

**RANCANG BANGUN SISTEM INFORMASI MANAJEMEN
INVENTARIS PADA DATA CENTER DI PERUSAHAAN
MANUFAKTUR SEMIKONDUKTOR BERBASIS WEB
MENGGUNAKAN.NET**

PROYEK AKHIR

Disusun Oleh:

Adil Gunawan

3312311049

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

Rasa syukur kami haturkan kepada Tuhan Yang Maha Esa atas segala berkah dan anugerah-Nya sehingga laporan proyek tugas akhir ini, dengan judul "Sistem Informasi Manajemen Inventaris di perusahaan semikonduktor", dapat terselesaikan dengan sempurna. Dokumen ini disusun sebagai salah satu persyaratan untuk menuntaskan program studi di Politeknik Negeri Batam dan sebagai wujud implementasi ilmu pengetahuan yang telah diperoleh selama masa perkuliahan.

Proyek tugas akhir ini memiliki tujuan untuk mendesain, mengimplementasikan, dan menguji platform informasi berbasis web guna menunjang administrasi inventaris pada data center secara produktif, sistematis, dan presisi. Platform ini dikembangkan menggunakan ASP.NET Core sebagai framework pengembangan backend dan SQL Server sebagai database. Dengan memanfaatkan teknologi tersebut, kami berusaha menciptakan solusi yang dapat mendukung proses pencatatan item masuk, item keluar, manajemen penggolongan barang, monitoring kegiatan pengguna, dan pembuatan laporan yang dapat diekspor ke format PDF atau **Excel**.

Laporan ini menguraikan berbagai fase krusial, dimulai dari analisis requirement, perancangan, implementasi, hingga testing aplikasi. Diharapkan, dokumen ini dapat menyajikan gambaran yang komprehensif tentang tahapan-tahapan dalam membangun platform informasi berbasis web serta menjadi acuan bagi pengembangan aplikasi sejenis di masa mendatang.

Pada saat ini, penulis ingin mengungkapkan rasa syukur yang mendalam kepada:

1. Bapak Ir. Bambang Hendrawan, ST., MSM., CIPMP., CISCIP. selaku Direktur Politeknik Negeri Batam.
2. Bapak Ahmad Hamim Thohari, S.S.T., M.T. selaku Ketua Jurusan Teknik Informatika.

3. Ibu Yeni Rokhayati, S.Si., M.Sc. selaku Ketua Program Studi Teknik Informatika.
4. Ibu Maria, S. ST., M. Sn. sebagai Dosen Pembimbing yang sudah menginvestasikan waktu, usaha, dan pemikiran untuk memberikan petunjuk dan bantuan sepanjang proses pengerjaan proyek akhir ini hingga selesai.
5. Ibu Sartikha, S. ST., M.Eng dan Ibu Alena Uperiati, S.T, M.Cs. selaku Dosen Penguji yang telah memberikan masukan dan saran membangun demi penyempurnaan laporan ini.
6. Pimpinan dan seluruh staf PT Infineon Technologies Batam, khususnya kepada Bapak Arif Wahyu Prasetya selaku pembimbing lapangan yang telah memberikan kesempatan, fasilitas, dan ilmu praktis selama penelitian berlangsung.
7. Ayah, ibu, dan keluarga tersayang yang senantiasa memberikan doa, motivasi hidup, dan dukungan mental maupun material yang tidak ada habisnya.
8. Kepada seluruh rekan-rekan mahasiswa Informatika angkatan 2023, penulis mengucapkan rasa syukur yang mendalam atas kebersamaan yang telah menghadirkan keceriaan dalam keseharian dan menjadikan proses perkuliahan ini terasa lebih bermakna serta lancar.

Akhir kata, penulis berharap laporan ini akan dapat menjadi kontribusi nyata yang bermanfaat, baik bagi pengembangan ilmu pengetahuan, peningkatan efisiensi manajemen inventaris di perusahaan, maupun sebagai referensi bagi pihak-pihak yang membutuhkan.

DAFTAR ISI

KATA PENGANTAR	2
DAFTAR TABEL.....	6
DAFTAR GAMBAR.....	7
ABSTRAK.....	9
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	12
BAB II TINJAUAN PUSTAKA	14
2.1. Penelitian Terkait	14
2.2. Landasan Teori	15
2.2.1. Sistem Informasi	15
2.2.2. Inventaris Barang.....	15
2.2.3. Sistem Manajemen Inventaris.....	16
2.2.4. ASP.NET Core.....	16
2.2.5. <i>SQL Server</i>	17
2.2.6. <i>User Acceptance Testing(UAT)</i>	17
2.2.7. <i>Black Box Testing</i>	17
2.2.8. <i>Swagger</i>	17
2.3. Metode Pengembangan Produk	19
BAB III ANALISIS DAN PERANCANGAN.	22
3.1. Analisis Kebutuhan.....	22
3.1.1 Proses Bisnis Berjalan	22
3.1.2 Gambaran Umum Sistem.....	23
3.2. Perancangan	25
3.2.1 Kebutuhan Fungsional	25
3.2.2 Kebutuhan Non-Fungsional.....	26
3.2.3 Diagram <i>Use Case</i>	27
3.2.4 Skenario <i>Use Case</i>	29

3.2.5	<i>Activity diagram</i>	39
3.2.6	ER Diagram	49
3.2.7	Perancangan Antarmuka (Wireframe)	50
BAB IV IMPLEMENTASI DAN PENGUJIAN.....		53
4.1.	Hasil Implementasi	53
4.1.1	Tools yang Digunakan untuk Impelementasi :	53
4.1.2	Tools yang Digunakan untuk Impelementasi	54
4.1.3	Tampilan hasil Implementasi.....	55
4.1.4	Implementasi Basis Data	61
4.2.	Pengujian <i>User Acceptance Testing</i> (UAT).....	65
BAB V KESIMPULAN.....		81
5.1	Kesimpulan	81
5.2	Saran.....	82
DAFTAR PUSTAKA		85
DAFTAR LAMPIRAN.....		85

DAFTAR TABEL

Tabel 1. Perbandingan Penelitian Sebelumnya.....	14
Tabel 2. Kebutuhan Fungsional	26
Tabel 3. Kebutuhan Non Fungsional	27
Tabel 4. Skenario <i>Use Case</i> Login	29
Tabel 5. Skenario <i>Use Case</i> Register.....	30
Tabel 6. Skenario <i>Use Case</i> Dashboard.....	31
Tabel 7. Skenario <i>Use Case</i> Manajemen Barang	32
Tabel 8. Skenario <i>Use Case</i> Edit Barang.....	33
Tabel 9. Skenario <i>Usecase</i> Hapus Barang	34
Tabel 10. Skenario <i>Use Case</i> Detail Barang.....	35
Tabel 11. Skenario <i>Use Case</i> Mengelola User	36
Tabel 12. Skenario <i>Use case</i> Ekspor Laporan	37
Tabel 13. Skenario <i>Use Case</i> Logout	38
Tabel 14. Pengujian Fitur <i>Login</i>	66
Tabel 15. Pengujian Fitur Registrasi.....	68
Tabel 16. Pengujian Fitur Input Barang.....	70
Tabel 17. Pengujian Fitur Edit Barang	72
Tabel 18. Pengujian Fitur Hapus Barang.....	73
Tabel 19. Pengujian Fitur Report Barang	75
Tabel 20. Pengujian Fitur Filter Kategori Barang	77
Tabel 21. Pengujian Fitur Detail Barang	79
Tabel 22. Pengujian Fitur <i>Logout</i>	80

DAFTAR GAMBAR

Gambar 1 Ilustrasi Tahapan Metode Agile.....	19
Gambar 2 Gambaran Umum Sistem.....	24
Gambar 3 Gambaran Alur Sistem.....	25
Gambar 4. Use Case Diagram.....	28
Gambar 5. Activity Diagram Register	40
Gambar 6. Activity Diagram Login.....	41
Gambar 7. Activity Diagram Tambah Barang.....	42
Gambar 8 Activity Diagram Edit Barang	43
Gambar 9. Activity Diagram Hapus Barang.....	44
Gambar 10 Activity Diagram Pencarian & Filter Barang	45
Gambar 11 Activity Diagram Lihat Detail Barang.....	46
Gambar 12 Activity Diagram Export Barang.....	47
Gambar 13 Activity Diagram Logout.....	48
Gambar 14 ERD Diagram.....	49
Gambar 15 <i>Design Wireframe</i> Dashboard.....	50
Gambar 16 <i>Design Wireframe</i> Products	51
Gambar 17 <i>Design Wireframe</i> Product Detail.....	52
Gambar 18 Tampilan Dashboard.....	55
Gambar 19 Tampilan <i>Products</i>	56
Gambar 20 Tampilan <i>Product Detail</i>	57
Gambar 21 Tampilan Halaman <i>Category</i>	58
Gambar 22 Tampilan Halaman <i>Settings</i>	59
Gambar 23 Tampilan Halman <i>Report</i>	60
Gambar 24 Implementasi Basis Data.....	61
Gambar 25 Gambar Tabel <i>Users</i>	62
Gambar 26 Gambar Tabel <i>Items</i>	62
Gambar 27 Gambar Tabel <i>Categories</i>	63
Gambar 28 Gambar Tabel Data Center	63
Gambar 29 Gambar Tabel <i>Report</i>	63

DAFTAR LAMPIRAN

Lampiran A. Dokumen Proses Pengumpulan Requirement	16
Lampiran B. Link Produk	16
Lampiran C. Dokumen Pengujian UAT	16

ABSTRAK

Penelitian bertujuan untuk merancang sebuah platform inventaris berbasis web yang dapat memfasilitasi administrasi data aset secara lebih sistematis, presisi, dan produktif. Platform ini didesain untuk menyelesaikan permasalahan dokumentasi inventaris yang masih dijalankan secara konvensional, yang berpotensi mengakibatkan kesalahan informasi, keterlambatan pencarian data, serta minimnya transparansi dalam administrasi aset.

Metodologi yang diterapkan dalam pembangunan sistem ini adalah pendekatan pengembangan aplikasi berbasis web dengan konsep REST API. Platform dibangun menggunakan teknologi *backend* .NET dan frontend berbasis HTML, Tailwind CSS, serta JavaScript. Fitur kunci yang dikembangkan mencakup administrasi data produk, kategori, dan lokasi data center, serta fungsi tambah, edit, delete, pencarian, dan filter data secara real-time.

Sistem ini juga menerapkan pembagian hak akses pengguna (*role-based access control*) antara admin dan *user* untuk meningkatkan keamanan dan kontrol terhadap data. Admin memiliki akses full terhadap pengelolaan data, sedangkan *user* hanya dapat melihat informasi yang tersedia.

Hasil dari riset ini memperlihatkan bahwa platform yang dibangun mampu mengoptimalkan efektivitas dalam administrasi inventaris, meminimalisir kesalahan dokumentasi, serta memfasilitasi proses supervisi dan pengambilan keputusan. Dengan ini, platform ini diharapkan dapat menjadi solusi yang produktif dalam menunjang pengelolaan aset di lingkungan data center.

Kata kunci: sistem inventaris, web, REST API, manajemen aset, data center

BAB I PENDAHULUAN

1.1. Latar Belakang

Data center merupakan infrastruktur vital yang berfungsi sebagai pusat pengelolaan, penyimpanan, dan distribusi data dalam suatu perusahaan. Di dalam data center terdapat berbagai perangkat keras dan komponen pendukung, seperti server, switch, router, kabel, dan perangkat jaringan lainnya. Banyaknya barang yang keluar masuk menuntut adanya sistem pengelolaan inventaris yang terstruktur dan akurat. Tanpa adanya sistem yang baik, proses pencatatan barang rawan menimbulkan kesalahan, duplikasi data, kehilangan informasi, bahkan potensi kerugian perusahaan.

Proses pengelolaan inventaris yang masih dilakukan secara manual atau menggunakan media sederhana sering kali menimbulkan kendala, seperti sulitnya melakukan pencarian data, keterlambatan pencatatan barang masuk dan keluar, serta kesulitan dalam penyusunan laporan. Kondisi tersebut menunjukkan bahwa dibutuhkan suatu sistem informasi berbasis teknologi yang dapat membantu proses pengelolaan inventaris agar lebih efektif, efisien, dan akurat.

Seiring dengan perkembangan teknologi web, salah satu solusi yang dapat diimplementasikan adalah membangun Sistem Manajemen Inventaris Data Center berbasis web. Sistem ini akan memungkinkan staf untuk mencatat, memantau, serta mengelola barang secara digital. Rohman dan Bhakti pada penelitiannya (2022) menyimpulkan bahwa sistem informasi yang dikembangkan memberikan pengaruh positif terhadap perusahaan karena penyajian laporan menjadi lebih akurat dan efisien terhadap transaksi persediaan serta pengeluaran stok barang, serta memudahkan kepala gudang untuk mengontrol stok dan transaksi barang. Fitur utama yang dirancang antara lain autentikasi dan otorisasi pengguna, manajemen barang, pencarian data, serta pembuatan laporan dalam format PDF/Excel. Dengan memanfaatkan ASP.NET Core

sebagai framework pengembangan aplikasi web dan SQL Server sebagai basis data, sistem ini diharapkan mampu memberikan kinerja yang optimal, keamanan data yang lebih baik, serta kemudahan dalam integrasi.

Dengan adanya sistem ini, diharapkan pengelolaan inventaris barang pada data center dapat berjalan lebih efisien, akurat, transparan, serta mendukung kebutuhan audit dan pengambilan keputusan. Selain itu, pengembangan sistem ini juga menjadi sarana pembelajaran dan penerapan keilmuan bagi penulis dalam bidang rekayasa perangkat lunak, basis data, dan pengembangan aplikasi web.

1.2. Rumusan Masalah

Berdasarkan latar belakang yang telah dijelaskan sebelumnya, rumusan masalah yang didapatkan adalah:

1. Bagaimana merancang sistem informasi inventaris yang dapat mempermudah proses pencatatan barang pada data center ?
2. Bagaimana membangun sistem inventaris berbasis web dengan menggunakan *ASP.NET Core* dan *SQL Server* sebagai media penyimpanan data?
3. Bagaimana sistem dapat menghasilkan laporan data inventaris (barang masuk, barang keluar, dan stok) secara cepat, akurat, dan mudah dipahami?
4. Bagaimana menguji kinerja dan keandalan sistem agar dapat digunakan secara efektif di lingkungan data center
5. ?

1.3. Tujuan

Penelitian ini memiliki beberapa masalah yang perlu diperhatikan, yaitu:

1. Merancang dan membangun sistem informasi inventaris berbasis web yang dapat mempermudah proses pencatatan barang masuk dan barang keluar pada data center.
2. Menguji fungsionalitas, kinerja, dan keandalan sistem sehingga dapat digunakan secara efektif dan efisien dalam kegiatan operasional data center.

1.4. Batasan Masalah

Penelitian ini memiliki beberapa masalah yang perlu diperhatikan, yaitu:

1. Sistem difokuskan pada pengelolaan inventaris perangkat di lingkungan data center (misalnya server, switch, router, storage, kabel, dan perangkat jaringan). Pengelolaan aset non-data center berada di luar lingkup.
2. Fitur yang direalisasikan meliputi autentikasi berbasis peran (Admin, Inventaris, Auditor), manajemen barang (tambah, edit, hapus/soft delete, detail, pencarian/filter), pencatatan transaksi (barang masuk, barang keluar), serta pelaporan dengan ekspor PDF/Excel.
3. Pengujian dan demonstrasi menggunakan data dummy atau data yang telah dianonimkan; tidak menggunakan atau menampilkan data internal sensitif. Struktur data disiapkan generik agar dapat diadaptasi di kemudian hari tanpa mengekspos informasi perusahaan.

1.5. Manfaat

Penelitian ini diharapkan dapat memberikan beberapa manfaat, di antaranya

1. Meningkatkan transparansi dalam pengelolaan aset di lingkungan data center.
2. Memberikan fondasi yang dapat diperluas ke integrasi sistem korporat (ERP/CMDB) di masa depan.

3. Mengurangi duplikasi data dan kesalahan pencatatan melalui model data terstruktur.
4. Memudahkan pencarian serta pemantauan status barang secara *real-time*.

BAB II TINJAUAN PUSTAKA

2.1. Penelitian Terkait

Tabel 1. Perbandingan Penelitian Sebelumnya

Peneliti/Tahun	Judul/Topik Penelitian	Metode/Teknologi	Kelebihan	Kekurangan/Celah
Hendra1, Yulia Wahyuningsih Fernandi Mahendrasusila (2024)	Rancang Bangun Sistem Transaksi Perusahaan Berbasis Web Dengan Metode Agile Development	Agile, JavaScript, PostgreSQL	Penggunaan diagram konteks dan DFD (Data Flow Diagram) mempermudah pembuatan proses yang terlibat pada sistem	Sistem belum mencakup proses inventaris
Arif Rohman & Henny Dwi Bhakti (2022)	Perancangan Sistem Informasi Persediaan Barang Berbasis Web	Waterfall, PHP, MySQL, XAMPP	Sistem mempercepat pengolahan data stok barang dan laporan otomatis	Sistem belum mencakup <i>multi-user</i> dan integrasi dengan sistem lain yang mendukung
Antonius Oko Pranoto & Eko Sedyono (2021)	Perancangan Sistem Informasi Inventaris Barang Berbasis Web	Wawancara, DFD, ERD, PHP-MySQL	Sistem mempermudah perangkat desa dalam manajemen inventaris	Tidak terdapat menu edit transaksi, sehingga data harus dihapus terlebih dahulu ketika harus melakukan perubahan data
Deni Yuniarti, Galih Widyatmojo (2025)	Sistem Informasi Persediaan Barang Berbasis Website (Studi Kasus: Instalasi Gizi RS Harapan Sehat Jati Barang)	Gap Analysis, TIA-942	Rancangan data center yang dihasilkan sudah sesuai standar internasional yang mencakup aspek lokasi, keamanan, listrik, pendingin, dan kebakaran	Hasil penelitian yang dibuat hanya pada tahap desain (perancangan), belum diimplementasikan secara nyata

2.2. Landasan Teori

2.2.1. Sistem Informasi

Sistem informasi merupakan kombinasi antara teknologi, manusia, dan prosedur yang berfungsi untuk mengumpulkan, mengolah, menyimpan, dan mendistribusikan informasi guna mendukung pengambilan keputusan serta pengawasan dalam suatu organisasi (Jogiyanto, 2005; Abdul Kadir, 2003, dalam Deni Yuniarti *et al.*, 2025). Sistem informasi membantu organisasi dalam mengelola data transaksi harian, mendukung operasi, dan menyediakan laporan yang relevan untuk kebutuhan manajerial. Dalam konteks perusahaan modern, sistem informasi menjadi pondasi penting untuk mendukung efisiensi dan efektivitas pengelolaan sumber daya organisasi.

2.2.2. Inventaris Barang

Inventaris merupakan kegiatan pengelolaan aset atau barang yang dimiliki oleh suatu organisasi untuk memastikan ketersediaan, kondisi, dan lokasi barang tercatat dengan baik. Menurut Rohman dan Bhakti (2022), inventaris barang mencakup aktivitas pencatatan barang masuk, barang keluar, serta kondisi barang agar dapat diketahui kebutuhan dan ketersediaan stok secara akurat. Pengelolaan inventaris yang masih dilakukan secara manual rentan terhadap kesalahan pencatatan dan kehilangan data, sehingga diperlukan sistem informasi berbasis komputer yang dapat meningkatkan efisiensi dan akurasi (Pranoto & Sedyono, 2021).

2.2.3. Sistem Manajemen Inventaris

Sistem manajemen inventaris adalah sistem informasi yang dirancang untuk mengelola data barang dari proses penerimaan, penyimpanan, peminjaman, hingga pelaporan. Sistem ini berfungsi untuk mempercepat proses pengambilan keputusan terkait pengadaan, distribusi, dan pengawasan aset (Galih Widyatmojo *et al.*, 2025).

Menurut Pranoto dan Sedyono (2021), sistem manajemen inventaris berbasis web memungkinkan data dapat diakses secara real-time, meminimalisasi duplikasi, serta meningkatkan efektivitas kerja staf dalam mengelola data barang. Implementasi sistem berbasis web juga mendukung fleksibilitas akses, di mana pengguna dapat melakukan pengelolaan data dari berbagai lokasi dengan koneksi internet.

2.2.4. ASP.NET Core

ASP.NET Core merupakan *framework open-source* yang dikembangkan oleh Microsoft untuk membangun aplikasi web modern dan layanan berbasis cloud. *Framework* ini mendukung bahasa pemrograman C#, memiliki performa tinggi, serta fleksibel untuk dikembangkan pada berbagai platform seperti Windows, Linux, maupun macOS.

ASP.NET mempermudah proses pengembangan sistem inventaris karena memiliki struktur modular dan keamanan yang kuat. Selain itu, ASP.NET Core dikenal memiliki performa unggul dalam pengolahan data, terutama pada operasi CRUD (Create, Read, Update, Delete), dibandingkan framework lain seperti Express.js dan Django.

2.2.5. SQL Server

SQL Server merupakan sistem manajemen basis data relasional (RDBMS) yang dikembangkan oleh Microsoft untuk mendukung penyimpanan, pengelolaan, dan manipulasi data secara terstruktur. *SQL Server* memiliki keunggulan dalam kecepatan eksekusi *query*, keamanan data, dan kemampuan integrasi yang tinggi dengan aplikasi berbasis .NET. Sistem ini juga mendukung pengelolaan data berskala besar yang cocok digunakan pada sistem manajemen inventaris dengan jumlah transaksi tinggi.

Dalam pengembangan sistem berbasis web, *SQL Server* berperan penting dalam menjamin keandalan data, mendukung *multiuser access*, dan mempermudah pembuatan laporan stok atau inventarisasi barang secara otomatis.

2.2.6. User Acceptance Testing (UAT)

User Acceptance Testing (UAT) merupakan tahap pengujian akhir dalam siklus pengembangan perangkat lunak yang dilakukan oleh pengguna akhir (*end user*) untuk memverifikasi apakah sistem yang telah dikembangkan sudah sesuai dengan kebutuhan dan dapat diterima sebelum sistem tersebut diimplementasikan secara resmi (Anjasmara & Sumitro, 2023).

Tujuan utama dari UAT adalah untuk memastikan bahwa seluruh fitur dan alur kerja sistem telah berjalan sesuai dengan skenario penggunaan yang telah didefinisikan sebelumnya (Anjasmara & Sumitro, 2023). Dalam konteks sistem informasi manajemen inventaris, UAT dilakukan untuk menguji apakah fitur-fitur seperti pencatatan barang masuk dan keluar, pengelolaan kategori, monitoring aktivitas pengguna, serta ekspor laporan dapat berjalan dengan baik sesuai kebutuhan operasional perusahaan.

2.2.7. Black Box Testing

Blackbox testing merupakan metode pengujian perangkat lunak yang bertujuan untuk mengidentifikasi kesalahan pada sistem secara terstruktur sehingga dapat segera ditangani dengan tepat (Susanto et al., 2021).

Metode ini berfokus pada fungsionalitas sistem dari sudut pandang pengguna, tanpa mengharuskan penguji memperhatikan rincian struktur internal maupun kode program yang mendasarinya (Desyani et al., 2022).

Dalam konteks pembangunan platform informasi manajemen inventaris berbasis web, *blackbox* testing diimplementasikan untuk menguji keseluruhan fitur kunci sistem, mencakup proses autentikasi, administrasi data barang, pencarian dan filter klasifikasi, serta export laporan, guna menjamin setiap fungsi beroperasi sesuai dengan spesifikasi requirement yang telah didesain.

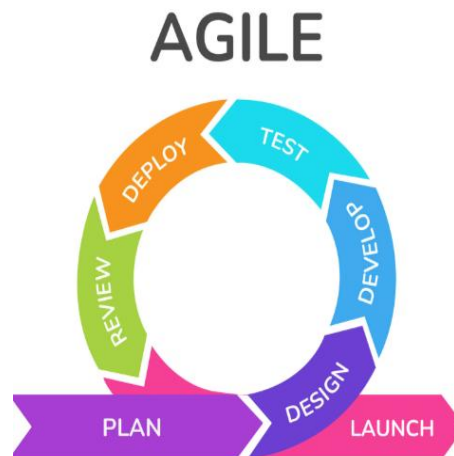
2.2.8. Swagger

Swagger adalah ekosistem alat berbasis standar OpenAPI Specification (OAS) yang digunakan untuk mendeskripsikan, mendokumentasikan, dan menguji layanan REST API secara terstruktur. Melalui Swagger, definisi API ditulis dalam format mesin-baca (JSON atau YAML) yang berisi informasi lengkap mengenai endpoint, parameter, tipe data, contoh permintaan dan respons, hingga skema keamanan. Definisi tersebut kemudian dapat ditampilkan sebagai dokumentasi interaktif melalui Swagger UI, sehingga pengembang dan pihak terkait dapat memahami perilaku API tanpa harus membaca kode sumber secara langsung.

Pada ASP.NET Core, Swagger biasanya diintegrasikan menggunakan pustaka Swashbuckle atau NSwag yang secara otomatis membangkitkan dokumen OpenAPI dari controller, atribut, anotasi, serta komentar XML di dalam kode. Hasil integrasi ini berupa halaman Swagger UI yang menampilkan seluruh endpoint beserta detailnya, lengkap dengan fitur “*Try it out*” untuk mengeksekusi permintaan langsung dari peramban. Dengan demikian, proses validasi fungsi API menjadi lebih cepat karena pengujian dapat dilakukan tanpa alat tambahan, dan dokumentasi akan selalu sinkron dengan implementasi aktual.

2.3. Metode Pengembangan Produk

Metodologi yang diterapkan dalam pembangunan sistem ini adalah Agile Development. Metodologi tersebut merupakan pendekatan yang digunakan dalam pengembangan aplikasi dan dikenal dengan *Software Development Life Cycle* (SDLC).



Gambar 1 Ilustrasi Tahapan Metode Agile

Metode ini berperan dalam pengembangan perangkat lunak, dimana prosesnya terus diulang dengan aturan dan solusi yang telah disetujui dan disepakati oleh tim secara terstruktur dan terorganisir (Wulansari et al., 2022).

Metode *agile development* memiliki cara kerja yang tanggap, cepat, dan elastis dalam beradaptasi dengan perubahan sesuai dengan requirements dari pengguna (Hardiansyah et al., 2023). Berdasarkan penelitian oleh Erwanto M & Umniati N (Erwanto & Umniati, 2022), ada pula dinyatakan melalui perbandingan metode *agile development* dengan salah satu metode pengembangan lainnya, dan diperoleh bahwa penerapan metode *agile development* sangat berguna dalam hal merancang suatu perangkat lunak berdasarkan spesifikasi yang ingin dicapai.

Metode *Agile* digunakan sebagai kerangka pengembangan perangkat lunak dalam proyek Sistem Informasi Manajemen Inventaris pada Data Center. *Agile* dipilih untuk memastikan fleksibilitas, kolaborasi tim, dan kemampuan sistem untuk

menyesuaikan dengan perubahan kebutuhan selama proses pengembangan. Berikut adalah alur metode *Agile* yang diterapkan:

1. *Planning*

Tahap ini dimulai dengan analisis kebutuhan pengguna dan bisnis untuk merancang backlog produk. Backlog ini berisi daftar fungsi sistem seperti manajemen barang, laporan inventaris, dan sistem pengelolaan data pengguna. Prioritas diberikan berdasarkan kebutuhan yang dianggap paling penting.

2. *Iteration(SprintPlanning)*

Pengembangan dilakukan dalam beberapa iterasi (*sprint*), dengan durasi singkat, seperti 1-2 minggu. Setiap iterasi mencakup pengembangan bagian sistem yang dapat diselesaikan secara mandiri, misalnya fitur login, filter kategori, atau laporan barang.

3. *DesignandDevelopment*

Pada setiap *sprint* yang direncanakan, desain sistem dibuat menggunakan diagram seperti *ERD* dan *wireframe*. Setelah itu, dilakukan pengembangan sistem menggunakan *ASP.NET Core* sebagai framework backend dan *SQL Server* sebagai basis data. Tailwind CSS digunakan untuk membangun antarmuka web responsif.

4. *TestingandFeedback*

Setelah pengembangan selesai pada setiap *sprint*, dilakukan pengujian fitur menggunakan metode *blackbox testing* dan *User Acceptance Testing (UAT)* untuk memastikan sistem memenuhi kebutuhan pengguna. Feedback yang diterima akan digunakan untuk perbaikan di iterasi berikutnya.

5. *DeliveryandDeployment*

Setelah semua iterasi selesai, dilakukan integrasi komponen menjadi sistem lengkap yang siap untuk diuji secara keseluruhan. *Deployment* sistem dilakukan pada lingkungan produksi setelah tahap validasi akhir.

6. *Review and Continuous Improvement*

Setelah *deployment*, tahap retrospeksi dilakukan untuk mengevaluasi keberhasilan *sprint* dan mengidentifikasi area yang perlu diperbaiki. Sistem yang sudah berjalan tetap dipantau dan diperbarui apabila terdapat perubahan kebutuhan.

BAB III ANALISIS DAN PERANCANGAN.

3.1. Analisis Kebutuhan

3.1.1 Proses Bisnis Berjalan

Untuk memastikan fungsionalitas aplikasi berjalan sesuai tujuan, sistem inventory data center dirancang dengan alur proses bisnis yang sistematis. Proses ini mencakup interaksi antara pengguna dan sistem, mulai dari tahapan awal pendaftaran hingga sistem menghasilkan rekomendasi berbasis data pengguna. Berikut ini merupakan tahapan proses bisnis yang terjadi di dalam sistem:

1. Pencatatan Barang Secara Terintegrasi
 - Sistem berbasis web memungkinkan semua transaksi barang masuk dan keluar dicatat langsung ke dalam database secara real-time.
 - Hal ini memastikan pencatatan lebih akurat dan mudah dilacak.
2. Sistem Basis Data Terpusat
 - Data inventaris tersimpan dalam basis data terpusat menggunakan SQL Server, sehingga memudahkan akses dan pencarian data barang untuk berbagai keperluan.
3. Otomasi Laporan
 - Pengguna dapat menghasilkan laporan barang dan stok secara otomatis.
 - Laporan tersebut dapat difilter berdasarkan tanggal, dan diekspor ke dalam format PDF atau Excel.
4. Monitoring Aktivitas Pengguna
 - Sistem mencatat setiap aktivitas pengguna, termasuk penambahan, penghapusan, atau pembaruan data barang.

- Fitur monitoring ini mempermudah admin dalam melacak tindakan pengguna dan menjaga keamanan data.

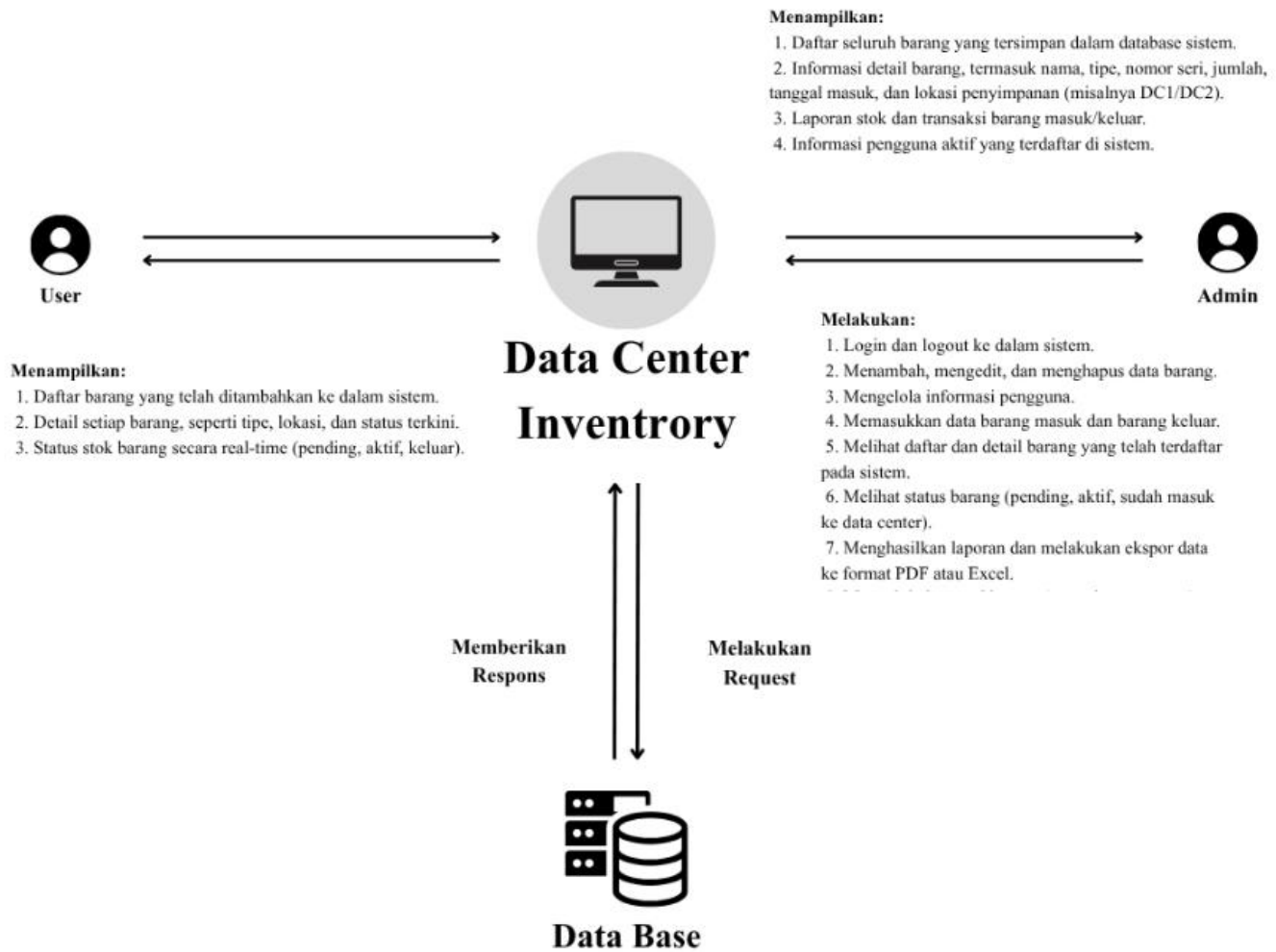
5. Akses Mudah dan Responsif

- Sistem berbasis web memungkinkan akses kapan saja dan di mana saja selama terkoneksi ke internet.
- Dengan antarmuka responsif yang dikembangkan menggunakan Tailwind CSS, pengguna dapat mengakses sistem melalui perangkat dengan berbagai ukuran layar.

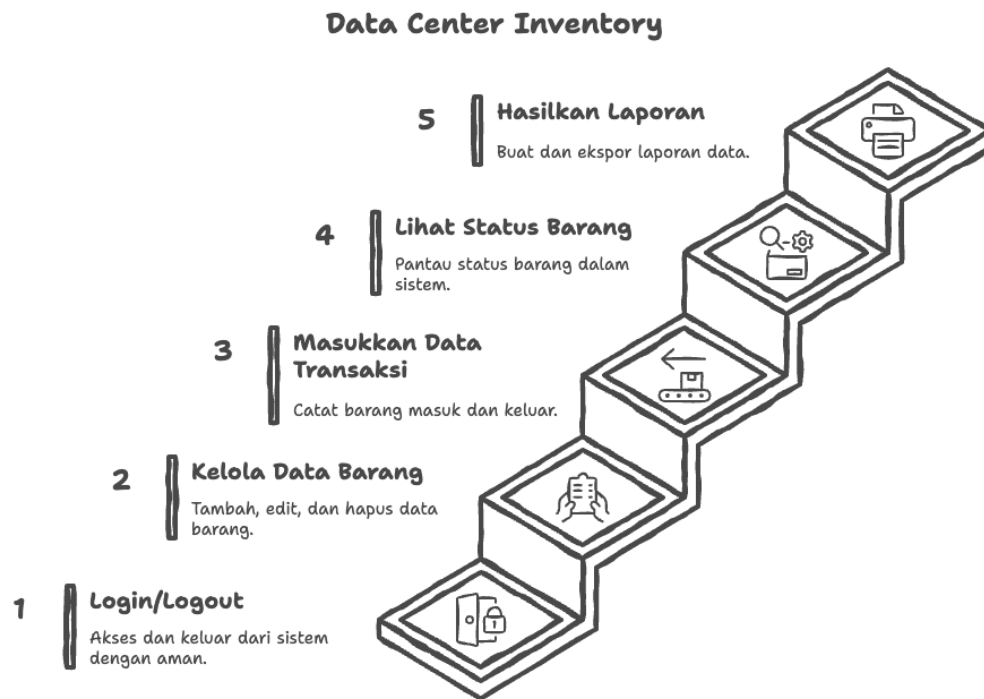
3.1.2 Gambaran Umum Sistem

Gambaran umum sistem menjelaskan secara keseluruhan bagaimana sistem informasi ini bekerja, aktor yang terlibat, serta interaksi antara pengguna dengan sistem. Sistem informasi Inventory Data Center merupakan aplikasi berbasis web yang dirancang untuk membantu proses pencatatan, pengelolaan, dan pelaporan data inventaris barang di lingkungan data center perusahaan.

Sistem ini terdiri atas dua aktor utama, yaitu *Admin* dan *User* (Petugas Data Center). Kedua aktor tersebut memiliki hak akses berbeda sesuai dengan peran dan tanggung jawabnya. Admin memiliki akses penuh untuk mengelola data, sementara user memiliki akses terbatas untuk melakukan input dan pemantauan data barang



Gambar 2 Gambaran Umum Sistem



Gambar 3 Gambaran Alur Sistem

Aplikasi ini memiliki kemampuan untuk merekam data barang masuk dan keluar, mengatur klasifikasi produk, memonitor aktivitas pengguna, dan menyusun laporan inventaris yang dapat diekspor dalam format PDF serta Excel. Aplikasi didesain untuk memaksimalkan efisiensi operasional, tingkat akurasi, dan kemudahan akses data persediaan.

3.2. Perancangan

3.2.1 Kebutuhan Fungsional

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

No	Kebutuhan Fungsional
F001	Pengguna dapat mendaftarkan akun ke sistem
F002	Pengguna dapat masuk ke dalam sistem

No	Kebutuhan Fungsional
F001	Pengguna dapat mendaftarkan akun ke sistem
F003	Sistem menampilkan halaman utama berisi ringkasan data inventaris seperti jumlah total barang, barang masuk, barang keluar, dan stok terkini, serta menyediakan navigasi ke seluruh fitur utama sistem.
F004	Sistem memungkinkan <i>admin</i> untuk menambah barang untuk masuk ke dalam sistem inventaris
F005	Sistem memungkinkan <i>admin</i> untuk mengedit barang di dalam sistem inventaris,
F006	Sistem memungkinkan <i>admin</i> untuk menghapus barang
F007	Sistem menyediakan fitur pencarian dan filter data barang berdasarkan kategori tertentu seperti nama, tipe, dan lokasi, guna memudahkan pengguna dalam menemukan data yang diinginkan.
F008	Sistem menampilkan informasi detail untuk setiap barang
F09	Sistem menghasilkan laporan barang masuk, barang keluar, dan stok barang berdasarkan periode tertentu. Laporan dapat diunduh dalam format PDF atau Excel.
F010	Pengguna dapat keluar dari akun

Tabel 2. Kebutuhan Fungsional

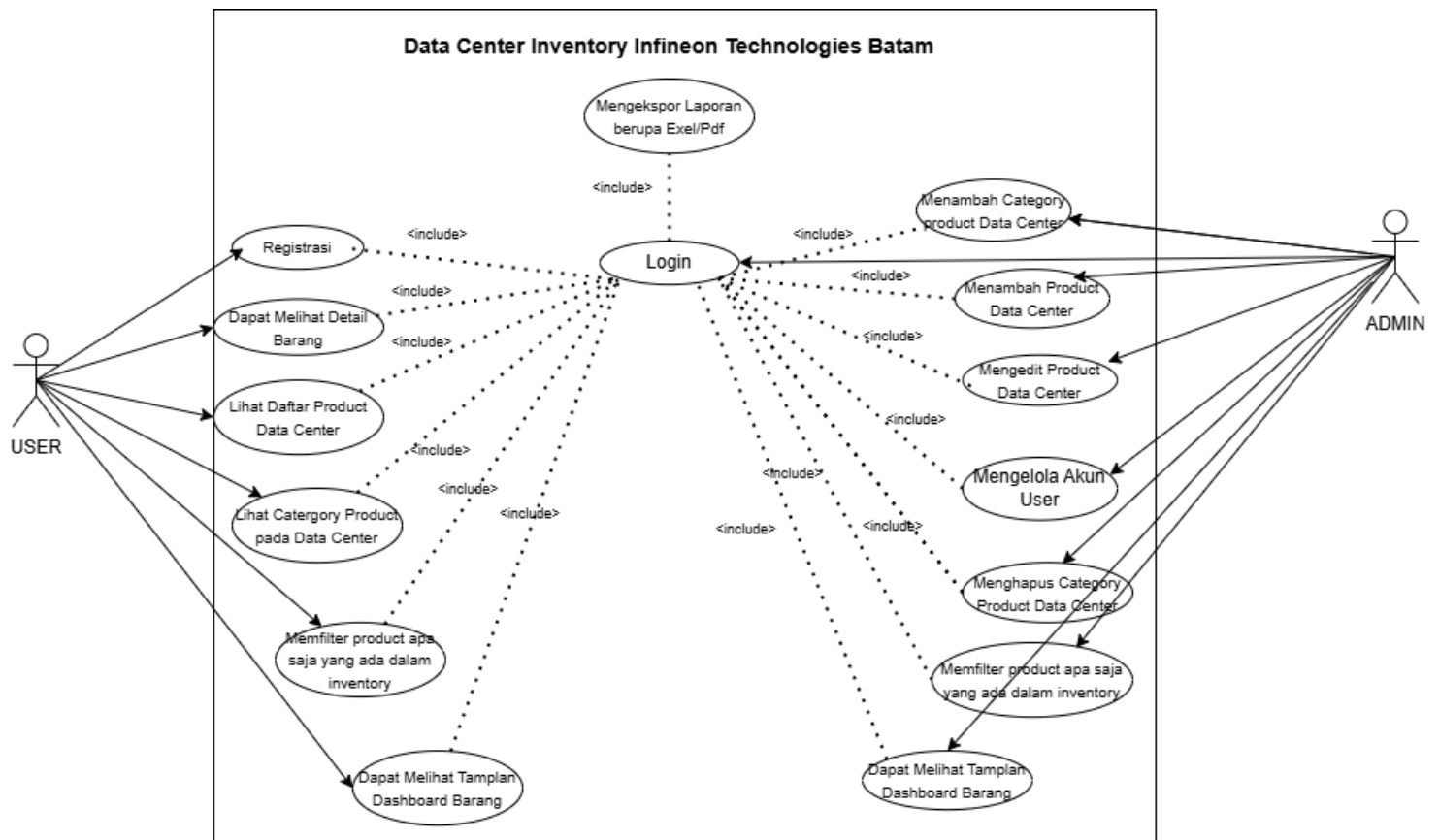
3.2.2 Kebutuhan Non-Fungsional

No	Kebutuhan Non-fungsional
NF001	Sistem harus dapat menampilkan data dan memproses transaksi dengan waktu respon maksimal 3 detik. Sistem juga harus mampu menangani minimal 500 data inventaris tanpa penurunan performa.
NF002	Sistem harus dapat beroperasi secara stabil meskipun digunakan oleh beberapa pengguna secara bersamaan, tanpa kehilangan data atau gangguan fungsi utama.
NF003	Antarmuka sistem dirancang agar mudah dipahami dan digunakan oleh pengguna non-teknis, dengan tata letak navigasi yang intuitif dan konsisten di setiap halaman.

NF004	Sistem harus tersedia dan dapat digunakan minimal 99% selama jam operasional data center untuk mendukung kegiatan pencatatan dan pengelolaan inventaris
NF005	Sistem wajib mengimplementasikan mekanisme login dan kontrol akses untuk memastikan bahwa data hanya dapat diakses oleh pengguna yang telah diverifikasi. Password disimpan dalam database menggunakan teknik hash untuk menjaga keamanan.

Tabel 3. Kebutuhan Non Fungsional

3.2.3 Diagram Use Case



Gambar 4. Use Case Diagram

Diagram *use case* memvisualisasikan keterkaitan dan interaksi yang terjalin antara aktor dengan sistem saat mengeksekusi berbagai fungsi yang disediakan. Pemetaan ini menjelaskan peranan masing-masing pengguna terhadap fitur aplikasi, serta mendefinisikan batasan dan lingkup sistem yang dikonstruksi.

Pada sistem Inventory Data Center, terdapat dua aktor utama, yaitu *Admin* dan *User*.

3.2.4 Skenario Use Case

Usecase Login

Skenario usecase *login* melibatkan dua aktor utama, yaitu *user* dan *admin*. Skenario ini menjelaskan langkah-langkah yang dilakukan oleh aktor saat melakukan *login* menggunakan kredensial (*email* dan *password*) hingga berhasil masuk ke dalam aplikasi. Selain alur kejadian normal, skenario ini juga mencakup kemungkinan alternatif jika pengguna belum terdaftar atau memasukkan kredensial yang salah. Berikut merupakan tabel skenario usecase *login*.

Kode Use Case	UC 001
Nama Use Case	Login
Aktor	User dan admin
Deskripsi	User melakukan login ke sistem menggunakan kredensial (email dan password) untuk mendapatkan akses ke fitur-fitur aplikasi
Kondisi Awal	User sudah terdaftar di sistem dan memilih opsi “Login”
Kondisi Akhir	User berhasil login dan mendapatkan akses ke fitur aplikasi
Alur Kejadian Normal	
Aktor	Sistem
1. User / admin memilih opsi “Login”	
	2. Sistem menampilkan halaman login
3. User / admin memasukkan kredensial (email dan password)	
4. User / admin menekan tombol “Login”	
	5. Sistem memverifikasi kredensial
	6. Sistem menampilkan halaman utama aplikasi untuk user, dan sistem akan menampilkan dasbor admin untuk admin
Alur Kejadian Alternatif	
1. Jika pengguna belum terdaftar, sistem menyediakan opsi "Register".	
2. Jika kredensial salah, sistem menampilkan pesan kesalahan	

Tabel 4. Skenario Use Case Login

Use Case Register

Skenario *use case Register* melibatkan dua aktor utama, yaitu *user* dan *admin*. Skenario ini menjelaskan langkah-langkah yang dilakukan oleh aktor ketika melakukan proses pendaftaran akun baru pada sistem. Proses ini mencakup pengisian data pribadi seperti nama, *email*, dan *password* yang akan digunakan untuk mengakses sistem di kemudian hari.

Kode Use Case	UC 002
Nama Use Case	Register
Aktor	User
Deskripsi	Pengguna baru dapat membuat akun untuk mengakses sistem inventaris.
Kondisi Awal	User sudah terdaftar di sistem dan memilih opsi “Login”
Kondisi Akhir	User berhasil membuat akun dan dapat login.
Alur Kejadian Normal	
Aktor	Sistem
1. User / admin memilih opsi “Register”	
	2. Sistem menampilkan form registrasi
3. User / admin mengisi data (nama, email, password, dll)	
4. User / admin menekan tombol “Daftar”	
	5. Sistem memvalidasi data dan menyimpan akun baru.
	6. Sistem menampilkan pesan “Registrasi berhasil”
Alur Kejadian Alternatif	
1. Jika email sudah digunakan, sistem menampilkan pesan error.	
2. Jika data tidak lengkap, sistem meminta user melengkapi form.	

Tabel 5. Skenario Use Case Register

Use Case Dashboard

Skenario *use case Dashboard* melibatkan dua aktor utama, yaitu *user* dan *admin*. Skenario ini menjelaskan proses yang terjadi setelah pengguna berhasil

melakukan *login* ke sistem. Setelah masuk, pengguna akan diarahkan ke halaman dashboard yang menjadi pusat navigasi utama dalam aplikasi.

Pada dashboard, pengguna dapat mengakses berbagai fitur seperti menambah data produk, melihat daftar produk, mengedit atau menghapus produk, melihat detail produk, serta mengelola data barang masuk dan keluar. Untuk *admin*, *dashboard* juga menampilkan ringkasan data inventaris yang lebih lengkap, termasuk status stok dan laporan aktivitas barang di data center.

Kode Use Case	UC 003
Nama Use Case	Dashboard
Aktor	<i>User</i> dan <i>admin</i>
Deskripsi	Menampilkan ringkasan data inventaris dan menu utama navigasi sistem.
Kondisi Awal	Pengguna telah <i>login</i> .
Kondisi Akhir	Dashboard ditampilkan sesuai hak akses pengguna.
Alur Kejadian Normal	
Aktor	Sistem
1. <i>User</i> / <i>admin</i> berhasil <i>login</i>	
	2. Sistem menampilkan ringkasan data barang, berupa barang masuk, keluar, lokasi penyimpanan, tipe produk dan status stok.
Alur Kejadian Alternatif	
1. Jika data gagal dimuat, sistem menampilkan pesan “Data tidak tersedia”.	

Tabel 6. Skenario Use Case Dashboard

Usecase Menambah Barang

Skenario *use case* untuk fitur Menambah Barang melibatkan dua aktor kunci, yakni *admin* dan *user* yang mendapat hak akses dalam melakukan entry data barang. Alur ini menjelaskan proses yang ditempuh oleh pengguna ketika akan memasukkan record barang baru ke dalam sistem inventaris data center.

Kode Use Case	UC 004
Nama Use Case	Menambah Barang
Aktor	Admin
Deskripsi	Admin dapat menambah data barang di sistem inventaris.
Kondisi Awal	Admin sudah login dan mengakses halaman manajemen barang.
Kondisi Akhir	Data barang berhasil disimpan, diubah, atau dihapus.
Alur Kejadian Normal	
Aktor	Sistem
1. Admin memilih menu “Manajemen Barang”	
	2. Sistem menampilkan daftar barang.
3. Admin menekan “Tambah Barang”	
	4. Sistem menampilkan form input.
5. Admin mengisi data barang	
	6. Sistem menyimpan data ke database
Alur Kejadian Alternatif	
1. Jika data gagal dimuat, sistem menampilkan pesan “Data tidak tersedia”.	

Tabel 7. Skenario Use Case Manajemen Barang

Use Case Edit Barang

Skenario ini menjelaskan langkah-langkah yang dilakukan oleh aktor ketika ingin melakukan perubahan terhadap data barang yang telah tersimpan di sistem.

Mekanisme ini berawal ketika user menentukan satu item dari inventaris yang tersedia, lalu meng-klik opsi Edit untuk mengakses form editing data. User dapat melakukan perubahan pada informasi seperti nama barang, klasifikasi, nomor seri, tempat penyimpanan, kondisi barang, atau waktu input berdasarkan kebutuhan. Setelah editing selesai dilakukan, sistem akan memvalidasi informasi yang telah diperbarui dan menyimpannya dalam database.

Kode Use Case	UC 005
Nama Use Case	Edit Barang

Aktor	<i>Admin</i>
Deskripsi	<i>Admin dapat mengedit data barang di sistem inventaris.</i>
Kondisi Awal	<i>Admin sudah login dan mengakses halaman manajemen barang.</i>
Kondisi Akhir	<i>Data barang berhasil diubah.</i>
Alur Kejadian Normal	
Aktor	Sistem
1. Admin memilih menu “Manajemen Barang”	
	2. Sistem menampilkan daftar barang.
3. Admin menekan “Edit Barang”	
	4. Sistem menampilkan form input.
5. Admin mengedit data barang	
	6. Sistem menyimpan data ke <i>database</i>
Alur Kejadian Alternatif	
1. Jika mengisi form tidak lengkap, sistem menampilkan pesan kesalahan	

Tabel 8. Skenario Use Case Edit Barang

Usecase Hapus Barang

Skenario ini menjelaskan langkah-langkah yang dilakukan oleh aktor ketika ingin menghapus data barang yang sudah tidak diperlukan atau sudah keluar dari data center.

Alur dimulai ketika user mengakses inventory list dalam sistem, lalu menentukan artikel yang akan dibuang. Selanjutnya, user menekan button Hapus dan sistem akan memunculkan window konfirmasi untuk memvalidasi bahwa tindakan tersebut benar-benar dikehendaki. Jika user mengklik tombol Ya, maka sistem akan menghilangkan data artikel dari repository dan mererefresh display inventory list.

Kode Use Case	UC 006
Nama Use Case	Hapus Barang
Aktor	<i>Admin</i>

Deskripsi	<i>Admin</i> dapat menghapus data barang di sistem inventaris.	
Kondisi Awal	Admin sudah <i>login</i> dan mengakses halaman manajemen barang.	
Kondisi Akhir	Data barang berhasil dihapus dari sistem.	
Alur Kejadian Normal		
Aktor		Sistem
1. Admin memilih menu “Manajemen Barang”		
		2. Sistem menampilkan daftar barang.
3. Admin menekan “Hapus Barang”		
		4. Sistem menampilkan alert .
		6. Sistem menghapus barang dari <i>database</i>
Alur Kejadian Alternatif		
1. Jika <i>admin</i> menghapus data, sistem menampilkan konfirmasi sebelum menghapus.		

Tabel 9. Skenario Usecase Hapus Barang

Use Case Detail Barang

Skenario use case Detail Barang melibatkan dua aktor utama, yaitu *admin* dan *user* yang memiliki hak akses untuk melihat informasi lengkap barang yang tersimpan di sistem. Skenario ini menjelaskan langkah-langkah yang dilakukan oleh aktor ketika ingin menampilkan detail data dari suatu barang yang ada dalam daftar inventaris data center.

Kode Use Case	UC 007
Nama Use Case	Detail Barang
Aktor	<i>Admin / user</i>
Deskripsi	Menampilkan informasi lengkap barang yang dipilih..
Kondisi Awal	Barang tersedia di daftar.
Kondisi Akhir	Detail barang berhasil ditampilkan.
Alur Kejadian Normal	
Aktor	Sistem

1. <i>User/Admin</i> Pengguna memilih barang dari daftar barang	
	2. Sistem menampilkan detail barang (nama, tipe, lokasi, stok,dll).
Alur Kejadian Alternatif	
-	

Tabel 10. Skenario Use Case Detail Barang

Use Case Mengelola User

Skenario *use case* Mengelola *User* melibatkan aktor utama yaitu admin. Skenario ini menjelaskan langkah-langkah yang dilakukan oleh admin ketika ingin melakukan pengelolaan terhadap data pengguna yang terdaftar dalam sistem.

Kode Use Case	UC 008
Nama Use Case	Mengelola <i>User</i>
Aktor	<i>Admin</i>
Deskripsi	Admin dapat melihat daftar <i>user</i> , mengubah data, atau menghapus akun dari sistem.
Kondisi Awal	Admin telah berhasil <i>login</i> ke <i>dashboard</i> admin.
Kondisi Akhir	Perubahan data <i>user</i> berhasil dilakukan (tambah, ubah, hapus)
Alur Kejadian Normal	
Aktor	Sistem
1. Membuka halaman manajemen <i>user</i> di <i>dashboard</i> admin	
	2. Menampilkan daftar seluruh <i>user</i> yang terdaftar
3. Admin dapat memilih untuk menambah, mengubah atau menghapus <i>user</i>	
6. Admin menekan tombol “ <i>Add New User</i> ”	
	7. Sistem menampilkan form untuk menambahkan <i>user</i>

8. Admin mengisi form untuk menambahkan <i>user</i>	
9. Admin klik tombol “Add”	
	10. <i>User</i> berhasil ditambah
11. Admin memilih untuk mengubah <i>user</i> dengan menekan ikon ubah	
	12. Sistem menampilkan form yang berisi data <i>user</i>
13. Admin mengubah data <i>user</i> dan klik tombol “Save”	
	14. Data <i>user</i> berhasil diubah
15. Admin memilih menghapus <i>user</i> dengan menekan ikon tempat sampah	
	16. Menampilkan pesan konfirmasi untuk menghapus <i>user</i>
17. Admin mengkonfirmasi penghapusan dengan menekan tombol “Yes”	
	18. <i>User</i> berhasil dihapus
Alur Kejadian Alternatif	
13.a <i>Input</i> pada formulir tidak valid, Sistem menampilkan pesan kesalahan dan meminta admin memperbaiki <i>input</i> .	

Tabel 11. Skenario Use Case Mengelola User

Use Case Laporan dan ekspor data

Skenario *use case* Generate Report dan Export melibatkan dua pelaku utama, yakni admin dan user yang diberi hak akses untuk meninjau serta mengunduh data inventaris. Skenario ini memaparkan serangkaian langkah yang dieksekusi oleh aktor saat ingin mengamati laporan aktivitas barang dan mentransfer data ke dalam format spesifik seperti PDF atau Excel untuk keperluan dokumentasi atau reporting.

Kode Use Case	UC 009
Nama Use Case	Generate dan Ekspor Laporan
Aktor	Admin/User
Deskripsi	Admin dapat menampilkan laporan barang masuk, keluar, dan stok, serta mengekspor ke PDF/Excel.
Kondisi Awal	Sudah berada di dalam halaman detail barang.
Kondisi Akhir	Laporan berhasil di tampilkan dan di ekspor
Alur Kejadian Normal	
Aktor	Sistem
1. Admin memilih menu “Laporan ”	
	2. Sistem menampilkan data berupa laporan.
3. Admin memilih opsi ekspor (PDF/Excel)	
	4. Sistem menghasilkan file laporan dan mengunduhnya
Alur Kejadian Alternatif	
-	

Tabel 12, Skenario Usecase Ekspor Laporan

Use Case Logout

Skenario *use case* untuk fitur Keluar Sistem melibatkan dua aktor primer, yaitu *user* dan *admin*. Alur skenario ini menjelaskan proses yang ditempuh oleh pengguna ketika akan mengakhiri sesi penggunaan sistem setelah menyelesaikan aktivitas di dalam aplikasi.

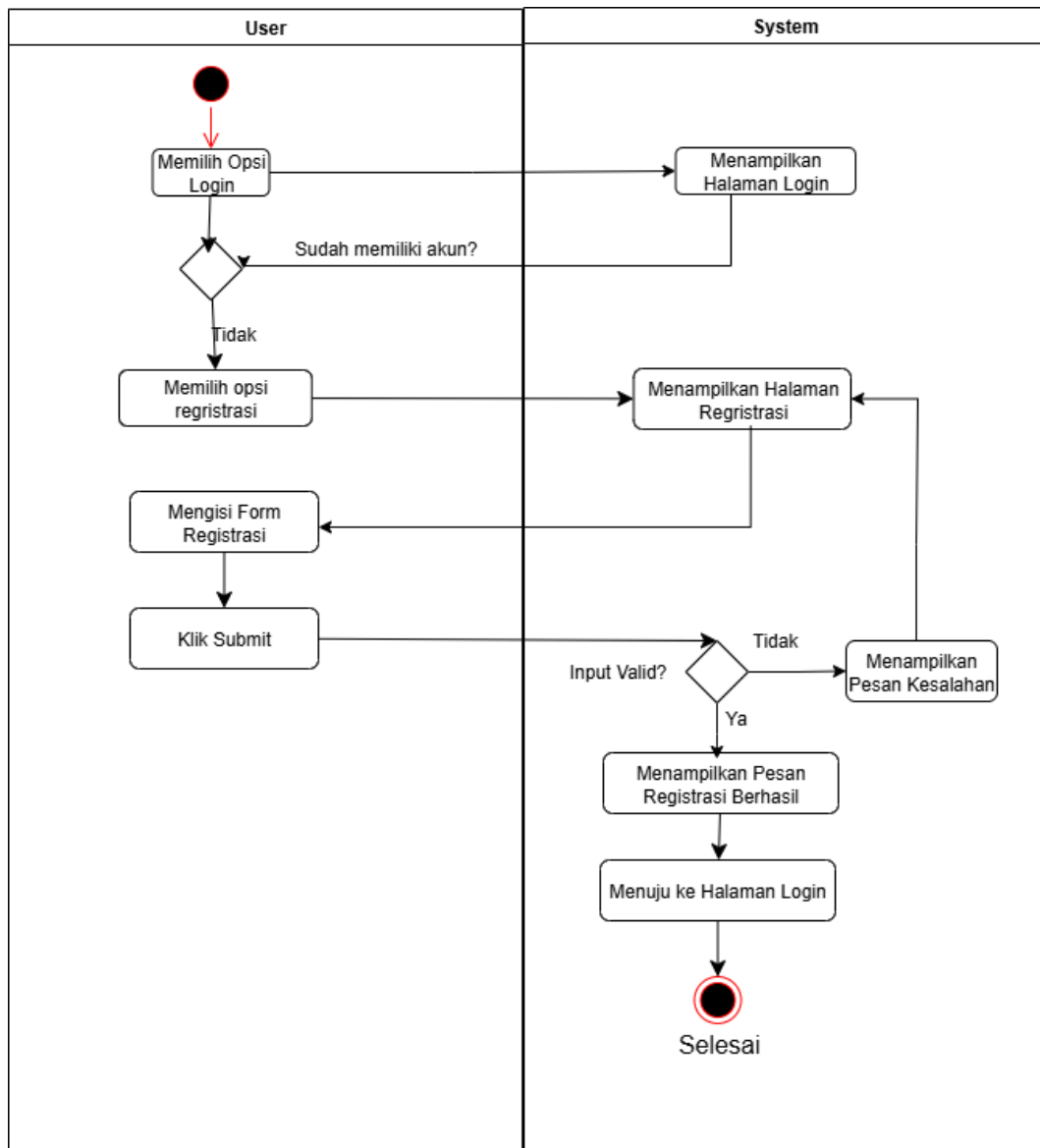
Kode Use Case	UC 010
Nama Use Case	Logout
Aktor	User dan Admin
Deskripsi	User / admin keluar dari akun mereka untuk mengakhiri sesi login di sistem.
Kondisi Awal	User / admin sedang dalam keadaan login dan berada di halaman sistem.
Kondisi Akhir	Sistem mengakhiri sesi login dan mengarahkan kembali ke halaman login.
Alur Kejadian Normal	
Aktor	Sistem
1. User atau admin melakukan login	
	2. Sistem mengarahkan user atau admin ke halaman utama masing masing
3. User atau admin memilih opsi "Logout".	
	4. Sistem meminta konfirmasi dari user atau admin untuk memastikan bahwa mereka ingin keluar dari sistem.
5. User atau admin mengonfirmasi	
	6. User atau admin diarahkan kembali ke halaman login nya masing masing.
Alur Kejadian Alternatif	
-	

Tabel 13. Skenario Use Case Logout

3.2.5 Activity diagram

Activity diagram adalah salah satu jenis diagram dalam *UML (Unified Modeling Language)* yang digunakan untuk menggambarkan alur kerja atau proses dalam sebuah sistem. Diagram ini menampilkan urutan aktivitas atau tindakan yang terjadi, dari awal hingga akhir, serta menunjukkan bagaimana aktivitas-aktivitas tersebut saling berkaitan satu sama lain

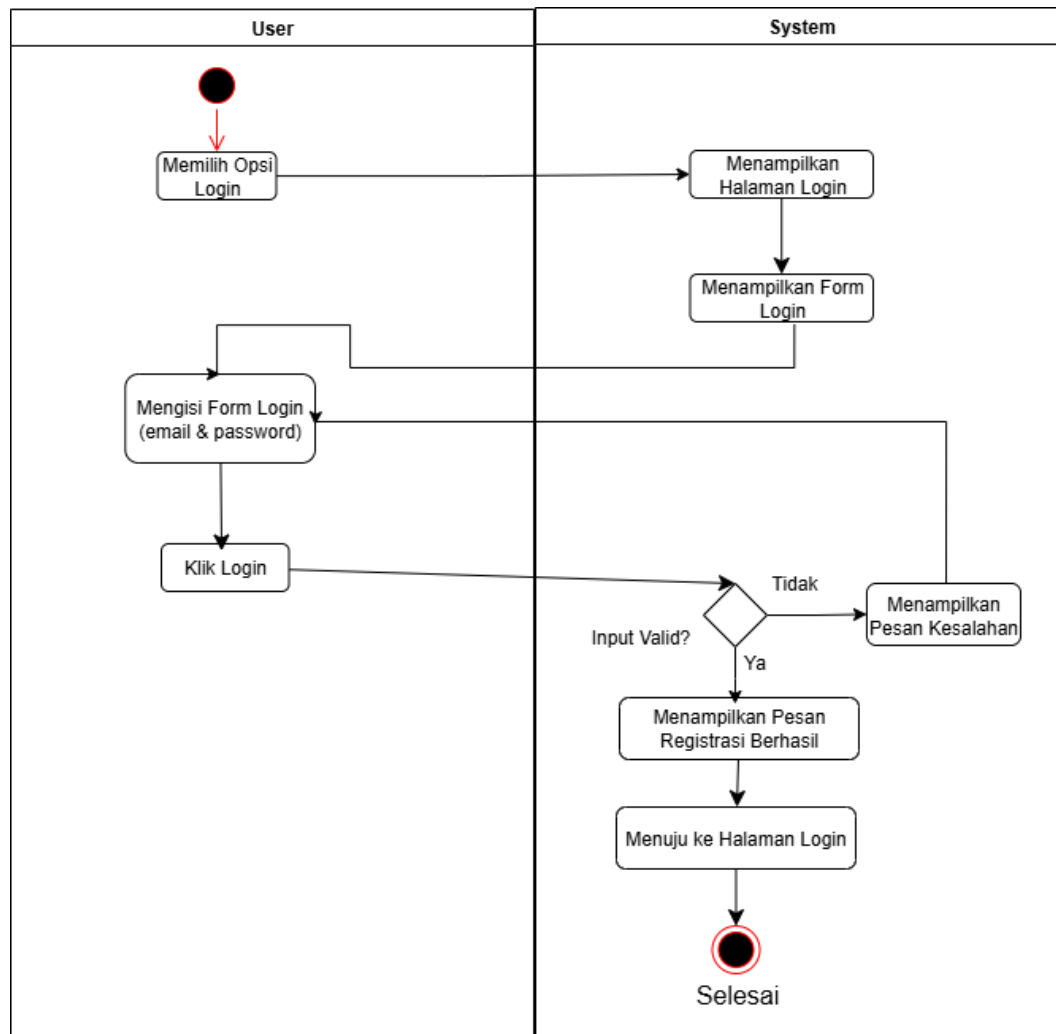
Diagram Activity Register



Gambar 5. Activity Diagram Register

Diagram ini mengilustrasikan tahapan pendaftaran user baru dalam sistem. Tahapan dimulai ketika user menentukan menu Register dan melengkapi data yang dibutuhkan seperti nama lengkap, *email*, *password*, dan konfirmasi *password*. Setelah seluruh data dientry, sistem akan melakukan pemeriksaan terhadap input user, seperti memastikan bahwa semua field sudah terisi, format email akurat, dan *password* konsisten dengan konfirmasi *password*.

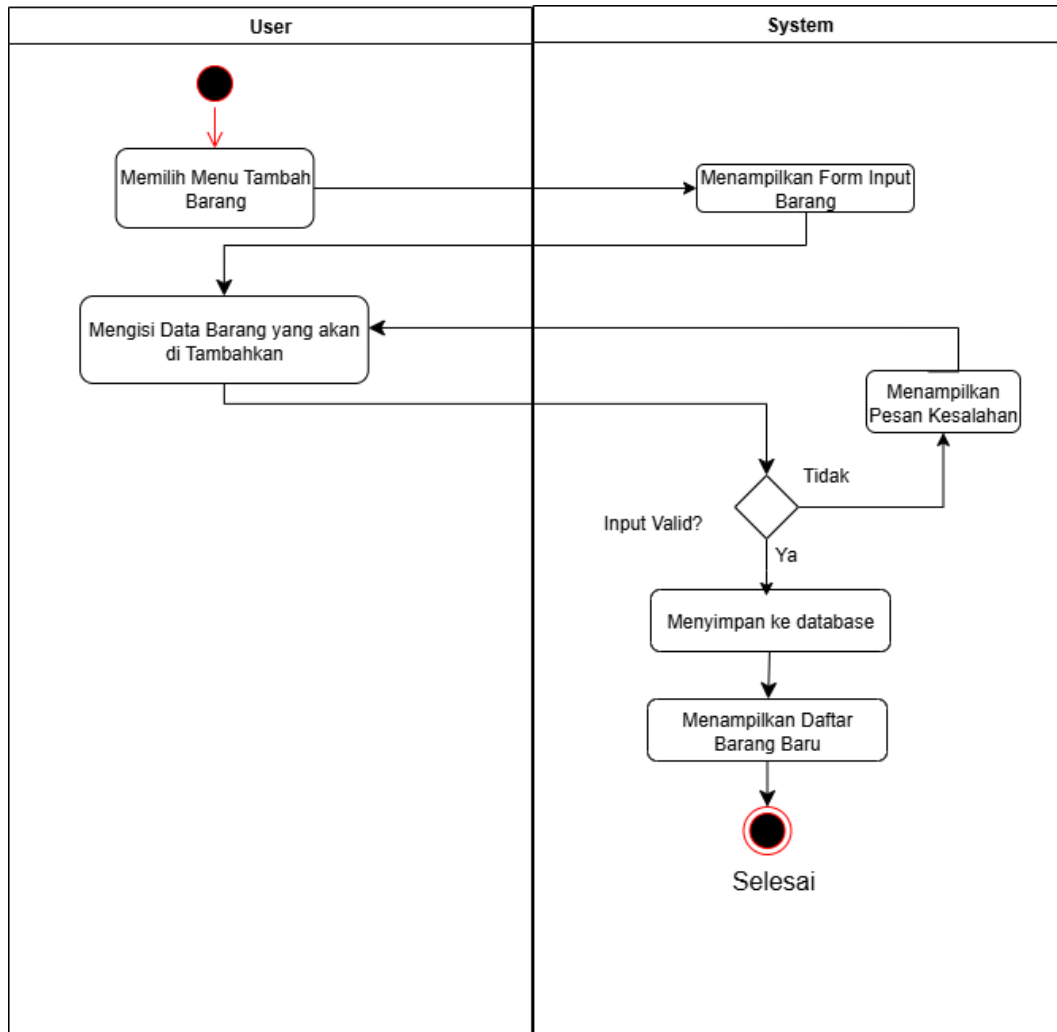
Diagram Activity Login



Gambar 6. Activity Diagram Login

Diagram ini menggambarkan proses autentikasi pengguna ketika akan mengakses sistem manajemen inventaris data center. Proses dimulai ketika pengguna membuka halaman *Login* dan memasukkan email serta *password* yang telah terdaftar sebelumnya. Setelah pengguna menekan tombol *Login*, sistem akan melakukan validasi data dengan mencocokkan kredensial yang dimasukkan terhadap data yang tersimpan di dalam database SQL Server.

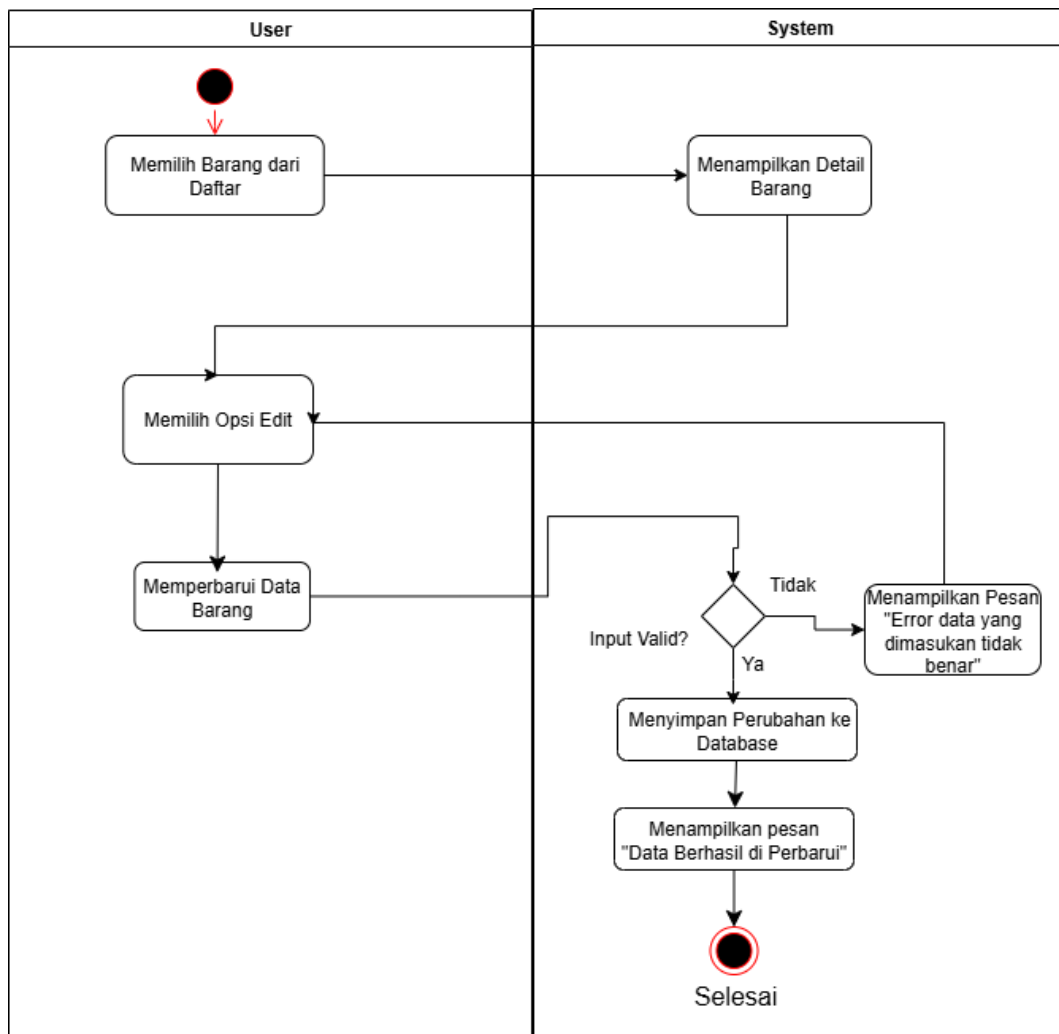
Activity Diagram Tambah Barang



Gambar 7. Activity Diagram Tambah Barang

Diagram ini menggambarkan alur proses ketika pengguna, khususnya *admin* atau petugas data center, melakukan penambahan data barang baru ke dalam sistem manajemen inventaris.

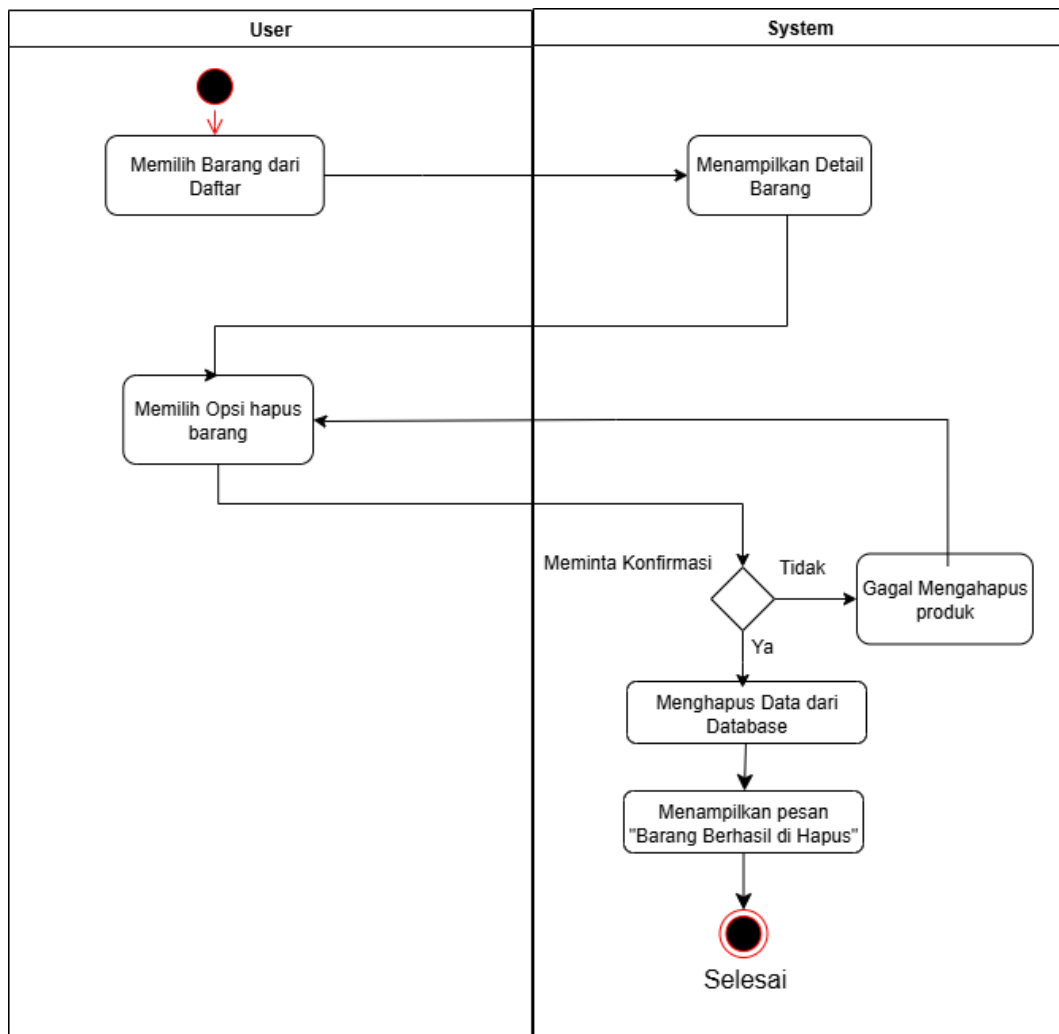
Activity Diagram Edit Barang



Gambar 8 Activity Diagram Edit Barang

Diagram ini menjelaskan alur proses ketika pengguna, khususnya admin atau petugas data center, melakukan perubahan atau pembaruan data pada barang yang sudah terdaftar di dalam sistem.

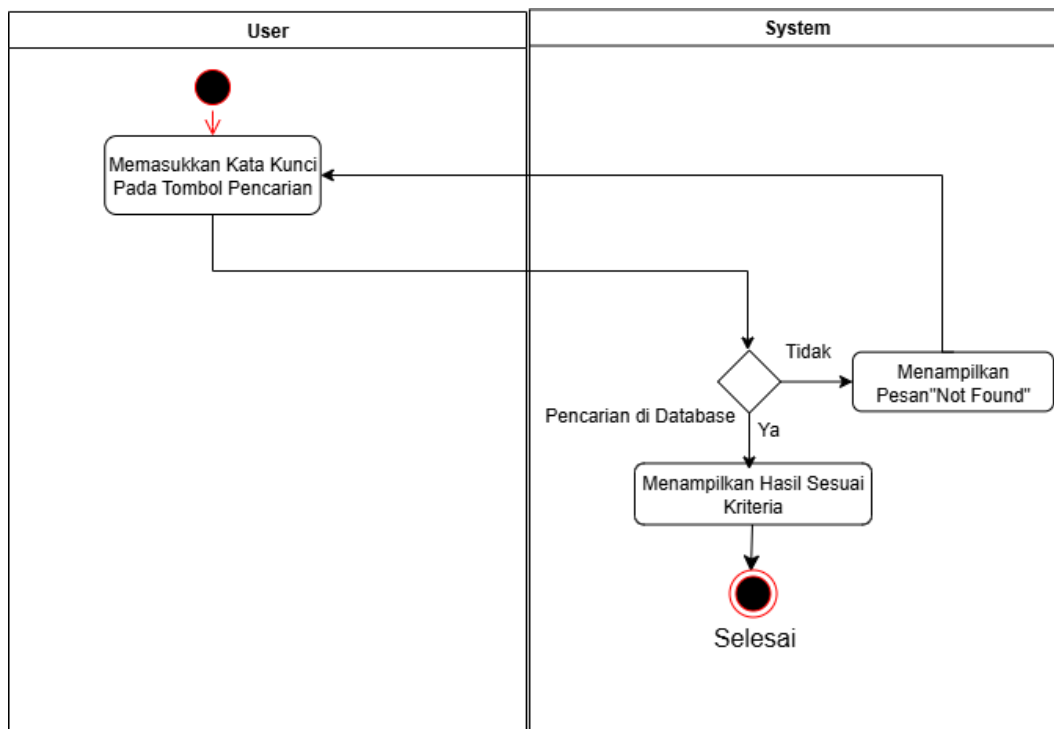
Activity Diagram Hapus Barang



Gambar 9. Activity Diagram Hapus Barang

Diagram ini menggambarkan alur proses ketika pengguna, khususnya admin atau petugas sistem, melakukan penghapusan data barang yang sudah terdaftar di dalam sistem inventaris

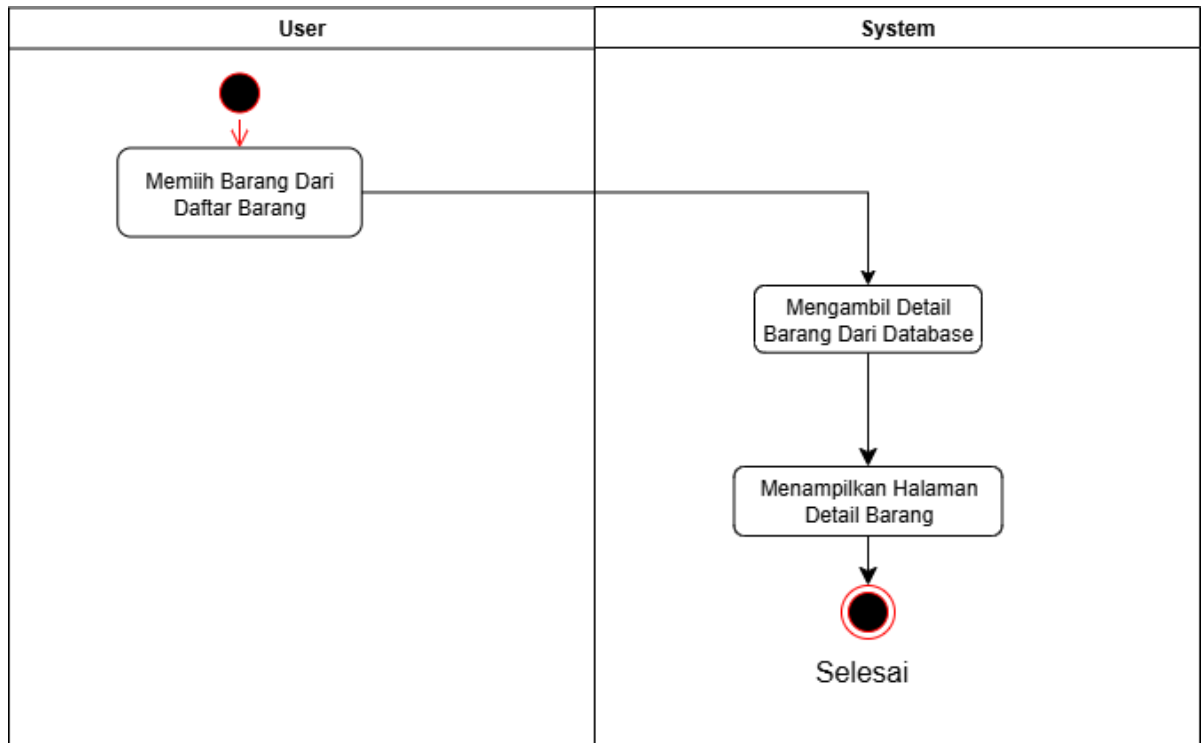
Activity Diagram Pencarian & Filter Barang



Gambar 10 Activity Diagram Pencarian & Filter Barang

Diagram ini merepresentasikan mekanisme yang dilakukan pengguna dalam mencari dan memilah data inventaris yang terdapat dalam sistem. Fungsi tersebut bertujuan untuk mempermudah pengguna dalam menemukan record barang spesifik secara cepat dan efisien tanpa harus menelusuri semua daftar inventaris secara manual.

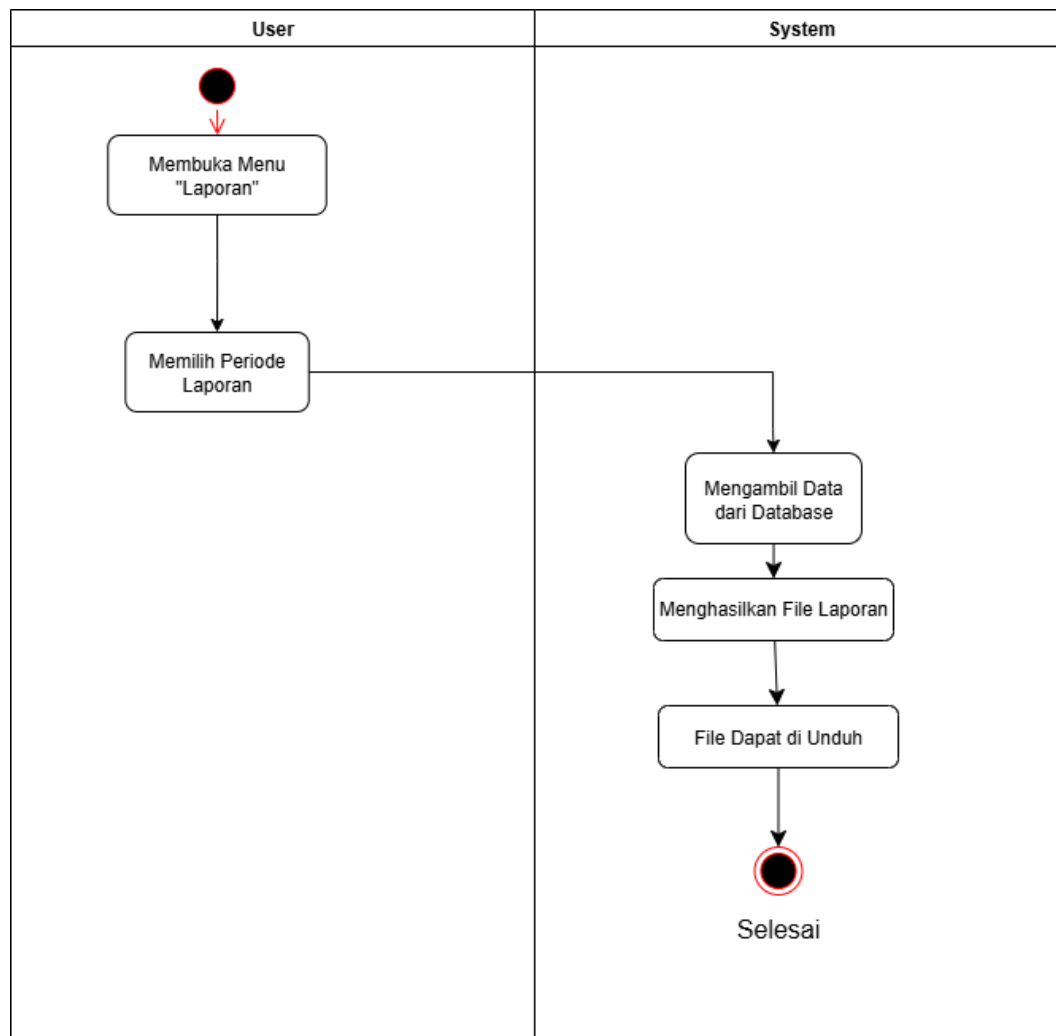
Activity Diagram Lihat Detail Barang



Gambar 11 Activity Diagram Lihat Detail Barang

Diagram tersebut menyajikan flow proses saat aktor ingin mengamati informasi yang lebih menyeluruh mengenai suatu barang yang sudah terdaftar dalam aplikasi inventaris. Sasaran dari proses ini adalah untuk memberikan detail yang komprehensif mengenai data barang, seperti nama item, kode referensi, jenis kategori, lokasi storage, kondisi barang, timestamp masuk, serta status stock.

Activity Diagram Export Laporan

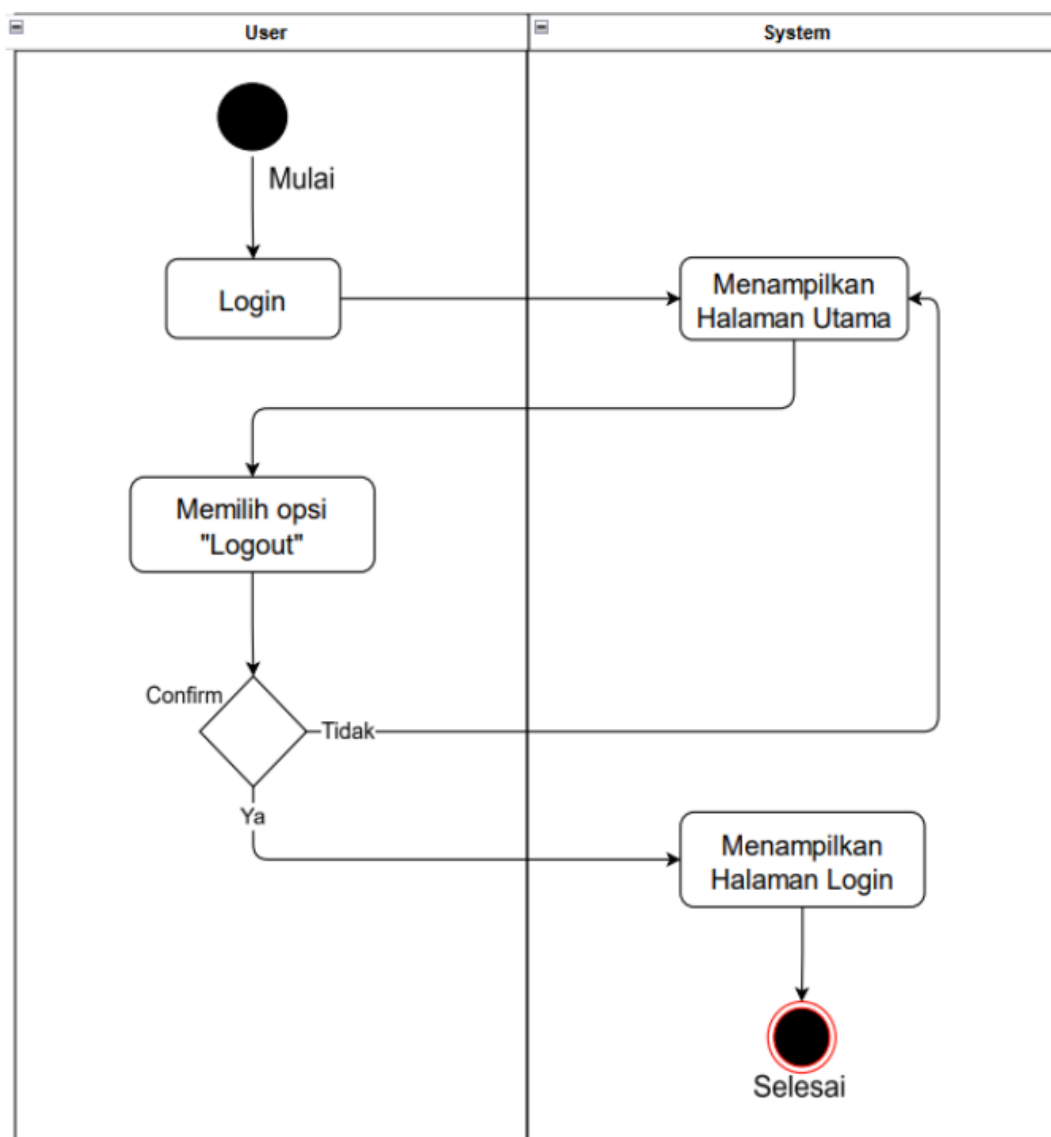


Gambar 12 Activity Diagram Export Barang

Diagram ini menggambarkan proses ekspor data laporan inventaris yang dilakukan oleh pengguna, baik dalam format PDF maupun Excel, sesuai dengan kebutuhan dokumentasi dan pelaporan di lingkungan data center. Proses ini bertujuan untuk mempermudah pengguna dalam mendapatkan ringkasan data barang yang telah

tercatat di sistem, termasuk informasi barang masuk, barang keluar, serta kondisi stok terkini.

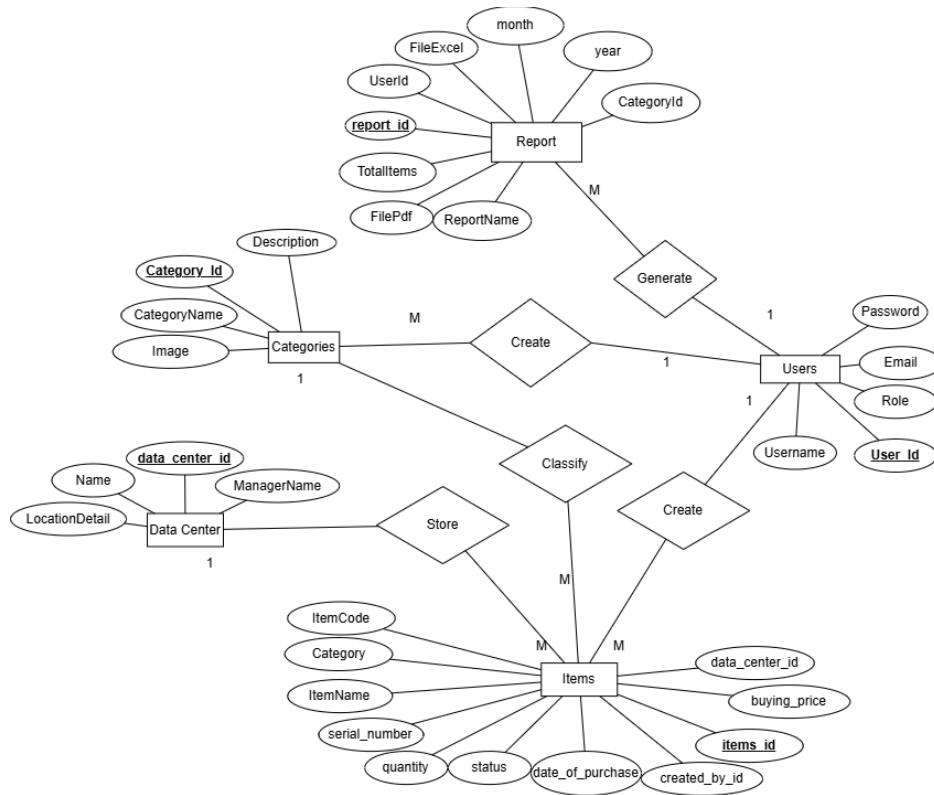
Activity Diagram Logout



Gambar 13 Activity Diagram Logout

Diagram ini menggambarkan proses keluarnya pengguna dari sistem aplikasi inventaris data center.

3.2.6 ER Diagram

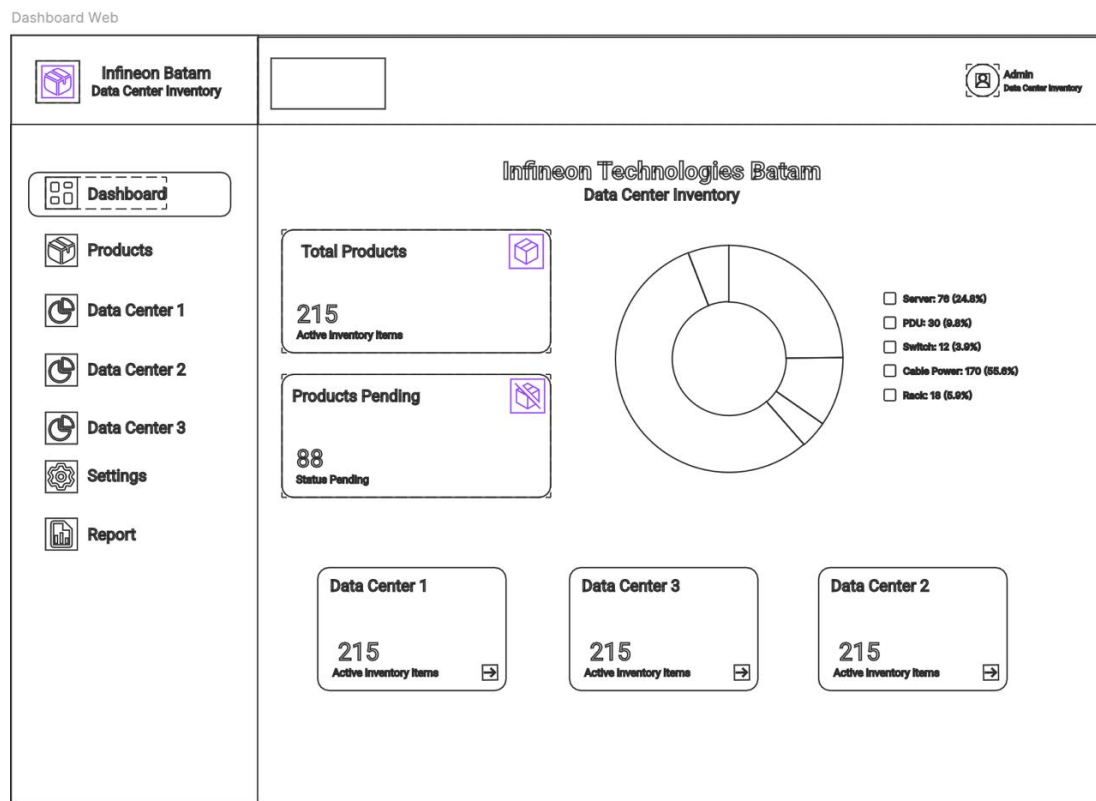


Gambar 14 ERD Diagram

Entity Relationship Diagram (ERD) adalah visualisasi konseptual yang dipergunakan untuk menampilkan relasi antar entitas dalam platform *Data Center Inventory*. ERD tersebut dikonstruksi untuk menjamin bahwa arsitektur basis data yang diimplementasikan sistem dapat saling berinteraksi, tersusun sistematis, serta mampu menunjang seluruh kebutuhan aktivitas inventaris pada data center.

3.2.7 Perancangan Antarmuka (Wireframe)

Wireframe Dashboard

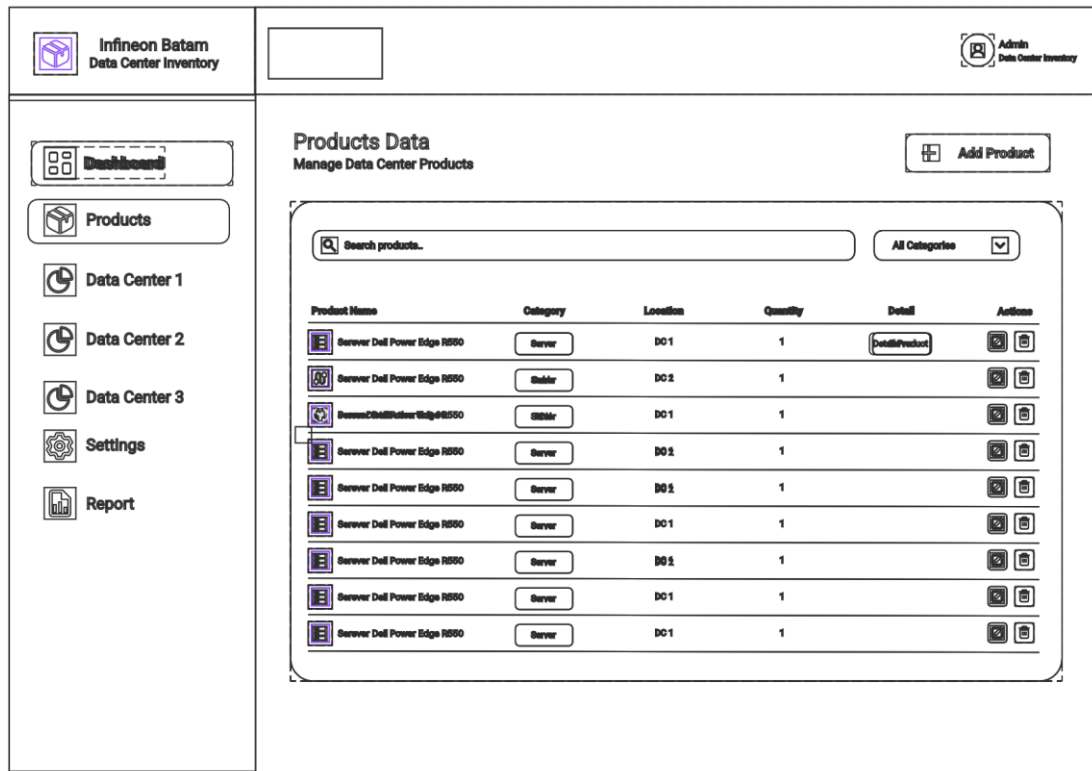


Gambar 15 Design Wireframe Dashboard

Wireframe dashboard merupakan rancangan awal tampilan antarmuka yang berfungsi sebagai gambaran konsep mengenai bagaimana informasi utama sistem akan disajikan kepada pengguna. Pada sistem *Data Center Inventory*, *dashboard* dirancang untuk menjadi halaman utama yang menampilkan ringkasan data inventaris secara informatif, sederhana, dan mudah dipahami.

Wireframe Products

Dashboard Web



Gambar 16 Design Wireframe Products

Wireframe halaman *Products* merupakan rancangan awal antarmuka yang digunakan untuk menampilkan daftar seluruh barang yang tercatat dalam sistem *Data Center Inventory*. Pada halaman ini, pengguna dapat melihat informasi penting terkait setiap item, seperti nama barang, kategori, serial number, lokasi data center, serta status barang.

Wireframe Product Detail

Dashboard Web

Infineon Batam
Data Center Inventory

Dashboard

Products

Data Center 1

Data Center 2

Data Center 3

Settings

Report

Admin
Data Center Inventory

Product Detail

Manage Data Center 1 Products

Edit Product

Primary Details

Product Name

: Server dell 1238

Product ID

: 456567890000000000

Product category

: Server

Buying Price

: Rp. 123222

Quantity

: 1

Location

: Data Center 1

Date of Purchase

: 12/02/2025

Status

: Pending

Gambar 17 Design Wireframe Product Detail

Halaman ini dirancang untuk memberikan gambaran menyeluruh terkait karakteristik dan kondisi barang sehingga memudahkan pengguna dalam proses verifikasi, pengecekan fisik, maupun audit inventaris.

BAB IV IMPLEMENTASI DAN PENGUJIAN

4.1. Hasil Implementasi

Pada tahap implementasi, dilakukan proses pengembangan sistem Data Center Inventory menjadi sebuah aplikasi berbasis web yang berfungsi untuk mendukung pengelolaan inventaris barang secara efektif dan efisien. Sistem ini mencakup berbagai fitur utama, seperti manajemen produk, kategori, laporan, dan pengaturan pengguna. Proses implementasi sistem melibatkan integrasi antarmuka pengguna, logika *backend*, dan *database*, dengan menggunakan teknologi dan *tools* yang relevan untuk mendukung performa dan fungsionalitas sistem.

4.1.1 Tools yang Digunakan untuk Impelemtasi :

Aplikasi Data Center *Inventory* dikembangkan dengan menggunakan kombinasi tools berikut untuk menunjang berbagai aspek pengembangan:

- **Figma:**

Digunakan untuk merancang prototipe dan wireframe antarmuka pengguna (UI/UX). Figma memungkinkan kolaborasi desain antar anggota tim secara *real-time*.

- **HTML (HyperText Markup Language)**

HTML digunakan untuk membentuk struktur dasar dari halaman web. Elemen-elemen seperti paragraf, tombol, *form input*, hingga *heading* disusun menggunakan tag-tag HTML yang sesuai dengan standar W3C

- **Visual Studio**

Merupakan IDE utama untuk pengembangan kode aplikasi berbasis **.NET**. Visual Studio mempermudah debugging dan pengelolaan proyek aplikasi.

- **C#**

C# merupakan bahasa pemrograman sisi server (*server-side*) yang digunakan untuk menangani logika backend, seperti mengelola data dari *form*, menghubungkan ke database, dan menghasilkan output dinamis.

- **GitHub**

Digunakan sebagai sistem manajemen kontrol versi untuk menyimpan, mendokumentasikan, dan melacak perubahan kode.

- **SQL Server**

Berfungsi sebagai sistem manajemen basis data relasional (RDBMS) untuk menyimpan semua data inventaris, kategori, pengguna, dan laporan.

- **Swagger**

Dimanfaatkan untuk dokumentasi dan pengujian endpoint API secara interaktif, memastikan API yang dibangun sesuai dengan spesifikasi sistem.

- **AutoMapper**

Dipakai untuk mempermudah pemetaan antara objek-objek dalam sistem backend yang berbeda, sehingga proses pengolahan data lebih efisien.

- **Tailwind CSS**

Framework CSS yang digunakan untuk mempercepat proses pembuatan antarmuka yang estetik dan responsif tanpa menulis kode CSS secara manual.

- **QuestPDF**

Digunakan untuk menghasilkan laporan inventaris dalam format PDF secara dinamis dan disesuaikan dengan data yang difilter oleh pengguna.

- **Draw.io**

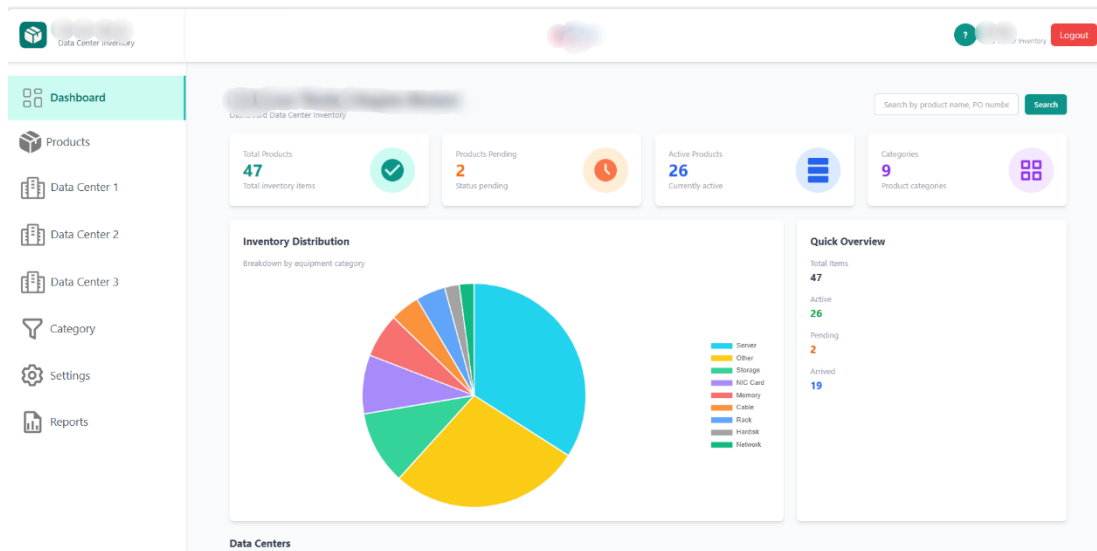
Membantu dalam membuat diagram seperti alur sistem, ERD, dan arsitektur aplikasi untuk dokumentasi teknis.

4.1.2 Tools yang Digunakan untuk Impelementasi

Sistem ini dibangun menggunakan *framework* .NET, yang mendukung pengembangan berbasis *Model-View-Controller* (MVC) untuk menghasilkan arsitektur yang terstruktur. Framework ini memberikan kemudahan dalam pengelolaan rute, middleware, serta integrasi dengan komponen lain seperti database dan API eksternal.

4.1.3 Tampilan hasil Implementasi

Dashboard Page



Gambar 18 Tampilan Dashboard

Pada tampilan dashboard sistem Data Center *Inventory*, informasi utama ditampilkan secara terstruktur untuk mendukung kemudahan pengguna dalam memantau dan mengelola data inventaris. Dashboard ini menyajikan ringkasan data inventaris meliputi jumlah total barang yang tersimpan, status barang seperti yang masih pending atau sudah aktif, serta kategori barang yang tersedia dalam sistem

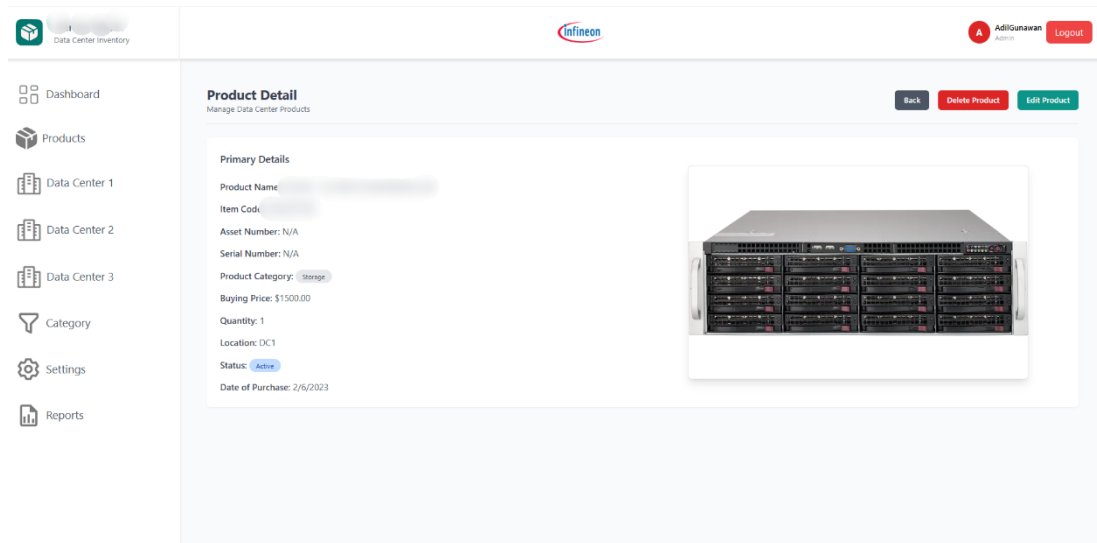
Products Page

PO Number	Product Name	Asset Number	Category	Location	Detail Product	Action
4	4x Dell R750 for oracle DB	-	Server	DC2	Detail Product	Edit Delete
4	3251 Server for LDAP refresh	-	Server	DC1	Detail Product	Edit Delete
4	64 Dell Memory Upgrade - 64 GB	-	Memory	DC1	Detail Product	Edit Delete
4	999 HPE DL360 Gen10 Plus - 4x AZ servers	-	Server	DC1	Detail Product	Edit Delete
4	13999 HPE DL360 Gen10 Plus - 4x AZ servers	48	Server	DC1	Detail Product	Edit Delete
452	32 D5224C - for SNW 45185200090 BL 89	-	Storage	DC1	Detail Product	Edit Delete
45	312 TZE-531	-	Other	DC1	Detail Product	Edit Delete
4	2 TZE-FX211	-	Other	DC1	Detail Product	Edit Delete
4	1336 D5212C (12x 4TB SATA) QID: 19505597	-	Hardisk	DC1	Detail Product	Edit Delete

Gambar 19 Tampilan Products

Pada antarmuka halaman Products, sistem menampilkan record komprehensif data barang yang tersimpan dalam inventaris Data Center *Inventory*. Antarmuka ini dikonstruksi untuk memberikan akses yang responsif dan praktis bagi pengguna untuk melihat data penting terkait setiap barang, seperti PO Number, nama produk, jenis kategori, lokasi penempatan di data center, serta status ketersediaan.

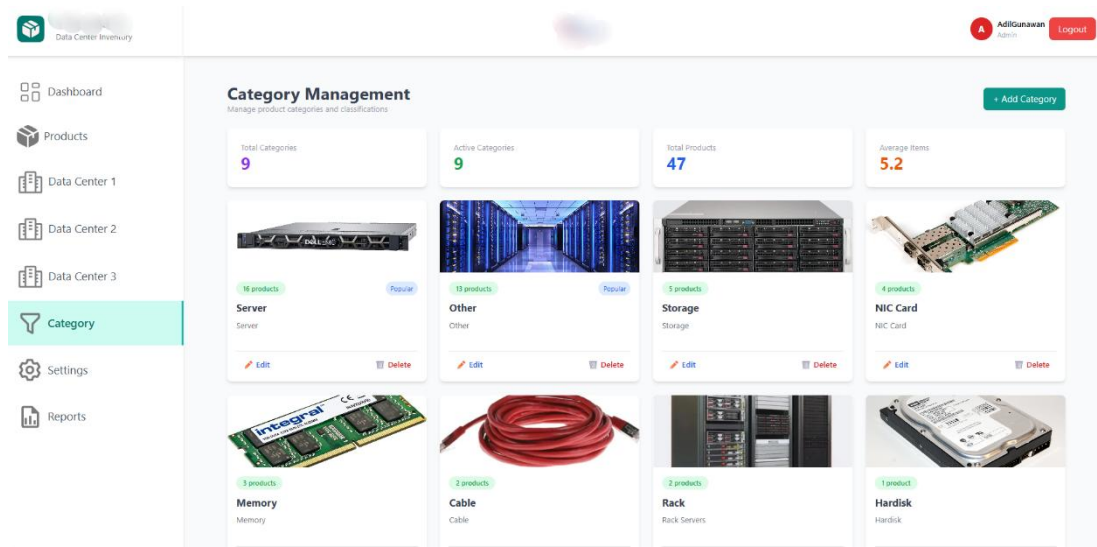
Products Detail Page



Gambar 20 Tampilan Product Detail

Halaman Product Detail berfungsi untuk menampilkan informasi lengkap terkait karakteristik dan kondisi barang yang terdaftar dalam sistem Data Center Inventory. Pada tampilan ini, detail utama barang seperti nama produk, kode item, kategori, lokasi penyimpanan, jumlah stok, harga pembelian, status barang misalnya "Arrived", dan tanggal pembelian disajikan dengan jelas. Informasi ini memudahkan pengguna dalam memverifikasi data inventaris secara rinci.

Category Product Page



Gambar 21. Tampilan Halaman Category

Halaman Category Management dirancang untuk mempermudah pengelolaan kategori produk dalam sistem Data Center Inventory. Pada halaman ini, pengguna dapat melihat jumlah total kategori, kategori yang aktif, total produk yang tercatat, serta rata-rata item per kategori. Setiap kategori seperti *Server*, *Other*, *Storage*, *NIC Card*, dan lainnya ditampilkan secara terorganisir dengan jumlah produk terkait.

Settings Page

The screenshot displays the 'Settings' page for 'Data Center Inventory'. The sidebar on the left includes links to Dashboard, Products, Data Center 1, Data Center 2, Data Center 3, Category, Settings (highlighted), and Reports. The top navigation bar shows the user 'AdiGunawan' with a 'Logout' button. The main content area is titled 'Settings' and 'Manage User Roles'. It features a summary section with 'Total Users: 9', 'Admin Users: 4', and 'Regular Users: 5'. Below this is a table of users with columns for ID, Username, Email, Role, Created, and Action.

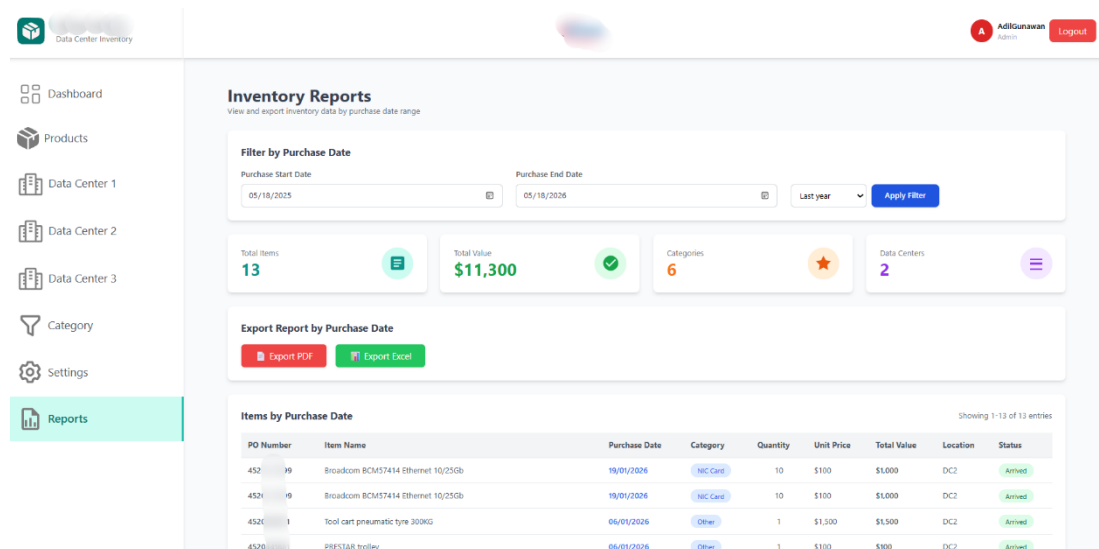
ID	Username	Email	Role	Created	Action
1		@gmail.com	Admin	1/4/2026	Admin Remove
2		md@gmail.com	User	4/29/2026	Assign Admin
3		tgmail.com	Admin	4/23/2026	Admin Remove
4		freesea.com	Admin	2/11/2026	Admin Remove
5	lunyan	@infineon.com	Admin	3/10/2026	Admin Remove
6	u3	plinfineon.com	User	2/11/2026	Assign Admin
7		.com	User	4/10/2026	Assign Admin
8		.com	User	4/23/2026	Assign Admin
9		@gmail.com	User	12/29/2023	Assign Admin

Gambar 22. Tampilan Halaman Settings

Halaman Settings dirancang untuk memudahkan pengelolaan peran pengguna dalam sistem Data Center Inventory. Pada tampilan ini, total jumlah pengguna, pengguna dengan peran admin, dan pengguna reguler ditampilkan secara terorganisir. Daftar pengguna meliputi kolom *Username*, *Email*, *Role*, serta tanggal akun dibuat.

Administrator ini memiliki hak akses yang berbeda untuk mengelola peran pengguna, seperti menugaskan peran admin melalui tombol *Assign Admin* atau menghapus pengguna dari sistem menggunakan tombol *Remove*. Dengan tata letak yang sederhana dan fungsi yang jelas, halaman ini memastikan pengelolaan dan pembaruan hak akses berjalan dengan mudah dan efisien.

Report Page



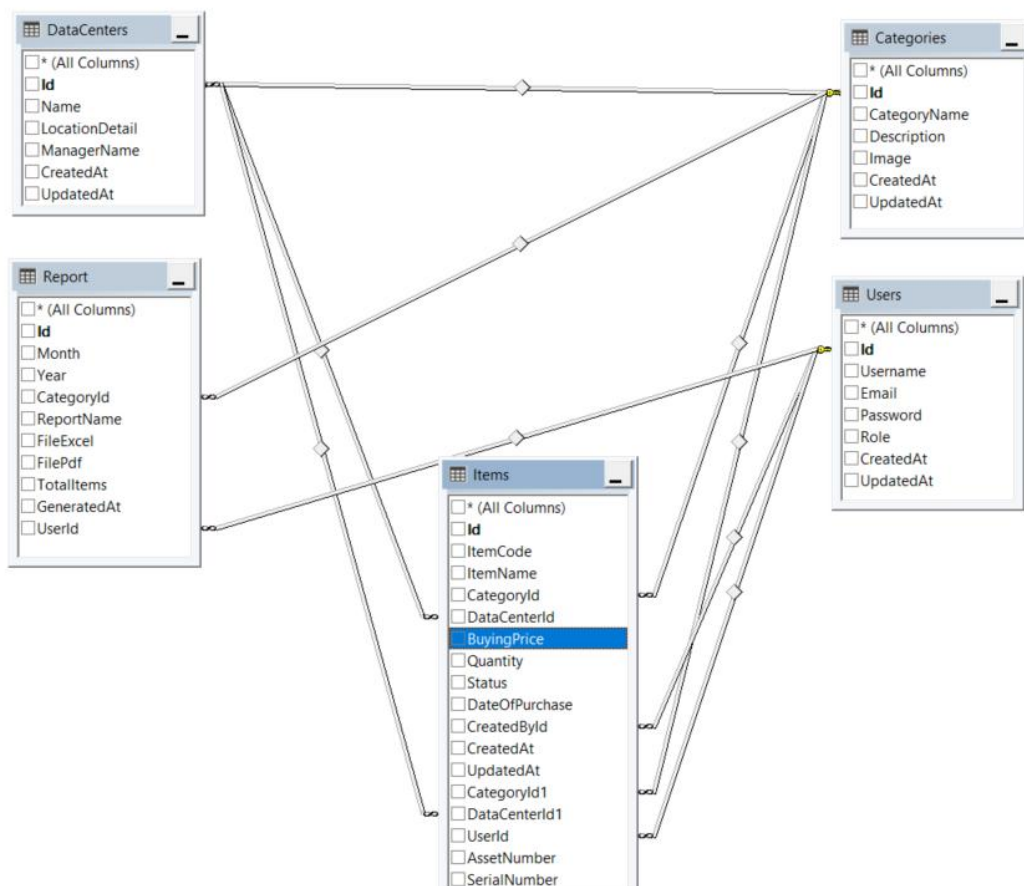
Gambar 23. Tampilan Halman Report

Halaman *Inventory Reports* dirancang untuk mempermudah pengguna dalam mengakses, memfilter, dan mengeksport data laporan inventaris berdasarkan rentang tanggal pembelian. Pada halaman ini, pengguna dapat menentukan periode laporan melalui fitur *Filter by Purchase Date*, yang memungkinkan pencarian data yang lebih spesifik sesuai kebutuhan.

Ringkasan statistik, seperti jumlah total barang *Total Items*, nilai total barang *Total Value*, jumlah kategori *Categories*, dan jumlah data center terkait, ditampilkan secara langsung untuk memberikan gambaran cepat tentang data inventaris. Selain itu, halaman ini menyediakan opsi untuk mengeksport laporan dalam format *PDF* atau *Excel*, sehingga mempermudah proses dokumentasi dan analisis lanjut.

4.1.4 Implementasi Basis Data

Implementasi basis data pada sistem Data Center *Inventory* menggunakan SQL Server sebagai manajemen basis data relasional (RDBMS) yang dipilih karena keandalannya dalam menangani data skala besar dan relasi antar tabel yang kompleks. Struktur basis data sistem ini dirancang berdasarkan kebutuhan fungsional aplikasi, sehingga setiap entitas yang teridentifikasi diwakili oleh tabel yang saling berhubungan.



Gambar 24. Implementasi Basis Data

Berdasarkan struktur relasi antar tabel yang ditampilkan pada gambar di atas, sistem ini terdiri dari sejumlah entitas inti yang berperan dalam mendukung proses pencatatan dan pengolahan data inventory. Penjelasan detail mengenai masing-masing tabel beserta fungsinya disajikan pada uraian berikut.

1. Tabel *Users*

Tabel *Users* berfungsi sebagai penyimpanan data utama dari akun pengguna yang terdaftar dalam sistem. Tabel ini memuat atribut-atribut penting yang digunakan untuk autentikasi dan manajemen pengguna, antara lain:

Id	Username	Email	Password	Role	CreatedAt	UpdatedAt
----	----------	-------	----------	------	-----------	-----------

Tabel ini menjadi komponen kunci dalam sistem untuk mengelola informasi

Gambar 25. Gambar Tabel Users

pengguna sekaligus autentikasi akses kepada berbagai fitur. Sebagai salah satu tabel inti, tabel ini terhubung dengan entitas lain yang bergantung pada data pengguna, seperti laporan dan manajemen barang.

2. Tabel *Items*

Tabel *Items* berfungsi sebagai inti dari pengelolaan inventaris dalam sistem Data Center Inventory. Tabel ini menyimpan informasi lengkap terkait data barang yang tercatat dalam data center, termasuk detail kategori barang, lokasi penyimpanan, harga, serta status barang tersebut.

Id	ItemCode	ItemName	CategoryId	DataCenter...	BuyingPrice	Quantity	Status
----	----------	----------	------------	---------------	-------------	----------	--------

Data ini memungkinkan pengguna untuk mengelola inventaris dengan detail lengkap, termasuk melihat jumlah stok, lokasi penyimpanan, dan status terkini setiap barang.

Gambar 26. Gambar Tabel Items

3. Tabel *Categories*

Tabel *Categories* digunakan untuk menyimpan data kategori produk yang tersedia dalam sistem Data Center Inventory. Tabel ini berperan penting dalam pengelompokan produk berdasarkan jenis atau fungsinya, sehingga memudahkan pengguna untuk mengelola dan mencari barang berdasarkan kategori tertentu.

Id	CategoryN...	Description	Image	CreatedAt	UpdatedAt
1	Server	Server	uploads/cat...	2023-08-20 ...	2026-03-10 ...
8	Storage	Storage	uploads/cat...	2026-03-02 ...	2026-03-10 ...
9	Other	Other	uploads/cat...	2026-03-02 ...	2026-03-10 ...
10	Memory	Memory	uploads/cat...	2026-03-02 ...	2026-03-10 ...

Gambar 27. Gambar Tabel Categories

Tabel *Categories* berfungsi sebagai referensi dalam pengelolaan data produk, memungkinkan sistem untuk mengelompokkan dan memfilter barang dengan lebih efisien.

4. Tabel Data Center

Tabel *DataCenters* digunakan untuk menyimpan informasi tentang pusat data (data center) yang menjadi lokasi penyimpanan barang dalam sistem Data Center Inventory. Tabel ini berfungsi untuk mengatur dan mencatat detail setiap data center yang dikelola dalam sistem.

Id	Name	LocationDe...	ManagerN...	CreatedAt	UpdatedAt
----	------	---------------	-------------	-----------	-----------

Gambar 28. Gambar Tabel Data Center

Tabel *DataCenters* menjadi referensi penting dalam sistem untuk mengelompokkan produk dan laporan berdasarkan lokasi data center.

5. Tabel Reports

Tabel *Reports* berfungsi sebagai tempat penyimpanan data laporan inventaris dalam sistem Data Center Inventory. Laporan ini mencakup informasi penting terkait kategori barang, total barang, dan file laporan yang dihasilkan untuk kebutuhan dokumentasi atau analisis lebih lanjut.

Id	Month	Year	CategoryId	ReportName	FileExcel	FilePdf	TotalItems	GeneratedAt	UserId
----	-------	------	------------	------------	-----------	---------	------------	-------------	--------

Gambar 29. Gambar Tabel Report

Tabel *Reports* mendukung pengelolaan laporan yang komprehensif, memungkinkan pengguna untuk menghasilkan laporan yang relevan dengan format yang fleksibel, seperti Excel atau PDF. Informasi ini digunakan untuk mendokumentasikan transaksi inventaris dan memberikan catatan yang akurat terkait data barang.

4.2. Pengujian *User Acceptance Testing* (UAT)

Pengujian sistem merupakan langkah vital dalam pembuatan aplikasi untuk memastikan bahwa semua fitur berfungsi sesuai requirement. Pada proyek ini, pengujian diimplementasikan menggunakan metode *blackbox* testing untuk menguji functionality sistem dari sudut pandang pengguna tanpa melihat struktur kode. Setelah semua pengujian teknis selesai, dijalankan juga *User Acceptance Testing* (UAT) oleh dosen pembimbing sebagai bentuk validasi dan konfirmasi bahwa aplikasi telah mencapai standar kelayakan operasional.

Pengujian dilakukan pada beberapa fitur utama sistem, yang masing-masing diuji berdasarkan skenario *input* dan *output* menggunakan pendekatan *blackbox*. Rincian pengujian disajikan sebagai berikut:

1. Login

- a. Fitur yang di uji : *Login*
- b. Tanggal Pengujian : 29-03-2026
- c. Lingkungan Pengujian:
 - Hardware : Laptop HP
 - Software : Microsoft Edge, Windows11
- d. Nama dan Peran Penguji:
 - Nama : Adil
 - Peran : *Software Tester*
- e. Deskripsi
Fitur login merupakan fungsi yang memungkinkan pengguna untuk mengakses aplikasi menggunakan akun berupa *email* dan *password* yang valid.
- f. Langkah - Langkah:
 - 1. Pengguna membuka aplikasi.
 - 2. Pengguna menekan tombol "*Login*".
 - 3. Pengguna memasukkan email dan *password*

g. Alternative Flow:

Jika email atau password yang dimasukkan tidak valid, maka sistem akan menampilkan pesan kesalahan dan mengarahkan pengguna kembali ke halaman logi

Fitur 1: Login						
Test ID	Input Email	Partisi	Input Password	Partisi	Hasil yang Diharapkan	Status
TC-01-001	adilgun@gmail.com	Valid	11111111	Valid	Berhasil	Sukses
TC-01-002	adilgun@gmail.com	Valid	12121212	Invalid	Login Gagal, "Invalid Email or Password"	Sukses
TC-01-003	adilgun@gmail.com	Valid	(kosong)	-	Login Gagal, "Password is required"	Sukses
TC-01-004	adilgun@gmail.com	Invalid	11111111	Valid	Login Gagal, "Invalid Email or Password"	Sukses
TC-01-005	adilgun@gmail.com	Invalid	12121212	Invalid	Login Gagal, "Invalid Email or Password"	Sukses
TC-01-006	(kosong)	-	11111111	Valid	Login Gagal, "Email is required"	Sukses
TC-01-007	(kosong)	-	(kosong)	-	Login Gagal, "Username and password are required"	Sukses

Tabel 14. Pengujian Fitur Login

2. Registrasi

- a. Fitur yang di uji : *Login*
b. Tanggal Pengujian : 29-03-2026
c. Lingkungan Pengujian:

- Hardware : Laptop HP
- Software : Microsoft Edge, Windows11

d. Nama dan Peran Penguji:

- Nama : Adil
- Peran : Software Tester

e. Deskripsi:

Fitur login merupakan fungsi yang memungkinkan pengguna untuk mengakses aplikasi menggunakan akun berupa *email* dan *password* yang valid.

f. Langkah - Langkah:

1. Pengguna membuka aplikasi.
2. Pengguna menekan tombol "*Login*".
3. Pengguna memasukkan *email* dan *password*

g. Alternative Flow:

Jika *email* atau *password* yang dimasukkan tidak valid, maka sistem akan menampilkan pesan kesalahan dan mengarahkan pengguna kembali ke halaman login.

Fitur 2: Registrasi										
Test ID	Input Username	Partisi	Input Email	Partisi	Input Password	Partisi	Input Confirm Password	Partisi	Hasil yang diharapkan	Status
TC-02-001	Adil	Valid	Adilgun@gmail.com	Valid	11111111	Valid	11111111	Valid	Registrasi Berhasil	Sukses
TC-02-002	Adil	Valid	Adilgun@gmail.com	Valid	11111111	Valid	12121212	Invalid	Registrasi Gagal	Sukses
TC-02-003	Adil	Valid	Adilgun@gmail.com	Valid	12121212	Invalid	12121212	Invalid	Registrasi Gagal	Sukses
TC-02-004	Adil	Valid	Adilgun@gmail.com	Valid	12121212	Invalid	11111111	Valid	Registrasi Gagal	Sukses
TC-02-005	Adil	Valid	Adilgun@gmail.com	Invalid	11111111	Valid	11111111	Valid	Registrasi Gagal	Sukses
TC-02-006	(kosong)	-	Adilgun@gmail.com	Valid	11111111	Valid	11111111	Valid	Registrasi Gagal	Sukses

TC-02-007	(kosong)	-	(kosong)	-	11111111	Valid	11111111	Valid	Registrasi Gagal	Sukses
TC-02-008	(kosong)	-	(kosong)	-	12121212	Invalid	12121212	Invalid	Registrasi Gagal	Sukses
TC-02-009	(kosong)	-	(kosong)	-	11111111	Valid	12121212	Invalid	Registrasi Gagal	Sukses
TC-02-010	Adil	Valid	Adilgun@gmail.com	Valid	11111111	Valid	12121212	Invalid	Registrasi Gagal	Sukses
TC-02-011	(kosong)	-	Adilgun@gmail.com	Valid	11111111	Valid	11111111	Valid	Registrasi Gagal	Sukses
TC-02-012	Adil	Valid	Adilgun@gmail.com	Valid	11111111	Valid	(kosong)	Invalid	Registrasi Gagal	Sukses
TC-02-013	Adil	Valid	(kosong)	-	11111111	Valid	11111111	Valid	Registrasi Gagal	Sukses
TC-02-014	Adil	Valid	(kosong)	-	(kosong)	Invalid	11111111	Valid	Registrasi Gagal	Sukses
TC-02-015	Adil	Valid	Adilgun@gmail.com	Valid	(kosong)	Invalid	(kosong)	Invalid	Registrasi Gagal	Sukses
TC-02-016	Adil	Valid	Adilgun@gmail.com	Valid	(kosong)	Invalid	(kosong)	Invalid	Registrasi Gagal	Sukses

Tabel 15. Pengujian Fitur Registrasi

3. Fitur Input Barang

- a. Fitur yang di uji : Input Barang
- b. Tanggal Pengujian : 29-03-2026
- c. Lingkungan Pengujian:
 - Hardware : Laptop HP
 - Software : Microsoft Edge, Windows11
- d. Nama dan Peran Penguji:
 - Nama : Adil
 - Peran : Software Tester
- e. Deskripsi:

Fitur Input Barang adalah fungsi utama yang memungkinkan pengguna untuk memasukkan data barang baru ke dalam sistem inventaris.

f. Langkah - Langkah:

1. Pengguna membuka aplikasi sistem inventaris.
2. Login menggunakan akun *admin*.
3. Pilih menu Manajemen Barang dari dashboard.
4. Klik tombol Tambah Barang untuk membuka form input barang baru.
5. Masukkan data pada atribut sebagai berikut:
 - Nama Barang: Nama barang terkait (wajib)
 - PO Number: Nomor PO barang (wajib)
 - Kategori: Pilih kategori barang (wajib)
 - Harga Barang: Input harga barang (wajib)
 - Jumlah Barang (Quantity): Input jumlah barang (wajib)
 - Lokasi Penyimpanan: Pilih lokasi penyimpanan barang (wajib)
 - Status Barang: Pilih status barang antara *Active* atau *Arrived* (wajib)
6. Klik tombol Simpan untuk menyimpan data barang ke dalam sistem.
7. Sistem menampilkan pesan hasil:
8. Jika data valid, sistem menyimpan data barang dan menampilkan notifikasi sukses.
9. Jika data tidak valid, sistem memberikan pesan error sesuai atribut yang bermasalah (misalnya, "Lokasi Penyimpanan harus diisi", "Harga Barang harus berupa angka", dll.).

g. Alternative Flow:

1. Jika pengguna tidak melengkapi seluruh atribut wajib, maka sistem akan menampilkan pesan kesalahan dan meminta pengguna untuk memperbaiki input. Contoh:
 - "Nama Barang harus diisi" jika nama barang kosong.
 - "PO Number wajib diisi", jika PO number tidak diinput.
 - "Kategori tidak valid" jika kategori tidak sesuai dengan pilihan yang tersedia.
2. Jika terjadi gangguan seperti koneksi ke database gagal, sistem menampilkan pesan "Data gagal disimpan. Silakan coba lagi nanti."
3. Pengguna dapat menyimpan kembali data setelah memperbaiki informasi yang salah atau setelah gangguan teratasi.

Fitur 3 : Input Barang									
Test ID	Nama Barang	PO Number	Kategori Barang	Harga	Quantity	Lokasi Penyimpanan	Status	Hasil Yang Diharapkan	Status
TC-03-001	Server Blade HPE	9999999	Server	\$1000	1	DC1	Pending	Barang berhasil di simpan	Sukses
TC-03-002	(Kosong)	9999999	Server	\$1000	1	DC1	Pending	Gagal, pesan: "Nama barang harus diisi."	Sukses
TC-03-003	Server Blade HPE	(Kosong)	Server	\$1000	1	DC1	Pending	Gagal, pesan: "PO number harus di isi"	Sukses
TC-03-004	Server Blade HPE	9999999	(Kosong)	\$1000	1	DC1	Pending	Gagal, pesan : "Kategori tidak valid"	Sukses
TC-03-005	Server Blade HPE	9999999	Server	\$-1000	1	DC1	Pending	Gagal, pesan : "Jumlah harga harus di isi"	Sukses
TC-03-006	Server Blade HPE	9999999	Server	\$1000	-1	DC1	Pending	Gagal, pesan : "jumlah barang tidak valid"	Sukses
TC-03-007	Server Blade HPE	9999999	Server	\$1000	1	(Kosong)	(Kosong)	Gagal, pesan : "Lokasi penyimpanan harus di isi"	Sukses
TC-03-008	Server Blade HPE	9999999	Server	\$1000	1	DC1	Pending	Gagal, pesan : "Status barang harus di isi"	Sukses

Tabel 16. Pengujian Fitur Input Barang

4. Fitur Edit Barang

- a. Fitur yang di uji : Edit Barang
- b. Tanggal Pengujian : 29-03-2026
- c. Lingkungan Pengujian:
 - Hardware : Laptop HP
 - Software : Microsoft Edge, Windows11
- d. Nama dan Peran Penguji:
 - Nama : Adil
 - Peran : Software Tester
- e. Deskripsi:

Fitur Input Barang adalah fungsi utama yang memungkinkan pengguna untuk memasukkan data barang baru ke dalam sistem inventaris.

- f. Langkah - Langkah:
 1. Pengguna login ke dalam sistem sebagai Admin.
 2. Dari menu utama, pengguna memilih Manajemen Barang.
 3. Sistem menampilkan daftar barang yang tersimpan.
 4. Pengguna memilih barang yang akan diubah dari daftar barang.
 5. Pengguna menekan tombol Edit Barang.
 6. Sistem menampilkan form Edit Barang yang telah terisi data sebelumnya.
 7. Pengguna memperbarui data tertentu, misalnya nama barang, harga barang, atau kategori.
 8. Setelah selesai, pengguna menekan tombol Simpan.
 9. Sistem memvalidasi data input:
 - Jika data valid, sistem menyimpan perubahan di database dan menampilkan pesan "Data berhasil diperbarui".
 - Jika data tidak valid, sistem memberikan pesan error sesuai dengan input yang salah, seperti:
 - "Nama Barang harus diisi."
 - "Harga Barang harus berupa angka positif."
- g. Alternative Flow:
 1. Apabila admin tidak melengkapi data wajib saat menekan Simpan, sistem akan memberikan notifikasi bahwa data perlu diperbaiki.

2. Jika sistem gagal menyimpan data ke database karena gangguan teknis, maka sistem akan menampilkan pesan error dan memberikan opsi untuk mencoba lagi tanpa kehilangan perubahan yang telah dibuat.

Fitur 4 : Edit Barang									
Test ID	Nama Barang	PO Number	Kategori Barang	Harga	Quantity	Lokasi Penyimpanan	Status	Hasil Yang Diharapkan	Status
TC-04-001	Server Blade HPE	9999999	Server	\$1000	1	DC1	Pending	Barang berhasil di simpan	Sukses
TC-04-002	(Kosong)	9999999	Server	\$1000	1	DC1	Pending	Gagal, pesan: "Nama barang harus diisi."	Sukses
TC-04-003	Server Blade HPE	(Kosong)	Server	\$1000	1	DC1	Pending	Gagal, pesan: "PO number harus di isi"	Sukses
TC-04-004	Server Blade HPE	9999999	(Kosong)	\$1000	1	DC1	Pending	Gagal, pesan : "Kategori tidak valid"	Sukses
TC-04-005	Server Blade HPE	9999999	Server	\$-1000	1	DC1	Pending	Gagal, pesan : "Jumlah harga harus di isi"	Sukses
TC-04-006	Server Blade HPE	9999999	Server	\$1000	-1	DC1	Pending	Gagal, pesan : "jumlah barang tidak valid"	Sukses
TC-04-007	Server Blade HPE	9999999	Server	\$1000	1	(Kosong)	(Kosong)	Gagal, pesan : "Lokasi penyimpanan harus di isi"	Sukses
TC-04-008	Server Blade HPE	9999999	Server	\$1000	1	DC1	Pending	Gagal, pesan : "Status barang harus di isi"	Sukses

Tabel 17. Pengujian Fitur Edit Barang

5. Fitur Hapus Barang

- a. Fitur yang di uji : Hapus Barang
- b. Tanggal Pengujian : 29-03-2026
- c. Lingkungan Pengujian:
 - Hardware : Laptop HP
 - Software : Microsoft Edge, Windows11
- d. Nama dan Peran Penguji:
 - Nama : Adil
 - Peran : Software Tester
- e. Deskripsi:

Fitur Hapus Barang memungkinkan admin untuk mengelola data barang di sistem Data Center Inventory dengan menghapus data barang yang sudah tidak diperlukan atau sudah keluar dari data center.

- f. Langkah - Langkah:
 1. Admin memilih menu Manajemen Barang.
 2. Sistem menampilkan daftar barang yang tersimpan di dalam inventaris.
 3. Admin memilih barang yang akan dihapus, kemudian menekan tombol Hapus Barang.
 4. Sistem menampilkan form konfirmasi untuk memastikan bahwa barang tersebut benar-benar akan dihapus.
 5. Admin mengonfirmasi penghapusan dengan menekan tombol Ya.
 6. Sistem menghapus barang dari database dan menampilkan pesan "Barang berhasil dihapus.

Fitur 5 : Hapus Barang				
Test ID	Nama Barang	Keputusan Pada Dialog Konfirmasi	Hasil Yang Diharapkan	Status
TC-05-001	Server Blade HPE	Ya	Barang berhasil di hapus	Sukses
TC-05-002	Server Blade HPE	Batal	Barang tidak di hapus	Sukses
TC-05-003	Monitor(barang tidak ada)	Ya	Gagal, pesan: “Darta tidak di temukan”	Sukses
TC-05-004	Server Blade HPE	Ya(koneksi gagal)	Gagal, pesan :“Data gagal di hapus”	Sukses

Tabel 18. Pengujian Fitur Hapus Barang

6. Fitur Input Barang

- a. Fitur yang di uji : *Reports* Barang
- b. Tanggal Pengujian : 29-03-2026
- c. Lingkungan Pengujian:
 - Hardware : Laptop HP
 - Software : Microsoft Edge, Windows11
- d. Nama dan Peran Penguji:
 - Nama : Adil
 - Peran : Software Tester
- e. Deskripsi:

Halaman *Report Products* dirancang untuk mempermudah pengguna dalam membuat laporan inventaris berdasarkan kategori, waktu pembelian, atau jumlah produk.

- f. Langkah - Langkah:
 1. *Admin login* ke dalam aplikasi dengan kredensial yang valid.
 2. Admin memilih menu *Reports* dari dashboard.
 3. Sistem menampilkan halaman *Report Products*, lengkap dengan form input filter (tanggal awal dan akhir) serta opsi ekspor laporan.
 4. Admin memasukkan rentang tanggal dan menekan tombol *Filter*.
 5. Sistem menampilkan daftar produk yang sesuai dengan kriteria filter.
 6. Admin memilih opsi ekspor (PDF atau Excel) untuk menyimpan laporan.
 7. Sistem mengeksport laporan sesuai dengan format yang dipilih dan menampilkan pesan sukses, "Laporan berhasil diekspor."
- g. Alternative Flow:

Filter Tanpa Data Hasil

- Jika data tidak ditemukan untuk kriteria filter yang diberikan (misalnya, tidak ada barang dalam periode tersebut), sistem menampilkan pesan, "Tidak ada data yang ditemukan untuk kriteria ini."
- Admin tetap dapat memperbaiki filter yang dimasukkan dan mencoba kembali.

Test ID	Tanggal Awal	Tanggal Akhir	Format Ekspor	Kriteria Data	Hasil Yang Diharapkan	Status
TC-06-001	2023-01-01	2023-12-31	PDF	Data tersedia	Berhasil mengekspor laporan ke PDF	Sukses
TC-06-002	2023-01-01	2023-12-31	Excel	Data tersedia	Berhasil Mengekspor laporan ke Excel	Sukses
TC-06-003	2023-01-01	2023-02-31	PDF	Data tidak di temukan	Menampilkan pesan: “Tidak ada data yang ditemukan”	Sukses
TC-06-004	2023-01-01	2023-12-31	PDF	-	Gagal memfilter, pesan: “Tanggal akhir harus di isi”	Sukses
TC-06-005	2023-01-01	2023-12-31	PDF	-	Gagal memfilter, pesan: “Tanggal awal harus di isi”	Sukses
TC-06-006	2023-12-31	2023-01-01	Excel	-	Gagal memfilter, pesan: “Tanggal awal tidak boleh lebih besar dari tanggal akhir”	Sukses
TC-06-007	2023-01-01	2023-12-31	(Kosong)	Data tersedia	Gagal, pesan :“Format file ekspor harus di pilih”	Sukses
TC-06-008	2023-01-01	2023-12-31	PDF	Koneksi server gagal	Gagal, pesan :“Ekspor laporan gagal”	Sukses

Tabel 19. Pengujian Fitur Report Barang

7. Fitur Filter Kategori Barang

- a. Fitur yang di uji : *Filter Category*
- b. Tanggal Pengujian : 29-03-2026
- c. Lingkungan Pengujian:
 - Hardware : Laptop HP
 - Software : Microsoft Edge, Windows11
- d. Nama dan Peran Penguji:
 - Nama : Adil
 - Peran : Software Tester
- e. Deskripsi:

Fitur Filter Kategori berfungsi untuk mempermudah pengguna dalam menemukan barang tertentu berdasarkan kategori yang relevan. Proses filter melibatkan pemilihan kategori yang tersedia di dalam sistem.

- f. Langkah - Langkah:
 1. Pengguna *login* ke dalam sistem dengan akun yang valid.
 2. Pengguna membuka halaman *Products*.
 3. Sistem menampilkan semua barang di dalam daftar (default semua kategori barang ditampilkan).
 4. Pengguna memilih kategori tertentu dari menu filter kategori (contoh: Asset).
 5. Pengguna menekan tombol Terapkan Filter.
 6. Sistem memproses input dan menampilkan barang yang sesuai dengan kategori yang dipilih.
 7. Sistem menampilkan pesan "Filter berhasil diterapkan."
- g. Alternative Flow:
 1. Jika data barang untuk kategori yang dipilih tidak ditemukan, sistem menampilkan pesan: "Tidak ada data ditemukan untuk kategori ini."
 2. Daftar barang akan kosong, dan pengguna dapat memilih kategori lain.

Fitur: Filter Kategori Barang				
Test ID	Filter Kategori	Filter Lokasi	Hasil Yang Diharapkan	Status
TC-07-001	Nic card	DC 2	Menampilkan Barang kategori Nic card di di DC 2	Sukses
TC-07-002	Server	DC 1	Menampilkan Barang kategori Server di di DC 1	Sukses
TC-07-003	Monitor(kategori tidak ada)	DC 3	Gagal, pesan: “ Kategori tidak valid”	Sukses
TC-07-004	(Kosong)	DC 2	Menampilkan semua barang yang ada di DC 2	Sukses

Tabel 20. Pengujian Fitur Filter Kategori Barang

8. Detail Barang

- a. Fitur yang di uji : Detail Barang
- b. Tanggal Pengujian : 29-03-2026
- c. Lingkungan Pengujian:
 - Hardware : Laptop HP
 - Software : Microsoft Edge, Windows11

d. Nama dan Peran Penguji:

- Nama : Adil
- Peran : *Software Tester*

e. Deskripsi:

Fitur Menampilkan Informasi Detail Barang pada sistem Data Center Inventory dirancang untuk memberikan informasi lengkap mengenai karakteristik suatu barang.

f. Langkah - Langkah:

1. Pengguna login ke dalam sistem menggunakan akun yang valid.
2. Dari menu utama, pengguna memilih Kelola Data Barang.
3. Sistem menampilkan daftar barang dalam tabel.
4. Pengguna memilih salah satu barang pada daftar dan menekan tombol Detail Barang.
5. Sistem menampilkan halaman detail barang yang memuat informasi terkait seperti:

- Nama Barang
- PO Number
- Kategori
- Harga Beli
- Lokasi Penyimpanan
- Jumlah Barang
- Status Barang
- Tanggal Pembelian

6. Pengguna memverifikasi bahwa informasi yang ditampilkan sesuai dengan data yang dimasukkan sebelumnya.

g. Alternative Flow:

3. Jika data barang untuk kategori yang dipilih tidak ditemukan, sistem menampilkan pesan: "Tidak ada data ditemukan untuk kategori ini."
4. Daftar barang akan kosong, dan pengguna dapat memilih kategori lain.

Fitur: Detail Barang				
Test ID	Data Barang yang di pilih	Ketersediaan Data	Hasil Yang Diharapkan	Status
TC-08-001	Nic card	Tersedia	Menampilkan detail barang seperti nama, kategori, asset number, po number dll	Sukses
TC-08-002	Server	Tersedia	Menampilkan deteail barang seperti asset number, jumlah stok, po number dll	Sukses
TC-08-003	Barang Tidak di Temukan	Tidak Tersedia	Pesan: ”Data barang tidak di temukan”	Sukses
TC-08-004	(Kosong)	Data Rusak	Pesan: “Terjadi kesalahan sistem. Silahkan coba lagi nanti”	Sukses

Tabel 21. Pengujian Fitur Detail Barang

9. Fitur *Logout*

- a. Fitur yang di uji : *Filter Logout*
- b. Tanggal Pengujian : 29-03-2026
- c. Lingkungan Pengujian:
 - Hardware : Laptop HP
 - Software : Microsoft Edge, Windows11
- d. Nama dan Peran Penguji:
 - Nama : Adil
 - Peran : *Software Tester*

e. Deskripsi:

Fitur *logout* memungkinkan pengguna keluar dari aplikasi dan mengakhiri sesi aktif. Setelah *logout*, pengguna akan diarahkan kembali ke halaman *login*, dan sesi autentikasi akan dinonaktifkan.

f. Langkah - Langkah:

1. Pengguna masuk ke aplikasi dan berada di halaman dashboard.
2. Pengguna klik profile.
3. Pengguna memilih opsi logout.
4. Pengguna diarahkan ke halaman login.

g. Alternative Flow:

Jika sistem gagal memproses *logout* (contoh: masalah koneksi atau error internal), pengguna akan mendapatkan pesan error dan tetap berada di halaman dashboard sampai *logout* berhasil.

Fitur: Logout				
Test ID	Inputan	Partisi	Hasil Yang Diharapkan	Status
TC-09-001	Klik tombol Logout saat login aktif	Valid	Sistem menghapus sesi, mengarahkan ke halaman login	Sukses
TC-09-002	Sesi berakhir karena timeout (idle)	Valid	Sistem logout otomatis dan mengarahkan ke halaman login	Sukses
TC-09-003	Klik tombol logout saat jaringan bermasalah	Valid	Gagal Logout	Sukses

Tabel 22. Pengujian Fitur Logout

BAB V KESIMPULAN

5.1 Kesimpulan

Berdasarkan hasil pengembangan Sistem Informasi Manajemen Inventaris pada Data Center, berikut adalah kesimpulan yang dapat ditarik:

1. Pencapaian Tujuan Proyek

Tujuan Tujuan fundamental project ini, yakni mendesain dan mengembangkan aplikasi informasi inventaris berbasis web yang menunjang dokumentasi barang, manajemen record, monitoring kondisi item, dan generasi laporan, telah berhasil dicapai. Aplikasi berhasil dikembangkan menggunakan ASP.NET Core sebagai framework backend dan SQL Server untuk pengelolaan basis data.

.

2. Efisiensi Proses Pencatatan

Sistem ini berhasil mengotomatisasi proses pencatatan barang masuk dan keluar, yang sebelumnya dilakukan secara manual. Setiap transaksi kini direkam secara langsung ke dalam database terpusat, sehingga mempercepat proses pencatatan dan mengurangi risiko kesalahan manusia.

3. Kemudahan Monitoring

Dengan adanya fitur Status Barang dan monitoring aktivitas pengguna, sistem memungkinkan admin untuk memantau dengan mudah stok barang, lokasi penyimpanan, serta riwayat aktivitas pengguna. Hal ini meningkatkan transparansi dan akuntabilitas pengelolaan data inventaris.

4. Pembuatan Laporan yang Otomatis dan Akurat

Sistem berhasil menyediakan fitur untuk menghasilkan laporan barang masuk, barang keluar, dan stok barang secara otomatis. Laporan dapat difilter berdasarkan tanggal, kategori, atau lokasi, serta diekspor dalam format PDF atau Excel, mempermudah pengguna untuk mendokumentasikan data.

5. Pengujian Sistem yang Komprehensif

Pengujian sistem menggunakan metode *blackbox* testing dan *User Acceptance Testing* (UAT) menunjukkan bahwa semua fitur utama berjalan dengan baik dan memenuhi kebutuhan pengguna. Feedback dari pengujian telah diimplementasikan dalam iterasi pengembangan, sehingga meningkatkan kepuasan pengguna terhadap sistem.

5.2 Saran

Selama pengembangan proyek ini, terdapat banyak pembelajaran baik dari aspek teknis maupun non-teknis. Berikut adalah pelajaran yang dapat diambil dari proyek ini:

1. Pentingnya Pemahaman Kebutuhan Pengguna
 - Memahami kebutuhan pengguna dan mengintegrasikannya dalam desain sistem sangat penting. Proses pengumpulan kebutuhan menjadi tahap yang krusial untuk memastikan bahwa fitur yang dikembangkan relevan dengan kebutuhan praktis pengguna.
2. Penerapan Metode Agile yang Efektif
 - Metode Agile membantu memastikan pengembangan yang fleksibel dan progresif. Sprint yang terencana memungkinkan identifikasi kendala atau perubahan kebutuhan lebih awal, sehingga kesalahan

dapat diminimalkan. Selain itu, Agile mendorong kolaborasi erat antara tim pengembang dengan pengguna.

3. Fokus pada Keamanan Data
 - Keamanan menjadi aspek yang penting dalam sistem ini, terutama pada autentikasi pengguna. Penggunaan teknologi seperti JWT (JSON Web Token) terbukti efektif untuk menjaga keamanan akses sistem.
4. Signifikansi Pengujian yang Terstruktur
 - Pengujian yang komprehensif memberikan dampak langsung terhadap kualitas sistem. Dengan menggunakan skenario yang realistis, termasuk pengetesan fitur logout, laporan, dan filter, pengujian membantu mengidentifikasi dan mengatasi masalah sebelum sistem diimplementasikan di produksi.
5. Kolaborasi dalam Tim
 - Komunikasi yang baik antara anggota tim pengembang menjadi faktor kunci keberhasilan proyek. Diskusi rutin seperti meeting harian dan evaluasi hasil sprint mempercepat penyelesaian masalah dan mendukung implementasi solusi yang lebih baik.
6. Manajemen Masalah Teknis
 - Beberapa kendala teknis, seperti kegagalan koneksi database atau konfigurasi API, memerlukan kemampuan troubleshooting yang baik. Log dan debugging terbukti sangat membantu dalam mendeteksi dan memperbaiki masalah teknis selama proses pengembangan.
7. Kustomisasi untuk Skalabilitas
 - Sistem ini dirancang agar mudah diperluas. Hal ini menjadi pelajaran penting untuk memastikan desain awal sistem tidak hanya sesuai kebutuhan saat ini, tetapi juga mampu menangani peningkatan data atau kebutuhan tambahan di masa depan.

Proyek ini telah menunjukkan keberhasilan dalam membangun dan mengimplementasikan sistem inventaris yang terstruktur, efisien, dan sesuai

kebutuhan. Dengan pendekatan Agile yang diterapkan, sistem berhasil dikembangkan secara bertahap dan memastikan setiap iterasi memberikan nilai tambah. Pengalaman yang didapat dari pengembangan proyek ini menjadi pondasi penting untuk pengembangan proyek-proyek serupa di masa depan.

DAFTAR PUSTAKA

- Athallah, D., & Hidayatullah, D. (2022). *Penerapan ASP.NET dalam membangun sistem inventory dengan menggunakan metode prototype*. Jurnal Teknik Informatika dan Sistem Informasi, 9(1), 468–485.
- Mardiana, M., Mulyani, Y., Despa, D., Sulistiono, W. E., & Yulianto, R. (2024). *Pengembangan REST API dengan Express.js, ASP.NET Core dan Django: Studi Kasus Perbandingan Performa dengan Beban yang Beragam*. Prosiding Seminar Nasional Ilmu Teknik dan Aplikasi Industri (SINTA), Universitas Lampung.
- Deni Yuniarti, Galih Widyatmojo. (2025). *Sistem Informasi Persediaan Barang Berbasis Website (Studi Kasus: Instalasi Gizi RS Harapan Sehat Jati Barang)*. , Universitas Teknologi Digital.
- Pranoto, A. O., & Sedyono, E. (2021). *Perancangan Sistem Informasi Inventaris Barang Berbasis Web*. Jurnal Teknik Informatika dan Sistem Informasi, 7(2), 357–362.
- Rohman, A., & Bhakti, H. D. (2022). *Perancangan Sistem Informasi Persediaan Barang Berbasis Web*. Syntax Literate: Jurnal Ilmiah Indonesia, 7(9), 15305–15310.
- Pricillia, T., & Zulfachmi. (2021). Survey Paper: Perbandingan Metode Pengembangan Perangkat Lunak (Waterfall, Prototype, RAD). Bangkit Indonesia.
- Hardiansyah, F., Rizal, A., & Purnamasari, I. (2023). Implementasi metode agile SCRUM dalam pengembangan aplikasi pembelajaran olahraga. *JATI (Jurnal Mahasiswa Teknik Informatika)*, 7(2).
- Erwanto, M. S. I., & Umniati, N. (2022). Pengembangan aplikasi sales track pada PT. Hexaon Business Mitrasindo menggunakan agile development methods. *Jurnal Ilmiah Teknologi dan Rekayasa*, 27(1), 12–28.
- Anjasmara, V. M., & Sumitro, A. H. (2023). Pengembangan sistem informasi masjid Darul Arham menggunakan metode V-Model dan UAT (User Acceptance Testing). *Information System for Educators and Professionals*, 7(1), 47–58
- Susanto, J., Biqirrosyad, B., Junaidi, M. M., Sudrajat, Y., & Desyani, T. (2021)

Pengujian black box pada aplikasi desktop penjualan elektronik menggunakan metode equivalence partitioning. *Jurnal Teknologi Sistem Informasi Dan Aplikasi*, 4(1), 52.

Desyani, T., Nirmala, E., Lisdiarto, A., Ridwan, H., Pambudi, R. W. K., & Septiani, V. (2022). Pengujian black box pada aplikasi penjualan berbasis web menggunakan metode equivalence partitioning. *Jurnal Informatika Universitas Pamulang*, 7(1), 79–82.

Oktaviani, N., Widiarta, I. M., & Nurlaily, N. (2019). *Sistem Informasi Inventaris Barang Berbasis Web pada SMP Negeri 1 Buer*. JINTEKS, 1(2), 160–165.

DAFTAR LAMPIRAN

Lampiran A. Dokumen Proses Pengumpulan Requirement

<https://drive.google.com/drive/folders/1Lu1-hzcu-ZPbbunevVRGviaKVQ-Fk19f>

Lampiran B. Link Produk

https://github.com/adilgunawan/bth_dc_inventory.git

Lampiran C. Dokumen Pengujian UAT

<https://drive.google.com/drive/folders/1GlC0AQ1oh4mD2trcrNBbBU7isESPVUgE>