

Penerapan Metode FIFO dan ROP Pada Sistem Inventory UD. Salam 51 Berbasis Website

Dean Sanjaya Surya Putra AK¹, Anita Qoiriah²

^{1,2}Jurusan Teknik Informatika, Universitas Negeri Surabaya

¹deansanjaya.20082@mhs.unesa.ac.id

²anitaqoiriah@unesa.ac.id

Abstrak— UD Salam 51 merupakan usaha dagang yang bergerak di bidang penjualan barang-barang bangunan dan hingga saat ini masih menggunakan sistem pencatatan manual berbasis kertas. Sistem ini sering menimbulkan berbagai masalah, seperti kesalahan pencatatan, ketidakakuratan data stok, serta kesulitan dalam pengelolaan barang yang sering tertumpuk atau tertimbun. Penelitian ini bertujuan untuk mengatasi masalah tersebut dengan mengembangkan sistem inventori berbasis *website* yang menerapkan metode *First In, First Out (FIFO)* dan *Reorder Point (ROP)*. Metode *FIFO* dirancang untuk memastikan pengeluaran barang dilakukan berdasarkan urutan kedatangan guna mencegah barang tertimbun atau rusak akibat penyimpanan terlalu lama. Metode *ROP* digunakan untuk menentukan batas angka stok yang memicu pemesanan ulang secara optimal agar ketersediaan barang tetap terjaga dan kekurangan stok dapat dihindari. Untuk mendapatkan fungsi yang baik dan akurat, pengujian sistem dilakukan menggunakan metode *black box* dan *white box* yang bertujuan menguji keandalan algoritma *FIFO* dan *ROP* serta memastikan implementasinya sesuai kebutuhan. Dengan integrasi kedua metode tersebut, sistem ini diharapkan dapat meminimalkan risiko kesalahan pencatatan dan mendukung kelancaran operasional UD Salam 51.

Kata Kunci — FIFO, ROP, Sistem inventori, Website, Testing

I. PENDAHULUAN

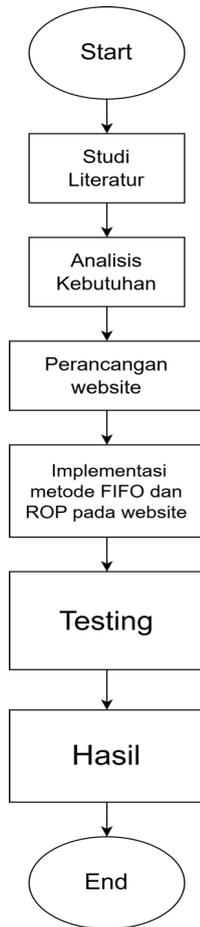
Perkembangan teknologi yang pesat hingga saat ini telah banyak memberikan manfaat dalam berbagai aspek kehidupan [1]. Penggunaan teknologi oleh manusia dalam memudahkan pekerjaan merupakan sebuah hal yang kreatif dalam kehidupan. Menurut *Global Digital Overview*, pengguna internet di dunia mencapai 4,7 miliar dari total penduduk populasi dunia sebanyak 7,83 miliar, dari data tersebut sudah dapat dipastikan bahwa sudah melebihi 50% penduduk dunia sudah menggunakan internet [2]. Salah satu bidang yang sangat memanfaatkan kemajuan teknologi yaitu bidang bisnis. Pada beberapa perusahaan, terdapat prosedur yang harus dicatatkan dengan detail dalam prosesnya. Contohnya seperti pencatatan barang keluar dan masuk gudang dan jumlah stok yang ada di gudang [3]. Teknologi informasi membantu perusahaan dalam mengelola bisnisnya agar lebih mudah dan cepat sehingga bisa bersaing dengan kompetitor lainnya. Pada suatu perusahaan, banyak sektor perusahaan yang dapat diterapkan teknologi informasi salah satunya adalah di bagian *inventory* (persediaan). Bagian *inventory* berfungsi sebagai pengelolaan persediaan barang, memastikan ketersediaan dan pengaturan stok yang lebih optimal bagi perusahaan [4].

Salam 51 adalah usaha dagang yang menyediakan alat dan bahan-bahan bangunan yang berlokasi di Gayungan, Surabaya. Dalam proses pergudangan, salam 51 masih menggunakan cara manual yaitu dengan pencatatan menggunakan buku. Pencatatan secara manual tersebut menyebabkan penumpukan data transaksi barang masuk dan keluar sehingga menyebabkan pengeluaran barang tidak berdasarkan waktu masuknya. Oleh karena itu, terkadang terdapat barang yang sudah rusak seperti paku yang berkarat dan kayu yang sudah lapuk tetap tersimpan didalam gudang dengan waktu yang lama. Kebiasaan yang terjadi ketika kegiatan pengendalian barang terjadi adalah kekurangan persediaan (*out of stock*) atau kelebihan barang (*over of stock*) [5]. Kelebihan barang bisa terjadi dikarenakan jumlah pemesanan barang yang terlalu tinggi. Hal ini menyebabkan biaya persediaan yang besar dan kualitas barang yang akan menurun. Sedangkan kekurangan barang bisa terjadi dikarenakan rendahnya jumlah barang ketika di stok kembali yang akan mengakibatkan penyerahan barang kepada konsumen akan terganggu. Oleh karena itu, perusahaan harus mampu menentukan kapan dan berapa jumlah barang yang harus di stok kembali berdasarkan jumlah barang saat ini di gudang.

Berdasarkan permasalahan tersebut, mendorong penulis untuk membuat sebuah sistem berbasis *website* yang menerapkan metode *FIFO (First in First Out)* dan *ROP (Reorder Point)*. Metode *FIFO (First In First Out)* adalah teknik yang mengutamakan penggunaan barang yang pertama kali masuk sebagai produk yang dijual atau digunakan terlebih dahulu. Tujuan dari metode ini adalah untuk memastikan barang yang masuk lebih awal segera diprioritaskan untuk didistribusikan. Dengan menerapkan metode *FIFO*, risiko kerusakan barang yang disimpan terlalu lama di gudang dapat diminimalkan. Hal ini juga membantu mengontrol alur masuk dan keluarnya barang, sehingga kualitas barang yang disimpan tetap terjaga [6]. *Reorder Point (ROP)* adalah ambang batas di mana pemesanan ulang dilakukan untuk menghindari kehabisan stok [7]. Ini melibatkan permintaan yang diharapkan dan dibutuhkan selama waktu tunggu (*lead time*). Ketika persediaan terus menurun, *ROP* dihitung berdasarkan waktu tunggu, dan sering kali dilengkapi dengan *safety stock*. *Safety stock* berfungsi untuk mengurangi risiko kekurangan stok selama waktu tunggu, dengan mempertimbangkan kemungkinan terjadinya permintaan yang lebih tinggi dari perkiraan [8].

II. METODE PENELITIAN

Pada penelitian ini, penulis melalui beberapa tahapan yang terstruktur dengan runtutnya sebagai berikut :



Gbr. 1 Diagram Alur Penelitian

A. Studi Literatur

Dalam tahapan studi literatur ini, penelitian akan memfokuskan pada beberapa konsep utama yang mendukung pengembangan website, yaitu metode *First in First out (FIFO)* dan *Reorder Point (ROP)*. Metode *FIFO* digunakan dalam pengelolaan inventaris untuk memastikan barang yang masuk pertama kali juga akan keluar lebih dahulu, sedangkan *ROP* menentukan batas minimum stok untuk memicu pesanan ulang. Selain itu, teknologi yang digunakan dalam pengembangan website ini meliputi *React.js* untuk menangani tampilan antarmuka, *express.js* untuk menangani pengolahan data dan *MongoDB* sebagai basis data untuk menyimpan informasi data barang. *React JS* adalah pustaka *Java Script* yang sangat populer dan digunakan secara luas untuk membuat antarmuka pengguna (*user interface*) yang efisien dan fleksibel, terutama pada *frontend* aplikasi website [9]. *Express.js* adalah *framework web* untuk *Node.js* yang dirancang untuk membangun aplikasi web dan *API* yang cepat dan efisien. Dikenal karena kesederhanaan dan fleksibilitasnya, *Express* menyediakan berbagai fitur yang

memudahkan pengembang dalam menangani *route*, *middleware*, dan permintaan *HTTP* [10]. Pengujian website akan dilakukan menggunakan 2 metode yaitu metode *blackbox testing* untuk memverifikasi fungsionalitas sistem tanpa melihat kode sumber, serta *whitebox testing* untuk memastikan setiap komponen berfungsi dengan benar secara independent dengan melihat kode sumbernya.

B. Analisis Kebutuhan

Pada tahapan ini, akan dilakukan pengumpulan beberapa data serta informasi yang telah didapat dari beberapa referensi seperti jurnal dan yang lainnya sehingga peneliti dapat menentukan analisis kebutuhan baik secara fungsional maupun non fungsional. Kebutuhan fungsional berkaitan dengan fitur yang akan dikembangkan dalam aplikasi, sedangkan kebutuhan non-fungsional tidak secara langsung terhubung dengan fitur tertentu. Sebagai gantinya, kebutuhan non-fungsional menetapkan batasan atau standar yang harus dipenuhi oleh kebutuhan fungsional [11]. Adapun kebutuhan fungsional sebagai berikut :

- Sistem memungkinkan karyawan untuk menambah, melihat, mengeluarkan, dan menghapus data barang.
- Sistem mengimplementasikan metode *FIFO* untuk pengeluaran barang sesuai urutan kedatangan yang tercatat system
- Sistem harus menghitung *Reorder Point (ROP)* berdasarkan rerata penggunaan, *lead time* dan nilai *Safety Stock* yang diinput.
- Sistem harus menampilkan notifikasi jika stok barang berada di bawah *ROP*.
- Sistem memungkinkan pengguna mengganti password dengan validasi yang sesuai.
- Pimpinan dapat melihat laporan transaksi barang seperti penambahan, pengeluaran, dan penghapusan barang setelah melakukan *login*.

Sedangkan kebutuhan non fungsional pada sistem pengelolaan barang berbasis website ini adalah sebagai berikut

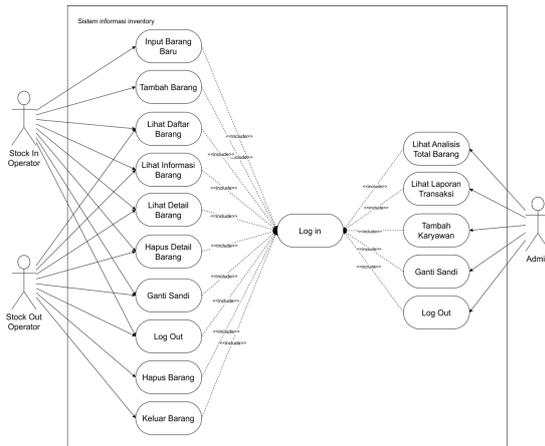
- Sistem mendukung login bagi karyawan dan pimpinan dengan autentikasi username dan password yang sudah terdaftar di sistem.
- Sistem menyediakan fitur *logout* untuk mengakhiri sesi pengguna dengan aman.
- Sistem mudah digunakan dengan antarmuka yang jelas dan *responsive*.
- Sistem mudah diakses di berbagai perangkat yang telah terkoneksi dengan internet.
- Pimpinan dapat menambahkan akun karyawan dengan data login yang sesuai.

C. Perancangan Website

Pada tahapan ini, dibuat beberapa model sistem berdasarkan hasil analisa kebutuhan yang didapatkan. Tidak hanya itu, pada tahapan ini dibuat juga desain *user interface* (antarmuka pengguna) untuk penggambaran garis besar tampilan website. Pemodelan sistem yang digunakan yaitu

Unified Modelling Language (UML) yang terdiri dari Use-Case Diagram, Class Diagram dan Activity Diagram.

a) Use Case Diagram



Gbr. 2 Use Case Diagram

Gbr. 2 merupakan use case diagram pada penelitian ini, dengan 3 actor yang memiliki hak akses berbeda setiap aktornya.

1) Stock In Operator

Merupakan actor yang memiliki hak akses untuk mengelola barang yang berada di gudang (*inventory*) menggunakan sistem informasi *website* ini. Proses pengelolaannya berupa tambah barang, lihat daftar barang, lihat informasi barang dan lain-lain.

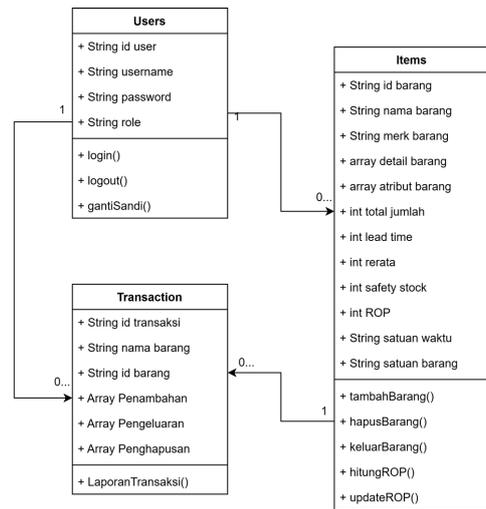
2) Stock Out Operator

Merupakan actor yang memiliki hak akses untuk mengelola barang yang berada di gudang (*inventory*) menggunakan sistem informasi *website* ini. Proses pengelolaannya berupa keluar barang, lihat daftar barang, hapus barang dan lihat informasi barang.

3) Admin

Merupakan actor yang bersifat sebagai pemimpin atau pemilik usaha dagang. Mampu melihat laporan transaksi yang berisi data-data barang di gudang. Tidak hanya itu, actor ini juga bisa menambah akun karyawan

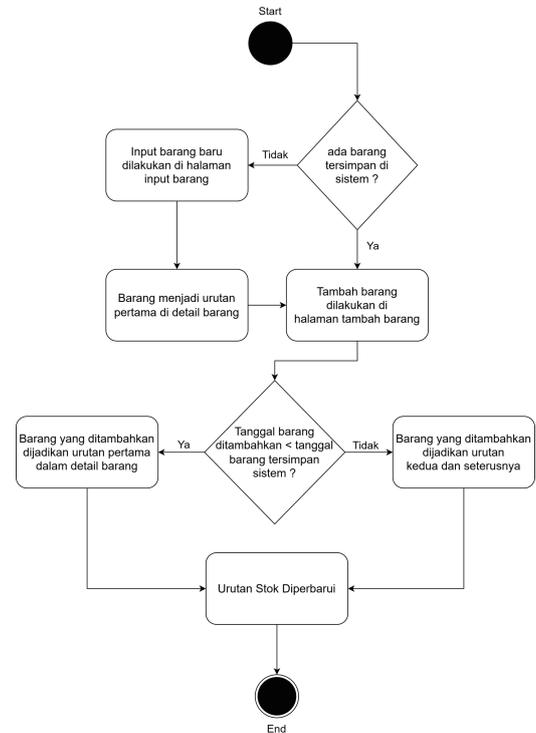
b) Class Diagram



Gbr. 3 Class Diagram

Gbr. 3 merupakan class diagram pada penelitian ini. Berdasarkan *class diagram*, tiap *collection* memiliki *atribut*, *value* dan *methodnya* masing-masing yang akan saling terhubung didalam sistem

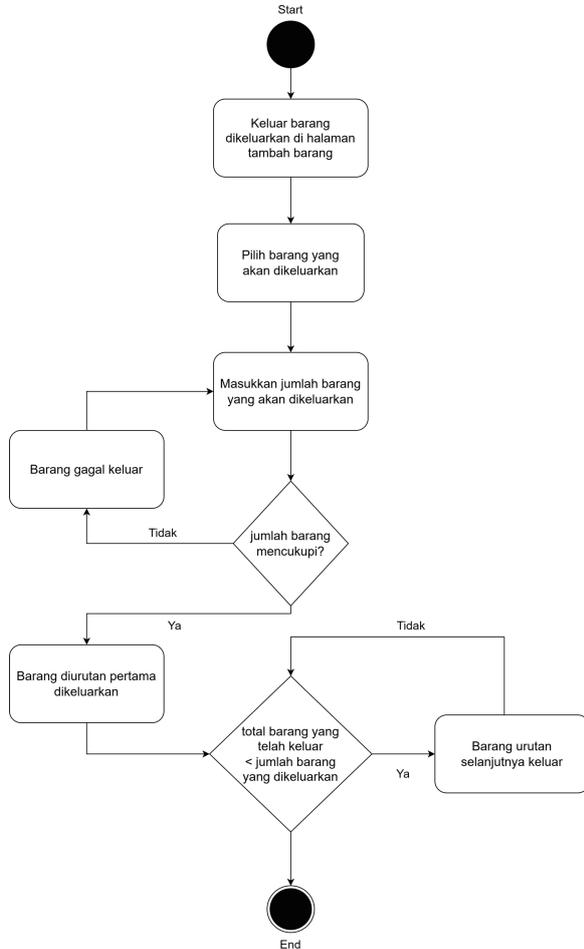
c) Activity Diagram



Gbr. 4 Activity Diagram First In

Gbr. 4 menjelaskan tentang proses tambah barang yang menerapkan metode *First In*. Proses dimulai dengan sistem memastikan terlebih dahulu apakah ada barang yang sudah terinput dalam sistem. Apabila sudah maka tambah barang bisa dilakukan sebaliknya jika belum maka pengguna harus menginput barang terlebih dahulu di halaman input barang. Setelah tambah barang dilakukan, sistem akan mengecek

tanggal barang yang baru saja ditambahkan dengan tanggal barang yang sudah tersimpan oleh sistem. Selanjutnya, sistem akan melakukan pengurutan barang berdasarkan tanggal, dimulai dari tanggal yang lebih lama hingga tanggal terbaru.

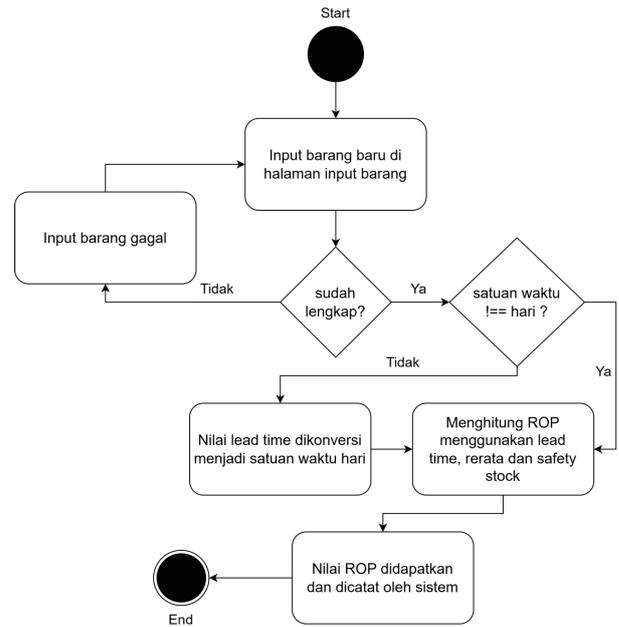


Gbr. 5 Activity Diagram First Out

Gbr. 5 menjelaskan tentang proses keluar barang yang menerapkan metode *First Out*. Proses dimulai ketika pengguna ingin mengeluarkan barang di halaman keluar barang. Pengguna akan mengisi beberapa keperluan untuk pengeluaran barang, seperti nama barang dan jumlah barang yang akan dikeluarkan. Setelah itu, sistem akan melakukan pengecekan terhadap jumlah barang yang tersimpan. Jika jumlah barang yang akan keluar lebih besar daripada jumlah barang yang tersedia, maka pengeluaran barang tidak akan terjadi. Sebaliknya, jika jumlah barang yang akan keluar lebih kecil atau sama dengan jumlah barang yang tersimpan, maka pengeluaran dapat dilakukan.

Pengeluaran barang akan dilakukan dengan memprioritaskan barang yang terurut pertama dalam detail barang, yaitu barang yang pertama kali masuk (berdasarkan tanggal masuk). Jika jumlah barang di urutan pertama tidak mencukupi untuk memenuhi permintaan, maka barang di urutan berikutnya akan

digunakan untuk mengisi kekurangan tersebut, dan seterusnya, hingga seluruh jumlah barang yang diminta terpenuhi.



Gbr. 6 Activity Diagram perhitungan ROP

Gbr. 6 menjelaskan alur proses sistem dalam menentukan nilai *Reorder Point (ROP)* sebuah barang. Proses dimulai ketika barang diinputkan oleh karyawan melalui halaman input barang. Sistem pertama-tama memverifikasi kelengkapan data yang diinputkan oleh karyawan. Apabila data belum lengkap, proses input barang tidak dapat dilanjutkan. Sebaliknya, apabila semua data telah lengkap, proses input barang berhasil dilakukan. Selanjutnya, sistem akan memeriksa satuan waktu yang diinputkan karyawan untuk *lead time*. Jika satuan waktu selain dalam bentuk hari, sistem secara otomatis mengonversi nilai *lead time* ke dalam satuan hari. Setelah konversi selesai, sistem menghitung nilai *ROP* barang tersebut menggunakan formula yang melibatkan *lead time* (dalam hari), asumsi rata-rata penjualan, dan *safety stock*. Proses berakhir setelah nilai *ROP* berhasil dihitung dan disimpan dalam sistem. Rata-rata penjualan ini akan diupdate setiap minggunya tergantung dengan penjualan barang dalam minggu tersebut.

d) Rancangan Antarmuka

Logo/SVG/ICON

Login Salam 51

Username
Masukkan Username

Password
Masukkan Password

Login

Gbr. 7 Rancangan halaman login

Gbr. 7 adalah desain antarmuka untuk halaman login dan merupakan halaman pertama ketika website dibuka. Untuk masuk kedalam sistem, pengguna perlu melakukan proses login menggunakan *username* dan *password* yang sudah terdaftar oleh sistem.

Input barang

Nama barang: Masukkan nama barang

Merk Barang: Masukkan merk barang

Jumlah barang: Masukkan jumlah barang

Barang Masuk: Hari/Bulan/Tahun

Lama waktu pengantaran: Masukkan lama waktu pengantaran

Satuan Barang: Masukkan satuan barang

Rata-rata penjualan: Masukkan rata-rata penjualan

Satuan Waktu: Masukkan satuan waktu

Tempat Simpan: Masukkan Tempat Simpan

Safety Stock: Masukkan safety stock

Atribut tambahan (jika ada) **Tambah Atribut**

Atribut Value **Hapus Atribut**

Hapus Konfirmasi

Gbr. 8 Rancangan halaman input barang

Gbr. 8 adalah desain antarmuka untuk halaman input barang. Halaman ini berfungsi untuk pengguna memasukkan barang baru kedalam sistem. Setelah pengguna selesai menginputkan barang, sistem akan langsung menghitung nilai *ROP* berdasarkan informasi barang yang diinput oleh pengguna.

List Barang						
No	Nama Barang	Merk	Jumlah barang	Satuan Barang	ROP	Aksi
1	kayu	A	108	meter	120	📄 🗑️ +
2	besi	B	98	milimeter	150	📄 🗑️ +
3	paku	C	27	gram	30	📄 🗑️ +

Gbr. 9 Rancangan halaman list barang

Gbr. 9 adalah desain antarmuka untuk halaman list barang. Halaman ini berfungsi untuk menampilkan informasi yang sudah tercatat oleh sistem kepada pengguna. Informasi yang ditampilkan seperti nama, merk jumlah, angka *ROP* dan lain sebagainya.

List Barang / Paku				
No	Jumlah	Tanggal Masuk	Tempat simpan	Aksi
1	4	23-7-2024	rak A	🗑️
2	11	28-7-2024	rak A	🗑️
3	12	01-8-2024	rak B	🗑️

Gbr. 10 Rancangan halaman detail barang

Gbr. 10 adalah desain antarmuka untuk halaman detail barang. Halaman ini berfungsi untuk menampilkan urutan barang berdasarkan tanggal barang tersebut diinput oleh pengguna. Halaman ini bisa diakses melalui halaman list barang

Informasi Barang Kayu

Nama barang: Kayu

Merk Barang: A

Lama waktu pengantaran: 5 Hari

Satuan Barang: KG

Rata-rata penjualan: 2

Safety Stock: 10

Atribut Tambahan (jika ada)

Atribut Tambahan (jika ada)

Gbr. 11 Rancangan halaman informasi barang

Gbr. 11 adalah desain antarmuka untuk halaman informasi barang. Halaman ini berfungsi untuk menampilkan informasi sebuah barang secara lengkap. Pengguna bisa mengakses halaman ini melalui halaman list barang.

Tambah barang

Nama barang

Jumlah Barang

Tanggal Barang Masuk

Tempat Simpan

Gbr. 12 Rancangan halaman tambah barang

Gbr. 12 adalah desain antarmuka untuk halaman tambah barang. Halaman ini berfungsi untuk menambahkan barang yang sudah tersimpan sistem. Dalam proses penambahan ini, proses *FIFO In* akan bekerja. Sistem akan membandingkan tanggal masuk barang terlebih dahulu sebelum barang tersebut ditambahkan dan disimpan oleh sistem.

Keluar barang

Nama barang

Merk

Total Stok

Jumlah Barang

Gbr. 13 Rancangan halaman keluar barang

Gbr. 13 adalah desain antarmuka untuk halaman keluar barang. Halaman ini berfungsi untuk mengeluarkan barang dari sistem. Dalam proses pengeluaran ini, proses *FIFO Out* akan bekerja. Sistem akan memprioritaskan untuk mengeluarkan barang dengan tanggal masuk paling awal. Dengan penerapan seperti itu, maka dapat dipastikan bahwa metode FIFO akan bekerja dengan baik dan benar.

D. Implementasi Metode *FIFO* dan *ROP* pada *Website*

Pada tahapan ini, website inventori akan menerapkan metode *FIFO* dan *ROP* sebagai fokus utama penelitian. Metode *FIFO* akan memastikan barang yang masuk terlebih dahulu juga akan keluar terlebih dahulu, sementara *ROP* akan digunakan untuk menjaga agar stok tidak terlalu rendah dan berlebihan. Penerapan kedua metode ini bertujuan untuk mencegah kerusakan barang akibat penyimpanan terlalu lama di gudang.

Penggunaan algoritma *First In First Out (FIFO)* diterapkan pada proses tambah barang dan keluar barang. Sebelum

barang dicatat oleh sistem, karyawan terlebih dahulu mengisi informasi terkait barang, seperti nama barang, jumlah barang, tanggal masuk gudang, *lead time*, *safety stock*, dan asumsi rata-rata penjualan harian. Berdasarkan tanggal masuk, sistem akan mengurutkan barang dari yang paling awal hingga yang terbaru. Dengan pengurutan ini, karyawan dapat memastikan bahwa barang yang lebih dahulu masuk akan dikeluarkan terlebih dahulu. Hal ini bertujuan untuk mencegah penurunan kualitas barang, menjaga kelayakan barang untuk dijual, serta meminimalkan risiko kerugian akibat penurunan harga.

Penggunaan algoritma *Reorder Point* diterapkan pada proses input barang. Sistem akan mengawali proses dengan memverifikasi kelengkapan data yang dibutuhkan. Apabila data yang diinput lengkap, maka sistem akan menghitung nilai *ROP* dari barang tersebut berdasarkan nilai *lead time*, rata-rata penjualan dan *safety stock*. Proses berakhir setelah nilai *ROP* berhasil dihitung dan disimpan didalam sistem. Rata-rata penjualan ini akan diupdate setiap minggunya yang dihitung dari rata-rata penjualan barang tersebut di minggu ini.

TABEL I
ALGORITMA YANG DIGUNAKAN

Algoritma	Informasi yang dibutuhkan	Penjelasan
<i>First in First Out</i>	<ul style="list-style-type: none"> Tanggal barang masuk 	Menggunakan tanggal barang masuk sebagai acuan untuk barang tersebut harus keluar kembali.
<i>Reorder Point</i>	<ul style="list-style-type: none"> <i>Lead time</i> <i>Safety stok</i> <i>Rerata penjualan</i> 	<i>Lead time</i> adalah waktu yang dibutuhkan barang untuk masuk kembali kedalam gudang. <i>Safety</i> stok adalah stok barang yang disimpan perusahaan untuk mencegah terjadinya permintaan tiba-tiba. Rata-rata penjualan adalah rerata yang diambil dari data pengeluaran barang dalam seminggu

E. Testing

Pada tahapan ini, testing menggunakan dua metode, yaitu pengujian blackbox testing dan whitebox testing.

a) Blackbox testing

Blackbox testing memiliki keuntungan karena penguji tidak memerlukan pengetahuan tentang kode program yang ada [12]. Pengujian ini dilakukan dari sudut pandang pengguna untuk memverifikasi apakah sistem berfungsi sesuai dengan spesifikasi yang telah ditentukan. Pengujian berfokus pada output yang dihasilkan berdasarkan input yang diberikan tanpa memperhatikan bagaimana sistem menghasilkan output tersebut.

b) Whitebox testing

Whitebox testing bertujuan untuk menguji setiap komponen perangkat lunak secara mendalam, dengan memastikan bahwa setiap unit kode berfungsi dengan benar secara independent [13]. Dalam pengujian ini, penguji memiliki akses penuh terhadap kode sumber aplikasi dan merancang tes berdasarkan pemahaman terhadap struktur dan logika program. Metode ini sangat berguna untuk mengidentifikasi kesalahan atau celah dalam implementasi kode yang tidak dapat dideteksi dalam *blackbox testing*.

Dalam konteks aplikasi ini, pengujian difokuskan pada dua fungsi utama, yaitu *FIFO* dan *ROP*.

1. Fungsi *FIFO* akan diuji dengan *whitebox testing* untuk memastikan bahwa barang yang pertama kali masuk adalah barang yang pertama kali keluar, sesuai dengan urutannya. Pengujian ini dilakukan dengan memeriksa jalur kode yang mengatur urutan barang dan memastikan bahwa tidak ada celah atau kesalahan dalam pengurutan.
2. Fungsi *ROP* akan memverifikasi bahwa sistem secara otomatis memicu peringatan atau pesanan ulang ketika stok barang mencapai batas minimum yang telah ditentukan. Di sini, pengujian kode akan memastikan bahwa logika penghitungan *ROP* berjalan dengan benar dan bahwa kondisi untuk memicu pesanan ulang dieksekusi dengan tepat saat jumlah stok mencapai *threshold* yang telah ditetapkan.

III. HASIL DAN PEMBAHASAN

Pada bab ini, menjelaskan tentang hasil dari penelitian yang mencakup perancangan website dan penerapan metode *FIFO* dan *ROP* sampai dengan beberapa pengujian terhadap website pengelolaan barang.

A. Perancangan dan penerapan metode pada website

a) Perancangan Website

Dalam proses pembangunan website untuk pengelolaan barang ini, digunakan framework *React.js*, yang merupakan salah satu framework berbasis *JavaScript* dan bersifat *open-source* atau gratis untuk digunakan. Untuk memulai pembuatan proyek menggunakan *React* dengan pendekatan modern dan optimal, dapat menjalankan perintah di terminal, yaitu:

```
npm create vite@latest
```

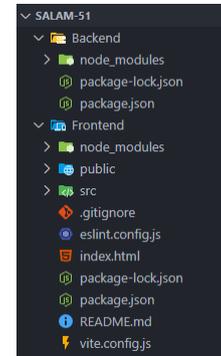
Gbr. 14 Perintah untuk membuat proyek *React.js*

Dalam folder yang sama, buat juga sebuah proyek menggunakan *Express.js* untuk menangani proses data, seperti menyimpan, mengolah, dan mengirim data ke aplikasi frontend. *Express.js* dipilih karena kesederhanaannya dalam pengaturan server dan dukungannya untuk mengelola permintaan *HTTP* dengan efisien.

```
npm install express -save
```

Gbr. 15 Perintah untuk membuat proyek *Express.js*

Ketika proses inisiasi kedua proyek tersebut selesai, maka dapat dilihat struktur proyek yang telah dibuat seperti gambar dibawah ini.



Gbr. 16 Struktur folder proyek

Gbr. 16 menunjukkan struktur folder untuk *react* dan *express* teratur dengan baik dan rapih sehingga membuat proses pengembangan proyek menjadi lebih baik. Dari perancangan website dengan *framework react JS* dan *express JS* menghasilkan beberapa tampilan halaman sebagai berikut :



Gbr. 17 Halaman login

Gbr. 17 menunjukkan tampilan halaman login yang digunakan oleh pengguna untuk mengakses sistem. Setiap role memiliki hak akses yang berbeda, yang akan menentukan fitur-fitur apa saja yang dapat mereka akses setelah berhasil masuk ke dalam sistem.

Gbr. 18 Halaman input barang

Gbr. 18 menunjukkan halaman yang hanya bisa diakses oleh akun yang memiliki role sebagai stock in operator. Fungsi halaman ini adalah untuk menginput barang baru masuk kedalam database dan menyimpan informasi barang tersebut. Fitur unggulan halaman ini adalah kemampuannya untuk menangani barang dengan detail tambahan yang hanya dimiliki oleh beberapa barang. Jika kasus tersebut terjadi, pengguna dapat menambahkan atribut secara dinamis dengan menekan tombol ‘tambah atribut’. Fungsi ini memberikan fleksibilitas bagi pengguna untuk mencatat informasi khusus sebuah barang sesuai kebutuhan tanpa batasan format yang telah ada.

Gbr. 19 Halaman tambah barang

Gbr. 19 menunjukkan halaman tambah barang yang hanya bisa diakses oleh akun yang memiliki role stock in operator. Untuk mengakses halaman ini, pengguna hanya perlu menekan tombol ‘+’ di halaman list barang. Untuk mensubmit data barangnya, pengguna diharuskan mengisi semua informasi seperti jumlah barang, tanggal

barang, dan juga tempat simpan barangnya. Informasi ini akan digunakan untuk mengatur barang di gudang sesuai dengan metode FIFO.

Gbr. 20 Halaman keluar barang

Gbr. 20 merupakan halaman keluar barang yang berfungsi untuk mengeluarkan barang dari gudang. Halaman ini bisa diakses melalui sidebar dibagian kiri website dan hanya bisa diakses oleh akun yang memiliki role stock out operator. Untuk mengeluarkan barang, pengguna perlu untuk memilih nama barang di dropdown nama barang lalu mengisi jumlah barang yang ingin dikeluarkan. Setelah itu, sistem akan menghitung dan membuat perintah berapa banyak barang yang harus keluar dan dari tempat penyimpanan mana barang tersebut harus diambil.

b) Perhitungan ROP

Perhitungan angka ROP dilakukan ketika barang pertama kali diinput masuk kedalam sistem dan ketika barang tersebut sudah tersimpan didalam gudang selama lebih dari seminggu. Hal ini dikarenakan angka dari rata-rata penjualan barang akan selalu berubah-ubah tergantung dari penjualan barang tersebut dalam seminggu. Angka dari rata-rata penjualan barang akan selalu dihitung diakhir minggu tepatnya pada hari sabtu pukul 16.00. Selain rata-rata penjualan, informasi barang yang digunakan untuk menghitung ROP adalah safety stock, satuan waktu, lead time. Sebelum perhitungan, satuan waktu yang dipilih harus dikonversi terlebih dahulu ke dalam satuan hari sebelum digunakan dalam rumus ROP yaitu :

$$\text{Reorder Point} = (U \times L) + SS \quad (1)$$

Dengan :

- U : rata-rata penjualan harian
- L : lead time dalam satuan hari
- SS : Safety Stock

c) Penerapan Metode FIFO

Penerapan metode FIFO pada sistem dilakukan dengan 2 tahap utama yaitu :

1. Mengurutkan barang berdasarkan tanggal masuknya barang kedalam gudang setiap kali pengguna

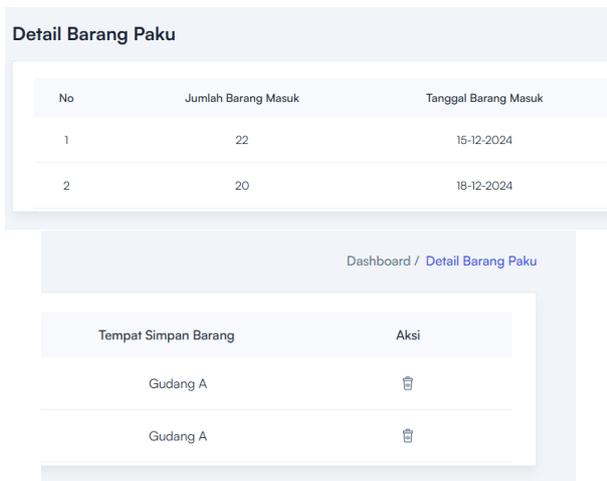
menambahkan barang. Hal ini bertujuan untuk memastikan bahwa data barang yang tersimpan tetap terurut sesuai urutan tanggal kedatangan.

2. Setelah barang sudah terurut berdasarkan tanggalnya, maka pengeluaran barang diambil dari data terdepan yaitu data dengan tanggal paling awal mengikuti prinsip dari metode FIFO.

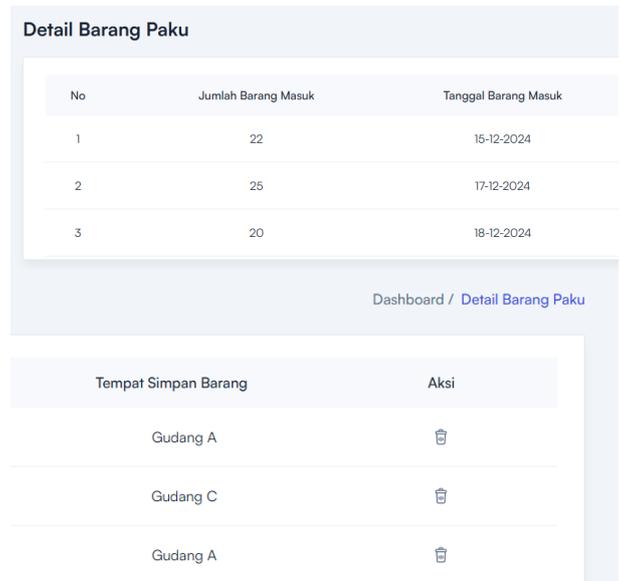


Gbr. 21 Urutan barang awal

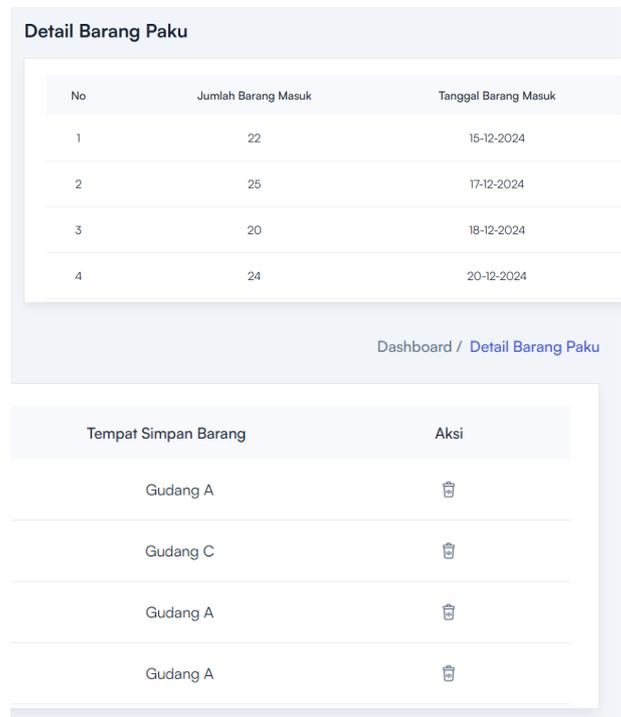
Gbr. 21 menunjukkan kondisi urutan barang ketika masuk kedalam sistem pertama kali. Proses penyortiran berdasarkan tanggal masuknya barang kedalam gudang ketika pengguna menambahkan barang di halaman tambah barang.



Gbr. 22 Penambahan barang di awal urutan



Gbr. 23 Penambahan barang ditengah urutan



Gbr. 24 Penambahan barang di akhir urutan

Gbr. 22, Gbr. 23, dan Gbr. 24 menunjukkan 3 skenario penambahan barang dalam sistem. Gbr. 22 menunjukkan skenario barang masuk kedalam gudang diawal urutan, Gbr. 23 menunjukkan skenario barang masuk kedalam gudang di tengah urutan dan Gbr. 24 menunjukkan skenario barang masuk diakhir urutan. Dengan penerapan seperti itu, maka dapat dipastikan pengurutan barang dapat dilakukan oleh sistem berdasarkan waktu masuk barang kedalam gudang. Selanjutnya setelah barang berhasil diurutkan, barang yang akan keluar akan mengikuti juga berdasarkan

barang yang sudah diurutkan. Penerapan ini diterapkan di halaman keluar barang.

Gbr. 25 Input keluar barang

No	Jumlah Barang Masuk	Tanggal Barang Masuk
1	20	15-12-2024
2	25	17-12-2024
3	20	18-12-2024
4	24	20-12-2024

Gbr. 26 Urutan barang setelah proses pengeluaran

Gbr. 25 dan Gbr. 26 menunjukkan proses pengeluaran barang yang menerapkan metode FIFO. Pengeluaran tersebut memprioritaskan untuk mengeluarkan barang dengan tanggal paling awal yaitu urutan nomor 1 untuk keluar terlebih dahulu. Dapat dibuktikan dengan kondisi urutan barang di Gbr. 26

Gbr. 27 Input keluar barang lebih banyak

No	Jumlah Barang Masuk	Tanggal Barang Masuk
1	19	20-12-2024

Gbr. 28 Urutan barang setelah proses pengeluaran

Gbr. 27 dan Gbr. 28 menunjukkan proses pengeluaran barang dengan jumlah yang lebih banyak. Berdasarkan Gbr. 27 pengeluaran barang mencapai angka 70. Oleh karena itu, sistem akan memprioritaskan pengeluaran barang dari nomor urut 1 yang merupakan barang dengan tanggal masuk barang paling awal. Apabila urutan nomor 1 tidak mencukupi permintaan pengeluaran, maka sistem akan mengambil urutan nomor selanjutnya yaitu urutan nomor 2 dan seterusnya.

B. Pengujian

Pengujian pada sistem informasi ini menggunakan 2 teknik pengujian yaitu teknik blackbox testing dan whitebox testing. Berikut adalah hasil dari kedua pengujian yang digunakan.

a) Blackbox testing

TABEL III
BLACK BOX FIFO

Skenario	Perilaku yang diberikan	Hasil yang diharapkan	Status pengujian

FIFO In (barang masuk)			
Menginput barang pertama kali	Menginput barang baru di halaman input barang dengan tanggal masuk 01-01-2025 sejumlah 100	Barang menjadi urutan pertama dalam urutan barang tersebut	Valid
Menambahkan barang dengan tanggal setelah barang pertama	Menambahkan barang tersebut di halaman tambah barang dengan tanggal 02-01-2025 sejumlah 50	Barang akan menjadi urutan kedua dalam urutan barang tersebut	Valid
Menambahkan barang dengan tanggal sebelum barang pertama	Menambahkan barang tersebut di halaman tambah barang dengan tanggal 30-12-2024 sejumlah 50	Barang akan menjadi urutan pertama dalam urutan barang tersebut, menggantikan urutan pertama saat ini	Valid
Menambahkan barang dengan tanggal di antara dua barang	Menambahkan barang tersebut di halaman tambah barang dengan tanggal 31-12-2024 sejumlah 100	Barang akan menjadi urutan kedua dalam urutan barang tersebut menggantikan urutan kedua saat ini	Valid
FIFO Out (barang keluar)			
Mengeluarkan barang lebih besar dari stok total yang tersimpan	Mengeluarkan barang tersebut di halaman keluar barang dengan jumlah lebih besar dari yang tersedia (>200)	Sistem akan memberitahu bahwa pengeluaran barang tidak bisa terjadi karena stok barang saat ini tidak mencukupi	Valid

Mengeluarkan sebagian barang dari urutan pertama	Mengeluarkan barang tersebut di halaman keluar barang dengan jumlah 20	Sistem akan mengurangi urutan pertama barang tersebut sebesar 20 menjadi 30	Valid
Mengeluarkan seluruh barang dari urutan pertama	Mengeluarkan barang tersebut di halaman keluar barang dengan jumlah 30	Sistem akan mengurangi urutan pertama barang tersebut sebesar 30 menjadi 0 dan menghapusnya dari urutan barang	Valid
Mengeluarkan barang lebih besar dari urutan pertama tetapi lebih kecil dari stok total barang yang tersimpan	Mengeluarkan barang tersebut di halaman keluar barang dengan jumlah 150	Sistem akan mengurangi urutan pertama barang terlebih dahulu sebanyak 100 lalu menghapusnya dan mengurangi urutan selanjutnya sebanyak 50	Valid
Mengeluarkan barang yang menghabiskan seluruh stok barang tersimpan	Mengeluarkan barang tersebut di halaman keluar barang dengan jumlah 100	Sistem akan mengurangi urutan pertama barang terlebih dahulu sebanyak 150 lalu menghapusnya dan mengurangi urutan selanjutnya sebanyak 50 lalu menghapusnya juga	Valid

TABEL III
 BLACK BOX REORDER POINT

Skenario	Perilaku yang diberikan	Hasil yang diharapkan	Status pengujian
Menghitung ROP dengan stok lebih besar dari ROP	Menginput barang baru di halaman input barang dengan nilai jumlah barang = 100, lead time = 5, rerata penjualan = 10, safety stock = 10, dan satuan waktu yaitu hari	Sistem menghitung nilai ROP berdasarkan input dan mendapatkan nilai 60 tanpa peringatan kekurangan barang	Valid
Menghitung ROP dengan stok lebih kecil dari ROP	Menginput barang baru di halaman input barang dengan nilai jumlah barang 40, lead time = 5, rerata penjualan = 20, safety stock = 25, dan satuan waktu yaitu hari	Sistem menghitung nilai ROP berdasarkan input dan mendapatkan nilai 60 dengan peringatan kekurangan barang	Valid
Menghitung ROP dengan kondisi stok sama dengan nilai ROP	Menginput barang baru dengan jumlah barang = 60, lead time = 4, rerata penjualan = 10, safety stock = 20, dan satuan waktu yaitu hari	Sistem menghitung nilai ROP berdasarkan input dan mendapatkan nilai 60 dengan peringatan kekurangan barang	Valid

TABEL IV
 UPDATE REORDER POINT

Perilaku yang diberikan	Hasil yang diharapkan	Status pengujian
-------------------------	-----------------------	------------------

Sistem dijalankan pada hari selain sabtu	Proses perhitungan rerata dan ROP tidak akan dijalankan	Valid
Sistem dijalankan pada hari sabtu selain jam 16.00		Valid
Barang yang tersimpan belum sampai 6 hari	Proses perhitungan rerata dan ROP tidak akan dijalankan pada barang tersebut	Valid
Barang yang tersimpan sudah lebih dari 6 hari tanpa pengeluaran di minggu terakhir	Proses perhitungan rerata dan ROP tidak akan dijalankan dan sistem memperbarui nilai rerata menjadi 0 dan ROP sama dengan nilai safety stock	Valid
Barang yang tersimpan sudah lebih dari 6 hari dan ada pengeluaran di minggu terakhir	Sistem akan menghitung rerata pengeluaran barang dalam minggu terakhir dan memperbarui nilai rerata dan ROP barang tersebut	Valid

b) Whitebox testing

Pengujian whitebox testing ini akan berfokus kepada pengujian-pengujian metode yang digunakan dalam sistem seperti metode First in First Out (FIFO) yang berfokus pada pengurutan barang ketika masuk dan prioritas barang keluar berdasarkan tanggal masuk dan metode Reorder Point (ROP).

1. First In First Out (FIFO)

Dalam proses melakukan pengujian terhadap fungsi FIFO yang digunakan sistem pengelolaan barang ini, pengujian dilakukan untuk menguji 2 aspek yaitu :

1) Pengurutan barang masuk

Dalam pengujian ini, bertujuan untuk memastikan bahwa barang ditambahkan akan disisipkan kedalam detail barang dengan urutan yang benar berdasarkan tanggal yang mengikuti aturan FIFO. Berdasarkan pengujian yang

dilakukan, terdapat tiga scenario pengujian yang dilakukan. Sebelum masuk ke pengujian, disediakan urutan barang awal yang berfungsi sebagai data awal untuk memastikan bahwa pengujian memiliki kondisi dasar yang konsisten. Data awal memiliki 2 barang dengan barang pertama dengan tanggal 10-10-2023 dan barang kedua dengan tanggal 20-10-2023. Tabel 4.14 menjelaskan tentang pengujian yang dilakukan.

Tabel V
 Pengujian FIFO In

Data Awal	Data yang diterima	Hasil yang diharapkan
Urutan Pertama tanggal 10-10-2023 Urutan Kedua tanggal 20-10-2023	Barang masuk ditanggal 05-10-2023	Menjadi urutan pertama dalam susunan barang
	Barang masuk ditanggal 15-10-2023	Menjadi urutan kedua dalam susunan barang
	Barang masuk ditanggal 25-10-2023	Menjadi urutan terakhir dalam susunan barang

2) Prioritas barang keluar

Dalam pengujian ini, bertujuan untuk memastikan bahwa barang yang keluar adalah barang dengan urutan pertama karena telah melewati pengurutan barang ketika barang masuk. Proses pengujian ini adalah proses testing pengeluaran barang. Sebelum melakukan testing, dibuat data barang awal terlebih dahulu yang akan dikeluarkan nantinya. Setelah itu testing akan berjalan dengan mengeluarkan barang yang lebih besar daripada total jumlah barang yang tersimpan. Berdasarkan kasus tersebut, diharapkan sistem tidak dapat mengeluarkan barang karena barang yang tersimpan tidak cukup. Selanjutnya adalah pengujian barang ketika barang permintaan lebih kecil daripada jumlah barang yang tersimpan.

TABEL VI
 PENGUJIAN FIFO OUT

Data Awal	Jumlah yang akan dikeluarkan	Hasil yang diharapkan
Urutan pertama sebanyak 5 satuan Urutan kedua	50 satuan	Pengeluaran barang tidak akan terjadi

sebanyak 10 satuan Urutan ketiga sebanyak 8 satuan	3 satuan	Urutan pertama 2 satuan Urutan kedua 10 satuan Urutan ketiga 8 satuan
	5 satuan	Urutan pertama 10 satuan Urutan kedua 8 satuan
	10 satuan	Urutan pertama 5 satuan Urutan kedua 8 satuan
	15 satuan	Urutan pertama 8 satuan
	23 satuan	Pengeluaran barang terjadi dengan tidak tersisa barang

```

PASS tests/fifo-in.test.js
Pengujian saat barang masuk
  ✓ Menambahkan di urutan awal (35 ms)
  ✓ Menambahkan di urutan tengah (3 ms)
  ✓ Menambahkan di urutan akhir (3 ms)

PASS tests/fifo-out.test.js
Pengujian saat barang keluar
  ✓ Apabila permintaan keluar lebih besar daripada jumlah barang (35 ms)
  ✓ Mengurangi jumlah yang benar dari detail barang (1) (2 ms)
  ✓ Mengurangi jumlah yang benar dari detail barang (2) (1 ms)
  ✓ Mengurangi jumlah yang benar dari detail barang (3) (2 ms)
  ✓ Mengurangi jumlah yang benar dari detail barang (4) (1 ms)
  ✓ Mengurangi jumlah yang benar dari detail barang (5) (1 ms)

Test Suites: 2 passed, 2 total
Tests: 9 passed, 9 total
Snapshots: 0 total
Time: 1.798 s, estimated 2 s
Ran all test suites.
    
```

Gbr. 29 Hasil pengujian whitebox FIFO

Gbr. 29 menunjukkan hasil dari pengujian yang dijalankan proses FIFO (First in First Out) pada pengelolaan barang masuk dan keluar. Semua pengujian menunjukkan status PASS, yang mengartikan bahwa sistem telah menerapkan metode FIFO dengan baik dan benar sesuai yang diharapkan.

2. Reorder Point (ROP)

Dalam proses melakukan pengujian terhadap fungsi ROP yang digunakan sistem pengelolaan barang ini, pengujian dilakukan untuk menguji 2 aspek yaitu :

1) Konversi satuan waktu

Sistem menyediakan 3 satuan waktu yaitu jam, hari dan bulan yang harus dikonversi menjadi satuan hari sebelum dihitung menggunakan fungsi ROP. Proses pengujian ini dilakukan untuk memastikan bahwa perhitungan fungsi konversi berjalan sesuai harapan.

2) Perhitungan nilai ROP

Setelah mengonversi satuan waktu menjadi hari, seluruh nilai yang dibutuhkan untuk menghitung Reorder Point (ROP) seperti rerata penjualan harian, safety stock, dan lama waktu pengantaran, akan diuji untuk memastikan fungsi ROP menghasilkan nilai yang akurat. Pengujian ini adalah proses pengujian fungsi untuk memastikan bahwa perhitungan Reorder Point (ROP) berjalan dengan baik dan benar menggunakan berbagai satuan waktu seperti hari, jam, dan bulan. Nilai yang digunakan untuk menghitung ROP meliputi rerata penjualan per hari (rerata), safety stock, dan lama waktu pengantaran dalam satuan tertentu. Setiap pengujian memverifikasi hasil perhitungan ROP berdasarkan konversi yang sesuai untuk setiap satuan waktu.

```
PASS src/tests/rop.test.js
Mengonversi satuan waktu yang dipilih user menjadi hari
  ✓ mengonversi lead time dari 48 jam ke 2 hari (5 ms)
  ✓ mengonversi lead time dari 12 jam ke 0,5 hari (1 ms)
  ✓ mengembalikan lead time jika satuan waktu adalah hari (1 ms)
  ✓ mengonversi lead time dari 0,5 bulan ke 15 hari
  ✓ mengonversi lead time dari 1 bulan ke 30 hari (1 ms)
  ✓ mengonversi lead time dari 1,5 bulan ke 45 hari (1 ms)
Menghitung ROP
  ✓ menghitung nilai ROP menggunakan satuan hari (1 ms)
  ✓ menghitung nilai ROP menggunakan satuan jam (1 ms)
  ✓ menghitung nilai ROP menggunakan satuan bulan (1 ms)

Test Suites: 1 passed, 1 total
Tests: 9 passed, 9 total
Snapshots: 0 total
Time: 3.919 s
Ran all test suites.
```

Gbr. 30 Hasil pengujian whitebox Reorder Point

Gbr. 30 menunjukkan hasil dari pengujian otomatis yang telah dijalankan untuk mengonversi satuan waktu dan menghitung Reorder Point (ROP). Semua pengujian berhasil (status PASS) tanpa adanya kegagalan, yang menunjukkan bahwa fungsi bekerja sesuai dengan ekspektasi.

IV. KESIMPULAN

Simpulan yang dapat diambil dari penelitian "Penerapan Metode FIFO dan ROP pada Sistem Inventory UD Salam 51 Berbasis Website" adalah sebagai berikut :

1. Perancangan sistem berbasis website yang menerapkan metode FIFO dilakukan melalui beberapa tahapan yang

terstruktur. Pertama, dilakukan analisis kebutuhan pengguna untuk menentukan fitur utama sistem, seperti pencatatan barang masuk dan keluar berdasarkan urutan waktu. Kedua, dirancang arsitektur sistem yang terdiri dari frontend, backend, dan basis data. Sistem ini dirancang dengan fokus pada penerapan metode FIFO, yaitu dengan memastikan barang yang masuk terlebih dahulu akan keluar lebih dulu. Pengurutan data barang berdasarkan tanggal diimplementasikan pada tingkat basis data menggunakan MongoDB. Backend dikembangkan menggunakan Node.js untuk mengolah transaksi barang masuk dan keluar sesuai logika FIFO, sementara frontend dirancang menggunakan React.js untuk memberikan antarmuka yang intuitif bagi pengguna, seperti fitur pencatatan dan monitoring stok barang.

2. Untuk menentukan batas angka minimal suatu barang, metode ROP digunakan dengan memperhitungkan rata-rata penggunaan barang, lead time dan safety stock. Dengan demikian, sistem mampu memberikan peringatan untuk melakukan pemesanan dengan tepat waktu sebelum stok barang habis sehingga pengelolaan stok menjadi lebih optimal
3. Hasil pengujian algoritma FIFO dan ROP menggunakan berbagai metode pengujian, seperti Blackbox Testing dan Whitebox Testing, menunjukkan bahwa sistem berfungsi dengan baik sesuai dengan spesifikasi yang dirancang. Pengujian membuktikan bahwa algoritma FIFO mampu mengelola urutan barang secara akurat, dan algoritma ROP berhasil menghitung batas minimum stok barang dengan presisi, sehingga mendukung pengelolaan inventaris yang efektif dan efisien.

UCAPAN TERIMA KASIH

Puji syukur kehadirat dan Allah SWT, yang telah memberikan kesehatan dan rahmat yang melimpah sehingga penulis dapat menyelesaikan artikel ini dengan baik. Terima kasih kepada kedua orang tua serta keluarga yang selalu mendokan, memberikan dukungan dan semangat. Terimakasih juga penulis ingin sampaikan kepada dosen pembimbing yang senantiasa memberikan saran dan masukan yang terbaik. Tak lupa, penulis mengucapkan terima kasih kepada teman-teman yang senantiasa membantu serta mendukung selama proses penelitian. Terakhir, terima kasih kepada diri sendiri karena telah mampu menyelesaikan artikel ini dengan baik.

REFERENSI

- [1] Mulyani, F., & Haliza, N. (2021). Analisis Perkembangan Ilmu Pengetahuan dan Teknologi (Iptek) Dalam Pendidikan. Deleted Journal, 3(1), 101–109. <https://doi.org/10.31004/jpdk.v3i1.1432>
- [2] Herman, dan Alvi Geovanny. (2022). Analisis Rendering Performa Antara Server Side Dan Client Side Pada Web Application. Tiban Indah, Kec. Sekupang, Kota Batam: Jurnal Ilmiah Betrik, Volume.13, No.03.
- [3] Astiyani, E., Apriani, D., & Sari, M. M. (2020). Sistem Informasi Pengendalian Persediaan Barang Menggunakan Metode Economic Order Quantity (EOQ). IJAcc, 1(2), 132–139. <https://doi.org/10.33050/jakbi.v1i2.1414>

- [4] S. I. W. Jacobus and J. S. Sumarauw, "Analisis Sistem Manajemen Pergudangan Pada Cv. Pasific Indah Manado Warehousing Management System Analysis on Cv. Pasific Indah Manado," *Anal. Sist. Manajemen* 2278 J. EMBA, vol. 6, no. 4, pp. 2278–2287, 2018.
- [5] Dianto, Z. N., & Widati, E. (2023). ANALISIS MANAGEMENT INVENTORY UNTUK MENGHINDARI DEATH STOCK PRODUCT DI TB. SINAR BARU. *VALUE*, 4(1), 50–72. <https://doi.org/10.36490/value.v4i1.631>
- [6] Rahman, A. A., & Novianty, N. (2022). Implementasi Metode FIFO Pada Sistem Informasi Persediaan Barang Dagang. *Jurnal Teknologi Informasi*, 44-47.
- [7] Asnal, H., Sani, N., Anam, M. K., Erlinda, S., & Jamaris, M. (2022). Sistem Monitoring Persediaan Stok Onderdil Menggunakan Metode Reorder Point Pada Sani Computer. *JSR Jaringan Sistem Informasi Robotik*, 6(2), 305–310. <https://doi.org/10.58486/jsr.v6i2.171>
- [8] Mardiaty, D., & Saputra, Y. (2023). Rancang Bangun Inventory System Menggunakan Metode Reorder Point (Rop) Pada Toko Bangunan Irhas Padang. *ZONAsi Jurnal Sistem Informasi*, 5(1), 163–178. <https://doi.org/10.31849/zn.v5i1.12758>
- [9] A Jartarghar, H. ., Rao Salanke, G. ., A.R, A. K. ., G.S, S. ., & Dalali, S. . (2022). React Apps with Server-Side Rendering: Next.js. *Journal of Telecommunication, Electronic and Computer Engineering (JTEC)*, 14(4), 25–29.
- [10] Guntara, R. G., & Azkarin, V. (2023, July 8). Pembangunan REST API Human Resource Information System Domain Pengelolaan User Dengan Menggunakan Framework Express JS dan Node.js.
- [11] Purwandari, N., & Fauzi, A. (2020). Perancangan sistem informasi manajemen pada toko XYZ berbasis desktop. *Jurnal Sistem Informasi Bisnis (JUNSIBI)*, 1(2), 54–64. <https://doi.org/10.55122/junsibi.v1i2.171>
- [12] Suwirmayanti, N. L. G. P., Aryanto, I. K. a. A., Putra, I. N. W., Sukerti, N. K., & Hadi, R. (2020). Penerapan Helpdesk System dengan Pengujian Blackbox Testing. *Jurnal Ilmiah Intech Information Technology Journal of UMUS*, 2(02). <https://doi.org/10.46772/intech.v2i02.290>
- [13] H. R. R. Zen, I. Nuryasin, and Prodi Informatika, Universitas Muhammadiyah Malang, Jl. Raya Tlogomas No.246, Babatan, Tegalondo, Kecamatan Lowokwaru, Kota Malang, "PENERAPAN WHITEBOX TESTING PADA PENGUJIAN SISTEM MENGGUNAKAN TEKNIK BASIS PATH," Jun. 2024. [Online]. Available: <https://doi.org/10.35145/joisie.v8i1.4229>