

BAB II

TINJAUAN PUSTAKA

2.1 Sistem Informasi

Menurut Syahputra dan Saptari (Syahputra & Saptari, 2023), Sistem Informasi (SI) adalah kombinasi dari teknologi informasi dan aktivitas orang yang menggunakan teknologi itu untuk mendukung operasi manajemen.

Menurut Praguna dan Nugroho, 2021 (Angelo & Ridho, 2022) , Sistem informasi adalah sistem di dalam perancangan yang mengumpulkan keperluan pemrosesan transaksi sehari-hari, membantu operasi, menyusun kegiatan strategis dan manajemen organisasi, dan menyediakan laporan pihak ketiga yang dibutuhkan.

Dari pendapat para ahli diatas, dapat disimpulkan bahwa Sistem Informasi (SI) merupakan suatu sistem yang mengintegrasikan teknologi informasi dengan aktivitas manusia untuk mendukung berbagai fungsi organisasi. SI tidak hanya digunakan untuk mendukung operasi manajemen, tetapi juga berperan dalam pemrosesan transaksi harian, perencanaan strategis, pengambilan keputusan manajerial, serta penyediaan informasi bagi pihak internal maupun eksternal organisasi. Dengan demikian, SI menjadi komponen penting dalam meningkatkan efisiensi dan efektivitas kinerja organisasi secara keseluruhan.

2.2 Pemesanan

Menurut Maulana, Firdaus dan Guslendra (Maulana dkk., 2024), Pemesanan adalah suatu aktifitas yang dilakukan oleh konsumen sebelum

membeli. Untuk mewujudkan kepuasan konsumen maka perusahaan harus mempunyai sebuah sistem pemesanan yang baik. Menurut Kamus Besar Bahasa Indonesia yang dimaksud pemesanan adalah proses, perbuatan, cara memesan (tempat, barang, dsb) kepada orang lain.

Menurut Suwarno, Hamimi, dan Edora (Suwarno dkk., 2023), Pemesanan adalah suatu aktifitas yang dilakukan oleh konsumen sebelum membeli. Untuk mewujudkan kepuasan konsumen maka perusahaan harus mempunyai sebuah sistem pemesanan yang baik.

Dari pendapat para ahli diatas, dapat disimpulkan bahwa pemesanan merupakan suatu aktivitas awal yang dilakukan oleh konsumen sebelum melakukan pembelian, yang berfungsi sebagai langkah penting dalam proses transaksi.

2.3 PNB

Menurut Purba (Purba, 2022), PNB adalah seluruh penerimaan yang diterima oleh negara dalam bentuk penerimaan dari sumber daya alam, bagian pemerintah atas laba badan usaha milik negara serta Penerimaan Negara Bukan Pajak lainnya.

2.4 Web Mobile

Menurut Rio dan Marsehan (Rio & Marsehan, 2023), *Web Mobile* atau *Mobile website* secara prinsip adalah sama dengan *website* secara umum, yang dibuat dengan *HTML*, *CSS*, *javascript*, *PHP*. Namun perbedaannya adalah *mobile website* dibuat agar optimal untuk ukuran layar *mobile device* seperti *smartphone* atau *tablet*.

Menurut Setiawan dan Eko (Setiawan & Eko, 2022), *Web Mobile* adalah aplikasi yang telah dirancang khusus untuk platform mobile (misalnya iOS, android, atau windows mobile).

Dari pendapat para ahli diatas, dapat disimpulkan bahwa *Web Mobile* merupakan bentuk pengembangan *website* atau aplikasi yang dirancang khusus untuk perangkat mobile seperti smartphone dan tablet. Meskipun secara teknis dibangun dengan teknologi *web* yang sama seperti *HTML*, *CSS*, *JavaScript*, dan *PHP*, *Web Mobile* memiliki fokus utama pada optimasi tampilan dan fungsi agar sesuai dengan ukuran layar dan sistem operasi perangkat mobile, seperti *iOS*, *Android*, atau *Windows Mobile*.

2.5 Hypertext Preprocessor (PHP)

Menurut Manuhutu dan Otniel, 2021 (Angelo & Ridho, 2022), *PHP* adalah bahasa perancangan yang bisa ditempatkan untuk membangun perangkat lunak. *PHP* merupakan salah satu bahasa pemrograman yang dapat dijalankan baik di bagian *server* maupun *side server*.

Menurut Maulana, Firdaus dan Guslendra (Maulana dkk., 2024), *PHP* adalah singkatan dari *Personal Home Page* yang merupakan bahasa standar yang digunakan dalam dunia *website*. *PHP* adalah bahasa pemrograman yang berbentuk script yang diletakkan didalam *web server*.

Dari pendapat para ahli diatas, dapat disimpulkan bahwa *PHP* merupakan bahasa pemrograman berbasis skrip yang dirancang untuk pengembangan *web* dan dijalankan di sisi *server*. *PHP* banyak digunakan dalam

dunia *website* karena kemampuannya untuk membangun perangkat lunak dan mengelola konten *web* secara dinamis melalui *web server*.

2.6 Visual Studio Code

Menurut Romzi dan Kurniawan (Romzi & Kurniawan, 2020), Visual Studio Code adalah *editor* kode sumber yang ringan namun kuat yang berjalan didesktop dan tersedia untuk Windows, macOS, dan Linux. Muncul dengan dukungan built-in untuk *JavaScript*, *TypeScript* dan *Node.js* dan memiliki ekosistem ekstensi yang kaya untuk bahasa lain (seperti *C ++*, *C #*, *Java*, *Python*, *PHP*, *Go*) dan runtime (seperti *.NET* dan *Unity*).

Menurut Ramdhan dan Nufriana (Firnando dkk., t.t.), Visual Studio Code adalah kode *editor* sumber yang dikembangkan oleh Microsoft untuk Windows, Linux dan mac OS. Visual Code memudahkan dalam penulisan *code* yang mendukung beberapa jenis bahasa pemrograman yang digunakan dan memberi variasi warna sesuai dengan fungsi dalam rangkaian *code* tersebut.

Dari pendapat para ahli diatas, dapat disimpulkan bahwa Visual Studio Code (VS Code) merupakan *editor* kode sumber yang ringan namun kuat, dikembangkan oleh Microsoft, dan tersedia untuk berbagai sistem operasi seperti Windows, macOS, dan Linux. VS Code mendukung berbagai bahasa pemrograman dan menyediakan fitur seperti pewarnaan sintaks, integrasi dengan berbagai runtime, serta ekosistem ekstensi yang kaya, sehingga memudahkan pengembang dalam menulis dan mengelola kode secara efisien.

2.7 *Payment Gateway*

Menurut Damanik, 2012 (Siahaan & Sianturi, 2024), *Payment Gateway* adalah cara untuk memproses pembayaran antara pelanggan, bisnis, dan bank yang dirancang untuk memfasilitasi transaksi pembelian, otorisasi pembayaran, dan deteksi penipuan.

Menurut Alfian dan Magdalena, 2020 (Fikri dkk., 2024), *Payment Gateway* adalah layanan pembayaran online yang bertugas untuk memverifikasi informasi transaksi berdasarkan aturan dari penyedia layanan.

Dari pendapat para ahli diatas, dapat disimpulkan bahwa *Payment Gateway* merupakan layanan yang memfasilitasi proses pembayaran online antara pelanggan, bisnis, dan institusi keuangan. Fungsinya mencakup verifikasi transaksi, otorisasi pembayaran, serta deteksi penipuan, sehingga memastikan transaksi berjalan secara aman, valid, dan sesuai dengan ketentuan penyedia layanan.

2.8 *Bootstrap*

Menurut Kaban (Anis dkk., 2022), *Bootstrap* adalah sebuah *framework* CSS yang paling banyak diminati oleh para pengembang *website*. Dengan memakai *bootstrap* kita dengan mudah dapat mendesign tampilan *website* yang responsif. Responsif maksudnya, lebar halaman *website* akan disesuaikan secara otomatis berdasarkan perangkat yang digunakan untuk mengaksesnya, baik itu ketika diakses menggunakan PC, Laptop, Tablet ataupun Smartphone maka *website* akan menyesuaikan dengan lebar perangkat yang di gunakan pengunjung.

Menurut Somya (Zakaria, 2022), *Bootstrap* adalah salah satu *framework CSS* yang biasa digunakan untuk memperindah tampilan suatu *website*, *Bootstrap* biasa digunakan oleh *front-end programmer* namun tidak menutup peluang juga apabila *back-end programmer* menggunakannya.

Dari pendapat para ahli diatas, dapat disimpulkan bahwa *Bootstrap* merupakan *framework CSS* yang banyak digunakan oleh pengembang *web* karena kemampuannya dalam membuat tampilan *website* yang responsif dan menarik. *Framework* ini memudahkan proses desain antarmuka yang otomatis menyesuaikan dengan berbagai ukuran perangkat, serta dapat digunakan oleh baik *front-end* maupun *back-end programmer*, menjadikannya alat yang fleksibel dan populer dalam pengembangan *web modern*.

2.9 Zachman

Menurut Singgirit dkk (Singgirit dkk. 2022), *Zachman Framework* merupakan *framework* arsitektural yang paling banyak dikenal dan diadaptasi. Para arsitek data enterprise mulai menerima dan menggunakan *framework* ini sejak pertama kali diperkenalkan oleh John A Zachman di IBM System Journal pada tahun 1987 dan kemudian dikembangkan pada tahun 1992 dengan tujuan untuk menyediakan struktur dasar organisasi yang mendukung akses, integrasi, interpretasi, pengembangan, pengelolaan, dan perubahan perangkat arsitektural dari sistem informasi organisasi (enterprise).

2.10 *Agile*

Menurut Apriliyani dkk (Apriliyani, Setiawan, dan Muchayan 2022), Konsep *Agile* Software Development dicetuskan oleh Kent Beck dan 16 rekannya dengan menyatakan bahwa *Agile* Software Development merupakan metode membangun software dengan melakukannya dan membantu orang lain.

Menurut Apriliyani dkk (Apriliyani dkk. 2022), *Agile* Development Methods merupakan sekelompok metodologi pengembangan perangkat lunak yang didasarkan pada prinsip-prinsip yang sama maupun pengembangan sistem jangka pendek yang membutuhkan adaptasi cepat dari pengembang terhadap perubahan dalam bentuk apapun. *Agile* development methods merupakan salah satu dari metodologi pengembangan perangkat lunak yang bersifat cepat, ringan, bebas bergerak, dan waspada.

Dari pendapat para ahli diatas, dapat disimpulkan bahwa metode *Agile* merupakan metode pengembangan perangkat lunak yang berfokus pada kolaborasi, kecepatan, dan fleksibilitas, dengan prinsip membangun sambil beradaptasi terhadap perubahan. Metode ini menekankan proses yang ringan, responsif, dan mudah menyesuaikan diri sehingga mampu menghasilkan sistem secara lebih cepat dan efektif sesuai kebutuhan pengguna.

2.11 *Xampp*

Menurut Noviantoro dkk (Noviantoro dkk., 2022), *Xampp* merupakan perangkat lunak berbasis *web server* yang bersifat *open source* (bebas), serta mendukung berbagai sistem operasi, baik Windows, Linux, atau Mac OS.

Xampp digunakan sebagai *standalone server* atau biasa disebut dengan *localhost*.

Menurut Madcoms, 2011 (Maulani, 2021), *XAMPP* adalah perangkat lunak yang paling umum digunakan untuk menjalankan *server apache* dan melakukan pengembangan *web* berbasis *PHP*. *XAMPP* sebenarnya merupakan gabungan dari beberapa perangkat lunak yang berhubungan dengan *server*, *web*, dan pengembangannya.

Dari pendapat para ahli diatas, dapat disimpulkan bahwa *XAMPP* merupakan perangkat lunak berbasis *web server* yang bersifat *open source* dan dapat dijalankan di berbagai sistem operasi seperti Windows, Linux, dan Mac OS. *XAMPP* digunakan sebagai *standalone server (localhost)* yang umum dimanfaatkan untuk pengembangan *web* berbasis *PHP*, karena merupakan gabungan dari beberapa komponen penting seperti *Apache*, *MySQL*, *PHP*, dan *Perl*, yang mendukung proses pembangunan dan pengujian aplikasi *web* secara lokal sebelum dipublikasikan secara online.

2.12 Mysql

Menurut Kurniawan, 2010 (Hermiati dkk., 2021), *MySQL* merupakan suatu jenis *Database server* yang sangat terkenal. *MySQL* termasuk jenis *RDBMS (Relational Database Manajement System)*. *MySQL* mendukung bahasa pemrograman *PH*, bahasa permintaan yang terstruktur, karena pada penggunaannya *SQL* memiliki beberapa aturan yang telah distandarkan oleh asosiasi yang bernama *ANSI*. *MySQL* merupakan *RDBMS (Relational Database Management System) server*.

2.13 Database

Menurut Wulandari dan Nurmiati (Wulandari, 2022), *Database* merupakan suatu kesatuan yang dibentuk dari gabungan tabel dan file, di mana setiap tabel terdiri dari record yang disusun atas field-field yang ada di dalamnya.

Menurut Arthalita dan Prasetyo (Arthalita & Prasetyo, 2020), Basis data (*Database*) adalah kumpulan data yang disimpan secara sistematis di dalam komputer yang dapat diolah atau dimanipulasi menggunakan perangkat lunak (program aplikasi) untuk menghasilkan informasi.

Dari pendapat para ahli diatas, dapat disimpulkan bahwa *Database* (basis data) merupakan kumpulan data yang tersusun secara sistematis, baik dalam bentuk tabel, file, record, maupun field, yang disimpan di dalam komputer. Data ini dapat diolah dan dimanipulasi menggunakan perangkat lunak untuk menghasilkan informasi yang berguna, menjadikan *Database* sebagai komponen penting dalam sistem pengelolaan informasi.

2.14 HypetText MarkUp Language (HTML)

Menurut Pahlevi dkk 2018 (Noviantoro dkk., 2022), *HTML (Hyper Text Mark Up Language)* merupakan bahasa yang digunakan untuk mendeskripsikan struktur sebuah halaman *web*. *HTML* berfungsi untuk mempublikasi dokumen online. Statement dasar dari *HTML* disebut tags. Sebuah tag dinyatakan dalam sebuah kurung siku (<>). Tags yang ditujukan untuk sebuah dokumen atau bagian dari suatu dokumen haruslah dibuat berupa pasangan.

Terdiri dari tag pembuka dan tag penutup. Dimana tag penutup menggunakan tambahan tanda garis miring (/) di awal nama tag.

2.15 Jaringan Komputer

Menurut Sugiyanta dan Raja, 2017 (Amala dkk., 2023), Komputer jaringan diidentifikasi sebagai jaringan komputer atau sekumpulan terminal yang terhubung dalam satu jaringan atau lebih banyak computer.

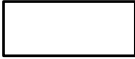
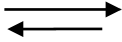
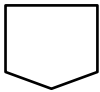
Menurut Pattipeilohy (Fadlan Abdillah Hasibuan & Subhiyanto, 2021), Jaringan komputer adalah sekelompok komputer otonom yang saling menggunakan protokol komunikasi melalui media komunikasi sehingga dapat berbagi data, informasi, program aplikasi dan perangkat seperti printer, scanner, CD Driver ataupun hardisk, serta memungkinkan untuk saling berkomunikasi secara elektronik.

Dari pendapat para ahli diatas, dapat disimpulkan bahwa jaringan komputer adalah sekumpulan komputer atau terminal yang saling terhubung dan berkomunikasi melalui protokol serta media komunikasi tertentu, yang memungkinkan pertukaran data, informasi, aplikasi, dan perangkat keras secara bersama-sama.

2.16 Aliran Sistem Informasi (ASI)

Menurut Ismael, 2017 (Adrianto, 2021), Aliran Sistem Informasi (ASI) merupakan bagan yang menunjukkan arus pekerjaan serta keseluruhan dari sistem. Bagan ini menjelaskan urutan dari prosedur - prosedur yang ada dalam sistem.

Table 2. 1 Simbol – simbol pada Aliran Sistem Informasi

NO	SIMBOL	KETERANGAN
1.		Proses Komputer
2		Dokumen
3.		Penghubung
4.		Garis Alir
5.		<i>Output</i> Monitor
6.		Manual Proses
7.		File Storage
8.		Penghubung Antar Halaman
9.		<i>Input</i> Menggunakan Keyboard
10		<i>Input-Output</i> Menggunakan Disket

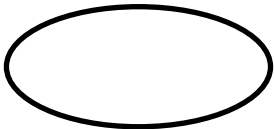
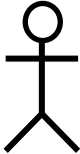
2.17 Unified Modeling Language (UML)

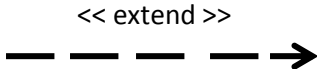
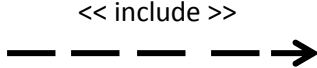
Menurut Saputra, 2023 (Siska Narulita dkk., 2024), *Unified Modelling Language (UML)* merupakan sebuah bahasa yang divisualisasikan dalam bentuk gambar atau grafik yang berfungsi untuk memberikan gambaran dan spesifikasi dalam pembangunan dan dokumentasi dari sebuah pengembangan sistem berorientasi objek (*Object oriented*).

2.15.1 Use Case Diagram

Menurut Wahyuda dan Theresiawati (Wahyuda & Theresiawati, 2022), *Use Case Diagram* merupakan kegiatan yang dilaksanakan suatu aktor, dimana setiap aktor memiliki *Use Case* diagram, yang menjadi model fungsional yang bisa mengilustrasi proses bisnissional yang dapat menggambarkan proses bisnis.

Table 2. 2 Simbol – simbol Use Case Diagram

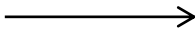
No	Simbol	Keterangan
1.	<i>Use Case</i> 	Fungsional yang disediakan sistem sebagai unit yang saling bertukar pesan antar unit atau <i>actor</i> : biasanya dinyatakan dengan menggunakan kata kerja diawal <i>frase name Use Case</i> .
2.	<i>Actor</i> 	Orang, proses atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat sendiri.

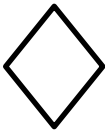
		Jadi walaupun simbol <i>actor</i> adalah gambar orang, biasanya dinyatakan menggunakan kata benda diawal frase nama.
3.	Asosiasi 	Komunikasi antar <i>actor</i> dan <i>usecase</i> yang berpartisipasi pada <i>usecase</i> atau <i>usecase</i> memiliki interaksi dengan <i>actor</i> .
4.	<i>Extend</i> 	Perilaku <i>usecase</i> memperluas perilaku <i>usecase</i> yang lain.
5.	Generalisasi 	Hubungan generalisasi dan spesialisasi antara dua buah <i>usecase</i> dimana fungsi yang satu adalah fungsi yang lebih umum dari yang lainnya
6.	<i>Include</i> 	Perilaku <i>usecase</i> merupakan bagian dari <i>usecase</i> yang lain

2.15.2 Activity Diagram

Menurut Wahyuda dan Theresiawati (Wahyuda & Theresiawati, 2022), *Activity Diagram* merupakan setiap aktivitas yang ada dalam sistem, dimana setiap aktivitas dalam sistem mewakili diagram aktivitas, yang menjadi model fungsional yang bisa mengilustrasi proses bisnis yang dapat menggambarkan proses bisnis.

Table 2. 3 Simbol – simbol Activity Diagram

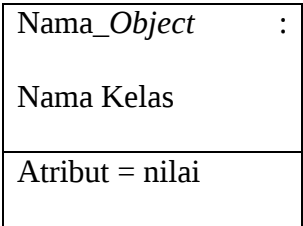
No	Simbol	Keterangan
1.	State 	Menggambarkan <i>state</i> untuk objek dari sistem.
2.	Activity 	Menggambarkan aktivitas dari sistem
3.	Initial State 	Menggambarkan titik awal dari setiap aktivitas.
4.	Final State 	Menggambarkan titik akhir dari Aktivitas
5.	Transition 	Menunjukkan aliran aktivitas dimana data mengalir dari satu aksi ke aksi berikutnya

6.	<i>Decision</i> 	Menggambarkan titik pengambilan Keputusan
7	Penggabungan / <i>Join</i> 	Penggabungan dimana yang mana lebih dari satu aktivitas lalu digabungkan menjadi satu.

2.15.3 Object Diagram

Menurut Mahendra, Susyanto, dan Siswanti (Indardo & Duha, 2023), *Object* diagram berhubungan dengan *Class* diagram karena pembagian dalam *Class* diagram harus digunakan dalam *Object* diagram pada sistem.

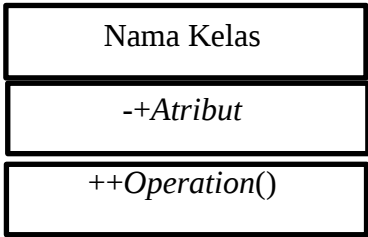
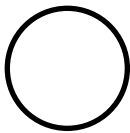
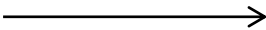
Table 2. 4 Simbol – simbol Objek Diagram

No	Simbol	Keterangan
1.	<i>Object</i> 	Objek dari sistem
2.	Link 	Relasi antar objek

2.15.4 Class Diagram

Menurut Wahyuda dan Theresiawati (Wahyuda & Theresiawati, 2022), *Class Diagram* merupakan setiap objek atau data yang memiliki anggota, baik itu *attributes (field dan properties)*, *operations (methods)*, dan events. melalui *Class diagram* nantinya program design didesain di tahapan perancangan sistem. *Class diagram* merupakan model struktural.

Table 2. 5 Simbol – Simbol *Class Diagram*

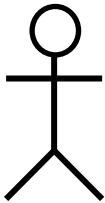
No	Simbol	Keterangan
1.	Kelas 	Kelas pada struktur sistem.
2.	<i>Interface</i>  Nama_Interface	Sama dengan konsep <i>interface</i> programan berorientasi objek.
3.	Asosiasi 	Relasi antar kelas dengan makna g satu digunakan oleh kelas yang
4.	Generalisasi 	Hubungan generalisasi dan i antara dua buah <i>usecase</i> dimana ng satu adalah fungsi yang lebih ri yang lainnya.

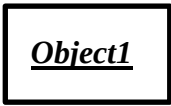
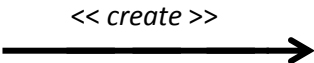
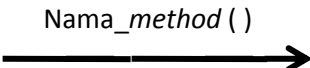
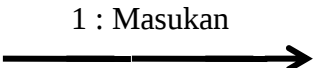
5.	Agregasi 	Relasi antar kelas dengan makna bagian (<i>whole-part</i>).
----	---	---

2.15.5 Sequence Diagram

Menurut Wahyuda dan Theresiawati (Wahyuda & Theresiawati, 2022), *Sequence Diagram* merupakan bagaimana berbagai obyek di sistem pada sistem berinteraksi satu sama lain dan apa yang dapat dilakukan seorang aktor dengan objek tersebut.

Table 2. 6 Simbol – simbol Sequence Diagram

No	Simbol	Keterangan
1.	<i>Actor</i> 	Orang, proses atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat sendiri. Jadi walaupun simbol <i>actor</i> adalah gambar orang, biasanya dinyatakan menggunakan kata benda diawal <i>frase</i> nama <i>actor</i> .
2.	<i>Lifeline</i> 	Menyatakan kehidupan suatu objek.

3.	Objek 	Menyatakan objek yang berinteraksi dengan pesan.
4.	Waktu Aktif 	Menyatakan objek dalam keadaan aktif dan berintegrasi, semua yang berhubungan dengan waktu aktif ini adalah sebuah tahapan yang akan dilakukan di dalamnya.
5.	Pesan tipe <i>create</i> 	Menyatakan suatu objek membuat objek lain, arah panah mengarah pada objek yang dibuat.
6.	Pesan tipe <i>call</i> 	Menyatakan suatu objek memanggil operasi/ metode yang ada pada objek lain atau dirinya sendiri.
7.	Pesan tipe <i>send</i> 	Menyatakan bahwa suatu objek mengirim data / masukan/ informasi ke objek lainnya, arah panah mengarah pada objek yang dikirim.