

**OPTIMASI PENGELOLAAN PERSEDIAAN BARANG
MENGUNAKAN METODE PERPETUAL PADA APLIKASI
INVENTORY DI PT. VISI KARYA PRAKARSA**

SKRIPSI

**Karya Tulis sebagai Syarat untuk Memperoleh
Gelar Sarjana Komputer dari Fakultas Teknologi Informasi
Universitas Bale Bandung**

Disusun Oleh :

HERLIN MARTIN MARCELLA

NPM. 301190013



**PROGRAM STRATA 1
PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS BALE BANDUNG
BANDUNG**

2023

LEMBAR PERSETUJUAN PEMBIMBING

**OPTIMASI PENGELOLAAN PERSEDIAAN BARANG MENGGUNAKAN
ALGORITMA PERPETUAL PADA APLIKASI INVENTORY DI PT. VISI
KARYA PRAKARSA**

Disusun oleh :

HERLIN MARTIN MARCELLA

NPM. 301190013

Telah terima dan disetujui untuk memenuhi persyaratan mencapai gelar

SARJANA KOMPUTER

Pada

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS BALE BANDUNG**

Baleendah, September 2023

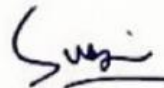
Disetujui oleh :

Pembimbing Utama



Mohammad Bayu Anggara, S. Kom., M.Kom.
NIK. 04104823002

Pembimbing Pendamping



Sutiyono S.T., M.Kom.
NIK. 01043180002

LEMBAR PENGESAHAN PENGUJI

**OPTIMASI PENGELOLAAN PERSEDIAAN BARANG MENGGUNAKAN
ALGORITMA PERPETUAL PADA APLIKASI INVENTORY DI PT. VISI
KARYA PRAKARSA**

Disusun oleh :
HERLIN MARTIN MARCELLA
NPM. 301190013

Telah terima dan disetujui untuk memenuhi persyaratan mencapai gelar
SARJANA KOMPUTER

Pada

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS BALE BANDUNG**

Baleendah, September 2023

Disetujui oleh :

Penguji 1



Yudi Herdiana, S.T, M.T.
NIK. 04104808008

Penguji 2



Yusuf Muharam, S.Kom., M.Kom
NIK. 04104820003

LEMBAR PENGESAHAN PROGRAM STUDI
OPTIMASI PENGELOLAAN PERSEDIAAN BARANG MENGGUNAKAN
ALGORITMA FIFO PERPETUAL PADA APLIKASI INVENTORY DI
PT. VISI KARYA PRAKARSA

Disusun oleh :
HERLIN MARTIN MARCELLA
NPM. 301190013

Skripsi ini telah diterima dan disetujui untuk memenuhi persyaratan
mencapai gelar

SARJANA KOMPUTER

Pada

PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS BALE BANDUNG


Baleendah, September 2023

Disetujui oleh :

Mengetahui,
Dekan


Yudi Hestianty, S.T., M.T.
NIK. 04101808008

Mengesahkan,
Ketua Program Studi


Yusuf Maharam, M.Kom.
NIK. 04101820003

HALAMAN PERNYATAAN

Yang bertanda tangan dibawah ini, saya:

Nama : Herlin Martin Marcella
NPM : 301190013
Jurusan : Teknik Informatika
Fakultas : Fakultas Teknologi Informasi
Judul : Optimasi Pengelolaan Persediaan Barang
Menggunakan Metode Perpetual Pada Aplikasi
Inventory Di PT. Visi Karya Prakarsa

Dalam rangka pengembangan kajian ini, penulis dengan tulus mengungkapkan bahwa penulisan skripsi ini merujuk pada hasil penelitian yang telah dilakukan, serta merupakan hasil pemikiran dan pemaparan yang bersifat orisinal dari penulis sendiri, baik dalam konteks penulisan laporan maupun pelaksanaan aktivitas pemrograman yang dicantumkan sebagai bagian integral dari penelitian ini. Setiap kontribusi atau karya dari pihak lain akan diidentifikasi dengan jelas sesuai dengan pedoman penulisan karya ilmiah yang berlaku, dengan penekanan pada penggunaan referensi yang akurat dan sistematis.

Baleendah, September 2023

Herlin Martin Marcella

NPM, 301190013

ABSTRAK

Penelitian ini berfokus pada perancangan dan implementasi aplikasi persediaan barang (*inventory*) dengan menerapkan metode *perpetual*. Aplikasi *inventory* merupakan suatu alat yang diciptakan dengan tujuan menghasilkan laporan digital mengenai aliran barang masuk dan keluar. Metode *perpetual* ini dapat membantu dalam pencatatan dan penilaian persediaan berdasarkan periode dalam memantau stok barang, serta meningkatkan efisiensi kerja karyawan.

Studi ini dilakukan di PT. Visi Karya Prakarsa, dimana berdasarkan hasil wawancara dengan staf, ditemukan bahwa sistem pencatatan persediaan masih manual dengan masalah seperti kesalahan input data, *overstocking* dan *understocking*, keterlambatan pengiriman dari *supplier*, serta masalah kerusakan barang. Keadaan ini menyebabkan kurangnya efisiensi dalam pengelolaan persediaan dan ketidakakuratan dalam laporan persediaan, yang pada gilirannya mempengaruhi pengaturan siklus pengadaan barang dan meningkatkan biaya pengadaan. Oleh karena itu, tujuan dari penelitian ini adalah merancang sistem keluar masuk barang di PT. Visi Karya Prakarsa yang mampu menghasilkan laporan persediaan, sambil mengatasi masalah ketidakakuratan dalam pengaturan siklus pengadaan. Pendekatan kuantitatif digunakan dalam penelitian ini, dengan penerapan model pengembangan SDLC waterfall untuk merancang aplikasi. Visual Studio Code digunakan untuk mengimplementasikan kode program HTML, CSS, dan PHP. Balsamiq mockup yang digunakan untuk merancang antarmuka aplikasi. Bootstrap 4.7 digunakan sebagai framework website, dan MySQL digunakan sebagai basis data.

Hasil dari penelitian ini diharapkan dapat memberikan manfaat dalam bentuk aplikasi *inventory* dengan menggunakan metode *perpetual*. Aplikasi ini diharapkan dapat membantu perusahaan bisa mengatasi masalah persediaan dalam periode tertentu setiap bulannya, menganalisis keberhasilan dan kelemahan perusahaan, serta memberikan arahan untuk perbaikan manajemen perusahaan ke depannya.

Kata Kunci : *Aplikasi Inventory, Metode Perpetual, Persediaan Barang, Waterfall*

ABSTRACT

This research focuses on the design and implementation of inventory applications by applying the perpetual method. Inventory application is a tool created with the aim of producing digital reports regarding the flow of incoming and outgoing goods. This perpetual method can assist in recording and valuing inventories based on the period in monitoring stocks, as well as increasing employee work efficiency. This study was conducted at PT. Vision Karya Prakarsa, where based on the results of interviews with staff, it was found that the inventory recording system was still manual with problems such as data input errors, overstocking and understocking, delays in delivery from suppliers, and problems with damage to goods. This situation causes a lack of efficiency in inventory management and inaccuracies in inventory reporting, which in turn affects the regulation of the procurement cycle and increases procurement costs. Therefore, the purpose of this research is to design a system for entering and leaving goods at PT. Vision Karya Prakarsa which is able to generate inventory reports, while addressing the problem of inaccuracies in the regulation of the procurement cycle. A quantitative approach is used in this study, with the application of the SDLC waterfall development model to design applications. Visual Studio Code is used to implement HTML, CSS, and PHP program code. Balsamiq mockup used to design application interfaces. Bootstrap 4.7 is used as a website framework, and MySQL is used as a database.

The results of this study are expected to provide benefits in the form of inventory applications using the perpetual method. This application is expected to help companies overcome inventory problems in a certain period each month, analyze the company's successes and weaknesses, and provide directions for improving company management in the future.

Keywords: Inventory Application, Perpetual Method, Inventory, Waterfall

KATA PENGANTAR

Puji dan Syukur kami panjatkan ke Hadirat Allah SWT, karena berkat limpahan Rahmat dan Karunia-Nya sehingga penulis dapat menyelesaikan laporan skripsi ini dengan baik dan tepat pada waktunya. Skripsi ini merupakan salah satu syarat untuk menyelesaikan program studi Teknik Informatika jenjang Strata-1 Universitas Bale Bandung. Dalam skripsi ini membahas mengenai Optimasi Pengelolaan Persediaan Barang Menggunakan Metode Perpetual Pada Aplikasi Inventory Di PT. Visi Karya Prakarsa,

Laporan skripsi ini dapat dibuat dengan bantuan dari berbagai pihak untuk menyelesaikan tantangan dan hambatan selama pengerjaannya. Oleh karena itu, pada kesempatan kali ini penulis ingin mengucapkan terima kasih yang sebesar-besarnya kepada:

1. Allah SWT yang telah memberikan karunia-Nya selama proses pengerjaan laporan skripsi ini.
2. Kedua Orang Tua yaitu Bapak Alm. Dadang Hermanto dan Ibu Euis Amanah serta kedua saudara kandung saya yang telah memberikan doa dan dukungan selama proses pembuatan laporan skripsi ini.
3. Bapak Yudi Herdiana, S.T., M.T selaku Dekan Fakultas Teknologi Informasi Universitas Bale Bandung.
4. Bapak Yusuf Muharam, M.Kom. selaku Ketua Program Studi Teknik Informatika Universitas Bale Bandung.
5. Bapak Mohammad Bayu Anggara, M.Kom. selaku dosen pembimbing utama.
6. Bapak Sutiyono W P., S.T., M.T., M.Kom. selaku dosen pembimbing pendamping.
7. Dosen Penguji 1.
8. Dosen Penguji 2.
9. Seluruh Dosen beserta staf Fakultas Teknologi Informasi Universitas Bale Bandung.
10. CTO PT. Visi Karya Prakarsa yaitu Bapak Lucky Luqman

11. Siffa Rahwamati dan Tiana Rizki Ananda yang telah bersedia membantu penulis dan memberi semangat dalam penulisan skripsi ini.
12. Rekan-rekan FTI angkatan 2019 yang senantiasa saling membantu dan memberikan semangat dalam proses penelitian maupun penulisan laporan
13. Semua pihak yang memberikan dukungan dan bantuannya kepada penulis untuk menyelesaikan skripsi ini namun tidak bisa disebutkan satu persatu

Penulis menyadari bahwa masih banyak kekurangan dalam penyusunan skripsi ini, namun penulis telah melakukan semua yang diperlukan dengan yang terbaik. Oleh karena itu, segala kritik dan saran yang membangun akan penulis terima dengan senang hati. Akhir kata, semoga laporan skripsi ini dapat diterima dan bermanfaat bagi kita semua.

Baleendah, September 2023

Penulis

DAFTAR ISI

ABSTRAK	v
<i>ABSTRACT</i>	vi
KATA PENGANTAR.....	vii
DAFTAR ISI.....	ix
DAFTAR GAMBAR.....	xii
DAFTAR TABEL	xiv
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	3
1.3 Batasan Masalah.....	3
1.4 Tujuan Penelitian.....	3
1.5 Manfaat Penelitian.....	4
1.6 Metodologi Penelitian	4
1.6.1 Metode Pengumpulan Data	4
1.6.2 Metode Pengembangan Sistem	5
1.7 Sistematika Penulisan.....	6
BAB II TINJAUAN PUSTAKA.....	8
2.1 Landasan Teori	8
2.1.1 Acuan Penelitian Pertama	9
2.1.2 Acuan Penelitian Kedua.....	10
2.2 Dasar Teori	11
2.2.1 Persediaan	11
2.2.2 Sistem.....	12
2.2.3 Sistem Penilaian Persediaan.....	13

2.2.4	Sistem Pencatatan Persediaan	13
2.2.5	Metode SDLC <i>Waterfall</i>	15
2.2.6	<i>Unified Modelling Language (UML)</i>	17
2.2.7	Balsamiq Mockup	20
2.2.8	<i>Bootsrap</i> versi 4.7	20
2.2.9	XAMPP versi 7.4	21
2.2.10	MySQL.....	21
2.2.11	Visual Studio Code	21
2.2.12	PHP versi 7.4.5.....	22
2.2.13	<i>Cascading Style Sheet (CSS)</i>	22
2.2.14	<i>Hypertext Markup Language (HTML)</i>	22
2.2.15	Metode pengujian <i>Black Box Testing</i>	23
2.2.16	Metode pengujian penelitian	23
BAB III METODOLOGI PENELITIAN		25
3.1	Kerangka Pikir.....	25
3.2	Deskripsi.....	26
3.2.1	Identifikasi Masalah	26
3.2.2	Pengumpulan Data	26
3.2.3	Pengembangan Sistem	27
3.2.4	Pelaporan.....	31
BAB IV ANALISIS DAN PERANCANGAN		32
4.1	Analisis	32
4.1.1	Analisis Masalah	32
4.1.2	<i>Analisis Software</i>	32
4.1.3	Analisis Pengguna.....	33
4.1.4	<i>User Interface</i>	33

4.1.5	Fitur-Fitur	34
4.1.6	Analisis Data	35
4.1.7	Analisis Biaya	35
4.2	Perancangan.....	36
4.2.1	UML (Unified Modelling Language)	36
4.2.2	Struktur Tabel.....	49
4.2.3	Desain antarmuka.....	52
BAB V IMPLEMENTASI DAN PENGUJIAN		59
5.1	Impelementasi	59
1.1.1	Listing Program.....	59
1.1.2	Implementasi Sistem	78
1.1.3	Spesifikasi Sistem	79
1.1.4	Instalasi Sistem	79
1.1.5	Menjalankan Sistem.....	80
1.2	Pengujian	96
1.2.1	Uji Validitas.....	97
5.2.2	Uji Reliabilitas	101
BAB VI KESIMPULAN		106
6.1	Kesimpulan.....	106
6.1	Saran	107
DAFTAR PUSTAKA		109
LAMPIRAN		110

DAFTAR GAMBAR

Gambar 2.1 Bentuk umum sistem.....	12
Gambar 2.2 Tahapan metode <i>waterfall SDLC</i>	15
Gambar 3.1 Kerangka pikir.....	25
Gambar 4.1 <i>Use Case Diagram Admin</i>	36
Gambar 4.2 <i>Activity diagram login</i>	40
Gambar 4.3 <i>Activity diagram dashboard</i>	41
Gambar 4.4 <i>Activity diagram supplier</i>	42
Gambar 4.5 <i>Activity diagram barang</i>	43
Gambar 4.6 <i>Activity diagram transaksi barang masuk</i>	44
Gambar 4.7 <i>Activity diagram transaksi barang keluar</i>	45
Gambar 4.8 <i>Activity diagram laporan</i>	46
Gambar 4.9 <i>Activity diagram user management</i>	47
Gambar 4.10 <i>Class diagram aplikasi inventory</i>	48
Gambar 4.11 Halaman <i>login</i>	52
Gambar 4.12 Halaman <i>dashboard</i>	53
Gambar 4.13 Desain tampilan antarmuka <i>supplier</i>	53
Gambar 4.14 Desain tampilan antarmuka satuan barang.....	54
Gambar 4.15 Desain tampilan antarmuka jenis barang	55
Gambar 4.16 Desain tampilan antarmuka data barang	55
Gambar 4.17 Desain tampilan antarmuka transaksi barang masuk	56
Gambar 4.18 Desain tampilan antarmuka transaksi barang keluar.....	57
Gambar 4.19 Desain tampilan antarmuka laporan.....	57
Gambar 4.20 Halaman <i>user management</i>	58
Gambar 5.1 Halaman <i>Login</i>	81
Gambar 5.2 Halaman <i>dashboard</i>	81
Gambar 5.3 Halaman <i>supplier</i>	82
Gambar 5.4 Halaman tambah <i>supplier</i>	83
Gambar 5.5 Halaman edit tabel <i>supplier</i>	83
Gambar 5.6 Halaman data satuan barang.....	84

Gambar 5.7 Halaman tambah satuan barang.....	85
Gambar 5.8 Halaman ubah satuan barang.....	85
Gambar 5.9 Halaman hapus satuan barang	86
Gambar 5.10 Halaman jenis barang	87
Gambar 5.11 Halaman tambah jenis barang	87
Gambar 5.12 Halaman edit jenis barang	88
Gambar 5.13 Halaman hapus jenis barang.....	89
Gambar 5.14 Halaman data barang	89
Gambar 5.15 Halaman tambah data barang	90
Gambar 5.16 Halaman ubah data barang	90
Gambar 5.17 Halaman hapus data barang.....	91
Gambar 5.18 Halaman riwayat data barang masuk	92
Gambar 5.19 Halaman input barang masuk.....	92
Gambar 5.20 Halaman hapus barang masuk.....	93
Gambar 5.21 Halaman riwayat barang keluar.....	93
Gambar 5.22 Halaman input barang keluar	94
Gambar 5.23 Halaman hapus barang keluar	95
Gambar 5.24 Halaman transaksi laporan	95
Gambar 5.25 Halaman <i>user management</i>	96
Gambar 5.26 Grafik pertanyaan kuisisioner 1	102
Gambar 5.27 Grafik pertanyaan kuisisioner 2	103
Gambar 5.28 Grafik pertanyaan kuisisioner 3	103
Gambar 5.29 Grafik pertanyaan kuisisioner 4	103
Gambar 5.30 Grafik pertanyaan kuisisioner 5	104
Gambar 5.31 Grafik pertanyaan kuisisioner 6	104
Gambar 5.32 Grafik pertanyaan kuisisioner 7	104
Gambar 5.33 Grafik pertanyaan kuisisioner 8	105

DAFTAR TABEL

Tabel 2.1 Acuan Penelitian	8
Tabel 2.2 <i>Use Case Diagram</i>	17
Tabel 2.3 <i>Activity Diagram</i>	18
Tabel 2.4 <i>Class Diagram</i>	20
Tabel 4.1 Analisis Biaya	35
Tabel 4.2 Deskripsi Aktor	37
Tabel 4.3 Deskripsi <i>Use Case Login</i>	37
Tabel 4.4 Deskripsi <i>Use Case Dashborad</i>	37
Tabel 4.5 Deskripsi <i>Use Case</i> Tambah Barang	38
Tabel 4.6 Deskripsi <i>Use Case</i> Edit Barang	38
Tabel 4.7 Deskripsi <i>Use Case</i> Hapus Barang	39
Tabel 4.8 Deskripsi <i>Use Case</i> Pencetakan Barang	39
Tabel 4.9 Perancangan tabel <i>login</i>	49
Tabel 4.10 Perancangan tabel barang masuk	49
Tabel 4.11 Perancangan tabel barang keluar	50
Tabel 4.12 Perancangan tabel <i>supplier</i>	50
Tabel 4.13 Perancangan tabel barang	51
Tabel 4.14 Perancangan tabel satuan barang	51
Tabel 5.1 Spesifikasi perangkat keras	79
Tabel 5.2 Spesifikasi perangkat lunak	79
Tabel 5.3 Uji Validitas	97
Tabel 5.4 Hasil kuisioner responden	101

DAFTAR LAMPIRAN

Lampiran 1 : Hasil Wawancara dengan Narasumber.....	110
Lampiran 2 : Dokumentasi Wawancara.....	114
Lampiran 3 : Dokumentasi kantor PT. Visi Karya Prakarsa.....	115

BAB I

PENDAHULUAN

1.1 Latar Belakang

Persediaan adalah suatu aktiva yang meliputi barang-barang yang dimiliki perusahaan dengan maksud untuk dijual dalam suatu periode usaha tertentu, atau persediaan barang yang masih dalam pengerjaan/proses produksi. Secara umum, persediaan dalam sebuah bisnis dapat membantu atau menghambat tugas operasional perusahaan yang harus diselesaikan secara berurutan untuk memproduksi barang dan mendistribusikannya ke konsumen. Jika pasokan bahan baku tidak dapat diandalkan, maka akan mengganggu proses produksi. Hal inilah yang perlu menjadi fokus perhatian perusahaan untuk melakukan pencatatan persediaan bahan baku. (Ratnasari & Nurdiniah, 2022)

PT. Visi Karya Prakarsa merupakan perusahaan yang bergerak di bidang teknologi informasi. Perusahaan ini berlokasi di daerah Bandung Barat, Jl. Jupiter Barat No.37, Sekejati, Kec. Buahbatu. Sebagai salah satu perusahaan yang sedang berkembang, PT Visi Karya Prakarsa juga membutuhkan sistem informasi yang semakin banyak untuk meningkatkan tingkat efektifitas dan efisiensi kinerja pekerjaan yang dilakukan oleh karyawan. Bisnis utama di perusahaan ini adalah pembuatan produk aplikasi AntriQue dengan kegiatan utama nya yaitu membuat produk aplikasi bernama AntriQue yang merupakan aplikasi layanan antrian yang berbasis *web dan mobile*.

Hasil observasi dan wawancara peneliti menunjukkan bahwa salah satu masalah yang sering dihadapi oleh perusahaan ini yaitu mengenai *inventory management* dimana proses pencatatan yang dilakukan oleh karyawan PT. Visi Karya Prakarsa masih menggunakan sistem manual, umumnya mengakibatkan terjadinya kesalahan input keluarnya barang, *overstocking* dan *understocking*, keterlambatan pengiriman barang dari supplier, kerusakan atau kecacatan pada barang menjadi tidak efisiensi dalam pengelolaan persediaan, dan tidak akuratnya laporan pencatatan persediaan barang sehingga tidak tepatnya pengaturan siklus

pengadaan barang yang berdampak pada biaya pengadaan yang tinggi. Seringkali, proses inventaris produk yang dilakukan menyebabkan ketidaksesuaian antara data jumlah stok barang dengan jumlah barang yang tersedia setiap bulannya.

Penelitian terdahulu yang memiliki kesamaan dengan topik penelitian ini yaitu dilakukan oleh Sangadah & Mutiah pada tahun 2021 yang membuat penelitian berjudul “Penerapan Perhitungan Persediaan Barang Dagang dengan Metode FIFO (Studi Kasus Pada Swalayan Surya Balong Ponogoro)” yang bertujuan untuk mengetahui pengelolaan persediaan barang dagang di Swalayan pada periode tahun 2020 dari bulan Januari sampai dengan Maret 2021. Berikutnya penelitian dari Haryo Kusumo dan Sindhu Rakasiwi pada tahun 2021 dengan judul “*Information System Supply Chain Management with FIFO Perpetual Method.*” membahas tentang memudahkan dan mengontrol nilai persediaan dan biaya persediaan secara real time. Dan terakhir penelitian dari Gita Chendara Langi Madik, Anak Agung, dan Fitri Sukmawati pada tahun 2020 dengan judul “Aplikasi Pembelian dan Pengelolaan Persediaan Bahan Baku Menggunakan Metode Perpetual” yang menjelaskan aplikasi ini dapat digunakan untuk memperkirakan kebutuhan barang, memantau persediaan, dan memperhitungkan biaya pembelian barang.

Oleh karena itu dalam pembuatan aplikasi ini, peneliti menerapkan pencatatan persediaan barang menggunakan metode perpetual. Selain itu, penulis mengharapkan agar aplikasi ini dapat membantu mengelola *management inventory* yang dibutuhkan perusahaan dalam memantau persediaan barang serta membantu karyawan dalam pengerjaan menjadi efektif dan efisien. Pada pengujian Aplikasi *Inventory* ini, peneliti menggunakan metode *Black Box* yakni metode pengujian yang hanya menguji pada fungsionalitas antar muka. Sedangkan masukan dan pengeluaran data dengan pengujian penelitian berupa uji validitas, uji realibilitas, dan analisis perbandingan.

Berdasarkan latar belakang tersebut, penulis tertarik untuk meneliti serta membuat aplikasi persediaan pada perusahaan ini dan menganalisis kesesuaiannya dalam bentuk penulisan ilmiah yang berjudul : “OPTIMASI PENGELOLAAN PERSEDIAAN BARANG MENGGUNAKAN METODE PERPETUAL PADA APLIKASI INVENTORY DI PT. VISI KARYA PRAKARSA.” Penelitian ini mengimplementasikan metode perpetual pada aplikasi inventory di PT. Visi Karya

Prakarsa, sehingga dapat digunakan oleh pihak perusahaan dengan mudah dan dapat menangani masalah-masalah terkait pengelolaan persediaan barang.

1.2 Rumusan Masalah

Dari latar belakang yang telah dikemukakan di atas, maka perumusan masalah dalam penelitian ini adalah sebagai berikut:

1. Bagaimana merancang sistem keluar masuknya barang di PT. Visi Karya Prakarsa?
2. Bagaimana aplikasi yang dirancang dapat menghasilkan laporan persediaan menggunakan metode perpetual secara tepat dan optimal?
3. Bagaimana cara mengatasi tidak tepatnya pengaturan siklus pengadaan barang di PT. Visi Karya Prakarsa?

1.3 Batasan Masalah

Dengan munculnya permasalahan-permasalahan tersebut, muncul sebagai hasil dari proses pembuatan aplikasi inventori baru dan implementasinya untuk menciptakan sistem yang efektif, efisien, dan mampu menghasilkan informasi yang cepat dan akurat dibatasi pada:

1. Penelitian ini hanya membahas persediaan untuk pengontrolan stok barang pada PT. Visi Karya Prakarsa.
2. Sistem ini hanya menggunakan pencatatan perpetual untuk mengelola inventaris dengan sistem periode dan menghasilkan laporan inventaris.
3. Pada sistem ini hanya mencakup persediaan tanpa adanya harga atau nominal yang memuat rata-rata perbulannya.
4. Sistem ini dibangun dengan berbasis web menggunakan *tools composer*, *visual studio code*, dan bahasa pemrograman PHP dengan MySQL yang berfungsi sebagai basis data.

1.4 Tujuan Penelitian

Adapun tujuan yang ingin dicapai dari penelitian ini, adalah :

1. Merancang sistem keluar masuk barang di PT. Visi Karya Prakarsa

2. Merancang aplikasi yang dapat menghasilkan laporan persediaan dengan menggunakan metode perpetual secara tepat dan optimal.
3. Mengatasi ketidak tepatannya pengaturan siklus pengadaan barang di PT. Visi Karya Prakarsa.

1.5 Manfaat Penelitian

1. Untuk memudahkan karyawan dalam pembuatan laporan pencatatan persediaan seperti pemasukan, pengeluaran dan laporan dokumentasi barang.
2. Meningkatkan kinerja perusahaan sehingga kesalahan dan masalah yang sebelumnya terjadi dapat diatasi.
3. Membantu mengelola *management inventory* yang dibutuhkan perusahaan dalam memantau persediaan barang serta membantu karyawan dalam pengerjaan menjadi efektif dan efisien.

1.6 Metodologi Penelitian

Metode penelitian yang digunakan dalam penelitian ini, jenis data yang digunakan yaitu data kuantitatif, sumber data yang dalam penelitian ini adalah data sekunder yang mengacu pada analisis data lapangan yang sesuai fakta di PT. Visi Karya Prakarsa.

1.6.1 Metode Pengumpulan Data

1. Survei

Metode ini melibatkan pengumpulan data melalui kuesioner atau wawancara dengan responden yang relevan dengan topik penelitian. Dalam konteks persediaan barang, metode survei dapat dilakukan dengan mengumpulkan data dari manajer gudang, pengelola inventaris, atau pihak-pihak terkait lainnya.

2. Studi Kasus

Metode ini melibatkan analisis mendalam dari satu kasus atau beberapa kasus yang terkait dengan topik penelitian. Dalam konteks persediaan barang, metode studi kasus dapat dilakukan dengan menganalisis bagaimana suatu perusahaan atau organisasi mengelola prsediaan barang mereka.

3. Observasi

Metode ini melibatkan pengumpulan data dengan mengamati perilaku atau kejadian secara langsung. Dalam konteks persediaan barang, metode observasi dapat dilakukan dengan mengamati aktivitas pengelolaan persediaan barang di suatu gudang atau pusat distribusi.

1.6.2 Metode Pengembangan Sistem

Metode yang digunakan untuk pengembangan perangkat lunak yang dibangun menggunakan metode SDLC atau yang juga dikenal dengan model *waterfall*. Model *waterfall* merupakan model pengembangan sistem informasi yang sistematis dan berurutan.

a. Analisis Kebutuhan

Proses pendefinisian kebutuhan dilakukan untuk menentukan kebutuhan perangkat lunak sehingga pengguna dapat memahami fungsionalitas perangkat lunak yang dibutuhkan pengguna. Pada tahap ini dilakukan pengumpulan data dengan cara melakukan observasi dan wawancara yang dilakukan pada PT. Visi Karya Prakarsa yang berlokasi di Jl. Jupiter Barat.36 No.37, Sekejati, Kec. Buahbatu, Kota Bandung. Proses pengumpulan kebutuhan dilakukan dengan ketat untuk menetapkan kebutuhan perangkat lunak sehingga pengguna dapat memahami perangkat lunak seperti apa yang dibutuhkan pada tahap ini, kebutuhan tersebut harus dicatat.

b. Desain Sistem dan Perangkat Lunak (*System and Software Design*)

Tahap ini mengubah kebutuhan perangkat lunak dari tahap sebelumnya menjadi representasi desain sehingga dapat diimplementasikan sebagai perangkat lunak pada tahap selanjutnya. Pada tahap ini dirancang Data Flow Diagram (DFD), Perancangan Basis Data, hingga perancangan antar muka perangkat lunak.

c. Pengkodean

Desain pada tahap sebelumnya diterjemahkan ke dalam perangkat lunak. Hasil dari fase ini adalah program komputer berdasarkan desain yang dibuat selama fase desain.

- d. Pengujian Sistem (*Integration and System Testing*)
Fungsi utama perangkat lunak harus dapat berjalan secara logika dan fungsional. Harus dapat dipastikan semua bagian diuji. Hal ini dilakukan untuk meminimalkan kesalahan (*error*) dan memastikan *output* yang dihasilkan memenuhi kebutuhan pengguna. Tahap ini diujikan dengan *blackbox testing*.
- e. Pendukung dan Pemeliharaan (*Operation and Maintenance*)
Pada tahap ini, perangkat lunak yang telah di implementasikan dan diujikan sebelumnya diberikan kepada pengguna dan diberikan *maintenance*.

1.7 Sistematika Penulisan

Pada bab ini, penulis membahas sistematika laporan skripsi pada bab ini dengan tujuan untuk memudahkan pembaca dalam memahami isi laporan ini. Penulis menyusun laporan skripsi ini ke dalam beberapa bab yang merupakan satu kesatuan dan saling melengkapi, seperti yang terlihat di bawah ini.

BAB I (PENDAHULUAN)

Bab ini mencakup latar belakang penelitian dan gambaran secara garis besar mengenai kesulitan yang ditemukan oleh peneliti, rumusan masalah, batasan masalah, tujuan penelitian tugas akhir, metode penelitian tugas akhir, dan sistematika penulisan.

BAB II (TINJAUAN PUSTAKA)

Tinjauan literatur ini memberikan teori dan konsep dasar terkait dengan topik yang dibahas, seperti pengertian dari inventory, FIFO, perpetual, laporan penelitian sebelumnya, beberapa modul pada metode RAD dan perangkat pendukung penyebaran perangkat keras lainnya. Penulis sebagai dasar untuk bekerja dengan sistem.

BAB III (METODOLOGI PENELITIAN)

Metode penelitian ini meliputi langkah-langkah pembuatan sistem berupa bahan, metode dan langkah kerja yang penulis lakukan untuk menyelesaikan pekerjaan ini, dijelaskan dengan bantuan sistem dan alat.

BAB IV (ANALISIS, PERANCANGAN DAN HASIL BIDANG INFORMATIKA ANALISIS)

Bab ini membahas tentang metode yang telah ditempuh oleh penulis untuk membuat sistem yang direncanakan berupa rangkaian alat dan fungsi yang tergabung dalam sistem dan rangkaian hasil pengujian sistem.

BAB V (IMPLEMENTASI DAN PENGUJIAN TEKNIK INFORMATIKA)

Bab ini berisi mengenai proses penggabungan pada implementasi sistem, spesifikasi sistem, instalasi sistem dan proses menjalankan sistem yang sudah dirancang dan hasil pengujian pada sistem yang telah dirangkai.

BAB VI (KESIMPULAN DAN SARAN)

Bab ini menarik kesimpulan dari hasil analisis data pada bab-bab sebelumnya. Saran-saran yang diberikan akan sangat bermanfaat karena dapat membantu dalam membuat dan mengembangkan isi laporan sesuai dengan tujuan penelitian skripsi ini.

BAB II

TINJAUAN PUSTAKA

2.1 Landasan Teori

Landasan teori ini berisi ringkasan berdasarkan jurnal terkait penelitian yang diambil. Pada penelitian ini, penulis mempelajari penelitian terdahulu sebagai acuan dalam penelitian, berikut ini adalah jurnal-jurnal yang digunakan dalam penelitian:

Tabel 2.1 Acuan Penelitian

No	Judul Penelitian	Masalah	Metode	Solusi
1	Penerapan Perhitungan Persediaan Barang Dagang Dengan Metode FIFO (Studi Kasus Pada Swalayan Surya Balong Ponorogo)	1. Belum cukup untuk menilai atau mengevaluasi keberhasilan usaha 2. Perlunya pengendalian internal persediaan barang dagang	Metode FIFO	Membantu mengurangi masalah yang menyebabkan rendahnya harga pokok penjualan, sehingga peluang untuk meningkatkan pendapatan menjadi lebih besar.
2	<i>Information System Supply Chain Management with FIFO Perpetual Method</i>	1. Masih menggunakan sistem manual dalam pencatatan persediaan barang	Metode FIFO perpetual	Meningkatkan pengelolaan persediaan dan penyajian laporan dengan mengimplementas

		2. Sering terjadinya permasalahan pada saat pencatatan dan evaluasi persediaan		ikan sistem informasi SCM.
3	Aplikasi Pembelian dan Pengelolaan Persediaan Bahan Baku Menggunakan Metode Perpetual	1. Pencatatan bahan yang masuk dan keluar masih menggunakan kertas manual 2. Tidak ada dokumentasi bukti transaksi dalam perhitungan persediaannya	Metode perpetual	Menyajikan catatan akuntansi dan laporan jurnal, buku besar, laporan penerimaan, laporan pembelian, laporan retur dan kartu stock yang memudahkan pengelolaan persediaan.

2.1.1 Acuan Penelitian Pertama

Berdasarkan tabel 2.1 Acuan Penelitian nomor 1 dengan judul penelitian Penerapan Perhitungan Persediaan Barang Dagang Dengan Metode FIFO (Studi Kasus Pada Swalayan Surya Balong Ponorogo) yang ditulis oleh Lailatus Sangadah dan Nur Sayidatul Muntiah pada Jurnal Manajemen dan Bisnis Fakultas Ekonomi UMP tahun 2021.

Tujuan dari penelitian ini adalah untuk menilai manajemen persediaan barang dagang di Swalayan Surya Balong pada periode 2020 dari bulan Januari hingga Maret 2021. Dengan pendekatan FIFO, yang ditetapkan dari data pembelian dan penjualan persediaan barang dagang, digunakan dalam investigasi ini. Menurut temuan penelitian, dengan menggunakan metode FIFO untuk penilaian persediaan

dapat membantu mengurangi masalah yang terjadi di Swalayan terkait persediaan barang dagang karena dinilai sangat efektif dalam menghasilkan harga pokok penjualan yang rendah dan peluang yang lebih besar untuk meningkatkan laba. Hal ini menunjukkan bahwa, dibandingkan dengan teknik penilaian persediaan yang dilakukan sebelumnya, metode penilaian persediaan yang digunakan tidak tepat, sehingga menimbulkan masalah yang akan merusak pendapatan yang ada.

2.1.2 Acuan Penelitian Kedua

Berdasarkan tabel 2.1 Acuan Penelitian nomor 2 dengan judul *Information System Supply Chain Management with FIFO Perpetual Method* yang ditulis oleh Haryo Kusumo dan Sindhu Rakasiwi pada Jurnal *Advance Sustainable Science, Engineering and Technology* tahun 2021.

Pada penelitian ini, peneliti mengangkat masalah mengenai persediaan yang menggunakan sistem manual. Penelitian ini bertujuan untuk meningkatkan pengelolaan persediaan dan penyajian laporan dengan mengimplementasikan sistem informasi *Supply Chain Management (SCM)* dengan metode Perpetual FIFO di PT. Von Mustika. Sistem ini dibangun dengan menggunakan bahasa pemrograman PHP dan database MySQL yang berfungsi untuk mengatasi permasalahan pengelolaan persediaan dan penyajian laporan pada PT. Von Mustika. Hasil dari penelitian ini adalah pencatatan laporan perusahaan akan lebih detail, laporan akan lebih jelas, efektif dan efisien untuk mengontrol nilai persediaan dan biaya persediaan secara real time sehingga pengambilan keputusan perusahaan dilakukan secara cepat dan akurat serta sistem ini memungkinkan setiap transaksi dicatat dalam nilai persediaan dan biaya persediaan, kemudian akan dimasukkan ke dalam laporan persediaan, neraca dan laba rugi sehingga dapat memudahkan admin dalam pengendalian internal persediaan

2.1.3 Acuan Penelitian Ketiga

Berdasarkan tabel 2.1 Acuan Penelitian nomor 3 dengan judul *Aplikasi Pembelian dan Pengelolaan Persediaan Bahan Baku Menggunakan Metode Perpetual* yang ditulis oleh Gita Chendana Langi Madika¹, Anak Agung Gde Agung², Fitri Sukmawati³ pada Jurnal *e-Proceeding of Applied Science* tahun 2020

Penelitian ini membahas tentang restoran yang bergerak di bidang kuliner yang menghasilkan kerang atau seafood. Tentang restoran yang mengalami kesulitan karena pengelolaan bahan baku yang masih belum teratasi sehingga persediaan lebih cepat habis. Persediaan yang tidak terkontrol menyebabkan sulit untuk mengetahui kapan pihak restoran harus membeli bahan baku kembali. Aplikasi Pembelian dan Pengelolaan Persediaan Bahan Baku pada penelitian ini menggunakan perpetual yang memuat fungsionalitas diantaranya mengelola setoran modal, mengelola pembelian bahan baku, mengelola penerimaan bahan baku, mengelola retur pembelian, serta menyajikan laporan keuangan yang dapat menghasilkan jurnal, buku besar, laporan pembelian, laporan penerimaan, laporan retur pembelian serta kartu stok. Fungsionalitas yang diujikan menggunakan metode *Blackbox Testing*

2.2 Dasar Teori

2.2.1 Persediaan

Persediaan adalah bahan habis pakai (perlengkapan), persediaan barang dagangan, persediaan bahan baku, persediaan bahan penolong, persediaan barang setengah jadi dan persediaan barang jadi. Menurut (PSAK No 14 ; 2014) persediaan adalah aset : Tersedia untuk dijual dalam kegiatan usaha normal. Dalam proses produksi dan atau dalam perjalanan. Dalam bentuk bahan atau perlengkapan (*supplies*) untuk digunakan dalam proses produksi atau pemberian jasa.

Menurut Dewi Lestari, Dr. Subagyo, M.M., Ir. Arthur Daniel Limantara, M.M pada tahun 2019 persediaan adalah bahan atau barang yang disimpan yang akan digunakan untuk memenuhi tujuan tertentu. Setiap perusahaan yang melakukan kegiatan usaha umumnya memiliki persediaan. Keberadaannya tidak saja dianggap sebagai beban (*liability*) karena merupakan pemborosan, tetapi sekaligus juga dapat dianggap sebagai kekayaan (*asset*) yang dapat segera dicairkan dalam bentuk uang tunai (*cash*). Apabila jumlah persediaan terlalu besar (*overstock*) mengakibatkan timbulnya dana menganggur yang besar, juga menimbulkan resiko kerusakan barang yang lebih besar dan biaya penyimpanan yang tinggi. Namun jika persediaan terlalu sedikit mengakibatkan resiko terjadinya kekurangan persediaan (*stockout*) karena seringkali barang tidak dapat didatangkan secara mendadak dan sebesar yang dibutuhkan, yang menyebabkan terhentinya

proses produksi, tertundanya penjualan, bahkan hilangnya pelanggan. (Vol & Persed, 2019)

Berdasarkan definisi sebelumnya, penulis menyimpulkan bahwa persediaan mengacu pada bahan atau barang yang disimpan yang akan digunakan untuk alasan tertentu, seperti pembuatan atau perakitan, penjualan kembali, atau suku cadang untuk suatu peralatan atau mesin. Di sisi lain, frasa persediaan adalah istilah luas yang mengacu pada sumber daya organisasi apa pun yang disimpan untuk mengantisipasi permintaan yang memuaskan, baik permintaan internal maupun eksternal.

2.2.2 Sistem

Menurut Jogiyanto dalam penelitian [Khairuddin Nasution] Sistem merupakan sekumpulan elemen –elemen yang saling berkaitan dan memproses masukan (*input*) sehingga menghasilkan keluaran (*output*)”. Pengertian sistem dibagi menjadi dua pendekatan yaitu dilihat dari pendekatan yang menekankan pada prosedur dan di lihat dari pendekatan yang menekankan pada elemen atau komponen. Pendekatan sistem yang berfokus pada prosedur. Sistem digambarkan sebagai jaringan prosedur - proses yang dihubungkan, dikumpulkan bersama-sama untuk melakukan suatu kegiatan guna mencapai tujuan tertentu. Pendekatan sistem yang menyoroti bagian-bagian atau komponen yang ditentukan oleh Jogianto. (Nasution et al., 2010)

Bentuk umum sebuah sistem terdiri dari masukan (*input*), proses, dan keluaran (*output*); dalam bentuk umum sistem ini, ada satu atau lebih masukan. Dalam bentuk yang paling dasar, sistem ini memiliki satu atau lebih *input* yang diproses dan menghasilkan *output* seperti gambar dibawah ini:



Gambar 2.1 Bentuk umum sistem

2.2.3 Sistem Penilaian Persediaan

Persediaan adalah salah satu jenis aset lancar dalam bisnis. Persediaan didefinisikan sebagai barang dagangan yang disimpan untuk dijual dalam operasi perusahaan yang biasa dilakukan jika perusahaan tersebut adalah perusahaan dagang. Sementara itu, persediaan didefinisikan sebagai bahan baku yang terdapat dalam proses produksi atau yang dimiliki untuk proses produksi jika perusahaan tersebut adalah perusahaan manufaktur.

Untuk menilai persediaan, akuntansi memberikan beberapa metode, salah satunya adalah berdasarkan *cost flow assumption* (arus keluar masuknya harga beli). Terdapat dua metode berdasarkan cost flow assumption ini, yaitu: metode FIFO (*First In First Out*) dan metode *Average*. FIFO merupakan metode dimana barang pertama yang masuk berarti barang tersebutlah yang pertama keluar. Dengan metode FIFO, biaya persediaan dihitung berdasarkan asumsi bahwa barang akan dijual atau dipakai sendiri dan sisa dalam persediaan menunjukkan pembelian atau produksi yang terakhir.

2.2.4 Sistem Pencatatan Persediaan

Berdasarkan penelitian dari [Ratnasari, Indah Nurdiniah, Dade] Dalam pencatatan persediaan dapat digunakan dengan menggunakan sistem pencatatan persediaan. Sistem pencatatan persediaan diimplementasikan untuk mencatat jumlah dan pergerakan masuk dan keluar, serta untuk mengetahui saldo akhir persediaan. Sistem pencatatan persediaan berkaitan dengan proses pengalokasian biaya persediaan. Sebagai nilai persediaan akhir dan beban sebagai harga pokok penjualan (Ratnasari & Nurdiniah, 2022)

Ada dua sistem pencatatan untuk persediaan bahan baku yaitu sistem perpetual dan sistem periodik.

1) Sistem Perpetual

Sistem pencatatan persediaan perpetual adalah pencatatan yang dilakukan pada saat setiap transaksi pembelian dan penggunaan persediaan yang didokumentasikan dalam pembukuan. Menurut [H. Budianto and D. Ferriswara] sistem ini disebut sistem perpetual karena aliran persediaan dapat terus-menerus diketahui setiap saat. Nilai atau harga pokok komoditas yang dijual dan jumlah persediaan pada akhir periode akuntansi dapat diketahui setiap saat dalam sistem

perpetual. Pendekatan ini digunakan untuk mencatat persediaan komoditas dengan jumlah jenis yang terbatas dan harga satuan yang cukup tinggi untuk setiap kategori barang. Buku pembantu persediaan (kartu persediaan) memberikan dukungan untuk estimasi persediaan. Kartu persediaan akan membantu perusahaan menjadi lebih teliti saat menerima, menyimpan, dan mengeluarkan barang. Beberapa ciri-ciri sistem perpetual adalah sebagai berikut:

- a. Pembelian barang dagang, retur pembelian, potongan pembelian, ongkos angkut, dan diskon pembelian barang dagang, akan dicatat ke dalam akun tunggal yaitu persediaan.
- b. Kedua, setiap terjadi penjualan, maka akuntan akan mendebet akun harga pokok penjualan dan mengkredit persediaan.

Salah satu kelebihan sistem pencatatan persediaan metode perpetual yakni perusahaan tidak perlu melakukan perhitungan fisik (*stock opname*) pada stok tersisa. Alasannya, perusahaan dapat mengetahui stok yang sebenarnya di lapangan dengan mudah berkat adanya pencatatan yang dilakukan setiap waktu.

2) Sistem Periodik

Sistem pencatatan persediaan dengan metode periodik merupakan pencatatan yang dilakukan langsung dengan mengecek persediaan setiap akhir periode. Menurut [F. A. Dunia, Wasilah Abdullah, and Catur Sasongko] dalam menetapkan nilai persediaan menggunakan Sistem pencatatan persediaan dengan metode periodik perusahaan harus melakukan perhitungan fisik terhadap persediaannya, agar dapat ditentukan nilai persediaan di akhir periode serta mengetahui biaya atau harga pokok dari persediaan yang dipakai. Berbeda dengan sistem pencatatan perpetual, sistem ini tidak mendebet ataupun mengkredit akun Persediaan saat terjadi transaksi yang berkaitan dengan bertambah atau berkurangnya persediaan. Demikian juga ketika terjadi penjualan barang dagang, akun harga pokok penjualan tidak dicatat, namun akan dihitung pada akhir periode akuntansi. Beberapa ciri-ciri sistem periodik adalah sebagai berikut:

- a. Ketika terjadi pembelian barang dagang, akan didebet ke akun Pembelian. Transaksi retur pembelian akan dikredit ke akun Retur Pembelian.

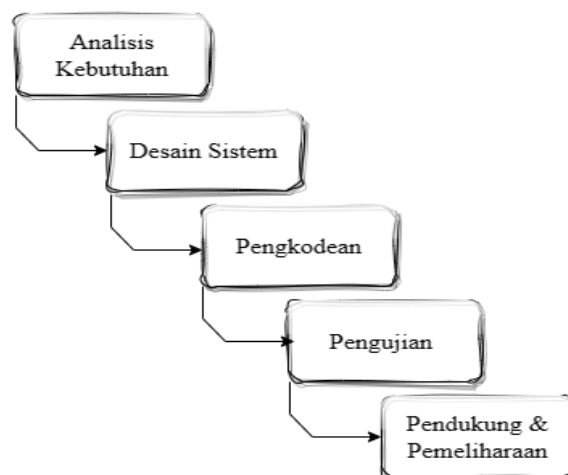
Transaksi diskon pembelian, akan dikredit ke akun Diskon Pembelian, sedangkan saat pembayaran ongkos angkut pembelian, akan dicatat ke sisi debet akun Ongkos Angkut Masuk.

- b. Kedua, saat terjadi penjualan barang dagangan, maka akun kas atau piutang akan didebet dan akun penjualan dicatat di sisi kredit. Perusahaan yang menjual barang dagangan dalam kuantitas banyak, frekuensi yang tinggi, nilainya relatif kecil dan harga jual-beli yang relatif stabil, adalah yang memakai sistem pencatatan persediaan metode periodik.

Kelebihan metode periodik yakni perusahaan selalu dapat mengetahui besarnya persediaan dalam gudang secara akurat karena sering melakukan stock opname. Namun, sistem pencatatan ini mempunyai kelemahan, yaitu sulit mengetahui jumlah persediaan awal dan akhir pada satu periode.

2.2.5 Metode SDLC Waterfall

(Qadafi & Wahyudi, 2020) menyatakan bahwa *System Development Life Cycle* (SDLC) adalah proses menciptakan atau memperbaiki sistem dengan menggunakan model dan prosedur yang sebelumnya telah digunakan untuk menciptakan sistem lain. Sementara itu, model *waterfall* juga dikenal sebagai model siklus hidup *sekuensial linier* atau konvensional. Jadi, jika langkah pertama tidak diselesaikan, maka tidak akan dapat menyelesaikan langkah kedua, ketiga, dan seterusnya. Langkah 3 hanya dapat dilakukan jika langkah 1 dan 2 telah diselesaikan. Model *waterfall* strategi siklus hidup sistem yang berurutan atau terorganisir yang dimulai dengan:



Gambar 2.2 Tahapan metode waterfall SDLC

1. Analisis Kebutuhan

Pada tahap ini, penulis mengidentifikasi masalah di PT. Visi Karya Prakarsa yang berkaitan dengan persediaan barang perusahaan, seperti halnya permasalahan yang masih menggunakan pendataan secara manual dengan pencatatan di buku besar, setiap barang yang ada di perusahaan belum memiliki nomor aset, informasi mengenai kondisi barang, serta proses peminjaman dan pengembalian barang masih menggunakan sistem pencatatan buku besar secara manual. Setelah tahap identifikasi masalah selesai dilakukan, maka langkah selanjutnya adalah membuat kesepakatan waktu kerja dengan pihak pengguna yaitu dari bagian inventori PT. Visi Karya Prakarsa. Proses pengumpulan kebutuhan dilakukan dengan ketat untuk menetapkan kebutuhan perangkat lunak sehingga pengguna dapat memahami perangkat lunak seperti apa yang dibutuhkan; pada tahap ini, kebutuhan tersebut harus dicatat.

2. Desain Sistem dan Perangkat Lunak (*System and Software Design*)

Teknik multi-langkah untuk merancang aplikasi perangkat lunak yang mencakup struktur data, arsitektur perangkat lunak, representasi antarmuka, dan prosedur pengkodean.

3. Pengkodean

Desain harus dikonversi ke dalam perangkat lunak; hasil akhirnya adalah program komputer berdasarkan desain yang dibuat pada langkah sebelumnya.

4. Pengujian Sistem (*Integration and System Testing*)

Pengujian berfokus pada program dalam hal logika dan fungsionalitas untuk menjamin bahwa semua bagian telah diuji untuk menghilangkan kesalahan dan memastikan output sesuai dengan yang direncanakan. (Syafarina et al., 2016)

5. Pendukung dan Pemeliharaan (*Operation and Maintenance*)

Tidak menutup kemungkinan sebuah perangkat lunak mengalami perubahan setelah dikirimkan ke pengguna, misalnya karena adanya kesalahan yang muncul dan tidak terdeteksi pada saat pengujian, atau perangkat lunak harus beradaptasi dengan lingkungan yang baru. Tahap dukungan dan pemeliharaan dapat mengulangi proses pengembangan yang dimulai dengan

analisis spesifikasi untuk perubahan perangkat lunak yang sudah ada, tetapi tidak untuk membuat program baru. (Agus Heryanto, Hilmi Fuad, 2014)


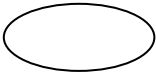
2.2.6 *Unified Modelling Language (UML)*





Menurut Qadafi, A. Wahyudi pada penelitiannya di tahun 2020 *Unified Modeling Language (UML)* adalah bahasa untuk menggambarkan, memvisualisasikan, membangun, dan mendokumentasikan artefak (data yang digunakan atau dibuat selama proses pengembangan perangkat lunak). UML adalah notasi yang komprehensif untuk memvisualisasikan model sistem. Meskipun sistem berisi informasi dan fungsi, namun paling sering digunakan untuk memodelkan sistem komputer. UML digunakan di hampir semua profesi yang membutuhkan pemodelan, tidak hanya pengembangan perangkat lunak (Qadafi & Wahyudi, 2020)

1. *Use Case Diagram*

Menurut Maria, S.dan Listiana pada tahun 2019. “Ketika memodelkan pemrograman perangkat lunak berorientasi objek, Diagram Kasus Penggunaan harus dibuat pada awalnya. Tindakan aktor akan dijelaskan dalam *Use Case Diagram*. Yang disebut aktor dalam hal ini adalah pengguna aplikasi, mirip dengan pengembangan perangkat lunak terstruktur saat membangun DFD. Istilah “*Use Case*” berhubungan dengan langkah sebelumnya yaitu menentukan kebutuhan Rekayasa Perangkat Lunak.”

Tabel 2.2 Use Case Diagram

No	Symbol	Nama	Keterangan
1		<i>Actor</i>	Menentukan peran yang akan diambil oleh pengguna saat terlibat dengan kasus penggunaan.
2		<i>Use Case</i>	Deskripsi urutan kejadian sistem yang menghasilkan hasil yang dapat diukur bagi seorang aktor.


3		<i>Association</i>	Yang menggabungkan dua objek menjadi satu.
4		<i>Extend</i>	Pada saat tertentu, menentukan bahwa kasus penggunaan target memperluas perilaku kasus penggunaan sumber.
5		<i>Generalization</i>	Objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dengan objek di atasnya (<i>ancestor</i>).
6		<i>Include</i>	Menentukan sumber kasus penggunaan secara langsung.


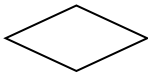


2. Activity Diagram

Menurut Sukrianto, D. dan Agustina, S. pada tahun 2018 mengilustrasikan cara kerja diagram aktivitas Diagram ini berfungsi mirip dengan diagram alir dalam beberapa hal, namun perbedaan utama antara diagram ini dan notasi diagram alir adalah bahwa diagram ini mengizinkan tindakan paralel. Diagram aktivitas juga biasa digunakan untuk menggambarkan konsep-konsep berikut:

- Setiap rangkaian tindakan yang digambarkan dalam desain proses bisnis adalah proses bisnis sistem yang ditentukan.
- Pengaturan atau pengelompokan tampilan antarmuka sistem/pengguna di mana setiap tindakan dianggap memiliki desain antarmuka tampilan.
- Desain pengujian di mana setiap aktivitas dianggap membutuhkan pengujian yang kasus pengujiannya harus ditentukan.
- Desain menu yang muncul pada perangkat lunak.

Tabel 2.3 Activity Diagram

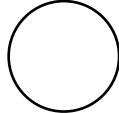


No	Simbol	Nama	Keterangan
1.		Status awal	Aktivitas sistem sering kali diawali dengan kata kerja.

2		Aktivitas	Deskripsi urutan kejadian sistem yang menghasilkan hasil yang dapat diukur bagi seorang aktor.
3		Percabangan	Jika ada lebih dari satu pilihan aktivitas, maka akan ada hubungan percabangan.
4		Penggabungan	Sebuah organisasi penggabungan yang menggabungkan banyak aktivitas menjadi satu.
5		Status akhir	Diagram memiliki kondisi akhir yang merepresentasikan kinerja sistem.

3. *Class Diagram*

Menurut Sukrianto, D. dan Agustina, S. pada tahun 2018 *Class diagram* adalah representasi logis dari kelas-kelas yang ada di dalam sistem dan hubungannya. Dalam konteks pengembangan perangkat lunak berorientasi objek, class diagram mempresentasikan struktur statis dari sistem dengan menggambarkan kelas-kelas yang ada dalam sistem tersebut beserta atribut-atribut yang menjadi properti kelas-kelas tersebut serta operasi-operasi atau metode-metode yang mewakili tindakan yang dapat dilakukan oleh setiap kelas. Atribut adalah variabel yang khusus dimiliki oleh sebuah kelas untuk menyimpan informasi yang relevan dengan kelas tersebut, sedangkan operasi atau metode adalah fungsi-fungsi yang dimiliki oleh suatu kelas untuk melakukan tindakan-tindakan tertentu, yang dapat mengubah atribut-atribut atau menghasilkan output yang diperlukan dalam konteks sistem yang direpresentasikan oleh class diagram.

Tabel 2.4 Class Diagram

No	Simbol	Nama	Keterangan
1.	<div style="border: 1px solid black; padding: 5px; margin: 5px;"> <div style="border: 1px solid black; padding: 2px; margin-bottom: 2px;">Nama_kelas</div> <div style="border: 1px solid black; padding: 2px; margin-bottom: 2px;">+ atribut</div> <div style="border: 1px solid black; padding: 2px;">+ operasi</div> </div>	Kelas	Kelas pada struktur sistem
2		Antarmuka/ <i>interface</i>	Dalam pemrograman berorientasi objek, hal ini dianalogikan dengan gagasan antarmuka.
3		Asosiasi	Hubungan kelas dengan makna afiliasi yang luas sering kali disertai dengan <i>multiplicity</i> .
4		Asosiasi berarah	Hubungan antar kelas di mana satu kelas memanfaatkan kelas lainnya.

2.2.7 Balsamiq Mockup

Balsamiq adalah salah satu *tools* yang digunakan para UI *designer* maupun UX *designer* untuk merancang desain tampilan yang akan dibuat. Aplikasi ini dianggap cukup ramah pemula karena tidak membutuhkan kode untuk bisa mengoperasikannya. Penggunaan *tools* pada aplikasi ini cukup mendrag dan drop elemen-elemen desain yang diperlukan sehingga proses desain pun bisa menjadi lebih cepat. Balsamiq juga bisa digunakan untuk membuat prototype interaktif untuk situs atau aplikasi yang sedang dirancang. Fitur ini membuat desainer bisa mengetahui *usability* rancangannya sehingga revisi bisa dikurangi pada tahap-tahap selanjutnya. (Menggunakan & Dan, 2019)

2.2.8 Bootstrap versi 4.7

Bootstrap merupakan suatu kerangka kerja (framework) berbasis HTML, CSS, dan JavaScript yang memfasilitasi proses pengembangan situs web responsif dengan efisiensi. Bootstrap menawarkan sekumpulan komponen antarmuka dasar yang telah dipredesain untuk menghadirkan tampilan yang estetis, bersih, dan ringan.

Di samping itu, Bootstrap menyediakan kemampuan grid yang mempermudah pengaturan tata letak dengan singkat dan mudah. (Rencana et al., 2022)

2.2.9 XAMPP versi 7.4

XAMPP adalah singkatan dari X (salah satu dari empat sistem operasi). Ini adalah perangkat lunak gratis yang mendukung banyak sistem operasi dan merupakan kumpulan dari banyak aplikasi. Apache, MYSQL, PHPP, dan Perl adalah beberapa contohnya. Xampp adalah sebuah utilitas yang membundel beberapa produk perangkat lunak ke dalam satu paket. Sudah ada Apache (Web Server), Mysql (*Database*), PHP (*server side scripting*), Perl, FTP server, *PhpMyAdmin*, dan beberapa *library* tambahan lainnya yang sudah termasuk di dalam paket Xampp. (Siregar, 2018)

2.2.10 MySQL

MySQL merupakan sebuah sistem manajemen basis data relasional (RDBMS) yang bersifat open-source. Dalam konteks ilmiah, MySQL dapat diartikan sebagai sebuah perangkat lunak yang digunakan untuk mengelola dan menyimpan data dalam bentuk tabel yang terhubung melalui relasi atau hubungan. RDBMS seperti MySQL memungkinkan pengguna untuk mengatur, menyimpan, mengambil, dan memanipulasi data dengan menggunakan bahasa query seperti SQL (Structured Query Language). MySQL banyak digunakan dalam pengembangan perangkat lunak, aplikasi web, dan sistem informasi untuk menyimpan dan mengelola data secara efisien. (Safitri & Nirmala, 2019)

2.2.11 Visual Studio Code

Visual Studio Code merupakan sebuah aplikasi editor teks gratis yang dikembangkan oleh Microsoft. Aplikasi ini dirancang untuk mendukung berbagai bahasa pemrograman tanpa memerlukan perubahan aplikasi editor, dan dapat dijalankan pada berbagai sistem operasi, termasuk Windows, Linux, dan macOS. Keunggulan utama Visual Studio Code adalah kemampuannya untuk mempermudah programmer saat berpindah dari satu bahasa pemrograman ke yang lain tanpa memerlukan perubahan aplikasi editor, serta memahami dan mengkonfigurasi alat bantu (*tools*) yang diperlukan dalam lingkungan pengembangan bahasa pemrograman yang baru. Aplikasi ini juga memberikan

kebebasan kepada penggunanya untuk menyesuaikan tema, menggunakan debugger, mengintegrasikan ekstensi (*extension*), dan menyesuaikan berbagai fitur lainnya sesuai dengan kebutuhan mereka. (Ferdinandus et al., n.d.)

2.2.12 PHP versi 7.4.5

PHP (*Hypertext Preprocessor*) adalah singkatan berulang yang pada awalnya merujuk kepada '*Personal Home Page*.' Diciptakan pertama kali oleh Rasmus Lerdorf pada tahun 1995, PHP adalah bahasa pemrograman skrip yang dapat disisipkan ke dalam kode HTML. PHP telah merasionalisasi singkatannya menjadi '*PHP: Hypertext Preprocessor*.' Bahasa pemrograman ini memiliki kemampuan yang mumpuni dalam mengembangkan situs web dinamis dan memungkinkan para pengembang untuk membangun sistem manajemen konten (*Content Management System, CMS*). Untuk menjalankan PHP, diperlukan infrastruktur berupa Web Server seperti Apache, PHP Interpreter, dan seringkali MySQL sebagai sistem basis data yang mendukung. (Ferdinandus et al., n.d.)

2.2.13 Cascading Style Sheet (CSS)

CSS (*Cascading Style Sheet*) adalah bahasa *style sheet* yang mengatur format tampilan halaman web yang ditulis dengan penanda markah songage. CSS sering digunakan untuk membuat halaman HTML dan XHTML, tetapi sekarang dapat digunakan untuk dokumen XML apa pun, termasuk SVG, XUL, dan bahkan Android. (Safitri & Nirmala, 2019)

2.2.14 Hypertext Markup Language (HTML)

Menurut Hidayatullah dan Kawistara pada tahun 2015 dalam jurnal Fitri Ayu dan Nia Permata Sari (ISSN:2549-0222) "*Hypertext Merkup Languange (HTML)* adalah bahasa standard yang digunakan untuk menampilkan halaman web". Yang bisa dilakukan dengan HTML yaitu:

1. Mengatur tampilan dari halaman web dan isinya.
2. Membuat tabel dalam halaman web.
3. Mempublikasikan halam web secara *online*.
4. Membuat form yang bisa digunakan untuk menangani registrasi dan transaksi via web.

Contoh: Setiap dokumen HTML diawali dan diakhiri dengan tag HTML.

2.2.15 Metode pengujian *Black Box Testing*

Menurut Oky Irnawati, *Black Box Testing* adalah bentuk pengujian desain yang berfokus pada apakah unit program yang ditentukan dalam spesifikasi terpenuhi atau tidak. Teknik pengujian pada *black box testing* hanya dilakukan dengan menjalankan atau mengeksekusi sebuah unit atau modul, kemudian memonitor apakah hasil dari unit tersebut sesuai dengan proses bisnis yang diantisipasi. Pengujian yang dilakukan memonitor hasil eksekusi data uji, memeriksa operasi program, dan *black box* hanya melihat tampilan luar (*interface*) saja; dengan kata lain tidak perlu mengetahui cara kerja internal. (- AMIK BSI Bekasi & - AMIK BSI Bekasi, 2018)

2.2.16 Metode pengujian penelitian

Ada beberapa metode pengujian yang dapat digunakan untuk menguji hasil penelitian tentang persediaan barang, di antaranya:

1. Uji Validitas

Metode ini digunakan untuk mengukur seberapa validnya instrumen penelitian yang digunakan dalam penelitian. Ada beberapa jenis uji validitas, antara lain uji validitas isi, uji validitas konstruk, dan uji validitas kriteria.

2. Uji Reliabilitas

Metode ini digunakan untuk mengukur seberapa konsisten hasil pengukuran yang diperoleh dari instrumen penelitian. Ada beberapa jenis uji reliabilitas, antara lain uji reliabilitas internal, uji reliabilitas eksternal, dan uji reliabilitas *test-retest* berupa kuisioner.

Pengujian reliabilitas pada kuisioner mengukur sejauh mana alat ukur tersebut konsisten dalam mengukur apa yang seharusnya diukur. Salah satu metode yang umum digunakan untuk mengukur reliabilitas pada kuisioner adalah menggunakan uji Cronbach's Alpha. Uji Cronbach's Alpha mengukur sejauh mana item-item dalam kuisioner saling berkaitan atau konsisten dalam mengukur konstruk yang sama. Nilai Cronbach's Alpha berkisar antara 0 hingga 1, dan semakin tinggi nilai tersebut, semakin tinggi reliabilitas kuisioner.

Proses pengujian reliabilitas kuisioner dengan uji Cronbach's Alpha melibatkan pengumpulan data dari sejumlah responden, kemudian dilakukan analisis statistik terhadap data tersebut. Hasilnya adalah nilai Cronbach's Alpha, yang dapat memberikan gambaran seberapa konsisten kuisioner dalam mengukur variabel yang dituju.

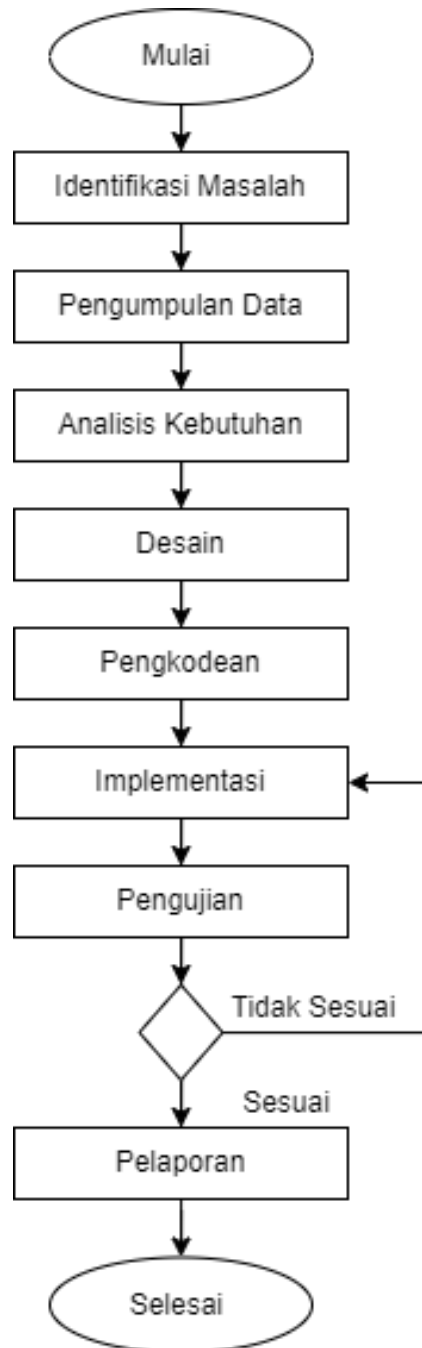
3. Analisis Perbandingan

Metode ini digunakan untuk membandingkan hasil penelitian dengan penelitian sebelumnya atau dengan standar yang telah ditetapkan. Analisis perbandingan dapat membantu dalam memperoleh pemahaman yang lebih baik tentang hasil penelitian dan mengevaluasi keefektifan dari strategi pengelolaan persediaan barang yang telah diimplementasikan.

BAB III

METODOLOGI PENELITIAN

3.1 Kerangka Pikir



Gambar 3.1 Kerangka pikir

3.2 Deskripsi

Deskripsi adalah suatu kaidah yang berkaitan dengan upaya mengubah data menjadi sesuatu yang dapat diartikulasikan dan dipresentasikan dengan cara yang jelas dan tepat untuk mencapai tujuan tertentu, sehingga dapat dipahami dan dimengerti oleh pembaca.

3.2.1 Identifikasi Masalah

Tahap pertama dalam pembuatan aplikasi *inventory* yang ada di kerangka pikir penulis ialah mencari permasalahan atau rumusan masalah. Dalam hal ini penulis meminta izin kepada pihak perusahaan untuk melakukan penelitian kemudian melakukan observasi dan wawancara agar bisa mengetahui tentang masalah yang ada di PT. Visi Karya Prakarsa sehingga penulis bisa membuat aplikasi tersebut. Beberapa masalah yang terdapat pada perusahaan yaitu pencatatan barang masih menggunakan sistem manual dengan beberapa masalah seperti terjadinya kesalahan input keluarnya barang, *overstocking* dan *understocking*, keterlambatan pengiriman barang dari *supplier*, kerusakan atau kecacatan pada barang menjadi tidak efisiensi dalam pengelolaan persediaan, dan tidak akuratnya laporan pencatatan persediaan barang sehingga tidak tepatnya pengaturan siklus pengadaan barang yang berdampak pada biaya pengadaan yang tinggi. Sehingga proses pencatatan persediaan barang yang dilakukan mengakibatkan selisih dari data jumlah stok barang dengan jumlah barang fisik yang ada setiap bulannya.

3.2.2 Pengumpulan Data

1. Survei

Survei merupakan proses akuisisi data melalui instrumen kuesioner yang terstruktur atau melalui interaksi wawancara dengan responden yang memiliki relevansi dengan fokus penelitian. Dalam kerangka penelitian terkait persediaan barang di PT. Visi Karya Prakarsa, pendekatan survei diimplementasikan melalui pengumpulan data yang diperoleh dari narasumber yang memiliki kompetensi di bidang manajemen gudang, inventaris, serta stakeholder lain yang berkaitan. Metode survei ini memungkinkan pengumpulan informasi yang rinci dan mendalam mengenai

aspek-aspek tertentu yang relevan dengan penelitian mengenai kelola persediaan barang dalam perusahaan tersebut.

2. Observasi

Metode ini mendasarkan diri pada proses pengumpulan data dengan melaksanakan pengamatan langsung terhadap pelaksanaan operasional yang sedang berjalan di lingkungan PT. Visi Karya Prakarsa. Dalam konteks pengelolaan persediaan barang, pendekatan observasi diterapkan dengan menyelenggarakan pemantauan dan pencatatan langsung terhadap aktivitas-aktivitas yang berlangsung di gudang atau titik distribusi pusat perusahaan. Metode observasi ini memberikan kesempatan untuk memperoleh wawasan mendalam tentang praktik-praktik yang dijalankan secara riil, memungkinkan analisis detail terhadap aspek-aspek yang relevan dengan pengelolaan persediaan.

3. Studi Kasus

Studi kasus dalam penelitian ini melibatkan analisis mendalam dari suatu kasus atau beberapa kasus yang terkait dengan topik penelitian. Dalam penelitian ini penulis menggunakan 3 jurnal yang terdiri dari Penerapan Perhitungan Persediaan Barang Dagang dengan Metode FIFO (Studi Kasus Pada Swalayan Surya Balong Ponorogo, Aplikasi Pembelian dan Pengelolaan Bahan Baku Menggunakan Metode Perpetual serta *Information System Supply Chain Management with Perpetual Method*.

3.2.3 Pengembangan Sistem

Implementasi metodologi Siklus Hidup Pengembangan Perangkat Lunak (*Software Development Life Cycle, SDLC*) model air terjun (*waterfall*) pada proses pengembangan aplikasi inventaris dilakukan dengan menjalani serangkaian tahap yang dijalankan secara berurutan, sebagaimana tertera berikut ini:

1. Analisis Kebutuhan

Fase ini melibatkan perencanaan menyeluruh dari proyek, termasuk alokasi sumber daya, penjadwalan, dan estimasi anggaran yang diperlukan. Penulis juga diharapkan untuk menyusun rencana rinci untuk pengembangan aplikasi inventaris, yang mencakup langkah-langkah pengembangan, identifikasi tim

yang terlibat, dan penentuan tanggung jawab individu di dalam tim. Setelah data yang diperlukan terkumpul, langkah berikutnya adalah merencanakan dan merancang aplikasi inventaris. Tahap ini dimulai dengan analisis mendalam terhadap kebutuhan perangkat lunak dan keras yang akan digunakan dalam perancangan aplikasi persediaan barang.

2. Desain Sistem

Setelah menyelesaikan analisis kebutuhan, tahap desain dilaksanakan untuk merancang arsitektur sistem dan antarmuka pengguna aplikasi persediaan. Langkah ini mencakup perencanaan basis data, perancangan struktur program, dan merancang antarmuka pengguna (UI) yang intuitif dan user-friendly, dengan menggunakan alat desain seperti Balsamiq.

Sistem desain diarahkan pada pengembangan skema basis data yang tepat untuk menyimpan dan mengelola data inventaris. Hal ini melibatkan identifikasi tabel, kolom, serta hubungan antara entitas yang ada dalam aplikasi. Selain itu, struktur program juga dirancang untuk memastikan komponen-komponen perangkat lunak berinteraksi secara efektif, termasuk bagaimana proses bisnis akan diimplementasikan dalam kode..

Bagian yang sangat penting dari tahap desain adalah merancang antarmuka pengguna (UI) yang dapat dipahami dengan mudah oleh pengguna akhir. Dalam konteks ini, alat desain seperti Balsamiq digunakan untuk membuat tampilan kasar (wireframe) yang memberikan gambaran visual awal tentang tata letak dan elemen-elemen antarmuka. Tujuan utamanya adalah menciptakan antarmuka yang ramah pengguna, intuitif, dan memenuhi kebutuhan fungsional pengguna aplikasi.

Dengan melakukan proses ini secara cermat, tahap desain dapat menghasilkan panduan yang kuat untuk mengembangkan aplikasi persediaan barang yang kokoh dan sesuai dengan kebutuhan yang telah diidentifikasi sebelumnya.

3. Pengkodean

Tahap ini melibatkan aspek perencanaan komprehensif dari proyek secara menyeluruh, yang mencakup identifikasi dan alokasi sumber daya yang diperlukan, penjadwalan kegiatan, serta estimasi anggaran yang relevan. Selain

itu, dalam tahap ini penulis juga diharapkan untuk merumuskan rencana detail terkait pengembangan aplikasi inventaris. Rencana ini mencakup berbagai tahapan dalam proses pengembangan serta aspek perancangan yang melibatkan::

a. Perancangan Diagram

Pada tahap ini dilakukan dengan membuat desain menggunakan bahasa pemodelan *Unified Modelling Language* (UML). UML yang digunakan dalam penelitian ini terdiri dari:

1. *Use case diagram* digunakan untuk menggambarkan secara umum hubungan antara aktor (*actor*) dan kasus penggunaan (*use case*) dalam suatu sistem aplikasi. Diagram ini membantu dalam pemahaman tentang interaksi antara aktor (entitas yang berinteraksi dengan sistem) dan berbagai fungsi (*use case*) yang ada dalam sistem tersebut. Dalam konteks penelitian ini, terdapat satu aktor yaitu "Admin" dan enam kasus penggunaan (*use case*) yang mencakup:
 - a) *Dashboard*: Menampilkan tampilan umum yang memberikan ringkasan data dan informasi penting kepada Admin mengenai kinerja dan status sistem.
 - b) *Supplier*: Mengelola informasi dan interaksi terkait dengan pemasok barang, seperti penambahan, penghapusan, atau perubahan data supplier.
 - c) *Data Barang*: Melibatkan kegiatan pengelolaan data barang, termasuk penambahan, penghapusan, atau perubahan informasi terkait barang yang ada dalam sistem.
 - d) *Riwayat Barang Masuk dan Keluar*: Mencatat serta menampilkan riwayat barang yang masuk dan keluar dari sistem, memberikan jejak transaksi terkait persediaan.
 - e) *Transaksi Laporan*: Menyediakan fungsionalitas untuk menghasilkan laporan transaksi, seperti laporan penjualan, penerimaan, dan sejenisnya.

- f) *User Management*: Mengelola pengaturan dan izin pengguna dalam sistem, termasuk penambahan, penghapusan, dan pengelolaan hak akses.

Dalam diagram *Use Case*, aktor "Admin" akan berinteraksi dengan masing-masing kasus penggunaan (*use case*) sesuai dengan fungsinya. Ini memberikan pandangan yang lebih jelas tentang bagaimana Admin akan menggunakan sistem dan fungsi apa yang tersedia untuknya.

2. *Activity diagram* digunakan untuk memodelkan rangkaian aktivitas dalam suatu sistem, mencakup representasi visual dari urutan aktivitas yang dilakukan oleh pengguna atau entitas lain dalam aplikasi. Diagram ini memberikan pandangan keseluruhan tentang bagaimana aktivitas berlangsung, berinteraksi, dan mengalir dalam sistem. Dalam konteks yang Anda berikan, penulis akan membuat beberapa diagram aktivitas yang berkaitan dengan sistem aplikasi. Di antaranya:

- Proses akses *login*
- Proses akses menu *dashboard*
- Proses akses menu *supplier*
- Proses akses menu data barang
- Proses akses menu riwayat barang masuk
- Proses akses menu riwayat barang keluar
- Proses akses menu laporan
- Proses akses menu *user management*

3. *Class Diagram*, untuk menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Class diagram yang akan dibuat adalah sebagai berikut:

- *Class diagram* admin
- *Class diagram* supplier
- *Class diagram* barang masuk
- *Class diagram* barang keluar
- *Class diagram* satuan barang
- *Class diagram* jenis barang

- *Class diagram* data barang

b. Perancangan *Interface*

Pada tahapan perancangan antarmuka, desain aplikasi dibuat sederhana menggunakan figma mockup namun dapat mengilustrasikan aplikasi yang dibangun dan tentunya memerhatikan interaksi yang akan terjadi dalam aplikasi antara user dan sistem.

4. Pengujian

Setelah melakukan perancangan diagram dan antarmuka, tahap selanjutnya adalah melakukan implementasi perancangan yang telah dibuat selanjutnya dengan cara menerjemahkan kedalam bentuk pengkodean secara nyata memanfaatkan bahasa pemrograman PHP, HTML, *sublime text* serta *framework Bootstrap*.

Sedangkan pada tahap pengujian yang telah dibuat untuk memastikan apakah hasil sesuai dengan rancangan yang diharapkan atau belum. Apabila masih ada kekurangan maka kembali ke tahap implementasi untuk diperbaiki sampai benar-benar sesuai dengan rancangan pengujian pada sistem digunakan metode pengujian *Black Box Testing*

5. Pemeliharaan

Tahap pemeliharaan dan pendukung aplikasi inventaris setelah tahap pengujian. Dalam tahap ini, dilakukan pemantauan kontinu terhadap kinerja aplikasi, penanganan serta mitigasi permasalahan yang muncul, dan pelaksanaan penyempurnaan atau koreksi aplikasi bila dianggap perlu. Tindakan ini merupakan wujud kesinambungan dalam rangka memastikan integritas dan kinerja optimal dari aplikasi, serta merespons perubahan kebutuhan dan dinamika yang dapat memengaruhi fungsionalitas aplikasi inventaris.

3.2.4 Pelaporan

Tahap akhir dari perjalanan adalah penyusunan laporan skripsi, yang merupakan bagian esensial dalam persyaratan kelulusan, dan penyusunannya diarahkan sesuai dengan pedoman yang diuraikan dalam panduan Penulisan Skripsi Fakultas Teknologi Informasi Universitas Bandung. Laporan skripsi ini adalah hasil akhir dari upaya penelitian yang mencerminkan kesesuaian dengan standar ilmiah dan kepatuhan terhadap ketentuan yang ditetapkan oleh panduan Penulisan Skripsi.

BAB IV

ANALISIS DAN PERANCANGAN

4.1 Analisis

Analisis dilakukan sebagai langkah awal penelitian untuk mengetahui kebutuhan-kebutuhan yang diperlukan dalam perancangan aplikasi. Berikut adalah analisa program yang telah penulis lakukan guna membantu dalam perancangan aplikasi *inventory*. Pada analisis ini akan membahas mengenai analisis masalah, analisis *software*, analisis penggunaa, *user interface*, fitur-fitur, analisis serta analisis biaya.

4.1.1 Analisis Masalah

Langkah pertama yaitu menganalisis sistem yang sedang berjalan dengan tujuan untuk mengetahui lebih jelas bagaimana cara kerja sistem tersebut dan masalah yang dihadapi sistem untuk dijalankan landasan usulan pembuatan aplikasi.

Berdasarkan analisis langsung pada sistem kerja laporan *inventory* yang sedang berjalan di PT. Visi Karya Prakarsa ditemukan beberapa kendala yaitu sistem laporan *inventory* ini belum berjalan sesuai dengan sebagaimana mestinya, dimana yang seharusnya laporan keluar masuknya barang harus efektif dan tercatat setiap harinya untuk melakukan perbandingan pemasukan komponen dari segi pembuatan barang yang ada di PT. Visi Karya Prakarsa selain itu pencatatan barang komponen juga belum dapat diakses secara sistem sehingga berdampak pada validasi pencatatan barang yang kurang akurat.

4.1.2 Analisis Software

Berdasarkan analisis kebutuhan maka untuk memenuhi kebutuhan dalam pembuatan aplikasi *inventory* memerlukan software sebagai alat penunjang pembuatan aplikasi tersebut. Adapun software yang dibutuhkan sebagai alat penunjang pada penelitian ini adalah sebagai berikut :

1. Sistem Operasi Windows 10
2. *Framework Bootstrap* digunakan untuk mempercepat dan mempermudah pembuatan tampilan website agar bisa dibuka secara responsive sehingga dapat mendukung untuk segala jenis resolusi, baik itu tablet, smartphone ataupun juga PC dan laptop
3. *Visual Studio Code* sebagai *text editor* yang digunakan untuk melakukan pengkodean dalam pembuatan aplikasi *inventory*
4. XAMPP digunakan untuk web server
5. *Web Browser* digunakan untuk melihat halaman web aplikasi *inventory* yang sudah dibuat
6. Draw Io digunakan untuk membuat/merancang diagram UML
7. Balsamiq Mockup digunakan untuk membuat rancangan antarmuka aplikasi *inventory* yang akan dibuat

4.1.3 Analisis Pengguna

Penganalisaan pengguna adalah berkaitan dengan yang akan memakai aplikasi *inventory* ini. Pengguna aplikasi *inventory* ini adalah karyawan yang ada di bagian gudang, karena pengguna aplikasi *inventory* diharuskan mengetahui hal-hal yang bersangkutan dengan data pemasukan dan pengeluaran barang sehingga dapat menggunakan aplikasi dengan baik sebagaimana fungsinya. Agar dapat mewujudkan tujuan dari dibuatnya aplikasi *inventory* yang bisa memberikan kemudahan dan kenyamanan bagi penggunanya untuk menghasilkan laporan *inventory* yang akurat dan tepat. Maka strategi yang akan diterapkan dalam aplikasi *inventory* ini adalah dilihat dari sisi *user interface* dan fitur-fitur yang diterapkan dalam aplikasi *inventory* tersebut.

4.1.4 User Interface

User interface dari aplikasi *inventory* sangat berpengaruh pada pengalaman dan kenyamanan user saat menggunakannya. Hal ini mencakup perangkat yang dibuat ditujukan untuk menghasilkan informasi yang akurat dan tepat, maka *user interface* juga harus menyesuaikan dengan kebutuhan *user* berupa penyajian laporan *inventory* yang mudah dipahami. Maksud dan tujuan memberikan tampilan

tersebut agar aplikasi dapat menjadi sarana informasi yang memuat laporan untuk membantu user mengetahui tingkat kinerja perusahaan dari segi pembuangan dengan mudah.

Berikut adalah *user interface* aplikasi *inventory* menggunakan metode perpetual sebagai pengukur kinerja perusahaan di PT. Visi Karya Prakarsa, meliputi:

1. *User interface* halaman *Login*
2. *User interface* halaman *Dashboard*
3. *User interface* halaman *Supplier*
4. *User interface* halaman Data Barang
5. *User interface* halaman Transaksi Barang Masuk
6. *User interface* halaman Transaksi Barang Keluar
7. *User interface* halaman Laporan
8. *User interface* halaman *User Management*

4.1.5 Fitur-Fitur

Fitur-fitur digunakan dalam aplikasi *inventory* ini dimaksudkan agar user dapat dengan mudah mengolah data dan menyajikan laporan *inventory*, ditunjang lagi dengan grafik perbandingan yang akan membantu user untuk melihat tingkat pemasukan dan pengeluaran pada beberapa periode tertentu. Berikut adalah fitur-fitur yang disediakan aplikasi *inventory*:

1. Halaman *dashboard* yang berfungsi sebagai halaman utama dan memuat laporan *inventory* berdasarkan periode yang dipilih serta admin dapat melakukan pencetakan laporan barang sesuai kebutuhan.
2. Halaman *supplier* yang memuat beberapa yang *supply* barang yang didalamnya terdapat 4 *method* yaitu *create*, *read*, *update* dan *delete*.
3. Halaman data barang yang memuat beberapa kolom seperti data satuan barang, jenis barang, dan data barang yang didalamnya terdapat 4 *method* yaitu *create*, *read*, *update* dan *delete*.
4. Halaman riwayat data barang masuk dan riwayat data barang keluar dengan 4 *method* yaitu *create*, *read*, *update* dan *delete*.

5. Halaman laporan berisi cetak laporan yang bisa disesuaikan dengan tanggal kebutuhan.
6. Halaman *user management* yaitu memuat beberapa user yang menggunakan aplikasi tersebut dengan *role* gudang.

4.1.6 Analisis Data

Untuk mendukung perancangan aplikasi *inventory* menggunakan metode perpetual, penulisan memerlukan beberapa data sebagai bahan penelitian dan perancangan aplikasi. Adapun data-data yang digunakan dan diolah pada aplikasi *inventory* yaitu sebagai berikut:

1. Data Admin

Analisis data yang akan didapatkan oleh admin yaitu detailnya keluar masuk barang yang dikelola PT. Visi Karya Prakarsa berupa laporan harian, mingguan dan bulanan. Dengan itu beberapa admin perusahaan mengetahui stock barang apa saja yang masih ada, sehingga admin dapat mengelolanya dengan baik.

2. Data barang

Analisis data barang yang akan digunakan yaitu detailnya berupa alat-alat elektro seperti beberapa jenis kabel, RJ45, arduino dan lainnnnya. Ada beberapa barang yang di supply kepada supplier da nada juga digunakan oleh kebutuhan pembuatan barang dalam project PT. Visi Karya Prakarsa.

4.1.7 Analisis Biaya

Untuk mendukung perancangan aplikasi *inventory* menggunakan metode Perpetual, penulis memerlukan beberapa data sebagai bahan penelitian dan perancangan aplikasi. Adapun data-data yang digunakan dan diolah pada aplikasi keuangan yaitu sebagai berikut:

Tabel 4.1 Analisis Biaya

No	Jenis Kebutuhan	Biaya
1	Biaya ATK	Rp. 500.000
2	Internet	Rp. 600.000
3	Analisis	Rp. 350.000
4	Programming	Rp. 1.000.000

5	Biaya Bahan Bakar	Rp. 400.000
Jumlah		Rp. 2.850.000

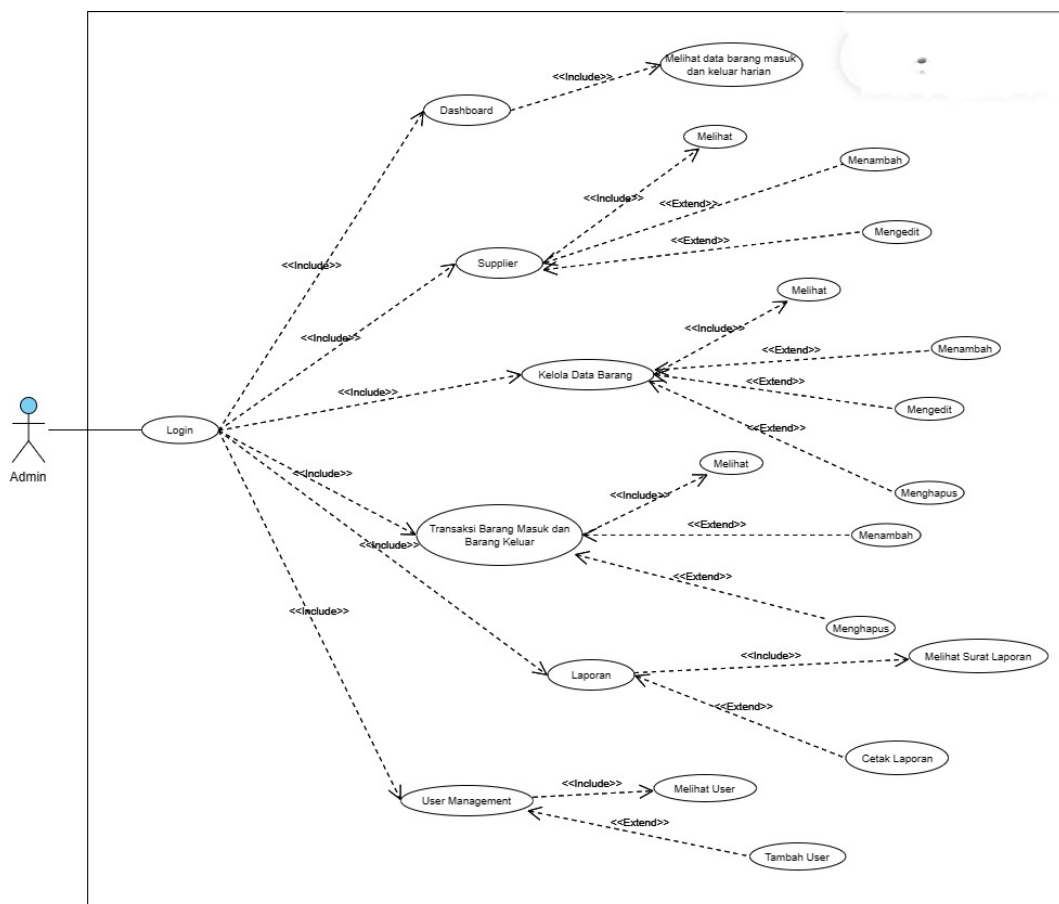
4.2 Perancangan

Sebelum masuk dalam pembuatan aplikasi dibutuhkan perancangan yaitu merancang software dalam bentuk UML yang terdiri dari *use case*, *activity diagram* dan *class diagram*.

4.2.1 UML (Unified Modelling Language)

1. Use Case Diagram

Pada rancangan *Use Case Diagram* bertujuan untuk memberikan gambaran interaksi actor dengan aplikasi inventory berbasis web. Berikut adalah penjelasan *use case diagram* aplikasi inventory berbasis web:



Gambar 4.1 Use Case Diagram Admin

Penjelasan mengenai gambar Use Case Diagram yang terdapat di atas dapat secara terinci diuraikan dalam sebuah tabel deskripsi yang telah disediakan :

a. Deskripsi Aktor

Tabel 4.2 Deskripsi Aktor

No	Aktor	Deskripsi
1	Admin	Admin merupakan pengguna yang dapat melakukan berbagai operasi pada aplikasi inventory berupa melihat laporan, menambah data, mengubah data, menghapus data serta mencetak laporan data.

b. Deskripsi Use Case

1) Skenario use case login

Tabel 4.3 Deskripsi Use Case Login

Nama	Login
Aktor	Admin
Skenario Utama	
Kondisi awal	Aktor belum memiliki akses pada aplikasi
Aktor	Sistem
Mengakses halaman login	Menampilkan halaman login
Mengisi form login	Memvalidasi username dan password. Jika berhasil, maka akan menampilkan halaman utama
Kondisi akhir	Hak akses pengguna sudah dipastikan sebagai admin maka akan menampilkan halaman utama

2) Skenario use case Dashboard

Tabel 4.4 Deskripsi Use Case Dashboard

Nama	<i>Dashboard</i>
Aktor	Admin
Skenario Utama	

Kondisi awal	Aktor berhasil <i>login</i> sebagai admin dan mengakses <i>dashboard</i>
Aktor	Sistem
Admin mengisi form <i>login</i> terlebih dahulu	Setelah berhasil <i>login</i> admin menampilkan halaman <i>dashboard</i>
Kondisi akhir	Menampilkan informasi halaman <i>dashboard</i> sesuai dengan hak akses pengguna

3) Skenario *use case* Tambah Barang

Tabel 4.5 Deskripsi Use Case Tambah Barang

Nama	Tambah Barang
Aktor	Admin
Skenario Utama	
Kondisi awal	Aktor berhasil login sebagai admin dan mengakses data barang
Aktor	Sistem
Admin masuk ke menu data barang	Menampilkan <i>form</i> input barang
Admin dapat menambah barang	Input barang yang akan ditambahkan dan menyimpannya
Kondisi akhir	Data barang berhasil ditambahkan kedalam <i>database</i>

4) Skenario *use case* form Edit Barang

Tabel 4.6 Deskripsi Use Case Edit Barang

Nama	Edit barang
Aktor	Admin
Skenario Utama	

Kondisi awal	Aktor berhasil login sebagai admin dan mengakses data barang
Aktor	Sistem
Admin masuk ke menu data barang	Menampilkan tabel barang
Admin memilih barang yang akan di edit	Data berhasil
Kondisi akhir	Data barang berhasil berubah dari <i>database</i>

5) Skenario *use case* form Hapus Barang

Tabel 4.7 Deskripsi Use Case Hapus Barang

Nama	Hapus Barang
Aktor	Admin
Skenario Utama	
Kondisi awal	Aktor
Admin masuk ke menu data barang	Data berhasil terhapus dari database
Admin memilih barang yang akan dihapus	Menampilkan tabel barang
Admin memilih data barang yang akan dihapus	Data berhasil terhapus
Kondisi akhir	Data barang berhasil terhapus dari database

6) Skenario *use case* pencetakan laporan

Tabel 4.8 Deskripsi Use Case Pencetakan Barang

Nama	Laporan
Aktor	Admin

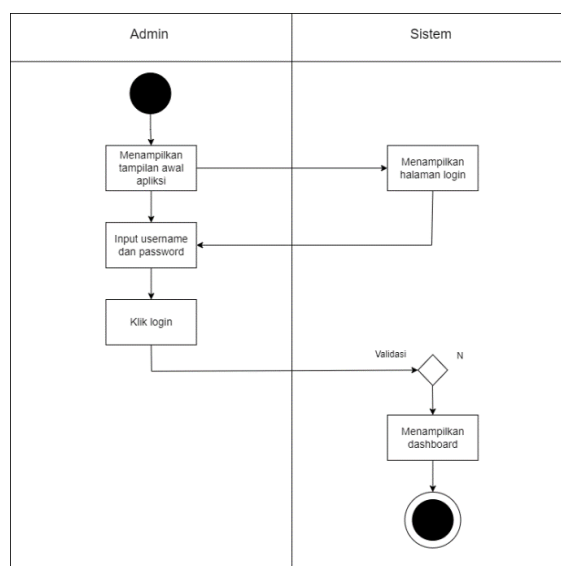
Skenario Utama	
Kondisi awal	Aktor berhasil login sebagai admin atau pengguna dan mengakses halaman dashboard aplikasi inventory serta melihat data barang yang akan dicetak
Aktor	Sistem
Admin / pengguna masuk ke halaman login	Menampilkan data barang yang tersimpan
Admin memilih tanggal atau periode yang akan dicetak	Menampilkan data sesuai tanggal
Kondisi akhir	Data barang berhasil dicetak

2. Activity Diagram

Activity diagram menggambarkan urutan aktivitas proses pada sebuah sistem. Berikut adalah *activity diagram* pada aplikasi *inventory* di PT. Visi Karya Prakarsa:

1. Activity diagram Login

Dalam *activity diagram login*, terdapat alur yang menggambarkan proses untuk melakukan autentikasi masuk ke dalam aplikasi, seperti yang terlihat pada gambar di bawah ini :

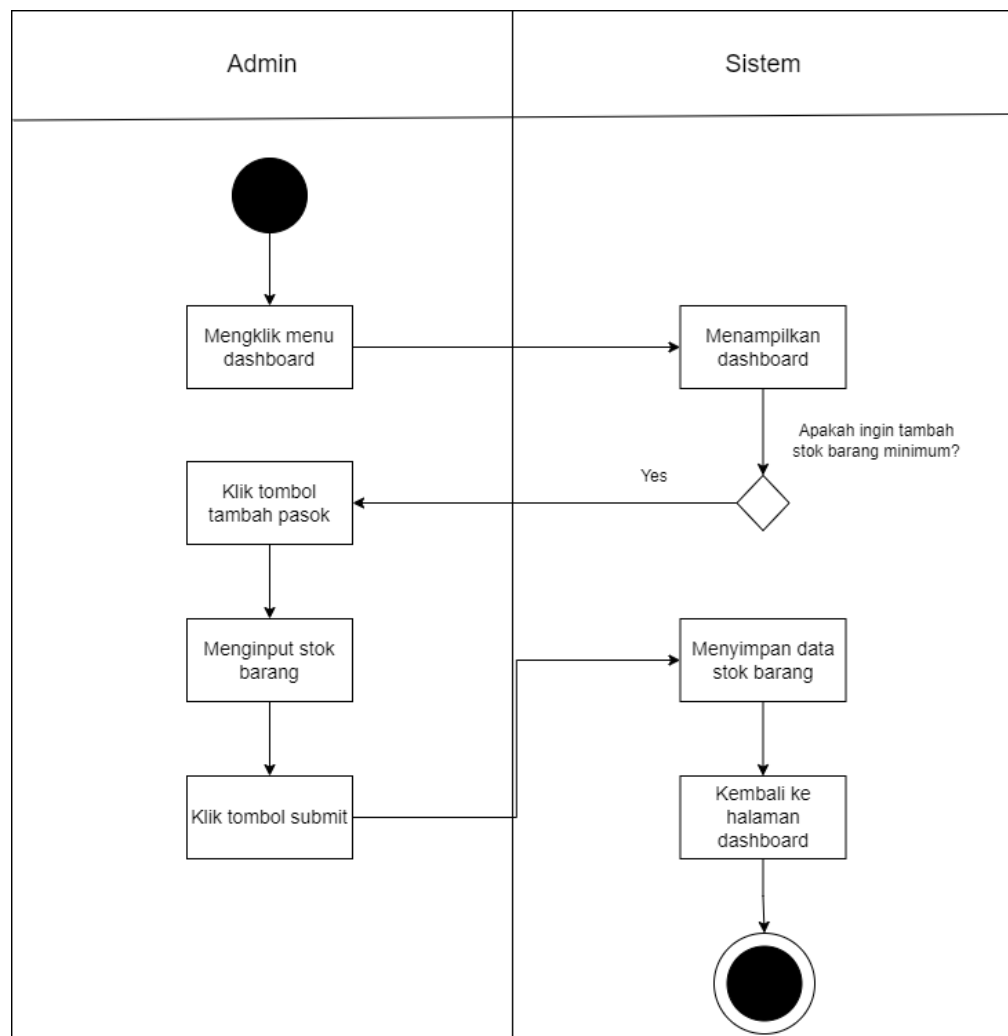


Gambar 4.2 Activity diagram login

Pada gambar 4.2 menunjukkan bahwa admin akan melakukan *login* aplikasi *inventory* dengan memasukkan *username* dan *password*, kemudian sistem akan melakukan validasi terhadap input *login* yang dimasukkan oleh admin, apabila data sudah sesuai maka sistem akan menampilkan halaman *dashboard* namun apabila data tidak sesuai sistem akan menampilkan peringatan dan form login kembali.

2. Activity diagram Dashboard

Dalam *activity diagram dashboard*, terdapat alur yang menggambarkan alur yang terjadi ketika sudah masuk ke dalam aplikasi, seperti yang terlihat pada gambar di bawah ini :



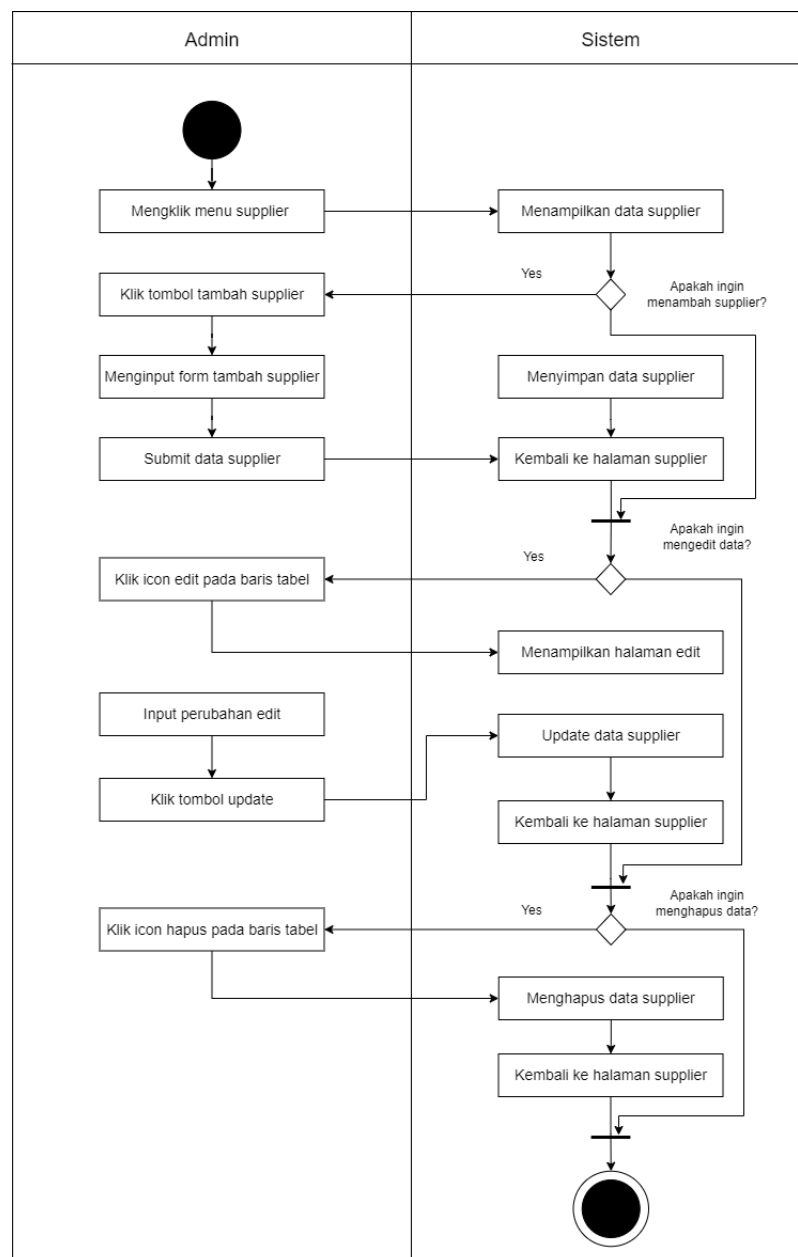
Gambar 4.3 Activity diagram dashboard

Pada gambar 4.3 menjelaskan bahwa admin yang sudah melakukan *login* atau mengklik menu *dashboard* secara otomatis sistem akan menampilkan

dashboard sebagai halaman utama yang berisi jumlah persediaan barang per periode berupa diagram. Pada menu ini admin juga dapat melakukan input stok barang sesuai dengan periode yang dipilih pada *dashboard* apabila dibutuhkan.

3. Activity diagram supplier

Dalam *activity diagram supplier*, terdapat urutan proses yang mengilustrasikan aliran barang dari pemasok, sebagaimana yang tergambar pada gambar di bawah ini :

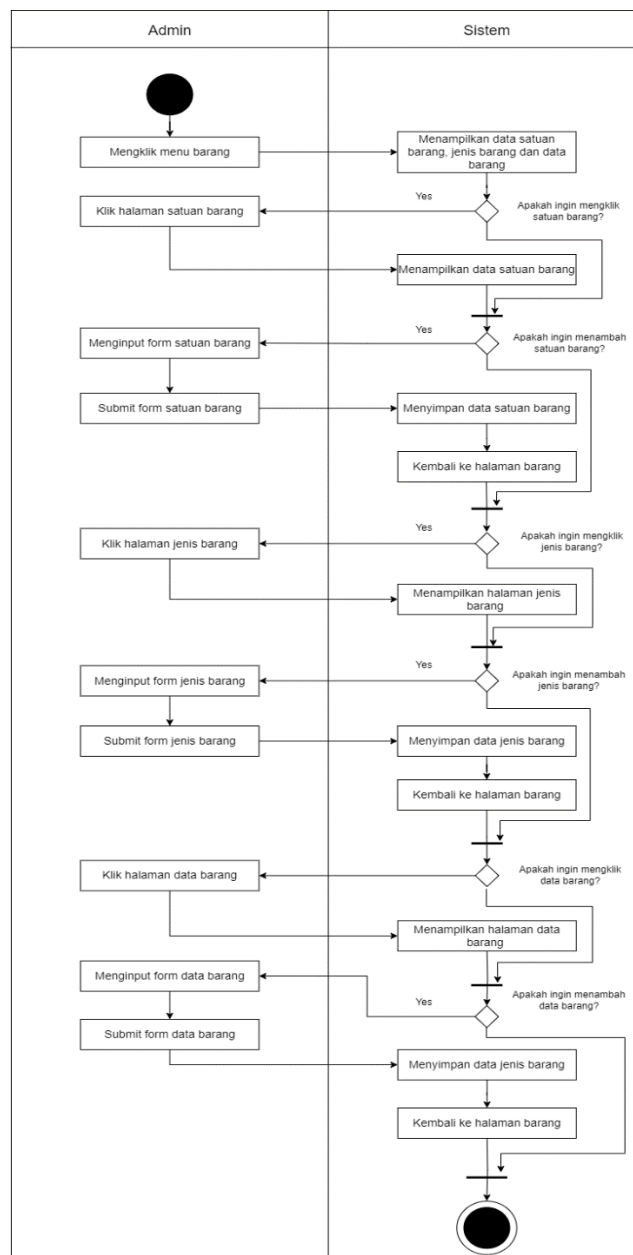


Gambar 4.4 Activity diagram supplier

Pada gambar 4.4 menjelaskan bahwa jika admin mengklik menu *supplier* maka sistem akan menampilkan tabel *supplier*, admin juga dapat melakukan tambah data dengan cara mengklik tombol tambah, mengubah data dengan cara mengklik tombol edit pada tabel yang dipilih serta dapat menghapus data.

4. Activity diagram barang

Dalam *activity diagram* yang menggambarkan barang, terdapat representasi visual dari beberapa elemen fitur yang mencakup satuan barang, jenis barang, dan data barang, seperti yang ditampilkan pada ilustrasi di bawah ini :

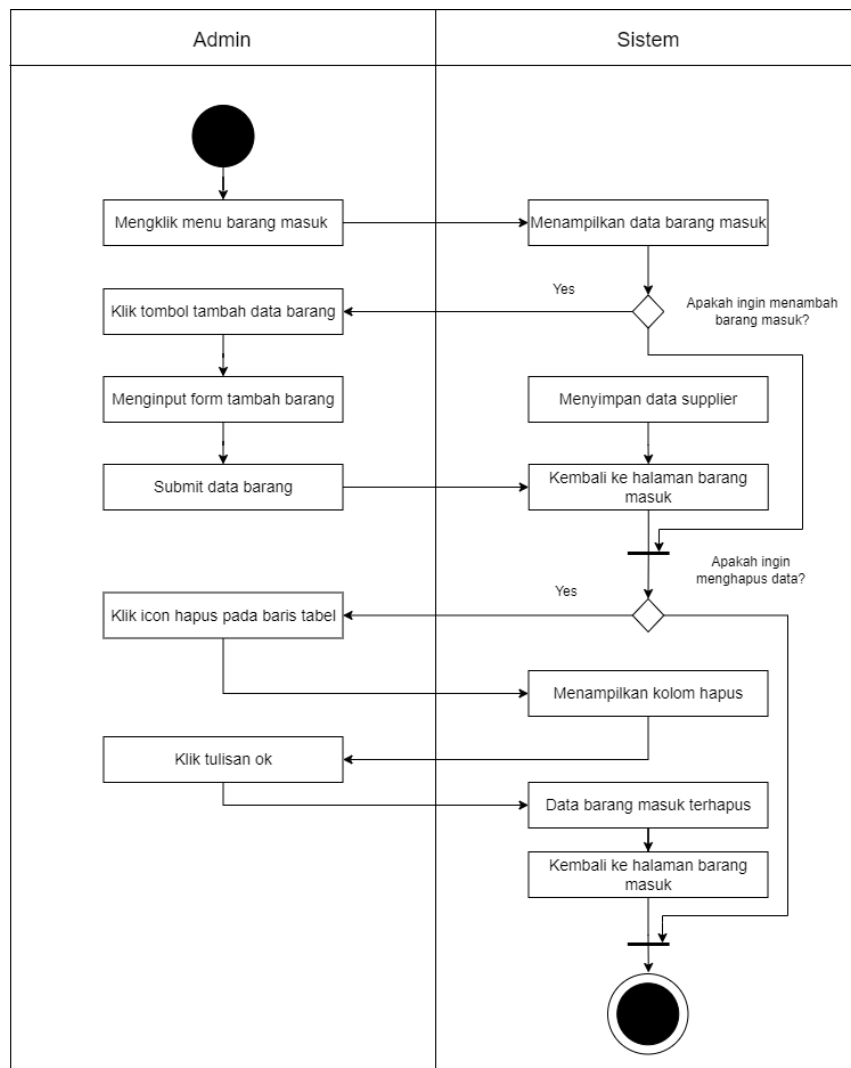


Gambar 4.5 Activity diagram barang

Pada gambar 4.5 menjelaskan bahwa jika admin mengklik menu barang maka sistem akan menampilkan beberapa pilihan halaman untuk satuan barang, jenis barang dan data barang. Masing-masing dari halaman tersebut admin juga dapat melakukan tambah data dengan cara mengklik tombol tambah, mengubah data dengan cara mengklik tombol edit pada tabel yang dipilih serta dapat menghapus data dengan cara mengklik tombol hapus pada tabel yang dipilih.

5. Activity diagram transaksi barang masuk

Dalam *activity diagram* transaksi barang masuk, terdapat urutan proses yang mengilustrasikan tahapan pemasukan barang, sebagaimana tercermin dalam gambar yang disajikan di bawah ini:

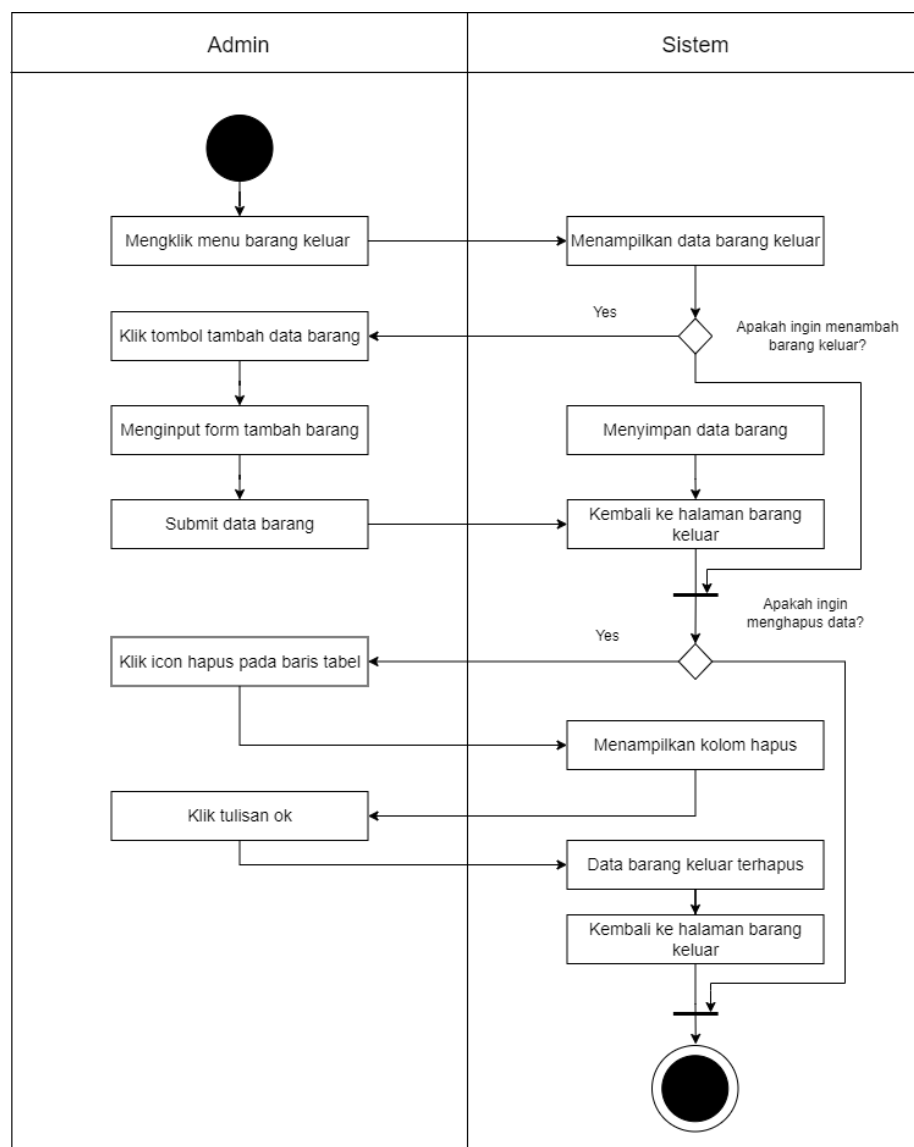


Gambar 4.6 Activity diagram transaksi barang masuk

Pada gambar 4.6 menjelaskan bahwa jika admin mengklik halaman barang masuk maka sistem akan menampilkan tabel data transaksi barang masuk, admin juga dapat melakukan tambah data dengan cara mengklik tombol tambah, dan dapat menghapus data dengan cara mengklik tombol hapus pada tabel yang dipilih.

6. Activity diagram transaksi barang keluar

Dalam *activity diagram* transaksi barang keluar, terdapat urutan proses yang mengilustrasikan tahapan pengeluaran barang, sebagaimana tercermin dalam gambar yang disajikan di bawah ini:

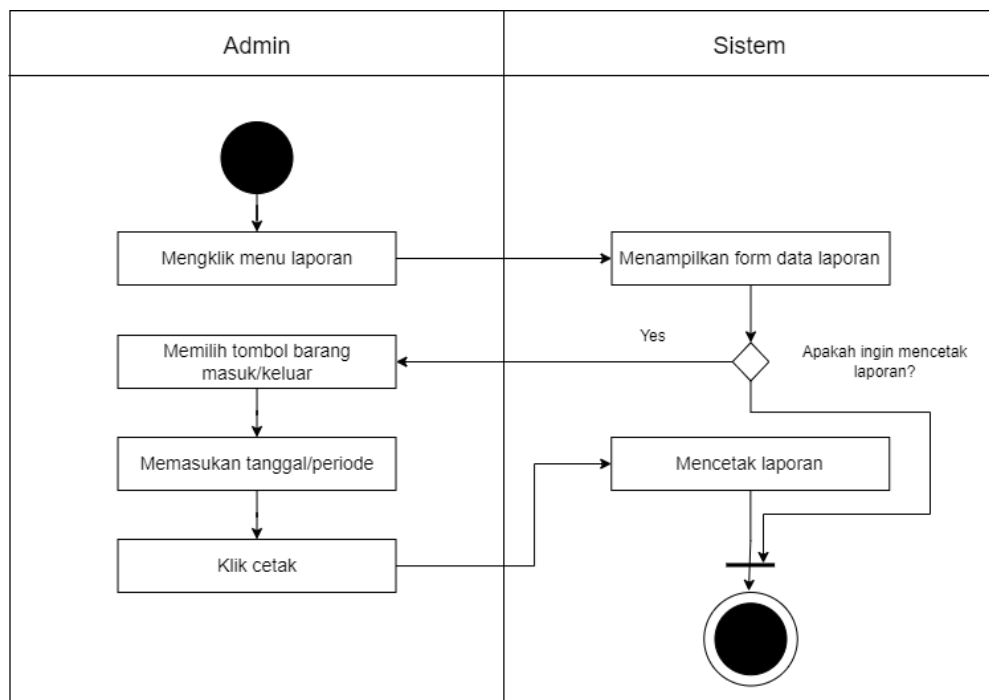


Gambar 4.7 Activity diagram transaksi barang keluar

Pada gambar 4.7 menjelaskan bahwa jika admin mengklik halaman barang keluar maka sistem akan menampilkan tabel data transaksi barang keluar, admin juga dapat melakukan tambah data, dan dapat menghapus data dengan cara mengklik tombol hapus pada tabel yang dipilih.

7. Activity diagram laporan

Dalam *activity diagram* yang berfokus pada laporan, terdapat sebuah proses yang mengilustrasikan langkah-langkah pencetakan laporan pada suatu periode tertentu, sebagaimana yang tergambar dalam gambar di bawah ini:

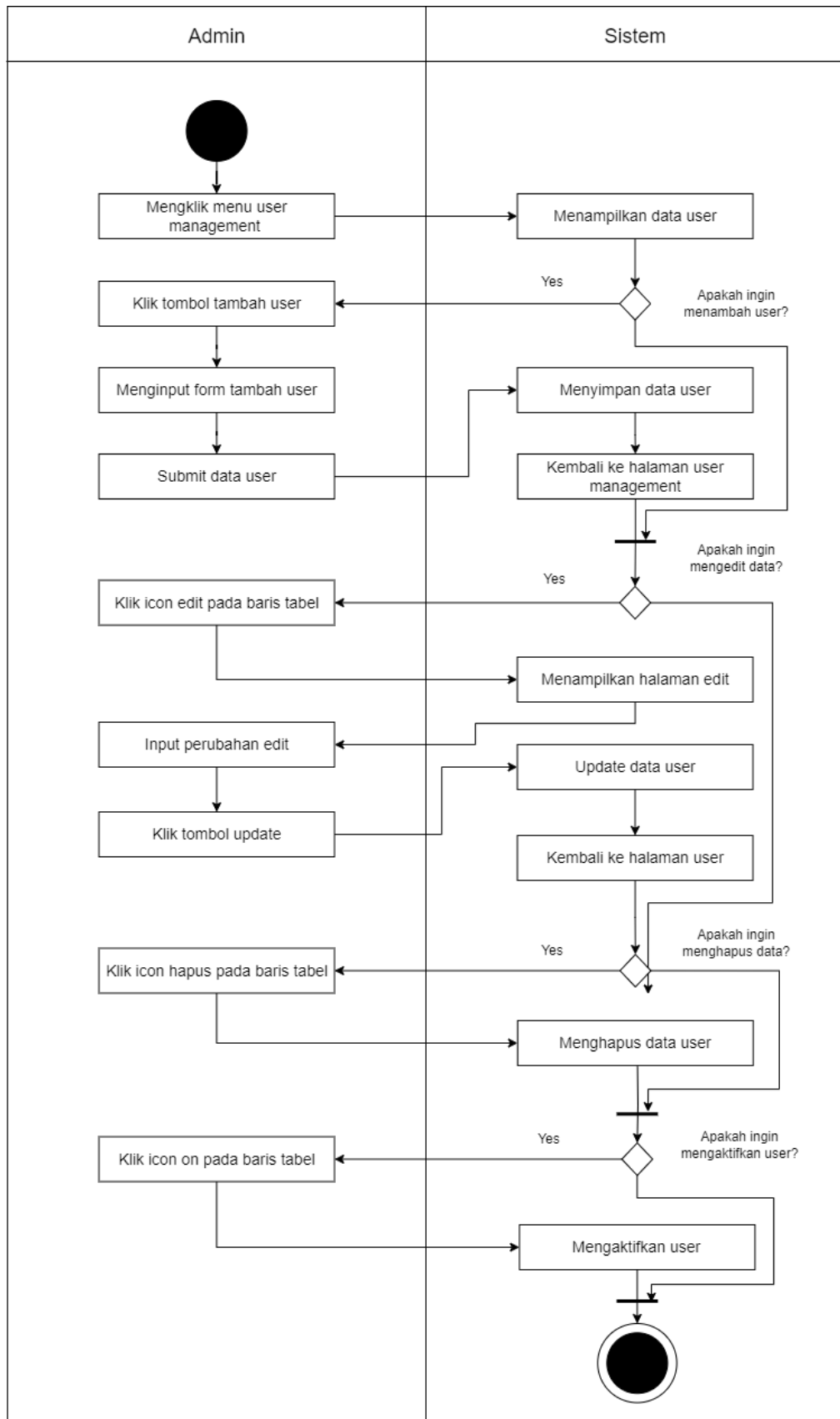


Gambar 4.8 Activity diagram laporan

Pada gambar 4.8 diatas menjelaskan bahwa admin yang sudah mengklik halaman laporan secara otomatis sistem akan menampilkan form laporan yang akan dibutuhkan untuk dicetak sesuai dengan periode yang dipilih pada persediaan barang apabila dibutuhkan.

8. Activity diagram user management

Dalam diagram aktivitas manajemen pengguna (*user management*), terdapat suatu proses yang menggambarkan pengelolaan status aktif dan non-aktif dari pengguna aplikasi, sebagaimana tercermin dalam gambar di bawah ini:

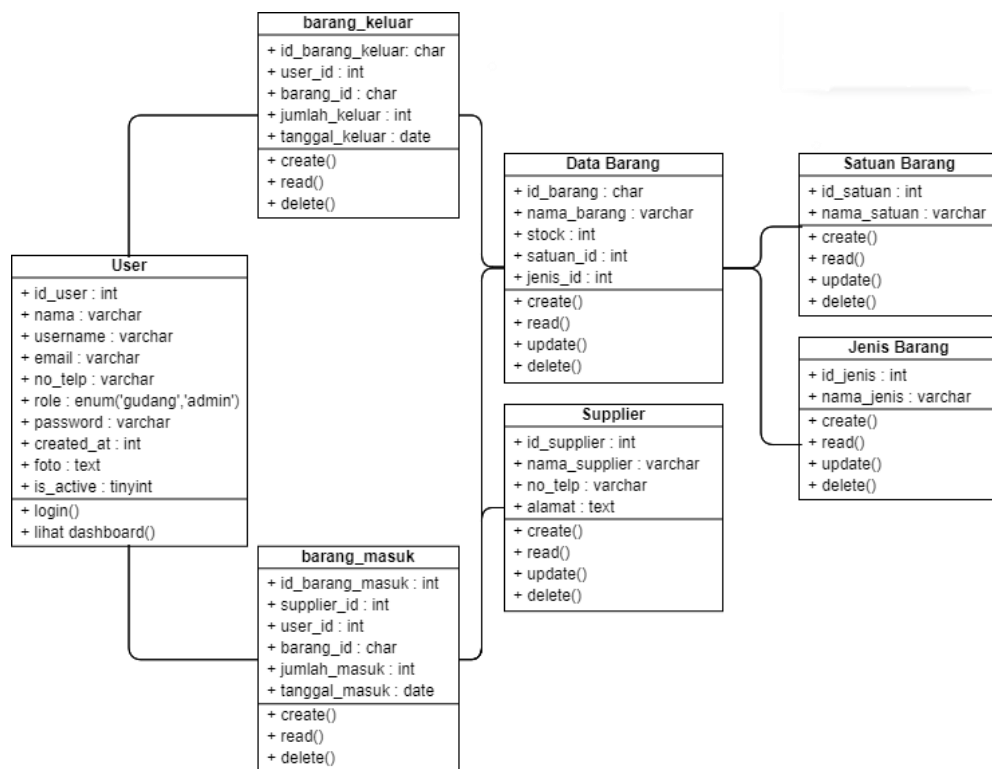


Gambar 4.9 Activity diagram user management

Pada gambar 4.9 menjelaskan bahwa jika admin mengklik menu user management maka sistem akan menampilkan tabel datauser, admin juga dapat melakukan tambah data dengan cara mengklik tombol tambah, mengubah data dengan cara mengklik tombol edit pada tabel yang dipilih serta dapat menghapus data dengan cara mengklik tombol hapus pada tabel yang dipilih. Untuk mengaktifkan user tersebut bisa mengklik tombol icon on pada tabel paling ujung.

7) Class Diagram

Class diagram adalah diagram yang menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Berikut adalah class diagram dari aplikasi *inventory*:



Gambar 4.10 Class diagram aplikasi inventory

Grafik di atas menggambarkan beberapa struktur dalam diagram kelas aplikasi sistem inventarisasi berdasarkan hak akses pengguna terhadap entitas "barang_masuk" dan "barang_keluar," di mana "barang_masuk" berasal dari pemasok. Selain itu, "barang_masuk" dan "barang_keluar" juga memiliki

kemampuan untuk mengakses informasi terkait entitas "data_barang," "satuan_barang," dan "jenis_barang."

4.2.2 Struktur Tabel

Tabel-tabel yang terdapat dalam basis data yang digunakan dalam aplikasi *inventory* menggunakan metode Perpetual sebagai pengukur kinerja perusahaan di PT. Visi Karya Prakarsa adalah sebagai berikut:

1. Perancangan tabel *user*

Dalam perancangan tabel untuk menyimpan informasi pengguna (*user*) dalam basis data, berikut perancangan tabel untuk pengguna:

Tabel 4.9 Perancangan tabel login

Field	Type	Size	Indeks	Deskripsi
Id	Int	11	PK	Id user
Nama	Varchar	50		Nama
Username	Varchar	50		Username
Email	Varchar	100		Email
No_Telp	Varchar	15		No telpon
Role	Enum('gudang', 'admin')			
Password	Varchar	255		Password
Created_at	Int	11		
Foto	Text			Foto
Is_active	Tinyint	1		

2. Perancangan tabel barang masuk

Dalam perancangan tabel untuk menyimpan informasi barang masuk dalam basis data, berikut perancangan tabel untuk barang masuk:

Tabel 4.10 Perancangan tabel barang masuk

Field	Type	Size	Indeks	Deskripsi
Id_barang_masuk	Char	16	PK	Barang masuk
Supplier_id	Int	11	FK	Supplier

User_id	Int	11	FK	User
Barang_id	Char	7	FK	Barang
Jumlah_masuk	Int	11		Jumlah masuk
Tanggal_masuk	Date			Tanggal masuk

3. Perancangan tabel barang keluar

Dalam perancangan tabel untuk menyimpan informasi barang keluar dalam basis data, berikut perancangan tabel untuk barang keluar:

Tabel 4.11 Perancangan tabel barang keluar

Field	Type	Size	Indeks	Deskripsi
Id_barang_keluar	Char	16	PK	Barang keluar
User_id	Int	11	FK	User
Barang_id	Char	7	FK	Barang
Jumlah_keluar	Int	11		Jumah keluar
Tanggal_keluar	Date			Tanggal keluar

4. Perancangan tabel supplier

Dalam perancangan tabel untuk menyimpan informasi supplier dalam basis data, berikut perancangan tabel untuk supplier:

Tabel 4.12 Perancangan tabel supplier

Field	Type	Size	Indeks	Deskripsi
Id_supplier	Int	11	PK	Supplier
Nama_supplier	Varchar	50		Varchar
No_telp	Varchar	15		Varchar
Alamat	Text			Date

5. Perancangan tabel barang

Dalam perancangan tabel untuk menyimpan informasi barang dalam basis data, berikut perancangan tabel untuk barang:

Tabel 4.13 Perancangan tabel barang

Field	Type	Size	Indeks	deskripsi
Id_barang	Char	7	PK	ID barang
Nama_barang	Varchar	255		Nama barang
Stok	Int	11		Stok
Satuan_id	Int	11	FK	Satuan barang
Jenis_id	Int	11	FK	Jenis barang

6. Perancangan tabel satuan barang

Dalam perancangan tabel untuk menyimpan informasi satuan dalam basis data, berikut perancangan tabel untuk satuan barang:

Tabel 4.14 Perancangan tabel satuan barang

Field	Type	Size	Indeks	deskripsi
Id_satuan	Char	11	PK	ID satuan barang
Nama_satuan	Varchar	20		Nama satuan barang

7. Perancangan tabel jenis barang

Dalam perancangan tabel untuk menyimpan informasi jenis barang dalam basis data, berikut perancangan tabel untuk jenis barang:

Field	Type	Size	Indeks	deskripsi
Id_jenis	Char	11	PK	ID jenis barang
Nama jenis	Varchar	20		Nama jenis barang

4.2.3 Desain antarmuka

1. Desain tampilan antarmuka *login*

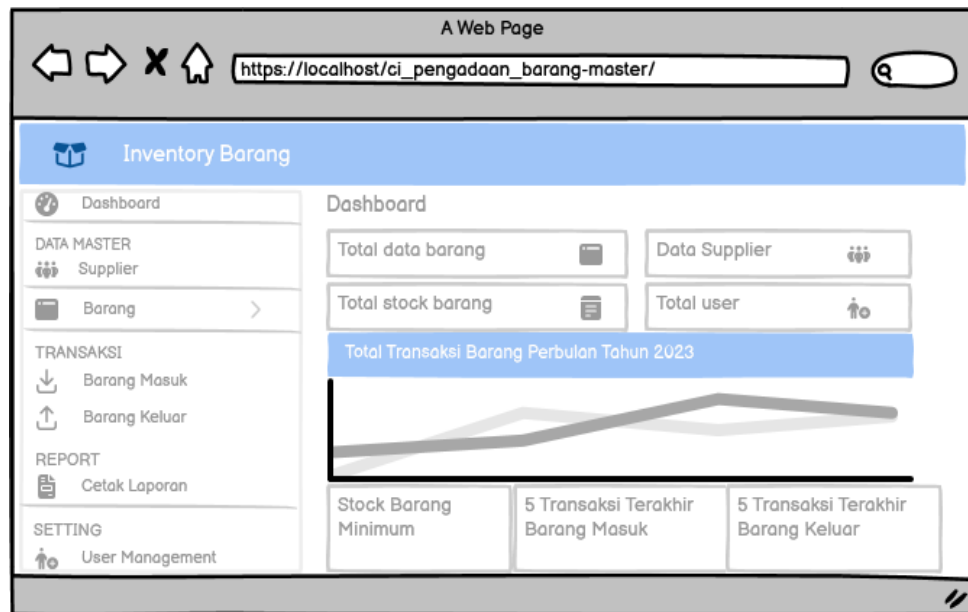
Berikut dibawah ini merupakan tampilan antarmuka *login* aplikasi *inventory* yang akan dirancang:

Gambar 4.11 Halaman login

Pada antarmuka *login*, terdapat elemen tampilan yang melibatkan halaman yang memuat input field untuk memasukkan "*username*" dan "*password*." Setelah kedua input tersebut diisi dengan benar, pengguna akan diberikan akses untuk melanjutkan ke halaman berikutnya.

2. Desain tampilan antarmuka *dashboard*

Dibawah ini terdapat representasi visual dari antarmuka *dashboard* aplikasi sistem *inventory* yang sedang direncanakan.

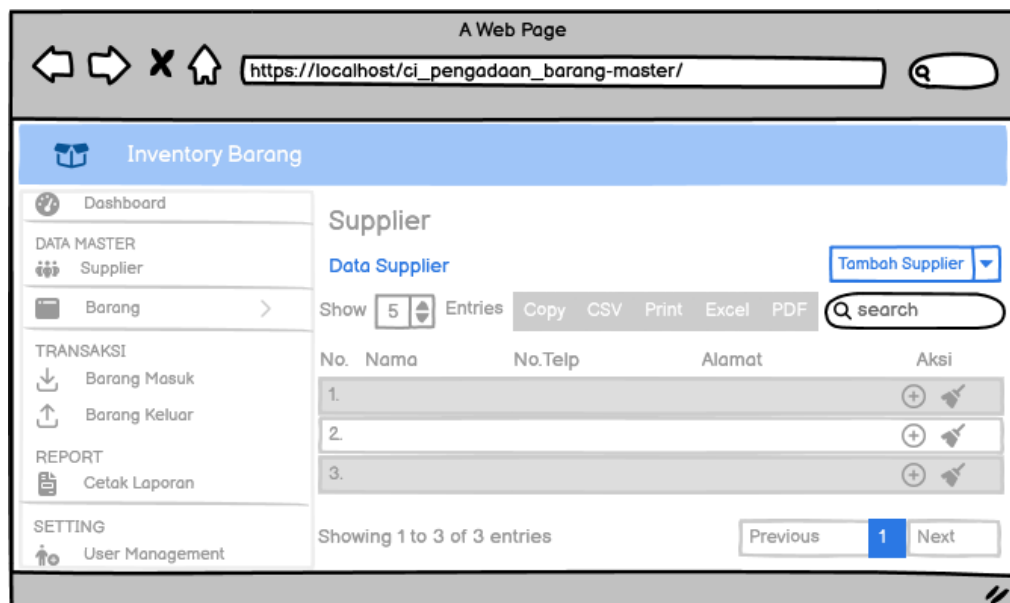


Gambar 4.12 Halaman dashboard

Dalam antarmuka *dashboard*, terdapat tampilan halaman yang menyajikan informasi mengenai total data barang, data *supplier*, total stok barang, dan jumlah pengguna. Selain itu, dalam tampilan dashboard juga tersedia grafik bulanan yang memvisualisasikan data pemasukan dan pengeluaran barang.

3. Desain tampilan antarmuka *supplier*

Berikut dibawah ini merupakan tampilan antarmuka *supplier* aplikasi *inventory* yang akan dirancang:

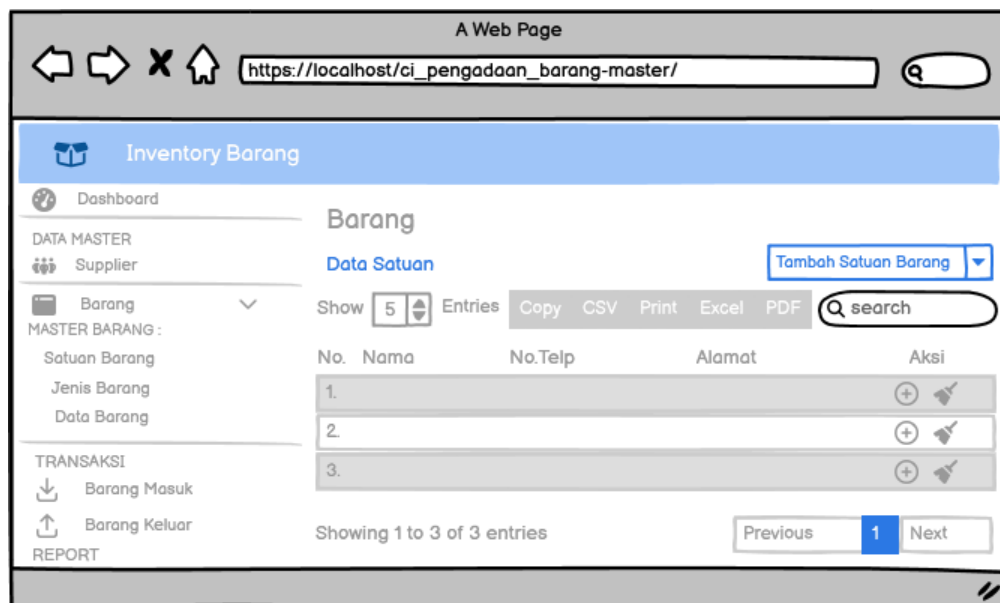


Gambar 4.13 Desain tampilan antarmuka *supplier*

Pada antarmuka *supplier*, terdapat tampilan halaman yang menginformasikan mengenai barang yang diterima dari pemasok. Informasi tersebut meliputi data seperti nomor telepon dan alamat pemasok. Dalam kolom penyediaan (*supply*), terdapat ikon yang memungkinkan untuk melakukan pengeditan (edit) dan penghapusan (hapus) data.

4. Desain tampilan antarmuka data satuan barang

Berikut dibawah ini merupakan tampilan antarmuka data satuan barang aplikasi *inventory* yang akan dirancang:

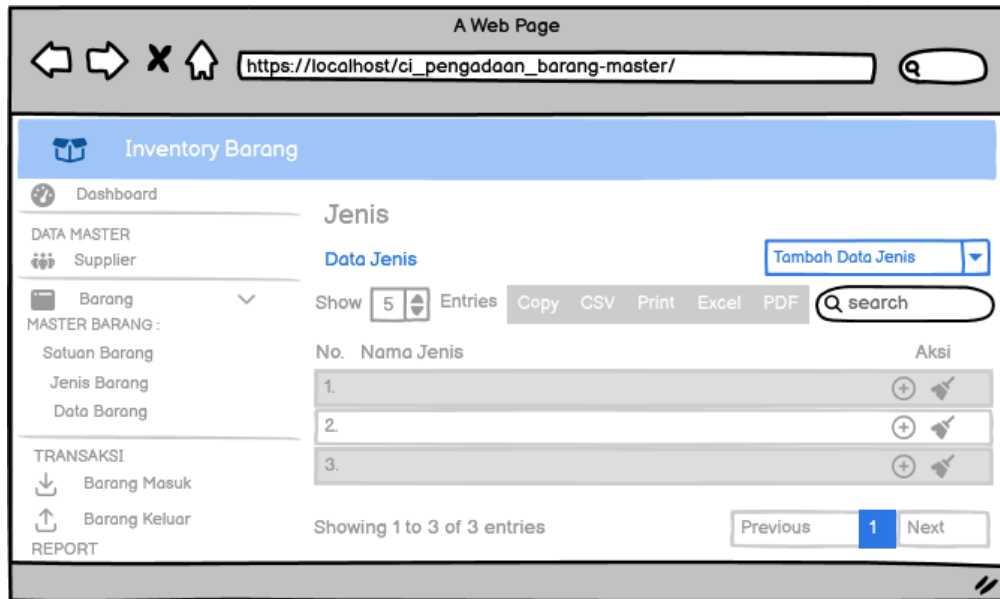


Gambar 4.14 Desain tampilan antarmuka satuan barang

Dalam antarmuka barang, terdapat berbagai opsi yang dapat diakses, termasuk pilihan untuk mengelola satuan barang yang memungkinkan akses ke data satuan barang. Selain itu, terdapat juga fitur aksi yang memungkinkan pengguna untuk melakukan pengeditan dan penghapusan data satuan barang.

5. Desain tampilan antarmuka barang jenis barang

Berikut dibawah ini merupakan tampilan antarmuka jenis barang aplikasi *inventory* yang akan dirancang:

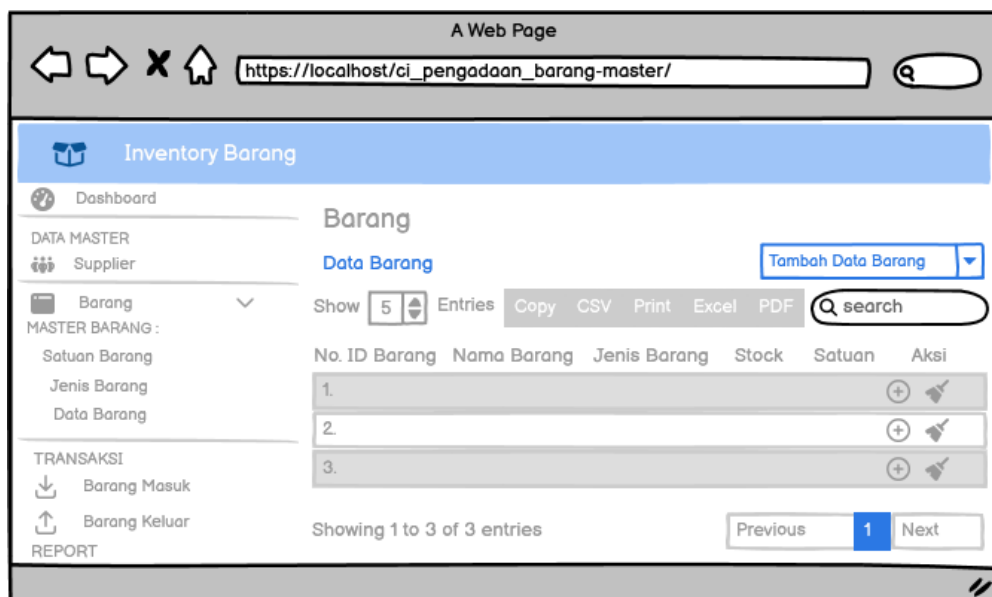


Gambar 4.15 Desain tampilan antarmuka jenis barang

Pada antarmuka barang, terdapat beberapa pilihan yang dapat diakses diantaranya ada satuan barang yang dapat mengakses data satuan barang serta ada aksi yang dapat digunakan sebagai pengeditan dan penghapusan data satuan barang

6. Desain tampilan antarmuka data barang

Berikut dibawah ini merupakan tampilan antarmuka data barang aplikasi *inventory* yang akan dirancang:

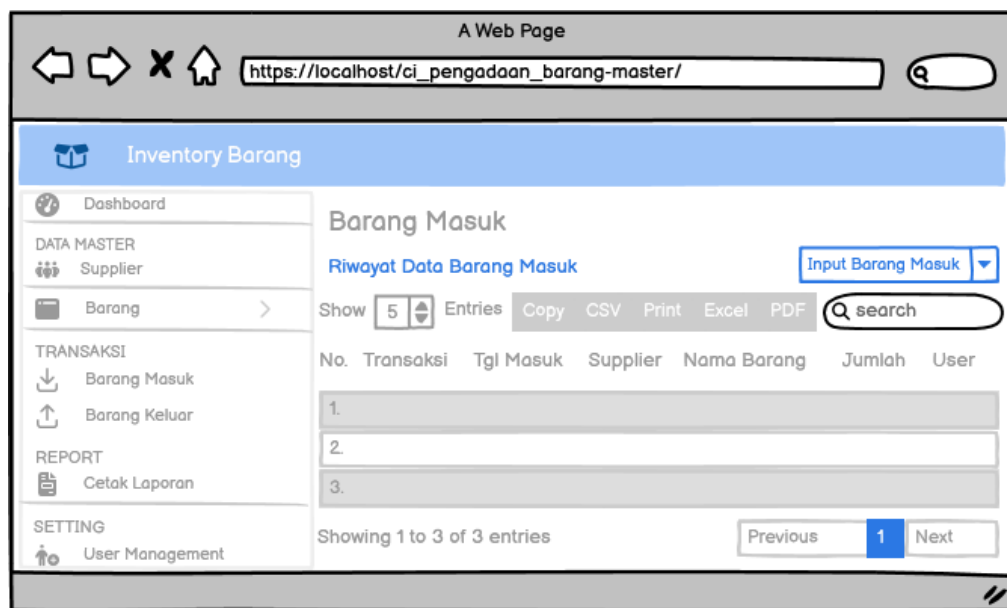


Gambar 4.16 Desain tampilan antarmuka data barang

Dalam antarmuka data barang, terdapat kolom yang memuat informasi mengenai jenis barang, nama barang, dan stok satuan barang. Pengguna memiliki kemampuan untuk menambahkan, mengedit, serta menghapus data barang melalui kolom yang tersedia.

7. Desain tampilan antarmuka riwayat data barang masuk

Berikut dibawah ini merupakan tampilan antarmuka riwayat data barang masuk aplikasi *inventory* yang akan dirancang:

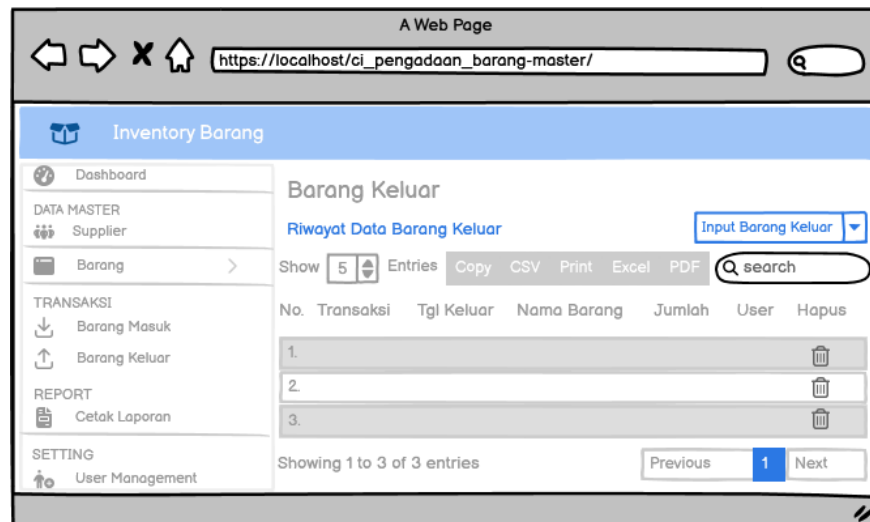


Gambar 4.17 Desain tampilan antarmuka transaksi barang masuk

Dalam antarmuka transaksi data barang masuk, terdapat kolom yang memuat informasi mengenai tanggal masuk, asal pemasok, serta jumlah barang yang masuk. Penting untuk dicatat bahwa riwayat data barang masuk dalam kolom tersebut hanya dapat dihapus, dan tidak memungkinkan untuk mengedit data tersebut karena representasi yang akurat dari informasi *real-time* sangat penting.

8. Desain tampilan antarmuka riwayat data barang keluar

Berikut dibawah ini merupakan tampilan antarmuka riwayat data barang keluar aplikasi *inventory* yang akan dirancang:

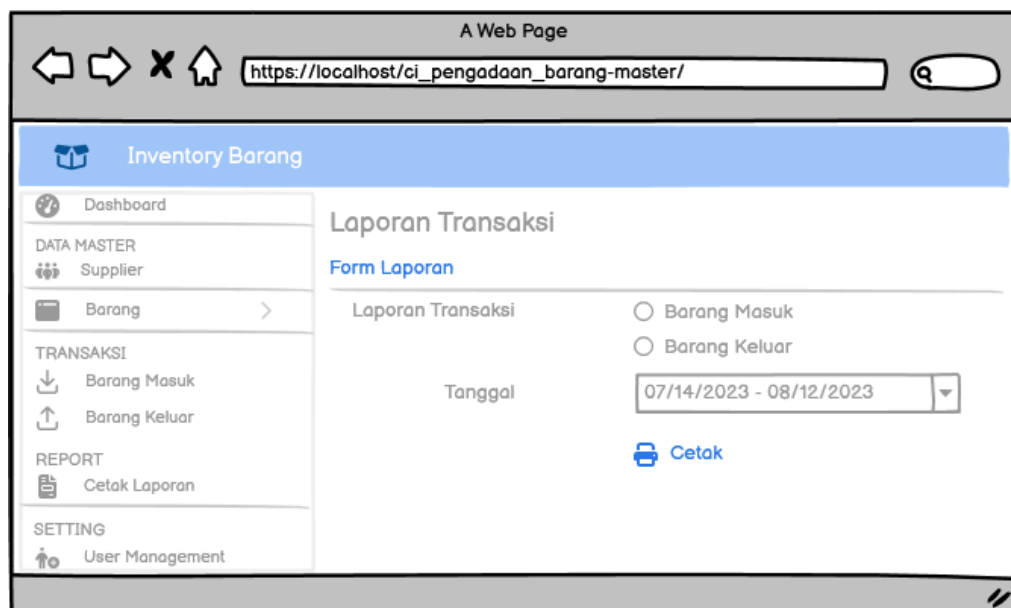


Gambar 4.18 Desain tampilan antarmuka transaksi barang keluar

Dalam antarmuka transaksi data barang keluar, terdapat kolom yang memuat informasi mengenai tanggal keluar, nama barang yang dikeluarkan, serta jumlah barang yang dikeluarkan. Penting untuk dicatat bahwa riwayat data barang keluar dalam kolom tersebut hanya dapat dihapus, dan tidak memungkinkan untuk mengedit data tersebut karena representasi yang akurat dari informasi *real-time* sangat penting.

9. Desain tampilan antarmuka laporan transaksi

Berikut dibawah ini merupakan tampilan antarmuka laporan transaksi aplikasi *inventory* yang akan dirancang:

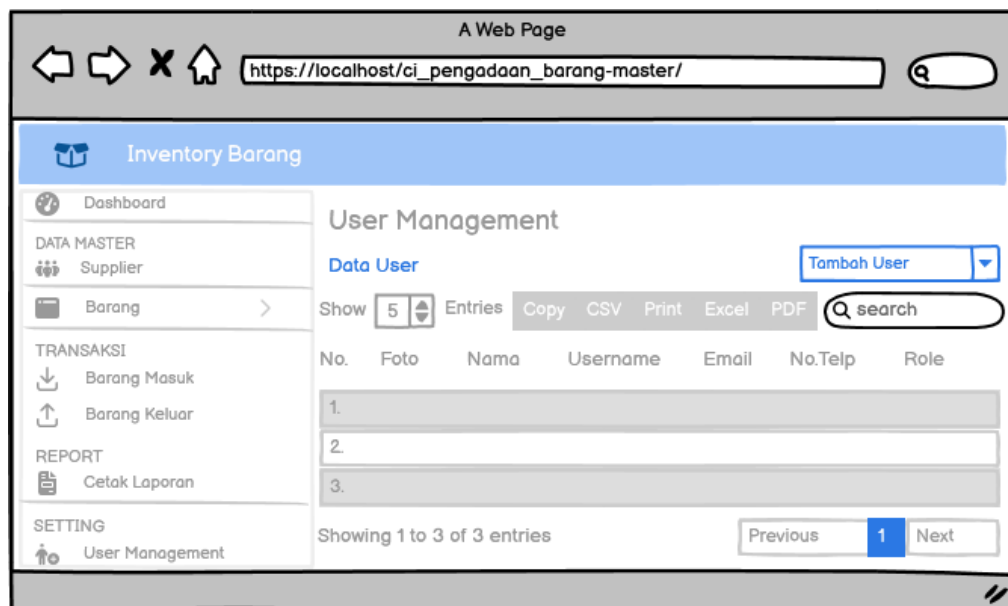


Gambar 4.19 Desain tampilan antarmuka laporan

Dalam antarmuka laporan transaksi, terdapat sebuah formulir transaksi yang memungkinkan pengguna untuk memasukkan periode waktu yang ingin dicetak dalam laporan. Pengguna dapat memilih opsi untuk laporan transaksi barang masuk atau barang keluar, dan setelah itu, mereka dapat mengklik opsi "cetak" untuk mencetak laporan sesuai dengan periode yang telah ditentukan.

10. Desain tampilan antarmuka *user management*

Berikut dibawah ini merupakan tampilan antarmuka *user management* aplikasi *inventory* yang akan dirancang:



Gambar 4.20 Halaman *user management*

Dalam antarmuka manajemen pengguna (*user management*), terdapat data pengguna aplikasi yang mencakup foto profil, nama lengkap, username yang sesuai dengan alamat email, dan nomor telepon. Pengguna memiliki kemampuan untuk menambahkan, mengedit, menghapus, serta mengaktifkan dan menonaktifkan pengguna melalui kolom data yang tersedia.

BAB V

IMPLEMENTASI DAN PENGUJIAN

5.1 Implementasi

Setelah melaksanakan analisis dan perancangan maka selanjutnya adalah pengimplementasian untuk menjalankan analisis dan perancangan yang sudah dibuat ke dalam bentuk aplikasi.

1.1.1 Listing Program

1. Listing program *controller auth*

```
<!-- Outer Row -->
<div class="row justify-content-center mt-5 pt-lg-5">

    <div class="col-xl-10 col-lg-12 col-md-9">

        <div class="card o-hidden border-0 shadow-lg">
            <div class="card-body p-lg-5 p-0">
                <!-- Nested Row within Card Body -->
                <div class="row">
                    <div class="col-lg-6 d-none d-lg-block bg-login-
image"></div>
                    <div class="col-lg-6">
                        <div class="p-5">
                            <div class="text-center mb-4">
                                <h1 class="h4 text-gray-900">Inventory
Barang</h1>
                                <span class="text-muted">Login</span>
                            </div>
                            <?= $this->session->flashdata('pesan'); ?>
                            <?= form_open("", ['class' => 'user']); ?>
                            <div class="form-group">
```

```

        <input autofocus="autofocus"
autocomplete="off" value="<?= set_value('username'); ?>"
type="text" name="username" class="form-control form-control-
user" placeholder="Username">
        <?= form_error('username', '<small class="text-
danger">', '</small>'); ?>
    </div>
    <div class="form-group">
        <input type="password" name="password"
class="form-control form-control-user" placeholder="Password">
        <?= form_error('password', '<small class="text-
danger">', '</small>'); ?>
    </div>
    <button type="submit" class="btn btn-primary
btn-user btn-block">
        Login
    </button>
    <div class="text-center mt-4">
        <a class="small" href="<?= base_url('register')
?>">Buat Akun!</a>
    </div>
    <?= form_close(); ?>
</div>
</div>
</div>
</div>
</div>
</div>
</div>

```

2. Listing program *dashboard*

```

<div class="row">

  <div class="col-xl-3 col-6 mb-4">
    <div class="card border-left-primary shadow h-100 py-2">
      <div class="card-body">
        <div class="row no-gutters align-items-center">
          <div class="col mr-2">
            <div class="text-xs font-weight-bold text-primary
text-uppercase mb-1">Total Data Barang</div>
            <div class="h5 mb-0 font-weight-bold text-gray-
800"><?= $barang; ?></div>
          </div>
          <div class="col-auto">
            <i class="fas fa-folder fa-2x text-gray-300"></i>
          </div>
        </div>
      </div>
    </div>
  </div>

  <div class="col-xl-3 col-6 mb-4">
    <div class="card border-left-success shadow h-100 py-2">
      <div class="card-body">
        <div class="row no-gutters align-items-center">
          <div class="col mr-2">
            <div class="text-xs font-weight-bold text-success
text-uppercase mb-1">Data Supplier</div>
            <div class="h5 mb-0 font-weight-bold text-gray-
800"><?= $supplier; ?></div>
          </div>
          <div class="col-auto">

```

```

        <i class="fas fa-users fa-2x text-gray-300"></i>
    </div>
</div>
</div>
</div>
</div>
</div>

<div class="col-xl-3 col-6 mb-4">
    <div class="card border-left-info shadow h-100 py-2">
        <div class="card-body">
            <div class="row no-gutters align-items-center">
                <div class="col mr-2">
                    <div class="row no-gutters align-items-center">
                        <div class="col mr-2">
                            <div class="text-xs font-weight-bold text-info
text-uppercase mb-1">Total Stok Barang</div>
                            <div class="h5 mb-0 font-weight-bold text-
gray-800"><?= $stok; ?></div>
                        </div>
                        <div class="col-auto">
                            <div class="progress progress-sm mr-2">
                                <div class="progress-bar bg-info"
role="progressbar" style="width: 50%" aria-valuenow="50" aria-
valuemin="0" aria-valuemax="100"></div>
                            </div>
                        </div>
                    </div>
                </div>
                <div class="col-auto">
                    <i class="fas fa-clipboard-list fa-2x text-gray-
300"></i>
                </div>
            </div>
        </div>
    </div>

```

```

        </div>
    </div>
</div>
</div>

<div class="col-xl-3 col-6 mb-4">
    <div class="card border-left-warning shadow h-100 py-2">
        <div class="card-body">
            <div class="row no-gutters align-items-center">
                <div class="col mr-2">
                    <div class="text-xs font-weight-bold text-warning
text-uppercase mb-1">Total User</div>
                    <div class="h5 mb-0 font-weight-bold text-gray-
800"><?= $user; ?></div>
                </div>
                <div class="col-auto">
                    <i class="fas fa-user-plus fa-2x text-gray-300"></i>
                </div>
            </div>
        </div>
    </div>
</div>
</div>
</div>

<div class="row">

    <!-- Area Chart -->
    <div class="col-xl-8 col-lg-7">
        <div class="card shadow mb-4">
            <!-- Card Header - Dropdown -->
            <div class="card-header bg-primary py-3 d-flex flex-row
align-items-center justify-content-between">

```

```

        <h6 class="m-0 font-weight-bold text-white">Total
Transaksi Barang Perbulan pada Tahun <?= date('Y'); ?></h6>
    </div>
    <!-- Card Body -->
    <div class="card-body">
        <div class="chart-area">
            <div class="chartjs-size-monitor">
                <div class="chartjs-size-monitor-expand">
                    <div class=""></div>
                </div>
                <div class="chartjs-size-monitor-shrink">
                    <div class=""></div>
                </div>
            </div>
            <canvas id="myAreaChart" width="669" height="320"
class="chartjs-render-monitor" style="display: block; width: 669px;
height: 320px;"></canvas>
        </div>
    </div>
</div>

<!-- Pie Chart -->
<div class="col-xl-4 col-lg-5">
    <div class="card shadow mb-4">
        <!-- Card Header - Dropdown -->
        <div class="card-header bg-primary py-3 d-flex flex-row
align-items-center justify-content-between">
            <h6 class="m-0 font-weight-bold text-white">Transaksi
Barang</h6>
        </div>
        <!-- Card Body -->

```



```

<div class="card-header bg-warning py-3">
    <h6 class="m-0 font-weight-bold text-white text-center">Stok Barang Minimum</h6>
</div>
<div class="table-responsive">
    <table class="table mb-0 text-center table-striped table-sm">
        <thead>
            <tr>
                <th>Barang</th>
                <th>Stok</th>
                <th>Pasok</th>
            </tr>
        </thead>
        <tbody>
            <?php
            if ($barang_min) :
                foreach ($barang_min as $b) :
                    ?>
                    <tr>
                        <td><?= $b['nama_barang']; ?></td>
                        <td><?= $b['stok']; ?></td>
                        <td>
                            <a href="<?=
base_url('barangmasuk/add/') . $b['id_barang'] ?>" class="btn btn-
warning btn-sm"><i class="fa fa-plus"></i></a>
                        </td>
                    </tr>
                <?php endforeach; ?>
            <?php else : ?>
                <tr>
                    <td colspan="3" class="text-center">

```

```

                Tidak ada barang stok minim
            </td>
        </tr>
    <?php endif; ?>
</tbody>
</table>
</div>
</div>
</div>
<div class="col-md-4">
    <div class="card shadow mb-4">
        <div class="card-header bg-success py-3">
            <h6 class="m-0 font-weight-bold text-white text-center">5 Transaksi Terakhir Barang Masuk</h6>
        </div>
        <div class="table-responsive">
            <table class="table mb-0 table-sm table-striped text-center">
                <thead>
                    <tr>
                        <th>Tanggal</th>
                        <th>Barang</th>
                        <th>Jumlah</th>
                    </tr>
                </thead>
                <tbody>
                    <?php foreach ($transaksi['barang_masuk'] as $tbm) : ?>
                        <tr>
                            <td><strong><?= $tbm['tanggal_masuk'];
?></strong></td>
                            <td><?= $tbm['nama_barang']; ?></td>

```

```

        <td><span class="badge badge-success"><?=
$tbm['jumlah_masuk']; ?></span></td>

    </tr>

    <?php endforeach; ?>
</tbody>
</table>
</div>
</div>
</div>
<div class="col-md-4">
    <div class="card shadow mb-4">
        <div class="card-header bg-danger py-3">
            <h6 class="m-0 font-weight-bold text-white text-
center">5 Transaksi Terakhir Barang Keluar</h6>
        </div>
        <div class="table-responsive">
            <table class="table mb-0 table-sm table-striped text-
center">
                <thead>
                    <tr>
                        <th>Tanggal</th>
                        <th>Barang</th>
                        <th>Jumlah</th>
                    </tr>
                </thead>
                <tbody>
                    <?php foreach ($transaksi['barang_keluar'] as $tbk) :
?>

                        <tr>
                            <td><strong><?= $tbk['tanggal_keluar'];
?></strong></td>

                            <td><?= $tbk['nama_barang']; ?></td>

```

```

        <td><span class="badge badge-danger"><?=
$tbk['jumlah_keluar']; ?></span></td>
    </tr>
    <?php endforeach; ?>
</tbody>
</table>
</div>
</div>
</div>
</div>

```

3. Listing program barang masuk

```

<?= $this->session->flashdata('pesan'); ?>
<div class="card shadow-sm border-bottom-primary">
    <div class="card-header bg-white py-3">
        <div class="row">
            <div class="col">
                <h4 class="h5 align-middle m-0 font-weight-bold text-
primary">
                    Riwayat Data Barang Masuk
                </h4>
            </div>
            <div class="col-auto">
                <a href="<?= base_url('barangmasuk/add') ?>" class="btn
btn-sm btn-primary btn-icon-split">
                    <span class="icon">
                        <i class="fa fa-plus"></i>
                    </span>
                    <span class="text">
                        Input Barang Masuk
                    </span>
                </a>
            </div>
        </div>
    </div>

```

```

</div>

</div>

</div>

<div class="table-responsive">
    <table class="table table-striped w-100 dt-responsive nowrap"
id="dataTable">
        <thead>
            <tr>
                <th>No. </th>
                <th>No Transaksi</th>
                <th>Tanggal Masuk</th>
                <th>Supplier</th>
                <th>Nama Barang</th>
                <th>Jumlah Masuk</th>
                <th>User</th>
                <th>Hapus</th>
            </tr>
        </thead>
        <tbody>
            <?php
                $no = 1;
                if ($barangmasuk) :
                    foreach ($barangmasuk as $bm) :
                        ?>
                        <tr>
                            <td><?= $no++; ?></td>
                            <td><?= $bm['id_barang_masuk']; ?></td>
                            <td><?= $bm['tanggal_masuk']; ?></td>
                            <td><?= $bm['nama_supplier']; ?></td>
                            <td><?= $bm['nama_barang']; ?></td>
                            <td><?= $bm['jumlah_masuk'] . ' '.
$bm['nama_satuan']; ?></td>

```

```

        <td><?= $bm['nama']; ?></td>
        <td>
            <a onclick="return confirm('Yakin ingin
hapus?')" href="<?= base_url('barangmasuk/delete/') .
$bm['id_barang_masuk'] ?>" class="btn btn-danger btn-circle btn-
sm"><i class="fa fa-trash"></i></a>
        </td>
    </tr>
<?php endforeach; ?>
<?php else : ?>
    <tr>
        <td colspan="8" class="text-center">
            Data Kosong
        </td>
    </tr>
<?php endif; ?>
</tbody>
</table>
</div>
</div>

```

4. Listing program barang keluar

```

<?= $this->session->flashdata('pesan'); ?>
<div class="card shadow-sm border-bottom-primary">
    <div class="card-header bg-white py-3">
        <div class="row">
            <div class="col">
                <h4 class="h5 align-middle m-0 font-weight-bold text-
primary">
                    Riwayat Data Barang Keluar
                </h4>
            </div>
        </div>
    </div>

```

```

<div class="col-auto">
    <a href="<?= base_url('barangkeluar/add') ?>" class="btn
btn-sm btn-primary btn-icon-split">
        <span class="icon">
            <i class="fa fa-plus"></i>
        </span>
        <span class="text">
            Input Barang Keluar
        </span>
    </a>
</div>
</div>
</div>
<div class="table-responsive">
    <table class="table table-striped w-100 dt-responsive nowrap"
id="dataTable">
        <thead>
            <tr>
                <th>No. </th>
                <th>No Transaksi</th>
                <th>Tanggal Keluar</th>
                <th>Nama Barang</th>
                <th>Jumlah Keluar</th>
                <th>User</th>
                <th>Hapus</th>
            </tr>
        </thead>
        <tbody>
            <?php
            $no = 1;
            if ($barangkeluar) :
                foreach ($barangkeluar as $bk) :

```



```

        ?>
        <tr>
            <td><?= $no++; ?></td>
            <td><?= $bk['id_barang_keluar']; ?></td>
            <td><?= $bk['tanggal_keluar']; ?></td>
            <td><?= $bk['nama_barang']; ?></td>
            <td><?= $bk['jumlah_keluar'] . ' ' .
$bk['nama_satuan']; ?></td>
            <td><?= $bk['nama']; ?></td>
            <td>
                <a onclick="return confirm('Yakin ingin
hapus?')" href="<?= base_url('barangkeluar/delete/') .
$bk['id_barang_keluar'] ?>" class="btn btn-danger btn-circle btn-
sm"><i class="fa fa-trash"></i></a>
            </td>
        </tr>
    <?php endforeach; ?>
    <?php else : ?>
        <tr>
            <td colspan="7" class="text-center">
                Data Kosong
            </td>
        </tr>
    <?php endif; ?>
</tbody>
</table>
</div>
</div>

```

5. Listing program laporan

```

<div class="row justify-content-center">
    <div class="col-lg-8">

```

```

<div class="card shadow-sm border-bottom-primary">
  <div class="card-header bg-white py-3">
    <h4 class="h5 align-middle m-0 font-weight-bold text-
primary">
      Form Laporan
    </h4>
  </div>
  <div class="card-body">
    <?= $this->session->flashdata('pesan'); ?>
    <?= form_open(); ?>
    <div class="row form-group">
      <label class="col-md-3 text-md-right"
for="transaksi">Laporan Transaksi</label>
      <div class="col-md-9">
        <div class="custom-control custom-radio">
          <input value="barang_masuk" type="radio"
id="barang_masuk" name="transaksi" class="custom-control-
input">
          <label class="custom-control-label"
for="barang_masuk">Barang Masuk</label>
        </div>
        <div class="custom-control custom-radio">
          <input value="barang_keluar" type="radio"
id="barang_keluar" name="transaksi" class="custom-control-
input">
          <label class="custom-control-label"
for="barang_keluar">Barang Keluar</label>
        </div>
        <?= form_error('transaksi', '<span class="text-danger
small">', '</span>'); ?>
      </div>
    </div>
  </div>

```

```

<div class="row form-group">
  <label class="col-lg-3 text-lg-right"
for="tanggal">Tanggal</label>
  <div class="col-lg-5">
    <div class="input-group">
      <input value="<?= set_value('tanggal'); ?>"
name="tanggal" id="tanggal" type="text" class="form-control"
placeholder="Periode Tanggal">
      <div class="input-group-append">
        <span class="input-group-text"><i class="fa fa-
fw fa-calendar"></i></span>
      </div>
    </div>
    <?= form_error('tanggal', '<small class="text-
danger">', '</small>'); ?>
  </div>
</div>
<div class="row form-group">
  <div class="col-lg-9 offset-lg-3">
    <button type="submit" class="btn btn-primary btn-
icon-split">
      <span class="icon">
        <i class="fa fa-print"></i>
      </span>
      <span class="text">
        Cetak
      </span>
    </button>
  </div>
</div>
<?= form_close(); ?>
</div>

```

```

        </div>
    </div>
</div>

```

6. Listing program *user management*

```

<?= $this->session->flashdata('pesan'); ?>
<div class="card shadow-sm mb-4 border-bottom-primary">
    <div class="card-header bg-white py-3">
        <div class="row">
            <div class="col">
                <h4 class="h5 align-middle m-0 font-weight-bold text-
primary">
                    Data User
                </h4>
            </div>
            <div class="col-auto">
                <a href="<?= base_url('user/add') ?>" class="btn btn-sm
btn-primary btn-icon-split">
                    <span class="icon">
                        <i class="fa fa-user-plus"></i>
                    </span>
                    <span class="text">
                        Tambah User
                    </span>
                </a>
            </div>
        </div>
    </div>
</div>
<div class="table-responsive">
    <table class="table table-striped dt-responsive nowrap"
id="dataTable">
        <thead>

```

```

<tr>
    <th width="30">No.</th>
    <th>Foto</th>
    <th>Nama</th>
    <th>Username</th>
    <th>Email</th>
    <th>No. telp</th>
    <th>Role</th>
    <th>Aksi</th>
</tr>
</thead>
<tbody>
    <?php
    $no = 1;
    if ($users) :
        foreach ($users as $user) :
            ?>
            <tr>
                <td><?= $no++; ?></td>
                <td>
                    " class="img-thumbnail rounded-circle">
                </td>
                <td><?= $user['nama']; ?></td>
                <td><?= $user['username']; ?></td>
                <td><?= $user['email']; ?></td>
                <td><?= $user['no_telp']; ?></td>
                <td><?= $user['role']; ?></td>
                <td>
                    <a href="<?= base_url('user/toggle/') .
$user['id_user'] ?>" class="btn btn-circle btn-sm <?=

```

```

$user['is_active'] ? 'btn-secondary' : 'btn-success' ?>" title="<?=
$user['is_active'] ? 'Nonaktifkan User' : 'Aktifkan User' ?>"><i
class="fa fa-fw fa-power-off"></i></a>

        <a href="<?= base_url('user/edit/') .
$user['id_user'] ?>" class="btn btn-circle btn-sm btn-warning"><i
class="fa fa-fw fa-edit"></i></a>

        <a onclick="return confirm('Yakin ingin
menghapus data?')" href="<?= base_url('user/delete/') .
$user['id_user'] ?>" class="btn btn-circle btn-sm btn-danger"><i
class="fa fa-fw fa-trash"></i></a>

    </td>

</tr>

<?php endforeach;
else : ?>

<tr>

    <td colspan="8" class="text-center">Silahkan
tambahkan user baru</td>

</tr>

<?php endif; ?>

</tbody>

</table>

</div>

</div>

```

1.1.2 Implementasi Sistem

Implementasi sistem adalah tahap penerapan sistem yang akan dilakukan jika sistem telah disetujui termasuk program yang telah dibuat pada tahap perancangan sistem agar siap untuk dioperasikan. Adapun waktu dan tempat penerapan sistem yang sudah dibuat sebagai berikut :

1. Waktu dan Tempat Implementasi

Tempat : PT. Visi Karya Prakarsa

Alamat : Bandung Barat, Jl. Jupiter Barat No.37, Sekejati, Kec.

Buahbatu.

Waktu : Bulan Agustus

1.1.3 Spesifikasi Sistem

Spesifikasi sistem akan menjelaskan tentang spesifikasi perangkat keras dan perangkat lunak yang digunakan dalam pengimplementasian aplikasi *inventory*.

1) Spesifikasi perangkat keras

Dibawah ini merupakan spesifikasi perangkat keras yang digunakan sebagai berikut :

Tabel 5.1 Spesifikasi perangkat keras

Prosesor	Intel with integrated graphics, 1066 Mhz, dual core(s), 12 Logical Processor(s)
RAM	4GB
HDD	320GB

2) Spesifikasi perangkat lunak

Di bawah ini merupakan spesifikasi perangkat lunak yang digunakan sebagai berikut:

Tabel 5.2 Spesifikasi perangkat lunak

Sistem Operasi	Windows 10
Database	MySQL Versi 7.7
Bahasa Pemrograman	PHP Versi 7.7.2, HTML, CSS

1.1.4 Instalasi Sistem

Instalasi sistem ini memuat penjelasan mengenai langkah-langkah yang harus dilakukan untuk instalasi aplikasi dan instalasi *database*.

1. Instalasi Aplikasi

1) XAMPP

XAMPP ini dibutuhkan untuk menjalankan aplikasi dengan memanggil localhost dan juga untuk melakukan running MySQL yang berada didalam XAMPP. Berikut di bawah ini merupakan tahapan-tahapan instalasi XAMPP:

- a. Unduh installer aplikasi XAMPP melalui web apachefriends.org
- b. Lakukan double klik pada file XAMPP yang sudah diunduh
- c. Klik *next* pada jendela installer
- d. Pilih komponen yang akan diinstal atau biarkan default untuk menginstal keseluruhan. Disarankan untuk menginstal keseluruhan
- e. Pilih folder instalasi, lalu klik *next*
- f. Pilih bahasa yang ingin digunakan, lalu klik *next*
- g. Jalankan instalasi dengan mengklik *next* pada jendela berikutnya
- h. Tunggu hingga proses instalasi selesai
- i. Setelah instalasi selesai, klik finish dan XAMPP siap digunakan.

2) Browser (Google Chrome)

Browser ini berguna untuk mengaktifkan dan menjalankan aplikasi keuangan yang sudah dibuat. Berikut di bawah ini merupakan tahapan-tahapan instalasi Browser Google Chrome:

- a. Unduh installer Google Chrome
- b. Jika diminta, klik jalankan atau simpan
- c. Jika memilih simpan maka lakukan klik dua kali pada tombol download untuk memulai proses instalasi
- d. Tunggu hingga proses instalasi selesai
- e. Setelah instalasi selesai, Google Chrome sudah bisa digunakan.

2. Instalasi *Database*

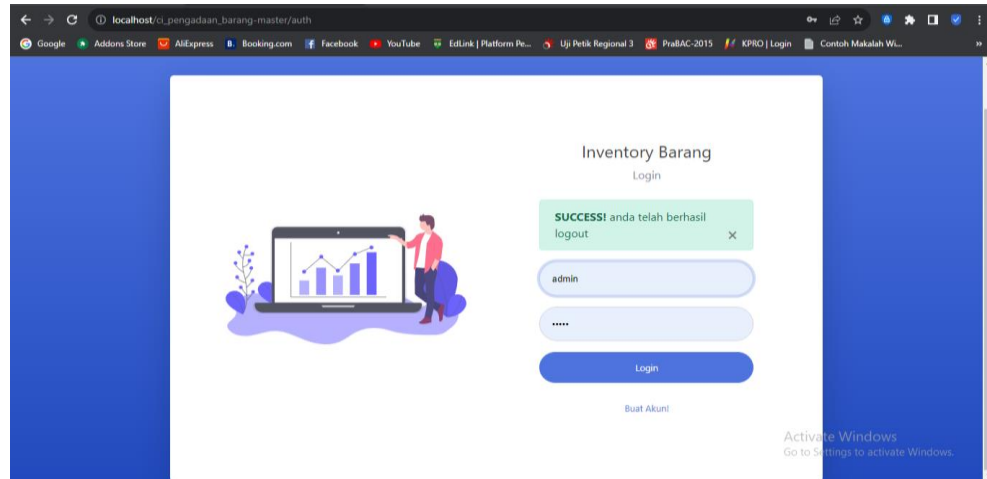
Berkaitan dengan *database* yang digunakan pada aplikasi *inventory* merupakan MySQL dan MySQL itu sendiri sudah otomatis didalam aplikasi XAMPP maka yang harus dilakukan untuk menjalankan *database* MySQL hanyalah dengan mengaktifkan XAMPP dan memanggil PHPMyAdmin di *browser*.

1.1.5 Menjalankan Sistem

Pada bagian ini akan dijelaskan bagaimana cara-cara menjalankan sistem aplikasi *inventory*

1. Halaman *login*

Sebelum memasuki aplikasi *inventory*, *user* diharuskan untuk melakukan login dengan menggunakan email sebagai *username* dan *password* aplikasi.

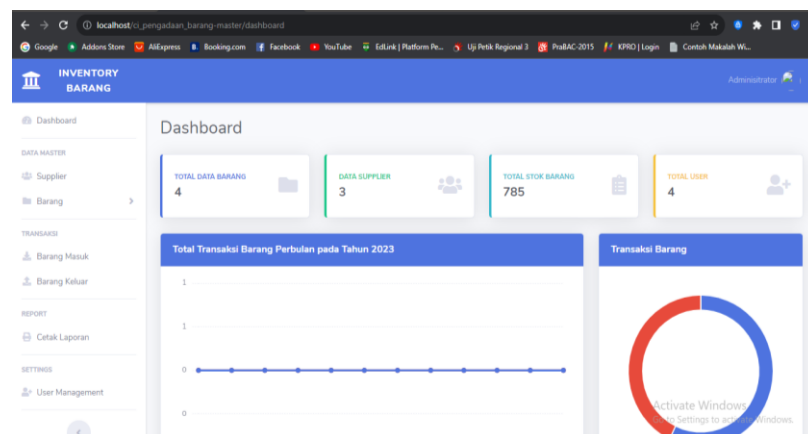


Gambar 5.1 Halaman Login

Pada halaman *login*, pengguna memiliki opsi untuk memasukkan *username* dan *password* yang telah terdaftar dalam sistem guna mengakses halaman berikutnya. Jika pengguna belum memiliki akun, mereka dapat membuat akun baru dan kemudian menunggu proses verifikasi untuk dapat mengakses sistem.

2. Halaman *Dashboard*

Setelah admin memasuki aplikasi maka tampilan yang pertama kali muncul yaitu halaman *dashboard* sebagai halaman utama, pada halaman *dashboard* aplikasi keuangan memuat laporan barang dan dapat disesuaikan periodenya sesuai kebutuhan

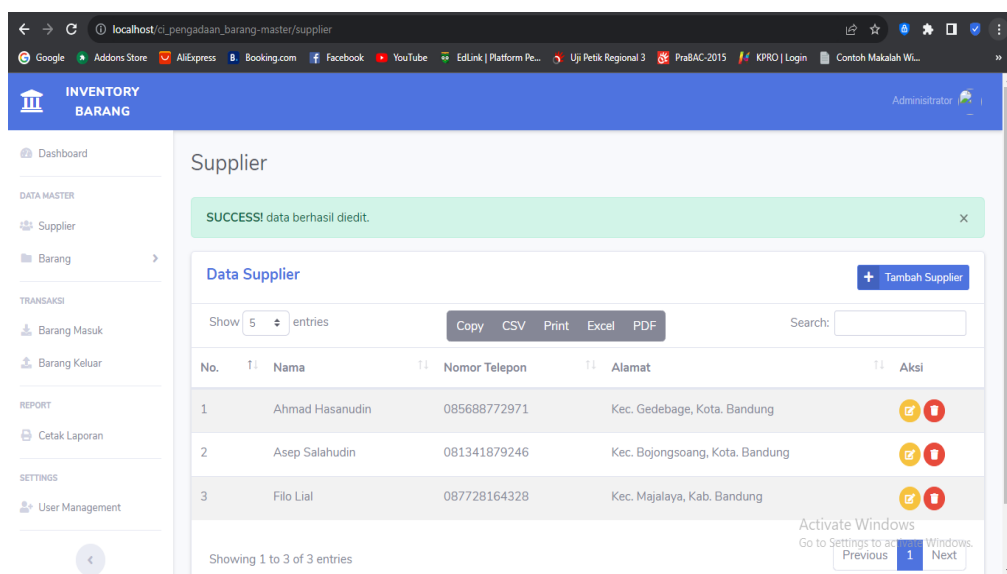


Gambar 5.2 Halaman dashboard

Pada halaman *dashboard*, setelah berhasil melakukan *login*, pengguna dapat mengakses tampilan yang menampilkan aktivitas pengelolaan barang dalam periode waktu tertentu. Selain itu, halaman tersebut juga menyertakan grafik yang memvisualisasikan total transaksi barang setiap bulan beserta perkembangannya.

3. Halaman *Supplier*

Halaman *supplier* memuat tabel *supply* dari data laporan barang yang berisi pengambilan barang oleh *supplier* sebagai data yang diolah dalam aplikasi *inventory*.



Gambar 5.3 Halaman *supplier*

Pada halaman *supplier*, pengguna dapat mengakses kolom data lengkap mengenai *supplier*, dan terdapat beberapa fitur yang memungkinkan pengguna untuk menambahkan data *supplier* baru, mengedit informasi *supplier* yang ada, serta menghapus data *supplier* yang tidak diperlukan.

4. Halaman Tambah Tabel *Supplier*

Pada halaman *supplier*, admin dapat melakukan tambah data dengan cara mengklik tombol tambah kemudian sistem akan memunculkan form tambah jenis seperti di bawah ini.

Gambar 5.4 Halaman tambah supplier

Pada halaman tambah *supplier*, pengguna memiliki opsi untuk mengisi data seperti nama *supplier*, nomor telepon, dan alamat lengkap dari pemasok. Setelah data-data tersebut diisi dengan benar dan valid, pengguna dapat mengklik tombol "Simpan" untuk menambahkan data *supplier* baru ke dalam sistem.

5. Halaman Edit Tabel *Supplier*

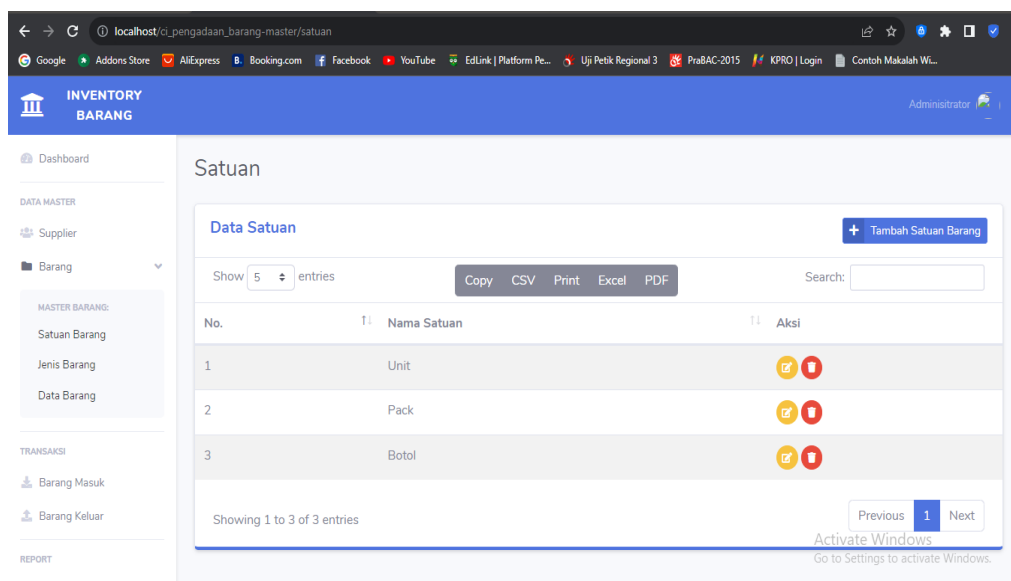
Pada halaman *supplier*, admin dapat melakukan edit data dengan cara mengklik tombol edit pada baris data yang dipilih kemudian sistem akan memunculkan form ubah jenis seperti di bawah ini.

Gambar 5.5 Halaman edit tabel supplier

Pada halaman edit supplier, prosesnya serupa dengan tambah supplier. Pengguna hanya perlu mengganti data yang akan diubah. Setelah perubahan dilakukan dan data telah diperbarui sesuai kebutuhan, pengguna dapat mengklik tombol "Simpan" untuk menyimpan perubahan tersebut.

6. Halaman Data Satuan Barang

Halaman satuan barang memuat beberapa barang berupa unit, pack atau yang lainnya seperti gambar dibawah ini :



Gambar 5.6 Halaman data satuan barang

Pada halaman satuan barang, terdapat kolom yang mencantumkan informasi mengenai satuan dalam bentuk unit, pack, dan pcs. Dalam kolom tersebut, terdapat ikon yang memungkinkan pengguna untuk melakukan pengeditan (edit) dan penghapusan (hapus) data satuan barang yang ada. Selain itu, pengguna juga memiliki opsi untuk menambahkan data satuan barang baru jika diperlukan.

7. Halaman Tambah Satuan Barang

Pada halaman satuan barang, admin dapat melakukan tambah data dengan cara mengklik tombol tambah kemudian sistem akan memunculkan *form* tambah rincian seperti di bawah ini.

Gambar 5.7 Halaman tambah satuan barang

Pada halaman tambah satuan barang, pengguna hanya perlu menambahkan nama satuan barang yang baru. Setelah nama satuan tersebut dimasukkan, pengguna dapat menyimpannya, dan data satuan barang baru akan secara otomatis ditambahkan ke dalam sistem.

8. Halaman Ubah Satuan Barang

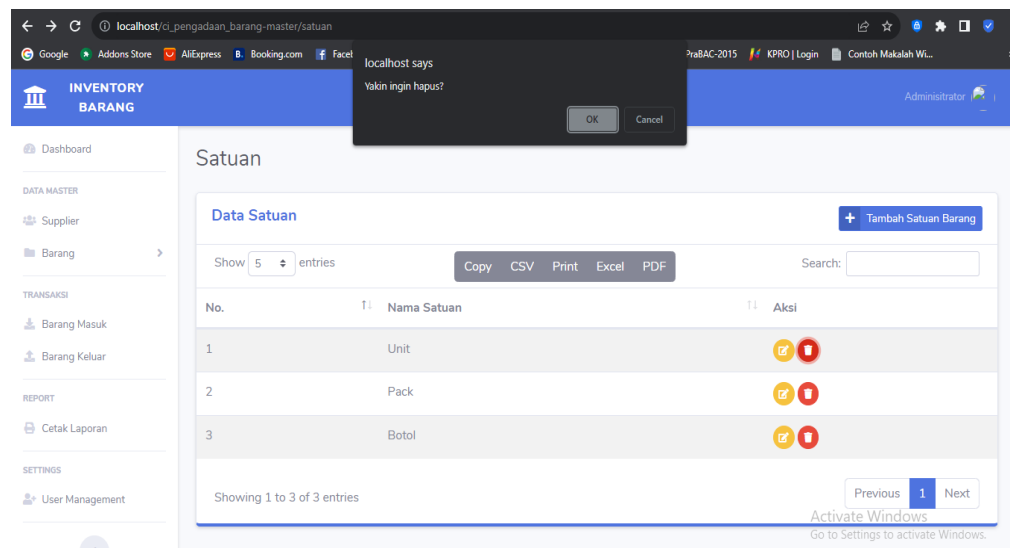
Pada halaman satuan barang, admin dapat melakukan ubah data dengan cara mengklik tombol edit pada baris data yang dipilih kemudian sistem akan memunculkan *form* ubah rincian seperti di bawah ini.

Gambar 5.8 Halaman ubah satuan barang

Pada halaman ubah satuan barang, pengguna hanya perlu mengubah data nama satuan menjadi yang benar sesuai kebutuhan. Setelah perubahan dilakukan, pengguna dapat menyimpan data tersebut, dan data satuan barang akan secara otomatis diperbarui dalam sistem.

9. Halaman Hapus Satuan Barang

Pada halaman satuan barang, admin dapat melakukan hapus data dengan cara mengklik tombol *delete* seperti di bawah ini.

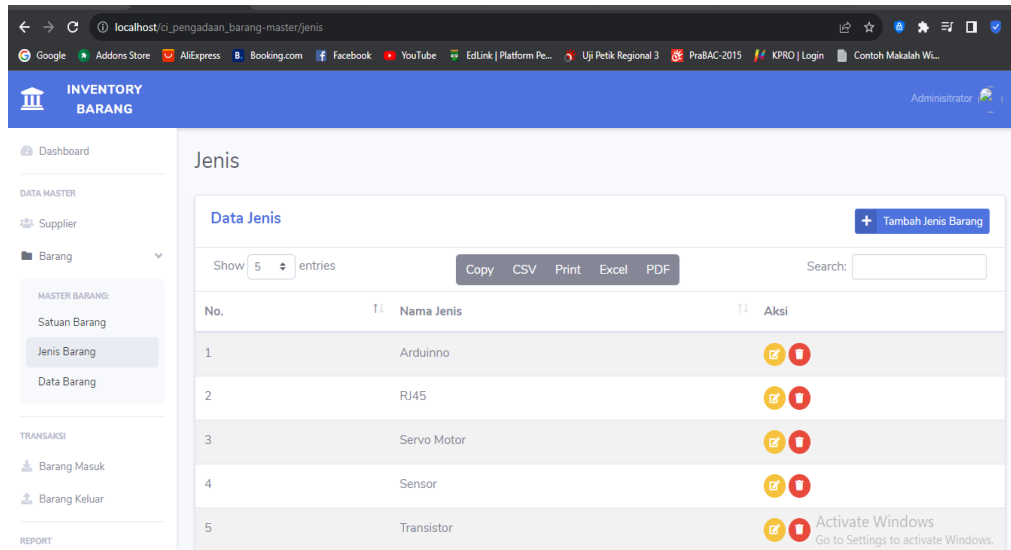


Gambar 5.9 Halaman hapus satuan barang

Pada halaman penghapusan satuan barang, pengguna dapat menghapus data satuan barang dengan mengklik ikon sampah. Selanjutnya, akan muncul notifikasi yang bertujuan untuk memastikan bahwa pengguna benar-benar ingin menghapus data tersebut. Untuk mengonfirmasi penghapusan, pengguna dapat mengklik opsi "OK" jika yakin akan menghapusnya.

10. Halaman Jenis Barang

Pada halaman jenis barang berfungsi untuk melihat, mengedit, menambah, dan menghapus beberapa jenis barang atau komponen dalam periode tertentu sesuai dengan kebutuhan seperti dibawah ini.

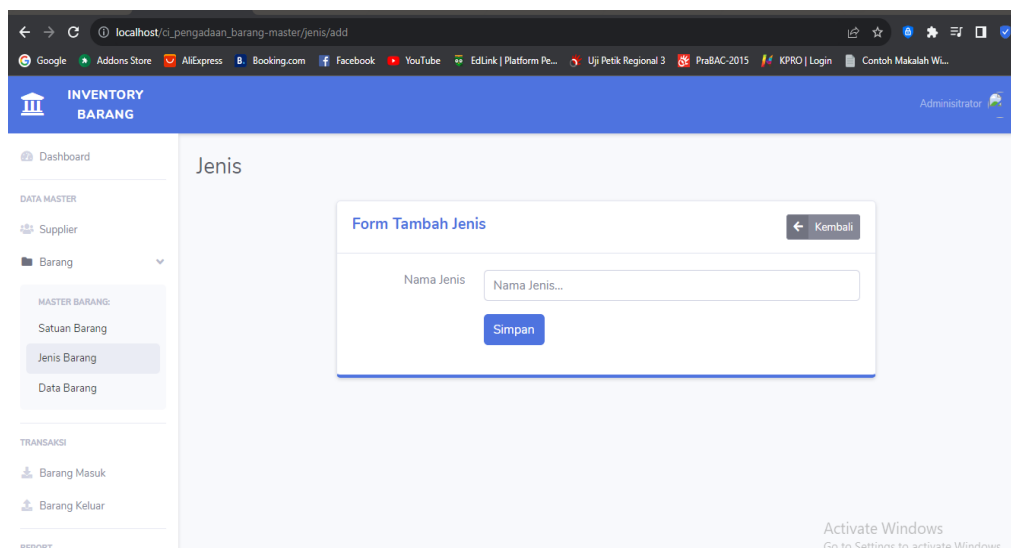


Gambar 5.10 Halaman jenis barang

Pada halaman jenis barang, setelah data sudah terdaftar, pengguna memiliki kemampuan untuk menambahkan, mengubah, dan menghapus data jenis barang sesuai dengan ikon yang tersedia.

11. Halaman Tambah Jenis Barang

Pada halaman jenis barang, admin dapat melakukan tambah data dengan cara mengklik tombol tambah kemudian sistem akan memunculkan *form* tambah rincian seperti di bawah ini.

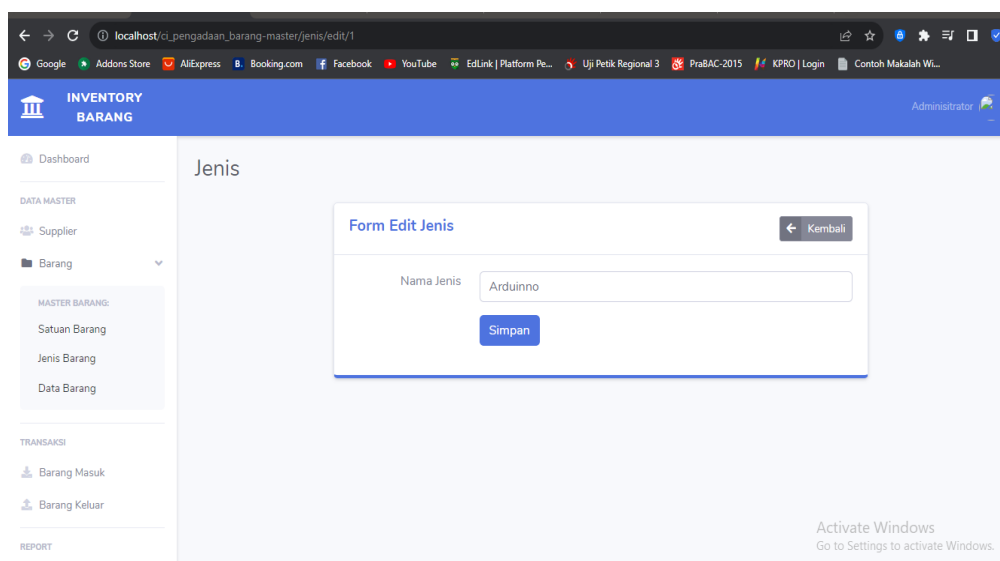


Gambar 5.11 Halaman tambah jenis barang

Dalam formulir penambahan jenis barang, pengguna hanya perlu memasukkan nama jenis barang yang ingin ditambahkan, dan setelah yakin dengan data yang dimasukkan, pengguna dapat mengklik tombol "Simpan". Data jenis barang akan secara otomatis ditambahkan ke dalam sistem setelah proses ini.

12. Halaman Edit Jenis Barang

Pada halaman jenis barang, admin dapat melakukan ubah data dengan cara mengklik tombol edit pada baris data yang dipilih kemudian sistem akan memunculkan *form* ubah rincian seperti di bawah ini.

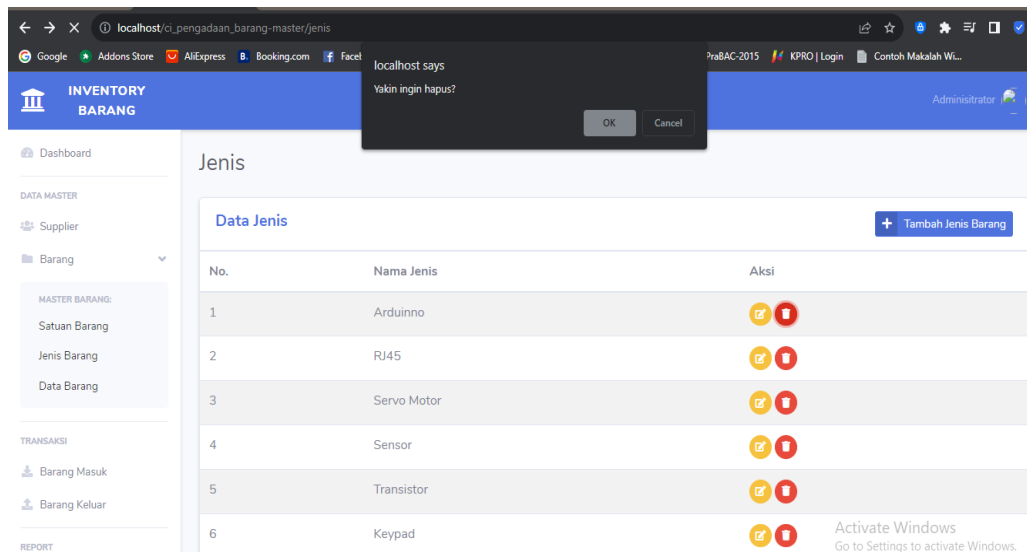


Gambar 5.12 Halaman edit jenis barang

Dalam formulir halaman edit jenis barang, pengguna dapat melakukan perubahan atau penggantian jenis barang dengan yang baru. Setelah perubahan dilakukan, pengguna dapat mengklik tombol "Simpan", dan data jenis barang akan diperbarui secara otomatis dalam sistem.

13. Halaman Hapus Jenis Barang

Pada halaman jenis barang, admin dapat melakukan hapus data dengan cara mengklik tombol delete pada baris data yang dipilih kemudian sistem akan memunculkan notifikasi untuk memvalidasi apakah admin yakin untuk melakukan penghapusan data atau tidak seperti di bawah ini.

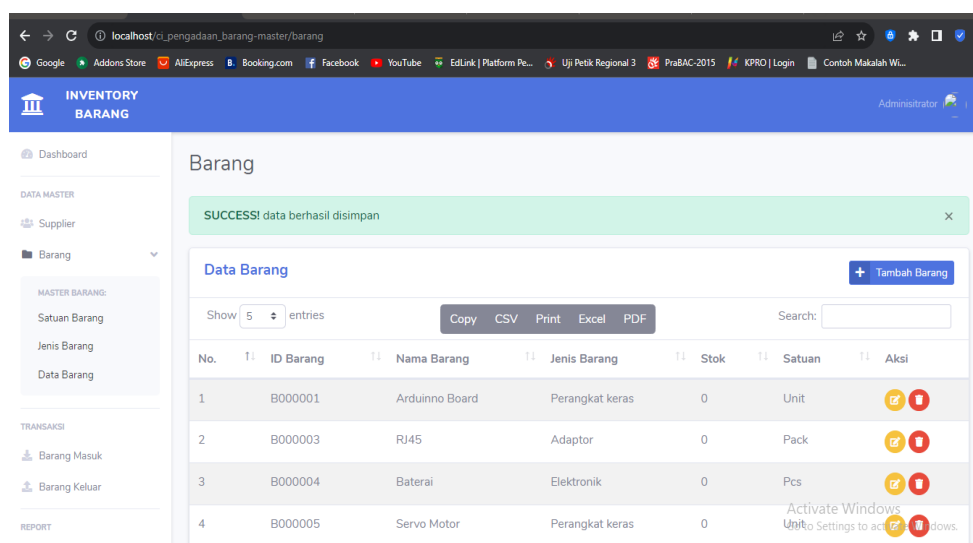


Gambar 5.13 Halaman hapus jenis barang

Pada halaman penghapusan jenis barang, pengguna dapat menghapus data jenis barang dengan mengklik ikon sampah. Selanjutnya, akan muncul notifikasi yang bertujuan untuk memastikan bahwa pengguna benar-benar ingin menghapus data tersebut. Untuk mengonfirmasi penghapusan, pengguna dapat mengklik opsi "OK" jika yakin akan menghapusnya.

14. Halaman Data Barang

Pada halaman data barang berfungsi untuk melihat keseluruhan barang dari beberapa satuan dan jenis barang diatas dalam periode tertentu sesuai dengan kebutuhan.



Gambar 5.14 Halaman data barang

15. Halaman Tambah Data Barang

Pada halaman data barang, admin dapat melakukan tambah data dengan cara mengklik tombol tambah kemudian sistem akan memunculkan *form* tambah rincian seperti di bawah ini.

Gambar 5.15 Halaman tambah data barang

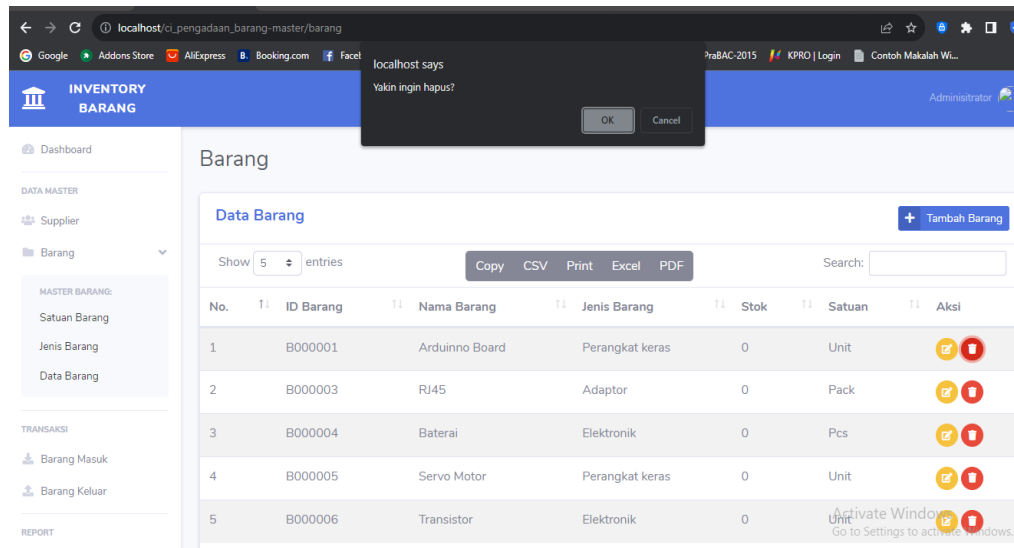
16. Halaman Ubah Data Barang

Pada halaman data barang, admin dapat melakukan ubah data dengan cara mengklik tombol edit pada baris data yang dipilih kemudian sistem akan memunculkan *form* ubah rincian seperti di bawah ini.

Gambar 5.16 Halaman ubah data barang

17. Halaman Hapus Data Barang

Pada halaman data barang, admin dapat melakukan hapus data dengan cara mengklik tombol *delete* pada baris data yang dipilih kemudian sistem akan memunculkan notifikasi untuk memvalidasi apakah admin yakin untuk melakukan penghapusan data atau tidak seperti di bawah ini.

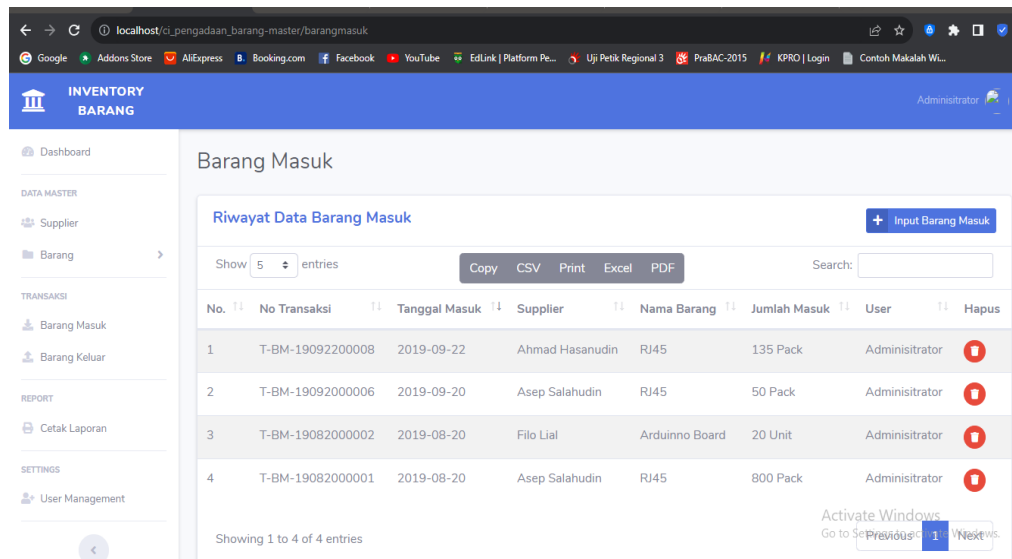


Gambar 5.17 Halaman hapus data barang

Pada halaman penghapusan data barang, pengguna dapat menghapus data barang dengan mengklik ikon sampah. Selanjutnya, akan muncul notifikasi yang bertujuan untuk memastikan bahwa pengguna benar-benar ingin menghapus data tersebut. Untuk mengonfirmasi penghapusan, pengguna dapat mengklik opsi "OK" jika yakin akan menghapusnya.

18. Halaman Riwayat Data Barang Masuk

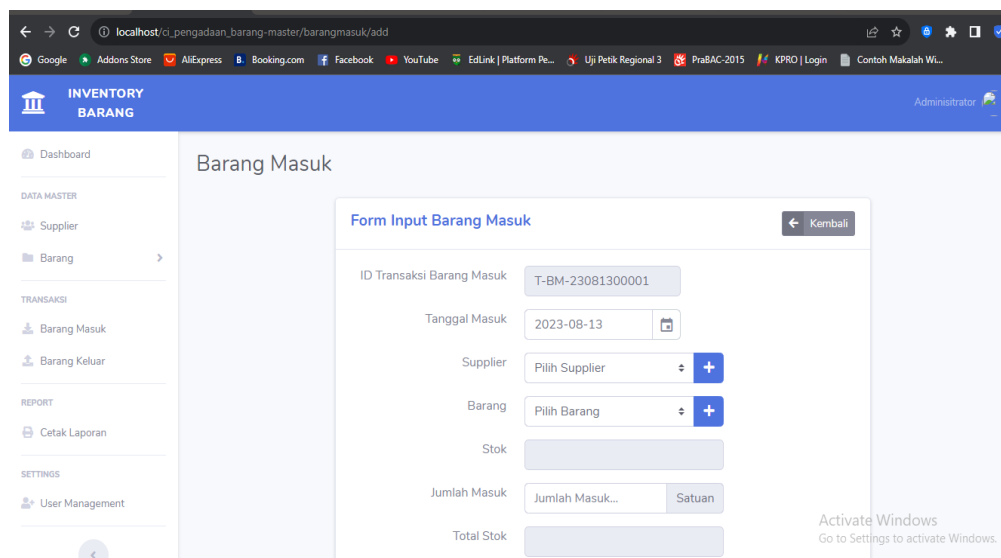
Halaman ini memuat riwayat tentang barang masuk dengan periode tertentu dan berisi tentang data data barang yang sudah masuk lengkap dengan identitas *supplier*.



Gambar 5.18 Halaman riwayat data barang masuk

19. Halaman Input Barang Masuk

Pada halaman barang masuk, admin dapat melakukan input data dengan cara mengklik tombol input barang masuk pada baris data yang dipilih kemudian sistem akan memunculkan halaman untuk membuat admin mengisi inputan barang seperti di bawah ini.

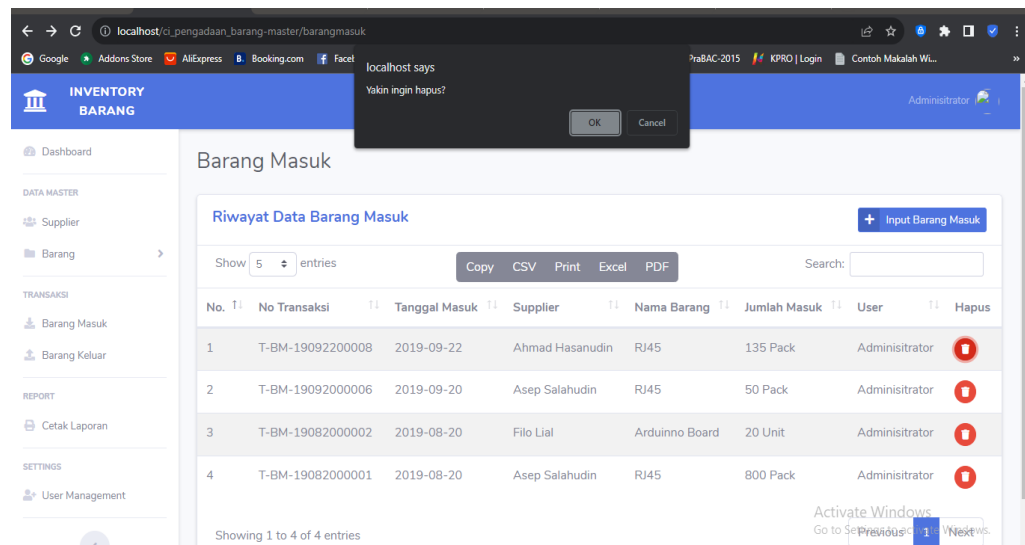


Gambar 5.19 Halaman input barang masuk

20. Halaman Hapus Barang Masuk

Admin dapat melakukan hapus data dengan cara mengklik tombol *delete* pada baris data yang dipilih kemudian sistem akan memunculkan notifikasi

untuk memvalidasi apakah admin yakin untuk melakukan penghapusan data atau tidak seperti di bawah ini.

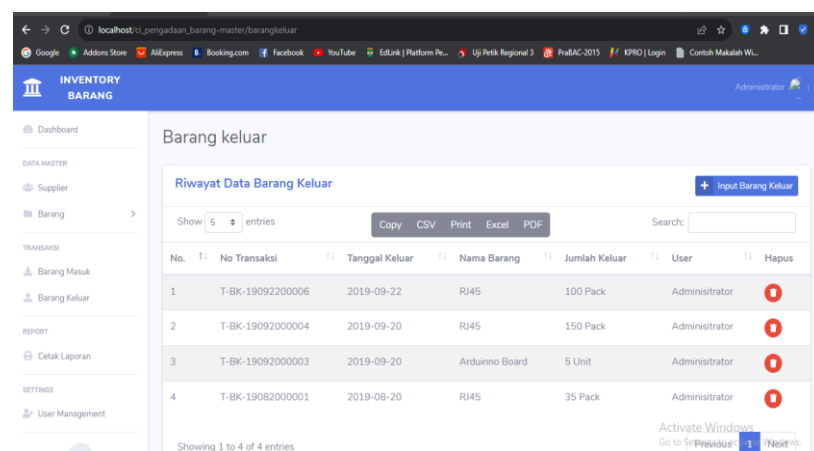


Gambar 5.20 Halaman hapus barang masuk

Pada halaman penghapusan riwayat data barang masuk, pengguna dapat menghapus data barang masuk dengan mengklik ikon sampah. Selanjutnya, akan muncul notifikasi yang bertujuan untuk memastikan bahwa pengguna benar-benar ingin menghapus data tersebut. Untuk mengonfirmasi penghapusan, pengguna dapat mengklik opsi "OK" jika yakin akan menghapusnya.

21. Halaman Riwayat Data Barang Keluar

Pada halaman ini memuat tentang pengeluaran barang keluar tiap periode tertentu dengan berisi beberapa kolom seperti dibawah ini.



Gambar 5.21 Halaman riwayat barang keluar

Pada halaman riwayat data barang keluar, terdapat informasi mengenai transaksi, termasuk nomor transaksi, tanggal keluarnya barang, dan jumlah barang yang keluar. Dalam kolom tersebut, terdapat ikon yang memungkinkan pengguna untuk menghapus data transaksi. Penting untuk dicatat bahwa data tersebut bersifat real-time dan tidak dapat diedit setelah transaksi terjadi.

22. Halaman Input Barang Keluar

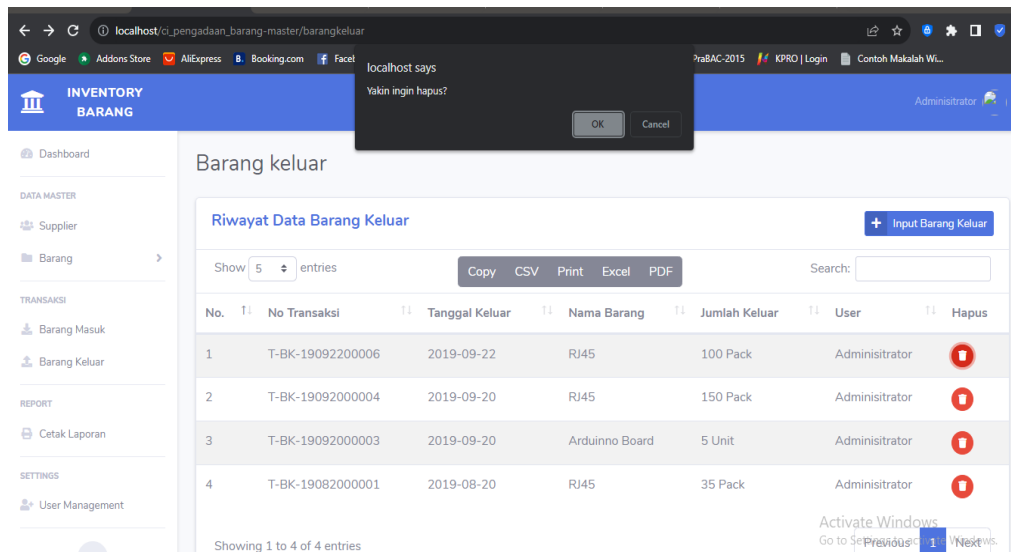
Pada halaman barang keluar, admin dapat melakukan input barang keluar dengan cara mengklik tombol input barang keluar pada baris data yang dipilih kemudian sistem akan memunculkan halaman untuk membuat admin mengisi inputan barang seperti di bawah ini.

Gambar 5.22 Halaman input barang keluar

Dalam formulir input barang keluar, pengguna dapat memasukkan tanggal periode pengeluaran barang dan selanjutnya mengisi jumlah barang yang keluar dari stok. Setelah data tersebut dimasukkan, pengguna dapat menyimpannya, dan data barang keluar yang baru akan otomatis ditampilkan dalam kolom yang sesuai pada halaman tersebut.

23. Halaman Delete Barang Keluar

Pada halaman barang keluar, admin dapat melakukan hapus data dengan cara mengklik tombol *delete* pada baris data yang dipilih kemudian sistem akan memunculkan notifikasi untuk memvalidasi apakah admin yakin untuk melakukan penghapusan data atau tidak seperti di bawah ini.

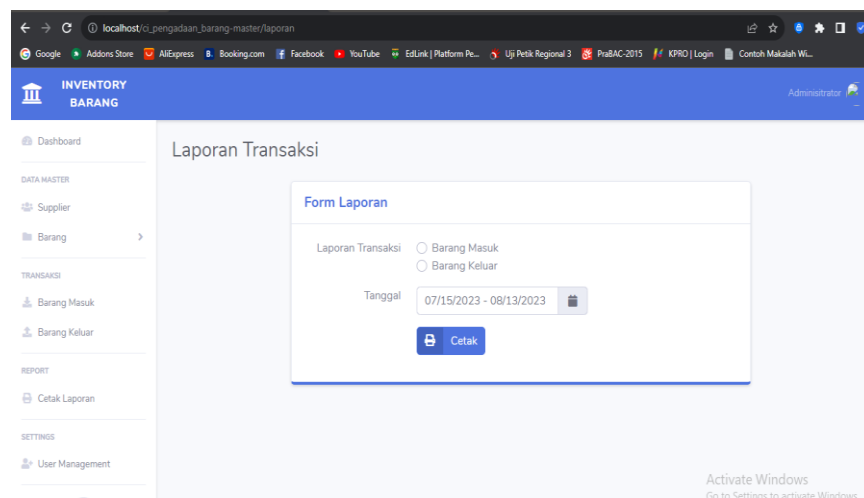


Gambar 5.23 Halaman hapus barang keluar

Pada halaman penghapusan transaksi data barang keluar, pengguna dapat menghapus data barang keluar dengan mengklik ikon sampah. Selanjutnya, akan muncul notifikasi yang bertujuan untuk memastikan bahwa pengguna benar-benar ingin menghapus data tersebut. Untuk mengonfirmasi penghapusan, pengguna dapat mengklik opsi "OK" jika yakin akan menghapusnya.

24. Halaman Transaksi Laporan

Transaksi laporan pada halaman aplikasi *inventory* bisa digunakan sesuai kebutuhan tanggal yang diinginkan dan langsung dicetak berupa file pdf pada kolom terbawah seperti berikut.

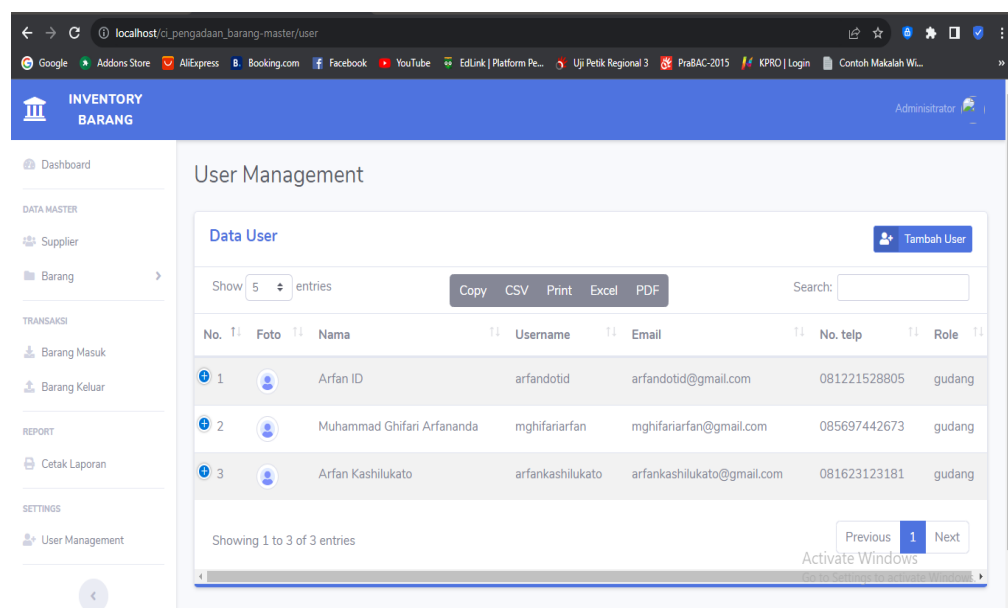


Gambar 5.24 Halaman transaksi laporan

Di halaman laporan transaksi, pengguna mengisi formulir laporan yang mencakup periode transaksi barang masuk atau barang keluar, serta memasukkan tanggal yang relevan. Setelah mengisi data laporan sesuai kebutuhan, pengguna dapat mengklik opsi "cetak" pada halaman tersebut untuk mencetak laporan sesuai dengan periode yang telah ditentukan.

25. Halaman *User Management*

Halaman *user management* ini memuat untuk beberapa admin yang akan mengelola aplikasi *inventory* dengan identitas email, nomor telepon dan role atau posisi dibagian gudang.



Gambar 5.25 Halaman *user management*

Pada halaman manajemen pengguna (*user management*), terdapat data mengenai pengguna yang menggunakan aplikasi inventaris, dan terdapat fitur untuk memasukkan data pengguna baru. Selain itu, pada kolom tersebut, terdapat juga opsi untuk mengatur peran (*role*) atau posisi pengguna, yang dapat diaktifkan atau dinonaktifkan sesuai kebutuhan dalam sistem.

1.2 Pengujian

Setelah tahap implementasi dalam proses perancangan dan pembangunan aplikasi selesai, langkah berikutnya adalah tahap pengujian. Pada tahap ini, tujuannya adalah untuk melakukan evaluasi terhadap kesesuaian dan kinerja

aplikasi yang telah dikembangkan berdasarkan analisis dan perancangan yang telah dilakukan sebelumnya.

Tahap pengujian ini dilakukan untuk memverifikasi apakah aplikasi yang dibuat memenuhi persyaratan fungsional dan non-fungsional yang telah ditentukan sebelumnya. Aktivitas pengujian melibatkan serangkaian langkah yang bertujuan untuk mengidentifikasi kelemahan, kesalahan, atau masalah kinerja yang mungkin ada dalam aplikasi.

Hasil dari tahap pengujian akan memberikan pemahaman yang lebih jelas tentang seberapa baik aplikasi yang dikembangkan berjalan dan berfungsi. Setelah pengujian selesai, perbaikan atau penyesuaian mungkin diperlukan sebelum aplikasi benar-benar siap untuk peluncuran.

1.2.1 Uji Validitas

Berikut adalah tabel yang akan menjelaskan bagaimana pengujian validitas pada aplikasi inventory dengan hasil pengujian seperti tabel dibawah :

Tabel 5.3 Uji Validitas

No.	Item Uji	Skenario Pengujian	Hasil yang diharapkan	Hasil Pengujian	Kesimpulan
1.	<i>Form login</i>	Mengisi username dan password	Admin dapat masuk ke aplikasi inventory	Sesuai harapan	Valid
2.	Halaman <i>dashboard</i>	Klik login kemudian masuk ke halaman dashboard	Admin dapat mengakses halaman dashboard	Sesuai harapan	Valid
3.	Halaman supplier	Klik menu supplier dibaris kiri	Admin dapat melihat tampilan supplier	Sesuai harapan	Valid

4.	Tambah data supplier	Klik tambah pada menu supplier	Admin dapat menambahkan data supplier	Sesuai harapan	Valid
5.	Edit data supplier	Klik edit pada baris data yang dipilih	Admin dapat melakukan edit data supplier	Sesuai harapan	Valid
6.	Hapus data supplier	Klik hapus pada baris data yang dipilih	Admin dapat menghapus data supplier	Sesuai harapan	Valid
7.	Halaman barang	Klik menu barang pada aplikasi	Admin dapat melihat tiga kolom pada halaman barang seperti satuan barang, jenis barang, dan data barang	Sesuai harapan	Valid
8.	Halaman satuan barang	Klik menu satuan barang pada aplikasi	Admin dapat mengakses data satuan barang	Sesuai harapan	Valid
9.	Halaman jenis barang	Klik menu jenis barang	Admin dapat mengakses halaman jenis barang	Sesuai harapan	Valid
10.	Halaman data barang	Klik halaman data barang	Admin dapat mengakses halaman data barang	Sesuai harapan	Valid

11.	Tambah data barang	Klik tambah pada menu data barang	Admin dapat menambahkan data barang	Sesuai harapan	Valid
12.	Edit data barang	Klik edit pada baris data yang dipilih	Admin dapat melakukan edit data barang	Sesuai harapan	Valid
13.	Hapus data barang	Klik hapus pada baris data yang dipilih	Admin dapat menghapus data barang	Sesuai harapan	Valid
14.	Halaman transaksi barang masuk	Klik menu barang masuk pada kolom transaksi	Admin data mengakses transaksi barang masuk setiap periodenya	Sesuai harapan	Valid
15.	Tambah barang masuk	Klik tambah pada menu barang masuk	Admin dapat menambahkan data barang masuk.	Sesuai harapan	Valid
16.	Hapus barang masuk	Klik hapus pada baris data yang dipilih	Admin dapat menghapus data barang masuk.	Sesuai harapan	Valid
17.	Halaman transaksi barang keluar	Klik menu barang keluar pada kolom transaksi	Admin data mengakses transaksi barang keluar setiap periodenya	Sesuai harapan	Valid

18.	Tambah barang keluar	Klik tambah pada menu barang keluar	Admin dapat menambahkan data barang keluar	Sesuai harapan	Valid
19.	Hapus barang keluar	Klik hapus pada baris data yang dipilih	Admin dapat menghapus data barang keluar	Sesuai harapan	Valid
20.	Halaman cetak laporan	Klik tombol cetak laporan	Admin dapat mencetak laporan barang yang ada pada halaman laporan	Sesuai harapan	Valid
21.	Halaman user management	Klik menu user management	Admin dapat melihat beberapa user atau pengguna yang menggunakan aplikasi inventory	Sesuai harapan	Valid
22.	Tambah user	Klik tombol tambah pada baris menu user	Admin dapat menambahkan data user management	Sesuai harapan	Valid
23.	Logout	Klik logout	Admin dapat keluar dari aplikasi inventory	Sesuai harapan	Valid

Berdasarkan Tabel 5.1 yang menggambarkan tabulasi hasil uji coba, dapat disimpulkan bahwa implementasi pengujian terhadap aplikasi telah berhasil mencapai tingkat kesuksesan dan kesesuaian dengan rencana serta analisis yang telah direncanakan sebelumnya. Seluruh fitur yang diimplementasikan beroperasi secara optimal sesuai dengan peran dan fungsi yang telah diintendedkan, yang sesuai dengan visi penulis dalam penulisan skripsi ini.

5.2.2 Uji Reliabilitas

Uji reliabilitas bertujuan untuk memastikan bahwa hasil yang diperoleh dari kuisioner adalah akurat dan dapat dipercaya. Berikut adalah deskripsi mengenai uji reliabilitas dalam menggunakan kuisioner:

- a) Tujuan Uji Reliabilitas: Tujuan utama dari uji reliabilitas adalah untuk mengukur sejauh mana kuisioner yang digunakan dalam penelitian ini dapat menghasilkan hasil yang konsisten dan dapat diandalkan. Dalam konteks aplikasi *inventory*, ini berarti memastikan bahwa kuisioner dapat secara tepat mengukur variabel-variabel yang terkait dengan pengelolaan stok, transaksi, atau aspek lain dalam sistem *inventory*.
- b) Metode Pengumpulan Data: Responden penelitian akan diminta untuk mengisi kuisioner yang berisi pertanyaan-pertanyaan terkait dengan variabel yang diteliti. Data dari kuisioner tersebut akan digunakan untuk analisis reliabilitas. Berikut pertanyaan yang diajukan dalam kuisioner:

Tabel 5.4 Hasil kuisioner responden

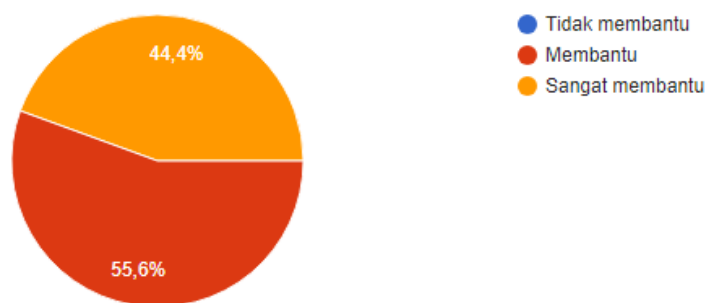
No	Pertanyaan	Pilihan
1	Apakah aplikasi <i>inventory</i> ini membantu dalam mengoptimalkan pengelolaan stok barang?	<ul style="list-style-type: none"> - Tidak membantu - Membantu - Sangat membantu
2	Apakah aplikasi ini membantu dalam mengidentifikasi kebutuhan stok barang secara lebih efisien?	<ul style="list-style-type: none"> - Tidak membantu - Membantu - Sangat membantu
3	Bagaimana tingkat kepuasan terhadap kemampuan aplikasi ini dalam memberikan	<ul style="list-style-type: none"> - Tidak puas - Cukup puas

	rekomendasi pengisian stok berdasarkan data historis?	- Puas sekali
4	Seberapa mudah untuk memasukkan data barang baru ke dalam sistem?	- Tidak mudah - Cukup mudah - Sangat mudah
5	Apakah aplikasi ini telah membantu mengurangi risiko kekurangan stok atau kelebihan stok yang tidak perlu?	- Tidak membantu - Membantu - Sangat membantu
6	Apakah penggunaan aplikasi ini telah meningkatkan efisiensi proses pengadaan barang?	- Tidak efisien - Efisien - Sangat efisien
7	Seberapa baik aplikasi ini dalam mengoptimalkan alokasi stok berdasarkan lokasi atau gudang penyimpanan?	- Tidak optimal - Cukup optimal - Sangat optimal

c) Analisis Data: Data yang dikumpulkan dari kuisioner akan dianalisis menggunakan metode statistik yang relevan, seperti Cronbach's Alpha. Nilai Cronbach's Alpha digunakan untuk mengukur konsistensi internal dalam kuisioner dalam bentuk grafik. Nilai yang lebih tinggi menunjukkan konsistensi yang lebih baik antara item-item dalam kuisioner. Berikut adalah jawaban dari pertanyaan-pertanyaan :

Apakah aplikasi inventory ini membantu dalam mengoptimalkan pengelolaan stok barang?

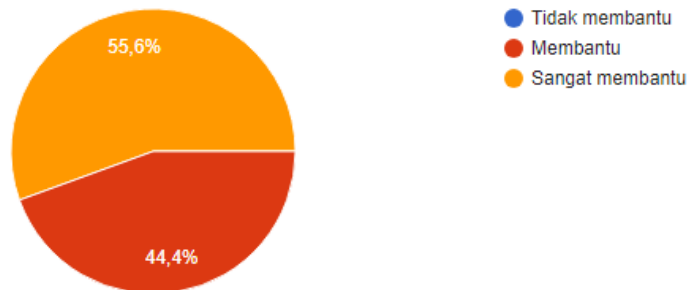
9 jawaban



Gambar 5.26 Grafik pertanyaan kuisioner 1

Apakah aplikasi ini membantu dalam mengidentifikasi kebutuhan stok barang secara lebih efisien?

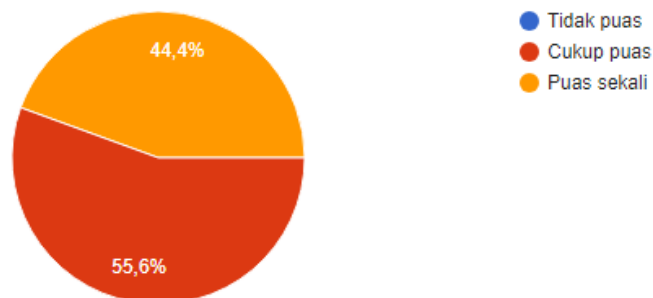
9 jawaban



Gambar 5.27 Grafik pertanyaan kuisisioner 2

Bagaimana tingkat kepuasan terhadap kemampuan aplikasi ini dalam memberikan rekomendasi pengisian stok berdasarkan data historis?

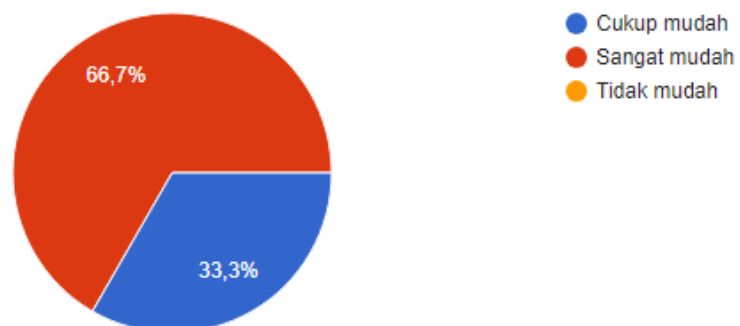
9 jawaban



Gambar 5.28 Grafik pertanyaan kuisisioner 3

Seberapa mudah untuk memasukkan data barang baru ke dalam sistem?

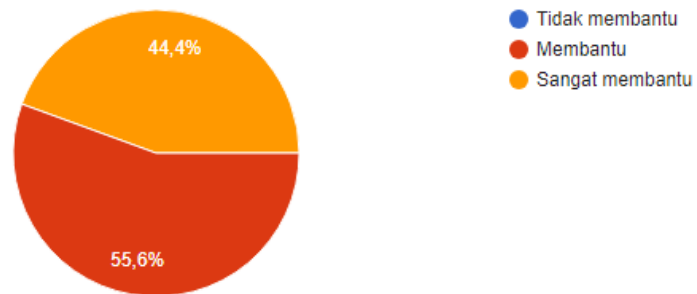
9 jawaban



Gambar 5.29 Grafik pertanyaan kuisisioner 4

Apakah aplikasi ini telah membantu mengurangi risiko kekurangan stok atau kelebihan stok yang tidak perlu?

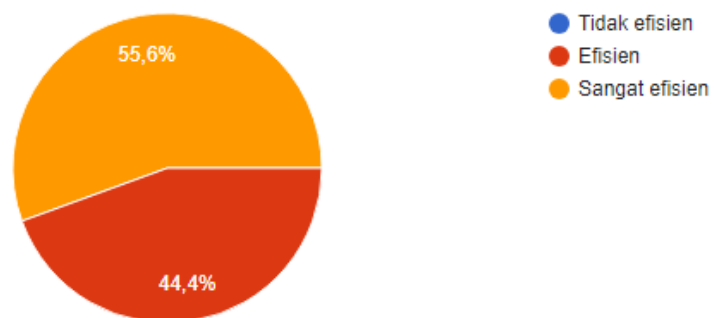
9 jawaban



Gambar 5.30 Grafik pertanyaan kuisisioner 5

Apakah penggunaan aplikasi ini telah meningkatkan efisiensi proses pengadaan barang?

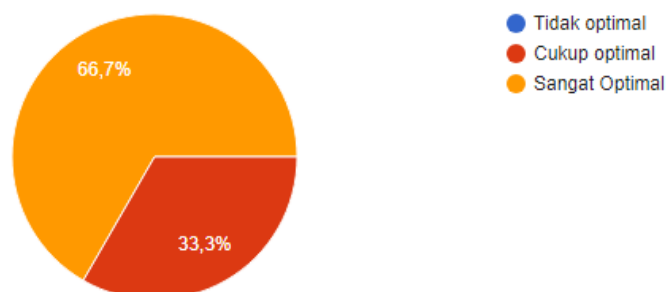
9 jawaban



Gambar 5.31 Grafik pertanyaan kuisisioner 6

Seberapa baik aplikasi ini dalam mengoptimalkan alokasi stok berdasarkan lokasi atau gudang penyimpanan?

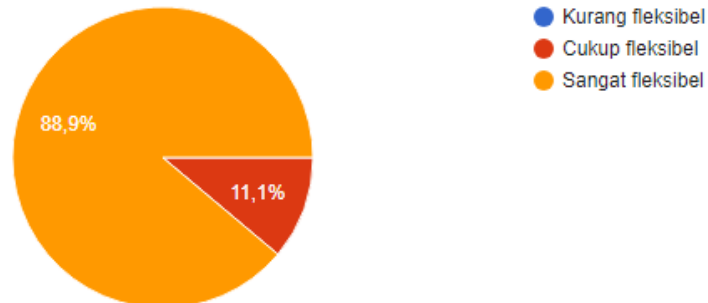
9 jawaban



Gambar 5.32 Grafik pertanyaan kuisisioner 7

Apakah aplikasi ini memberikan fleksibilitas dalam mengatur parameter optimasi sesuai dengan kebutuhan bisnis?

9 jawaban



Gambar 5.33 Grafik pertanyaan kuisisioner 8

- d) Penafsiran Hasil: Hasil dari uji reliabilitas akan memberikan gambaran tentang seberapa baik kuisisioner ini dalam mengukur variabel yang diteliti. Jika nilai reliabilitas tinggi (mendekati 1), ini menunjukkan bahwa kuisisioner memiliki tingkat konsistensi yang tinggi dalam mengukur variabel. Namun, jika nilai reliabilitas rendah, maka pertimbangan perbaikan atau revisi dalam kuisisioner mungkin diperlukan.
- Tidak optimal/Tidak membantu/Tidak efisien = 0%
 - Membantu/Cukup optimal/cukup puas = 42%
 - Sangat optimal/Sangat puas/Sangat membantu = 58%

Pada perhitungan grafik kuisisioner yang diisi responden pegawai PT. Visi Karya Prakarsa menunjukan dua variabel dimana aplikasi persediaan barang ini sangat membantu dan cukup optimal bagi pihak perusahaan. Dengan mengukur reliabilitas, penelitian dapat memastikan bahwa kuisisioner memiliki tingkat keandalan yang memadai dalam mengukur variabel yang ingin diuji.

BAB VI

KESIMPULAN

6.1 Kesimpulan

Berdasarkan penelitian yang dilakukan penulis melalui beberapa tahapan yang dilakukan pada bab-bab sebelumnya, maka penulis dapat menyimpulkan bahwa:

1. Melalui analisis permasalahan pada PT. Visi Karya Prakarsa, dapat disimpulkan bahwa memerlukan sistem manajemen *inventory* yang efektif untuk mengatur proses keluar masuknya barang. Sistem ini akan membantu perusahaan mengelola stok dengan lebih baik, mencegah kekurangan atau kelebihan persediaan, serta meminimalkan risiko kerugian atau peluang kehilangan barang.
2. Penelitian ini bertujuan untuk merancang dan mengimplementasikan sebuah aplikasi sistem manajemen inventori yang mampu menghasilkan laporan persediaan dengan menggunakan metode perpetual secara akurat dan optimal. Pendekatan yang digunakan dalam perancangan aplikasi ini mencakup integrasi teknologi informasi, otomasi proses keluar masuk barang, serta pelacakan transaksi secara *real-time* guna memastikan keakuratan dan efisiensi dalam pelaporan persediaan.
3. Studi ini bertujuan untuk menganalisis dan mengidentifikasi faktor-faktor yang menyebabkan tidak tepatnya pengaturan siklus pengadaan barang di PT. Visi Karya Prakarsa, serta merumuskan strategi yang dapat mengatasi masalah tersebut. Pendekatan yang diambil melibatkan analisis mendalam terhadap proses pengadaan saat ini, evaluasi kebutuhan dan permintaan, identifikasi pemicu ketidaktepatan, dan pengembangan rekomendasi perbaikan berdasarkan praktik terbaik dalam manajemen rantai pasokan.
4. Dengan menggunakan metode Perpetual pada aplikasi *inventory* maka perusahaan dapat mengetahui kinerja persediaan barang di PT. Visi Karya Prakarsa karena sistem secara otomatis dapat mengelola persediaan barang

setiap periodenya, sehingga pengguna dapat mengoptimasi laporan barang agar dapat meminimalisir kerugian perusahaan dalam menjalankan bisnis.

6.1 Saran

Dalam kerangka upaya memperdalam pemahaman dan mengembangkan wawasan terkait isu yang mendasari pengelolaan persediaan, terdapat panggilan yang kuat untuk mengarahkan fokus penelitian pada aspek yang masih perlu diungkapkan. Penelitian ini diinisiasi dengan niat untuk menganalisis dan mengkaji lebih mendalam elemen-elemen yang berperan dalam dinamika pengelolaan persediaan

1. Kajian ini terfokus pada pengkajian persediaan dengan tujuan mengelola stok barang di PT. Visi Karya Prakarsa. Untuk rekomendasi penelitian lebih lanjut, kami mengusulkan agar perluasan lingkup studi untuk melibatkan aspek lain dari manajemen persediaan, seperti optimalisasi rantai pasok dan strategi pengendalian persediaan yang lebih mendalam, dapat menjadi arah yang potensial. Selain itu, penelitian masa depan dapat menjelajahi implikasi dari aplikasi teknologi terkini dalam mengelola persediaan, serta menggali dampak potensial dari perubahan dinamika industri pada strategi manajemen stok yang diterapkan.
2. Dalam kerangka ini, sistem menerapkan metode pencatatan tak terputus guna *manage* inventaris melalui periode tertentu dan menghasilkan laporan inventaris. Untuk rekomendasi perluasan penelitian, kami mengusulkan eksplorasi lebih lanjut dalam hal mengintegrasikan teknologi modern seperti sensor *Internet of Things (IoT)* untuk memantau persediaan secara *real-time*. Selain itu, penelitian mendatang dapat mempertimbangkan penggunaan algoritma cerdas atau kecerdasan buatan untuk meramalkan permintaan dan memperbaiki efisiensi manajemen persediaan.
3. Pada rangkaian ini, sistem menyentuh domain persediaan tanpa menyertakan elemen harga atau aspek moneter yang menyoroti nilai rata-rata bulanan. Dalam konteks rekomendasi penelitian, kami merekomendasikan eksplorasi lebih mendalam dalam mengintegrasikan dimensi finansial ke dalam manajemen persediaan ini. Penelitian masa

depan bisa menjelajahi penggunaan metode valuasi dan akuntansi terkini yang memasukkan informasi harga ke dalam sistem manajemen persediaan ini. Selain itu, analisis dampak dari penggabungan dimensi moneter pada pengambilan keputusan persediaan dan analisis risiko terkait fluktuasi harga juga bisa menjadi area yang menarik untuk diinvestigasi lebih lanjut.

DAFTAR PUSTAKA

- AMIK BSI Bekasi, O. I., & - AMIK BSI Bekasi, G. B. A. L. (2018). Metode Rapid Application Development (RAD) pada Perancangan Website Inventory PT. SARANA ABADI MAKMUR BERSAMA (S.A.M.B) JAKARTA. *Evolusi : Jurnal Sains Dan Manajemen*, 6(2), 12–18. <https://doi.org/10.31294/evolusi.v6i2.4414>
- Agus Heryanto, Hilmi Fuad, D. D. (2014). Rancang Bangun Sistem Informasi Inventory Barang Berbasis Web Studi Kasus di PT. Infinetworks Global Jakarta. *Sisfotek Global*, 4(2), 2–5.
- Ferdinandus, S., Wowor, I. H., Kom, M., Lumenta, A. S. M., & Rumagit, M. T. A. (n.d.). *Perancangan Aplikasi Surat Masuk Dan Surat Keluar Pada PT. PLN (Persero) Wilayah Suluttenggo*. 1–7.
- Menggunakan, D., & Dan, P. H. P. (2019). *L e n t e r a d u m a i*, . 10, 46–57.
- Nasution, K., Tetap, D., Informatika, T., Islam, U., & Utara, S. (2010). *RANCANG BANGUN SISTEM PENGOLAH DATA SENSUS PENDUDUK BERBASIS MULTI USER DI KECAMATAN MEDAN KOTA* Khairuddin_nst@uisu.ac.id. 20–26.
- Qadafi, A. F., & Wahyudi, A. D. (2020). Sistem Informasi Inventory Gudang Dalam Ketersediaan Stok Barang Menggunakan Metode Buffer Stok. *Jurnal Informatika Dan Rekayasa Perangkat Lunak*, 1(2), 174–182. <https://doi.org/10.33365/jatika.v1i2.557>
- Ratnasari, I., & Nurdiniah, D. (2022). *Persediaan Bahan Baku Menggunakan Metode First In First Out (FIFO) Pada Kafe Kopilaku Kota Bekasi*. 6(2), 73–81.
- Rencana, D., Si, S., Pt, D. I., & Karya, V. (2022). *PENERAPAN METODE WARD & PEPPARD*. 16, 1–9.
- Safitri, M., & Nirmala, D. (2019). Aplikasi Inventory Manajemen Aset Berbasis Web. *IJCIT (Indonesian Journal on Computer and Information Technology)*, 4(1), 21–26.
- Siregar, V. M. M. (2018). Perancangan Sistem Informasi Inventaris Barang Pada Sekolah SMA Negeri 4 Pematangsiantar. *It Journal Research and Development*, 3(1), 54–61. [https://doi.org/10.25299/itjrd.2018.vol3\(1\).1899](https://doi.org/10.25299/itjrd.2018.vol3(1).1899)
- Syafarina, G. A., Kom, S., & Kom, M. (2016). *PERANCANGAN APLIKASI INVENTORY BARANG MATERIALS DAN PRODUCT*. 7(1), 25–33.
- Vol, C. A., & Persed, A. P. (2019). *1 , 2 , 3 1*, . 09(02), 25–47.

LAMPIRAN

Lampiran 1 : Hasil Wawancara dengan Narasumber

Narasumber : Lucky Luqman
Jabatan/Posisi : CTO
Hari / Tanggal : 28 Maret 2023
Instansi : PT. Visi Karya Prakarsa

Wawancara ini bertujuan untuk mengumpulkan data yang esensial dalam rangka penelitian berjudul "Optimasi Pengelolaan Persediaan Barang dengan Penerapan Metode Perpetual pada Aplikasi Inventory di PT. Visi Karya Prakarsa". Berikut ini disajikan daftar pertanyaan wawancara berserta jawaban yang terkait:

No.	Pertanyaan	Jawaban
1.	Perusahaan bergerak di bidang apa?	Perusahaan kami bergerak di sektor Teknologi Informasi yang mengkhususkan diri dalam penyediaan layanan dan jasa, sambil juga mengelola persediaan barang elektronik dan perangkat keras (<i>hardware</i>).
2.	Bagaimana struktur karyawan pada perusahaan ini?	Dalam struktur organisasi kami, peran dan tanggung jawab masing-masing posisi sangat terdefinisi. Komisaris atau advisor bertugas sebagai penasihat bagi klien serta menjadi saluran komunikasi utama ketika ada permasalahan di perusahaan, yang kemudian disampaikan kepada Komisaris atau CEO. Selanjutnya, kami memiliki Direktur Utama atau CEO yang bertanggung jawab atas manajemen keseluruhan perusahaan.

		<p>Setelahnya, peran COO berfokus pada aspek operasional, termasuk dalam hal pemasaran, penjualan, dan operasional tim bisnis. Di samping itu, CTO memiliki peran yang aktif dalam pengembangan produk, baik dalam tahap R&D maupun produksi. Mereka berperan kunci dalam memastikan bahwa produk-produk kami terus berkembang dan sesuai dengan kebutuhan pasar.</p> <p>Kerja sama dan komunikasi antara semua tingkatan manajemen merupakan hal yang krusial dalam menjalankan operasional kami dengan sukses. CEO dan COO berupaya agar strategi perusahaan terintegrasi dengan baik, sementara CTO berfokus pada pengembangan teknologi dan produk. Komisaris atau advisor kami juga memberikan wawasan tambahan serta nasihat strategis yang berharga untuk mendukung pencapaian tujuan perusahaan.</p>
3.	Bagaimana strategi pemasaran dari perusahaan?	<p>Dalam upaya memperkuat eksistensi dan kehadiran kami dalam industri <i>e-commerce</i>, kami telah mengadopsi pendekatan yang mencakup berbagai strategi yang berorientasi pada <i>digital marketing</i>, serta berpartisipasi aktif di berbagai platform media sosial seperti <i>Instagram</i> dan <i>YouTube</i>. Selain itu, kami juga telah melaksanakan berbagai acara</p>

		kolaboratif dan afiliasi yang telah memberikan kontribusi positif dalam pengembangan bisnis kami.
4.	Bagaimana proses klien/supplier melakukan pembelian dengan perusahaan saat akan menggunakan layanan	Proses penandatanganan kontrak dengan klien kami melibatkan serangkaian tahapan yang dimulai dengan pertemuan. Dalam pertemuan tersebut, perusahaan kami secara komprehensif memperkenalkan produk dan layanan yang kami tawarkan. Selanjutnya, proses negosiasi dan kesepakatan dilakukan dengan cermat. Setelah semua pihak mencapai kesepakatan, kami melanjutkan dengan proses instalasi dan konfigurasi aplikasi di lokasi klien.
5.	Apa ada kendala dalam prosesnya?	Dalam situasi di mana persediaan barang tidak tersedia dalam susunan yang lengkap dengan komponen yang diperlukan, langkah selanjutnya adalah menginisiasi pemesanan komponen yang kurang agar produksi barang dapat berjalan dengan lancar.
6.	Komponen apa saja yang disediakan dari perusahaan?	Dalam konteks komponen persediaan barang untuk produksi alat kami, kami menyediakan berbagai komponen penting seperti Arduino, RJ45, servo motor, transistor, sensor, solenoid, layar LCD, buzzer, dan berbagai jenis baterai.
7.	Bagaimana proses sistem persediaan produk tersebut di perusahaan?	Sistem pencatatan kami saat ini belum memiliki struktur yang terorganisir, karena masih menggunakan pendekatan manual.

8.	Apa ada permasalahan terkait persediaan barang?	Beberapa isu yang kami identifikasi meliputi laporan kehilangan barang, potensi kerugian akibat pencatatan yang belum optimal, kurangnya rekapan data yang efisien, masalah <i>overstocking</i> dan <i>understocking</i> , serta masalah kesalahan input data.
9.	Selama ini resiko pencatatan persediaan barang secara manual bagaimana?	Evaluasi risikonya belum efektif dalam memfasilitasi pemantauan laporan keuangan yang akurat, karena belum sejalan dengan standar pelaporan yang telah ditetapkan.
10	Apakah aplikasi inventory diperlukan untuk perusahaan?	Sangat penting bagi kami untuk memiliki aplikasi inventory yang mudah diakses dan dapat memberikan kemudahan dalam melihat laporan persediaan barang.

Pewawancara

Narasumber

Herlin Martin Marcella



Lucky Luqman

Lampiran 2 : Dokumentasi Wawancara

Nama Perusahaan : PT. Visi Karya Prakarsa

Lokasi : Jl. Jupiter Barat No.37, Sekejati, Kec. Buahbatu

Dokumentasi wawancara dengan CTO



Lampiran 3 : Dokumentasi kantor PT. Visi Karya Prakarsa

