

**PENGEMBANGAN SISTEM *INVENTORY*
SPARE PART DENGAN INTEGRASI *UPLOAD*
EXCEL DAN PENCATATAN RIWAYAT
BARANG**

TUGAS AKHIR

Disusun Oleh:

Bagus Dwi Putra

3312311132

Disusun untuk memenuhi salah satu syarat memperoleh gelar Ahli Madya
Teknik Informatika



**PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNIK INFORMATIKA
POLITEKNIK NEGERI BATAM
2026**

KATA PENGANTAR

Bismillahirrahmanirrahim.

Dengan puji syukur, penulis panjatkan ke hadirat Allah Yang Maha Esa, atas segala rahmat, karunia, dan bimbingannya sehingga laporan pengembangan dengan judul “Pengembangan Sistem *Inventory Spare Part* Dengan Integrasi *Upload Excel* dan Pencatatan Riwayat Barang” ini dapat diselesaikan dengan baik sesuai waktu yang telah ditentukan.

Penulis menyampaikan terima kasih sebesar-besarnya kepada berbagai pihak yang telah membantu dalam mensukseskan proses penyusunan laporan ini, terutama kepada:

1. Kedua orang tua penulis yang telah memberi semangat secara moral dan material selama kuliah ini berjalan.
2. Bapak Cahya Miranto sebagai dosen pembimbing yang telah memberikan arahan, bimbingan, serta masukan berharga selama penelitian dan pengembangan berlangsung.
3. Para dosen penguji yang telah memberikan saran konstruktif dan membangun untuk menyempurnakan laporan ini.
4. Rekan-rekan tim dan sahabat yang turut membantu, baik dalam pengumpulan data ataupun dalam diskusi dan motivasi.
5. Pihak Politeknik Negeri Batam yang telah memberikan izin, fasilitas, dan dukungan selama pelaksanaan penelitian.

Laporan ini ditulis dengan tujuan untuk memenuhi salah satu syarat akademik sekaligus sebagai dokumentasi atas pengembangan yang telah dilakukan. Penulis berharap laporan ini memberi manfaat, baik sebagai referensi bagi pembaca maupun sebagai bahan kajian untuk pengembangan penelitian selanjutnya. Penulis juga menyadari bahwa laporan ini masih jauh dari sempurna. Oleh karena itu, penulis dengan lapang dada menerima segala kritik dan saran yang membangun demi penyempurnaan laporan ini di masa mendatang.

Batam, 08 Juni 2026

Penulis

DAFTAR ISI

KATA PENGANTAR	ii
DAFTAR ISI.....	iii
DAFTAR TABEL.....	v
DAFTAR GAMBAR	vi
DAFTAR LAMPIRAN.....	viii
ABSTRACT.....	ix
BAB I PENDAHULUAN	10
1.1. Latar Belakang	10
1.2. Rumusan Masalah	11
1.3. Tujuan	12
1.4. Batasan Masalah.....	12
1.5. Manfaat	13
BAB II TINJAUAN PUSTAKA.....	14
2.1. Penelitian Terkait	14
2.2. Landasan Teori	18
2.3. Metode <i>Waterfall</i>	19
2.3.1. Analisis Kebutuhan	19
2.3.2. Desain Sistem.....	19
2.3.3. Implementasi	20
2.3.4. Pengujian.....	20
2.3.5. Pemeliharaan	20
2.4. Laravel.....	21
2.5. Basis data MySQL.....	21
BAB III ANALISIS DAN PERANCANGAN	22
3.1. Metode Pengembangan Produk.....	22
3.2. Analisis Kebutuhan	24
3.2.1. Proses Bisnis Berjalan.....	24
3.2.2. Gambaran Umum Sistem	25
3.3. Perancangan.....	26
3.3.1. Kebutuhan Fungsional	27
3.3.2. Kebutuhan Non-Fungsional	28
3.3.3. Diagram <i>Use case</i>	29

3.3.4.	<i>Skenario Use case</i>	29
3.3.5.	Activity diagram.....	39
3.3.6.	ER Diagram.....	40
3.3.7.	Perancangan Antarmuka (Wireframe)	40
3.4.	Rancangan Pengujian	48
3.4.1.	Tujuan Pengujian	48
3.4.2.	Alur Pengujian	48
3.4.3.	Responden Pengujian	49
3.4.4.	Metode Pengujian.....	49
BAB IV IMPLEMENTASI DAN PENGUJIAN.....		51
4.1	Hasil Implementasi.....	51
4.1.1.	Implementasi Teknologi Sistem.....	51
4.1.2.	Implementasi Halaman Login	52
4.1.3.	Implementasi <i>Dashboard</i> Sistem Sebagai <i>Admin</i>	52
4.1.4.	Implementasi Manajemen Data User	53
4.1.5.	Implementasi Manajemen Data Penjelasan Mesin	54
4.1.6.	Implementasi Manajemen Data Pengelolaan Data	55
4.1.7.	Implementasi Transaksi Barang Masuk dan Keluar	57
4.1.8.	Implementasi Transaksi Barang Masuk	58
4.1.9.	Implementasi Transaksi Barang Keluar	59
4.1.10.	Implementasi Validasi Stok Tidak Mencukupi di Transaksi	60
4.1.11.	Implementasi History Pencatatan Barang	62
4.1.12.	Implementasi Modul Laporan <i>PDF</i> dan Excel.....	63
4.1.13.	Implementasi <i>Dashboard</i> Sistem Sebagai <i>Staff</i>	64
4.2	Pengujian.....	65
4.2.1.	Pengujian <i>Black Box Testing</i>	66
4.2.2.	Pengujian <i>Unit Testing</i>	70
4.2.3.	Hasil Evaluasi Pengujian.....	78
BAB V KESIMPULAN DAN SARAN.....		80
5.1	Kesimpulan	80
5.2	Saran.....	81
DAFTAR PUSTAKA		82
DAFTAR LAMPIRAN.....		85

DAFTAR TABEL

Tabel 1. Perbandingan Penelitian Sebelumnya.....	14
Tabel 2. Kebutuhan Fungsional	27
Tabel 3. Kebutuhan Non-Fungsional	28
Tabel 4. <i>Use case Skenario</i> FR001	29
Tabel 5. <i>Use case Skenario</i> FR002	30
Tabel 6. <i>Use case Skenario</i> FR003	30
Tabel 7. <i>Use case Skenario</i> FR004	31
Tabel 8. <i>Use case Skenario</i> FR005	32
Tabel 9. <i>Use case Skenario</i> FR006	32
Tabel 10. <i>Use case Skenario</i> FR007	33
Tabel 11. <i>Use case Skenario</i> FR008	33
Tabel 12. <i>Use case Skenario</i> FR009	34
Tabel 13. <i>Use case Skenario</i> FR010	34
Tabel 14. <i>Use case Skenario</i> FR011	35
Tabel 15. <i>Use case Skenario</i> FR012	35
Tabel 16. <i>Use case Skenario</i> FR013	36
Tabel 17. <i>Use case Skenario</i> FR014	36
Tabel 18. <i>Use case Skenario</i> FR015	37
Tabel 19. <i>Use case Skenario</i> FR016	37
Tabel 20. <i>Use case Skenario</i> FR017	38
Tabel 21. <i>Use case Skenario</i> FR018	38
Tabel 22. Responden Pengujian.....	49

DAFTAR GAMBAR

Gambar 1. Metode <i>Waterfall</i>	22
Gambar 2. Proses Bisnis Berjalan Saat Ini.....	25
Gambar 3. Arsitektur Gambaran Umum Sistem	25
Gambar 4. <i>Use case</i> Diagram.....	29
Gambar 5. Flowchart Activity Diagram.....	39
Gambar 6. Entity Relation Diagram.....	40
Gambar 7. <i>Login</i> Page.....	41
Gambar 8. <i>Dashboard</i> Page	41
Gambar 9. Vitrox Machines <i>Dashboard</i> Page	42
Gambar 10. Vitrox Machines Display Page.....	42
Gambar 11. AOI Page	43
Gambar 12. AXI Page	43
Gambar 13. AOI Opening <i>Excel</i>	44
Gambar 14. AXI Opening <i>Excel</i>	44
Gambar 15. Basis data <i>Tally</i>	45
Gambar 16. Basis data Not <i>Tally</i>	45
Gambar 17. <i>Excel</i> Failed to <i>Upload</i>	46
Gambar 18. Delete All Data.....	46
Gambar 19. ICT Machine <i>Dashboard</i>	47
Gambar 20. ICT <i>Spare Part</i> Page	47
Gambar 21 - Implementasi Login	52
Gambar 22 - Implementasi Dashboard Setelah Login	52
Gambar 23 - Implementasi User Management	53
Gambar 24 - Implementasi Halaman Informasi Vitrox	54
Gambar 25 - Implementasi Halaman Mesin dari Vitrox.....	55
Gambar 26 - Implementasi Managemen Data 1	55
Gambar 27 - Implementasi Managemen Data 2	56
Gambar 29 - Implementasi Managemen Data 3	56
Gambar 30 - Implementasi Halaman Transaksi.....	57
Gambar 31 - Implementasi Transaksi Barang Masuk 1	58

Gambar 32 - Implementasi Transaksi Barang Masuk 2.....	58
Gambar 33 - Implementasi Transaksi Barang Keluar 1	59
Gambar 34 - Implementasi Transaksi Barang Keluar 2.....	59
Gambar 35 - Implementasi Validasi Stok di Transaksi 1	60
Gambar 36 - Implementasi Validasi Stok di Transaksi 2	61
Gambar 37 - Implementasi Halaman History Pencatatan Barang	62
Gambar 38 - Implementasi Ekspor PDF	63
Gambar 39 - Implementasi Ekspor Excel	63
Gambar 40 - Implementasi Untuk User Staff 1	64
Gambar 41 - Implementasi Untuk User Staff 2	64
Gambar 42 - Implementasi Untuk User Staff 3	65

DAFTAR LAMPIRAN

Lampiran A. Dokumen Proses Pengumpulan Requirement	85
Lampiran B. Link Produk	85
Lampiran C. Dokumen Pengujian Testing	85

ABSTRACT

The rapid advancement of information technology has driven digital transformation across various sectors, including logistics and inventory management, which play a crucial role in ensuring smooth business operations. Ineffective inventory management may lead to significant issues such as recording errors, discrepancies between system records and actual stock levels, and disruptions in goods distribution. Therefore, this study aims to design and develop a web-based inventory information system developed using the Laravel framework and MySQL database for MTSC Singapore Pte. Ltd. that incorporates Excel file upload functionality for bulk data entry, automated validation processes, and structured inventory tracking for incoming and outgoing goods.

The system was developed using the Waterfall software development methodology, consisting of requirements analysis, system design, implementation, and testing phases. The resulting web-based inventory system is capable of accelerating large-scale data entry processes, minimizing recording errors through automated validation, and providing a history tracking feature that enhances transparency and reporting accuracy.

The findings indicate that the developed system improves the efficiency of inventory management, reduces the risk of data inaccuracies, and facilitates the monitoring of stock movements. Furthermore, the system contributes to data-driven decision making within the organization and may serve as a valuable reference for future research and development in the field of inventory information systems.

Keywords: Inventory Management System, Web-Based Application, Excel File Upload, Data Validation, History Tracking

BAB I PENDAHULUAN

1.1. Latar Belakang

Perkembangan teknologi informasi dalam dua dekade terakhir telah membawa perubahan signifikan di berbagai sektor, termasuk *logistik* dan manajemen persediaan barang. Seiring perkembangan teknologi, digitalisasi proses bisnis telah mendorong lahirnya sistem *inventory spare part* berbasis *web* yang mampu mengintegrasikan data, mempercepat alur kerja, serta meningkatkan akurasi dalam pengambilan keputusan [1][2].

Untuk menopang perkembangan di atas dalam konteks *logistik* modern, pengelolaan *spare part* atau *inventory* menjadi tulang punggung yang memastikan kelancaran berbisnis, efisiensi distribusi serta keberlangsungan operasional perusahaan maupun industri secara internasional [3]. Dengan adanya pengelolaan *inventory* yang baik, perusahaan dapat menjaga ketersediaan barang, mencegah keterlambatan pengiriman produk, serta mengoptimalkan biaya penyimpanan. Sebaliknya, kesalahan dalam manajemen *inventory* berpotensi menimbulkan kerugian finansial yang besar, menghambat operasional, bahkan mengganggu stabilitas rantai pasok nasional dan internasional [4][5].

Salah satu permasalahan nyata yang sering muncul adalah masih banyak perusahaan, terutama skala menengah ke bawah, yang mengelola *inventory* menggunakan metode manual atau aplikasi *spreadsheet* sederhana seperti *Excel*. Metode ini memang fleksibel, tetapi rawan terhadap duplikasi data, kesalahan *input*, serta sulit dilakukan validasi ketika jumlah barang yang dikelola semakin banyak [6]. Akibatnya, sering terjadi perbedaan antara catatan stok di sistem dengan kondisi fisik barang di gudang, yang menimbulkan masalah serius seperti *overstock*, *stockout*, maupun kesalahan distribusi [7].

Untuk mengatasi permasalahan tersebut, dibutuhkan sistem *inventory* berbasis *web* yang terintegrasi dan mendukung pengelolaan data secara *real-time*. Sistem ini tidak hanya memfasilitasi proses pencatatan massal melalui fitur *upload Excel*, tetapi juga menyediakan validasi otomatis untuk meminimalisasi inkonsistensi data. Lebih lanjut, sistem perlu dilengkapi dengan fitur *history tracking* transaksi keluar-masuk barang, laporan periodik yang dapat diekspor, *role-based access control* untuk membedakan hak akses antara *Admin* dan pengguna

biasa, serta visualisasi data dalam bentuk grafik agar memudahkan analisis manajerial [8][9]. Dengan adanya sistem semacam ini, perusahaan dapat lebih mudah memantau ketersediaan barang, menekan biaya operasional, meningkatkan kecepatan pengambilan keputusan, serta menyediakan laporan historis yang dapat dijadikan bahan analisis.

Beberapa penelitian sebelumnya telah membahas pengembangan sistem *inventory* berbasis *web* dengan berbagai fokus. Misalnya, sistem pencatatan stok sederhana, integrasi dengan modul penjualan, serta implementasi notifikasi untuk peringatan kekurangan barang [10][11]. Namun, penelitian yang menggabungkan fitur *upload Excel* untuk *input* massal, validasi otomatis terhadap basis data, *history tracking* keluar-masuk barang, serta dukungan fungsi analitik dan pelaporan masih tidak banyak. Padahal kombinasi fitur tersebut sangat penting untuk menjawab tantangan pengelolaan *inventory* modern yang semakin kompleks.

Oleh karena itu, penelitian ini menawarkan kebaruan dengan merancang dan membangun sistem *inventory* berbasis *web* yang lebih komprehensif. Sistem ini dilengkapi dengan fitur *upload Excel*, validasi kecocokan data dengan basis data, *history tracking*, serta adanya laporan data periodik. Hasil penelitian ini diharapkan dapat membantu perusahaan meningkatkan efisiensi pengelolaan stok, meminimalisasi kesalahan pencatatan, mempercepat proses monitoring, serta mendukung pengambilan keputusan berbasis data.

1.2. Rumusan Masalah

Berdasarkan latar belakang yang telah dijabarkan, rumusan masalah dari penelitian ini adalah sebagai berikut:

1. Bagaimana merancang sistem *inventory spare part* berbasis *web* sesuai kebutuhan pengguna?
2. Bagaimana mengembangkan sistem *inventory spare part* dengan fitur unggulan seperti *upload Excel*, validasi otomatis, dan laporan periodik?
3. Bagaimana mengevaluasi kinerja sistem *inventory spare part* agar dapat berjalan efektif dan efisien menurut user nantinya berdasarkan parameter *blackbox* yang telah diajukan?

1.3. Tujuan

Tujuan adalah sasaran yang ingin dicapai dari proyek akhir. Penelitian ini bertujuan untuk:

1. Merancang sistem *inventory spare part* berbasis *web* sesuai kebutuhan fungsional dan non-fungsional.
2. Mengembangkan sistem *inventory spare part* dengan fitur utama *upload Excel*, validasi otomatis, laporan periodik, pencarian dan *filter*.
3. Melakukan evaluasi terhadap kinerja dan keandalan sistem agar sesuai dengan tujuan yang telah ditetapkan.

1.4. Batasan Masalah

Bagian ini menjelaskan ruang lingkup dan keterbatasan proyek agar lebih fokus, seperti lingkup pekerjaan yang dilakukan (apa saja yang dikerjakan), dan batasan teknis atau non-teknis (apa yang tidak termasuk). Batasan masalah dalam penelitian ini adalah sebagai berikut:

1. Sistem hanya difokuskan pada pengelolaan data stok barang tanpa integrasi dengan sistem lain seperti akuntansi atau penjualan.
2. Fitur *upload* hanya mendukung format file *Excel* (.xlsx atau .xls) dengan struktur data atau template tertentu.
3. Validasi data terbatas pada pencocokan data *inventory* terhadap data yang tersedia pada basis data sistem.
4. *Dashboard* visualisasi hanya berbasis data aktual tanpa *forecasting*.
5. Sistem laporan hanya dalam bentuk dasar (*Excel/PDF*), belum mendukung analitik canggih.

1.5. Manfaat

Bagian ini menjelaskan manfaat atau kontribusi yang diperoleh dari proyek, seperti manfaat teoritis/ilmiah (misalnya menambah pengetahuan atau bukti penerapan teknologi tertentu), dan manfaat praktis (misalnya membantu pengguna, memudahkan pekerjaan, meningkatkan efisiensi). Manfaat yang diharapkan dari penelitian ini adalah sebagai berikut:

1. Bagi Pengembang

Menjadi pengalaman dan kompetensi dalam merancang dan mengimplementasikan sistem *inventory spare part* berbasis *web* dengan fitur *upload* massal, validasi otomatis, dan menggunakan *dashboard* visual.

2. Bagi Pengguna

Mempermudah pencatatan stok barang, mempercepat *input* data secara massal, serta meningkatkan akurasi dan transparansi dalam monitoring persediaan.

3. Bagi Peneliti

Menjadi studi kasus dan referensi dalam pengembangan sistem informasi manajemen persediaan serta membuka peluang penelitian lanjutan seperti *forecasting* dan machine learning.

BAB II TINJAUAN PUSTAKA

Tinjauan Pustaka adalah kegiatan mengumpulkan data dan informasi melalui sumber terpercaya seperti buku, artikel, jurnal, karya ilmiah dan dalam bentuk tertulis atau digital yang terkait dengan pertanyaan penelitian. Tujuan tinjauan pustaka adalah membandingkan dan memperkuat hasil yang telah dicapai.

2.1. Penelitian Terkait

Tabel 1. Perbandingan Penelitian Sebelumnya

Peneliti/Tahun	Judul/Topik Penelitian	Metode/Teknologi	Kelebihan	Kekurangan/Celah
Hendi Sama, Eric (2023)	Perancangan dan Implementasi <i>Website Sistem Inventory</i> jdi SMK Globe National Plus 2 Kota Batam	<i>Scrum</i> , agile, PHP, MySQL	Simple, pencatatan lengkap	Tidak ada fitur <i>upload Excel</i> , tidak ada ekspor data ke <i>PDF</i>
Antonius Oko Pranoto, Eko Sedyono [2021]	Perancangan Sistem Informasi Inventaris Barang Berbasis <i>Web</i>	PHP, MySQL [menggunakan XAMPP]	Digitalisasi dari yang menggunakan buku, dipakai oleh perangkat desa	Tidak ada fitur <i>Excel</i> maupun ekspor data ke <i>PDF</i>
Harfizar, Ricky Dwi Prasetyo, Maya Indah Sari [2022]	Rancang Bangun <i>Website Sistem Informasi Inventory</i> Barang Pada PT Oni Utama Sukses	CodeIgniter, PHP, MySQL [menggunakan XAMPP], metode prototype	Sequence diagram yang jelas	Terlalu kompleks, prototype hanya berupa design UI/UX

Johani S Pasaribu [2021]	Development of a <i>Web Based Inventory Information System</i>	PHP, MySQL, Research method <i>Waterfall</i>	Simple namun on point.	Masih dalam bentuk <i>lokal web-based</i> [belum hosting]
Penelitian ini	Rancang Bangun Aplikasi <i>Inventory Spare Part</i> Berbasis <i>Web</i> Dengan Integrasi <i>Excel</i> dan Pencatatan <i>History</i> Barang	Laravel, MySQL	Kompleks, Memiliki fitur <i>upload Excel</i> , dapat print ke <i>PDF</i>	Cukup menjawab celah di penelitian sebelumnya

Dari perancangan yang dilakukan oleh Hendi Sama dan Eric (2023) yang berjudul “Perancangan dan Implementasi *Website Sistem Inventory* di SMK Globe National Plus 2 Kota Batam”, yang membahas tidak adanya alat teknologi untuk melakukan pencatatan asset ataupun barang milik sekolah di SMK Globe National Plus Batam sehingga dibutuhkannya perancangan tersebut untuk mempermudah pencatatan barang sekolah, dapat melakukan peminjaman barang, serta dapat digunakan di berbagai lokasi. Pada perancangan ini sistem *inventory* berbasis *websitenya* menggunakan metode *scrum* serta menggunakan Bahasa pemrograman PHP dengan basis data MySQL. Di perancangan ini terdapat banyak sekali informasi berguna seperti *acceptance criteria* diikuti *acceptance Testing* pada setiap aksi *User story* dan serta pendataan alur *websitenya* yang terstruktur. Kekurangan pada perancangan ini adalah tidak adanya fitur *upload Excel*, kita hanya bisa melihat data secara statis di dalam *web* tersebut dan tidak adanya fitur untuk mengekspor datanya kedalam bentuk *PDF*.

Dari perancangan Antonius Oko Pranoto dan Eko Sedyono, dalam perancangan berjudul “Perancangan Sistem Informasi Inventaris Barang Berbasis *Web*” pada tahun 2021 di Universitas Kristen Satya Wacana, perancangan ini

bertujuan untuk memfasilitasi performa dari perangkat desa dalam mengatur barang milik desa dimana semua data *input* masih dilakukan secara manual menggunakan kertas. Perancangan ini juga bertujuan untuk menghindari salah perhitungan fisik, laporan mutasi asset dan laporan barang dengan harapan perancangan ini dapat menaikkan performa dari perangkat desa dalam mengurus dan mendaftarkan data barang. Perancangan ini menggunakan Bahasa pemrograman PHP dengan basis data MySQL menggunakan integrasi dari aplikasi XAMPP. Perancangan ini sangat membantu pihak perangkat desa dalam hal digitalisasi aplikasi, serta mereka menggunakan dari segi pandang orang lain menggunakan kuesioner. Namun kekurangan pada perancangan ini adalah tidak adanya *use case skenario*, fitur *upload* data menggunakan *Excel*, dan fungsi ekspor menjadi bentuk *PDF*.

Dari perancangan oleh Harfizar, Ricky Dwi Prasetyo, Maya Indah Sari (2022) dengan judul “Rancang Bangun *Website* Sistem Informasi *Inventory* Barang Pada PT Oni Utama Sukses”, permasalahan yang dibahas pada perancangan ini adalah sistem untuk pengolahan data barang masuk dan data barang keluar masih dalam bentuk manual menggunakan kertas yang kemudian di salin ulang menggunakan software Microsoft *Excel*, kemudian dicetak dan diserahkan pada pimpinan perusahaan sehingga terkadang mengalami masalah dan kesulitan dalam menghitung barang, serta informasi yang di berikan kadang tidak sesuai dengan kenyataan. Maka dibuatlah prototype *web inventory* mereka guna untuk integrasi digital dan mempercepat kinerja perusahaan terutama di pergudangan. Perancangan ini menggunakan Bahasa PHP dan basis data MySQL dengan integrasi aplikasi XAMPP. Keunggulan di dalam perancangan ini adalah tujuan yang jelas serta penggunaan kata – kata dalam laporan yang tidak terlalu rumit. Namun kelemahan dari perancangan ini adalah hasil yang masih dalam bentuk prototipe dan masih dalam bentuk localhost, serta tidak adanya fitur untuk integrasi *upload* menggunakan *Excel* dan tidak dapat mengekspor data berbentuk *PDF*.

Dari perancangan Johni S Pasaribu pada tahun 2021 dengan judul “Development of a *Web Based Inventory* Information System”, perancangan ini dibuat karena adanya permintaan oleh CV. T. Kardin Pisau Indonesia dalam

digitalisasi di era informasi teknologi sekarang ini. Berharap untuk menghapus proses manual dalam menggunakan kertas dan celah untuk menyimpan data yang tidak sesuai. Metode yang digunakan adalah Research and Development (R&D). Penelitian ini dibuat menggunakan YII *Framework* atau turunan dari PHP dengan metode MVC. Kelebihan dalam perancangan ini adalah laporan yang disusun menggunakan bahasa Inggris dan memiliki antarmuka yang informatif dan mudah digunakan. Kekurangan di perancangan ini adalah *context diagram* dan *data flow diagram* (DFD) yang cukup kompleks untuk ditampilkan pada sebuah laporan tugas akhir sehingga sulit untuk dipahami.

Penelitian ini berjudul “PENGEMBANGAN SISTEM INVENTORY SPARE PART DENGAN INTEGRASI UPLOAD EXCEL DAN PENCATATAN RIWAYAT BARANG” yang bertujuan untuk membantu MTSC Singapore Pte. Ltd. dalam mengelola data inventory spare part secara lebih efektif dan terstruktur dibandingkan metode pencatatan manual yang sebelumnya digunakan. Sistem ini dikembangkan menggunakan framework Laravel dengan basis data MySQL serta menerapkan metode Waterfall sebagai metode pengembangan perangkat lunak. Sistem yang dibangun menyediakan fitur pengelolaan data spare part, transaksi barang masuk dan keluar, unggah data menggunakan file Excel, validasi data inventory, pencatatan riwayat transaksi barang (history tracking), serta pembuatan laporan dalam format PDF. Keunggulan penelitian ini dibandingkan penelitian sebelumnya terletak pada integrasi fitur unggah data Excel untuk mempercepat proses input data dalam jumlah besar, validasi data untuk mengurangi kesalahan pencatatan, dan pencatatan history barang yang memudahkan proses pelacakan transaksi. Dengan adanya integrasi fitur-fitur tersebut dalam satu sistem inventory berbasis web, penelitian ini diharapkan dapat memberikan solusi yang lebih komprehensif dalam pengelolaan inventory spare part serta menjawab beberapa keterbatasan yang ditemukan pada penelitian-penelitian sebelumnya.

2.2. Landasan Teori

2.2.1. Teori Umum

2.2.1.1 Sistem Informasi

Sistem informasi adalah suatu kombinasi dari teknologi informasi dan aktivitas manusia yang menggunakan teknologi tersebut untuk mendukung operasi dan manajemen dalam suatu organisasi. Menurut Laudon & Laudon (2020), sistem informasi dapat membantu perusahaan dalam mengelola, mengontrol, menganalisis, serta mengoptimalkan alur kerja sehingga lebih efisien.

2.2.1.2 Persediaan (*Inventory*)

Inventory atau persediaan merupakan barang-barang yang dimiliki oleh perusahaan baik berupa bahan baku, barang setengah jadi, maupun barang jadi, yang disimpan untuk digunakan dalam proses produksi atau dijual kembali. Manajemen persediaan bertujuan untuk menjaga ketersediaan barang agar tidak terjadi kelebihan maupun kekurangan stok. Dalam dunia industri, pengelolaan *inventory* sangat penting untuk menjaga kelancaran operasi perusahaan.

2.2.1.3 *Spare Part* (Suku Cadang)

Spare part adalah komponen suku cadangan yang digunakan untuk perawatan dan penggantian peralatan sebuah mesin. Manajemen *Spare part* merupakan proses pengelolaan stok, pemesanan, dan distribusi komponen cadangan agar ketersediaan suku cadang mesin tetap terjaga. Jika tidak dikelola dengan baik, dapat menyebabkan *downtime* atau tidak berfungsinya sebuah mesin yang berakibat pada kerugian sebuah perusahaan.

2.2.1.4 Aplikasi Berbasis *Web*

Aplikasi berbasis *web* adalah sebuah perangkat lunak yang dijalankan menggunakan *browser* dan dapat diakses melalui jaringan internet. Aplikasi *web* memungkinkan pengguna untuk mengakses data stok barang secara *real-time* tanpa terbatas pada perangkat tertentu.

2.3. Metode *Waterfall*

Metode *Waterfall* adalah model pengembangan perangkat lunak yang bersifat terstruktur dan beruntun, dimana setiap tahap harus diselesaikan sebelum melanjutkan ke tahap berikutnya. Tahapan dalam metode ini meliputi analisis kebutuhan, perancangan sistem, implementasi, pengujian, dan pemeliharaan. Model ini mempermudah pengawasan proyek karena alur kerja yang jelas dan sistematis.

2.3.1. Analisis Kebutuhan

Menurut Sommerville (2016) [18], tahap Analisis kebutuhan bertujuan untuk mengidentifikasi kebutuhan fungsional dan non-fungsional dari perangkat lunak yang akan dibangun. Rencana kegiatan pada proyek ini:

- Melakukan wawancara dengan manajemen MTSC Singapore Pte. Ltd. untuk mengetahui permasalahan pada sistem manual.
- Mengumpulan kebutuhan sistem seperti pencatatan *Spare Part*, fitur impor/ekspor data, serta pengaturan *User role*.
- Mendokumentasikan kebutuhan dalam bentuk daftar fungsional dan non-fungsional.

2.3.2. Desain Sistem

Menurut Pressman (2015) [17], tahap desain ini bertujuan untuk mentransformasikan kebutuhan yang telah di analisis menjadi sebuah rancangan sistem yang dapat di implementasikan. Rencana kegiatan pada proyek ini:

- Membuat rancangan arsitektur sistem berbasis *web* dengan menggunakan Laravel.
- Mendesain Entity Relationship Diagram atau ERD untuk basis data menggunakan MySQL.
- Membuat desain antarmuka untuk halaman – halaman yang diperlukan. Contoh halaman *login*, *dashboard*, manajemen *Spare Part*, serta laporan *inventory*.

2.3.3. Implementasi

Menurut Sommerville (2016) [18], tahap implementasi adalah proses mengubah desain sistem metode program yang menggunakan Bahasa pemrograman dan *framework* yang telah dipilih. Rencana kegiatan pada proyek ini adalah:

- Mengembangkan *front end* dan *back end* menggunakan Laravel MVC
- Mengembangkan aplikasi sesuai kebutuhan fungsional dan desain yang telah direncanakan.

2.3.4. Pengujian

Menurut Pressman (2015) [17], tahap pengujian ini bertujuan untuk memverifikasi dan memvalidasi hasil dari perangkat lunak yang telah dibuat untuk sesuai dengan kebutuhan pengguna serta bebas dari kesalahan. Rencana kegiatan pada proyek ini:

- Melakukan *Unit Testing* pada fungsi *login*, manajemen *Spare Part*, dan impor ekspor data.
- Melakukan *Black Box Testing* agar memastikan bahwa seluruh kebutuhan telah terpenuhi.

2.3.5. Pemeliharaan

Menurut Bassil (2012) [16], pada tahap pemeliharaan ini dilakukan untuk memperbaiki kesalahan yang tidak terdeteksi saat pengujian, serta mengembangkan sistem lebih lanjut jika ada kebutuhan baru. Rencana pada proyek ini:

- Memberikan dokumentasi sistem dan pelatihan penggunaan aplikasi kepada manajemen MTSC Singapore Pte. Ltd.
- Melakukan perbaikan bug yang muncul setelah implementasi.
- Menyediakan kemungkinan pengembangan fitur lebih lanjut seperti integrasi barcode scanner dan lain lain.

2.4. Laravel

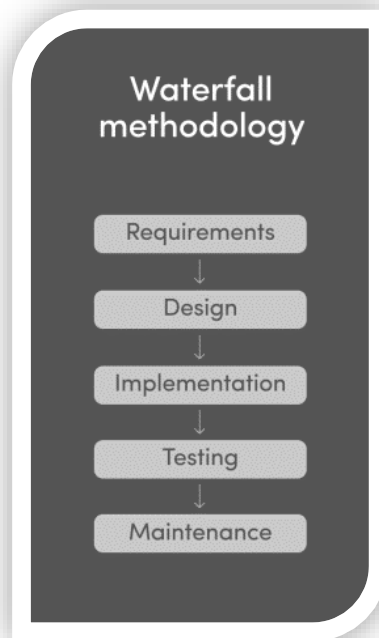
Laravel adalah sebuah Bahasa pemrograman berbasis *framework* PHP (Hypertext Preprocessor) yang merupakan Bahasa pemrograman server-side yang umum digunakan untuk pengembangan aplikasi *web* dinamis. Laravel dapat menyediakan berbagai fitur seperti routing, ORM (Eloquent), serta sistem autentikasi yang mempermudah pengembangan aplikasi.

2.5. Basis data MySQL

MySQL adalah sistem manajemen basiss data relasional (RDMBS) yang menggunakan *Structured Query Language* (SQL) untuk mengelola data. MySQL mendukung integritas data, relasi antar Tabel, dan efisiensi dalam mengelola basis data.

BAB III ANALISIS DAN PERANCANGAN

3.1. Metode Pengembangan Produk



Gambar 1. Metode *Waterfall*

Metode pengembangan perangkat lunak yang digunakan dalam Proyek Akhir ini adalah model *Waterfall*. Model *Waterfall* merupakan salah satu metode klasik dalam rekayasa perangkat lunak yang bersifat *linear* dan berurutan. Bassil, 2012 [16], model *Waterfall* membagi proses pengembangan perangkat lunak ke dalam beberapa tahap yang dilakukan secara berurutan, dimana setiap tahap harus diselesaikan sebelum melanjutkan ke tahap berikutnya.

Kelebihan dari model *Waterfall* ini adalah struktur pengembangannya yang jelas, terdokumentasi dengan baik serta mudah dipahami. Hal ini sesuai dengan kebutuhan perancangan Proyek Akhir, di mana sistem yang akan dikembangkan memiliki spesifikasi yang relatif stabil dan tidak banyak berubah.

Penggunaan metode Waterfall pada penelitian ini memberikan beberapa keuntungan yang sesuai dengan karakteristik proyek yang dikembangkan. Kebutuhan sistem inventory spare part telah dapat diidentifikasi sejak awal melalui proses observasi dan wawancara dengan pengguna, sehingga ruang lingkup pengembangan relatif jelas dan tidak mengalami perubahan yang signifikan selama proses pengembangan. Dengan adanya kebutuhan yang stabil, setiap tahapan dalam metode Waterfall, mulai dari analisis kebutuhan, perancangan sistem, implementasi, hingga pengujian, dapat dilakukan secara terstruktur dan terdokumentasi dengan baik.

Selain itu, metode Waterfall memudahkan proses penyusunan dokumentasi proyek akhir karena setiap tahapan menghasilkan artefak yang dapat dijadikan dasar untuk tahap berikutnya. Hal ini sangat membantu dalam penyusunan laporan akademik, karena seluruh proses pengembangan sistem dapat dijelaskan secara sistematis mulai dari identifikasi kebutuhan hingga evaluasi hasil pengujian. Dokumentasi yang lengkap juga memudahkan proses pemeliharaan dan pengembangan sistem di masa mendatang.

Metode Waterfall juga sesuai dengan kondisi pengembangan proyek yang dilakukan oleh satu orang pengembang dengan waktu pengerjaan yang terbatas. Dibandingkan metode Agile atau Scrum yang membutuhkan iterasi berulang dan komunikasi intensif dengan pengguna, Waterfall memungkinkan proses pengembangan berjalan lebih terarah sesuai jadwal yang telah ditentukan. Dengan demikian, metode ini dinilai mampu mendukung penyelesaian proyek secara efektif sekaligus memastikan bahwa kebutuhan utama pengguna dapat diimplementasikan dengan baik.

3.2. Analisis Kebutuhan

3.2.1. Proses Bisnis Berjalan

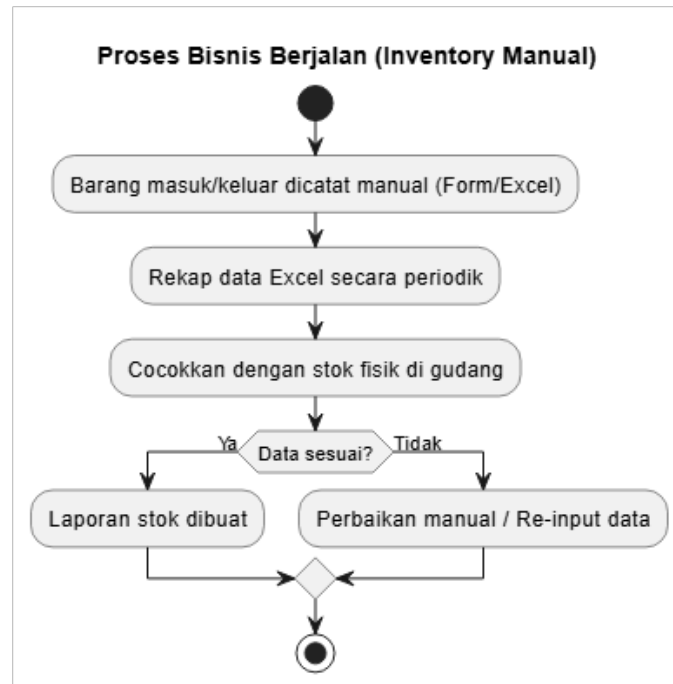
Saat ini, proses pengelolaan *inventory* pada perusahaan masih dilakukan dengan metode manual atau menggunakan aplikasi *spreadsheet* sederhana seperti Microsoft *Excel*. Alur kerja yang berjalan dapat dijelaskan sebagai berikut:

1. Pencatatan barang masuk dan keluar dilakukan oleh *Staff* menggunakan form kertas atau file *Excel*.
2. Rekapitulasi stok barang disusun secara periodik dengan cara menggabungkan file *Excel* dari beberapa bagian.
3. Laporan stok barang disusun manual dan biasanya membutuhkan waktu lama karena harus menghitung ulang serta mencocokkan data dengan kondisi fisik di gudang.
4. Validasi data sangat terbatas, sering terjadi duplikasi, kesalahan *input* atau ketidaksesuaian jumlah stok di *Excel* dengan kondisi nyata.
5. Akses informasi stok hanya bisa dilakukan oleh orang tertentu yang memiliki file *Excel*-nya. Sehingga kurang transparan dan rawan keterlambatan informasi.

Permasalahan yang dihadapi:

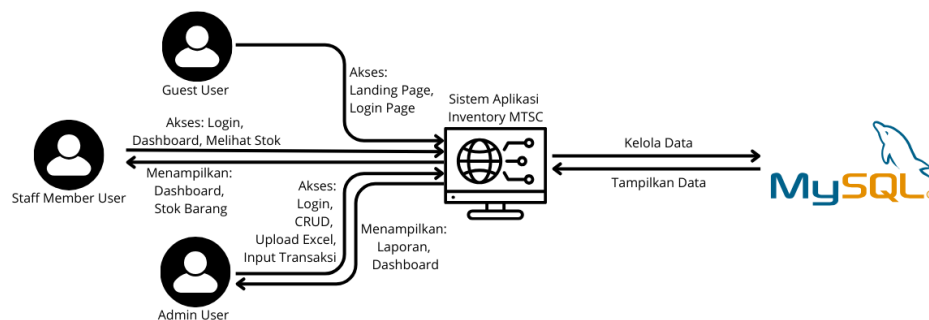
1. Tingginya risiko kesalahan *input* dan duplikasi data karena pencatatan manual.
2. Proses rekap stok dan pembuatan laporan memakan waktu lama
3. Tidak ada validasi otomatis sehingga ketidaksesuaian stok sering terlambat diketahui.
4. Monitoring stok sulit dilakukan secara *real-time*.
5. Keterbatasan akses karena hanya yang memiliki file yang bisa mengetahui data gudang.

Flowchart untuk proses bisnis saat ini:



Gambar 2. Proses Bisnis Berjalan Saat Ini

3.2.2. Gambaran Umum Sistem



Gambar 3. Arsitektur Gambaran Umum Sistem

Gambar di atas menunjukkan arsitektur sederhana dari sistem aplikasi ini yang terhubung dengan basis data MySQL untuk pengelolaan data. Sistem ini secara garis besar memiliki tiga jenis pengguna yang masing-masing memiliki tingkat dan fungsi berbeda.

1. *Guest User*

Secara akses, *Guest User* hanya dapat mengakses halaman utama sistem ini. Halaman tersebut digunakan digunakan untuk memperkenalkan sistem, menampilkan informasi umum, serta menjadi halaman awal sebelum pengguna melakukan *login*.

2. *Staff Member User*

Secara akses, *Staff Member* dapat melakukan *login*, diteruskan dengan melakukan *input* transaksi, mengakses laporan, dan melihat *dashboard*.

3. *Admin User*

Secara akses, *Admin* memiliki hak penuh atas sistem ini dan mencakup semua akses dari *Staff Member*. Admin juga dapat untuk melakukan *CRUD Spare Part*, mengupload data menggunakan *Excel*, dan pengelolaan *User* dan sistem.

Alur interaksi yang dapat disimpulkan adalah *Guest User* mengakses halaman awal kemudian *Staff* atau *Admin* melakukan proses *login*. Setelah *login*, *User* dapat melakukan berbagai hal aktivitas tergantung peran. Sistem *Inventory MTSC* ini akan berkomunikasi dengan MySQL untuk menyimpan atau menampilkan data hasil aktivitas *User*, dimana data akan ditampilkan kembali melalui antarmuka seperti laporan dan dasbor.

3.3. Perancangan

Bagian ini memuat rancangan sistem yang akan dikembangkan. Perancangan dilakukan berdasarkan analisis kebutuhan yang telah dijelaskan pada bagian sebelumnya.

3.3.1. Kebutuhan Fungsional

Bagian ini berisi daftar fungsi atau layanan yang harus disediakan oleh sistem agar dapat berjalan sesuai tujuan.

Tabel 2. Kebutuhan Fungsional

Kode	Deskripsi Kebutuhan Fungsional	Aktor
FR001	Sistem menyediakan halaman <i>login</i> untuk <i>Admin</i> dan <i>Staff Member</i> .	<i>Admin</i> , <i>Staff Member</i>
FR002	Sistem menyediakan fitur <i>logout</i> untuk <i>Admin</i> dan <i>Staff Member</i> .	<i>Admin</i> , <i>Staff Member</i>
FR003	<i>Admin</i> dapat menambahkan, mengubah, dan menghapus <i>User</i> beserta <i>rol</i> nya.	<i>Admin</i>
FR004	Sistem menerapkan <i>role-based access control</i> sesuai <i>role</i> (<i>Admin</i> vs <i>Member</i>).	<i>Admin</i> , <i>Staff Member</i>
FR005	<i>Admin</i> dapat melakukan <i>CRUD</i> master data barang (tambah, ubah, hapus).	<i>Admin</i>
FR006	<i>Admin</i> dapat mencatat barang masuk ke <i>inventory</i> .	<i>Admin</i>
FR007	<i>Admin</i> dapat mencatat barang keluar dari <i>inventory</i> .	<i>Admin</i>
FR008	Sistem menampilkan stok barang <i>real-time</i> .	<i>Admin</i> , <i>Staff Member</i>
FR009	Sistem mencatat <i>history</i> transaksi keluar/masuk barang.	<i>Admin</i> , <i>Staff Member</i>
FR010	<i>Admin</i> dapat melakukan <i>upload</i> data massal melalui file <i>Excel</i> (.xls/.xlsx).	<i>Admin</i>
FR011	Sistem melakukan validasi otomatis untuk mengecek kesesuaian data <i>Excel</i> dengan basis data.	<i>Admin</i>
FR012	Sistem memberikan peringatan jika ada duplikasi/inkonsistensi data.	<i>Admin</i>
FR013	<i>Admin</i> & <i>Staff Member</i> dapat mencetak laporan transaksi (harian/mingguan/bulanan).	<i>Admin</i> , <i>Staff Member</i>
FR014	Sistem mendukung ekspor laporan ke <i>PDF</i> dan <i>Excel</i> .	<i>Admin</i> , <i>Staff Member</i>

FR015	Sistem menampilkan <i>dashboard</i> visualisasi data berupa grafik/chart.	<i>Admin, Staff</i> Member
FR016	Sistem menyediakan fitur pencarian berdasarkan <i>parameter</i> dasar (kode/nama barang).	<i>Admin, Staff</i> Member
FR017	Sistem menyediakan fitur <i>filter</i> data agar pencarian lebih cepat.	<i>Admin, Staff</i> Member
FR018	<i>Guest</i> hanya dapat mengakses landing page tanpa <i>login</i> .	<i>Guest</i>

3.3.2. Kebutuhan Non-Fungsional

Berisi spesifikasi teknis yang tidak langsung terlihat dari fitur, tetapi penting untuk kualitas sistem.

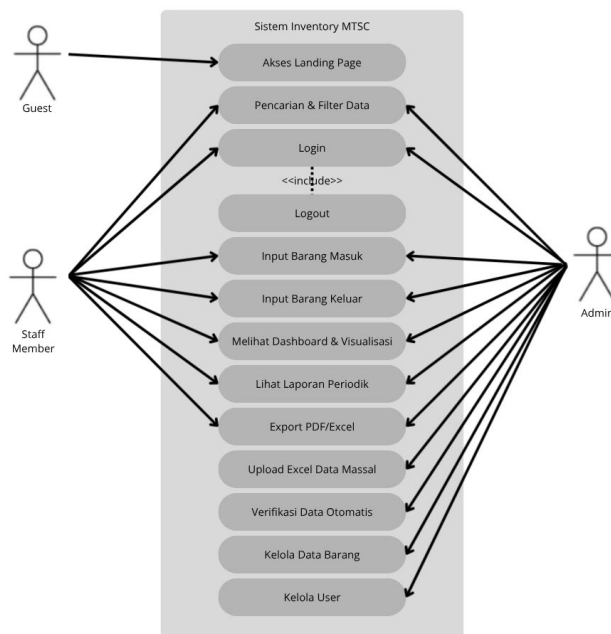
Tabel 3. Kebutuhan Non-Fungsional

Kode	Kategori	Deskripsi Kebutuhan Non-Fungsional
NFR001	Keamanan	Sistem menggunakan autentikasi berbasis session/token untuk <i>login</i> .
NFR002	Keamanan	Password <i>User</i> harus disimpan dengan enkripsi.
NFR003	Keamanan	<i>Role-based access control</i> membatasi hak akses sesuai <i>role</i> .
NFR004	Skalabilitas	Sistem dapat dikembangkan untuk mendukung jumlah data dan <i>User</i> lebih banyak tanpa penurunan kinerja.
NFR005	Kompatibilitas	Aplikasi dapat diakses melalui <i>browser</i> modern (Chrome, Edge, Firefox).
NFR006	Kompatibilitas	Mendukung <i>upload</i> file <i>Excel</i> dengan format standar (.xls/.xlsx).
NFR007	Pemeliharaan	Struktur kode menggunakan arsitektur MVC (Laravel).
NFR008	<i>User</i> Experience	<i>Dashboard</i> dilengkapi grafik visual yang mudah dipahami.

NFR009	Laporan	Laporan yang diekspor mempertahankan format tabel dan dapat dicetak dengan rapi.
--------	---------	--

3.3.3. Diagram Use case

Diagram ini menggambarkan interaksi antara pengguna (aktor) dengan sistem. Diagram harus dibuat sesuai dengan simbol standar UML.



Gambar 4. Use case Diagram

3.3.4. Skenario Use case

Menjelaskan detail langkah-langkah interaksi antara aktor dengan sistem untuk setiap *use case*.

Tabel 4. Use case Skenario FR001

No	FR001 - <i>Login</i> ke Sistem
Deskripsi	Sistem menyediakan halaman <i>login</i> untuk <i>Admin</i> dan Staff Member.
Aktor Utama	<i>Admin</i> , Staff Member

Prasyarat	Pengguna memiliki akun valid (<i>Username & password</i>).
Kondisi Akhir	Pengguna berhasil masuk ke sistem sesuai <i>rolenya</i> .
Skenario Utama	1. Pengguna membuka halaman <i>login</i>. 2. Pengguna memasukkan <i>Username & password</i>. 3. Sistem memvalidasi data <i>login</i>. 4. Sistem mengarahkan pengguna ke <i>dashboard</i> sesuai <i>role</i>.

Menjelaskan proses *login* bagi *Admin* dan *Staff Member* untuk mengakses sistem. Sistem akan memvalidasi akun dan mengarahkan pengguna ke *dashboard* sesuai dengan *rolenya*.

Tabel 5. *Use case Skenario FR002*

No	FR002 - <i>Logout</i> dari Sistem
Deskripsi	Sistem menyediakan fitur <i>logout</i> untuk <i>Admin</i> dan <i>Staff Member</i> .
Aktor Utama	<i>Admin</i> , <i>Staff Member</i>
Prasyarat	Pengguna sudah <i>login</i> ke sistem.
Kondisi Akhir	Pengguna keluar dari sistem dan diarahkan ke halaman <i>login</i> .
Skenario Utama	1. Pengguna menekan tombol <i>logout</i>. 2. Sistem mengakhiri sesi <i>login</i>. 3. Sistem mengarahkan pengguna ke halaman <i>login</i>.

Menjelaskan mekanisme pengguna keluar dari sistem. Setelah *logout* sesi pengguna akan diakhiri dan diarahkan kembali ke halaman *login*.

Tabel 6. *Use case Skenario FR003*

No	FR003 - <i>Kelola User dan Role</i>
----	-------------------------------------

Deskripsi	<i>Admin</i> dapat menambahkan, mengubah, dan menghapus <i>User</i> beserta <i>rolenya</i> .
Aktor Utama	<i>Admin</i>
Prasyarat	<i>Admin</i> sudah <i>login</i> ke sistem.
Kondisi Akhir	Data <i>User</i> berhasil disimpan/diubah/dihapus.
Skenario Utama	1. <i>Admin</i> masuk ke menu manajemen <i>User</i>. 2. <i>Admin</i> memilih tambah/edit/hapus. 3. <i>Admin</i> mengisi atau mengubah data <i>User</i>. 4. Sistem menyimpan perubahan ke basis data. 5. Sistem menampilkan konfirmasi berhasil.

Menjelaskan kemampuan *Admin* untuk mengolah data pengguna dan *rolenya*, termasuk menambah, mengedit, dan menghapus *User*. Sistem memastikan perubahan tersimpan di basis data.

Tabel 7. *Use case Skenario FR004*

No	FR004 - Hak Akses Berdasarkan <i>Role</i>
Deskripsi	Sistem menerapkan <i>role-based access control</i> sesuai <i>role</i> (<i>Admin</i> vs Staff Member).
Aktor Utama	<i>Admin</i> , Staff Member
Prasyarat	Pengguna sudah <i>login</i> ke sistem.
Kondisi Akhir	Pengguna hanya bisa mengakses fitur sesuai <i>role</i> .
Skenario Utama	1. Pengguna <i>login</i> ke sistem. 2. Sistem mengidentifikasi <i>role</i> pengguna.

	3. Sistem hanya menampilkan menu yang sesuai dengan <i>role</i> tersebut.
--	---

Menggambarkan penerapan *role access control* dimana sistem menampilkan menu dan fitur sesuai dengan peran pengguna.

Tabel 8. *Use case Skenario* FR005

No	FR005 - Kelola Data Barang
Deskripsi	<i>Admin</i> dapat melakukan <i>CRUD</i> master data barang.
Aktor Utama	<i>Admin</i>
Prasyarat	<i>Admin</i> sudah <i>login</i> ke sistem.
Kondisi Akhir	Data barang tersimpan atau diperbarui di basis data.
Skenario Utama	1. <i>Admin</i> membuka menu master barang. 2. <i>Admin</i> memilih tambah/edit/hapus barang. 3. <i>Admin</i> mengisi form barang. 4. Sistem menyimpan perubahan ke basis data.

Menjelaskan proses Create, Read, Update, Delete untuk master data barang oleh *Admin*. Perubahan yang dilakukan akan disimpan di basis data.

Tabel 9. *Use case Skenario* FR006

No	FR006 - Mencatat Barang Masuk
Deskripsi	<i>Admin</i> dapat mencatat barang masuk ke <i>inventory</i> .
Aktor Utama	<i>Admin</i>
Prasyarat	<i>Admin</i> sudah <i>login</i> ke sistem.
Kondisi Akhir	Data barang masuk tercatat di basis data.

Skenario Utama	1. <i>Admin</i> membuka menu barang masuk. 2. <i>Admin</i> mengisi detail barang masuk. 3. Sistem menyimpan data barang masuk ke basis data.
----------------	--

Menjelaskan Langkah *Admin* dalam mencatat transaksi barang masuk ke *inventory*. Sistem menyimpan data tersebut sebagai riwayat transaksi masuk.

Tabel 10. *Use case Skenario FR007*

No	FR007 - Mencatat Barang Keluar
Deskripsi	<i>Admin</i> dapat mencatat barang keluar dari <i>inventory</i> .
Aktor Utama	<i>Admin</i>
Prasyarat	<i>Admin</i> sudah <i>login</i> ke sistem.
Kondisi Akhir	Data barang keluar tercatat di basis data.
Skenario Utama	1. <i>Admin</i> membuka menu barang keluar. 2. <i>Admin</i> mengisi detail barang keluar. 3. Sistem menyimpan data barang keluar ke basis data.

Menjelaskan proses pencatatan barang keluar oleh *Admin*. Sistem akan menyimpan data transaksi keluar di basis data.

Tabel 11. *Use case Skenario FR008*

No	FR008 - Lihat Stok <i>Real-time</i>
Deskripsi	Sistem menampilkan stok barang <i>real-time</i> .
Aktor Utama	<i>Admin</i> , Staff Member
Prasyarat	Pengguna sudah <i>login</i> ke sistem.

Kondisi Akhir	Pengguna dapat melihat stok terkini.
Skenario Utama	1. Pengguna membuka menu stok. 2. Sistem menampilkan data stok barang <i>real-time</i> .

Menjelaskan fitur untuk menampilkan data stok barang terkini. Baik *Admin* ataupun Staff Member dapat melihat update stok secara langsung.

Tabel 12. *Use case Skenario FR009*

No	FR009 - <i>History</i> Transaksi
Deskripsi	Sistem mencatat <i>history</i> transaksi keluar/masuk barang.
Aktor Utama	<i>Admin</i> , Staff Member
Prasyarat	Pengguna sudah <i>login</i> ke sistem.
Kondisi Akhir	<i>History</i> transaksi tersimpan.
Skenario Utama	1. Pengguna membuka menu <i>history</i> . 2. Sistem menampilkan daftar transaksi keluar/masuk.

Menjelaskan kemampuan sistem dalam menampilkan daftar riwayat transaksi barang masuk dan keluar. Pengguna dapat menelusuri data transaksi yang sudah tercatat.

Tabel 13. *Use case Skenario FR010*

No	FR010 - <i>Upload</i> Data Massal
Deskripsi	<i>Admin</i> dapat melakukan <i>upload</i> data massal melalui file <i>Excel</i> (.xls/.xlsx).
Aktor Utama	<i>Admin</i>
Prasyarat	<i>Admin</i> sudah <i>login</i> ke sistem.
Kondisi Akhir	Data massal tersimpan di basis data.
Skenario Utama	1. <i>Admin</i> membuka menu <i>upload</i> . 2. <i>Admin</i> memilih file <i>Excel</i> . 3. Sistem memproses file dan menyimpan data.

Menjelaskan fitur *upload* file *Excel* bagi *Admin* untuk menambah banyak data sekaligus. Sistem memproses file dan menyimpan datanya ke basis data.

Tabel 14. *Use case Skenario* FR011

No	FR011 - Validasi Data <i>Excel</i>
Deskripsi	Sistem melakukan validasi otomatis untuk mengecek kesesuaian data <i>Excel</i> dengan basis data.
Aktor Utama	<i>Admin</i>
Prasyarat	<i>Admin</i> melakukan <i>upload</i> file <i>Excel</i> .
Kondisi Akhir	Data tervalidasi sesuai basis data.
Skenario Utama	1. <i>Admin upload</i> file <i>Excel</i>. 2. Sistem memvalidasi data dengan basis data. 3. Sistem menampilkan hasil validasi.

Menjelaskan bagaimana sistem secara otomatis memvalidasi data *Excel* yang di unggah agar sesuai dengan data yang ada di basis data.

Tabel 15. *Use case Skenario* FR012

No	FR012 - Peringatan Duplikasi/Inkonsistensi
Deskripsi	Sistem memberikan peringatan jika ada duplikasi/inkonsistensi data.
Aktor Utama	<i>Admin</i>
Prasyarat	<i>Admin</i> melakukan <i>input/upload</i> data.
Kondisi Akhir	Sistem menampilkan peringatan kesalahan.
Skenario Utama	1. <i>Admin input/upload</i> data. 2. Sistem mendeteksi data ganda/tidak konsisten. 3. Sistem menampilkan peringatan.

Menjelaskan bahwa sistem akan memberi peringatan jika ditemukan data ganda atau tidak konsisten selama *input* atau *upload* data.

Tabel 16. *Use case Skenario* FR013

No	FR013 - Cetak Laporan Transaksi
Deskripsi	<i>Admin</i> & Staff Member dapat mencetak laporan transaksi (harian/mingguan/bulanan).
Aktor Utama	<i>Admin</i> , Staff Member
Prasyarat	Pengguna sudah <i>login</i> .
Kondisi Akhir	Laporan transaksi berhasil dicetak.
Skenario Utama	1. Pengguna membuka menu laporan. 2. Pengguna memilih periode laporan. 3. Sistem menghasilkan laporan transaksi. 4. Pengguna mencetak laporan.

Menjelaskan fitur cetak laporan transaksi berdasarkan periode tertentu. Laporan dapat dicetak oleh *Admin* dan Staff Member.

Tabel 17. *Use case Skenario* FR014

No	FR014 - Ekspor Laporan
Deskripsi	Sistem mendukung ekspor laporan ke <i>PDF</i> dan <i>Excel</i> .
Aktor Utama	<i>Admin</i> , Staff Member
Prasyarat	Pengguna sudah <i>login</i> .
Kondisi Akhir	Laporan berhasil diekspor ke format yang dipilih.
Skenario Utama	1. Pengguna membuka menu laporan. 2. Pengguna memilih format ekspor (<i>PDF/Excel</i>). 3. Sistem menghasilkan file ekspor.

Menjelaskan kemampuan sistem mengekspor laporan kedalam format *PDF* atau *Excel* agar mudah dibagikan atau disimpan.

Tabel 18. *Use case Skenario FR015*

No	FR015 - <i>Dashboard</i> Visualisasi
Deskripsi	Sistem menampilkan <i>dashboard</i> visualisasi data berupa grafik/chart.
Aktor Utama	<i>Admin</i> , Staff Member
Prasyarat	Pengguna sudah <i>login</i> .
Kondisi Akhir	<i>Dashboard</i> tampil dengan grafik data.
Skenario Utama	1. Pengguna membuka menu <i>dashboard</i>. 2. Sistem menampilkan grafik data <i>inventory</i>.

Menjelaskan tampilan dasbor dengan grafik dan chart untuk memvisualisasikan data *inventory*. Fitur ini membantu pemantauan data secara cepat.

Tabel 19. *Use case Skenario FR016*

No	FR016 - Pencarian Data
Deskripsi	Sistem menyediakan fitur pencarian berdasarkan <i>parameter</i> dasar (kode/nama barang).
Aktor Utama	<i>Admin</i> , Staff Member
Prasyarat	Pengguna sudah <i>login</i> .
Kondisi Akhir	Pengguna menemukan data barang dengan cepat.
Skenario Utama	1. Pengguna memasukkan kata kunci pencarian. 2. Sistem menampilkan hasil pencarian sesuai <i>parameter</i>.

Menjelaskan fitur pencarian barang berdasarkan kode atau nama. Fitur ini membantu pengguna menemukan data dengan lebih cepat.

Tabel 20. *Use case Skenario* FR017

No	FR017 - <i>Filter</i> Data
Deskripsi	Sistem menyediakan fitur <i>filter</i> data agar pencarian lebih cepat.
Aktor Utama	<i>Admin</i> , Staff Member
Prasyarat	Pengguna sudah <i>login</i> .
Kondisi Akhir	Pengguna mendapatkan data sesuai <i>filter</i> .
Skenario Utama	1. Pengguna memilih <i>parameter filter</i> . 2. Sistem menampilkan data sesuai <i>filter</i> .

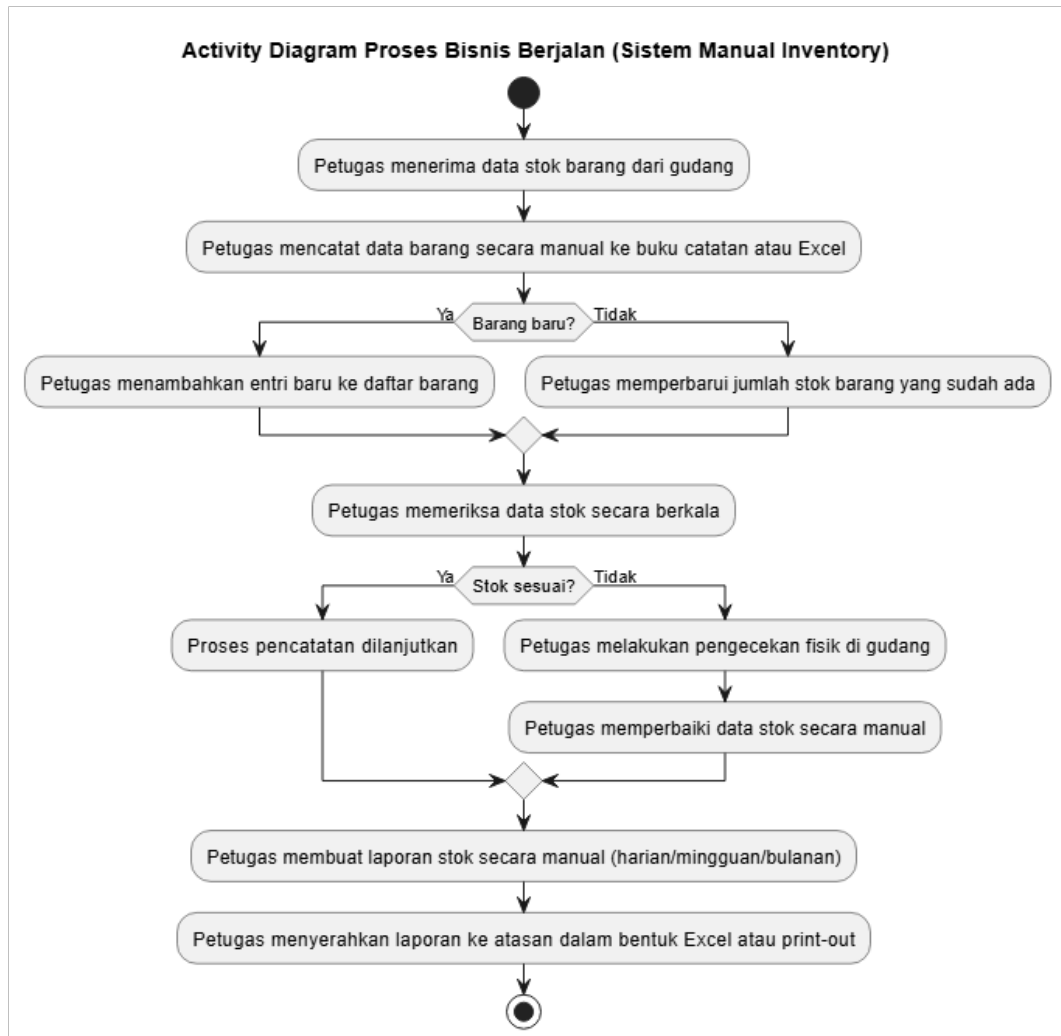
Menjelaskan fitur *filter* untuk mempersempit hasil pencarian berdasarkan *parameter* tertentu. Fitur ini meningkatkan efisiensi pencarian data.

Tabel 21. *Use case Skenario* FR018

No	FR018 - Akses Landing Page <i>Guest</i>
Deskripsi	<i>Guest</i> hanya dapat mengakses landing page tanpa <i>login</i> .
Aktor Utama	<i>Guest</i>
Prasyarat	Pengguna terhubung internet.
Kondisi Akhir	<i>Guest</i> hanya melihat informasi umum.
Skenario Utama	1. <i>Guest</i> membuka <i>website</i> . 2. Sistem menampilkan halaman landing page.

Menjelaskan akses terbatas bagi *Guest* yang hanya dapat melihat halaman awal sistem. *Guest* tidak perlu *login* dan hanya melihat informasi umum.

3.3.5. Activity diagram



Gambar 5. Flowchart Activity Diagram

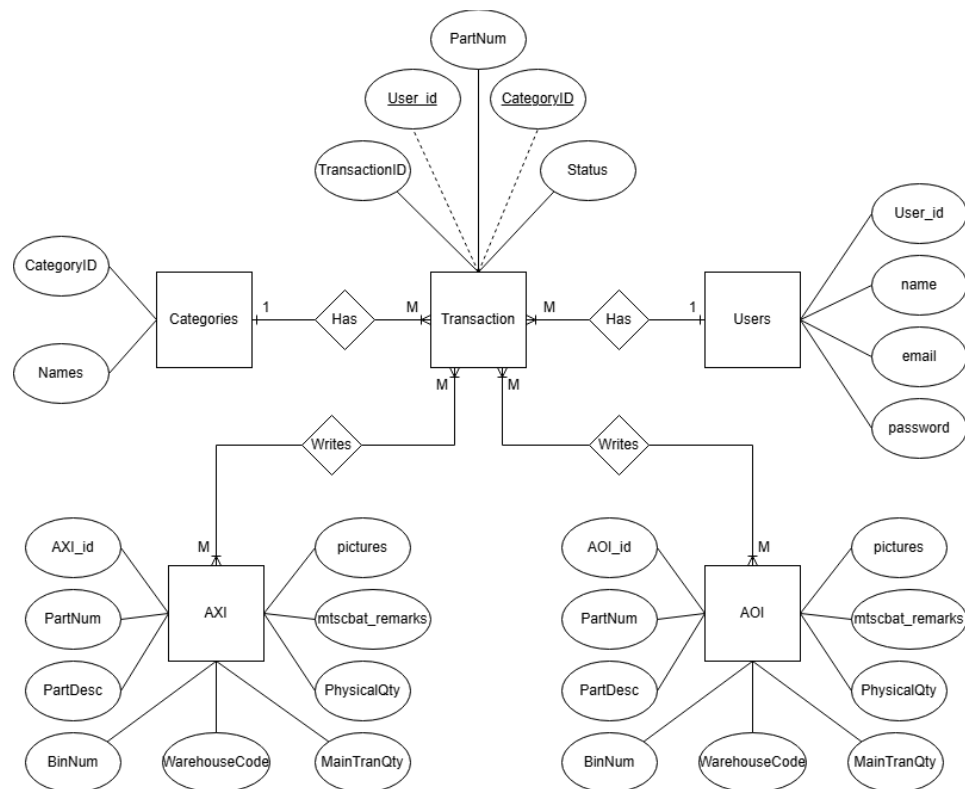
Activity diagram pada gambar 5 di atas ini menggambarkan alur proses bisnis pengelolaan *inventory* yang saat ini masih dilakukan secara manual. Proses dimulai ketika petugas mencatat setiap barang masuk dan keluar menggunakan formulir atau file *Excel*. Data dari beberapa transaksi kemudian direkap secara periodik setiap minggu untuk memperoleh data total stok barang yang tersedia.

Selanjutnya, hasil rekap dibandingkan dengan stok fisik di gudang guna memastikan kesesuaian data. Apabila data sesuai, maka sistem manual menghasilkan laporan stok yang siap dilaporkan kepada pihak manajemen. Namun jika ditemukan ketidaksesuaian, maka petugas akan harus melakukan perbaikan secara manual melalui *re-input* data kedalam file *Excel*.

Proses ini menunjukkan bahwa sistem yang berjalan masih memiliki beberapa kelemahan diantaranya tingginya potensi kesalahan pencatatan, duplikasi data, serta keterlambatan dalam pembaharuan informasi stok. Oleh karena itu, dibutuhkan sistem *inventory* berbasis *web* yang dapat mengotomasi proses tersebut dan memungkinkan validasi data secara *real-time*.

3.3.6. ER Diagram

Bagian ini memuat rancangan basis data yang menggambarkan entitas, atribut, dan relasi antar entitas. ER Diagram lengkap dengan hubungan antar entitas. Juga sertakan penjelasan singkat setiap entitas dan atributnya.

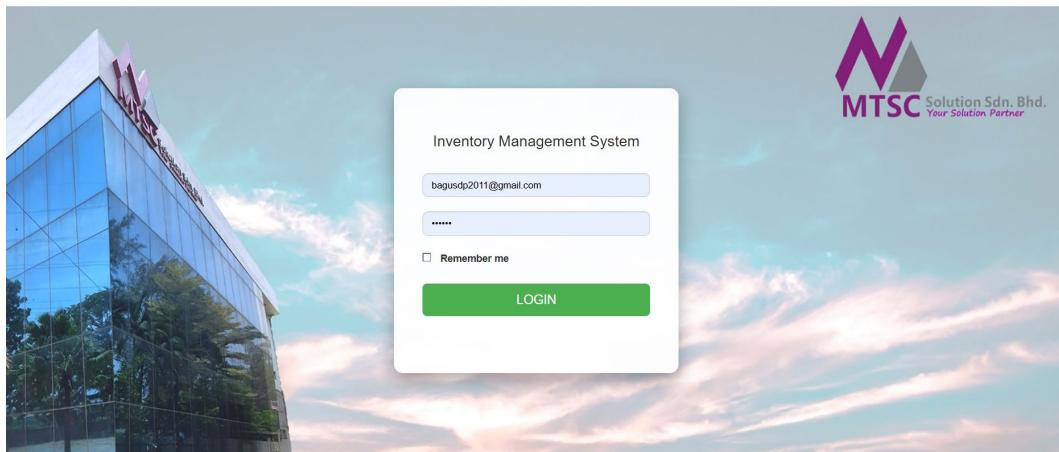


Gambar 6. Entity Relation Diagram

Gambar 6 menggambarkan entity relationship diagram dari sistem *inventory* MTSC.

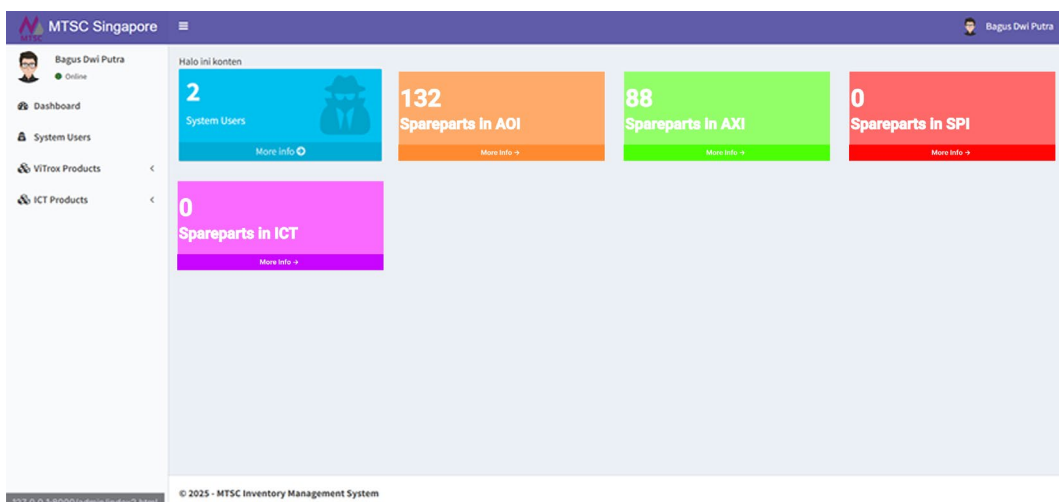
3.3.7. Perancangan Antarmuka (Wireframe)

Bagian ini menampilkan rancangan awal tampilan sistem atau rancangan perangkat keras jika proyek berbasis IoT.



Gambar 7. *Login Page*

Gambar di atas merupakan tampilan halaman *login* ke *website*.



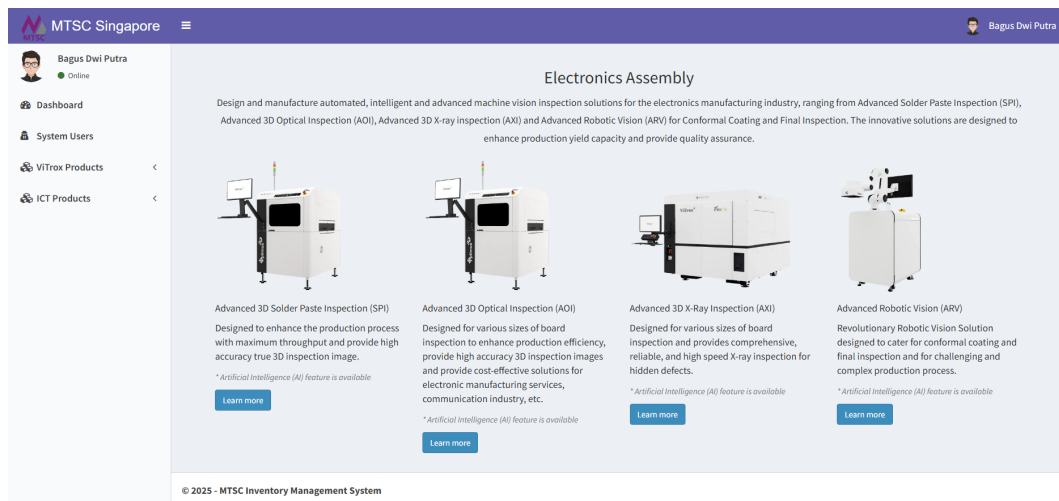
Gambar 8. *Dashboard Page*

Gambar di atas adalah gambar halaman dasbor setelah sukses *login* dengan *User*.



Gambar 9. Vitrox Machines *Dashboard* Page

Gambar di atas adalah gambar penjelasan dari perusahaan vitrox yang kami jadikan sebagai dasbor dari halaman vitrox.



Gambar 10. Vitrox Machines *Display* Page

Gambar di atas adalah gambar untuk penjelasan beberapa mesin yang ada dari merek Vitrox.

MTSC Singapore

Bagus Dwi Putra

Online

Dashboard

System Users

ViTroX Products

ICT Products

Upload Excel AOI

Choose File No file chosen

Import AOI

Tampilkan 10 data per halaman

Carit:

PartNum	PartDesc	WarehouseCode	BinNum	MainTranQty	PhysicalQty	Remarks
2112-0006	SOLID STATE DRIVE,512GB,2.5"	SCP	PO-MTSCBAT	2	1	1 Used in PCI
2140-0011	POWER SUPPLY, 1AC, 24VDC, 10A, DIN RAIL	SCP	PO-MTSCBAT	1	1	
2140-0012	POWER SUPPLY, AMAZON INTELLIGENT, 7.5KW	SCP	PO-MTSCBAT	1	2	Extra 1
2140-0014	UNINTERRUPTIBLE POWER SUPPLY,180 - 266V,300 WATTS / 500VA,3 MINUTES	SCP	PO-MTSCBAT	1	1	
2140-0057	Power Supply, 750W ATX, 110V-240V, 80 Plus Gold	SCP	PO-MTSCBAT	1	1	
2140-0064	POWER SUPPLY, DC to DC, Input: (48Vdc, 8A max), Output: (12Vdc x4, 5Vdc x1)	SCP	PO-MTSCBAT	1	1	
2150-0041	PLC, CP1E - 40 IO, N-TYPED, WITH BATTERY	SCP	PO-MTSCBAT	1	1	
2150-0133	GRAPHIC CARD,NVIDIA GTX1080,PCI-E X8 / X16 GEN3,8GB	SCP	PO-MTSCBAT	1	1	
2220-0205	RESISTOR,2200R,+/- 1%,0.5W	SCP	PO-MTSCBAT	1	1	
2240-0013	INDUCTOR, INRUSH CURRENT LIMITERS	SCP	PO-MTSCBAT	1	1	

Menampilkan 1 sampai 10 dari 64 data

Sebelumnya 1 2 3 4 5 6 7 Berikutnya

© 2025 - MTSC Inventory Management System

Gambar 11. AOI Page

Gambar adalah gambar sementara untuk halaman mesin Automated Optical Inspection.

MTSC Singapore

Bagus Dwi Putra

Online

Dashboard

System Users

ViTroX Products

ICT Products

Upload Excel AXI

Choose File No file chosen

Import AXI

Truncate AXI

+ Tambah Data Manual

Tampilkan 10 data per halaman

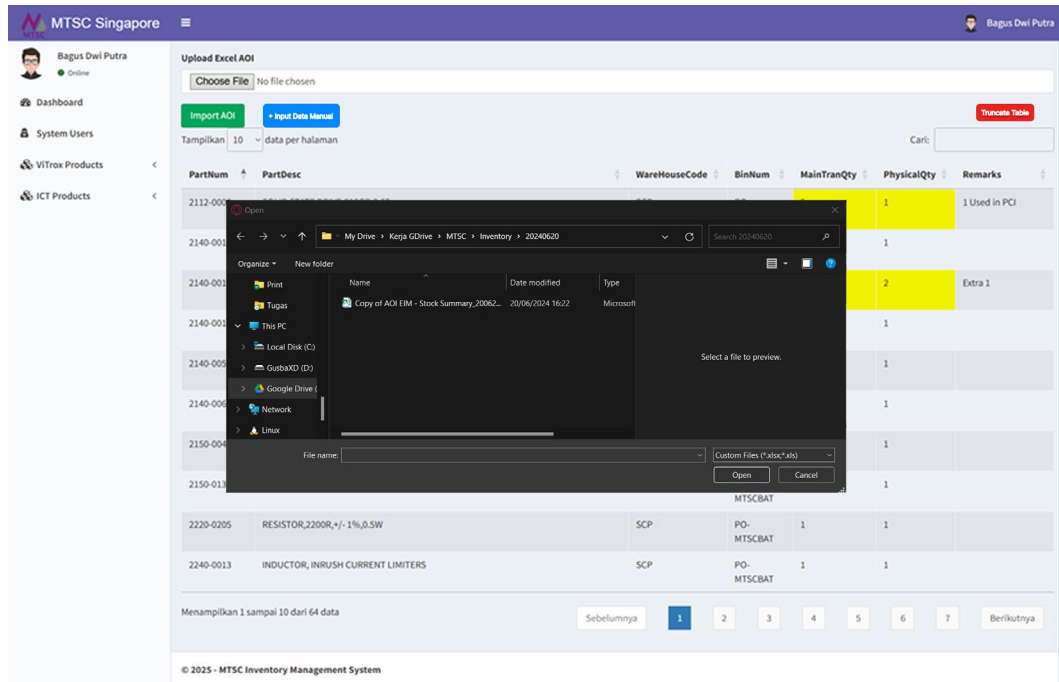
Carit:

Aksi	PartNum	PartDesc	WarehouseCode	BinNum	MainTranQty	PhysicalQty	Remarks	Pictures
	2110-0050	HARD DISK,SATA3 6GB/S,1TB	Batam	AXI	1	1		
	2140-0023	POWER SUPPLY, 1400W, 240VAC	Batam	PX-MTSCBAT	1	1		
	2140-0069	PC POWER SUPPLY,1U REDUNDANT PSU,80+ PLATINUM,920 WATT	Batam	PX-MTSCBAT	2	2		No image
	2140-0082	PC POWER SUPPLY,REGULATED POWER ADAPTER VOLTAGE,-,120 WATT (19 VDC)	Batam	PX-MTSCBAT	2	2		No image
	2140-0094	PC POWER SUPPLY,CWT KPL-060F-VI 4PIN DC 12V 5A REGULATED SWITCHING POWER ADAPTER,-,60 W	Batam	PX-MTSCBAT	1	1		No image

© 2025 - MTSC Inventory Management System

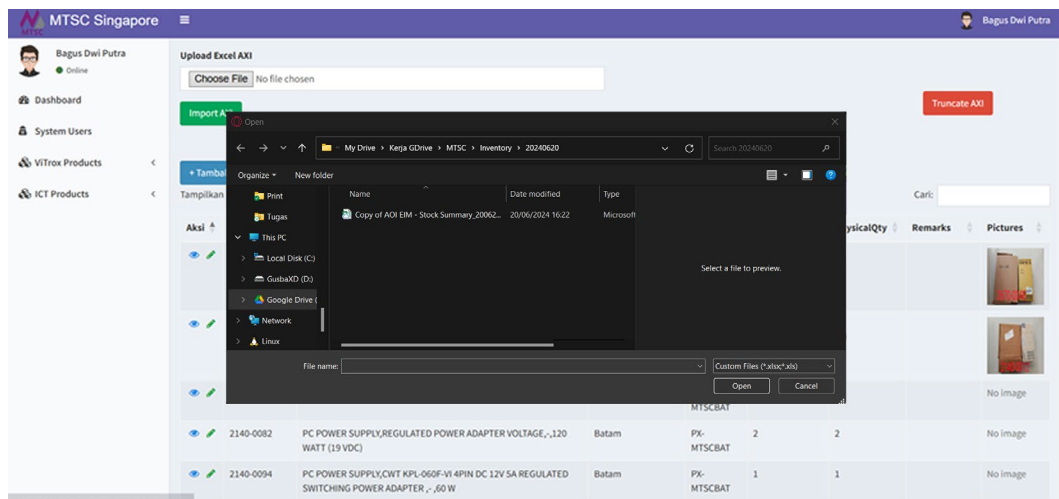
Gambar 12. AXI Page

Gambar adalah halaman sementara untuk merepresentasikan mesin AXI atau Automated Xray Inspection.



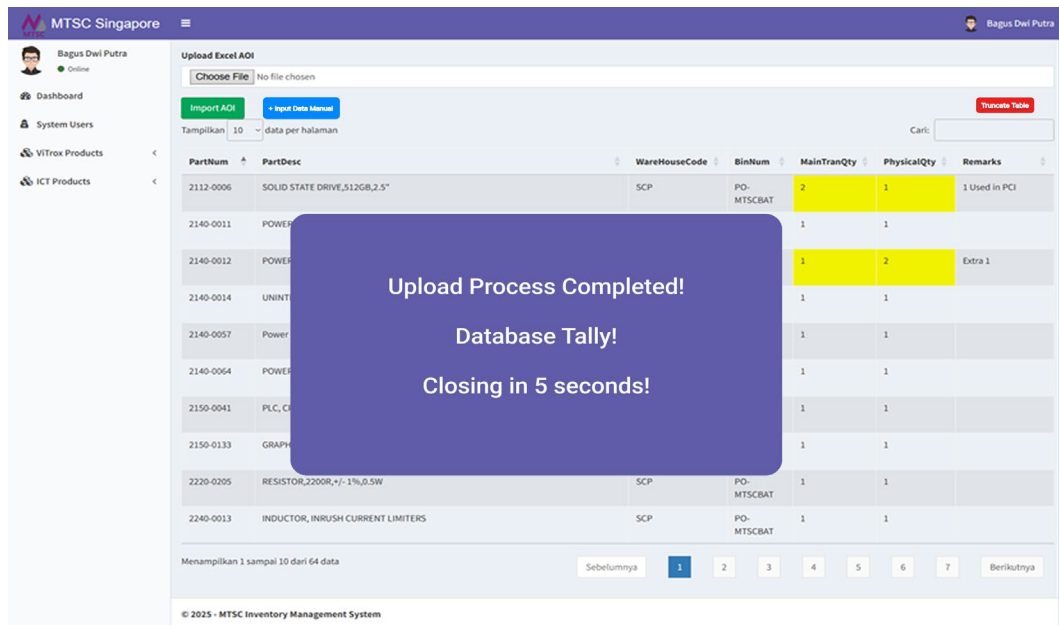
Gambar 13. AOI Opening *Excel*

Gambar di atas merupakan proses ketika kita akan mengupload file *Excel* kedalam basis data AOI.



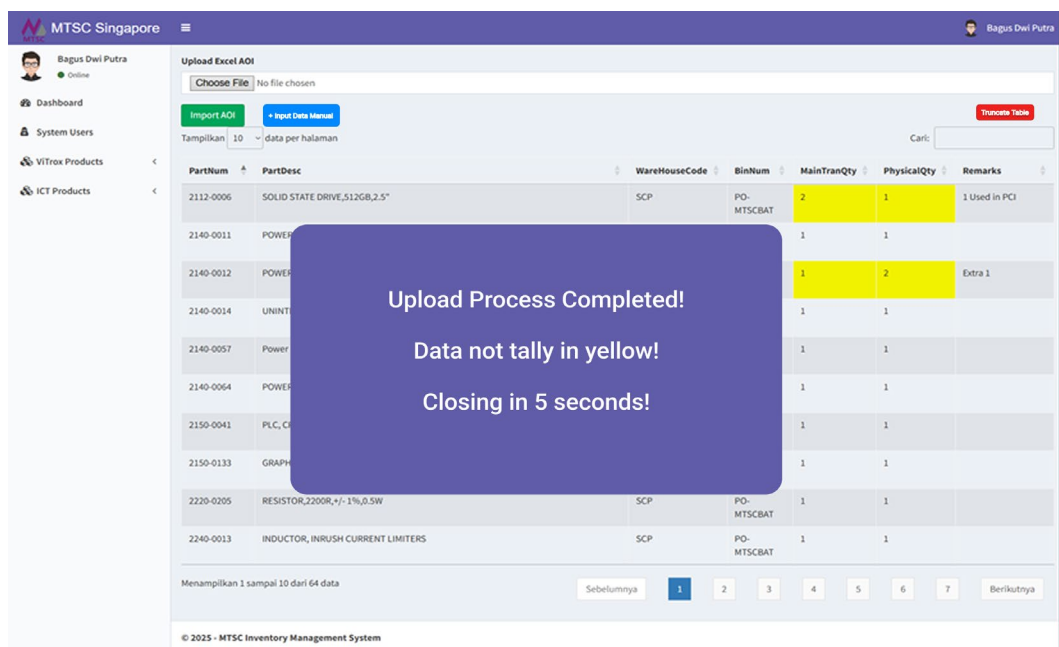
Gambar 14. AXI Opening *Excel*

Gambar di atas merupakan proses ketika kita akan mengupload file *Excel* kedalam basis data AXI.



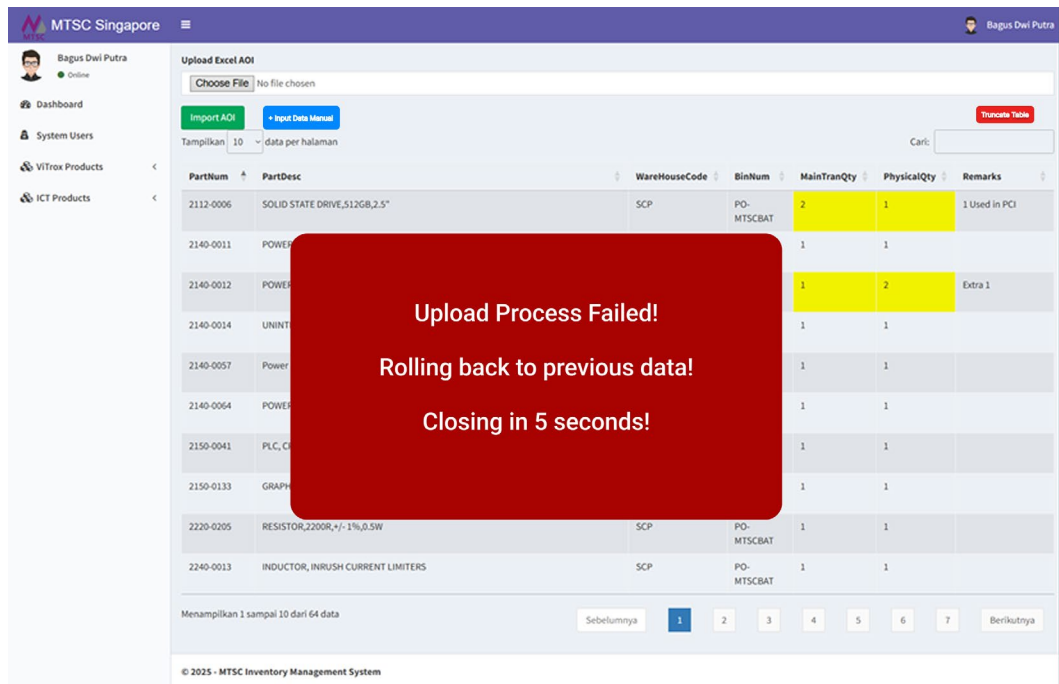
Gambar 15. Basis data *Tally*

Gambar di atas adalah proses interaksi ketika file *Excel* dengan basis data memiliki data yang sama.



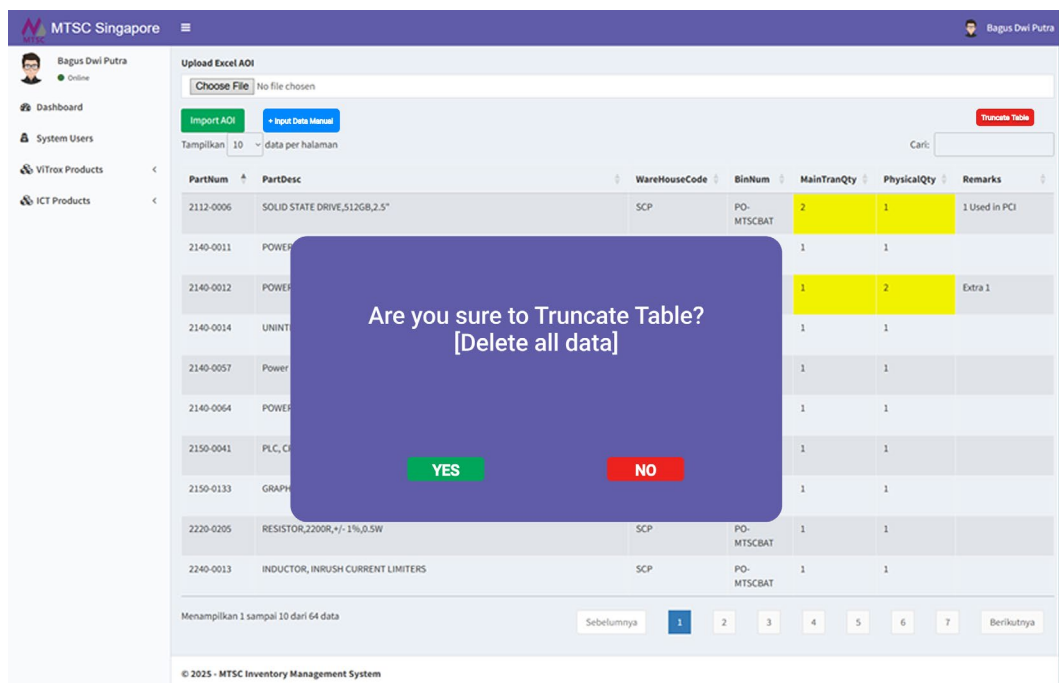
Gambar 16. Basis data Not *Tally*

Gambar di atas adalah proses interaksi ketika file *Excel* dengan basis data memiliki data yang berbeda stok. Data yang berbeda akan diwarnai dengan warna kuning.



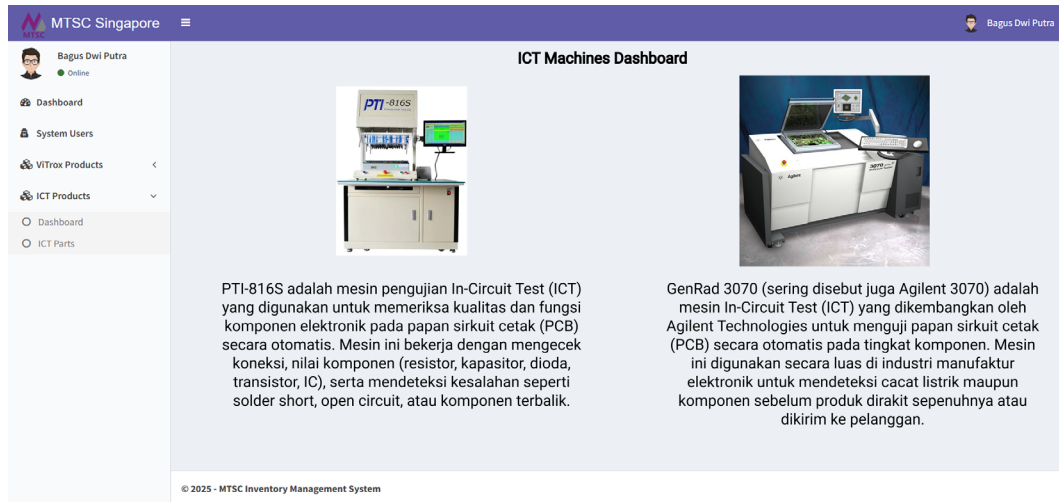
Gambar 17. *Excel Failed to Upload*

Gambar di atas adalah proses interaksi ketika *upload Excel* gagal.



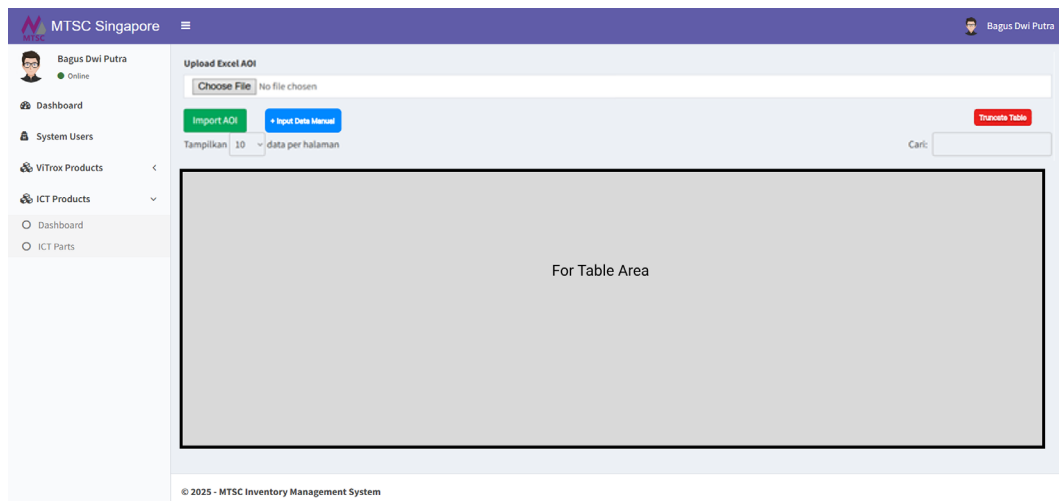
Gambar 18. Delete All Data

Gambar di atas adalah proses interaksi jika pengguna ingin menghapus semua data yang ada di dalam basis data.



Gambar 19. ICT Machine *Dashboard*

Gambar di atas merupakan halaman untuk sebagai dasbor dan untuk menjelaskan mesin ICT yang dimiliki oleh MTSC.



Gambar 20. ICT *Spare Part* Page

Gambar di atas adalah halaman untuk melihat isi Tabel dari *Spare Part* ICT yang dimiliki oleh MTSC.

3.4. Rancangan Pengujian

3.4.1. Tujuan Pengujian

Tujuan pengujian ini adalah untuk memastikan bahwa seluruh fitur sistem *inventory* telah berfungsi dengan benar, mudah digunakan dan sesuai dengan kebutuhan pengguna yang berada di MTSC. Pengujian ini juga bertujuan untuk menemukan serta memperbaiki kesalahan (bug) sebelum sistem diimplementasikan.

3.4.2. Alur Pengujian

Alur pengujian dilakukan secara berurutan dengan tahapan sebagai berikut:

1. Persiapan
 - a. Menyediakan data uji seperti data barang berupa *Excel*.
 - b. Menyiapkan server lokal berbasis localhost untuk menjalankan sistem Laravel.
 - c. Menjelaskan kepada responden mengenai tujuan dan cara penggunaan sistem.
2. Pelaksanaan Pengujian Internal (*Black Box Testing*)
 - a. Pengembang melakukan pengujian pada setiap fitur berdasarkan *skenario* pengujian.
 - b. Setiap hasil dicatat dalam Tabel pengujian untuk statusnya gagal atau berhasil.
 - c. Bug atau error yang di temukan akan di perbaiki dan diuji ulang.
3. Evaluasi Hasil Uji
 - a. Data hasil pengujian dikumpulkan
 - b. Hasil evaluasi untuk menentukan apakah sistem layak diterapkan.

- c. Apabila ditemukan kendala, dilakukan revisi sistem dan pengujian ulang.

3.4.3. Responden Pengujian

Responden uji dipilih dari pihak yang menggunakan sistem secara langsung, meliputi:

Tabel 22. Responden Pengujian

No	Jabatan/Peran	Jumlah	Keterangan
1	<i>Admin</i> (Supervisor)	1 orang	Pengguna utama yang melakukan <i>input</i> dan pengelolaan stok barang
2	Staff Member (<i>Staff</i> pendukung)	1 orang	Pengguna yang mengakses data stok dan laporan

Jumlah total responden sebanyak 2 orang dinilai cukup untuk memberikan umpan balik representatif terkait fungsionalitas, kemudahan dan efektivitas sistem.

3.4.4. Metode Pengujian

Metode pengujian yang digunakan dalam proyek akhir ini meliputi:

1. *Black Box Testing*

Metode ini digunakan untuk menguji fungsionalitas sistem tanpa melihat kode sumber. Fokusnya adalah pada *input* dan *output* sistem, apakah sesuai dengan kebutuhan fungsional yang telah ditentukan pada Tabel kebutuhan fungsional.

2. *Feature Testing (Unit Testing)*

Metode ini digunakan untuk menguji fitur-fitur tertentu secara terpisah guna memastikan bahwa logika bisnis dan fungsi yang dikembangkan berjalan sesuai dengan rancangan. Pengujian dilakukan pada fitur yang memiliki peran penting dalam sistem, yaitu fitur unggah file Excel,

validasi data duplikat saat proses impor data, serta halaman landing page untuk pengguna umum (guest user). Pengujian dilakukan menggunakan framework PHPUnit yang terintegrasi dengan Laravel untuk memverifikasi bahwa setiap fitur menghasilkan keluaran yang sesuai dengan hasil yang diharapkan.

BAB IV IMPLEMENTASI DAN PENGUJIAN

4.1 Hasil Implementasi

Tahap implementasi merupakan proses penerapan rancangan sistem yang telah dibuat pada tahap analisis dan perancangan menjadi sebuah aplikasi yang dapat digunakan secara nyata. Pada penelitian ini, sistem inventory dikembangkan menggunakan framework Laravel dengan arsitektur Model–View–Controller (MVC) serta menggunakan MySQL sebagai basis data. Implementasi sistem dilakukan dengan membangun beberapa modul utama yang sesuai dengan kebutuhan fungsional yang telah ditentukan pada Bab III. Modul tersebut meliputi modul autentikasi pengguna, manajemen data barang, pencatatan transaksi barang masuk dan keluar, fitur unggah data melalui file Excel, serta modul laporan dan *dashboard* visualisasi data.

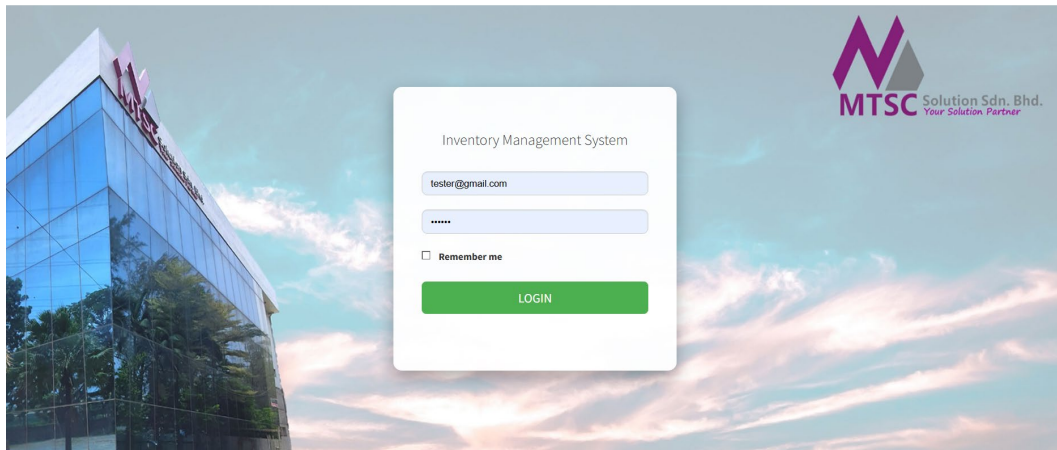
4.1.1. Implementasi Teknologi Sistem

Sistem inventory ini dibangun dengan spesifikasi teknologi sebagai berikut:

Komponen	Teknologi
Bahasa Pemrograman	PHP 7.4.29
Framework	Laravel Framework 8.83.29
Basis Data	MySQL Workbench 8.0
Frontend	HTML, CSS, JavaScript, Bootstrap 5
Library Tambahan	Laravel Excel
Web Server	Apache / Localhost

Penggunaan framework Laravel dipilih karena menyediakan berbagai fitur yang mempermudah pengembangan aplikasi berbasis web seperti sistem routing, ORM (Eloquent), autentikasi pengguna, serta struktur kode yang terorganisir.

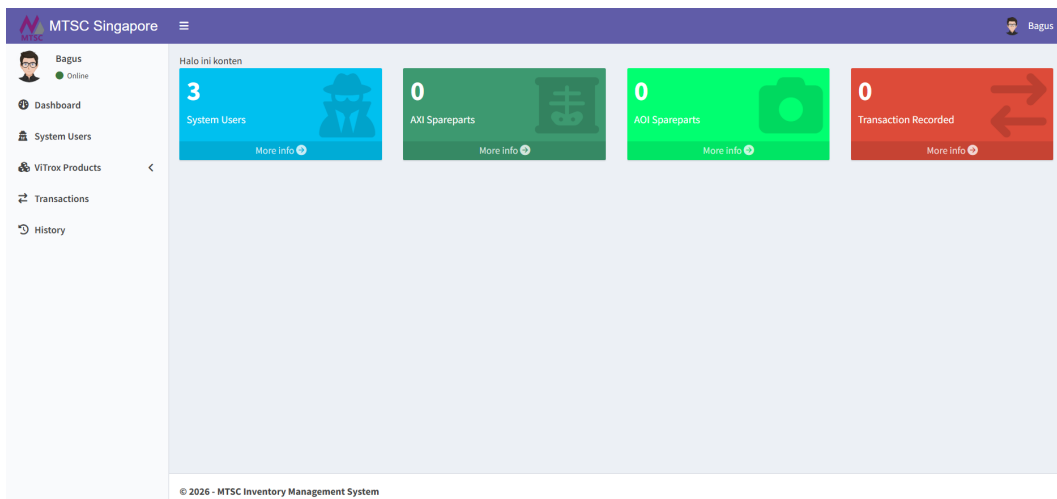
4.1.2. Implementasi Halaman Login



Gambar 21 - Implementasi Login

Halaman login merupakan halaman pertama yang diakses oleh pengguna sebelum masuk ke dalam sistem. Pada halaman ini, pengguna diminta untuk memasukkan username dan password yang telah terdaftar pada sistem. Sistem akan melakukan proses validasi terhadap data yang dimasukkan. Jika data valid, pengguna akan diarahkan menuju halaman *dashboard* sesuai dengan peran (role) yang dimiliki. Sebaliknya, jika data tidak valid, maka sistem akan menampilkan pesan kesalahan. Halaman ini berfungsi sebagai mekanisme keamanan agar hanya pengguna yang memiliki akun yang dapat mengakses sistem inventory.

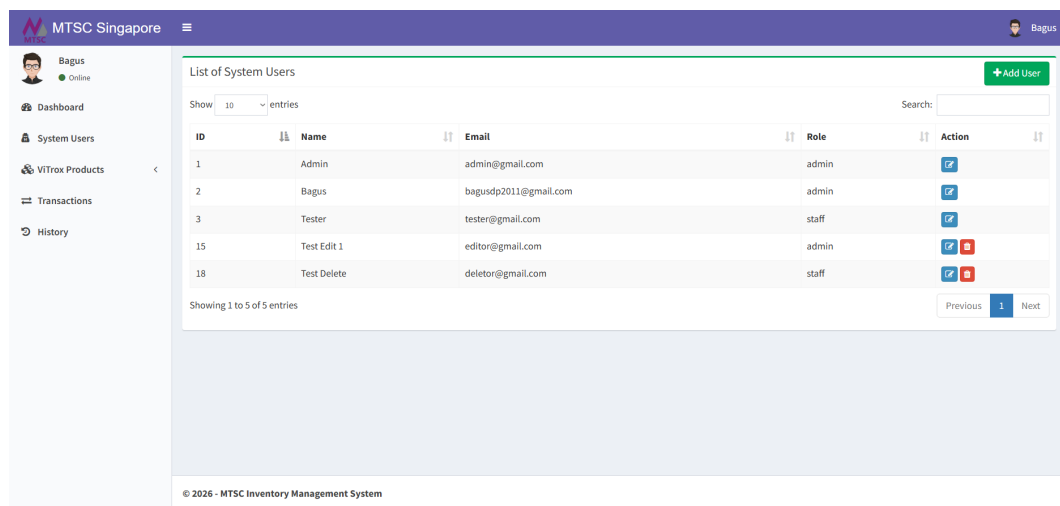
4.1.3. Implementasi *Dashboard* Sistem Sebagai *Admin*



Gambar 22 - Implementasi Dashboard Setelah Login

Dashboard merupakan halaman utama setelah pengguna berhasil login ke dalam sistem. Halaman ini menampilkan informasi ringkasan mengenai data inventory seperti jumlah total barang, jumlah transaksi barang masuk dan keluar, serta visualisasi data dalam bentuk grafik. Visualisasi data pada *dashboard* bertujuan untuk membantu pengguna dalam memahami kondisi inventory secara cepat dan mempermudah proses monitoring terhadap pergerakan barang dalam sistem.

4.1.4. Implementasi Manajemen Data User



ID	Name	Email	Role	Action
1	Admin	admin@gmail.com	admin	[Edit] [Delete]
2	Bagus	bagusdp2011@gmail.com	admin	[Edit] [Delete]
3	Tester	tester@gmail.com	staff	[Edit] [Delete]
15	Test Edit 1	editor@gmail.com	admin	[Edit] [Delete]
18	Test Delete	deletor@gmail.com	staff	[Edit] [Delete]

Gambar 23 - Implementasi User Management

Modul manajemen data user digunakan oleh *Admin* untuk mengelola akun pengguna yang dapat mengakses sistem inventory. Pada halaman ini, *Admin* dapat menambahkan pengguna baru, melihat daftar pengguna yang telah terdaftar, memperbarui informasi pengguna, serta menghapus akun yang sudah tidak digunakan. Data pengguna yang disimpan dalam sistem meliputi informasi seperti nama pengguna, username, password, serta peran pengguna (role) seperti *Admin* atau *Staff*. Pengaturan role ini digunakan untuk membatasi hak akses pengguna terhadap fitur yang tersedia di dalam sistem. Dengan adanya modul manajemen data user ini, pengelolaan akses terhadap sistem dapat dilakukan secara lebih terstruktur sehingga meningkatkan keamanan dan kontrol penggunaan aplikasi.

4.1.5. Implementasi Manajemen Data Penjelasan Mesin

MTSC Singapore

Bagus
Online

Dashboard

System Users

ViTrox Products

Transactions

History



ViTrox Corporation Berhad

World Leading Automated Vision Inspection Solution Provider

2000
Founded

0097
Bursa Malaysia Stock Code

20+
Countries Served

Global
Customer Network

About ViTrox

Since its inception in 2000, ViTrox has been designing and manufacturing innovative, leading-edge, and cost-effective automated vision inspection equipment and embedded electronic solutions for the semiconductor and electronics industries.

The company provides advanced technologies that help manufacturers improve quality, productivity, and operational efficiency through automation and machine vision solutions.

Core Business Solutions

MVS
Machine Vision System for semiconductor inspection.

ABI
Automated Board Inspection for PCB manufacturing.

IIES
Integrated Industrial Embedded Solutions.

SPI
Automated Solder Paste Inspection systems that ensure solder paste quality and improve SMT process control before component placement.

AXI / X-Ray
Advanced Automated X-Ray Inspection solutions for detecting hidden solder defects, BGA issues, voids, and internal assembly faults.

Company Milestones

2000 – ViTrox was established.
2004 – Converted into a public limited company.
2004 – Listed on Bursa Malaysia.
Today – Serving customers worldwide across Asia, Europe, Australia, and the Americas.

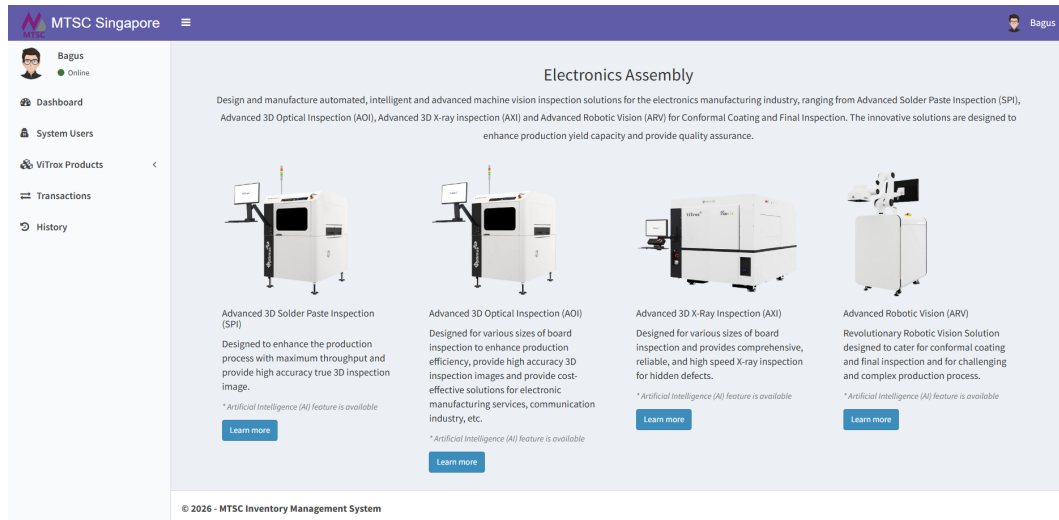
Global Presence

ViTrox serves customers from Malaysia, Singapore, Indonesia, Thailand, Vietnam, Philippines, Taiwan, China, Japan, Korea, India, Australia, Europe, Brazil, Mexico, the USA, and many other countries around the world.

[Visit Official Website](#)

© 2026 - MTSC Inventory Management System

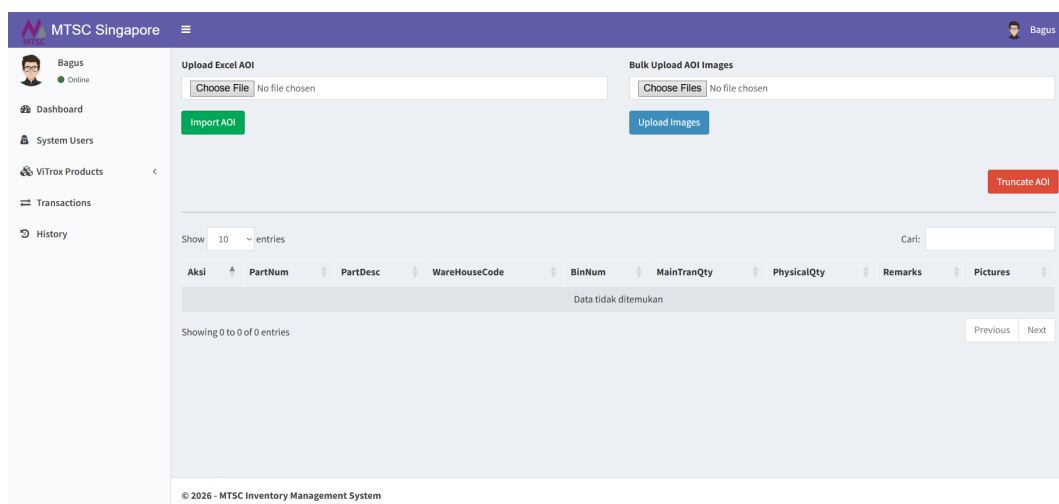
Gambar 24 - Implementasi Halaman Informasi Vitrox



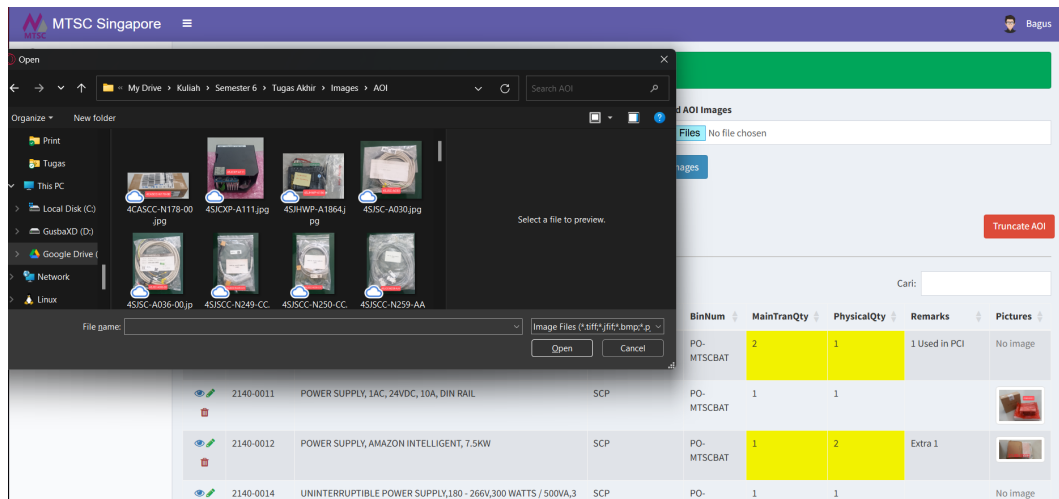
Gambar 25 - Implementasi Halaman Mesin dari Vitrox

Pada halaman ini, *Admin* dapat menambahkan data mesin baru, memperbarui informasi mesin, serta menghapus data mesin apabila sudah tidak digunakan. Informasi yang disimpan pada modul ini meliputi nama mesin, kode mesin, serta deskripsi atau keterangan terkait mesin tersebut. Dengan adanya fitur ini, pengguna dapat memperoleh informasi yang lebih jelas mengenai mesin yang berkaitan dengan penggunaan barang dalam sistem inventory.

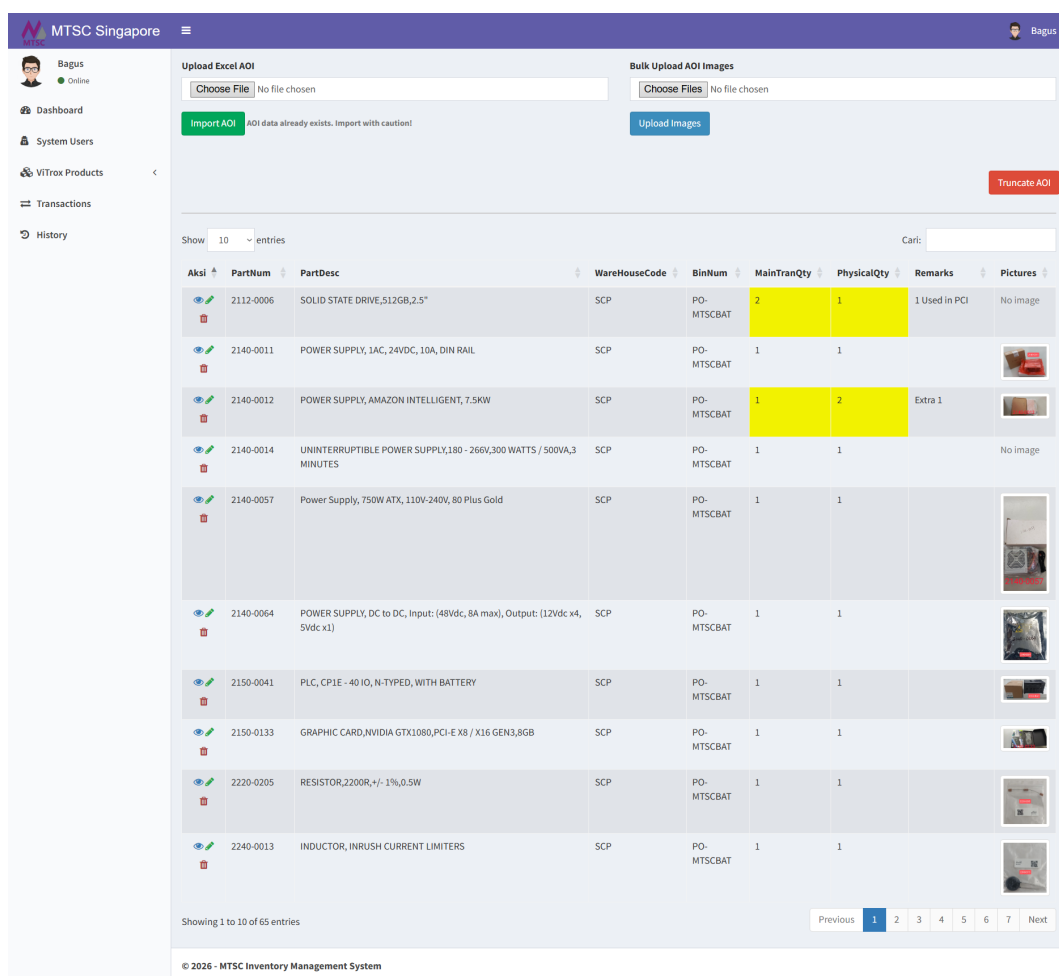
4.1.6. Implementasi Manajemen Data Pengelolaan Data



Gambar 26 - Implementasi Manajemen Data 1



Gambar 27 - Implementasi Manajemen Data 2



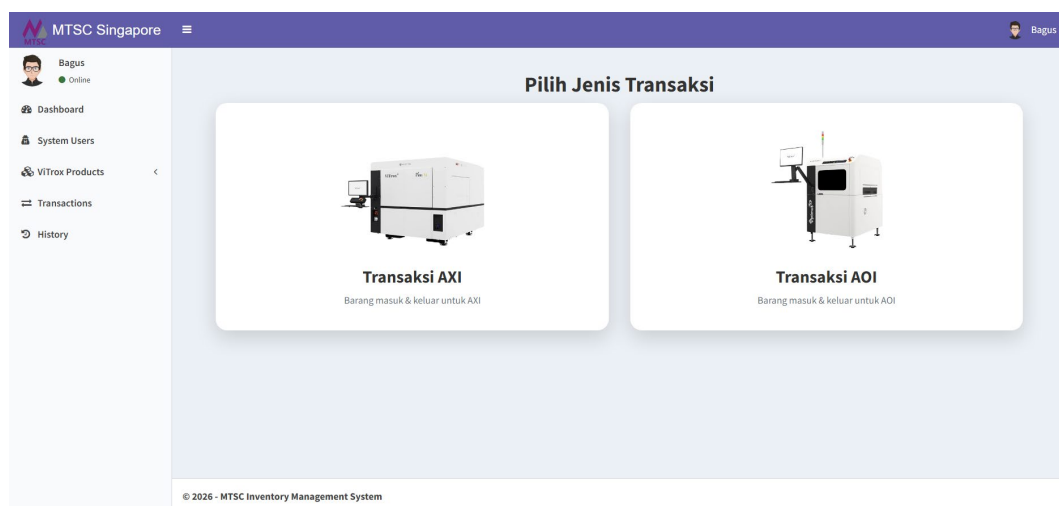
Gambar 28 - Implementasi Manajemen Data 3

Modul manajemen data barang digunakan oleh *Admin* untuk melakukan pengelolaan terhadap data master barang yang terdapat dalam sistem. Pada modul

ini, *Admin* dapat melakukan operasi Create, Read, Update, dan Delete (*CRUD*) terhadap data barang. Data barang yang disimpan dalam sistem meliputi informasi seperti kode barang, nama barang, kategori barang, serta jumlah stok yang tersedia. Setiap perubahan data akan langsung tersimpan pada basis data MySQL sehingga dapat diakses secara *real-time* oleh pengguna.

Aplikasi ini juga menyediakan fitur unggah data menggunakan file Excel untuk mempermudah proses input dan pembaruan data barang dalam jumlah besar. Admin dapat melakukan unggah file Excel secara berulang sesuai kebutuhan operasional tanpa perlu menghapus data yang telah ada sebelumnya. Sistem akan melakukan proses validasi berdasarkan part number sebagai identitas unik setiap barang. Apabila part number pada file Excel telah tersedia di dalam basis data, sistem akan secara otomatis memperbarui informasi barang yang bersangkutan, seperti nama barang, kategori, atau jumlah stok sesuai dengan data terbaru. Sebaliknya, apabila part number belum terdaftar, sistem akan menambahkan data tersebut sebagai barang baru. Dengan mekanisme ini, proses sinkronisasi data inventory dapat dilakukan dengan lebih cepat, akurat, dan efisien serta mengurangi risiko duplikasi data pada sistem.

4.1.7. Implementasi Transaksi Barang Masuk dan Keluar

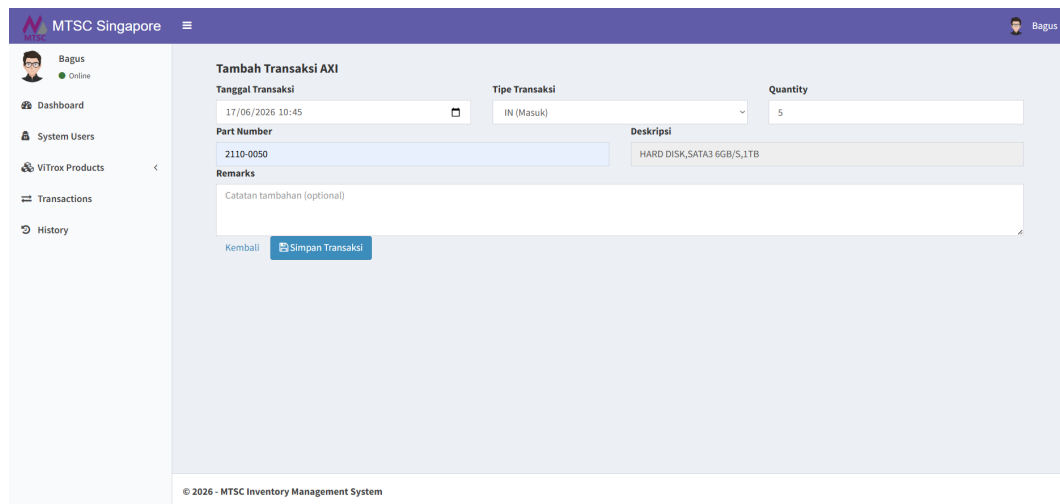


Gambar 29 - Implementasi Halaman Transaksi

Sistem inventory ini juga menyediakan fitur untuk mencatat transaksi barang masuk dan barang keluar dari gudang. Fitur ini memungkinkan *Admin* untuk mencatat setiap perubahan stok yang terjadi akibat aktivitas operasional. Ketika

transaksi barang masuk dicatat, maka jumlah stok barang akan bertambah secara otomatis. Sebaliknya, ketika transaksi barang keluar dicatat, maka jumlah stok barang akan berkurang sesuai dengan jumlah barang yang dikeluarkan. Setiap transaksi yang terjadi juga akan dicatat dalam history transaksi sehingga pengguna dapat melihat riwayat pergerakan barang secara lengkap.

4.1.8. Implementasi Transaksi Barang Masuk



MTSC Singapore

Bagus Online

Dashboard

System Users

WITrox Products

Transactions

History

Tambah Transaksi AXI

Tanggal Transaksi: 17/06/2026 10:45

Tipe Transaksi: IN (Masuk)

Quantity: 5

Part Number: 2110-0050

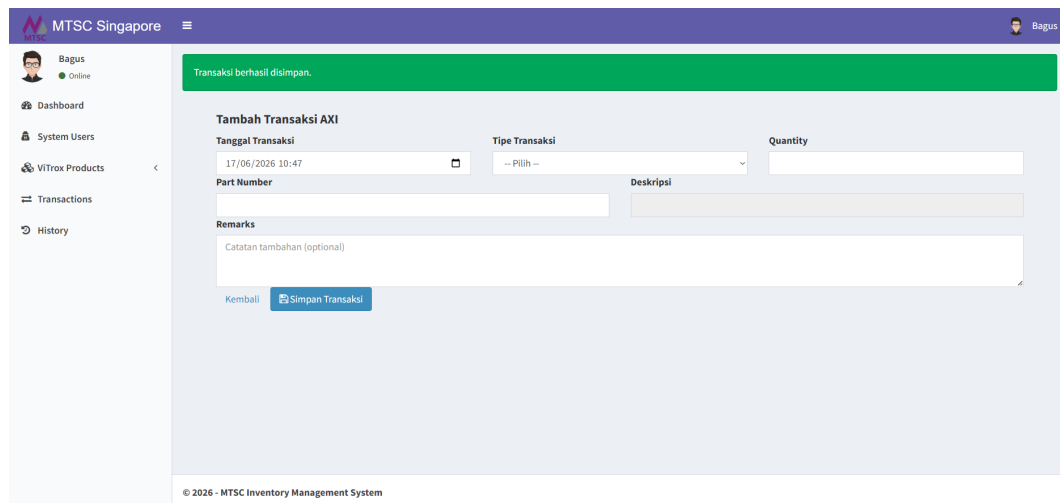
Deskripsi: HARD DISK,SATA3 6GB/S,1TB

Remarks: Catatan tambahan (optional)

Kembali Simpan Transaksi

© 2026 - MTSC Inventory Management System

Gambar 30 - Implementasi Transaksi Barang Masuk 1



MTSC Singapore

Bagus Online

Dashboard

System Users

WITrox Products

Transactions

History

Transaksi berhasil disimpan.

Tambah Transaksi AXI

Tanggal Transaksi: 17/06/2026 10:47

Tipe Transaksi: -- Pilih --

Quantity:

Part Number:

Deskripsi:

Remarks: Catatan tambahan (optional)

Kembali Simpan Transaksi

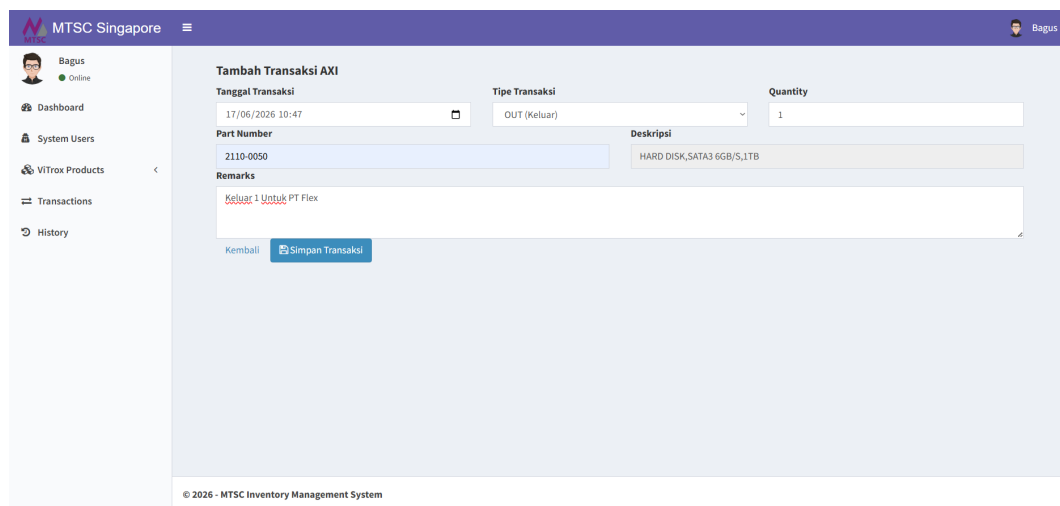
© 2026 - MTSC Inventory Management System

Gambar 31 - Implementasi Transaksi Barang Masuk 2

Halaman transaksi barang masuk digunakan untuk mencatat setiap barang yang masuk ke dalam gudang. Pada halaman ini, *Admin* dapat memasukkan data barang yang diterima beserta informasi pendukung seperti tanggal transaksi, jumlah

barang, serta keterangan tambahan jika diperlukan. Setelah data transaksi disimpan, sistem akan secara otomatis menambahkan jumlah stok barang yang bersangkutan pada basis data. Hal ini memastikan bahwa informasi stok yang ditampilkan dalam sistem selalu diperbarui secara otomatis sesuai dengan transaksi yang terjadi. Fitur ini membantu pengguna dalam melakukan pencatatan barang yang masuk secara sistematis sehingga mempermudah proses pengelolaan dan pengawasan stok barang di dalam gudang.

4.1.9. Implementasi Transaksi Barang Keluar

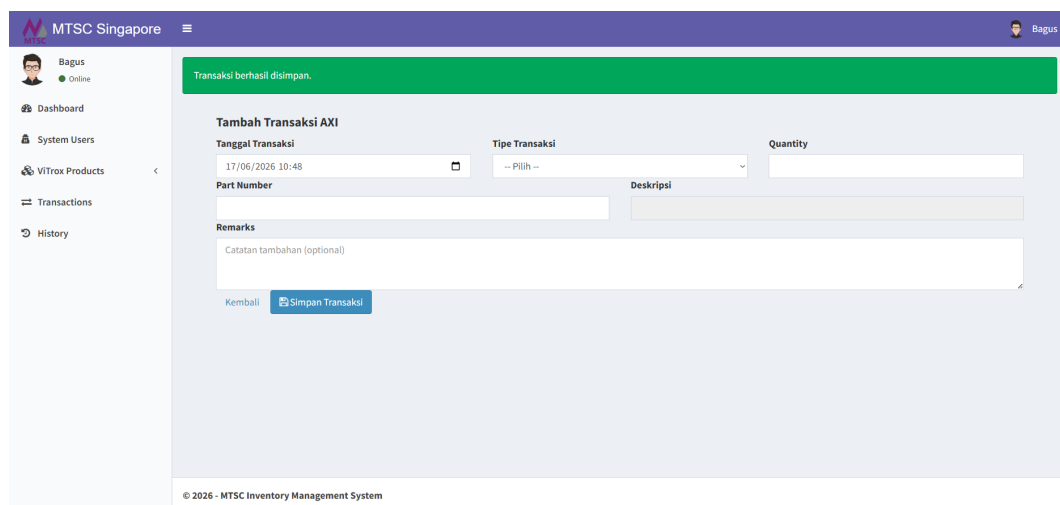


The screenshot shows the 'Tambah Transaksi AXI' (Add Transaction AXI) form in the MTSC Singapore web application. The form is titled 'Tambah Transaksi AXI' and includes the following fields and options:

- Tanggal Transaksi**: 17/06/2026 10:47
- Tipe Transaksi**: OUT (Keluar)
- Quantity**: 1
- Part Number**: 2110-0050
- Deskripsi**: HARD DISK,SATA3 6GB/S,1TB
- Remarks**: Keluar 1 Untuk PT Flex

At the bottom of the form, there are two buttons: 'Kembali' (Back) and 'Simpan Transaksi' (Save Transaction). The footer of the page indicates '© 2026 - MTSC Inventory Management System'.

Gambar 32 - Implementasi Transaksi Barang Keluar 1



The screenshot shows the 'Tambah Transaksi AXI' form in the MTSC Singapore web application after a successful transaction save. A green banner at the top of the form area displays the message 'Transaksi berhasil disimpan.' (Transaction successfully saved). The form fields are now empty, and the 'Tipe Transaksi' dropdown is set to '-- Pilih --' (Select). The 'Remarks' field contains the placeholder text 'Catatan tambahan (optional)' (Additional notes (optional)). The 'Simpan Transaksi' button is still visible at the bottom of the form. The footer of the page indicates '© 2026 - MTSC Inventory Management System'.

Gambar 33 - Implementasi Transaksi Barang Keluar 2

Halaman transaksi barang keluar digunakan untuk mencatat barang yang dikeluarkan dari gudang. Pada halaman ini, *Admin* dapat memilih barang yang akan dikeluarkan, memasukkan jumlah barang yang diambil, serta mencatat tanggal transaksi dan keterangan tambahan. Setelah transaksi disimpan, sistem akan secara otomatis mengurangi jumlah stok barang yang tersedia sesuai dengan jumlah barang yang dikeluarkan. Proses ini dilakukan secara otomatis oleh sistem sehingga meminimalkan kesalahan dalam perhitungan stok. Dengan adanya fitur ini, proses pencatatan barang keluar dan masuk dapat dilakukan dengan lebih mudah dan terorganisir, serta mempermudah pengguna dalam memantau pergerakan barang dalam sistem inventory.

4.1.10. Implementasi Validasi Stok Tidak Mencukupi di Transaksi

The screenshot displays the 'Tambah Transaksi AXI' (Add Transaction AXI) form in the MTSC Singapore system. The form includes fields for 'Tanggal Transaksi' (Transaction Date) set to 17/06/2026 10:49, 'Tipe Transaksi' (Transaction Type) set to 'OUT (Keluar)', and 'Quantity' set to 100. The 'Part Number' is 2110-0050, and the 'Deskripsi' (Description) is 'HARD DISK,SATA3 6GB/S,1TB'. The 'Remarks' field contains the text 'Validasi Tidak Cukup Stok' (Stock Validation Not Enough) in red. Below the form, there are buttons for 'Kembali' (Back) and 'Simpan Transaksi' (Save Transaction). The footer indicates '© 2026 - MTSC Inventory Management System'.

Gambar 34 - Implementasi Validasi Stok di Transaksi 1

The screenshot displays the MTSC Singapore Inventory Management System interface. At the top, a red banner indicates a stock validation error: "Stok tidak mencukupi." (Stock is insufficient). Below this, the "Tambah Transaksi AXI" (Add Transaction AXI) form is shown. The form includes fields for "Tanggal Transaksi" (Transaction Date) set to "17/06/2026 10:50", "Tipe Transaksi" (Transaction Type) with a dropdown menu, and "Quantity". There are also fields for "Part Number", "Deskripsi" (Description), and "Remarks" with a placeholder "Catatan tambahan (optional)". At the bottom of the form, there are two buttons: "Kembali" (Back) and "Simpan Transaksi" (Save Transaction). The left sidebar shows the user "Bagus" is online and lists navigation options: Dashboard, System Users, VITrox Products, Transactions, and History. The footer indicates the copyright is "© 2026 - MTSC Inventory Management System".

Gambar 35 - Implementasi Validasi Stok di Transaksi 2

Sistem inventory ini juga dilengkapi dengan mekanisme validasi untuk mencegah terjadinya transaksi barang keluar ketika jumlah stok barang tidak mencukupi. Ketika pengguna mencoba melakukan transaksi barang keluar dengan jumlah yang melebihi stok yang tersedia, sistem akan menampilkan pesan peringatan bahwa stok barang tidak mencukupi dan transaksi tidak dapat diproses. Fitur validasi ini bertujuan untuk menjaga keakuratan data inventory serta mencegah terjadinya kesalahan pencatatan yang dapat menyebabkan ketidaksesuaian antara data sistem dan kondisi stok barang yang sebenarnya.

4.1.11. Implementasi History Pencatatan Barang

MTSC Singapore

Bagus Online

Dashboard

System Users

VITrox Products

Transactions

History

History Transaksi Inventory

Export PDF Export Excel Copy

Cari:

#	Tanggal	Machine	Part Number	Deskripsi	Type	Qty	UOM	Warehouse	Bin	Ref Type	Ref ID	Remarks	Created By
13	17-06-2026 10:47	AXI	2110-0050	HARD DISK,SATA3 6GB/S,1TB	OUT	1	pcs	Batam	PX-MTSCBAT	AXI	2110-0050	Keluar 1 Untuk PT Flex	Bagus
12	17-06-2026 10:45	AXI	2110-0050	HARD DISK,SATA3 6GB/S,1TB	IN	5	pcs	Batam	PX-MTSCBAT	AXI	2110-0050		Bagus
11	16-06-2026 13:58	AXI	2110-0050	HARD DISK,SATA3 6GB/S,1TB	IN	555	pcs	Batam	PX-MTSCBAT	AXI	2110-0050	Tambah Transaksi	Bagus
10	15-06-2026 22:05	AXI	2110-0050	HARD DISK,SATA3 6GB/S,1TB	IN	100	pcs	Batam	PX-MTSCBAT	AXI	2110-0050		Bagus
9	15-06-2026 20:21	AOI	2112-0006	SOLID STATE DRIVE,512GB,2.5"	IN	100	pcs	SCP	PO-MTSCBAT	AOI	2112-0006	Tambahann Baru	Bagus
8	08-06-2026 13:49	AXI	2110-0050	Tambahan 10 untuk 2110-0050	IN	10	pcs	Batam	AXI	AXI	2110-0050	2110-0050 + 10	Bagus
7	04-05-2026 12:19	AOI	1234-Kecil	Testing masukkan data	OUT	5	pcs	Batam	AOI	AOI	1234-Kecil	Testing data	Bagus
6	04-05-2026 12:18	AOI	1234-Kecil	Testing masukkan data	IN	50	pcs	Batam	AOI	AOI	1234-Kecil	Testing data	Bagus
5	15-04-2026 21:51	AOI	2150-0133	Nvidia Card	IN	2	pcs	Batam	AOI	AOI	2150-0133	Akan ditambah kedalam warehouse	Bagus
4	15-03-2026 09:30	AXI	2110-0050	Tambahan item	IN	5	pcs	Batam	AXI	AXI	2110-0050	Tambah item	Bagus

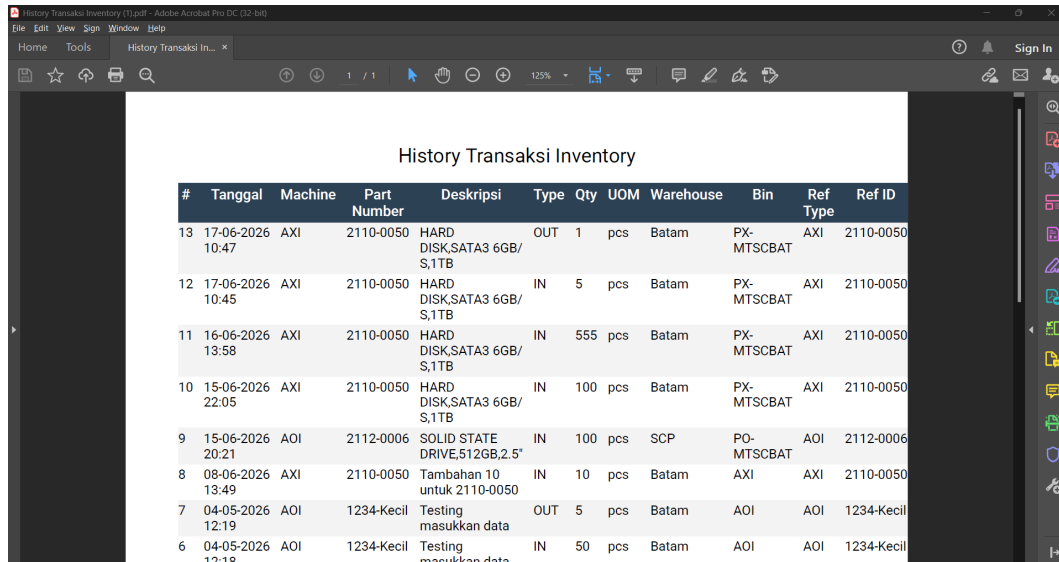
Menampilkan 1 sampai 10 dari 13 data

© 2026 - MTSC Inventory Management System

Gambar 36 - Implementasi Halaman History Pencatatan Barang

Modul history pencatatan barang digunakan untuk menampilkan riwayat seluruh transaksi yang terjadi dalam sistem inventory, baik transaksi barang masuk maupun barang keluar. Pada halaman ini, pengguna dapat melihat informasi transaksi secara lengkap seperti nama barang, jenis transaksi, jumlah barang, tanggal transaksi, serta pengguna yang melakukan transaksi tersebut. Dengan adanya fitur history ini, pengguna dapat melakukan penelusuran terhadap pergerakan barang secara lebih transparan dan terstruktur. Selain itu, data riwayat transaksi juga dapat digunakan sebagai bahan evaluasi dalam pengelolaan inventory di masa mendatang.

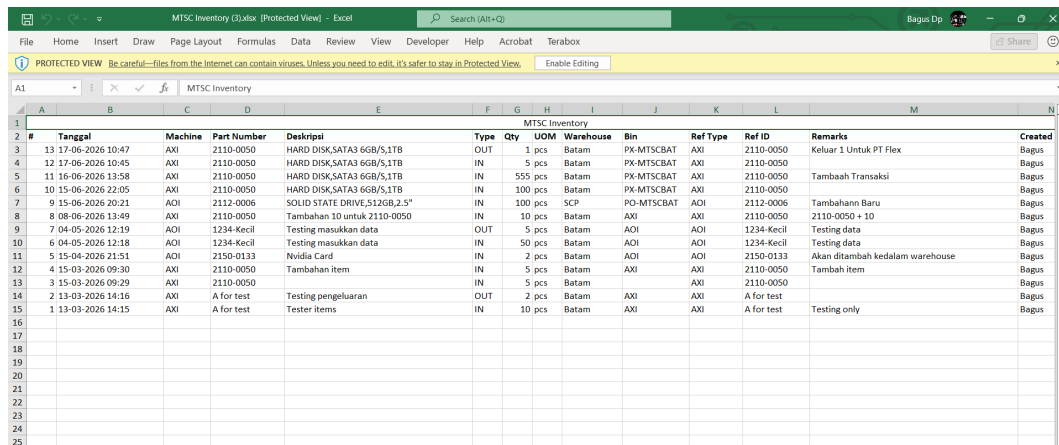
4.1.12. Implementasi Modul Laporan *PDF* dan Excel



History Transaksi Inventory

#	Tanggal	Machine	Part Number	Deskripsi	Type	Qty	UOM	Warehouse	Bin	Ref Type	Ref ID
13	17-06-2026 10:47	AXI	2110-0050	HARD DISK,SATA3 6GB/S,1TB	OUT	1	pcs	Batam	PX-MTSCBAT	AXI	2110-0050
12	17-06-2026 10:45	AXI	2110-0050	HARD DISK,SATA3 6GB/S,1TB	IN	5	pcs	Batam	PX-MTSCBAT	AXI	2110-0050
11	16-06-2026 13:58	AXI	2110-0050	HARD DISK,SATA3 6GB/S,1TB	IN	555	pcs	Batam	PX-MTSCBAT	AXI	2110-0050
10	15-06-2026 22:05	AXI	2110-0050	HARD DISK,SATA3 6GB/S,1TB	IN	100	pcs	Batam	PX-MTSCBAT	AXI	2110-0050
9	15-06-2026 20:21	AOI	2112-0006	SOLID STATE DRIVE,512GB,2.5"	IN	100	pcs	SCP	PO-MTSCBAT	AOI	2112-0006
8	08-06-2026 13:49	AXI	2110-0050	Tambahan 10 untuk 2110-0050	IN	10	pcs	Batam	AXI	AXI	2110-0050
7	04-05-2026 12:19	AOI	1234-Kecil	Testing masukkan data	OUT	5	pcs	Batam	AOI	AOI	1234-Kecil
6	04-05-2026 17:18	AOI	1234-Kecil	Testing masukkan data	IN	50	pcs	Batam	AOI	AOI	1234-Kecil

Gambar 37 - Implementasi Ekspor PDF



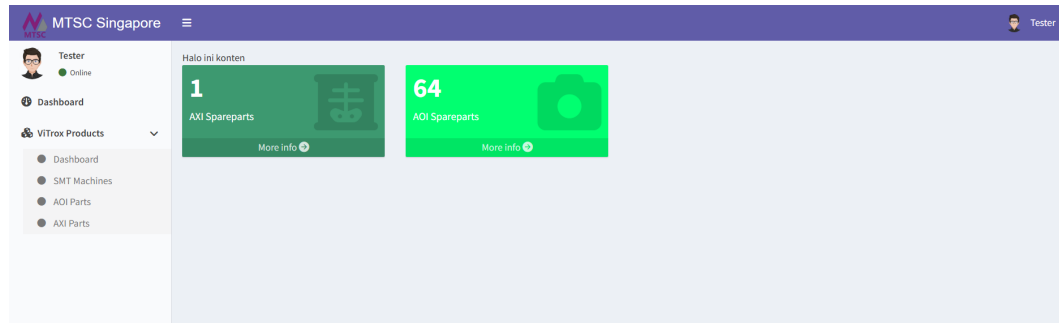
MTSC Inventory

#	Tanggal	Machine	Part Number	Deskripsi	Type	Qty	UOM	Warehouse	Bin	Ref Type	Ref ID	Remarks	Created
3	13 17-06-2026 10:47	AXI	2110-0050	HARD DISK,SATA3 6GB/S,1TB	OUT	1	pcs	Batam	PX-MTSCBAT	AXI	2110-0050	Keluar 1 Untuk PT Flex	Bagus
4	12 17-06-2026 10:45	AXI	2110-0050	HARD DISK,SATA3 6GB/S,1TB	IN	5	pcs	Batam	PX-MTSCBAT	AXI	2110-0050		Bagus
5	11 16-06-2026 13:58	AXI	2110-0050	HARD DISK,SATA3 6GB/S,1TB	IN	555	pcs	Batam	PX-MTSCBAT	AXI	2110-0050	Tambah Transaksi	Bagus
6	10 15-06-2026 22:05	AXI	2110-0050	HARD DISK,SATA3 6GB/S,1TB	IN	100	pcs	Batam	PX-MTSCBAT	AXI	2110-0050		Bagus
7	9 15-06-2026 20:21	AOI	2112-0006	SOLID STATE DRIVE,512GB,2.5"	IN	100	pcs	SCP	PO-MTSCBAT	AOI	2112-0006	Tambahann Baru	Bagus
8	8 08-06-2026 13:49	AXI	2110-0050	Tambahan 10 untuk 2110-0050	IN	10	pcs	Batam	AXI	AXI	2110-0050	2110-0050 + 10	Bagus
9	7 04-05-2026 12:19	AOI	1234-Kecil	Testing masukkan data	OUT	5	pcs	Batam	AOI	AOI	1234-Kecil	Testing data	Bagus
10	6 04-05-2026 12:18	AOI	1234-Kecil	Testing masukkan data	IN	50	pcs	Batam	AOI	AOI	1234-Kecil	Testing data	Bagus
11	5 15-04-2026 21:51	AOI	2150-0133	Nvidia Card	IN	2	pcs	Batam	AOI	AOI	2150-0133	Akan ditambah kedalam warehouse	Bagus
12	4 15-03-2026 09:30	AXI	2110-0050	Tambahan item	IN	5	pcs	Batam	AXI	AXI	2110-0050	Tambah item	Bagus
13	3 15-03-2026 09:29	AXI	2110-0050		IN	5	pcs	Batam	AXI	AXI	2110-0050		Bagus
14	2 13-03-2026 14:16	AXI	A for test	Testing pengeluaran	OUT	2	pcs	Batam	AXI	AXI	A for test	A for test	Bagus
15	1 13-03-2026 14:15	AXI	A for test	Tester items	IN	10	pcs	Batam	AXI	AXI	A for test	Testing only	Bagus

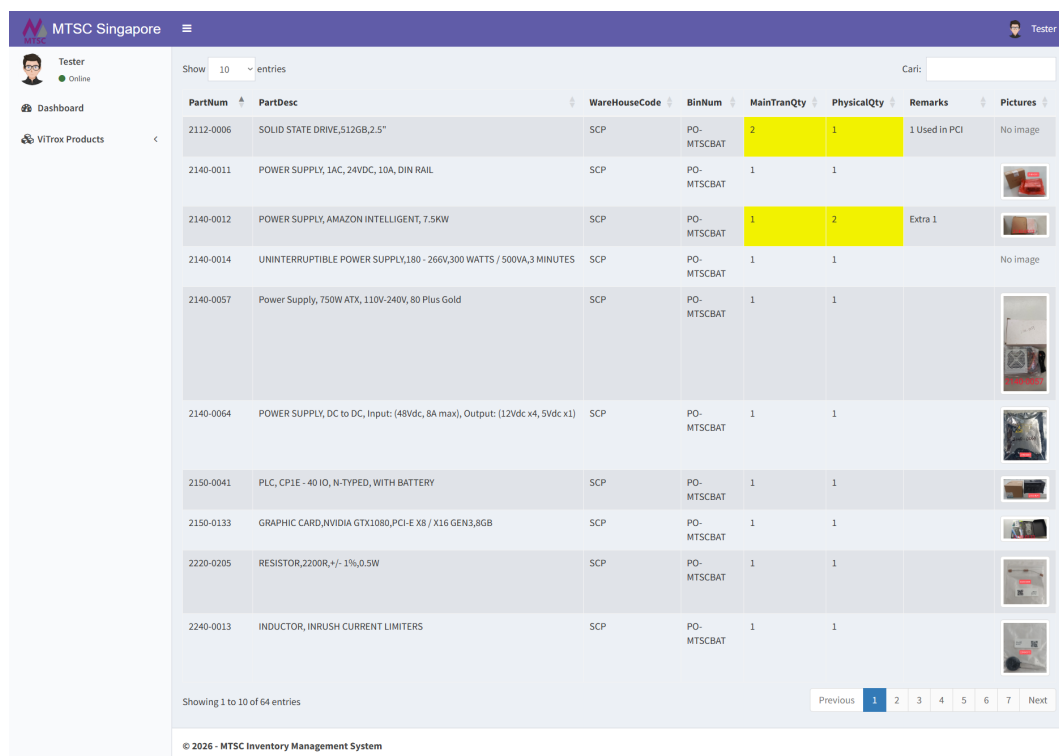
Gambar 38 - Implementasi Ekspor Excel

Modul laporan digunakan untuk menghasilkan laporan transaksi inventory berdasarkan periode tertentu, seperti laporan harian, mingguan, maupun bulanan. Pengguna dapat melihat laporan transaksi secara langsung melalui sistem maupun mengekspor laporan tersebut ke dalam format *PDF* atau Excel. Fitur ini mempermudah proses dokumentasi dan distribusi laporan kepada pihak manajemen. Dengan adanya modul laporan ini, pengguna dapat melakukan monitoring terhadap aktivitas inventory secara lebih terstruktur dan sistematis.

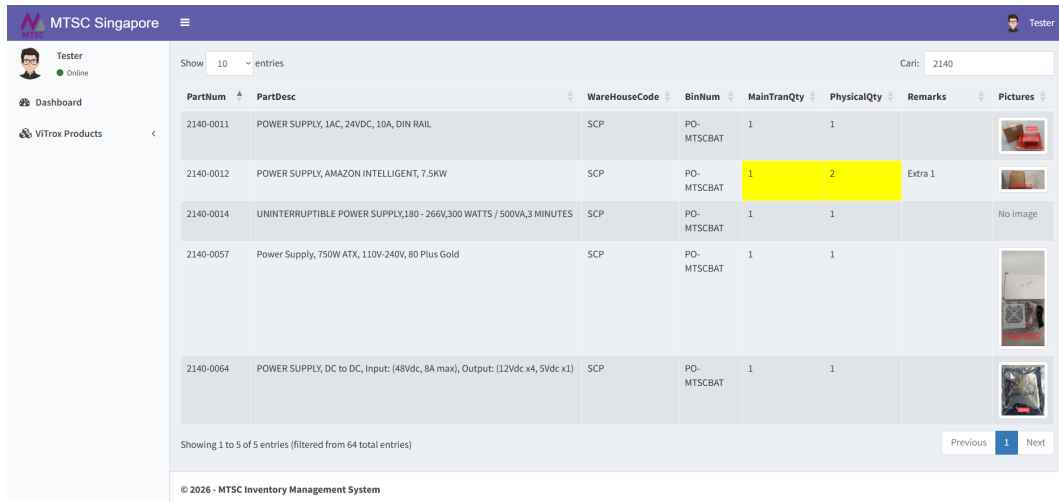
4.1.13. Implementasi *Dashboard* Sistem Sebagai *Staff*



Gambar 39 - Implementasi Untuk User Staff 1



Gambar 40 - Implementasi Untuk User Staff 2



PartNum	PartDesc	WarehouseCode	BinNum	MainTranQty	PhysicalQty	Remarks	Pictures
2140-0011	POWER SUPPLY, 1AC, 24VDC, 10A, DIN RAIL	SCP	PO-MTSCBAT	1	1		
2140-0012	POWER SUPPLY, AMAZON INTELLIGENT, 7.5KW	SCP	PO-MTSCBAT	1	2	Extra 1	
2140-0014	UNINTERRUPTIBLE POWER SUPPLY, 180 - 266V, 300 WATTS / 500VA, 3 MINUTES	SCP	PO-MTSCBAT	1	1		No image
2140-0057	Power Supply, 750W ATX, 110V-240V, 80 Plus Gold	SCP	PO-MTSCBAT	1	1		
2140-0064	POWER SUPPLY, DC to DC, Input: (48Vdc, 8A max), Output: (12Vdc x4, 5Vdc x1)	SCP	PO-MTSCBAT	1	1		

Showing 1 to 5 of 5 entries (filtered from 64 total entries)

© 2026 - MTSC Inventory Management System

Gambar 41 - Implementasi Untuk User Staff 3

Dashboard Staff merupakan halaman utama yang ditampilkan setelah pengguna dengan peran *Staff* berhasil melakukan proses login ke dalam sistem. Halaman ini dirancang untuk memberikan informasi yang relevan sesuai dengan tugas dan tanggung jawab *Staff* dalam mengelola data inventory. Pada *dashboard* ini, *Staff* dapat melihat ringkasan informasi terkait data barang yang tersedia, transaksi barang masuk dan keluar, serta akses menuju fitur-fitur yang diperbolehkan bagi *Staff*. Berbeda dengan *dashboard Admin*, *dashboard Staff* memiliki batasan akses tertentu sehingga *Staff* hanya dapat menggunakan fitur yang berkaitan dengan aktivitas operasional sehari-hari. Tampilan *dashboard* disusun secara sederhana dan informatif agar memudahkan *Staff* dalam melakukan navigasi serta memantau kondisi inventory secara cepat. Dengan adanya *dashboard* khusus untuk *Staff*, sistem dapat membantu meningkatkan efisiensi kerja serta meminimalkan kesalahan dalam pengelolaan data barang.

4.2 Pengujian

Tahap pengujian dilakukan untuk memastikan bahwa sistem inventory yang telah dikembangkan dapat berjalan dengan baik serta sesuai dengan kebutuhan pengguna. Pengujian dilakukan menggunakan metode Black Box Testing dan Unit Testing.

4.2.1. Pengujian *Black Box Testing*

Black Box Testing merupakan metode pengujian perangkat lunak yang dilakukan dengan cara menguji fungsi sistem tanpa melihat kode program yang digunakan. Pengujian ini berfokus pada kesesuaian antara input yang diberikan dengan *output* yang dihasilkan oleh sistem.

Pengujian dilakukan pada setiap fitur utama sistem berdasarkan kebutuhan fungsional yang telah ditentukan sebelumnya. Beberapa fitur yang diuji antara lain fitur login, pengelolaan data barang, pencatatan transaksi barang masuk dan keluar, *upload* data Excel, serta fitur laporan.

Berdasarkan hasil pengujian yang dilakukan, seluruh fungsi utama sistem dapat berjalan dengan baik sesuai dengan *skenario* yang telah ditentukan. Sistem mampu memproses input data dengan benar serta menghasilkan *output* yang sesuai dengan kebutuhan pengguna.

Kode Uji	Deskripsi Uji (Berdasarkan Kebutuhan Fungsional)	<i>Skenario</i> Pengujian	<i>Input / Data Uji</i>	<i>Output yang Diharapkan</i>	Status (Berhasil/Gagal)
FR001	Halaman <i>login</i> untuk <i>Admin & Staff</i> Member	Pengguna memasukkan <i>Username & password</i> yang valid	<i>Username & password</i> benar	Sistem menampilkan <i>dashboard</i> sesuai <i>role</i>	Berhasil
		Pengguna memasukkan <i>Username/password</i> salah	<i>Username/password</i> salah	Sistem menampilkan pesan " <i>Login gagal</i> "	Berhasil
FR002	Fitur <i>logout</i> untuk <i>Admin & Staff</i> Member	Pengguna menekan tombol <i>logout</i>	Klik tombol <i>logout</i>	Sistem mengakhiri sesi dan kembali ke	Berhasil

				halaman <i>login</i>	
FR003	Kelola <i>User & role</i>	<i>Admin</i> menambah <i>User</i> baru	Form tambah <i>User</i> diisi lengkap	Data <i>User</i> tersimpan di basis data & muncul di daftar <i>User</i>	Berhasil
		<i>Admin</i> mengubah data <i>User</i>	Edit data <i>User</i> lalu simpan	Data <i>User</i> terupdate di basis data	Berhasil
		<i>Admin</i> menghapus <i>User</i>	Klik tombol hapus pada <i>User</i>	Data <i>User</i> terhapus dari basis data	Berhasil
FR004	<i>Role-based access control</i>	<i>Admin</i> login ke sistem	Akun <i>Admin</i>	Sistem menampilkan semua menu <i>Admin</i>	Berhasil
		<i>Member</i> login ke sistem	Akun <i>Staff</i>	Sistem hanya menampilkan menu khusus <i>Staff</i>	Berhasil
FR005	<i>CRUD</i> master data barang	<i>Admin</i> menambah data barang baru	Isi form barang lengkap	Data barang tersimpan dan tampil di tabel	Berhasil melalui Transaksi
		<i>Admin</i> mengubah data barang	Edit data barang lalu simpan	Data barang berubah di tabel	Berhasil
		<i>Admin</i> menghapus data barang	Klik tombol hapus barang	Data barang hilang dari tabel	Berhasil

FR006	Mencatat barang masuk	<i>Admin input data</i> barang masuk	Isi form barang masuk	Data transaksi barang masuk tercatat di basis data	Berhasil
FR007	Mencatat barang keluar	<i>Admin input data</i> barang keluar	Isi form barang keluar	Data transaksi keluar tersimpan & stok berkurang otomatis	Berhasil
FR008	Menampilkan stok <i>real-time</i>	Pengguna membuka menu stok	-	Sistem menampilkan stok terbaru berdasarkan transaksi	Berhasil
FR009	<i>History</i> transaksi keluar/masuk	Pengguna membuka menu <i>history</i>	-	Sistem menampilkan daftar transaksi lengkap dengan tanggal dan jenis transaksi	Berhasil
FR010	<i>Upload</i> data massal via <i>Excel</i>	<i>Admin</i> memilih file <i>Excel</i> valid	File .xlsx berisi data barang	Sistem membaca file dan menambah	Berhasil

				data ke basis data	
		<i>Admin</i> mengunggah file dengan format salah	File .txt atau rusak	Sistem menolak file & menampilkan pesan error	Berhasil, dilimit hanya file xls/xlsx
FR011	Validasi otomatis data <i>Excel</i>	<i>Admin upload</i> data <i>Excel</i> dengan item duplikat	File <i>Excel</i> berisi kode barang ganda	Sistem menampilkan pesan validasi error	Berhasil, data duplikat akan di replace otomatis.
FR012	Peringatan duplikasi / inkonsistensi	<i>Admin input</i> data yang sudah ada	Kode barang duplikat	Sistem menampilkan pesan peringatan duplikasi	Berhasil, dilimit berdasarkan Part Number
FR013	Cetak laporan transaksi	Pengguna memilih periode laporan	Pilih “Laporan Bulanan”	Sistem menampilkan & mencetak laporan transaksi	Berhasil
FR014	Ekspor laporan ke <i>PDF / Excel</i>	Pengguna klik tombol ekspor	Pilih format <i>PDF/Excel</i>	Sistem menghasilkan file laporan sesuai format	Berhasil
FR015	<i>Dashboard</i> visualisasi data	Pengguna membuka <i>dashboard</i>	-	Sistem menampilkan grafik stok masuk, keluar, total barang	Berhasil

FR016	Pencarian berdasarkan kode/nama barang	Pengguna memasukkan kata kunci	“2140-0011” [AOI] / “2140-0023” [AXI]	Sistem menampilkan hasil pencarian sesuai kata kunci	Berhasil
FR017	<i>Filter</i> data	Pengguna memilih <i>filter</i> tertentu	<i>Filter</i> berdasarkan kategori/tanggal	Sistem menampilkan data sesuai <i>filter</i>	Berhasil
FR018	Akses landing page <i>Guest</i>	<i>Guest</i> membuka halaman utama	URL sistem diakses tanpa <i>login</i>	Sistem menampilkan landing page umum	Berhasil

4.2.2. Pengujian *Unit Testing*

```

PS C:\Users\ASUS\Herd\mtscsg> php artisan test
Warning: TTY mode is not supported on Windows platform.

PASS Tests\Feature\SystemFeatureTest
✓ fr001 admin can login successfully
✓ fr001 login fails with wrong password
✓ fr002 user can logout
✓ fr003 admin can create user
✓ fr004 staff cannot access admin page
✓ fr006 stock increases when item in
✓ fr007 stock reduces when item out
✓ fr008 out fails when stock not enough
✓ fr010 admin can upload valid excel
✓ fr011 add duplicate validation
✓ fr018 guest can access landing page

PASS Tests\Unit\ExampleTest
✓ that true is true

PASS Tests\Feature\ExampleTest
✓ it returns a successful response

Tests: 13 passed
Time: 2.34s

PS C:\Users\ASUS\Herd\mtscsg>

```

Pada tahap pengujian sistem, dilakukan pengujian menggunakan fitur *unit testing* yang disediakan oleh framework Laravel. Pengujian ini bertujuan untuk

memastikan bahwa setiap fungsi utama pada sistem berjalan sesuai dengan kebutuhan fungsional yang telah dirancang pada tahap analisis sistem.

Proses pengujian dilakukan dengan menjalankan perintah berikut pada terminal:

php artisan test

Perintah tersebut akan menjalankan seluruh *skenario* pengujian yang telah dibuat pada folder `tests/Feature`. Dalam penelitian ini, pengujian dilakukan menggunakan kelas `SystemFeatureTest` yang berisi beberapa *skenario* pengujian berdasarkan Functional Requirement (FR) yang telah ditentukan sebelumnya.

Berdasarkan hasil eksekusi pengujian, seluruh *skenario* pengujian yang dijalankan berhasil memperoleh status `passed`, yang menunjukkan bahwa fungsi-fungsi utama pada sistem telah berjalan sesuai dengan yang diharapkan.

[1]. Test | FR001 - Login Berhasil

```
public function test_fr001_admin_can_login_successfully()
{
    // FR001 - Halaman login untuk Admin & MTSC Staff
    $user = User::factory()->create([
        'name' => 'Robot Testing',
        'email' => 'robottest@gmail.com',
        'password' => bcrypt('password123'),
        'role' => 'admin'
    ]);

    $response = $this->post('/', [
        'email' => $user->email,
        'password' => 'password123'
    ]);

    $response->assertRedirect('/admin/dashboard');
    $this->assertAuthenticated();
}
```

Pengujian ini bertujuan untuk memastikan bahwa pengguna dengan kredensial yang valid dapat berhasil melakukan login ke dalam sistem. Pada pengujian ini, sistem terlebih dahulu membuat data pengguna dengan peran *Admin* menggunakan `factory`. Setelah itu dilakukan simulasi login dengan email dan password yang benar. Hasil yang diharapkan dari pengujian ini adalah sistem melakukan redirect ke halaman *dashboard Admin* dan status autentikasi pengguna menjadi aktif.

[2]. Test | FR001 - Login Gagal

```
public function test_fr001_login_fails_with_wrong_password()
{
    // FR001 - Login gagal jika password salah
    $user = User::factory()->create([
        'name' => 'Robot Testing',
        'email' => 'robottest@gmail.com',
        'password' => bcrypt('password123'),
        'role' => 'admin'
    ]);

    $response = $this->post('/', [
        'email' => $user->email,
        'password' => 'wrongpassword'
    ]);
    $response->assertRedirect();
    $response->assertSessionHasErrors(['email']);
    $this->assertGuest();
}
```

Pengujian ini dilakukan untuk memastikan bahwa sistem dapat menolak proses login ketika pengguna memasukkan password yang salah. Pada *skenario* ini dibuat sebuah akun pengguna, kemudian dilakukan percobaan login dengan password yang tidak sesuai. Hasil yang diharapkan adalah sistem menolak login, menampilkan pesan kesalahan pada session, dan pengguna tetap berada dalam kondisi belum terautentikasi.

[3]. Test | FR002 – Logout

```
public function test_fr002_user_can_logout()
{
    // FR002 - Fitur logout
    /** @var \Illuminate\Contracts\Auth\Authenticatable $userData */
    $userData = User::factory()->create([
        'name' => 'Robot Testing',
        'email' => 'robottest@gmail.com',
        'password' => bcrypt('password123'),
        'role' => 'admin'
    ]);
    $response = $this->actingAs($userData)->post('/logout');

    $response->assertRedirect('/');
    $this->assertGuest();
}
```

Pengujian ini bertujuan untuk memastikan bahwa pengguna yang telah login dapat keluar dari sistem dengan menggunakan fitur logout. Pada pengujian ini, sistem mensimulasikan pengguna yang telah login kemudian mengakses endpoint

logout. Hasil yang diharapkan adalah sistem melakukan redirect ke halaman utama dan status autentikasi pengguna berubah menjadi *Guest*.

[4]. Test | FR003 - *Admin* Tambah User

```
public function test_fr003_admin_can_create_user()
{
    // FR003 - Kelola User & Role (Tambah User)
    /** @var \Illuminate\Contracts\Auth\Authenticatable $admin */
    $admin = User::factory()->create(['role' => 'admin']);
    $response = $this->actingAs($admin)->post('/admin/register', [
        'name' => 'Test User',
        'email' => 'testuser@mail.com',
        'password' => 'password123',
        'password_confirmation' => 'password123',
        'role' => 'staff'
    ]);

    $response->assertStatus(302);
    $this->assertDatabaseHas('users', [
        'email' => 'testuser@mail.com'
    ]);
}
```

Pengujian ini bertujuan untuk memastikan bahwa *Admin* dapat menambahkan pengguna baru ke dalam sistem. Pada *skenario* ini, sistem membuat akun *Admin* terlebih dahulu, kemudian melakukan simulasi pengiriman data pendaftaran pengguna baru melalui endpoint yang tersedia. Hasil yang diharapkan adalah sistem berhasil menyimpan data pengguna baru ke dalam *database* dan melakukan redirect setelah proses penyimpanan berhasil.

[5]. Test | FR004 - Role Based Access Control

```
public function test_fr004_staff_cannot_access_admin_page()
{
    // FR004 - Staff hanya bisa akses menu staff
    /** @var \Illuminate\Contracts\Auth\Authenticatable $staff */
    $staff = User::factory()->create(['role' => 'staff']);

    $response = $this->actingAs($staff)
        ->get('/admin/user');

    $response->assertStatus(403);
}
```

Pengujian ini bertujuan untuk memastikan bahwa sistem menerapkan pembatasan akses berdasarkan peran pengguna (Role Based Access Control). Pada

skenario ini dibuat akun dengan peran *Staff*, kemudian dilakukan percobaan untuk mengakses halaman yang hanya diperuntukkan bagi *Admin*. Hasil yang diharapkan adalah sistem menolak akses tersebut dan memberikan status HTTP 403 (Forbidden).

[6]. Test | FR006 - CRUD Master Data Barang (Create)

```
public function test_fr006_stock_increases_when_item_in()
{
    /** @var \Illuminate\Contracts\Auth\Authenticated $admin */
    $admin = User::factory()->create(['role' => 'admin']);

    $aoi = Aoi::factory()->create([
        'PartNum'      => '0000-0001',
        'WareHouseCode' => 'BATAM',
        'BinNum'       => 'AOI',
        'MainTranQty'  => 10,
        'PhysicalQty'  => 10,
    ]);

    $response = $this->actingAs($admin)
        ->post(route('admin.transaction.aoi.aoiStore'), [
            'transaction_date' => now()->toDateString(),
            'part_number'      => '0000-0001',
            'part_desc'        => 'AOI Item',
            'warehouse_code'   => 'BATAM',
            'bin_code'         => 'AOI',
            'transaction_type' => 'IN',
            'quantity'         => 5,
            'remarks'          => 'Testing IN',
        ]);

    $response->assertRedirect(route('admin.transaction.aoi.aoiPage'));

    $this->assertDatabaseHas('aois', [
        'PartNum'      => '0000-0001',
        'PhysicalQty' => 15,
    ]);
}
```

Pengujian ini bertujuan untuk memastikan bahwa transaksi barang masuk (IN) akan menambah jumlah stok barang pada sistem. Pada *skenario* ini dibuat data barang awal dengan stok tertentu. Kemudian dilakukan simulasi transaksi barang masuk dengan jumlah tertentu. Hasil yang diharapkan adalah nilai *PhysicalQty* pada *database* bertambah sesuai dengan jumlah transaksi yang dilakukan.

[7]. Test | FR007 - Barang Keluar & Stok Berkurang

```
public function test_fr007_stock_reduces_when_item_out()
{
    /** @var \Illuminate\Contracts\Auth\Authenticated $admin */
    $admin = User::factory()->create(['role' => 'admin']);

    // Buat data AOI awal
    $aoi = Aoi::factory()->create([
        'PartNum'      => '0000-0001',
        'WareHouseCode' => 'BATAM',
        'BinNum'       => 'AOI',
        'MainTranQty'  => 10,
        'PhysicalQty'  => 10,
    ]);

    $response = $this->actingAs($admin)
        ->post(route('admin.transaction.aoi.aoiStore'), [
            'transaction_date' => now()->toDateString(),
            'part_number'      => '0000-0001',
            'part_desc'        => 'AOI Item',
            'warehouse_code'   => 'BATAM',
            'bin_code'         => 'AOI',
            'transaction_type' => 'OUT',
            'quantity'         => 5,
            'remarks'          => 'Testing OUT',
        ]);

    $response->assertRedirect(route('admin.transaction.aoi.aoiPage'));

    // Pastikan stok berkurang
    $this->assertDatabaseHas('aois', [
        'PartNum'      => '0000-0001',
        'PhysicalQty' => 5,
    ]);
}
```

Pengujian ini dilakukan untuk memastikan bahwa transaksi barang keluar (OUT) akan mengurangi jumlah stok yang tersedia pada sistem. Pada pengujian ini dibuat data barang dengan stok awal tertentu, kemudian dilakukan transaksi barang keluar. Hasil yang diharapkan adalah stok barang pada *database* berkurang sesuai dengan jumlah transaksi yang dilakukan.

[8]. Test | FR008 - Barang Keluar Stock Tidak Cukup

```
public function test_fr008_out_fails_when_stock_not_enough()
{
    /** @var \Illuminate\Contracts\Auth\Authenticatable $admin */
    $admin = User::factory()->create(['role' => 'admin']);

    $aoi = Aoi::factory()->create([
        'PartNum'      => '0000-0001',
        'WareHouseCode' => 'BATAM',
        'BinNum'       => 'AOI',
        'PhysicalQty'  => 2,
    ]);

    $response = $this->actingAs($admin)
        ->post(route('admin.transaction.aoi.aoiStore'), [
            'transaction_date' => now()->toDateString(),
            'part_number'      => '0000-0001',
            'warehouse_code'   => 'BATAM',
            'bin_code'         => 'AOI',
            'transaction_type' => 'OUT',
            'quantity'         => 5,
        ]);

    $response->assertSessionHas('error');

    // Stok tetap
    $this->assertDatabaseHas('aois', [
        'PartNum'      => '0000-0001',
        'PhysicalQty' => 2,
    ]);
}
```

Pengujian ini bertujuan untuk memastikan bahwa sistem dapat mencegah transaksi barang keluar apabila jumlah stok yang tersedia tidak mencukupi. Pada *skenario* ini dibuat data barang dengan stok yang kecil, kemudian dilakukan transaksi barang keluar dengan jumlah yang lebih besar dari stok yang tersedia. Hasil yang diharapkan adalah sistem menampilkan pesan error dan jumlah stok pada *database* tidak berubah.

[9]. Test | FR010 - Upload Excel Valid

```
public function test_fr010_admin_can_upload_valid_excel()
{
    // FR010 - Upload data massal via Excel
    /** @var \Illuminate\Contracts\Auth\Authenticatable $admin */
    $admin = User::factory()->create(['role' => 'admin']);

    Excel::fake();

    $response = $this->actingAs($admin)->post('/admin/vitrox/import-aoi', [
        'file' => UploadedFile::fake()->create('aoi.xlsx')
    ]);

    Excel::assertImported('aoi.xlsx');
    $response->assertStatus(302);
}
```

Pengujian ini bertujuan untuk memastikan bahwa sistem dapat menerima dan memproses *upload* data secara massal melalui file Excel. Pada *skenario* ini dilakukan simulasi *upload* file Excel menggunakan fitur fake dari library Laravel Excel. Hasil yang diharapkan adalah sistem berhasil memproses file yang diunggah dan melakukan redirect setelah proses import selesai.

[10]. Test | FR011 - Validasi Excel Duplikat

```
public function test_fr011_add_duplicate_validation()
{
    // FR011 - Validasi otomatis data Excel duplikat
    /** @var \Illuminate\Contracts\Auth\Authenticatable $admin */
    $admin = User::factory()->create(['role' => 'admin']);

    Aoi::factory()->create([
        'PartNum' => '2140-0011'
    ]);

    $response = $this->actingAs($admin)->post('/admin/vitrox/aoi/add', [
        'PartNum' => '2140-0011',
        'PartDesc' => 'Duplicate Item',
        'WareHouseCode' => 'WH01',
        'PhysicalQty' => 5,
        'MainTranQty' => 5
    ]);

    $response->assertSessionHas('error');

    $this->assertDatabaseCount('aois', 1);
}
```

Pengujian ini bertujuan untuk memastikan bahwa sistem dapat mendeteksi data yang memiliki Part Number yang sama (duplikat). Pada *skenario* ini terlebih

dahulu dibuat data dengan Part Number tertentu di dalam *database*. Kemudian dilakukan percobaan penambahan data dengan Part Number yang sama. Hasil yang diharapkan adalah sistem menolak penyimpanan data tersebut, menampilkan pesan error, dan jumlah data pada *database* tidak bertambah.

[11]. Test | FR018 - *Guest* Akses Landing Page

```
public function test_fr018_guest_can_access_landing_page()
{
    // FR018 - Guest dapat membuka landing page tanpa login
    $response = $this->get('/');

    $response->assertStatus(200);
}
```

Pengujian ini bertujuan untuk memastikan bahwa pengguna yang belum melakukan login (*Guest*) tetap dapat mengakses halaman landing page sistem. Pada *skenario* ini dilakukan permintaan akses ke halaman utama tanpa autentikasi. Hasil yang diharapkan adalah sistem memberikan status HTTP 200 (OK) yang menandakan halaman dapat diakses dengan normal.

4.2.3. Hasil Evaluasi Pengujian

Berdasarkan hasil pengujian yang telah dilakukan menggunakan metode Black Box Testing, seluruh kebutuhan fungsional sistem berhasil dijalankan sesuai dengan spesifikasi yang telah ditetapkan pada tahap analisis dan perancangan. Pengujian dilakukan terhadap 18 kebutuhan fungsional (FR001–FR018) yang mencakup proses autentikasi pengguna, pengelolaan data pengguna, pengelolaan data spare part, transaksi barang masuk dan keluar, unggah data Excel, validasi data inventory, pencatatan riwayat transaksi, serta pembuatan laporan. Hasil pengujian menunjukkan bahwa seluruh fungsi utama sistem telah menghasilkan keluaran yang sesuai dengan kebutuhan yang telah ditentukan.

Dari total 18 kebutuhan fungsional yang diuji, seluruhnya memperoleh status berhasil dengan tingkat keberhasilan pengujian sebesar 100%. Meskipun demikian, terdapat beberapa kebutuhan fungsional yang masih memiliki catatan terkait aspek implementasi maupun pengembangan lanjutan. Catatan tersebut tidak

memengaruhi fungsi utama sistem secara keseluruhan, namun menunjukkan adanya peluang untuk penyempurnaan fitur agar dapat memberikan pengalaman pengguna yang lebih optimal dan mendukung kebutuhan operasional yang lebih kompleks di masa mendatang.

Selain pengujian fungsional, dilakukan pula feature testing pada beberapa fitur yang memiliki logika proses yang lebih kompleks, seperti fitur unggah file Excel, validasi data duplikat, dan halaman landing page untuk pengguna umum. Pengujian ini bertujuan untuk memastikan bahwa fungsi-fungsi penting yang berkaitan dengan integritas data dan interaksi pengguna dapat berjalan sesuai dengan rancangan sistem. Berdasarkan hasil pengujian yang diperoleh, fitur-fitur tersebut berhasil dijalankan dengan baik sehingga dapat disimpulkan bahwa sistem yang dikembangkan telah memenuhi kebutuhan operasional utama MTSC Singapore Pte. Ltd., meskipun masih terdapat beberapa aspek yang dapat disempurnakan pada pengembangan berikutnya.

Dengan demikian, sistem inventory berbasis web ini dinilai layak untuk digunakan sebagai alat bantu dalam pengelolaan data persediaan barang pada MTSC Singapore Pte. Ltd. Sistem yang dikembangkan mampu mendukung proses pencatatan dan pemantauan inventory secara lebih terstruktur dibandingkan metode manual yang sebelumnya digunakan. Selain itu, fitur unggah data Excel, validasi data, dan pencatatan riwayat transaksi dapat membantu meningkatkan efisiensi kerja serta mengurangi risiko kesalahan dalam pengelolaan data persediaan. Hasil pengujian yang telah dilakukan juga menunjukkan bahwa seluruh kebutuhan fungsional sistem telah terpenuhi sesuai dengan tujuan pengembangan yang ditetapkan.

BAB V KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil perancangan, implementasi, serta pengujian sistem inventory spare part berbasis web yang telah dilakukan, maka dapat diambil beberapa kesimpulan sebagai berikut:

1. Sistem inventory berbasis web berhasil dirancang dan dikembangkan sesuai dengan kebutuhan pengguna yang telah dianalisis pada tahap awal penelitian. Sistem ini dibangun menggunakan framework Laravel dengan basis data MySQL sehingga mampu menyediakan pengelolaan data inventory secara terstruktur dan terintegrasi.
2. Sistem yang dikembangkan telah berhasil mengimplementasikan berbagai fitur utama yang dibutuhkan, seperti manajemen data barang, pencatatan transaksi barang masuk dan keluar, fitur *upload* data massal melalui file Excel, validasi otomatis data, pencatatan history transaksi, serta pembuatan laporan dalam format *PDF* dan Excel.
3. Berdasarkan hasil pengujian menggunakan metode Black Box Testing, sebagian besar fitur utama sistem telah berjalan sesuai dengan kebutuhan fungsional yang telah ditentukan. Dari total kebutuhan fungsional yang diuji, mayoritas berhasil memenuhi hasil yang diharapkan, meskipun masih terdapat beberapa fungsi yang memerlukan penyempurnaan lebih lanjut. Sistem mampu memproses data dan menghasilkan keluaran yang sesuai pada sebagian besar skenario pengujian yang dilakukan.
4. Berdasarkan hasil evaluasi pengujian, sistem mampu membantu proses pengelolaan inventory spare part secara lebih terstruktur dibandingkan metode pencatatan manual yang sebelumnya digunakan. Fitur unggah data Excel, validasi data, serta pencatatan riwayat transaksi memberikan dukungan terhadap peningkatan efisiensi dan akurasi pengelolaan data persediaan.

Dengan demikian, dapat disimpulkan bahwa tujuan dari proyek akhir ini, yaitu merancang dan mengembangkan sistem inventory berbasis web yang mampu meningkatkan efisiensi dan akurasi pengelolaan data persediaan barang, telah berhasil dicapai.

5.2 Saran

Berdasarkan hasil pengembangan dan pengujian sistem inventory yang telah dilakukan, terdapat beberapa saran yang dapat dijadikan sebagai bahan pengembangan lebih lanjut di masa mendatang, antara lain sebagai berikut:

1. Sistem inventory yang telah dikembangkan dapat ditingkatkan dengan menambahkan fitur barcode atau QR code scanning untuk mempercepat proses input dan pencatatan barang di gudang.
2. Untuk implementasi di lingkungan perusahaan, disarankan adanya pelatihan penggunaan sistem kepada pengguna agar seluruh fitur dapat dimanfaatkan secara optimal.
3. Penelitian selanjutnya juga dapat mengembangkan sistem agar dapat terintegrasi dengan sistem lain, sehingga pengelolaan data perusahaan dapat dilakukan secara lebih terpadu.

Dengan adanya pengembangan lebih lanjut, diharapkan sistem inventory ini dapat memberikan manfaat yang lebih besar dalam mendukung proses manajemen persediaan barang di perusahaan. Pengembangan fitur tambahan serta integrasi dengan sistem lain di masa mendatang diharapkan dapat meningkatkan efektivitas pengelolaan data dan mendukung pengambilan keputusan yang lebih akurat. Selain itu, hasil penelitian ini dapat menjadi referensi bagi pengembangan sistem inventory berbasis web pada penelitian selanjutnya.

DAFTAR PUSTAKA

- [1] K. C. Laudon and J. P. Laudon, *Management Information Systems*, 16th ed. Harlow, U.K.: Pearson, 2020. ISBN: 978-1292353364. [Online]. Available:
<https://www.pearson.com/en-us/subject-catalog/p/management-information-systems/P200000003812/9781292353364>
- [2] E. Turban, et al., *Information Technology for Management: On-Demand Strategies for Performance, Growth and Sustainability*, 11th ed. Hoboken, NJ: Wiley, 2018. ISBN: 978-1118890783. [Online]. Available:
<https://www.wiley.com/en-us/Information+Technology+for+Management%3A+On+Demand+Strategies+for+Performance%2C+Growth+and+Sustainability%2C+11th+Edition-p-9781118890783>
- [3] S. Chopra and P. Meindl, *Supply Chain Management: Strategy, Planning, and Operation*, 7th ed. Harlow, U.K.: Pearson, 2019. ISBN: 978-0134731889. [Online]. Available: <https://www.pearson.com/en-us/subject-catalog/p/supply-chain-management/P200000002443>
- [4] J. Heizer, B. Render, and C. Munson, *Operations Management: Sustainability and Supply Chain Management*, 12th ed. Harlow, U.K.: Pearson, 2017. ISBN: 978-0134422404.
- [5] M. Christopher, *Logistiks & Supply Chain Management*, 5th ed. Harlow, U.K.: Pearson, 2016. ISBN: 978-1292083797. [Online]. Available: <https://archive.org/details/logistics-supply-chain-management-5th-edition>
- [6] A. Gunasekaran and E. W. T. Ngai, "Adoption of supply chain management systems," *Omega*, vol. 40, no. 5, pp. 604–615, 2012. [Online]. Available: <https://doi.org/10.1016/j.omega.2011.09.002>
- [7] D. J. Bowersox, D. J. Closs, M. B. Cooper, and J. C. Bowersox, *Supply Chain Logistiks Management*, 5th ed. New York, NY: McGraw-Hill, 2019. ISBN: 978-1260084610.

- [8] H. Zhou and W. C. Benton, "Supply chain practice and information sharing," *Decision Sciences*, vol. 38, no. 4, pp. 659–692, 2007. [Online]. Available: <https://doi.org/10.1111/j.1540-5915.2007.00170.x>
- [9] C. Alvarenga, R. T. Schoenherr, and C. W. Petersen, "Real-time inventory management: A systematic literature review," *Journal of Business Logistiks*, vol. 39, no. 3, pp. 260–273, 2018. [Online]. Available: <https://doi.org/10.1111/jbl.12189>
- [10] R. A. Putra, et al., "Pengembangan Sistem *Inventory* Berbasis *Web*," *Jurnal Teknologi Informasi*, vol. 7, no. 2, pp. 55–63, 2021. [Online]. Available: <https://ejournal.example.ac.id/jti/article/view/1234>
- [11] D. N. Sari, et al., "Integrasi Sistem *Inventory* dengan Penjualan," *Jurnal Sistem Informasi*, vol. 16, no. 1, pp. 45–52, 2020. [Online]. Available: <https://ejournal.example.ac.id/jsi/article/view/987>, T. Sutabri, *Analisis Sistem Informasi*. Yogyakarta, Indonesia: Andi, 2012. ISBN: 978-9792910749.
- [12] K. C. Laudon and J. P. Laudon, *Management Information Systems: Managing the Digital Firm*, 15th ed. Harlow, U.K.: Pearson Education, 2018. ISBN: 978-0134639710.
- [13] J. Walkenbach, *Microsoft Excel 2013 Bible*. Indianapolis, IN: John Wiley & Sons, 2013. ISBN: 978-1118490372. [Online]. Available: <https://www.wiley.com/en-us/Microsoft+Excel+2013+Bible-p-9781118490372>
- [14] T. Connolly and C. Begg, *Basis data Systems: A Practical Approach to Design, Implementation, and Management*, 6th ed. Harlow, U.K.: Pearson, 2015. ISBN: 978-1292061184.
- [15] J. A. O'Brien and G. M. Marakas, *Management Information Systems*, 10th ed. New York, NY: McGraw-Hill, 2011. ISBN: 978-0073376813.
- [16] Bassil, Y. (2012). A Simulation Model for the *Waterfall* Software Development Life Cycle. *International Journal of Engineering & Technology (iJET)*.
- [17] Pressman, R. S. (2015). *Software Engineering: A Practitioner's Approach*. McGraw-Hill.

[18] Sommerville, I. (2016). Software Engineering. Pearson.

DAFTAR LAMPIRAN

Daftar lampiran disesuaikan dengan ruang lingkup Proyek Akhir, yaitu apakah berfokus pada pengembangan perangkat lunak, implementasi jaringan, atau *artificial intelligence*.

Lampiran A. Dokumen Proses Pengumpulan Requirement

Instrumen, dokumentasi, dll

Lampiran B. Link Produk

Repository GitHub: <https://github.com/BagusDP2011/MtscSG>

Link Figma:

<https://www.figma.com/design/csNktRY2rJ4jBS4D9M610K/MTSC?node-id=0-1&t=gxRLGERTq4FEbzTv-1>

Lampiran C. Dokumen Pengujian Testing

Ada di dalam dokumen ini.