BAB II
TINJAUAN PUSTAKA

2.1 Pengertian Sistem

Sistem merupakan sekumpulan komponen yang saling berhubungan, berinteraksi, dan bekerja sama untuk mencapai tujuan tertentu. Setiap komponen dalam sistem memiliki fungsi masing-masing yang saling mendukung sehingga menghasilkan keluaran yang sesuai dengan tujuan yang telah ditetapkan. Selain itu, sistem terdiri atas beberapa unsur, yaitu masukan (*input*), proses (*process*), keluaran (*output*), batas sistem (*boundary*), lingkungan (*environment*), dan pengendalian (*control*) (Efendi dkk., 2023).

2.2 Informasi

Informasi merupakan data yang telah diklasifikasikan atau diinterpretasikan sehingga menjadi lebih bermakna dan dapat digunakan dalam proses pengambilan keputusan. Data yang sebelumnya belum memiliki arti akan memberikan nilai tambah setelah melalui proses pengolahan sehingga dapat dimanfaatkan oleh pengguna untuk mendukung aktivitas operasional maupun pengambilan keputusan. Selain itu, informasi juga dapat dipahami sebagai sekumpulan fakta atau data yang diorganisasikan dengan cara tertentu sehingga mempunyai arti bagi penerimanya. Dengan demikian, informasi merupakan

hasil pengolahan data yang lebih berguna, bermakna, dan memiliki manfaat bagi pengguna (Lestari dkk., 2023).

2.3 Sistem Informasi

Sistem informasi merupakan suatu sistem yang berfungsi untuk mengolah data melalui serangkaian proses, mulai dari pengumpulan data, pengolahan, analisis, penyimpanan, hingga penyebaran informasi kepada pengguna. Informasi yang dihasilkan digunakan untuk memenuhi kebutuhan individu maupun organisasi dalam mendukung pelaksanaan kegiatan operasional dan proses pengambilan keputusan. Dengan adanya sistem informasi, proses pengelolaan data dapat dilakukan secara lebih efektif, efisien, dan terstruktur sehingga mampu menghasilkan informasi yang bermanfaat bagi organisasi (Sari dkk., 2024).

2.4 Penjualan

Penjualan merupakan kegiatan transaksi antara penjual dan pembeli yang dilakukan berdasarkan kesepakatan kedua belah pihak, di mana pembeli memperoleh barang atau jasa dengan memberikan sejumlah imbalan sesuai nilai yang telah disepakati. Penjualan menjadi salah satu aktivitas utama perusahaan karena berperan dalam menghasilkan pendapatan serta mendukung keberlangsungan operasional perusahaan. Berdasarkan mekanisme transaksinya,

penjualan dibedakan menjadi penjualan tunai dan penjualan kredit, yang masing-masing memiliki prosedur pencatatan dan pengelolaan pembayaran yang berbeda. Dalam pelaksanaannya, kegiatan penjualan melibatkan beberapa fungsi yang saling berkaitan, seperti bagian penjualan, kas, gudang, pengiriman, dan akuntansi, sehingga proses transaksi dapat berjalan secara efektif, akurat, dan terdokumentasi dengan baik (Nova & Utomo, 2023).

2.5 Sistem Informasi Penjualan

Sistem informasi penjualan merupakan serangkaian prosedur yang dirancang untuk mengelola seluruh aktivitas penjualan, mulai dari pencatatan pesanan, pengolahan transaksi, pembuatan dokumen, hingga penyajian informasi yang dibutuhkan oleh manajemen maupun bagian lain dalam perusahaan. Sistem ini berfungsi untuk mengolah data transaksi penjualan menjadi informasi yang akurat sehingga dapat mendukung kegiatan operasional, pengendalian persediaan, penyusunan laporan penjualan, serta pengambilan keputusan. Selain itu, penerapan sistem informasi penjualan juga memberikan berbagai manfaat, seperti membantu penyusunan strategi penjualan, mempermudah pengelolaan stok barang, menghasilkan laporan penjualan secara otomatis, serta meningkatkan efisiensi dan efektivitas proses bisnis perusahaan (Al Muftin & Hidayat, 2023).

2.6 Point Of Sale

Point of Sale (POS) merupakan sistem yang digunakan untuk mendukung proses transaksi penjualan dengan mengintegrasikan berbagai aktivitas, seperti pencatatan data barang, perhitungan transaksi, pengelolaan stok, pencetakan struk, serta penyusunan laporan penjualan secara otomatis. Penerapan sistem POS bertujuan untuk meningkatkan kecepatan dan ketepatan pelayanan kepada pelanggan, mengurangi kesalahan dalam pencatatan transaksi, serta mempermudah pemilik usaha dalam mengelola data penjualan dan persediaan barang. Dengan memanfaatkan sistem POS berbasis *web*, seluruh data transaksi dapat tersimpan secara terstruktur, mudah diakses, dan menghasilkan informasi yang lebih efektif sebagai dasar pengambilan keputusan (Fatich dkk., 2023).

2.7 UMKM (Usaha Mikro, Kecil, dan Menengah)

Usaha Mikro, Kecil, dan Menengah (UMKM) merupakan kelompok usaha yang memiliki peran penting dalam mendorong pertumbuhan perekonomian nasional. Menurut Asmuni, Harahap, dan Saragih (2023), sektor UMKM mendominasi jumlah pelaku usaha di Indonesia serta memberikan kontribusi yang signifikan terhadap peningkatan Produk Domestik Bruto (PDB) dan penyerapan tenaga kerja. Selain itu, perkembangan teknologi mendorong pelaku UMKM untuk melakukan digitalisasi usaha melalui pemanfaatan teknologi informasi, pemasaran digital, dan sistem pembayaran digital agar

mampu meningkatkan daya saing serta beradaptasi dengan perkembangan ekonomi digital.

2.8 Teknologi *Barcode*

Barcode merupakan teknologi identifikasi otomatis yang digunakan untuk merepresentasikan data dalam bentuk kode yang dapat dibaca oleh perangkat pemindai *(barcode scanner).* Dalam sistem informasi, *barcode* berfungsi untuk mempercepat proses identifikasi barang, mengurangi kesalahan pencatatan secara manual, serta meningkatkan akurasi dan efisiensi pengelolaan data. Penerapan teknologi *barcode* yang diintegrasikan ke dalam sistem berbasis *web* memungkinkan proses pencatatan, pelacakan, dan pemantauan data dilakukan secara lebih cepat, akurat, dan real-time, sehingga mampu mendukung efektivitas operasional serta meningkatkan produktivitas pengguna dalam mengelola data barang (Zhafira dkk., 2025).

2.9 Teknologi Olah Citra

Teknologi olah citra (*image processing*) merupakan teknik yang digunakan untuk mengolah dan menganalisis citra digital agar informasi yang terkandung di dalamnya dapat dikenali oleh sistem komputer. Dalam proses pembacaan *barcode*, citra yang ditangkap oleh kamera akan diproses melalui beberapa tahapan, seperti konversi citra menjadi hitam putih (*thresholding*),

pengurangan *noise* menggunakan operasi morfologi, proses pendeteksian *barcode*, hingga *decoding* untuk memperoleh data yang tersimpan pada *barcode*. Penerapan teknologi olah citra memungkinkan proses identifikasi *barcode* dilakukan secara otomatis dengan tingkat akurasi yang tinggi sehingga dapat mempercepat proses pengambilan data dan mengurangi kesalahan dalam pencatatan informasi (Najib, 2022). Pada penelitian ini, teknologi olah citra tidak hanya dimanfaatkan untuk proses pembacaan *barcode* menggunakan kamera, tetapi juga digunakan untuk mendeteksi barang secara langsung melalui kamera. Hasil deteksi kemudian dicocokkan dengan data barang yang tersimpan pada *database* sehingga hanya barang yang telah terdaftar yang dapat dikenali dan diproses pada transaksi penjualan.

2.10 Fitur Scan Barang

Fitur scan barang merupakan salah satu penerapan teknologi olah citra yang digunakan untuk mempercepat proses identifikasi produk pada sistem informasi penjualan. Pada penelitian ini, fitur scan barang terdiri atas dua metode, yaitu pemindaian *barcode* menggunakan kamera dan deteksi barang secara langsung menggunakan kamera. Pemindaian *barcode* dilakukan dengan membaca kode *barcode* yang terdapat pada produk sehingga sistem dapat menampilkan informasi barang secara otomatis. Sementara itu, deteksi barang secara langsung menggunakan kamera memungkinkan sistem mengenali produk

berdasarkan citra yang ditangkap kamera. Kedua metode tersebut bertujuan untuk meningkatkan kecepatan transaksi, mengurangi kesalahan input data, dan meningkatkan efisiensi pelayanan kepada pelanggan.

2.11 Website

Website merupakan media berbasis internet yang digunakan untuk menyampaikan informasi kepada pengguna secara daring melalui jaringan komputer yang saling terhubung. Menurut Sanusi, Ardiansyah, Ahmad, Usman, dan Saputra (2023), perkembangan teknologi informasi dan komunikasi mendorong perkembangan *website* sebagai media penyampaian informasi yang dapat diakses melalui jaringan internet. *Website* memberikan kemudahan bagi pengguna untuk memperoleh informasi kapan saja dan di mana saja, sehingga banyak dimanfaatkan sebagai media publikasi, promosi, maupun pengelolaan berbagai sistem informasi berbasis *web*.

2.12 Database

Basis data (*database*) adalah kumpulan data yang terstruktur secara sistematis dan tersimpan di suatu tempat agar dapat diakses, dimanipulasi, dan dikelola dengan mudah. Basis data digunakan untuk menyimpan informasi yang terkait satu sama lain dan dapat diakses oleh sistem atau aplikasi perangkat lunak (Siregar dkk., 2024).

Perangkat lunak yang dapat digunakan untuk mengelola basis data disebut sistem manajemen basis data atau *Database Management System* (DBMS). DBMS merupakan perangkat lunak yang dirancang untuk dapat melakukan pengaturan dan mengelola koleksi data dalam jumlah yang besar dan dapat memanipulasi data secara lebih mudah. DBMS merupakan *interface* atau antar muka antara pengguna basis data dengan data yang disimpan (Siregar dkk., 2024).

2.13 MySQL

MySQL merupakan *Relational Database Management System (RDBMS)* yang digunakan untuk mengelola basis data secara terstruktur dengan memanfaatkan *Structured Query Language (SQL).* Menurut Ardiansyah dan Fachrurrazi (2023), MySQL merupakan program *open source* yang sangat membantu dalam mengatur dan memelihara basis data sehingga dapat membangun *database* yang terorganisir dengan baik. Dengan kemampuannya dalam menyimpan, mengolah, memperbarui, dan menampilkan data secara efisien, MySQL banyak digunakan sebagai sistem manajemen basis data pada berbagai aplikasi berbasis *web* karena mudah diintegrasikan dengan bahasa pemrograman seperti PHP.

2.14 XAMPP

XAMPP merupakan perangkat lunak *open source* yang menyediakan paket lengkap untuk membangun lingkungan pengembangan aplikasi *web* secara lokal. Menurut Wahid, Imran, dan Al Mujaddidi (2024), XAMPP menyediakan berbagai layanan yang terdiri atas Apache sebagai *web* server, PHP sebagai bahasa pemrograman *server-side*, MySQL sebagai sistem manajemen basis data, serta Perl sebagai bahasa pemrograman. Keberadaan XAMPP memudahkan pengembang dalam menyiapkan lingkungan pengembangan lokal untuk membangun, menjalankan, dan menguji aplikasi *web* sebelum diimplementasikan pada server produksi.

2.15 Framework Laravel

Framework Laravel merupakan *Framework* berbasis PHP yang digunakan untuk mempermudah proses pengembangan aplikasi *web* dengan menyediakan struktur pengembangan yang terorganisir dan berbagai fitur pendukung. *Laravel* membantu pengembang dalam membangun aplikasi secara lebih cepat, aman, dan mudah dipelihara karena menyediakan fasilitas seperti pengelolaan *routing*, autentikasi, migrasi basis data, serta berbagai komponen yang mendukung pengembangan aplikasi modern. Dalam implementasinya, *Laravel* banyak dimanfaatkan untuk membangun sistem informasi berbasis *web* karena mampu mengoptimalkan kinerja PHP sehingga proses *web development*

menjadi lebih efisien, lebih aman, dan lebih sederhana (Christanto & Somya, 2023). Dalam penelitian ini, *Laravel* digunakan sebagai *Framework* utama untuk membangun sistem POS berbasis *web*. Penggunaan *Laravel* bertujuan agar sistem yang dikembangkan lebih terstruktur, mudah dipelihara, dan aman.

2.16 Unified Modeling Language (UML)

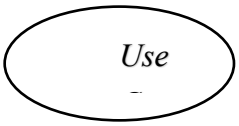
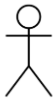
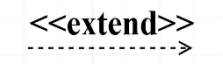
Unified Modeling Language (UML) merupakan bahasa pemodelan standar yang digunakan untuk merancang, memvisualisasikan, dan mendokumentasikan sistem perangkat lunak. Menurut Wahid, Imran, dan Al Mujaddidi (2024), UML menyediakan seperangkat notasi grafis yang digunakan untuk membuat model abstrak suatu sistem, seperti *Use Case* diagram, *Class* diagram, *Activity* diagram, dan *Sequence* diagram. Penggunaan UML membantu pengembang dalam memahami kebutuhan sistem, merancang struktur perangkat lunak, serta mempermudah proses pengembangan dan pemeliharaan sistem agar berjalan lebih terstruktur dan sistematis.

2.16.1 *Use Case* Diagram

Use Case Diagram merupakan salah satu diagram dalam *Unified Modeling Language* (UML) yang digunakan untuk menggambarkan interaksi antara aktor dengan sistem. Diagram ini berfungsi untuk mendeskripsikan kebutuhan fungsional sistem melalui hubungan antara pengguna (aktor) dengan

fungsi-fungsi yang disediakan oleh sistem. Menurut Dahtyar (2024), *Use Case* Diagram menggambarkan hubungan interaksi antara sistem dan aktor serta mendeskripsikan tipe interaksi yang dilakukan pengguna terhadap sistem sehingga memudahkan proses analisis kebutuhan dan perancangan sistem.

Tabel. 1.1 Simbol *Use Case*

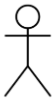
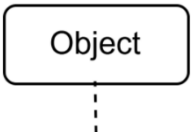
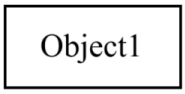
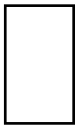
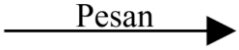
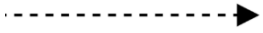
No	Simbol	Nama	Keterangan
1		*Use Case*	Fungsional yang disediakan sistem sebagai unit yang saling bertukar pesan antar unit atau aktor
2		*Actor*	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi di luar sistem itu sendiri
3		*Asosiasi*	Komunikasi antara aktor dan *Use Case* yang berpartisipasi pada *Use Case*
4		*Extend*	Perilaku *Use Case* yang memperluas perilaku *Use Case* yang lain

5		*Generalisasi*	Hubungan generalisasi dan spesialisasi antara dua buah *Use Case*
6		*Include*	Perilaku *Use Case* yang merupakan bagian dari *Use Case* yang lain

2.16.2 *Sequence* Diagram

Sequence Diagram merupakan salah satu diagram dalam *Unified Modeling Language* (UML) yang digunakan untuk menggambarkan interaksi antarobjek dalam suatu sistem berdasarkan urutan waktu. Diagram ini menunjukkan proses pertukaran pesan yang terjadi antarobjek sehingga alur komunikasi dalam sistem dapat dipahami dengan lebih jelas. Selain itu, *Sequence* Diagram membantu dalam memodelkan proses internal sistem, memastikan seluruh skenario penggunaan telah terdefinisi dengan baik, serta mengidentifikasi potensi permasalahan pada interaksi antarobjek. Menurut Syam, Mokram, dan Ranteliling (2024), *Sequence* Diagram merupakan alat visual UML yang menggambarkan interaksi antarobjek berdasarkan urutan waktu melalui pesan yang dikirim dan diterima, dengan komponen utama berupa aktor, lifeline, pesan (*message*), dan *activation box*.

Tabel. 1.2 Simbol *Sequence*

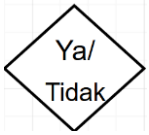
No	Simbol	Nama	Keterangan
1	 **Actor**	*Actor*	Menggambarkan seseorang atau sesuatu yang berinteraksi dengan sistem
2		*Lifeline*	Menyatakan kehidupan suatu objek dalam sistem
3		*Object*	Antarmuka yang sering berinteraksi
4		*Waktu aktif*	Menyatakan objek dalam keadaan aktif dan berinteraksi
5		*Messange*	Spesifikasi komunikasi antar objek yang memuat informasi tentang aktivitas yang terjadi
6		*Return Message*	Spesifikasi komunikasi antar objek yang memuat informasi balasan

2.16.3 *Activity* Diagram

Activity Diagram merupakan salah satu diagram dalam *Unified Modeling Language (UML)* yang digunakan untuk menggambarkan alur kerja (workflow)

atau urutan aktivitas yang terjadi di dalam suatu sistem. Diagram ini membantu memvisualisasikan proses bisnis maupun proses sistem perangkat lunak melalui serangkaian aktivitas yang saling berhubungan sehingga memudahkan pengembang dalam memahami alur proses yang akan dirancang. Menurut Sanjaya dan Yahfizham (2024), *Activity* Diagram digunakan untuk memvisualisasikan alur kerja atau aktivitas yang terjadi dalam suatu sistem maupun proses, sehingga dapat membantu dalam pemodelan proses bisnis serta aktivitas pada perangkat lunak.

Tabel. 1.3 Simbol *Activity*

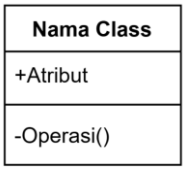
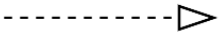
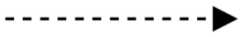
No	Simbol	Nama	Keterangan
1		*Initial Node*	Menggambarkan titik awal aktivitas
2		*Activity*	Menggambarkan aktivitas yang ada pada sistem
3		*Decission*	Pilihan untuk pengambilan suatu keputusan
4		*Fork/Join*	Penggabungan atau pemecahan lebih dari satu aktivitas

5		*Activity Final Node*	Menggambarkan Titik akhir aktivitas
6		*Swimeline*	Memisahkan organisasi yang bertanggung jawab terhadap aktivitas yang terjadi

2.16.4 *Class* Diagram

Class Diagram merupakan salah satu diagram dalam *Unified Modeling Language* (UML) yang digunakan untuk menggambarkan struktur statis suatu sistem. Diagram ini menunjukkan kelas-kelas yang terdapat dalam sistem beserta hubungan antar kelas, atribut, dan operasi (method) yang dimiliki oleh masing-masing kelas. Melalui *Class* Diagram, pengembang dapat memahami struktur objek serta hubungan antarobjek sehingga memudahkan proses perancangan sistem dan implementasi basis data. Menurut Akbar (2024), *Class* Diagram digunakan untuk menggambarkan tampilan statis suatu aplikasi, di mana komponen utamanya terdiri atas kelas-kelas beserta hubungan di antaranya. Setiap kelas merepresentasikan suatu konsep yang dapat memiliki atribut dan operasi, sedangkan hubungan antar kelas menunjukkan keterkaitan maupun pewarisan (inheritance) dalam sistem.

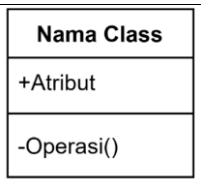
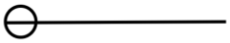
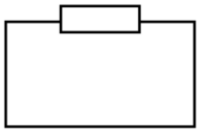
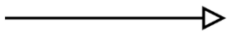
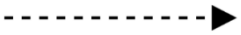
Tabel. 1.4 Simbol *Class*

No	Simbol	Nama	Keterangan
1		*Class*	Himpunan ojek yang berbagi atribut serta operasi yang sama
2		*Generalization*	Hubungan Dimana objek anak berbagi perilaku dan struktur data dari objek induk
3		*Realization*	Operasi yang benar-benar di lakukan oleh suatu objek
4		*Dependency*	Hubungan Dimana perubahan pada elemen mandiri mempengaruhi elemen yang bergantung padanya
5		*Association*	Menghubungkan objek satu dengan objek yang lain

2.16.5 *Object* Diagram

Menurut Susanto & Susilo (2023), *Object* Diagram adalah salah satu jenis diagram pada UML yang menggambarkan objek-objek pada suatu sistem dan hubungan antaranya.

Tabel. 1.5 Simbol *Object*

No	Simbol	Nama	Keterangan
1		*Instance specification*	Elemen yang mewakili sebuah contoh objek dalam sistem
2		*Link*	Hubungan konseptual antar objek dalam sistem
3		*Associasion*	Elemen yang menggambarkan hubungan antar objek yang dimodelkan
4		*Port*	Titik interaksi antara komponen dengan lingkungannya
5		*Generalization*	Cara berbagi atribut dan perilaku dari suatu kelas kepada kelas lain
6		*Dependency*	Relasi yang menunjukkan bahwa perubahan pada satu elemen mempengaruhi elemen lain

2.17 Black Box Testing

Black Box Testing merupakan metode pengujian perangkat lunak yang berfokus pada pengujian fungsi sistem berdasarkan masukan (input) dan keluaran

(output) tanpa memperhatikan struktur internal atau kode program. Menurut Haqqoni, Winarno, Musthofa, Sakdi, dan Saifudin (2024), *Black Box Testing* digunakan untuk memastikan setiap fungsi pada sistem berjalan sesuai dengan kebutuhan pengguna dengan memeriksa hasil keluaran yang dihasilkan dari setiap masukan yang diberikan. Melalui metode ini, kesalahan pada fungsi, antarmuka, maupun proses sistem dapat diidentifikasi sehingga kualitas perangkat lunak dapat ditingkatkan sebelum digunakan oleh pengguna.

2.18 Metode Waterfall

Metode *Waterfall* merupakan salah satu metode pengembangan perangkat lunak yang dilakukan secara bertahap dan berurutan, di mana setiap tahapan harus diselesaikan terlebih dahulu sebelum melanjutkan ke tahap berikutnya. Menurut Azfar dan Anggita (2024), metode *Waterfall* terdiri atas beberapa tahapan, yaitu analisis kebutuhan (requirement analysis), perancangan sistem (system design), implementasi (implementation), pengujian (testing), dan pemeliharaan (maintenance). Pendekatan ini memungkinkan proses pengembangan sistem dilakukan secara sistematis dan terdokumentasi dengan baik.