

BAB II LANDASAN TEORI

2.1 Tinjauan Pustaka

Dari uraian yang telah dijelaskan secara terperinci, penting untuk dilakukan tinjauan pustaka yang dapat memberikan dukungan terhadap penelitian yang akan dilakukan sebagai berikut.

Tabel 2. 1 Daftar *Literatur*

No Literatur	Penulis dan Tahun	Judul
Literatur 01	(Suwandi, Wicaksono, Dan Ramadani, 2019)	Sistem Pencatatan Persediaan Bahan Baku Dengan Metode Rata-Rata Bergerak.
Literatur 02	(Firlisia, Amroni, Dan Asfi, 2019)	Aplikasi Sistem Perhitungan Harga Pokok Produksi Atas Pemakaian Bahan Baku Menggunakan Metode <i>Moving Average</i> .
Literatur 03	(Saefudin, Hakim, Dan Soegijianto, 2021)	Penggunaan Metode <i>Moving Average</i> Pada Pengembangan Sistem Informasi Basis <i>Web</i> Studi Kasus Perusahaan Onderdil Kendaraan Bermotor.
Literatur 04	(Fauzi, Indriyani, Dan Yanto, 2020)	Implementasi Sistem Informasi <i>Inventory</i> Berbasis WEB (Studi Kasus: CV. Sinar Abadi Cemerlang).
Literatur 05	(Usdeldi Dan Damanik, 2020)	Perancangan Sistem Informasi Persediaan Barang Berbasis WEB di PT. Zokkas Sejahtera Jambi.

a. Literatur 01

Permasalahan yang terjadi pada sistem persediaan bahan baku di CV. Sinar Nugraha. Pertama, masih menggunakan pencatatan manual dengan buku yang seringkali menyebabkan kesalahan dalam pencatatan. Kedua, pencatatan mutasi persediaan juga masih dilakukan secara tertulis pada buku yang menyebabkan manajemen sulit mendapatkan informasi persediaan secara cepat. Ketiga, belum adanya dokumen yang sesuai dengan standar akuntansi pada

beberapa bagian terkait persediaan bahan baku. Keempat, fluktuasi harga bahan baku membuat sulit menentukan harga pokok bahan baku yang dapat berpengaruh pada harga penjualan produk. Terakhir, perlu merancang dan membuat aplikasi komputer untuk menerapkan sistem akuntansi pencatatan persediaan bahan baku dengan metode rata-rata bergerak pada CV. Sinar Nugraha.

Solusi untuk mengatasi masalah tersebut, penelitian dilakukan dengan membuat sistem terkomputerisasi untuk mencatat mutasi persediaan bahan baku, menyajikan data persediaan bahan baku secara update untuk kebutuhan manajemen, merancang dan membuat dokumen resmi untuk bagian yang berhubungan dengan persediaan bahan baku, menghitung harga pokok persediaan bahan baku dengan metode perhitungan rata-rata bergerak, serta merancang dan membuat aplikasi komputer untuk menerapkan sistem akuntansi pencatatan persediaan bahan baku dengan metode rata-rata bergerak.

Namun, kelemahan dari solusi ini adalah sistem tidak memberikan laporan arus masuk dan keluar persediaan, tidak adanya *fitur* konfigurasi hak akses pengguna, tidak adanya tampilan *form* pengolahan data pengguna, sistem belum dilakukan pengujian sistem sehingga belum teruji kelayakan sistemnya, perancangan sistem yang dibangun masih dilakukan dengan *structural design* sehingga akan menyebabkan penulisan fungsi code pada sistem menjadi berulang-ulang.

b. Literatur 02

Permasalahan yang terjadi pada pengelolaan persediaan bahan baku di Kedai Putu. Salah satu masalah yang dihadapi adalah kurangnya sistem yang mengatur keluar dan masuknya bahan baku, sehingga menyebabkan persediaan bahan baku tidak terkendali dan harga jual tidak sesuai dengan standar akuntansi. Masalah lainnya adalah ketidaktahuan dalam memperhitungkan jumlah bahan baku yang tersedia atau digunakan, yang dapat menyebabkan bahan baku tidak terpakai atau terbuang sia-sia.

Solusi untuk mengatasi masalah tersebut, penelitian dilakukan dengan menggunakan metode moving average untuk menghitung harga pokok produksi atas pemakaian bahan baku. Aplikasi tersebut diharapkan dapat mempermudah

manajemen dalam menghitung persediaan bahan baku dan meningkatkan keamanan dalam pemakaian dan pembelian bahan baku.

Namun, kelemahan dari solusi ini adalah, sistem tidak memberikan pengolahan data kategori bahan baku, tidak adanya *fitur* konfigurasi hak akses pengguna, tidak adanya tampilan *form* pengolahan data pengguna, sistem belum dilakukan pengujian sistem sehingga belum teruji kelayakan sistemnya, perancangan sistem yang dibangun masih dilakukan dengan *structural design* sehingga akan menyebabkan penulisan fungsi code pada sistem menjadi berulang-ulang.

c. Literatur 03

Permasalahan yang terjadi pada perusahaan Onderdil Kendaraan Bermotor dalam hal pengendalian stok barang di gudang. Peneliti merancang sistem informasi untuk proses pembelian, perkiraan stok, dan manajemen data barang dengan menyimpan data bisnis dalam database komputer dan dapat diolah melalui antarmuka yang mudah digunakan.

Solusi untuk mengatasi masalah tersebut, penelitian dilakukan dengan membuat sistem untuk membantu menghindari penumpukan atau kekurangan stok barang di gudang yang dapat menghambat proses pendistribusian dan menyebabkan kerusakan barang karena barang mengendap dalam waktu yang lama. Diperlukan teknik perkiraan untuk melakukan pembelian baru dengan memperhatikan transaksi penjualan pada waktu sebelumnya untuk memperkirakan penjualan pada waktu berikutnya.

Namun, kelemahan dari solusi ini adalah sistem tidak memberikan laporan arus masuk dan keluar persediaan, tidak adanya tampilan *form* pengolahan data pengguna, sistem belum dilakukan pengujian sistem sehingga belum teruji kelayakan sistemnya, tidak adanya *fitur* pengolahan data *supplier* barang, tidak adanya cetak kartu stok barang.

d. Literatur 04

Permasalahan terdapat pada perusahaan CV. Sinar Abadi Cemerlang yang memiliki permasalahan dalam proses pencatatan stok persediaan barang dan pembuatan laporan yang masih bersifat manual.

Solusi untuk mengatasi masalah tersebut, penelitian dilakukan dengan membuat rancangan sistem mengenai inventory berbasis website menggunakan metode waterfall yang dapat menyimpan data persediaan barang, data konsumen, transaksi barang keluar, dan membuat laporan yang akan memudahkan perusahaan untuk mengelola data yang ada.

Namun, kelemahan dari solusi ini adalah sistem tidak memberikan laporan arus masuk dan keluar persediaan, sistem tidak memberikan laporan pembelian dan penjualan, tidak adanya tampilan pengolahan data hak akses *user*, sistem belum dilakukan pengujian sistem sehingga belum teruji kelayakan sistemnya, tidak adanya cetak kartu stok barang.

e. Literatur 05

Permasalahan terdapat pada perusahaan PT. Zokkas Sejahtera Jambi menghadapi beberapa masalah dalam pengelolaan inventori mereka. Pertama, penggunaan spreadsheet sebagai alat pengelolaan inventori menyebabkan proses pengumpulan data, stok inventori, dan pelaporan barang tidak optimal karena jumlah item data yang besar. Kedua, tidak ada sistem informasi inventori yang optimal yang menyebabkan perusahaan kesulitan dalam mengelola inventori mereka.

Solusi untuk mengatasi masalah tersebut, penelitian dilakukan dengan pembuatan sistem informasi inventori berbasis web dengan metode FIFO untuk mengatasi masalah dan meningkatkan efisiensi pengelolaan inventori. Dengan mengimplementasikan solusi tersebut, perusahaan dapat meningkatkan efisiensi dan akurasi dalam pengelolaan inventori mereka.

Namun, kelemahan dari solusi ini adalah, sistem tidak memberikan pengolahan data kategori barang, tidak adanya *fitur* konfigurasi hak akses pengguna, sistem belum dilakukan pengujian sistem sehingga belum teruji kelayakan sistemnya, perancangan sistem yang dibangun masih dilakukan dengan *structural design* sehingga akan menyebabkan penulisan fungsi code pada sistem menjadi berulang-ulang.

2.2 Sistem Informasi

Sistem informasi adalah suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung operasi, bersifat manajerial dan kegiatan strategidari suatu organisasidan menyediakan pihak luar tertentu dengan laporan-laporan yang diperlukan (Suryadharma dan Triyani, 2019).

Sistem informasi merupakan proses pengumpulan, penyimpanan, analisis sebuah informasi dengan tutjuan tertentu. Sistem informasi yang terdiri dari data (*input*) dan menghasilkan laporan (*output*) sehingga diterima oleh sistem lainnya serta kegiatan strategi dalam suatu organisasi dalam melakukan Tindakan atau keputusan (Prehanto, 2020).

Sistem informasi adalah kumpulan dari sub-sub sistem yang saling terintegrasi dan berkolaborasi untuk menyelesaikan masalah tertentu dengan cara mengolah data dengan alat yang namanya komputer sehinga memiliki nilai tambah dan bermanfaat bagi pengguna (Setiyanto, Nurmaesah, dan Rahayu, 2019).

Berdasarkan dari uraian definisi sistem informasi di atas, penulis menyimpulkan bahwa sistem informasi merupakan proses pengumpulan, penyimpanan, dan analisis informasi dengan tujuan tertentu yang terdiri dari data sebagai input dan menghasilkan laporan sebagai output, sehingga dapat diterima oleh sistem lain dan membantu dalam pengambilan keputusan.

Sistem adalah sebuah tatanan (keterpaduan) yang terdiri atas jumlah komponen fungsional (dengan satuan fungsi dan tugas khusus yang saling berhubungan dan secara bersama-sama bertujuan untuk memenuhi suatu proses tertentu (Hasanudin, 2018).

Sistem adalah kumpulan/group dari sub sistem/bagian/komponen apapun baik fisik atau pun non fisik yang saling berhubungan satu sama lain dan bekerja sama secara harmonis untuk mencapai satu tujuan tertentu (Kadang, Patulak, dan Upa, 2022).

Sistem merupakan suatu kumpulan dari komponen-komponen yang terdiri dari manusia, perangkat lunak dan perangkat keras yang membentuk satu kesatuan untuk mencapai tujuan tertentu (Setiyanto, Nurmaesah, dan Rahayu, 2019).

Berdasarkan dari uraian definisi sistem di atas, penulis menyimpulkan bahwa sistem adalah tatanan yang terdiri dari komponen fungsional, sub sistem, atau bagian yang saling berhubungan dan bekerja sama secara harmonis untuk mencapai tujuan tertentu, serta membentuk satu kesatuan.

2.2.1 Karakteristik Sistem

Karakteristik Sistem Untuk mencapai tujuannya, suatu sistem harus memiliki sifat-sifat tertentu atau suatu karakteristik tertentu, yaitu (Waluyo, Hanafri, dan Sulaeman, 2019):

1. Komponen (*Components*)

Komponen adalah suatu sistem terdiri dari sejumlah unsur-unsur atau komponen yang saling berinteraksi dan bekerja sama yang membentuk suatu kesatuan.

2. Batasan sistem (*Boundary*)

Batasan sistem merupakan daerah yang membatasi antara satu sistem dengan sistem lainnya atau dengan lingkungan luarnya.

3. Lingkungan Luar Sistem (*Environments*)

Lingkungan yang dimaksudkan disini adalah segala sesuatu yang berada diluar batas sistem yang mempengaruhi operasi sistem. Dimana sistem yang mempengaruhi ini dapat bersifat menguntungkan ataupun sebaliknya, merugikan.

4. Penghubung (*Interface*)

Interface merupakan sarana penghubung antar subsistem yang memungkinkan sumberdaya-sumberdaya yang ada untuk mengalir dari satu subsistem ke subsistem lain.

5. Masukan (*Input*)

Input adalah segala sesuatu yang dimasukkan kedalam sistem. Masukan ini dapat berupa perawatan (*Maintenance Input*) dan masukan sinyal (*Signal Input*).

6. Keluaran (*Output*)

Output adalah hasil dari energi atau masukan yang diolah dan diklasifikasikan menjadi keluaran yang berguna atau menjadi sisa pembuangan yang tidak digunakan.

7. Pengolahan (*Process*)

Process adalah suatu sistem dapat mempunyai suatu bagian pengolah yang akan mengubah masukan menjadi keluaran.

8. Sasaran (*Objectives*)

Sasaran atau tujuan adalah alasan mengapa suatu sistem dirancang. Sasaran yang ditetapkan harus jelas dan spesifik agar sistem dapat memenuhi kebutuhan akan sistem.

2.3 *Inventory*

Inventory merupakan serangkaian kebijaksanaan dan pengendalian yang memonitor tingkat persediaan dan menentukan tingkat persediaan yang harus dijaga, kapan persediaan harus diisi dan berapa persen pesanan yang harus dilakukan. Sistem persediaan ini memiliki fungsi utama yaitu dapat menghasilkan arus informasi seperti (Rusmawan dan Wicaksono, 2022):

- a) Mendukung kerja rutin bagian inventory control dengan mendapatkan dan mencatat data persediaan.
- b) Mendukung keputusan yang diperlukan oleh bagian gudang dan bagian kontrol.
- c) Membantu persiapan laporan internal dan eksternal.

Persediaan merupakan salah satu unsur yang paling aktif dalam operasi perusahaan yang secara kontinu diperoleh, diubah, yang kemudian dijual kembali. Perusahaan dapat menghindari terjadinya kekurangan barang, keterlambatan jadwal pemenuhan produk yang dipesan konsumen dapat merugikan perusahaan dalam hal ini citra kurang baik (Vikaliana, Sofian, Solihati, Adji, dan Maulia, 2020).

Persediaan barang merupakan salah satu aktivitas kerja yang penting bagi perusahaan dagang, karena persediaan barang merupakan unsur utama dalam bidang perdagangan. Kesalahan kecil mengenai persediaan barang akan

mengakibatkan masalah yang fatal, baik itu penumpukan di gudang maupun kekosongan barang (Setiyanto, Nurmaesah, dan Rahayu, 2019).

Berdasarkan dari uraian definisi *inventory*/persediaan di atas, penulis menyimpulkan bahwa persediaan adalah kegiatan yang sangat penting dalam operasi perusahaan yang secara kontinu diperoleh, diubah, dan dijual kembali, karena dapat menghindari terjadinya kekurangan barang dan keterlambatan pemenuhan produk yang dapat merugikan perusahaan dalam hal ini citra kurang baik, dan kesalahan kecil dalam mengelola persediaan barang dapat mengakibatkan masalah yang fatal bagi perusahaan.

2.4 Sistem Informasi *Inventory*

Sistem informasi *inventory* adalah berdasarkan hasil penelitian yang telah dilakukan, sistem informasi *inventory* yang dibuat mampu berjalan secara efektif, yaitu dengan memberikan informasi yang tepat dan cepat terhadap ketersediaan stok produk di gudang. Serta dapat mempermudah pencarian barang yang akan digunakan (Mufida, Rahmawati, dan Hertiana, 2019).

Sistem informasi *inventory* barang merupakan suatu sistem yang dibuat untuk mengetahui jumlah barang yang terdapat di gudang. Di samping itu, penggunaan sistem persediaan barang yang baik diharapkan akan mengurangi resiko hilangnya ataupun pencurian terhadap persediaan barang. Persediaan barang (*inventory*) adalah hal yang sangat dibutuhkan pada perusahaan yang melakukan proses produksi (Pahlevi, Mulyani, dan Khoir, 2018).

Sistem informasi *inventory* adalah sebuah sistem yang digunakan untuk memasukkan data-data persediaan barang ke dalam database sehingga tidak terjadi kesalahan dalam input, output data, dan pembuatan laporan berdasarkan data yang diinginkan sesuai kebutuhan, dan disimpan menurut aturan tertentu dalam keadaan siap pakai dan tersimpan di dalam database (Reynaldi dan Rustam, 2022).

Berdasarkan dari uraian definisi sistem informasi *inventory* di atas, penulis menyimpulkan bahwa sistem informasi *inventory* dibuat untuk memberikan informasi yang tepat dan cepat terhadap ketersediaan stok produk di gudang serta mempermudah pencarian barang yang akan digunakan, mengurangi resiko

hilangnya atau pencurian data persediaan barang, dan memasukkan data persediaan barang ke dalam *database* untuk menghindari kesalahan *input/output* data dan membuat laporan sesuai kebutuhan dengan aturan penyimpanan tertentu.

2.5 Konsep Dasar Metode Persediaan

Menurut PSAK 14 ada beberapa macam metode penilaian persediaan yang umum digunakan yaitu: identifikasi khusus, biaya rata-rata (*Average*), masuk pertama keluar pertama (*FIFO*), masuk terakhir keluar pertama (*LIFO*) tetapi menurut PSAK 14 (revisi 2008) mengatakan bahwa metode yang boleh digunakan hanya metode *FIFO* dan *Average* (rata-rata). *Average cost* dan *moving average* adalah dua konsep yang sama dalam data keuangan (rupiah) dalam persediaan. *Average cost* adalah rata-rata biaya per unit barang atau jasa yang dihasilkan dari perhitungan yang dilakukan dengan membagi total biaya produksi dengan jumlah unit produksi atau barang yang telah beli dari *supplier*. *Average cost* dapat membantu perusahaan dalam menentukan harga pokok penjualan sehingga dapat menghasilkan keuntungan yang optimal. Sementara itu, *moving average cost* adalah metode rata-rata biaya per unit barang yang digunakan untuk menghitung rata-rata dari data dalam suatu periode waktu tertentu yang digambarkan dengan metode pencatatan perpetual. Metode ini biasanya digunakan untuk menghaluskan data dalam jangka waktu yang panjang, sehingga dapat memberikan gambaran yang lebih rinci dan akurat. (Swasono dan Prastowo, 2021).

2.5.1 Metode Penilaian Persediaan *Average Cost*

Metode *Average Cost* adalah metode penilaian yang didasari atas harga rata-rata dari periode yang bersangkutan, diaplikasikan untuk menghitung biaya dari persediaan akhir dan juga menghitung harga pokok penjualan pada setiap periode atas dasar biaya rata-rata per unit persediaan (Swasono dan Prastowo, 2021). Rumus dalam menentukan harga pokok penjualan dengan cara sebagai berikut:

$$\text{Harga Pokok Rata-Rata} = \frac{\text{Harga Pokok Persediaan}}{\text{Kuantitas Persediaan}} \dots\dots\dots (2.1)$$

2.5.2 Metode Pencatatan Persediaan

Dalam sistem perpetual, barang yang dijual dicatat menurut harga jual dan jenis barang yang terjual. Persediaan yang dibeli dicatat pada perkiraan persediaan. Pada saat persediaan dijual, akan dilakukan pencatatan atas harga jual dari persediaan tersebut beserta dengan harga pokok penjualan dari persediaan tersebut (Swasono dan Prastowo, 2021).

2.5.3 Average Berdasarkan Pencatatan Perpetual

Jika Toko sering membeli barang dan harga beli masing-masing pembelian berbeda, maka toko akan mengalami kesulitan dalam menentukan harga pokok barang yang dipakai/dijual dan harga pokok barang yang masih ada di gudang. Sebagai contoh data keluar masuk persediaan barang dagangan yang dikutip dari buku akuntansi dasar perhitungan persediaan sebagai berikut:

Tabel 2. 2 Data Keluar Masuk Persediaan Barang

Tanggal	Keterangan	Kwantitas (Unit)	Harga Perolehan Per Unit
1 Maret	Persediaan Awal	120	Rp. 200.000,-
5 Maret	Penjualan	84	
12 Maret	Pembelian	96	Rp. 210.000,-
19 Maret	Penjualan	48	
23 Maret	Penjualan	24	
27 Maret	Pembelian	60	Rp. 220.000,-
31 Maret	Pembelian	60	Rp. 220.000,-

Sumber: (Swasono dan Prastowo, 2021)

Dengan menggunakan data di atas, dan asumsi bahwa harga jual per unit adalah Rp. 300.000,- dimana pembelian maupun penjualan barang dagangan dilakukan secara kredit, maka besarnya nilai persediaan akhir, harga pokok penjualan dalam penyelesaiannya,

Average Menggunakan Sistem Pencatatan Perpetual Adalah:

Tabel 2. 3 Sistem Pencatatan Perpetual

Pembelian				Harga Pokok Penjualan			Saldo Persediaan		
Tgl	Uni t	HP	Total (Jt)	Unit	HP	Total (Jt)	Unit	HP	Total (Jt)
1 Mar							120	200.000	24

Tabel 2. 3 Sistem Pencatatan Perpetual (Lanjutan)

Pembelian				Harga Pokok Penjualan			Saldo Persediaan		
5 Mar				84	200.000	16,8	36	200.000	7,2
12 Mar	96	210.000	20,16				132	207.272,7	27,36
19 Mar				48	207.272,7	9,95	84	207.272,7	17,41
23 Mar				24	207.272,7	4,975	60	207.272,7	12,436
27 Mar	60	220.000	13,2				120	213.633	25,636
31 Mar	60	220.000	13,2				180	215.756	38,836

Sumber: (Swasono dan Prastowo, 2021)

Metode harga pokok rata-rata dalam system pencatatan perpetual dinamakan sebagai metode biaya rata-rata bergerak atau (*moving average cost method*). Dengan menggunakan metode penilaian rata – rata, harga pokok penjualan untuk masing – masing unit yang di jual pada 19 Maret adalah berdasarkan rata – rata perolehan per unit dari barang yang tersedia untuk di jual $[(7,2 \text{ jt} + 20,16 \text{ jt}) : (36 + 96)]$, yaitu sebesar Rp. 207.272,7 per unit. Saldo persediaan akhir pada tanggal 19 Maret (setelah penjualan 48 unit) adalah 84 unit dengan rata – rata perolehan per unit sebesar Rp. 207.272,7. Kemudian 24 unit yang dijual pada tanggal 23 Maret masih tetap menggunakan harga pokok penjualan sebesar Rp. 207.272,7 per unit, karena antara tanggal 12 Maret akhir (setelah pembelian 96 unit) sampai dengan tanggal 23 Maret awal tidak ada transaksi pembelian barang dagangan dari pemasok; dengan kata lain bahwa besarnya rata-rata harga perolehan per unit dari barang yang tersedia untuk dijual per tanggal 23 Maret adalah masih tetap sama sebesar Rp. 207.272,7.

Namun, setelah dilakukan pembelian barang dagangan dari pemasok pada tanggal 27 Maret, yaitu sebanyak 60 unit, maka besarnya rata – rata harga perolehan per unit dari barang yang tersedia untuk dijual adalah menjadi Rp. 213.633. Berikutnya, pada tanggal 31 Maret setelah dilakukan pembelian barang dagangan lagi dari pemasok sebanyak 60 unit juga, maka sekarang besarnya rata –rata harga perolehan per unit dari barang yang tersedia untuk dijual berubah menjadi Rp. 215.756.

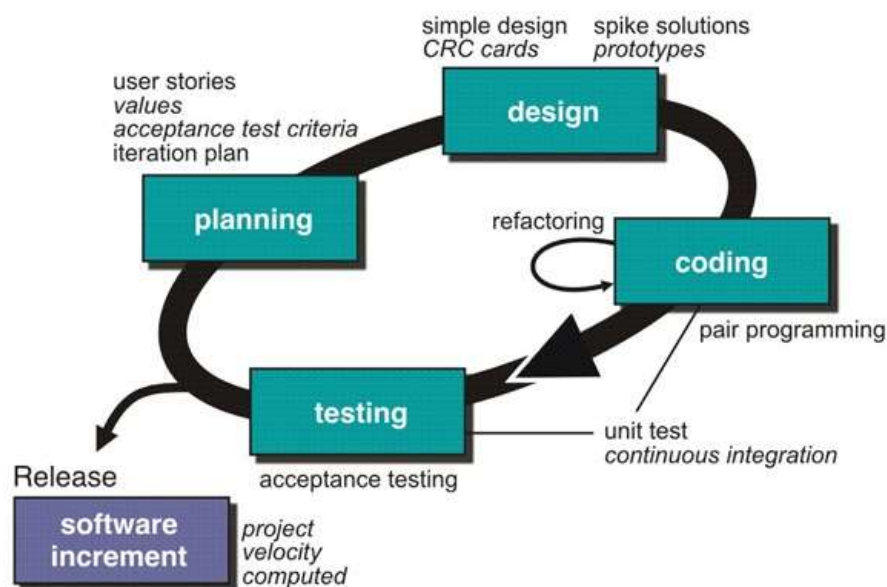
2.6 Website

Website adalah suatu media publikasi elektronik yang terdiri dari halaman-halaman *web (web page)* yang terhubung satu dengan yang lain menggunakan *link*

yang dilekatkan pada suatu *teks* atau *image*. *Website* dibangun dengan menggunakan bahasa *Hypertext Markup Language* (HTML) dan memanfaatkan protokol komunikasi *Hypertext Transfer Protocol* (HTTP) yang terletak pada *application layer* pada referensi *layer* OSI. Halaman *website* diakses menggunakan aplikasi yang disebut *internet browser* (Qadafi dan Wahyudi, 2020). *Website* juga dapat diartikan sebagai kumpulan halaman yang menampilkan informasi data *teks*, data gambar, data animasi, suara, *video*, baik yang bersifat statis maupun dinamis yang membentuk satu rangkaian bangunan yang saling terkait, di mana masing-masing dihubungkan dengan jaringan-jaringan halaman (*hyperlink*) (Meilano, Damanik, dan Tanto, 2019).

2.9 Metode Pengembangan *Extreme Programming*

Extreme Programming (XP) adalah metodologi pengembangan perangkat lunak yang ditujukan untuk meningkatkan kualitas perangkat lunak dan tanggap terhadap perubahan kebutuhan pelanggan. Jenis pengembangan perangkat lunak semacam ini dimaksudkan untuk meningkatkan produktivitas dan memperkenalkan pos pemeriksaan di mana persyaratan pelanggan baru dapat diadopsi (Pressman, 2019). Tahapan yang terdapat pada metode pengembangan perangkat lunak *Extreme Programming* dapat dilihat pada gambar sebagai berikut.



Gambar 2. 1 Tahapan Metode *Extreme Programming*

Sumber: (Pressman, 2019)

Paradigma yang diinginkan mencakup didalam seperangkat aturan dan praktik-praktik dalam empat *konteks* kegiatan kerangka kerja (Armanda dan Putra, 2020) yaitu:

1. *Planning* (Perencanaan)

Perencanaan yang bertujuan awal untuk mendengarkan kebutuhan secara fungsional dan non fungsional serta membuat penggambaran sistem secara *interface* sebagai keperluan outputnya.

- *User stories* adalah deskripsi singkat dari kebutuhan pengguna yang harus dipenuhi oleh perangkat lunak yang akan dikembangkan.
- *Acceptance Test Criteria* dan *Iteration Plan* adalah rencana iterasi yang berisi daftar *user stories* dan kriteria penerimaan untuk setiap user story.

2. *Design* (Perancangan)

Perancangan bertujuan untuk membangun dan menggambarkan sistem yang akan dibuat guna mempermudah proses pengkodean.

- *Simple Design CRC Cards* adalah kartu yang digunakan untuk merepresentasikan kelas-kelas dalam perangkat lunak yang akan dikembangkan.
- *Spike Solutions Prototypes* adalah prototipe sederhana yang dibuat untuk mengevaluasi teknologi atau konsep tertentu.

3. *Coding* (Pengkodean)

Pengkodean bertujuan untuk membuat hasil perancangan dengan melakukan konstruksi atau pembuatan sistem secara seluruh menggunakan kode.

- *Pair Programming* adalah praktik di mana dua orang bekerja bersama-sama pada satu komputer untuk menyelesaikan tugas pemrograman.
- *Refactoring* adalah proses memperbaiki kode tanpa mengubah fungsionalitasnya.

- *Unit Test* adalah jenis tes perangkat lunak di mana unit-unit kecil dari kode diuji secara terpisah untuk memastikan bahwa setiap unit berfungsi dengan benar.
- *Continuous Integration* adalah praktik di mana kode dari beberapa pengembang digabungkan secara otomatis dan diuji secara otomatis.

4. *Test* (Pengujian)

Pengujian dilakukan untuk menguji hasil pengkodean dan selanjutnya dilakukan evaluasi dan implementasi.

- *Acceptance Testing* adalah memverifikasi bahwa perangkat lunak memenuhi persyaratan dan kebutuhan pengguna serta bisnis. Hasil dari pengujian ini menentukan apakah perangkat lunak dapat diterima atau tidak.

5. *Software Increment* (Inkremen Perangkat Lunak)

Setelah semua tahapan selesai, perangkat lunak akan diinkremen dan siap untuk dirilis.

- *Release* sebagai tahap akhir dalam siklus pengembangan perangkat lunak. Ini melibatkan persiapan perangkat lunak yang telah selesai untuk disebarkan kepada pengguna akhir.
- *Project Velocity Computed* adalah kecepatan proyek dihitung untuk mengetahui seberapa cepat proyek tersebut berjalan.

2.10 Perancangan Sistem *Unified Modeling Language (UML)*

Pada perkembangan teknik pemrograman berorientasi objek, dibuatlah sebuah standarisasi bahasa permodelan untuk membangun perangkat lunak dengan menggunakan teknik pemrograman berorientasi objek. Bahasa permodelan ini adalah *Unified Modelling Language (UML)*. Menurut (Rosa A. S dan M. Shalahuddin, 2018). UML merupakan bahasa visual untuk permodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks pendukung untuk menspesifikasikan, menggambarkan, membangun, dan dokumentasi dari sistem perangkat lunak. *UML* terdiri dari 13 macam diagram

yang dikelompokkan dalam 3 kategori (Rosa A. S dan M. Shalahuddin, 2018) yaitu:


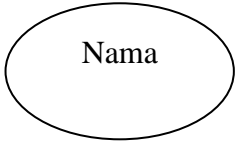
- a. *Structure Diagrams* yaitu kumpulan diagram yang digunakan untuk menggambarkan suatu struktur statis dari sistem yang dimodelkan.
- b. *Behavior Diagrams* yaitu kumpulan diagram yang digunakan untuk menggambarkan kelakuan sistem atau rangkaian perubahan yang terjadi pada sebuah sistem.
- c. *Interactions Diagrams* yaitu kumpulan diagram yang digunakan untuk menggambarkan interaksi sistem dengan sistem lain maupun interaksi antar subsistem pada suatu sistem.

Dari 13 model diagram tersebut, penelitian ini hanya mengambil 2 model diagram yaitu *Usecase Diagram* dan *Activity Diagram* (Rosa A. S dan M. Shalahuddin, 2018).



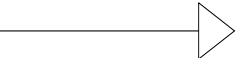


2.10.1 *Usecase Diagram*

Menurut (Rosa A. S dan M. Shalahuddin, 2018). “*Use case diagram* atau diagram *use case* merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat oleh pengembang sistem sebelum melakukan perancangan antarmuka pada suatu perangkat lunak”. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. simbol-simbol yang ada pada diagram *use case* dapat dilihat pada gambar 2.3 di bawah ini.

Tabel 2. 4 *Usecase Diagram*

No.	Simbol	Keterangan
1.	Aktor/ <i>Actor</i>  Nama Aktor	Orang, proses, atau sistem lain yang berinteraksi dengan sistem Informasi yang akan dibuat itu sendiri.
2.	<i>Use case</i>  Nama	Fungsionalitas yang disediakan sebagai unit-unit yang saling bertukar pesan antar unit atau aktor; biasanya dinyatakan dengan menggunakan kata kerja di awal <i>frase</i> nama <i>use case</i> .

Tabel 2.4 *Usecase Diagram (Lanjutan)*




No.	Simbol	Keterangan
3.	<i>Association</i> 	Komunikasi antara aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan aktor.
4.	<i>Extend/Ekstensi</i> <i><<extend>></i> 	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan itu.
5.	<i>Generalization/</i> <i>Generalisasi</i> 	Hubungan generalisasi dan spesialisasi (umum khusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih dari lainnya.
6	<i>Menggunakan/</i> <i>Include/Uses</i> <i><<include>></i>  <i>«uses»</i> 	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> di mana <i>use case</i> yang ditambahkan memerlukan <i>use case</i> ini ntuk menjalankan fungsinya atau sebagai syarat.

Sumber: (Rosa A. S dan M. Shalahuddin, 2018).



2.10.2 Activity Diagram

Diagram aktifitas atau *activity Diagram* menggambarkan workflow (aliran kerja) atau aktivitas sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Yang perlu diperhatikan adalah bahwa diagram aktivitas menggambarkan aktifitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem (Rosa A. S dan M. Shalahuddin, 2018).

Tabel 2. 5 *Activity Diagram*

No	Simbol	Keterangan
1	Status awal 	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal.
2	Aktivitas 	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
3	Percabangan 	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu.

Tabel 2.5 *Activity Diagram (Lanjutan)*

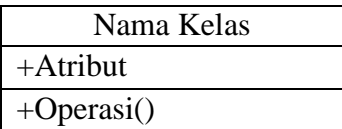


No	Simbol	Keterangan
4	penggabungan/ <i>join</i> 	Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu.
5	Status akhir 	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir.
6	<i>Swimlane</i>	<i>Swimlane</i> memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi.
	Nama <i>Swimlane</i>	

Sumber : (Rosa A. S dan M. Shalahuddin, 2018).

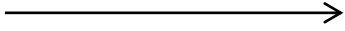
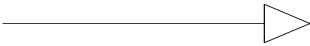
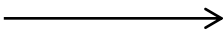
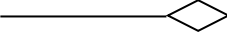
2.10.3 Class Diagram

Class Diagram menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi. Diagram kelas dibuat agar pembuat program atau *programmer* membuat kelas-kelas sesuai rancangan di dalam diagram kelas agar antara dokumentasi perancangan dan perangkat lunak sinkron (Rosa dan M shalahuddin, 2018).

Tabel 2. 6 *Class Diagram*

No	Simbol	Keterangan
1.	Kelas 	Kelas pada struktur sistem
2.	Antarmuka/ <i>Interfaces</i>  Nama_ <i>interfaces</i>	Sama dengan konsep <i>interfaces</i> dalam pemrogram berorientasi objek
3	Asosiasi / <i>association</i> 	Relasi antar kelas dengan makna umum, asosiasi biasanya disertai dengan <i>multiplicity</i> .

Tabel 2. 6 Simbol *Class Diagram* (Lanjutan)

No	Simbol	Keterangan
4	Asosiasi berarah / <i>directed association</i> 	Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya disertai dengan <i>multiplicity</i> .
5	Generalisasi 	Relasi antar kelas dengan makna generalisasi/spesialisasi (umum khusus).
6	Kebergantungan / <i>dependency</i> 	Kebergantungan antar kelas.
7	Agregasi / <i>aggregation</i> 	Relasi antar kelas dengan makna-bagian (<i>wholepart</i>).

Sumber: (Rosa A. S dan M. Shalahuddin, 2018).

2.11 Analisis PIECES

Analisis *PIECES* adalah metode analisis sebagai dasar untuk memperoleh pokok-pokok permasalahan yang lebih spesifik. Dalam menganalisis sebuah sistem, biasanya akan dilakukan terhadap beberapa aspek antara lain adalah kinerja, informasi, ekonomi, keamanan aplikasi, efisiensi dan pelayanan pelanggan. Analisis *PIECES* ini sangat penting untuk dilakukan karena dalam analisis ini biasanya akan ditemukan beberapa masalah utama maupun masalah yang bersifat gejala dari masalah utama (Nuryasin, Saputra, Hardi, dan Fadholur, 2019). Analisis *PIECES* digunakan untuk mengoreksi atau memperbaiki sistem dalam hal *Performances* (Kinerja), *Information* (Informasi), *Economics* (Ekonomi), *Control* (Keamanan), *Efficiency* (Efisiensi), dan *Service* (Layanan). Dari analisis *PIECES* yang dilakukan, didapatkan scenario kebutuhan teknologi. Lalu setelah itu, menetapkan kebutuhan sistem informasi dengan cara mengidentifikasi kebutuhan informasi dan mengidentifikasi fungsional serta non fungsionalnya (Putra, Sopian, dan Ratnasari, 2021).

2.12 Alat Pendukung

2.12.1 XAMPP

XAMPP adalah perangkat lunak bebas, yang mendukung banyak sistem operasi, merupakan kompilasi dari beberapa program. XAMPP merupakan *tool* yang menyediakan paket perangkat lunak ke dalam satu buah paket (Roza, Fauzan, Dan Rahayu, 2020). Paket XAMPP yaitu PHP, *MySQL*, paket ini dapat di *download* secara gratis dan legal (Waluyo, Hanafri, dan Sulaeman, 2019).

2.12.2 HTML

HTML (*Hyper Text Mark Up Language*) merupakan bahasa yang digunakan untuk mendeskripsikan struktur sebuah halaman *web*. HTML berfungsi untuk mempublikasi dokumen *online*. 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 ($\langle \rangle$). *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* (Pahlevi, Mulyani, dan Khoir, 2018).

2.12.3 CSS

CSS kepanjangan dari *Cascading Style Sheet* adalah bahasa-bahasa yang merepresentasikan halaman *web*. Seperti warna, *layout*, dan *font*. Dengan menggunakan CSS, seorang *web developer* dapat membuat halaman *web* yang dapat beradaptasi dengan berbagai macam ukuran layar. Pembuatan CSS biasanya terpisah dengan halaman HTML. Meskipun CSS dapat disisipkan di dalam halaman HTML. Hal ini ditujukan untuk memudahkan pengaturan halaman HTML yang memiliki rancangan yang sama (Pahlevi, Mulyani, dan Khoir, 2018).

2.12.4 PHP

PHP adalah bahasa pemrograman untuk dijalankan melalui halaman *web*, umumnya digunakan untuk mengolah informasi di *internet*. Sedangkan dalam pengertian lain PHP adalah singkatan dari *Hypertext Preprocessor* yaitu bahasa

pemrograman *webserverside* yang bersifat *open source* atau gratis. PHP merupakan script yang menyatu dengan HTML dan berada pada server (Pahlevi, Mulyani, dan Khoir, 2018).

2.12.5 Javascript

JavaScript adalah bahasa pemrograman web yang bersifat *Client Side Programming Language*. *Client Side Programming Language* adalah tipe bahasa pemrograman yang pemrosesannya dilakukan oleh client. Aplikasi client yang dimaksud merujuk kepada web browser seperti Google Chrome, Mozilla Firefox, Opera Mini dan sebagainya. *JavaScript* pertama kali dikembangkan pada pertengahan dekade 90'an. Meskipun memiliki nama yang hampir serupa, *JavaScript* berbeda dengan bahasa pemrograman Java. Untuk penulisannya, *JavaScript* dapat disisipkan di dalam dokumen HTML ataupun dijadikan dokumen tersendiri yang kemudian diasosiasikan dengan dokumen lain yang dituju. *JavaScript* mengimplementasikan fitur yang dirancang untuk mengendalikan bagaimana sebuah halaman web berinteraksi dengan penggunanya (Pahlevi, Mulyani, dan Khoir, 2018).

2.12.6 MySQL

MySQL adalah salah satu jenis *database* yang banyak digunakan untuk membuat sistem informasi berbasis *web* yang dinamis. MySQL termasuk jenis RDBMS (*Relational Database Management Sistem*). MySQL ini mendukung Bahasa pemrograman PHP. MySQL juga mempunyai *query* atau bahasa SQL (*Structured Query Language*) yang *simple* dan menggunakan *escape character* yang sama dengan PHP (Pahlevi, Mulyani, dan Khoir, 2018).

2.13 Pengujian Black Box

Pengujian *black box* merupakan pengujian program berdasarkan fungsi dari program. Tujuan dari metode black box adalah untuk menemukan kesalahan fungsi dari program. Apabila perintah yang diberikan proses menghasilkan *output* yang sesuai dengan kebutuhan fungsionalnya, maka program aplikasi yang bersangkutan sudah benar, tetapi apabila *output* (Putramawan, Mardiyah, dan M.

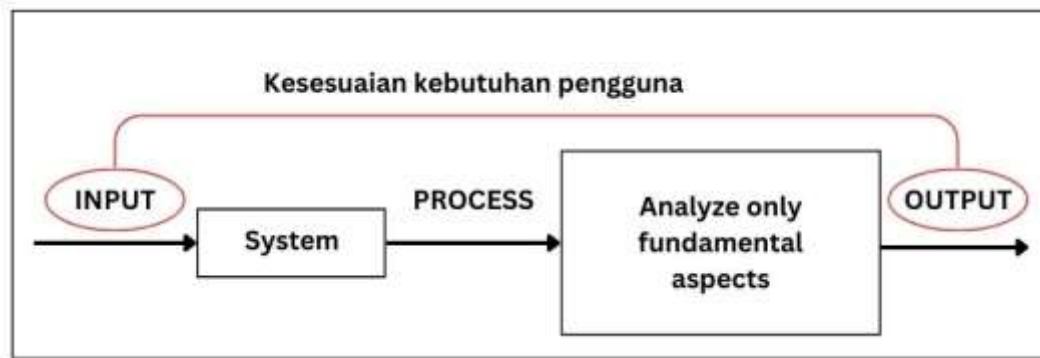
Irfan, 2019). *Black Box Testing* berfokus pada spesifikasi fungsional dari perangkat lunak. *Black Box Testing* cenderung untuk menemukan hal-hal berikut.

- a. Fungsi yang tidak benar atau tidak ada.
- b. Kesalahan antarmuka (*interface errors*).
- c. Kesalahan pada struktur data dan akses basis data.
- d. Kesalahan performansi (*performance errors*).
- e. Kesalahan inisialisasi dan terminasi.

Keuntungan menggunakan metode *Black Box Testing* adalah:

1. Penguji tidak perlu memiliki pengetahuan tentang Bahasa pemrograman tertentu.
2. Pengujian dilakukan dengan fokus pada perilaku eksternal sistem. Pengujian dilakukan tanpa memerlukan pengetahuan tentang kode atau desain internal sistem tersebut.
3. *Black Box Testing* mensimulasikan perspektif pengguna atau pemangku kepentingan eksternal yang berinteraksi dengan sistem. Dengan mempertimbangkan hanya masukan dan keluaran sistem, pengujian dapat mengevaluasi sejauh mana sistem memenuhi persyaratan, harapan, dan spesifikasi pengguna.
4. *Black Box Testing* mendorong cakupan pengujian yang lebih luas karena fokusnya pada validasi sistem terhadap persyaratan yang ditentukan. Dengan tidak terikat pada implementasi internal, pengujian dapat dilakukan dengan berbagai skenario pengujian dan data masukan yang berbeda.
5. Dengan fokus pada perilaku eksternal sistem, *Black Box Testing* membantu memastikan kualitas dari sudut pandang pengguna. Dengan menguji sistem seolah-olah sebagai entitas terpisah, pengujian dapat mengidentifikasi masalah yang mungkin terjadi ketika sistem berinteraksi dengan komponen eksternal atau lingkungan operasional yang berbeda.

Berikut adalah alur pengujian kelayakan sistem *Black Box Testing* yang dimodelkan seperti gambar dibawah ini.



Gambar 2. 2 Alur pengujian *Black Box Testing*
Sumber: (Lumbanraja, 2018)