabil, responsif, dan mampu mengolah data secara *real-time*. Selain itu, penggunaan alat bantu pemodelan dan visualisasi data yang tepat tidak hanya mempercepat proses pengembangan, tetapi juga meningkatkan kemampuan sistem dalam menyajikan informasi yang komunikatif bagi pengambilan keputusan manajerial. Secara keseluruhan, sinergi teknologi tersebut menjadi perangkat utama dalam mewujudkan sistem SCM yang efisien, aman, dan adaptif terhadap kebutuhan operasional UMKM.

2.1.8 Penelitian Sebelumnya

Tabel 2. 6 Tabel Perbandingan Penelitian Sebelumnya

No	Judul, Penulis, Tahun	Masalah Penelitian	Tujuan Penelitian	Metodelogi / Pemecahan Masalah	Jumlah Data / Sampel	Hasil Penelitian / Temuan Utama	Limitasi / Keterbatasan	Persamaan & Perbedaan dengan Penelitian Saat Ini
1	Sistem Informasi Manajemen Stok dan Produksi Pakaian Anak Berbasis Web pada UMKM Linda Collection Menggunakan Metode Safety Stock (Iskandar et al., 2025)	Menghadapi kendala dalam pengelolaan stok dan proses produksi akibat pencatatan manual yang mengakibatkan ketidaksesuaian data.	Mengembangkan sistem informasi manajemen stok dan produksi berbasis web yang dilengkapi dengan metode Safety Stock.	Pendekatan System Development Life Cycle (SDLC) model waterfall.	Data stok bahan baku dan transaksi produksi pakaian anak.	Sistem yang dibangun memungkinkan pengguna memantau stok secara real-time dan menghitung persediaan pengaman.	Fokus pada manajemen stok internal UMKM tanpa fitur pengadaan ke supplier.	Persamaan: Sama-sama menggunakan <i>Safety Stock</i> dan platform Web. Perbedaan: Jurnal meneliti industri konveksi yang bahannya tahan lama, sedangkan penelitian ini meneliti bahan kue yang memiliki masa kedaluwarsa.

2	Analisis Perhitungan Dalam Optimalisasi Manajemen Inventori Pada Pengadaan Bahan Baku dengan Metode Safety Stock di PT ABC (Nafi et al., 2025)	Mengantisipasi fluktuasi permintaan dan keterlambatan pengiriman, sehingga mengurangi risiko kekurangan bahan baku.	Menganalisis perhitungan dalam mengoptimalkan manajemen inventori pada pengadaan bahan baku dengan metode safety stock.	Menggunakan data historis penjualan dan lead time pengadaan untuk menghitung tingkat safety stock.	Data historis pengadaan bahan baku di PT ABC selama periode tertentu.	Implementasi metode safety stock dapat meningkatkan efisiensi manajemen inventori dan mengurangi biaya penyimpanan.	Penelitian bersifat analitis-matematis dan belum berbentuk sistem informasi digital.	Persamaan: Fokus pada pengadaan bahan baku hulu. Perbedaan: Jurnal hanya berupa analisis perhitungan manual/teoretis, sedangkan penelitian ini mengimplementasikannya ke dalam rancang bangun sistem web.
3	Manajemen Persediaan Menggunakan Metode Safety Stock dan Reorder Point (Brahmantyo et al., 2023)	Tidak ada perhitungan permintaan stok barang, tidak ada pengelolaan safety stock, dan tidak terdeteksinya	Mengurangi keterlambatan pengiriman barang serta dapat memastikan adanya suatu persediaan	Metode yang digunakan yaitu Safety Stock bertujuan untuk mengetahui jumlah stok	Data transaksi barang masuk/keluar di PT. Indonesia Berkah Mandiri.	Sistem yang dapat melakukan manajemen persediaan, menghemat waktu, dan meminimalisir	Sistem belum mendukung integrasi rantai pasok (SCM) secara luas ke pihak pemasok.	Persamaan: Objek sama-sama industri kuliner. Perbedaan: Jurnal fokus pada kontrol gudang satu arah, sedangkan penelitian ini membangun alur SCM yang menghubungkan admin

		antara data pengiriman dan data stok awal.	melalui safety stock.	aman dalam gudang.		kesalahan dalam persediaan barang.		gudang dengan supplier secara dua arah.
4	Analisis Pengendalian Persediaan Bahan Baku Dengan Metode Economic Order Quantity (Eoq), Safety Stock, Dan Reorder Point (Study Kasus Ukm Bubur Ayam Alhamdulillah)	Hanya mengandalkan perkiraan kasar untuk menentukan jumlah bahan mentah yang perlu dibeli dan kapan harus melakukan pemesanan ulang.	Menentukan tingkat persediaan keselamatan yang harus disiapkan, dan mengidentifikasi waktu yang tepat untuk melakukan pemesanan ulang.	Metode penelitian yang digunakan adalah kuantitatif deskriptif melalui wawancara dan observasi.	17 jenis bahan baku utama pada UMKM Bubur Ayam Alhamdulillah.	Penggunaan metode terstruktur dapat menentukan persediaan keselamatan untuk menghindari kekurangan bahan baku.	Tidak adanya aplikasi atau software untuk monitoring stok secara otomatis.	Persamaan: Studi kasus pada UMKM kuliner. Perbedaan: Kondisi di jurnal hanya memberikan rekomendasi perhitungan bagi pemilik, sedangkan penelitian ini memberikan solusi berupa aplikasi pemantauan stok otomatis.

	(Sahabuddin Et Al., 2024)							
5	Pengembangan Sistem Informasi Manajemen Persediaan Barang Berbasis Website Menggunakan Metode Safety Stock Dan Reorder Point (Studi kasus UD. AL-BAROKAH) (Nazila Ramadina, n.d.)	Permasalahan khususnya di bagian manajemen persediaan antara lain tidak ada perhitungan permintaan stok barang.	Membuat aplikasi Manajemen Persediaan berbasis website agar dapat mengurangi keterlambatan pengiriman barang.	Metode yang digunakan yaitu Safety Stock untuk stok aman dan Reorder Point untuk pengadaan ulang.	Data komponen besi tua untuk alat berat di UD. AL-BAROKAH.	Sistem yang dapat mengelola manajemen persediaan dan meminimalisir kesalahan stok barang.	Karakteristik barang tidak memiliki risiko penyusutan kualitas atau kedaluwarsa.	Persamaan: Pengembangan sistem informasi berbasis web. Perbedaan: Objek di jurnal adalah komponen besi (barang mati), sedangkan penelitian ini menangani bahan baku makanan dan buket yang memiliki fluktuasi permintaan musiman (wisuda/hari raya).

2.1.9 Kerangka Teoritis (*Theoretical Framework*)

Kerangka teoritis merupakan hasil sintesis dari teori-teori relevan dan analisis penelitian terdahulu yang digunakan sebagai landasan dalam menyelesaikan masalah persediaan pada penelitian ini.

Tabel 2. 7 Kerangka Teoritis

