

BAB II

KAJIAN PUSTAKA

A. Kajian Teoritis

1. Sistem

a. Pengertian Sistem

Sistem merupakan suatu kumpulan dari variable – variable yang saling berinteraksi dan saling berkaitan satu sama lain (Fatta, 2017:3). Sedangkan sistem menurut (Fathoni et al., 2024:3) adalah suatu kesatuan komponen yang saling terkait untuk mencapai tujuan secara harmonis. Sistem penting untuk menciptakan kerja yang efisien dan terkoordinasi. Sedangkan menurut (Achmad Fikri Sallaby & Indra Kanedi, 2020), Kumpulan data atau lebih komponen yang saling berkaitan dan berinteraksi untuk mencapai keputusan tertentu disebut sistem.

Didasarkan pada berbagai pendapat di atas, sistem dapat digambarkan sebagai kumpulan unsur atau komponen yang terorganisir. Komponen-komponen dalam sistem saling terkait, baik secara fisik maupun non-fisik, dan bekerja sama untuk mencapai tujuan tertentu. Sistem juga penting untuk mengatur dan mengoptimalkan kerja.

b. Karakteristik Sistem

Karakteristik sistem yaitu sistem yang mengandung komponen, batas, lingkungan, koneksi, masukan, keluaran, pengolah, dan tujuan sistem (Agustini & Kurniawan, 2019).

Berikut merupakan bagian dari karakter sistem:

1) Komponen Sistem (*Component System*):

Alat, manusia, dan lain-lain adalah beberapa contoh komponen sistem yang bekerja sama atau berinteraksi satu sama lain untuk membentuk satu kesatuan.

2) Batas Sistem (*Boundary System*):

Batas sistem adalah area yang memisahkan satu sistem dari sistem lain atau lingkungannya. Batas ini memungkinkan sistem dilihat sebagai satu unit dan menunjukkan ruang lingkup sistem.

3) Lingkungan Luar Sistem (*Environment System*):

Segala sesuatu di luar batas sistem yang mempengaruhi operasinya disebut lingkungan luar; lingkungan ini dapat berdampak positif atau negatif.

4) Penghubung Sistem (*Interface System*):

Penghubung sistem berfungsi untuk menghubungkan satu subsistem ke subsistem lainnya

- a) Sumber daya mengalir dari satu sistem ke sistem lain: Ini mengacu pada proses di mana sumber daya, baik itu energi,

informasi, atau materi, mengalir dari satu sistem ke sistem lain untuk mendukung operasi atau proses yang lebih besar.

- b) *Output* dari satu subsistem menjadi input bagi subsistem lain: Ini menunjukkan hubungan antara subsistem di mana output dari satu subsistem digunakan sebagai input untuk subsistem lain dalam rangkaian proses yang lebih besar atau sistem yang kompleks.
- c) Subsistem berintegrasi untuk membentuk satu kesatuan: Ini menggambarkan bagaimana subsistem-subsistem yang berbeda bekerja bersama atau terhubung secara sinergis untuk membentuk sebuah sistem atau kesatuan yang lebih besar dengan tujuan tertentu.

5) Masukan Sistem (*Input System*):

Masukan adalah energi yang dimasukkan ke sistem; ini dapat berupa input perawatan atau signal input.:

- a) *Maintenance input*: Ini adalah energi atau input yang diperlukan untuk menjaga atau memelihara operasi sistem, seperti program komputer atau perangkat lunak yang diperlukan agar sistem tetap berfungsi.
- b) *Signal input* adalah energi atau *input* yang diproses oleh sistem dan dikonversi menjadi *output*, seperti data atau informasi yang dimasukkan ke dalam sistem untuk diproses atau dianalisis.

6) Keluaran Sistem (*Output System*):

Output sistem dapat digunakan sebagai input untuk subsistem atau sistem yang lebih besar. Energi yang telah diproses dan diubah menjadi output yang bermanfaat, seperti informasi, dan sisa pembuangan, seperti panas komputer.

7) Pengolahan Sistem (*Process System*):

Melalui proses transformasi, pengolahan sistem mengubah *input* menjadi *output*. Itu juga mempertimbangkan batas-batas sistem atau subsistem serta pengaruh lingkungan, yang dapat positif atau negatif.

8) Sasaran Sistem (*Objective System*):

Setiap sistem memiliki tujuan yang menentukan *input* dan *output*.

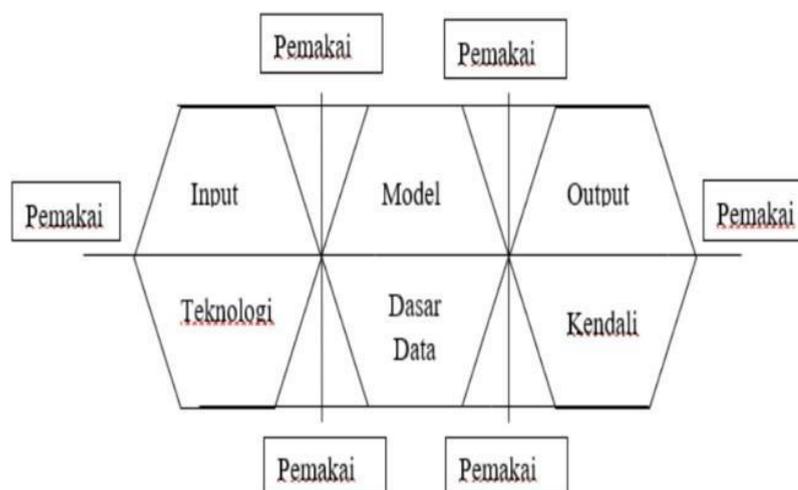
2. Informasi

Menurut (Hasan & Muhammad, 2020), Informasi adalah data yang telah diubah menjadi informasi yang berguna bagi mereka yang membutuhkannya dan berguna untuk pengambilan keputusan di masa mendatang. Sedangkan pengertian informasi menurut (Abdur Rochman, 2019) adalah salah satu pendukung untuk menentukan sikap dan merumuskan konsep, gagasan, serta mengambil keputusan. Informasi juga berfungsi sebagai alat penting dalam menentukan sikap serta dalam merumuskan konsep, gagasan, dan keputusan. Dengan demikian, informasi

didefinisikan sebagai data yang telah diproses sehingga menjadi bermakna dan berguna bagi mereka yang membuat keputusan.

3. Sistem Informasi

Menurut (Setiyanto et al., 2019) Sistem informasi adalah kumpulan subsistem yang terintegrasi yang bekerja sama untuk memecahkan masalah melalui proses data berbasis komputer untuk memberikan nilai lebih dan keuntungan bagi pengguna. Sedangkan menurut (Fathoni et al., 2024:6) sistem informasi merupakan proses pengumpulan data, penyimpanan, dan analisisnya sehingga menghasilkan informasi untuk mencapai tujuan. Berikut merupakan konsep sistem informasi menurut (Prehanto, 2020:21):



Gambar 2. 1 Konsep Sistem Informasi

Sumber: (Prehanto, 2020:21)

c. Blok *Input*

Blok *input* mencakup data masukan, seperti dokumen-dokumen dasar, yang dikirim ke sistem informasi.

d. Blok Model

Blok model berisi prosedur, logis, dan metode matematika yang disimpan di basis data dan diubah oleh pengguna sistem dan semua tingkat manajemen.

e. Blok Keluaran

Sistem informasi keluaran berkualitas tinggi dan bermanfaat bagi semua pemakai sistem dan tingkatan manajemen.

f. Blok Teknologi

Blok teknologi adalah model pengolahan input yang digunakan untuk menyimpan dan mengakses data. Software, manusiaware/brainware hardware (perangkat lunak, perangkat keras, dan teknisi), dan produk keluaran yang membantu mengendalikan diri termasuk dalam blok teknologi.

g. Blok Basis Data

Basis Data adalah kumpulan data yang memiliki hubungan satu sama lain. Memori komputer dan perangkat lunak berfungsi sebagai media penyimpanan basis data.

h. Blok Kendali

Bencana alam, kecurangan, sabotase, dan hoax informasi adalah contoh kesalahan yang dapat dicegah dan diatasi dengan pengendalian yang berguna.

4. Balen Coffee

Balen Coffee berdiri sejak Februari 2019 pertama kali memulai usahanya melalui *Car Free Day* kota Madiun yang pada saat itu berada di Jalan Pahlawan. Bermodalkan sebuah meja dengan alat kopi seadanya, Balen Coffee berusaha menjual produk Kopi Susu kekinian yang pada saat itu belum terlalu dikenal oleh masyarakat Madiun. Menggunakan teknik seduhan Cold Brew, terciptalah sebuah Es Kopi Balen. Es Kopi Balen merupakan kombinasi kopi susu dengan sirup gula aren yang dibuat sendiri menggunakan perpaduan kayu manis sebagai penambah cita rasa dan aroma. Didampingi dengan varian es kopi susu menarik lainnya membuat daya tarik masyarakat akan produk Balen Coffee semakin meningkat. Hal ini dibuktikan dengan terjualnya 30 hingga 40 cup es kopi susu dalam 3 jam waktu berjualan selama *Car Free Day*. Antusias yang tinggi, serta respon positif dari masyarakat membuat Balen Coffee termotivasi untuk mengembangkan usahanya lebih besar lagi. Namun karena keterbatasan modal dan juga kondisi dimana *Car Free Day* berubah sistem baru menjadi *Sunday Market* membuat Balen Coffee harus vakum untuk beberapa bulan.

Pada Mei 2020, Balen Coffee kembali hadir dengan konsep baru dan juga varian-varian baru yang lebih menarik. Memanfaatkan kemajuan teknologi, Balen Coffee menjual produknya melalui media sosial secara *online*. Dengan sistem *delivery order*, Balen Coffee melakukan pesan antar secara mandiri kepada para *customer* nya. Menyajikan varian botol 500 ml dan 1 liter, konsumen Balen Coffee dapat dengan mudah menikmati Es Kopi Balen secara beramai-ramai bersama kerabat dan juga keluarga. Keinginan untuk mengembangkan Balen Coffee untuk memiliki sebuah kedai pada akhirnya dapat terealisasi pada Agustus 2020 setelah modal yang didapatkan cukup. Harapan kedepannya, Balen Coffee dapat terus berkembang sehingga dapat memajukan bisnis perkopian di Madiun.

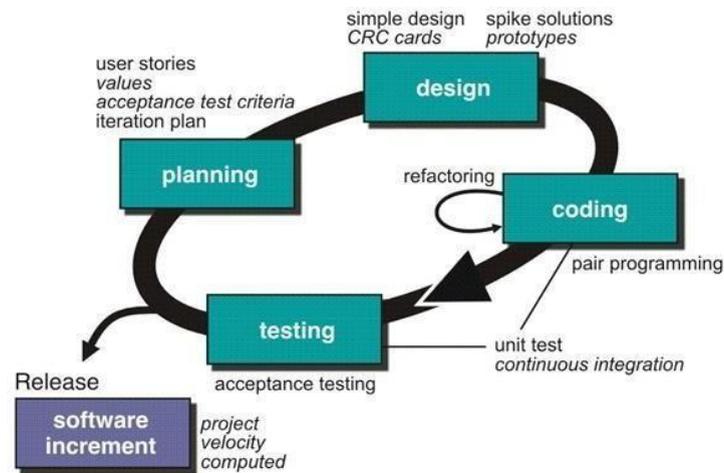
5. *Extreme Programming*

Extreme Programming (XP) adalah metode pengembangan perangkat lunak yang bertujuan untuk meningkatkan kualitas perangkat lunak dan meningkatkan respons terhadap perubahan dan kebutuhan pelanggan. Metode ini memungkinkan pengembangan secara fleksibel menyesuaikan diri dengan perubahan yang terjadi selama siklus pengembangan, sehingga XP membantu tim pengembangan menghasilkan perangkat lunak yang lebih efisien sesuai.

Menurut (Sri Ramadhani et al., 2019) *Extreme Programming* adalah salah satu model proses *Agile Software Development*, yang

merupakan salah satu metodologi pengembangan sistem berbasis *Software Development Life Cycle* (SDLC). SDLC memberikan struktur dan tahapan umum yang diperlukan untuk pengembangan perangkat lunak. Sementara XP menyediakan praktik dan pendekatan spesifik yang dapat digunakan untuk mengimplementasi setiap tahapan SDLC dengan lebih efektif. Dengan demikian, SDLC dan *Extreme Programming* dapat saling melengkapi. SDLC menyediakan kerangka kerja umum untuk pengembangan perangkat lunak, sedangkan XP menyediakan praktik yang dapat diadopsi dalam setiap tahapan SDLC untuk mencapai tujuan pengembangan yang diinginkan.

Berdasarkan definisi di atas, *Extreme Programming* adalah teknik pengembangan perangkat lunak yang bertujuan untuk meningkatkan kualitas dan kemampuan untuk menyesuaikan diri dengan perubahan sesuai dengan kebutuhan pelanggan. Metode ini memungkinkan pengembangan fleksibel dan efisien menyesuaikan diri dengan perubahan selama siklus pengembangan. XP melengkapi *Software Development Life Cycle* (SDLC) dengan menyediakan praktik spesifik untuk setiap tahap, sehingga keduanya dapat digunakan bersama untuk mencapai pengembangan perangkat lunak yang optimal. *Planning, design, coding, dan testing* adalah beberapa langkah dalam metode *extreme programming*.



Gambar 2. 2 Metode *extreme programming*

Sumber: (Sri Ramadhani et al., 2019)

6. *Unified Modeling Language (UML)*

Menurut (Booch, 2013) *Unified Modeling Language (UML)* adalah bahasa pemodelan visual tujuan umum yang digunakan untuk menentukan, memvisualisasikan, membangun, dan mendokumentasikan artefak sistem perangkat lunak. UML menunjukkan struktur, perilaku, dan interaksi sistem. UML terdiri dari sebelas jenis diagram, yang dibagi menjadi tiga kategori. Berikut ini adalah penjelasan singkat tentang pembagian kategori:

- a) Struktur diagram adalah kumpulan diagram yang menunjukkan struktur statis sistem yang dimodelkan. diagram kelas, objek, komponen, struktur komposit, paket, dan *deployment* termasuk dalam diagram ini.

- b) Diagram perilaku adalah kumpulan diagram yang digunakan untuk menunjukkan perilaku sistem atau berbagai perubahan yang terjadi dalam sistem. diagram kasus, diagram aktif, dan diagram mesin statis termasuk dalam diagram ini.
- c) *Sequence diagram* adalah salah satu dari banyak diagram interaksi yang digunakan untuk menunjukkan interaksi antara subsistem dan satu sistem dengan sistem lainnya.

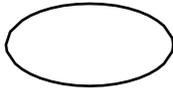
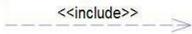
Berikut merupakan jenis-jenis diagram UML antarlain:

a) *Use Case Diagram*

Menurut (Triandini & Suardika, 2012:17) *use case* merupakan sebuah aktivitas yang dilakukan sistem dan hasil dari tanggapan perintah yang berasal dari pengguna. Diagram ini membantu menjelaskan fungsionalitas sistem dari perspektif pengguna, menunjukkan bagaimana pengguna berinteraksi dengan berbagai kasus penggunaan sistem. Sementara itu, skenario atau urutantindakan yang dilakukan oleh aktor (pengguna atau sistem lain) dalam interaksi dengan sistem yang sedang dikembangkan disebut sebagai *use case*. Interaksi antara fungsionalitas atau layanan yang harus diberikan sistem kepada aktor tertentu serta bagaimana aktor tersebut berinteraksi menggunakan fungsionalitas tersebut dalam berbagai situasidikenal sebagai *use case* (Saputra & Fahrizal, 2021). Nama,

symbol beserta keterangan dari *use case diagram* dapat dilihat pada tabel dibawah:

Tabel 2. 1 *Use Case Diagram*

No	Nama / Simbol	Keterangan
1	Aktor 	Aktor akan diberikan nama untuk merepresentasikan peran yang akan dimainkan. Aktor merupakan orang yang menggunakan sistem.
2	<i>Use Case</i> 	<i>Use case</i> digambarkan dengan symbol oval yang berisi nama <i>use case</i> di dalamnya. <i>Use case</i> juga mencerminkan aktivitas yang dilakukan oleh suatu sistem.
3	<i>Association</i> 	Komunikasi antara seorang actor dengan <i>use case</i> yang terkait dengan <i>use case</i> tersebut melibatkan interaksi dengan actor tersebut.
4	<i>Generlisasi</i> 	Hubungan <i>generalisasi</i> dan <i>specialisasi</i> antara dua buah <i>use case</i> yang penggunaan suatu fitur lebih umum dibandingkan yang lain.
5	<i>Include</i> 	Menunjukkan bahwa suatu <i>use case</i> seluruhnya

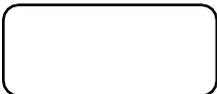
		merupakan fungsionalitas dari <i>use case</i> lainnya.
6	<i>Extend</i>	Menunjukkan bahwa suatu <i>use case</i> merupakan tambahan fungsional dari <i>use case</i> yang dapat digunakan secara mandiri.

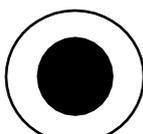


b) *Activity Diagram*

Salah satu jenis diagram dalam *Unified Modeling Language* (UML) yang digunakan untuk memodelkan alur kerja atau aktivitas dalam sebuah sistem adalah aktivitas diagram. Diagram ini membantu menggambarkan urutan aktivitas, kondisi keputusan, dan aliran kontrol antar aktivitas dalam suatu proses. *Activity Diagram* sangat berguna dalam menganalisis dan merancang proses bisnis serta alur kerja dalam sistem perangkat lunak (Saputra & Fahrizal, 2021). Komponen *Activity Diagram* dapat dilihat pada tabel dibawah:

Tabel 2. 2 *Activity Diagram*

No	Nama/Symbol	Keterangan
1	Titik Awal 	Titik awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah titik awal.
2	Aktivitas 	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.

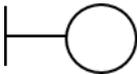
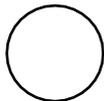
3	Percabangan		Asosiasi percabangan jika ada pilihan aktivitas lebih dari satu
4	Penggabungan		Asosiasi penggabungan jika lebih dari satu aktivitas digabungkan menjadi satu.
5	Status akhir		Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir .

c) *Sequence Diagram*

Diagram ini adalah gabungan dari *class diagram* dengan objek dengan gambaran model statis dan dinamis (Saputra & Fahrizal, 2021). Diagram yang diatur secara temporal ini menunjukkan proses operasi, jenis pesan yang dikirim, dan waktu pelaksanaan. Objektif yang berhubungan dengan proses operasional diurutkan dari kiri ke kanan. Tabel berikut menunjukkan simbol dan keterangan *Sequence diagram*:

Tabel 2. 3 *Sequence diagram*

No	Nama/Simbol	Keterangan
1	Aktor 	Aktor adalah individu yang berinteraksi dengan sistem, memberikan <i>input</i> kepada sistem atau menerima <i>output</i> dari sistem.

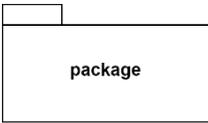
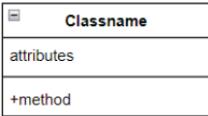
No	Nama/Symbol	Keterangan
2	<i>Boundary object/ Interface</i> 	Sekumpulan kelas yang mewakili interaksi antara satu atau lebih actor dalam suatu sistem.
3	<i>Entity object</i> 	Sekumpulan kelas yang terdiri dari kumpulan entitas untuk membentuk gambaran awal sistem dan dapat menjadi dasar dalam penyusunan <i>database</i> .
4	<i>Control object</i> 	Mengkoordinasikan perilaku sistem dan dinamika suatu sistem, menangani tugas utama dan mengontrol alur kerja suatu sistem.
5	<i>Message</i> 	Menggambarkan pesan atau hubungan antar objek yang menunjukkan urutan kejadian yang terjadi.
6	<i>Message to self</i> 	Menggambarkan pesan atau hubungan objek itu sendiri.
5	<i>Lifeline</i> 	Garis titik-titik yang berhubungan dengan objek dan menunjukkan urutan pesan yang dikirim dan diterima oleh actor dan objek. Urutan pesan dibaca secara

No	Nama/Simbol	Keterangan
		berurutan dari atas hingga ke bawah diagram.

d) *Class Diagram*

Salah satu jenis diagram dalam *Unified Modeling Language* (UML) yang digunakan untuk memodelkan struktur statis sebuah sistem adalah *Class Diagram*. Menurut (Bergström et al., 2022) *class diagram* merupakan bantuan penting untuk memahami sistem perangkat lunak. *Class Diagram* menunjukkan kelas-kelas dalam sistem, atribut dan metode mereka, serta hubungan mereka satu sama lain. Komponen-komponen dalam *class diagram* dapat dilihat pada tabel di bawah ini:

Tabel 2. 4 Komponen-komponen *class diagram*

No	Nama/simbol	Keterangan
1	<i>Package</i> 	Suatu bungkusan paket yang terdiri dari satu atau lebih <i>class</i> .
2	<i>Class</i> 	<i>Class</i> merupakan kategori atau klasifikasi yang gunanya untuk menjelaskan sekumpulan suatu objek, yang digambarkan dalam bentuk persegi panjang, memiliki dua atau tiga bagian yaitu nama

No	Nama/simbol	Keterangan
		<i>class</i> , atribut, dan metode yang dimiliki <i>class</i> tersebut.
3	<i>Interface</i> 	<i>Interface</i> berperan sebagai antarmuka komponen agar tidak langsung mengakses komponen.
4	Asosiasi 	Asosiasi merupakan garis penghubung antar <i>class</i> . Pada garis tersebut juga terdapat <i>multiplicity</i> yang mempresentasikan jumlah maksimum dan minimum dari suatu <i>class</i> dengan <i>class</i> lainnya.
5	<i>Generalization</i> 	<i>Generalization</i> merupakan relasi yang menunjukkan hubungan antara <i>subclass</i> dengan <i>superclass</i> , dimana <i>subclass</i> merupakan bagian yang diturunkan dari <i>superclass</i> . <i>Superclass</i> merupakan <i>class</i> umum dibandingkan dengan <i>subclass</i> .
6	<i>Dependency</i> 	<i>Dependency</i> merupakan relasi yang menunjukkan hubungan yang maknanya bergantung diantara <i>class</i> .
7	<i>Aggregation</i> 	<i>Aggregation</i> merupakan jenis <i>whole-part relationship</i> dimana bagian komponen dapat berdiri sendiri walaupun komponen agregatnya tidak ada. <i>Aggregation</i>

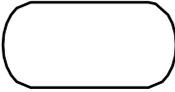
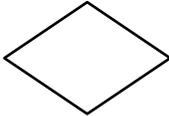
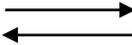
No	Nama/symbol	Keterangan
		dilambangkan dengan bentuk <i>diamond</i> kosong.
8	<i>Composite</i> 	<i>Composite</i> merupakan jenis <i>whole-part relationship</i> dimana bagian komponen tidak dapat berdiri sendiri jika telah diasosiasikan. <i>Composite</i> dilambangkan dengan bentuk <i>diamond</i> berwarna hitam.
9	Asosiasi berarah 	Asosiasi berarah mengacu kepada kemampuan suatu objek untuk melihat dan berkomunikasi dengan objek lain.

7. Flowchart

Flowchart merupakan diagram yang menunjukkan urutan atau langkah-langkah dalam sebuah program atau prosedur sistem secara logis (Yulianeu & Oktamala, 2022). Opini lain juga diungkapkan oleh (Nita & Kartikawati, 2020) *flowchart* merupakan bagan yang menggambarkan urutan aktivitas atau langkah program dari awal hingga akhir dengan menggunakan symbol atau gambar yang mewakili fungsi langkah program dan garis aliran yang menunjukan urutan symbol yang akan dikerjakan. Umumnya *flowchart* sering dimanfaatkan untuk memvisualisasikan struktur kepengurusan dalam suatu entitas organisasi atau perusahaan. Selain itu, *flowchart* juga biasa digunakan dalam lingkup bisnis untuk menyusun strategi pemasaran atau perencanaan produksi. Dalam hal ini, setiap

flowchart selalu memiliki awal dan akhir yang membantu pembaca dalam memahami jalannya informasi yang disajikan. Simbol-simbol yang digunakan untuk menyajikan kegiatan manual dan kegiatan pemrosesan dapat dilihat pada tabel dibawah:

Tabel 2. 5 *Flowchart*

No	Nama/Simbol	Keterangan
1	<i>Terminator</i> 	Menunjukkan suatu permulaan (<i>start</i>) atau akhir (<i>stop</i>) dalam suatu proses kegiatan.
2	<i>Input/Output</i> 	Simbol yang digunakan untuk proses <i>input</i> atau <i>output</i> data.
3	<i>Process</i> 	Menunjukkan pengolahan yang dilakukan oleh computer. Contohnya pada bidang industry khususnya proses produksi barang
4	<i>Decision</i> 	Memilih proses berdasarkan kondisi yang ada. Simbol ini digunakan untuk aliran proses suatu algoritma.
5	<i>Document</i> 	Menyatakan <i>input</i> yang berasal dari dokumen dalam bentuk kertas atau <i>output</i> dicetak ke kertas.
6	<i>Flow direction</i> 	Menghubungkan diantara symbol, symbol yang satu dengan yang lainnya.

Flowchart dibedakan menjadi 5 jenis, antara lain:

a) *System Flowchart*

Gambaran aliran sistem adalah representasi visual dari rangkaian kerja sistem. Diagram ini menunjukkan urutan langkah-langkah dari prosedur sistem tersebut..

b) *Document Flowchart*

Bagan alir dokumen, juga disebut bagan alir formulir atau bagan alir kertas, adalah diagram yang menunjukkan aliran laporan dan formulir dengan salinannya.

c) *Schematic Flowchart*

Bagan alir sistem, atau bagan alir skematik, menggambarkan prosedur dalam sistem. Yang membedakan mereka dari yang lain adalah bahwa mereka menggunakan gambar komputer dan peralatan yang digunakan.

d) *Program Flowchart*

Flowchart program adalah representasi grafis dari langkah-langkah proses program. Diagram alur sistem adalah dasar untuk diagram alur program. Dua jenis diagram alur program adalah diagram alur logika program dan diagram alur program komputer terinci. Diagram alur logika program menunjukkan langkah-langkah logis yang ada dalam program komputer dan telah disiapkan oleh analis sistem. Diagram alur program komputer terinci menunjukkan

intruksi-intruksi program yang lebih rinci dan telah disiapkan oleh pemrogram.

e) *Process Flowchart*

Dalam teknik industri, bagan alir proses juga disebut bagan alir proses sangat digunakan. Analisis sistem juga dapat menggunakan bagan alir ini untuk menunjukkan proses dalam suatu prosedur.

8. *Website*

Website yang terdiri dari berbagai *hyperlink* yang dihubungkan ke beberapa halaman yang memuat informasi dalam bentuk teks, gambar, video, suara, animasi, atau kombinasi dari semua ini (NovriaRahma et al., 2022). *Website* terdiri dari halaman *web* yang terhubung satu sama lain dan dilengkapi dengan domain yang dikenal sebagai URL atau *World Wide Web*. Sedangkan menurut (Surentu et al., 2020) Dengan jangkauan waktu dan ruang yang tak terbatas, web adalah *platform* promosi yang terkenal saat ini. Pendapat lain juga diungkapkan oleh (Yao et al., 2022) *World Wide Web* adalah lingkungan interaktif *open-domain* besar yang secara inheren memenuhi persyaratan melalui kumpulan halaman yang saling berhubungan dengan teks, gambar, dan elemen interaktif alami. Dari pengertian di atas, dapat ditarik kesimpulan yaitu *website* merupakan media yang menghubungkan banyak halaman yang menyajikan informasi dalam berbagai bentuk dan merupakan salah satu media promosi terpopuler karena

memiliki jangkauan yang luas tanpa batasan waktu dan ruang.

9. PHP

Menurut (Murni et al., 2023) PHP adalah Bahasa pemrograman yang dikenal sebagai bahasa *scripting*, yang berarti bahwa PHP dapat disematkan dalam Bahasa dan aplikasi lain. Sedangkan menurut (Sitanggang Rianto et al., 2022) PHP adalah skrip yang terintegrasi dengan HTML dan dijalankan di *server*. Pendapat lain juga diungkapkan oleh (Chavan & Pawar, 2021) PHP merupakan Bahasa skrip yang terkenal dan sering dikaitkan dengan *web*. Dari pemahaman di atas, PHP adalah bahasa pemrograman *scripting* yang dapat dimasukkan ke dalam bahasa dan aplikasi lain serta dijalankan di *server*.

10. Database

Menurut (Alasi et al., 2020) *database* adalah sekumpulan data yang disusun, disimpan, dan dapat diakses secara elektronik melalui sistem komputer. *Database* sangat kompleks, bahkan semakin hari semakin dikembangkan. Semakin banyak data pada *database* maka semakin banyak informasi yang harus diamankan. Sedangkan menurut (Ramadhan & Mukhaiyar, 2020) Kumpulan data yang disusun dan disimpan secara elektronik, biasanya dalam bentuk tabel yang saling berhubungan, disebut *database*. *Database* digunakan untuk mengelola, mengakses, dan memanipulasi data secara efisien. Fungsi utama dari

database mencakup penyimpanan, pengelolaan, dan pengambilan data. *Database* memungkinkan pengguna untuk melakukan berbagai operasi seperti menambah, mengubah, dan menghapus data, serta melakukan pencarian dan analisis data. *Database* sering dikelola menggunakan Sistem Manajemen Basis Data (DBMS) yang menyediakan alat dan antarmuka untuk berinteraksi dengan data secara efektif dan aman. Beberapa fungsi dari *database* menurut (Ramadhan & Mukhaiyar, 2020) antara lain:

- a) Mempermudah identifikasi data melalui pengelompokan data, seperti dengan pembuatan beberapa *field* yang berbeda-beda.
- b) Meminimalisir adanya data ganda.
- c) Mempermudah penggunaan oleh *user* dalam berbagai hal, misalnya penginputan data baru.
- d) Penyimpanan data secara digital.
- e) Menjadi alternative lain terkait masalah penyimpanan ruang dalam suatu aplikasi.

Adapun beberapa jenis *database* menurut (Ramadhan & Mukhaiyar, 2020) antara lain:

a) *Operational Database*

Agar mudah digunakan, *database* jenis ini menyimpan data yang sangat rumit. *Database* pelanggan biasanya menggunakan *database* ini.

b) *Relational Database*

Pengguna dapat mengakses atau mencari informasi dalam berbagai *table* dalam *database* ini.

c) *Distributed Database*

Jenis *database* ini memungkinkan data didistribusikan secara tersebar namun saling berhubungan dan dapat diakses bersama.

d) *External Database*

Karena kemudahan akses publiknya, *database* terakhir ini sering digunakan sebagai kebutuhan komersial.

B. Kajian Empiris

Penelitian Rancang Bangun Sistem Pemesanan Menu Pada Balen Coffee Kota Madiun Menggunakan Metode *Extreme Programming*, terdapat jurnal ilmiah yang membahas topik perancangan sistem pemesanan menu berbasis *web app*. Untuk menjelaskan topik sistem pemesanan menu berbasis *web app*, diperlukan beberapa penelitian dari berbagai sumber.

Pada penelitian yang dilakukan oleh (Hasim et al., 2022) dengan judul “Implementasi Progressive *Web App* Menggunakan *Codelgniter* pada Sistem Penjualan Toko Aneka Snack” menyatakan bahwa peneliti tersebut menggunakan metode *Extreme Programming*. Batasan dalam sistem tersebut yaitu menu *home*, halaman tentang kami, halaman produk, halaman *login* admin dan *customer* dan menu pembayaran. Dengan pengujian sistem menggunakan *Lighthouse* dan *Blackbox*.

Penelitian lain dilakukan oleh (Dzikria & Rizal, 2023) dengan judul "Rancang Bangun Sistem Pemesanan Mandiri Restoran Berbasis *Progressive Web Apps*" menghasilkan sebuah sistem pemesanan dengan memanfaatkan teknologi *QR code* dengan tujuan efisiensi dan kepraktisan pemesanan di restoran. Peneliti menggunakan metode *Waterfall*. Batasan dalam sistem tersebut yaitu halaman *scan QR code* untuk masuk aplikasi, halaman menu makanan dan halaman pembayaran. Pengujian sistem ini menggunakan *Blackbox*.

Penelitian lain dilakukan oleh (Simangunsong et al., 2023) dengan judul "Rancang Bangun Sistem Informasi *Online Marketplace* Berbasis *Web Application* (Studi Kasus: *KLG Campus Residence*) menghasilkan sebuah sistem yang bertujuan untuk mempermudah proses jual-beli yang dilakukan. Batasan dalam sistem tersebut antarlain halaman *login*, halaman *register*, halaman *add item*, halaman *store*, halaman *item detail*, halaman *review*, *order form* dan halaman *order detail*. Pengujian sistem ini menggunakan *Blackbox*.

C. Kerangka Berpikir

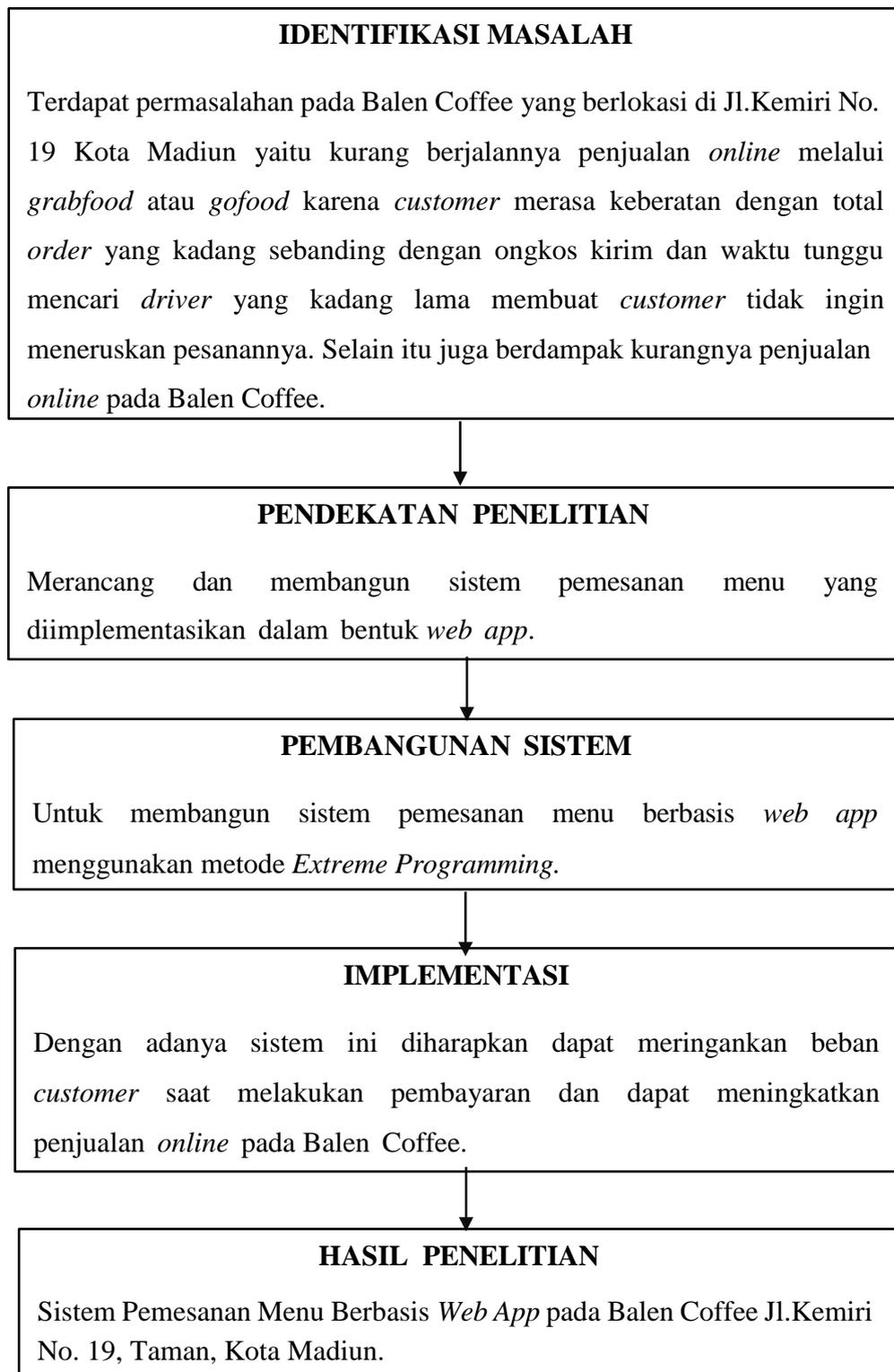
Balen Coffee Kota Madiun merupakan tempat berjualan makanan dan minuman berkonsep *cafe*. Sistem penjualan yang digunakan saat ini yaitu datang langsung ke tempat (*dine-in*) dan menggunakan aplikasi *online* seperti *grabfood* dan *gofood*. Namun beberapa *customer* memilih untuk memesan melalui *whatsapp* lalu melakukan pembayaran di kasir atau melalui transfer kemudian mengambil pesannya jika sudah selesai. Saat ini mereka merasa sedikit terbebani dengan biaya ongkos kirim pesanan karena jika melalui *grabfood* atau *gofood* kadang tidak sepadan dengan total pembelian. Selain itu, lamanya waktu tunggu mencari *driver* yang kadang terlalu lama membuat pengguna kadang tidak ingin meneruskan pesannya. Hal itu juga berdampak pada kurang berjalannya penjualan *online* pada Balen Coffee.

Dalam penelitian ini, model pengembangan sistem yang digunakan adalah metode *Extreme Programming*, salah satu cabang dari metode *Agile*. Metode ini memungkinkan pengembangan untuk menyesuaikan diri dengan perubahan yang terjadi selama siklus pengembangan. *Planing, design, coding, dan testing* adalah komponen metode *Extreme Programming*.

Sistem pemesanan menu pada Balen Coffee berbasis *web app* dibangun menggunakan bahasa pemrograman PHP dan *MySQL* untuk *database*. Uji kelayakan kualitas pada sistem dilakukan oleh peneliti menggunakan pengujian *blackbox*. Sistem pemesanan menu yang akan

diimplementasikan pada Balen Coffee *customer* dapat melihat dan memilih menu yang akan dibeli, kemudian jika *customer* memenuhi total minimal pembelian akan mendapatkan *voucher* diskon dalam bentuk koin untuk meningkatkan loyalitas pelanggan, *customer* juga dapat mengetahui estimasi lamanya waktu tunggu proses pemesanan.

Hasil pada penelitian ini adalah Sistem Pemesanan Menu Berbasis *Web App* Menggunakan Metode *Extreme Programming* pada Balen Coffee Kota Madiun. Dalam sistem ini sistem pemesanan menu yang dibangun dapat membantu dalam pemasaran dan pengelolaan pemesanan. Berdasarkan uraian tersebut dapat dibuat sebuah kerangka berpikir yang telah disusun oleh peneliti yang dijelaskan dalam gambar 2.2 sebagai berikut:



Gambar 2. 3 Kerangka Berpikir