

PERANCANGAN SISTEM INFORMASI MANAJEMEN INVENTORY BERBASIS *WEB* DENGAN METODE *RAPID APPLICATION DEVELOPMENT* (STUDI KASUS: PT TARINDO JUWANA)

Desi Armanda Sari *¹, Singgih Saptadi

¹Departemen Teknik Industri, Fakultas Teknik, Universitas Diponegoro,
Jl. Prof. Soedarto, SH, Kampus Undip Tembalang, Semarang, Indonesia 50275

Abstrak

Di era teknologi informasi yang berkembang pesat, cara operasional bisnis mengalami transformasi signifikan, termasuk dalam manajemen inventaris yang krusial bagi kelangsungan bisnis. Penelitian ini berfokus pada pengembangan dan implementasi sistem informasi di PT. Tarindo untuk meningkatkan efisiensi administrasi dan operasional. Hasil penelitian menunjukkan bahwa sistem informasi yang dirancang memungkinkan pembuatan laporan persediaan dengan cepat dan mudah, sehingga mengoptimalkan proses administrasi. Sistem ini juga menyediakan informasi stok barang secara real-time di setiap gudang, yang mengurangi ketidaksesuaian antara pesanan dan stok barang yang tersedia, meningkatkan akurasi pengelolaan persediaan, dan memperbaiki efisiensi operasional. Selain itu, aplikasi monitoring kebutuhan bahan baku yang tepat waktu membantu mengurangi ketidaksesuaian antara pemesanan bahan baku ke pemasok dan kebutuhan sebenarnya di setiap gudang, serta meningkatkan efisiensi dalam rantai pasokan. Sistem informasi ini terintegrasi dengan database, yang mengurangi risiko kehilangan data dan mempermudah pengoperasian, sehingga mengoptimalkan manajemen persediaan barang di PT. Tarindo. Metode Rapid Application Development (RAD) digunakan dalam pengembangan sistem ini karena kemampuannya untuk membangun aplikasi yang sesuai dengan kebutuhan bisnis secara cepat dan efektif, mengurangi waktu dan biaya pengembangan, serta meningkatkan efisiensi dan akurasi dalam manajemen persediaan.

Kata kunci: *Manajemen Inventaris, Sistem Informasi, Rapid Application Development (RAD), Efisiensi Operasional, Informasi Stok Real-time.*

Abstract

[DESIGN OF A WEB-BASED INVENTORY MANAGEMENT INFORMATION SYSTEM USING RAPID APPLICATION DEVELOPMENT METHOD (CASE STUDY: PT TARINDO JUWANA)] *In the rapidly evolving era of information technology, business operations have undergone significant transformations, especially in crucial areas like inventory management. This study focuses on the development and implementation of an information system at PT. Tarindo to enhance administrative and operational efficiency. The findings reveal that the designed information system enables the quick and easy creation of inventory reports, thereby optimizing administrative processes. This system also provides real-time stock information across warehouses, reducing discrepancies between orders and available stock, improving inventory management accuracy, and enhancing operational efficiency. Additionally, the timely monitoring application for raw material needs helps reduce mismatches between raw material orders from suppliers and actual needs in each warehouse, thereby improving supply chain efficiency. The integrated database system reduces the risk of data loss and simplifies operations, thereby optimizing inventory management at PT. Tarindo. The Rapid Application Development (RAD) method was utilized in developing this system due to its capability to quickly and effectively build applications tailored to business needs, reducing development time and costs while increasing efficiency and accuracy in inventory management.*

Keywords: *Inventory Management, Information System, Rapid Application Development (RAD), Operational Efficiency, Real-time Stock Information.*

1. Pendahuluan

Perkembangan teknologi informasi sekarang ini telah mempengaruhi cara suatu bisnis dalam beroperasi. Di berbagai perusahaan manufaktur dan jasa, sistem informasi telah menjadi suatu aset yang krusial dalam mendukung beragam kegiatan operasional. Sistem informasi juga berperan sebagai alat atau pendukung untuk meningkatkan produktivitas, efisiensi, dan kemudahan dalam menjalankan operasi. Dengan memilih sistem informasi yang sesuai, pengolahan data dapat dilakukan dengan efektif dan efisien.

Salah satu aspek penting dari pengelolaan bisnis adalah manajemen inventaris. Keakuratan dan ketepatan inventaris dapat membantu perusahaan menghindari masalah kekurangan atau kelebihan stok serta mengurangi biaya produksi [1]. Inventori merujuk pada konsep yang menggambarkan sumber daya yang tersedia atau belum dimanfaatkan sepenuhnya [2]. Salah satu contoh penerapan teknologi informasi adalah dalam sistem manajemen inventaris barang, di mana pencatatan inventori sangat penting bagi perusahaan untuk melaporkan stok barang di gudang, baik barang jadi maupun bahan baku, guna mendukung pengambilan keputusan dan meningkatkan kualitas sistem yang ada [3].

Untuk meningkatkan penggunaan teknologi informasi dalam manajemen persediaan, perusahaan dapat mengadopsi metode pengembangan perangkat lunak yang cepat dan fleksibel, seperti *Rapid Application Development* (RAD). Dengan metode RAD, perusahaan dapat membangun aplikasi yang sesuai dengan kebutuhan bisnis secara cepat dan efektif [4]. Penggunaan metode RAD dalam pengembangan sistem persediaan memungkinkan perusahaan untuk mengurangi waktu dan biaya yang diperlukan untuk pengembangan sistem, sambil meningkatkan efisiensi dan akurasi dalam pengelolaan persediaan. *Rapid Application Development* adalah salah satu metode yang muncul sebagai tanggapan terhadap kelemahan metode pengembangan *waterfall* dan variasinya [5]. Dengan menerapkan metode *Rapid Application Development*, proses perancangan sistem dapat dipersingkat, sehingga menghasilkan sistem yang lebih baik dalam waktu yang lebih singkat [6].

PT. Tarindo adalah suatu perusahaan penghasil Kran Air Kuningan dan Kran Air PVC dengan No. Registrasi IDM 000350006 dengan diketuai oleh Komisaris Ir.H. Teguh Budi Pranyono dan H. Nur Adi Widodo SE. PT. Tarindo dirintis oleh Bapak Soetarjo semenjak tahun 1975 sampai saat ini, dengan merek dagang "AMICO". PT tarindo ini memiliki 3 cabang lokasi perusahaan yaitu Tarindo 1 yang terletak di jalan Emas No. 489 Growong Lor, Juwana, Pati 59185. Kemudian Tarindo II terletak di jalan Emas No. 489 Growong Lor, Juwana, Pati 59185 dan Tarindo III terletak di jalan Raya Juwana-Tayu, Growong Kidul, Juwana, Pati. Berdasarkan hasil wawancara dengan

saudara Nurhayati selaku staff divisi administrasi di PT Tarindo serta pengamatan secara langsung. Perusahaan ini menggunakan 3 cabang pabrik dalam melakukan proses produksi produk kran di mana di setiap cabang pabriknya terdapat gudang untuk penyimpanan persediaan bahan baku, barang setengah jadi, maupun barang jadi. Alurnya di mulai dari gudang cabang pabrik Tarindo I yang menyimpan bahan baku mentah berupa tuangan kuningan dan bijih plastik yang nantinya akan diolah menjadi produk gotri setengah jadi dan kran pvc setengah jadi. Produk gotri setengah jadi akan dikirim ke gudang cabang Tarindo II untuk diolah menjadi produk gotri jadi dan produk kran pvc setengah jadi dikirim ke cabang Tarindo III. Produk gotri yang sudah jadi nantinya juga akan dikirim ke gudang Tarindo III untuk dilakukan *assembly* dan inspeksi akhir. Di PT Tarindo III ini juga merupakan tempat atau cabang di mana terjadinya penerimaan pesanan dari klien serta permintaan pesanan ke *supplier*. Dalam rekapitulasi pencatatan stoknya, bagian administrasi PT Tarindo III perlu menunggu dan mengumpulkan semua informasi barang masuk perperiode dikurang dengan barang keluar dari setiap cabang PT Tarindo, sehingga pergudangan pada PT Tarindo ini kerap kali mengalami beberapa kendala antara lain, ketidaksesuaian antara laporan dan pengiriman barang, kesulitan pelacakan barang masuk, keluar, dan produksi, ketidaktersediaan laporan produksi harian, serta kelebihan pengiriman dari pemasok maupun ketidaksesuaian banyaknya permintaan dengan stok yang ada. Untuk mengatasi masalah ini, diperlukan pengembangan sistem yang sesuai guna mengelola dan mencatat berbagai transaksi keluar masuknya barang pada PT. Tarindo secara lebih terintegrasi antar gudang di setiap cabang perusahaan. Dengan ruang lingkup proyek yang terbatas dan kebutuhan akan kecepatan sistem, oleh karena itu metode *Rapid Application Development* dipilih sebagai pendekatan pengembangan yang cocok.

2. Literature Review

2.1 System Information

Sistem informasi adalah sistem yang mengelola transaksi harian dan mendukung fungsi operasional, manajerial, serta strategi organisasi untuk menyediakan informasi yang dibutuhkan dalam pengambilan keputusan. Sistem ini menyediakan informasi di semua tingkatan organisasi, kapan saja diperlukan, dengan menyimpan, mengambil, mengubah, mengolah, dan mengkomunikasikan data menggunakan teknologi informasi. Pengembangan sistem informasi diperlukan untuk mengatasi masalah, mendukung pertumbuhan organisasi, memanfaatkan peluang, dan memenuhi instruksi pimpinan atau pemerintah [7].

- a. Menurut Anggraeni dan Irviani (2017), fungsi utama sistem informasi meliputi:

- b. Memungkinkan akses data langsung tanpa perantara.
- c. Meningkatkan produktivitas pengembangan dan pemeliharaan sistem.
- d. Menjamin kualitas dan keterampilan penggunaan sistem informasi.
- e. Mengidentifikasi kebutuhan keterampilan pendukung sistem.
- f. Mengantisipasi konsekuensi ekonomi.
- g. Menetapkan investasi untuk sistem informasi.
- h. Mengembangkan proses perencanaan yang efektif.

Sistem informasi terdiri dari beberapa komponen utama:

- a. Blok masukan (*input block*): Metode dan media untuk memasukkan data ke dalam sistem.
- b. Blok model (*model blocks*): Prosedur, logika, dan metode matematika untuk memproses data.
- c. Blok keluaran (*output blocks*): Hasil yang berkualitas dan dokumentasi yang berguna.
- d. Blok teknologi (*technology blocks*): Teknologi untuk menjalankan model, menerima input, menghasilkan output, menyimpan dan mengakses data, serta mengendalikan sistem.
- e. Blok basis data (*data base block*): Kumpulan data relevan yang disimpan dan dioperasikan oleh perangkat lunak.
- f. Blok kendali (*control blocks*): Kontrol untuk mencegah kerusakan dan memperbaiki masalah.

Teknologi dalam sistem informasi terdiri dari teknisi (*brainware*), perangkat keras (*hardware*), dan perangkat lunak (*software*).

2.2 Business Process

Proses bisnis adalah serangkaian alat untuk mengatur kegiatan dan meningkatkan pemahaman tentang keterkaitan antar kegiatan tersebut [8]. Definisi lain dari proses adalah sekelompok kegiatan yang dirancang untuk menghasilkan *output* tertentu bagi pelanggan bisnis [9]. Menurut Hammer dan Champy yang dikutip dalam Grosskopf dkk. (2009), proses bisnis terdiri dari kegiatan-kegiatan yang mengambil masukan dan menghasilkan *output* yang bernilai bagi pelanggan.

Kegiatan proses bisnis dapat dilakukan secara manual atau dengan bantuan sistem informasi [8]. Sebuah proses bisnis harus memiliki tujuan yang jelas, masukan, keluaran, menggunakan sumber daya, melibatkan beberapa tahapan kegiatan, mempengaruhi lebih dari satu unit dalam organisasi, dan menciptakan nilai bagi pelanggan [9].

Menurut Grosskopf dkk. (2009), sebuah proses bisnis terdiri dari serangkaian kegiatan yang dilakukan secara terkoordinasi dalam lingkungan bisnis dan teknis, yang bersama-sama mewujudkan strategi bisnis. Meskipun umumnya diterapkan di dalam suatu organisasi, proses bisnis juga dapat berinteraksi dengan proses yang dilakukan oleh organisasi lain.

2.3 Manajemen Inventory

Inventory adalah aset yang dimiliki oleh perusahaan untuk dijual dalam operasi bisnis normal atau untuk digunakan/dikonsumsi. Ini merupakan investasi aset lancar terbesar bagi perusahaan perdagangan dan manufaktur [10]. *Inventory* mencakup berbagai jenis, mulai dari bahan mentah, barang setengah jadi, hingga barang jadi, yang memiliki berbagai fungsi [11]:

- a. Mengurangi risiko keterlambatan dalam pengadaan material untuk proses produksi.
- b. Meminimalkan risiko penerimaan bahan baku yang salah sehingga harus dikembalikan.
- c. Menyimpan barang musiman untuk memastikan ketersediaan saat tidak ada di pasar.
- d. Menjaga stabilitas operasional produksi untuk memastikan kelancaran proses produksi.
- e. Mengoptimalkan penggunaan mesin dengan mencegah berhentinya produksi karena kehabisan *inventory* (*stock out*).
- f. Meningkatkan layanan pelanggan dengan memastikan ketersediaan barang sesuai kebutuhan.
- g. Manajemen *inventory* adalah sistem yang bertanggung jawab untuk merencanakan dan mengawasi *inventory* dari bahan baku hingga pengiriman ke pelanggan [12].

Manajemen *inventory* yang efektif bertujuan untuk mencapai keuntungan optimal dengan cara:

- a. Maksimalkan pelayanan pelanggan.
- b. Minimalkan biaya operasional.
- c. Minimalkan investasi dalam *inventory*.

Karena *inventory* disimpan di gudang, peran manajemen *inventory* dan manajemen gudang sangatlah terkait. Terkadang, gudang berperan sebagai pusat distribusi karena perputaran *inventory* yang cepat. Manajemen *inventory* melibatkan kebijakan dan kontrol untuk memantau serta menyesuaikan tingkat *inventory*, sehingga sumber daya yang diperlukan tersedia tepat jumlah dan waktu [13].

2.4 Analisis Gap

Analisis *gap* digunakan untuk menentukan langkah-langkah yang diperlukan untuk bergerak dari kondisi saat ini menuju kondisi yang diinginkan di masa depan. Ini melibatkan perbandingan antara kinerja aktual dan kinerja yang diharapkan atau potensial. Analisis ini juga mengidentifikasi tindakan yang diperlukan untuk mengurangi kesenjangan atau mencapai tujuan yang diharapkan, serta memperkirakan waktu, biaya, dan sumber daya yang diperlukan [14]. Langkah dari tool ini adalah sebagai berikut [14]:

- a. Menyusun *Gap Analysis checklist* yang berfungsi untuk mengidentifikasi *gap* antara prosedur tertulis dengan proses yang dilakukan berdasarkan persyaratan pada ISO 9001:2015.
- b. Memberikan *Gap tools* kepada orang-orang yang berpotensi untuk dijadikan infroman.

- c. Data hasil penelitian dari gap didapatkan dan dikumpulkan.
- d. Hasil *checklist* dihitung dan dapat menentukan kesenjangan menggunakan skor penentu gap.

2.5 Analisis PIECES

Metode PIECES digunakan untuk menganalisis sistem dengan fokus pada enam aspek utama: Kinerja, Informasi, Ekonomi, Pengendalian, Efisiensi, dan Layanan [15]. Analisis ini penting sebelum pengembangan sistem informasi untuk mengidentifikasi masalah yang ada.

- a. Kinerja: Evaluasi seberapa baik sistem dalam mencapai tujuan yang diinginkan, diukur dari *response time* dan *throughput*.
- b. Informasi: Penilaian terhadap kualitas data yang masuk (*inputs*) dan hasil keluaran (*outputs*) dari sistem.
- c. Ekonomi: Pertimbangan efisiensi biaya dan manfaat dari prosedur yang ada.
- d. Pengendalian: Evaluasi terhadap keamanan dan kemampuan pengendalian sistem untuk mendeteksi kesalahan atau kecurangan.
- e. Efisiensi: Penilaian terhadap efisiensi operasional sistem yang dapat ditingkatkan.
- f. Layanan: Peningkatan kualitas layanan kepada pengguna dengan memastikan sistem mudah digunakan dan memberikan layanan yang baik.

Metode PIECES memberikan kerangka kerja yang komprehensif untuk mengevaluasi dan memperbaiki sistem informasi agar lebih efektif dan efisien [16].

2.6 Diagram Use Case

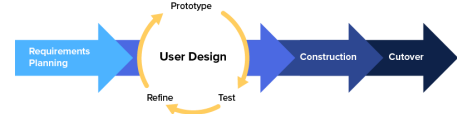
Use case adalah representasi diagramatis dari aktivitas sistem sebagai tanggapan terhadap permintaan pengguna [17]. Diagram *Use case* secara statis menunjukkan hubungan antara *Use case* (kemampuan sistem) dan aktor (pengguna atau sistem lain). Digunakan untuk memodelkan dan mengontrol perilaku sistem sesuai kebutuhan pengguna. Diagram ini menggambarkan serangkaian interaksi yang terjadi antara sistem dan pengguna dalam situasi-situasi tertentu.

| SIMBOL | NAMA | KETERANGAN |
|--------|----------------|---|
| | Actor | Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i> . |
| | Dependency | Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri. |
| | Generalization | Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>). |
| | Include | Menspesifikasikan bahwa <i>use case</i> sumber secara eksplisit. |
| | Extend | Menspesifikasikan bahwa <i>use case</i> target memperluas perilaku dari <i>use case</i> sumber pada suatu titik yang diberikan. |
| | Association | Apa yang menghubungkan antara objek satu dengan objek lainnya. |
| | System | Menspesifikasikan paket yang menampilkan sistem secara terbatas. |
| | Use Case | Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor. |
| | Collaboration | Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemen-elemennya (<i>energi</i>). |
| | Note | Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu number daya komputasi. |

Gambar 1. Use case diagram

2.7 Rapid Application Development

Rapid Application Development (RAD) adalah metode pengembangan perangkat lunak yang diperkenalkan oleh James Martin pada tahun 1991 [18]. Metode ini memungkinkan pengembang untuk membangun aplikasi dengan cepat dan efektif melalui penggunaan prototipe dan iterasi. RAD masih relevan dalam pengembangan perangkat lunak saat ini, terutama dalam konteks *Agile Development*, di mana fleksibilitas menjadi kunci utama [19].



Gambar 2. Metode Rapid Application Development

Dalam model RAD, terdapat empat tahapan utama

[20]:

- a. Perencanaan Kebutuhan (*Requirements Planning*): Tahapan di mana pengguna dan pengembang berdiskusi untuk menetapkan kebutuhan sistem informasi yang akan dikembangkan. Komunikasi yang baik antara kedua pihak sangat penting untuk kesuksesan tahapan ini.
- b. Desain Pengguna (*User Design*): Tahapan untuk merancang solusi yang sesuai dengan kebutuhan pengguna, dan menjamin bahwa rencana pengembangan sistem dapat mengatasi masalah yang ada. Penggunaan *Unified Modeling Language* (UML) umum dalam menggambarkan desain sistem.
- c. Konstruksi (*Construction*): Tahapan di mana pengembangan sebenarnya dimulai dengan penulisan kode program. Desain sistem yang telah disepakati diimplementasikan menjadi aplikasi yang dapat digunakan.
- d. Implementasi (*Cutover*): Tahapan pengujian menyeluruh terhadap sistem yang telah dibangun. Pengujian *Black Box Testing* dilakukan untuk memastikan bahwa semua komponen sistem berfungsi sesuai spesifikasi dan meminimalkan risiko *bug* atau kegagalan sistem [21].

2.8 HTML

HTML (*Hyper Text Markup Language*) adalah bahasa markup yang digunakan untuk membuat halaman web dan menampilkan berbagai informasi di dalam browser Internet [22]. Berasal dari bahasa yang sebelumnya banyak digunakan dalam penerbitan dan percetakan yang disebut *SGML* (*Standard Generalized Markup Language*), *HTML* kini telah menjadi standar yang luas digunakan untuk menampilkan halaman web. Saat ini, *HTML* merupakan standar Internet yang ditetapkan dan diawasi penggunaannya oleh *World Wide Web Consortium* (*W3C*). *HTML* terdiri dari kode-kode *tag* yang memberi instruksi kepada browser untuk menampilkan tampilan sesuai dengan yang diinginkan.

Sebuah file *HTML* dapat dibuka menggunakan berbagai *browser web* seperti *Mozilla Firefox*, *Microsoft Internet Explorer*, *Opera*, *Google Chrome*, *Safari*, dan lainnya.

2.9 Python

Python adalah bahasa pemrograman yang sangat populer saat ini. Diciptakan oleh Guido van Rossum dan dirilis pada tahun 1991, *Python* memiliki berbagai aplikasi yang luas. *Python* dapat digunakan untuk pengembangan aplikasi *web* di sisi *server*, pengembangan perangkat lunak, penyelesaian persamaan matematika, pembuatan skrip sistem, serta pemrograman *mikrokontroler (Micro-Python)* [23]. Beberapa fungsi utama *Python* termasuk kemampuannya untuk membuat aplikasi *web* di *server*, berintegrasi dengan perangkat lunak untuk mengatur alur kerja, terhubung ke sistem *database*, mengelola dan memodifikasi file, menangani data besar, serta melakukan perhitungan matematika yang kompleks. *Python* juga sangat efektif untuk pembuatan *prototipe* cepat maupun pengembangan perangkat lunak siap produksi [23].

2.10 Framework Django

Django adalah sebuah kerangka kerja yang menggunakan bahasa pemrograman *Python* untuk membangun situs *web*. Kerangka kerja ini bersifat open source dan menggunakan konsep *Model View Controller (MVC)* untuk pengembangan aplikasi *web*. Keunggulan *Django* dibandingkan dengan kerangka kerja lainnya termasuk kemampuannya dalam merepresentasikan model data dari basis data sebagai objek dalam paradigma pemrograman berorientasi objek (OOP), menjaga kerahasiaan dan integritas data, desain *URL* yang fleksibel dan mudah dipahami, sistem template yang kuat dan mudah digunakan, serta sistem cache yang dapat diimplementasikan dengan mudah. *Django* juga mendukung pengembangan aplikasi multi bahasa dan tidak memerlukan instalasi *web server* saat proses pengembangan, yang cukup dijalankan secara langsung [24].

2.11 MYSQL

MySQL adalah sebuah server basis data yang memiliki kemampuan untuk mengelola dan menyediakan data dengan cepat kepada pengguna secara multi-*User*, menggunakan bahasa *SQL (Structured Query Language)* sebagai perintah dasarnya. *MySQL* bersifat *Free Software dan Shareware*, yang berarti penggunaannya gratis. *MySQL* dapat digunakan baik sebagai klien maupun sebagai *server*, dan merupakan sistem manajemen basis data relasional (RDBMS) yang menggunakan *SQL* sebagai bahasa permintaannya. *SQL* sendiri adalah bahasa permintaan *database* yang terstruktur, dirancang untuk merelasikan tabel-tabel dalam *database* serta antar *database* [24].

3. Metode Penelitian

3.1 Objek dan Waktu Penelitian

Objek Penelitian ini adalah manajemen inventori pada gudang/warehouse di ketiga cabang PT Tarindo. Tempat pelaksanaan penelitian dilakukan di PT Tarindo yang berlokasi di Juwana, Kabupaten Pati. Waktu penelitian yaitu mulai dari 2 Januari 2024 sampai dengan 31 Januari 2024.

3.2 Teknik Pengumpulan Data

Untuk mendapatkan data terkait dengan objek yang diteliti, dilakukan pengumpulan data dengan teknik sebagai berikut.

a. Observasi

Pengumpulan data melalui pengamatan dan pencacatan terhadap permasalahan yang diselidiki pada objek penelitian dengan tujuan untuk mendapatkan data seperti aliran informasi dan data terkait penyimpanan barang serta alur produksi.

b. Wawancara

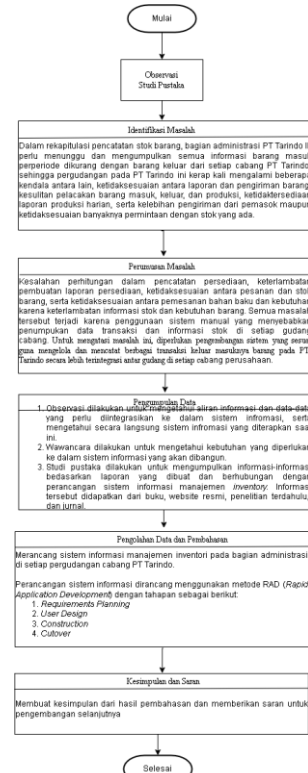
Pengumpulan data melalui tatap muka dan tanya jawab secara langsung dengan *stakeholder* dan berhubungan dengan penelitian untuk memperoleh informasi mengenai aliran informasi serta mengidentifikasi kebutuhan.

c. Studi Literatur

Studi literatur dilakukan untuk mempelajari teori dan konsep yang dapat digunakan sebagai tinjauan Pustaka dalam penelitian.

3.3 Flowchart Penelitian

Flowchart metodologi penelitian pada PT Tarindo dijelaskan dalam Gambar 3.



Gambar 3. Flowchart Penelitian

3.4 Perancangan Sistem

Dalam perancangan sistem pada penelitian ini, digunakan metode pengembangan RAD yang merupakan salah satu metode perancangan sistem maupun perangkat lunak atau biasa yang dikenal SDLC (*Software Development Life Cycle*). Mempertimbangkan ruang lingkup proyek yang terbatas dan kebutuhan akan kecepatan sistem, oleh karena itu metode *Rapid Application Development* dipilih sebagai pendekatan pengembangan yang cocok. RAD merupakan salah satu dari beberapa metode pengembangan perangkat lunak yang digunakan dalam *Agile Development*. Metode ini memungkinkan tim pengembang untuk membangun aplikasi dengan cepat dan efektif dengan menggunakan pendekatan yang lebih fleksibel. Dalam model pengembangan sistem menggunakan metode RAD ini terdapat empat tahapan yang perlu dilakukn sebagai berikut.

- a. Rencana Kebutuhan (*Requirements Planning*)
Pada tahapan ini pengguna dan penulis saling berdiskusi untuk meneliti dan memecahkan masalah yang sedang terjadi, serta mengidentifikasi dan menentukan kebutuhan apa saja dan fungsi utama apa yang diharapkan ada di sistem informasi manajemen *inventory* yang akan dibangun. Dikarenakan tahapan ini merupakan langkah awal keberhasilan pengembangan sistem informasi, maka komunikasi antara penulis dengan pengguna harus dilakukan dengan baik dan komunikatif.
- b. Desain Penggunaan (*User Design*)
Tahapan membuat rancangan yang akan diusulkan agar sesuai dengan kebutuhan, berjalan sesuai dengan rencana dan diharapkan dapat mengatasi masalah yang sedang terjadi. Pada penelitian ini, desain sistem yang digambarkan menggunakan *Tools Unified Modeling Language (UML)* ditujukan untuk menggambarkan arsitektur aplikasi *web* yang akan dikembangkan.
- c. *Construction*
Tahapan ini adalah tahapan dimulainya pembuatan sistem yang sudah direncanakan. Memulai menyusun suatu kode program atau bisa disebut dengan *coding*. Untuk mengubah desain sistem yang sebelumnya sudah dibuat menjadi sebuah aplikasi yang telah dirancang agar dapat digunakan. Dalam tahapan ini akan digunakan *HTML*, *CSS*, dan *Javascript* untuk pengembangan *Front-end*, lalu *Python*, *Framework Django* untuk *Back-End* dan *MySQL* sebagai *database*.
- d. *Cutover*
Tahapan ini adalah pengujian keseluruhan sistem yang sudah dibangun untuk memastikan sistem berfungsi dengan baik, sesuai dengan kebutuhan yang telah diidentifikasi, dan tidak memiliki *bug* atau kesalahan yang signifikan. Pengujian

dilakukan secara menyeluruh dengan *Black Box Testing* supaya dapat mengurangi resiko cacat sistem.

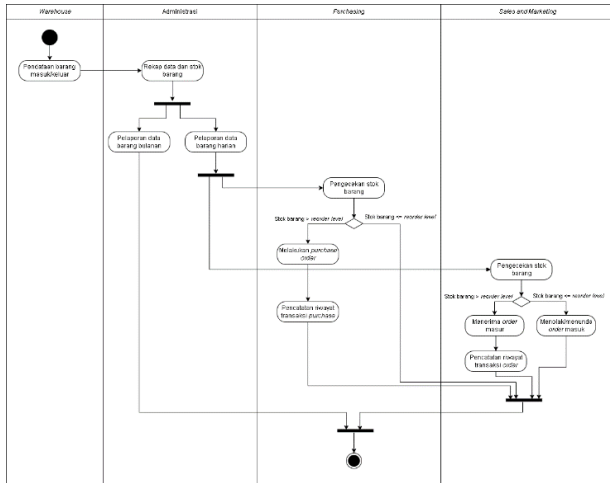
4. Hasil dan Pembahasan

4.1 Analisis Sistem Berjalan

Analisis sistem berjalan dilakukan oleh penulis untuk mengidentifikasi kelemahan sistem tersebut, baik dari cara kerja sistem maupun pihak pelaksananya dan segala sesuatu yang terlibat dalam sistem tersebut. Sistem manajemen *inventory* yang sedang berjalan saat ini masih menggunakan cara manual. Dalam rekapitulasi pencatatan stoknya, bagian administrasi PT Tarindo III perlu menunggu dan mengumpulkan semua informasi barang masuk perperiode dikurang dengan barang keluar dari setiap cabang PT Tarindo. Kesalahan perhitungan dalam pencatatan persediaan, keterlambatan pembuatan laporan persediaan, ketidaksesuaian antara pesanan dan stok barang, serta ketidaksesuaian antara pemesanan bahan baku dan kebutuhan barang, semuanya dipengaruhi oleh penggunaan sistem manual dalam manajemen *inventory* saat ini. Sistem manual menyebabkan penumpukan data transaksi dan keterlambatan informasi stok di setiap gudang cabang. Berikut proses pendataan barang yang masuk dan keluar yang berjalan pada PT Tarindo:

- 1) Barang yang akan masuk dan keluar diterima oleh staff gudang dari tiap cabang PT Tarindo.
- 2) Barang masuk akan dicatat dalam buku, yang dicatat adalah kode barang, nama barang, kategori barang, kode *supplier*, stok barang, *staff* yang menerima. Lalu disimpan sebagai stok barang digudang. Pencatatan ini dilakukan oleh masing-masing *staff* gudang cabang PT Tarindo.
- 3) Barang keluar akan dicatat dalam buku, dan yang dicatat adalah kode barang, nama barang, kategori barang, tanggal barang keluar, waktu barang keluar, jumlah barang yang diambil, *staff* yang bertanggung jawab saat pengambilan barang. Pencatatan ini dilakukan oleh masing-masing *staff* gudang cabang PT Tarindo.
- 4) Setelah selesai pencatatan di setiap gudang cabang PT Tarindo, maka *staff* divisi administrasi di cabang PT Tarindo III akan merekap data barang yang masuk dan barang yang keluar setiap harinya sebagai acuan melakukan pembelian bahan baku ke *supplier* oleh divisi *purchasing* ataupun penerimaan pesanan dari customer oleh divisi *sales and marketing*. Selain itu dibuat juga laporan rekap perbulan ditulis dengan *microsoft excel* lalu diprint dan dilaporkan sebagai laporan bulanan *inventory* barang di gudang.

Gambar 4. menunjukkan *activity diagram* yang menggambarkan proses bisnis sistem yang berjalan saat ini.



Gambar 4. Proses Bisnis Sistem Saat ini

4.2 Perancangan Perbaikan Sistem

Berdasarkan identifikasi masalah dari sistem yang sudah berjalan, selanjutnya dilakukan perbaikan sistem sesuai dengan metode yang dipilih yaitu *Rapid Application Development*.

4.2.1 Requirements Planning

Berdasarkan permasalahan yang ada, maka dibutuhkan sebuah sistem agar proses bisnis di PT Tarindo dapat berlangsung lebih efektif dan efisien. Kebutuhan sistem sendiri diperoleh dari wawancara yang dilakukan dengan pembimbing lapangan dan pegawai di lapangan.

4.2.1.1 Analisis Gap

Selanjutnya dilakukan analisis gap pada proses bisnis pada sistem lama dengan sistem baru. Analisis gap terjadi pada penerapan sistem informasi manajemen *inventory* dengan berbasis web di PT Tarindo pada tabel sebagai berikut:

Tabel 1. Analisis Gap Sistem

| No. | Kondisi saat ini | Kebutuhan | Full-filment | | | Solusi |
|-----|---|---|--------------|---|---|--|
| | | | N | P | F | |
| 1. | Pencatatan dan perhitungan persediaan masih menggunakan buku catatan. | Pencatatan persediaan perlu diotomatisasi dan diintegrasikan ke dalam sistem yang dapat menghitung secara otomatis dan bisa edit data | | ✓ | | Pembuatan sistem manajemen persediaan yang berbasis <i>web</i> . |

Tabel 1. Analisis Gap Sistem (Lanjutan)

| No. | Kondisi saat ini | Kebutuhan | Full-filment | | | Solusi |
|-----|---|---|--------------|---|---|--|
| | | | N | P | F | |
| 2. | Keterlambatan pembuatan laporan persediaan karena penumpukan data transaksi dan ketergantungan pada rekap gudang cabang. | barang yang tersedia. Diperlukan sistem yang dapat mengintegrasikan data transaksi secara <i>real-time</i> dari setiap gudang cabang dan menghasilkan laporan persediaan yang cepat dan akurat. | | ✓ | | Pengembangan sistem manajemen persediaan yang terhubung secara langsung dengan setiap gudang cabang. |
| 3. | Ketidaksesuaian antara pesanan dan stok barang disebabkan oleh keterlambatan informasi stok di setiap gudang. | Dibutuhkan sistem yang dapat memberikan informasi stok secara <i>real-time</i> untuk menghindari ketidaksesuaian antara pesanan dan stok barang. | | ✓ | | Implementasi sistem manajemen persediaan dengan pembaruan stok secara <i>real-time</i> . |
| 4. | Ketidaksesuaian antara pemesanan bahan baku dengan kebutuhan barang karena keterlambatan informasi kebutuhan barang di setiap gudang. | Diperlukan sistem yang dapat memberikan informasi mengenai kebutuhan bahan baku secara akurat dan memastikan informasi kebutuhan barang tersedia secara <i>real-time</i> . | | | | Penerapan sistem manajemen persediaan yang dapat memantau kebutuhan barang secara <i>real-time</i> . |

4.2.1.2 Analisis PIECES

Dalam tahap analisis ini penulis akan menganalisis performa atau kinerja, informasi, ekonomi, kontrol, efisiensi, dan layanan dari sistem lama dan dibandingkan dengan sistem baru yang akan dibuat. Berikut adalah hasil dari analisis PIECES yang disajikan dalam bentuk tabel:

Tabel 2. Analisis PIECES

| No | Analisis | Sistem Lama | Sistem Baru |
|----|---------------------------------------|---|--|
| 1. | <i>Performance</i> (Performa/kinerja) | Kinerja sistem terganggu oleh kesalahan perhitungan dalam pencatatan persediaan, keterlambatan pembuatan laporan, dan ketidaksesuaian antara pesanan dan stok barang. | Diusulkan untuk meningkatkan kinerja dengan otomatisasi persediaan, pembuatan laporan yang cepat dan akurat, serta integrasi real-time antara pesanan dan stok barang. |
| 2. | <i>Information</i> (Informasi) | Informasi stok dan transaksi terbatas dan tidak selalu akurat karena penggunaan sistem manual. | Menawarkan peningkatan dengan integrasi sistem yang dapat memberikan informasi stok yang real-time dan akurat. |
| 3. | <i>Economy</i> (Ekonomi) | Biaya operasional meningkat karena keterlambatan dan kesalahan dalam manajemen persediaan. | Diusulkan untuk mengurangi biaya operasional dengan menghilangkan kesalahan dan keterlambatan dalam proses manajemen persediaan. |
| 4. | <i>Control</i> (Kontrol) | Kontrol terhadap persediaan dan proses manajemen persediaan | Memperkuat kontrol dengan otomatisasi dan integrasi sistem yang |

Tabel 2. Analisis PIECES (Lanjutan)

| No | Analisis | Sistem Lama | Sistem Baru |
|----|-------------------------------|---|--|
| | | kurang efektif karena keterbatasan sistem manual. | memungkinkan pemantauan dan pengelolaan persediaan secara lebih efisien. |
| 5. | <i>Efficiency</i> (Efisiensi) | Efisiensi operasional terganggu oleh keterlambatan dan kesalahan dalam proses manajemen persediaan. | Meningkatkan efisiensi dengan otomatisasi proses yang mempercepat pencatatan, pembuatan laporan, dan pengelolaan persediaan. |
| 6. | <i>Service</i> (Layanan) | Pelayanan terhadap pelanggan terganggu oleh ketidaksesuaian antara pesanan dan stok barang. | Meningkatkan pelayanan pelanggan dengan memastikan ketersediaan stok yang akurat dan respons cepat terhadap pesanan. |

4.2.2 User Design

Pada tahap ini akan dilakukan proses desain dan melakukan perbaikan-perbaikan apabila masih terdapat ketidaksesuaian desain antara *User* dan perancang sistem. Desain sistem digambarkan menggunakan *Tools Unified Modeling Language* (UML) berupa untuk menggambarkan arsitektur aplikasi web yang akan dikembangkan.

4.2.2.1 Use case diagram

Use case diagram merupakan diagram yang menjelaskan aktivitas apa saja yang dilakukan sistem yang akan dibangun dan apa saja yang berinteraksi dengan sistem tersebut. Dalam merancang *Use case diagram* dibutuhkan beberapa tahap antara lain identifikasi aktor, identifikasi use case, dan pemodelan *use case*.

A. Identifikasi Aktor

Identifikasi aktor akan menjelaskan daftar pelaku atau aktor yang terlibat dalam sistem informasi penjualan beserta deskripsi tentang masing-masing aktor yang ada. Berikut ini adalah aktor aktor yang

terlibat dalam sistem *inventory* barang pada PT Tarindo.

Tabel 3. Identifikasi Aktor

| ID | Aktor | Deskripsi |
|-------|----------------------------------|--|
| AC-01 | <i>Staff Warehouse</i> (gudang) | Aktor yang dapat mengelola data barang, mengelola data barang masuk, mengelola data barang keluar, dan mengelola data persediaan barang. |
| AC-02 | <i>Staff Administrasi</i> | Aktor yang dapat menambah, menghapus, dan meng-update seluruh data sistem. |
| AC-03 | <i>Staff Purchasing</i> | Aktor yang dapat mengelola barang masuk. |
| AC-04 | <i>Staff Sales and Marketing</i> | Aktor yang dapat mengelola barang keluar. |

B. Identifikasi Use Case

Setelah melakukan identifikasi aktor, tahapan selanjutnya adalah melakukan identifikasi *Use case* yang menjelaskan hubungan antara suatu aktivitas dengan aktor yang terlibat.

Tabel 4. Identifikasi Use Case

| ID | Use case | Deskripsi | Aktor |
|--------|-------------------------|--|--|
| UC-01 | <i>Login</i> | <i>Use case</i> ini menggambarkan kegiatan memasukkan data <i>id User</i> dan <i>password</i> yang sudah ada dalam <i>database server</i> untuk dapat masuk ke dalam sistem. | 1. <i>Staff Administrasi</i> 2. <i>Staff Warehouse</i> 3. <i>Staff Purchasing</i> 4. <i>Staff Sales and Marketing</i> |
| UC-02* | <i>Reset Password</i> | <i>Use case</i> ini menggambarkan kegiatan mereset <i>password</i> lama menjadi <i>password</i> baru. | 1. <i>Staff Administrasi</i> 2. <i>Staff Warehouse</i> 3. <i>Staff Purchasing</i> 4. <i>Staff Sales and Marketing</i> |
| UC-03 | <i>Manajemen Barang</i> | <i>Use case</i> ini menggambarkan kegiatan melihat data barang, menambahkan data barang, | 1. <i>Staff Administrasi</i> |

Tabel 4. Identifikasi Use Case (Lanjutan)

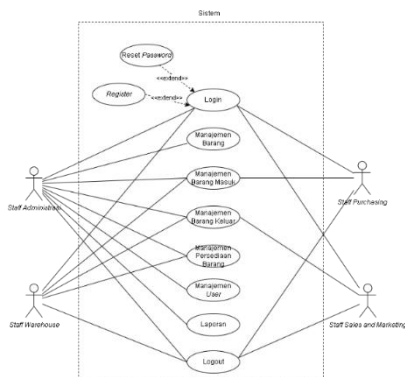
| ID | Use case | Deskripsi | Aktor |
|-------|------------------------------------|--|---|
| UC-04 | <i>Manajemen Barang Masuk</i> | mengubah data barang dan menghapus data. <i>Use case</i> ini menggambarkan kegiatan melihat data barang masuk, menambahkan data barang masuk, mengubah dan menghapus data barang masuk. | 1. <i>Staff Administrasi</i> 2. <i>Staff Warehouse</i> 3. <i>Staff Purchasing</i> |
| UC-05 | <i>Manajemen Barang Keluar</i> | <i>Use case</i> ini menggambarkan kegiatan melihat data barang keluar, menambahkan data barang keluar, mengubah dan menghapus data barang keluar. | 1. <i>Staff Administrasi</i> 2. <i>Staff Warehouse Sales and Marketing</i> |
| UC-06 | <i>Manajemen Persediaan Barang</i> | <i>Use case</i> ini menggambarkan kegiatan melihat data persediaan barang, menambahkan data persediaan barang, mengubah data persediaan barang dan menghapus data persediaan barang. | 1. <i>Staff Administrasi</i> 2. <i>Staff Warehouse</i> |
| UC-07 | <i>Manajemen User</i> | <i>Use case</i> ini menggambarkan kegiatan melihat data <i>User</i> , menambahkan data <i>User</i> , mengubah data <i>User</i> dan menghapus data <i>User</i> . | 1. <i>Staff Administrasi</i> |
| UC-08 | <i>Laporan</i> | <i>Use case</i> ini menggambarkan | 1. <i>Staff Administrasi</i> |

Tabel 4. Identifikasi Use case (Lanjutan)

| ID | Use case | Deskripsi | Aktor |
|-------|----------|--|-----------------------|
| UC-09 | Logout | kan kegiatan melihat dan mencetak laporan barang masuk, laporan barang keluar, laporan daftar barang, laporan persediaan barang, dan laporan permintaan barang. Use case ini menggambar-kan proses keluar dari sistem | 1. Staff Administrasi |

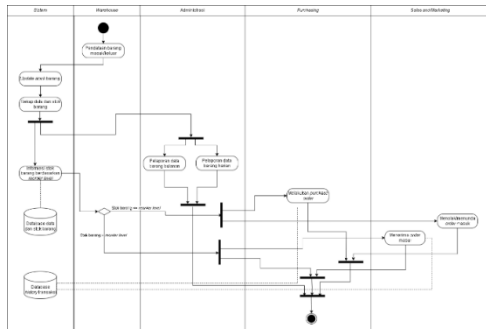
C. Pemodelan Use case diagram

Setelah melakukan analisis untuk mengidentifikasi siapa saja aktor yang terkait dan Use case yang ada selanjutnya akan dilakukan pemodelan Use case diagram.



Gambar 5. Use case diagram Sistem Informasi 4.2.2.2 Proses Bisnis Rancangan Sistem

Berdasarkan Use case yang telah ditetapkan sebelumnya, dilakukan pemodelan langkah-langkah proses atau aktivitas dari Sistem Manajemen Inventory pada PT Tarindo.



Gambar 6. Proses Bisnis Rancangan Sistem

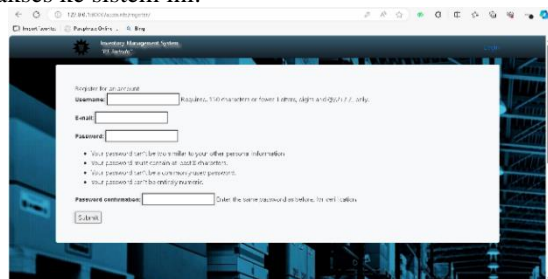
Berdasarkan gambar 6., dapat dilihat bahwa terdapat beberapa perbedaan pada alur atau proses bisnis antara sistem yang berjalan dengan rancangan sistem yang baru. Pada proses bisnis sistem berjalan, divisi administrasi harus melakukan rekap terlebih dahulu secara manual kemudian membuat laporan stok barang ketika terjadi pendataan barang dari *staff warehouse*. Divisi *purchasing* dan divisi *sales marketing* baru bisa melaksanakan tugasnya untuk melakukan pengecekan stok barang dengan *reorder level* setelah divisi administrasi menyelesaikan laporan stok barangnya. Selanjutnya dari hasil pengecekan stok barang dilakukan pengambilan keputusan mengenai *purchasing* dan order. Sedangkan untuk proses bisnis rancangan sistem yang baru, ketika *staff warehouse* melakukan pendataan barang masuk/keluar sistem akan otomatis melakukan rekap pendataan dan melakukan pengecekan stok berdasarkan *reorder level*. Informasi yang didapatkan akan disimpan di *database* sistem. Pada rancangan sistem yang baru ini setiap perubahan data maupun stok barang bekerja secara terintegrasi dan *real-time*, sehingga pada divisi administrasi tidak perlu melakukan rekap manual terlebih dahulu untuk membuat laporan bulanan maupun harian serta divisi *purchasing* dan *sales marketing* bisa langsung mendapatkan informasi yang dibutuhkan dari sistem tanpa perlu menunggu hasil laporan dari divisi administrasi dahulu, yang mana dapat membuat pengambilan keputusan di dua divisi ini bisa dilakukan lebih cepat dan akurat.

4.2.3 Contruction

Pada tahap ini Aplikasi Sistem Informasi Manajemen Inventori berbasis *Web* yang telah dibangun terdiri dari halaman administrator (*backend*) dan halaman pengguna (*frontend*) yang dapat diakses oleh *User* dengan ketentuan tertentu.

1. Form Register

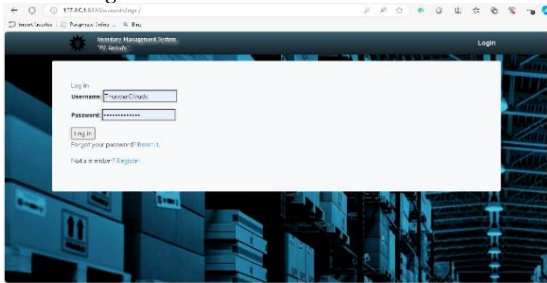
Form register adalah akses masuk paling awal bagi *User* yang baru pertama kali akan mengakses sistem informasi manajemen *inventory* PT Tarindo ini. Pada bagian ini *User* diminta untuk meng-*input Username, email, password, dan password confirmation*. Setelah memasukkan semua data yang diminta *User* masih harus menunggu konfirmasi dari admin agar akunnya tersebut diaktivasi sebagai bagian dari *staff* PT Tarindo yang berhak memiliki akses ke sistem ini.



Gambar 7. Form Register

2. Form Login

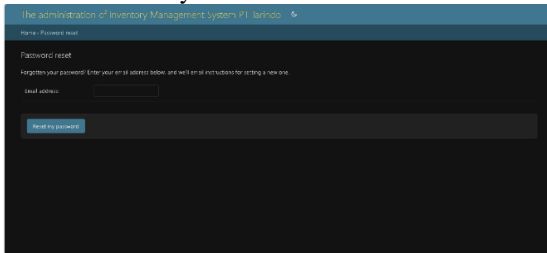
Form login merupakan akses masuk ke dalam sistem yang tampil pada saat awal program digunakan. *User* diminta memasukkan *Username* dan *password* yang sebelumnya sudah dibuat pada tahap *register*. Hanya *User* yang sudah diaktivasi yang bisa berhasil *login* ke sistem ini.



Gambar 8. Form Login

3. Form Restart Password

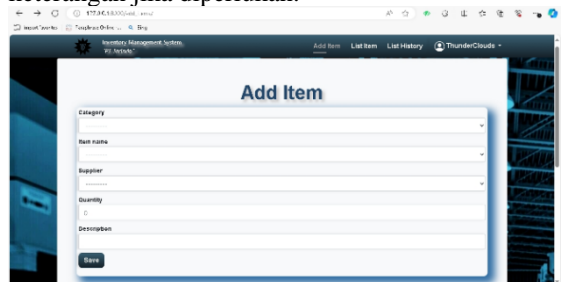
Form restart password digunakan ketika *User* mengalami kendala lupa terhadap *password* yang telah dibuat sebelumnya. Pada *form* ini *User* akan diminta membuat *password* baru dan melakukan aktivasi ulang. Untuk dapat melakukan *restart password*, *User* harus meng-input *email* yang sudah terdaftar sebelumnya.



Gambar 9. Form Restart Password

4. Form Add Items

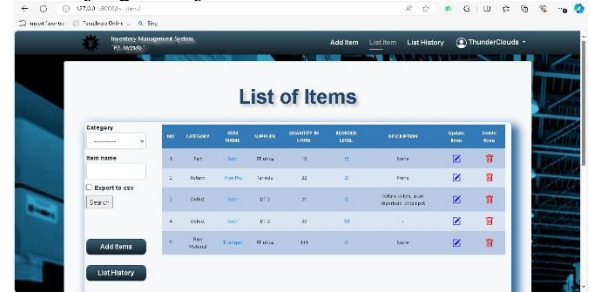
Form add items digunakan untuk membuat *list* stok pada suatu *item* yang sudah ada. Pada bagian ini *User* hanya bisa secara bebas mengisi *quantity* dan keterangan dari suatu *item* yang sudah ada. Penambahan *item* baru hanya bisa dilakukan oleh *User* administrasi. Semisal PT Tarindo hanya memiliki *item* berupa gotri dan kran, maka *User* selain administrasi hanya bisa mengatur jumlah stok *item* tersebut saat ini dengan jenis kategori yang sudah ada (*return, raw material, defect, part, finished good*) dan keterangan jika diperlukan.



Gambar 10. Form Add Item

5. List of Items

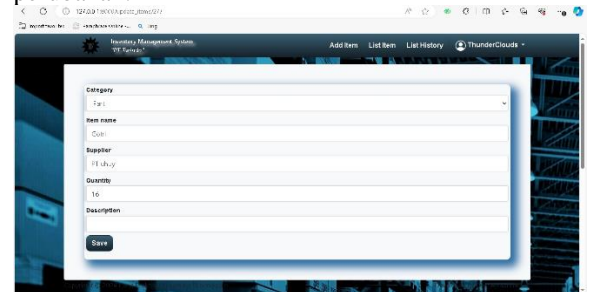
List of items menampilkan *list item* atau produk hasil dari pengisian *form add item* sebelumnya. Pada bagian ini terdapat fitur *search* untuk membantu mencari *item* tertentu dengan lebih cepat terutama ketika *list* sudah terlalu banyak. Kemudian ada bagian *export to csv* yang digunakan ketika *User* membutuhkan data *item* tertentu dalam format file yaitu *csv*. Terdapat juga fitur *update* dan *delete* untuk menunjang manajemen *list item* tersebut.



Gambar 11. List Items

6. Form Update

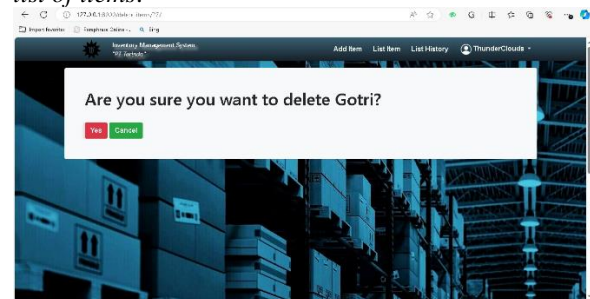
Form update digunakan untuk meng-*update* informasi pada suatu *item* apabila diperlukan perubahan.



Gambar 12. Form Update

7. Delete

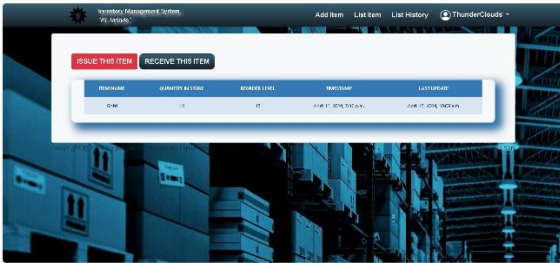
Bagian ini digunakan sebagai konfirmasi lanjutan ketika *User* mengakses fitur *delete* yang terdapat pada *list of items*.



Gambar 13. Form Delete

8. Stock Detail

Bagian ini menunjukkan detail dari suatu *item*. Detail yang dimaksud di sini adalah kapan pertama kali *item* tersebut di-*input* dan kapan terakhir kali di *update*. Terdapat fitur *issue* dan *receive* untuk memajemen stok *item* tersebut.



Gambar 14. Stock Detail

9. Form Issue

Form issue digunakan untuk melakukan perubahan stok ketika pada suatu item akibat adanya sejumlah barang/item tersebut yang keluar sehingga stok berkurang.



Gambar 15. Form Issue

10. Form Receive

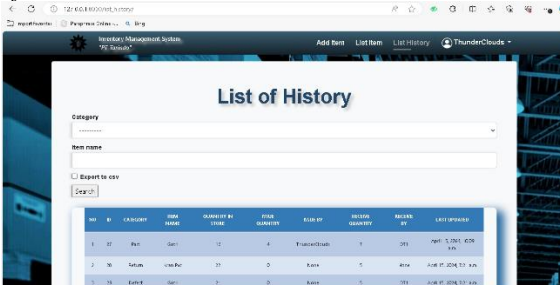
Form receive digunakan untuk melakukan perubahan stok ketika pada suatu item akibat adanya sejumlah barang/item tersebut yang masuk sehingga terdapat penambahan jumlah stok item tersebut.



Gambar 16. Form Receive

11. List History

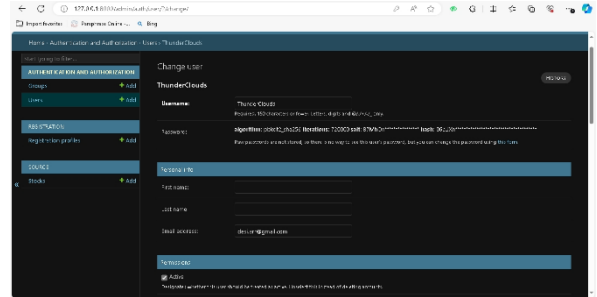
List history memuat semua informasi perubahan pada item-item yang terdapat pada list of items. Pada bagian ini terdapat fitur search dan export to csv yang fungsinya sama seperti fitur yang terdapat bagian list of items.



Gambar 17. List History

12. Manage User

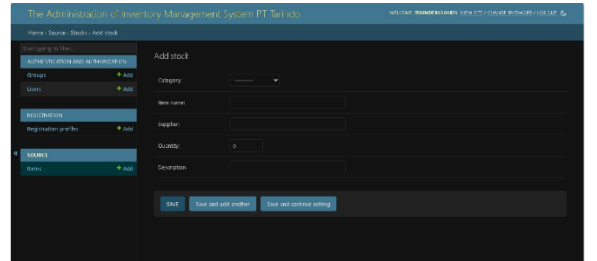
Manage User digunakan untuk manajemen User yang teraktivasi serta User yang perlu diaktivasi di sistem ini. Pada bagian ini dapat diatur akses apa saja yang bisa dilakukan oleh tiap User dan konfigurasi lainnya. Manage User hanya bisa diakses oleh User administrasi.



Gambar 18. Manage User

13. Manage Item

Manage Item digunakan untuk manajemen item-item yang sudah atau perlu ada di sistem ini. Pada bagian ini bisa menambahkan barang atau item baru yang sebelumnya tidak dapat dilakukan di bagian form add item jika diperlukan. Manage item hanya bisa diakses oleh User administrasi.



Gambar 19. Manage Item

4.2.4 Cutover

Setelah sistem selesai dibangun, dilakukan integrasi dan system testing untuk memastikan sistem tidak error ketika sudah diterima oleh klien. Pengujian dilakukan dengan Black Box Testing. Black Box Testing merupakan salah satu pengujian yang berfokus pada bagian fungsional sistem dan tujuannya untuk memverifikasi apakah program sudah dibuat sesuai dengan kebutuhan pengguna. Berikut merupakan tabel hasil pengujian sistem menggunakan Black Box Testing.

Tabel 5. Pengujian Black Box Testing

| No | Fungsional | Skenario | Hasil | Status |
|----|------------|---|---|--------|
| 1. | Login | Pengguna meng-inputkan Username dan password dengan benar | Pengguna berhasil masuk ke dalam sistem | Sesuai |
| 2. | Login | Pengguna meng- | Pengguna gagal | Sesuai |

Tabel 5. Pengujian *Black Box Testing* (Lanjutan)

| No | Fungsional | Skenario | Hasil | Status |
|----|----------------------------------|---|--|--------|
| 3. | Registrasi | inputkan <i>Username</i> dan <i>password</i> dengan salah Pegguna memasukkan <i>Username, email,</i> dan <i>password</i> sesuai ketentuan | masuk ke dalam sistem Pegguna berhasil registasi | Sesuai |
| 4. | Regristasi | Pegguna memasukkan <i>Username, email,</i> dan <i>password</i> tidak sesuai ketentuan | Pegguna gagal registasi | Sesuai |
| 5. | Menambah stok <i>item</i> baru | Pegguna masuk ke halaman <i>add item,</i> kemudian mengisi <i>form</i> yang ada kemudian klik <i>save.</i> | Sistem menyimpan <i>item</i> ke <i>database</i> produk | Sesuai |
| 6. | Menampilkan <i>list of items</i> | Pegguna masuk ke halaman <i>list of items,</i> sistem akan menampilkan <i>list item</i> yang sebelumnya disimpan melalui halaman <i>add items</i> | Sistem menampilkan halaman <i>list item</i> | Sesuai |
| 7. | <i>Update item</i> | Pegguna memilih fitur | Sistem menampilkan | Sesuai |

Tabel 5. Pengujian *Black Box Testing* (Lanjutan)

| No | Fungsional | Skenario | Hasil | Status |
|----|--------------------|---|--|--------|
| | | | <i>update</i> halaman pada <i>update item</i> dan <i>list of item,</i> semua perubahan yang dilakukan di <i>form</i> yang terdapat di halaman tersebut tersimpan di <i>database.</i> | |
| 8. | <i>Delete item</i> | Pegguna memilih fitur <i>delete</i> pada halaman <i>list of items,</i> sistem akan memasuki halaman <i>delete item</i> dan kemudian klik <i>button yes.</i> | <i>Item</i> terpilih akan terhapus dari <i>list of items</i> dan <i>database</i> sistem. | Sesuai |
| 9. | <i>Delete item</i> | Pegguna memilih fitur <i>delete</i> pada halaman <i>list of items,</i> sistem akan memasuki halaman <i>delete item</i> dan kemudian klik | Sistem akan kembali ke halaman <i>list of item</i> dan terpilih tidak jadi terhapus dari <i>list of items</i> dan <i>database</i> sistem. | Sesuai |

Tabel 5. Pengujian *Black Box Testing* (Lanjutan)

| No | Fungsional | Skenario | Hasil | Status |
|-----|------------------------|---|--|--------|
| 10. | <i>List history</i> | <i>button cancel.</i> Pengguna masuk ke sistem halaman <i>list history</i> , akan menampilkan <i>list item</i> yang sebelumnya disimpan melalui halaman <i>add items</i> dan semua riwayat perubahan yang pernah dilakukan | Sistem menampilkan halaman <i>list history</i> | Sesuai |
| 11. | <i>Detail stok</i> | Pengguna masuk ke halaman <i>detail stok</i> melalui halaman <i>list of items</i> dengan meng-klik salah satu <i>item</i> yang diinginkan. | Sistem menampilkan halaman <i>detail stok</i> dan menampilkan detail dari <i>item</i> terpilih | Sesuai |
| 12. | <i>Receive of item</i> | Pengguna memilih fitur <i>receive of item</i> pada halaman <i>detail stok</i> , sistem akan memasuki halaman <i>receive of</i> | Sistem menampilkan halaman <i>receive of item</i> dan semua perubahan yang dilakukan di <i>form</i> yang terdapat di halaman | Sesuai |

Tabel 5. Pengujian *Black Box Testing* (Lanjutan)

| No | Fungsional | Skenario | Hasil | Status |
|-----|--|--|---|--------|
| 13. | Melakukan pencarian pada <i>list of history</i> | <i>item</i> yang tersebut berisi <i>form</i> yang dapat diedit dan kemudian klik <i>save</i> . Pengguna mengakses fitur <i>search</i> pada halaman <i>list history</i> dengan menggunakan <i>input category</i> atau <i>item name</i> sebagai kriteria pencarian , kemudian klik <i>button search</i> . | Sistem akan menampilkan <i>list history</i> dengan kriteria yang sudah ditentukan sebelumnya. | Sesuai |
| 14. | Menyimpan dan menunduh <i>list item</i> yang diinginkan ke dalam bentuk file <i>csv</i> | Pengguna masuk ke halaman <i>list of items</i> dan mengakses fitur <i>export to csv</i> . | Sistem akan mengunduh <i>list item</i> yang diinginkan ke dalam bentuk file <i>csv</i> | Sesuai |
| 15. | Menyimpan dan menunduh <i>list history</i> yang diinginkan ke dalam bentuk file <i>csv</i> | Pengguna masuk ke halaman <i>list of history</i> dan mengakses fitur <i>export to csv</i> . | Sistem akan mengunduh <i>list history</i> yang diinginkan ke dalam bentuk file <i>csv</i> | Sesuai |
| 16. | <i>Logout</i> | Pengguna mengakses fitur <i>logout</i> melalui <i>button</i> | Pengguna berhasil keluar dari sistem | Sesuai |

Tabel 5. Pengujian Black Box Testing (Lanjutan)

| No | Fungsional | Skenario | Hasil | Status |
|----|------------|-------------------|-------|--------|
| | | logout di navbar. | | |

5. Kesimpulan

Berdasarkan hasil pengamatan dan analisis setelah dilakukan perbaikan sistem di PT Tarindo, dapat disimpulkan bahwa implementasi sistem informasi telah memberikan dampak positif yang signifikan bagi perusahaan. Pertama, sistem informasi baru memungkinkan pembuatan laporan persediaan barang secara cepat dan efisien, yang mengoptimalkan proses administrasi di PT Tarindo. Kedua, pengembangan sistem informasi yang menyajikan informasi stok barang secara real-time di setiap gudang berhasil mengurangi kesenjangan antara pesanan dan ketersediaan barang, meningkatkan akurasi pengelolaan persediaan, serta meningkatkan efisiensi operasional secara keseluruhan. Selanjutnya, aplikasi monitoring kebutuhan bahan baku yang tepat waktu telah berhasil mengurangi ketidaksesuaian antara pemesanan bahan baku kepada supplier dan kebutuhan sebenarnya di gudang-gudang PT Tarindo, serta meningkatkan efisiensi dalam rantai pasokan perusahaan. Terakhir, integrasi sistem informasi dengan database telah mengurangi risiko kehilangan data dan mempermudah pengoperasian, yang pada akhirnya mengoptimalkan pengelolaan persediaan barang secara menyeluruh di PT Tarindo. Dengan demikian, implementasi perbaikan sistem informasi di PT Tarindo memberikan kontribusi positif dalam meningkatkan efisiensi dan produktivitas perusahaan.

Daftar Pustaka

- [1] S. Kurniawan, Rancang Bangun Sistem Inventory Pada Pt. Maruhide Indonesia Berbasis Dekstop. Semnas Ristek (Seminar Nasional Riset ..., 317–322., s.l.: s.n., 2020.
- [2] S. Monalisa, P. Putra and F. Kurnia, Rancang Bangun Sistem Informasi Inventory Obat Pada Rumah Sakit Jiwa Tampan Berbasis Web,” Query J. Inf. Syst., vol. 5341, no. October, pp. 58–65, s.l.: s.n., 2018.
- [3] A. M. B. and F. A. J., Perancangan Sistem Inventory dengan Metode Rapid Application Development (Studi Kasus PT XYZ) , s.l.: s.n., 2019.
- [4] B. Rudianto and Y. E. Achyani, Penerapan Metode Rapid Application Development pada Sistem Informasi Persediaan Barang berbasis Web, s.l.: s.n., 2020.
- [5] A. Dennis, B. H. Wixom and R. M. Roth, System Analysis & Design, 5th ed, s.l.: s.n., 2014.
- [6] S. Aswati and Y. Siagian, “Model Rapid Application Development,” Sesindo, pp. 317–324, s.l.: s.n., 2016.
- [7] G. Oktaviani, Pengantar Sistem Informasi, 2019.
- [8] Grosskopf, Decker and Weske, The Process: Business Process Modeling using BPMN. Meghan Kiffer Press, 2009.
- [9] S. System, THE BUSINESS PROCESS MODE, 2004.
- [10] T. S. Syamfithriani, M. A. and M. D. E. Putra, SUPPLY CHAIN MANAGEMENT (Studi Kasus : PT . Aretha Nusantara Farm). Jurnal Cloud Information, 1, 42–49, 2016.
- [11] Y. W. K. Kustanto, Kata Kunci: Duta.com, 2(April), 78–91, 2015.
- [12] B. Suyono, W. Agustin and Y. Efendi, PENGELOLAAN SISTEM INVENTORI PADA PT. INDOMARCO PRISTAMA MENGGUNAKAN ANALISIS ECONOMIC ORDER QUANTITY, 2018.
- [13] P. S. Wardhani, Perencanaan dan Pengendalian Persediaan Komponen dengan menggunakan Metode EOQ. Media Mahardika, 13(3), 310–328, 2015.
- [14] M. F. Putra, E. Suhendar and R. Usman, GAP ANALYSIS PERSIAPAN IMPLEMENTASI ISO 9001:2015 PADA PT. TRANS POWER MARINE, TBK, 2019.
- [15] W. Ragil, Pedoman Sosialisai Pedoman Standar. Jakarta: Mitra Wacana Media., 2010.
- [16] H. S. Alvin, Rancang Bangun Sistem Informasi Inventory Barang Menggunakan Metode FIFO (First In-First Out) Berbasis Web Pada PT Cipta Rasa Multindo, 2020.
- [17] J. W. Satzinger, R. B. Jackson and S. D. Burd, Systems Analysis and Design in a Changing World, Sixth Edition. Boston: CENGAGE Learning, 2012.
- [18] N. Alkaf, B. Saputra and H. S. Purba, Rancangan Sistem Manajemen Skripsi Berbasis Web Menggunakan Metode Rapid Application Development (RAD). 9(5), 1621–1631, 2022.
- [19] S. Al-Saqqa, S. Sawalha and H. Abdelnabi, Agile software development: Methodologies and trends. International Journal of Interactive Mobile Technologies, 14(11), 246–270, 2020.
- [20] R. Parlita, T. A. N. S. M. Nisaa and Haque, Studi Literatur Kekurangan dan Kelebihan Pengujian Black Box. Teknomatika, 10(02), 131–140, 2020.
- [21] T. S. Jaya, Pengujian Aplikasi dengan Metode Blackbox Testing Boundary Value Analysis. Jurnal Informatika Pengembangan IT (JPIT), 3(2), 45–46, 2018.

- [22] D. D. Prasetya, Modul Ajar Praktikum Pemrograman Web, 2016.
- [23] A. Ma'arif, BUKU AJAR PEMROGRAMAN LANJUT BAHASA PEMROGRAMAN PYTHON, 2020.
- [24] A. Nurmalasari, PENGGUNAAN FRAMEWORK DJANGO DAN MYSQL PADA APLIKASI SISTEM AKADEMIK TK INSAN KAMI, 2019.