

**RANCANG BANGUN SISTEM MANAJEMEN
ASET BERBASIS *QR CODE* UNTUK
OPTIMALISASI PENGELOLAAN LOGISTIK DI
PT DALAZ TEKNIK UTAMA**

PROYEK AKHIR

Disusun Oleh:

Yunita Caroline Sianturi

3312311104

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



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

KATA PENGANTAR

Segala puji dan syukur penulis naikkan kepada Tuhan Yesus Kristus atas penyertaan dan kasih-Nya yang setia, sehingga penulis dapat menyelesaikan Laporan Tugas Akhir berjudul "RANCANG BANGUN SISTEM MANAJEMEN ASET BERBASIS *QR CODE* UNTUK OPTIMALISASI PENGELOLAAN LOGISTIK DI PT DALAZ TEKNIK UTAMA" dengan baik.

Laporan ini disusun sebagai salah satu syarat kelulusan akademik di Program Studi Teknik Informatika, Politeknik Negeri Batam, sekaligus sebagai dokumentasi atas seluruh proses perancangan dan pembangunan sistem yang telah penulis kerjakan selama pelaksanaan magang di PT Dalaz Teknik Utama. Melalui laporan ini, penulis berharap dapat menyampaikan hasil kerja secara sistematis dan memberikan gambaran yang jelas mengenai solusi yang dirancang untuk mendukung pengelolaan logistik perusahaan.

Terselesaikannya laporan ini tidak lepas dari dukungan dan kontribusi berbagai pihak. Untuk itu, penulis menyampaikan terima kasih yang sebesar-besarnya kepada :

1. Ir. Bambang Hendrawan, S.T., MSM., CIPMP., CISCIP., selaku Rektor Politeknik Negeri Batam, yang telah memimpin institusi dengan penuh dedikasi serta memastikan tersedianya kebijakan, sarana, dan prasarana yang mendukung proses pendidikan dan penyelesaian Tugas Akhir ini.
2. Ahmad Hamim Thohari, S.S.T., M.T., selaku Kepala Jurusan Informatika, yang telah menciptakan iklim akademik yang kondusif sehingga penulis dapat menempuh perkuliahan dan merampungkan laporan ini dengan baik.
3. Yeni Rokhayati, S.Si., M.Sc., selaku Ketua Program Studi Teknik Informatika, yang senantiasa memberikan arahan dan dukungan akademik yang membantu kelancaran proses perkuliahan hingga penyusunan laporan ini.
4. Alena Uperiati, S.T., M.Cs., selaku Dosen Pembimbing, yang telah meluangkan waktu untuk memberikan bimbingan, masukan, serta arahan yang sangat berarti dalam proses penulisan dan pengembangan sistem.

5. M. Dzikri Fadilah, selaku IT *Supervisor* sekaligus Pembimbing Perusahaan di PT Dalaz Teknik Utama, yang telah berbagi pengalaman dan pengetahuan lapangan, serta memberikan kepercayaan kepada penulis untuk berkontribusi langsung dalam lingkungan kerja profesional.
6. PT Dalaz Teknik Utama, atas kesempatan, kepercayaan, dan fasilitas yang diberikan sehingga penulis dapat melaksanakan magang dan mengembangkan sistem dengan baik.
7. James Sianturi, Eva Manurung, Steven Sianturi, dan Samuel Sianturi, keluarga tercinta yang menjadi sumber semangat terbesar penulis. Terima kasih atas doa yang tidak pernah putus, dukungan moral dan materi, serta cinta yang selalu hadir di setiap langkah perjalanan ini.
8. Efrylia Tampubolon, Novriyanti br Hombing, Lihua Situmorang, Novia Sitohang, Jeliyanti Tambunan, Cheline Sidabutar, Ribka Tambunan, dan Doni Sianipar, serta seluruh teman-teman Inti El-Shaddai 2025, KIAN, dan Grace *Cell* sahabat-sahabat terbaik yang selalu hadir dengan dukungan dan semangat, terutama di saat penulis paling membutuhkannya.
9. Semua pihak yang tidak dapat penulis sebutkan satu per satu, yang telah membantu dalam penyelesaian laporan ini.

Penulis berharap laporan ini dapat bermanfaat bagi pembaca, menjadi referensi bagi rekan-rekan yang akan mengerjakan proyek serupa, serta memberikan kontribusi nyata bagi perkembangan ilmu di bidang teknologi informasi dan manajemen aset. Penulis menyadari bahwa laporan ini masih memiliki kekurangan, sehingga kritik dan saran yang membangun akan senantiasa penulis terima dengan terbuka demi perbaikan ke depannya.

Batam, 17 Juli 2026



Yunita Caroline Sianturi

DAFTAR ISI

KATA PENGANTAR	i
DAFTAR ISI.....	iii
DAFTAR TABEL.....	v
DAFTAR GAMBAR	vii
DAFTAR LAMPIRAN.....	ix
ABSTRAK	x
BAB I PENDAHULUAN	1
1.1. Latar Belakang	1
1.2. Rumusan Masalah	3
1.3. Tujuan	3
1.4. Batasan Masalah.....	3
1.5. Manfaat	4
1.6. Kolaborasi dan Sinergi dalam Pelaksanaan Proyek	5
BAB II TINJAUAN PUSTAKA.....	7
2.1. Tinjauan Solusi dan Sistem Terkait	7
2.2. Landasan Teori.....	9
2.2.1 Sistem Informasi	10
2.2.2 Manajemen Aset.....	10
2.2.3 QR Code (<i>Quick Response Code</i>)	10
2.2.4 Website.....	10
2.2.5 PHP (<i>Hypertext Preprocessor</i>)	11
2.2.6 Pola Arsitektur MVC (<i>Model-View-Controller</i>).....	11
2.2.7 MySQL.....	11
2.2.8 PT Dalaz Teknik Utama.....	11
2.3. Metode Pengembangan Produk.....	12
2.3.1. <i>Requirements Analysis</i> (Analisis Kebutuhan).....	13
2.3.2. <i>Design</i> (Perancangan)	14
2.3.3. <i>Implementation</i> (Implementasi)	14
2.3.4. <i>Testing</i> (Pengujian)	15
2.3.5. <i>Maintenance</i> (Pemeliharaan)	16
BAB III ANALISIS DAN PERANCANGAN	17

3.1. Analisis Kebutuhan	17
3.1.1 Proses Bisnis Berjalan.....	17
3.1.2 Gambaran Umum Sistem	18
3.2. Perancangan	20
3.2.1 Kebutuhan Fungsional	20
3.2.2 Kebutuhan Non-Fungsional	21
3.2.3 Diagram <i>Use case</i>	22
3.2.4 Skenario <i>Use case</i>	23
3.2.5 <i>Activity</i> Diagram.....	44
3.2.6 ER Diagram.....	59
3.2.7 Perancangan Antarmuka Pengguna	61
BAB IV IMPLEMENTASI DAN PENGUJIAN.....	67
4.1. Hasil Implementasi.....	67
4.1.1 Implementasi Basis Data.....	68
4.1.2 Implementasi Sistem	69
4.1.3 Implementasi Antarmuka.....	72
4.2. Pengujian <i>User Acceptance Testing</i> (UAT).....	94
4.2.1 Pengujian Perangkat Lunak (<i>Functional Testing</i>).....	94
4.2.2 Pengujian Perangkat Lunak (<i>Non-Functional Testing</i>).....	135
BAB V KESIMPULAN.....	143
5.1 Kesimpulan	143
5.2 Saran.....	144
DAFTAR PUSTAKA	145
DAFTAR LAMPIRAN.....	148
Lampiran 1 Dokumen Proses Pengumpulan <i>Requirement</i>	148
Lampiran 2 Dokumentasi Perancangan Sistem.....	148
Lampiran 3 <i>Link</i> Produk.....	148
Lampiran 4 Dokumen Pengujian UAT	148
Lampiran 5 Demo Aplikasi.....	148

DAFTAR TABEL

Tabel 2. 1 Tinjauan Solusi dan Sistem Terkait	8
Tabel 3. 1 Kebutuhan Fungsional	21
Tabel 3. 2 Kebutuhan Non-Fungsional	21
Tabel 3. 3 Skenario <i>Use case Login</i>	24
Tabel 3. 4 Skenario <i>Use case</i> Melihat <i>Dashboard</i> Statistik Aset	25
Tabel 3. 5 Skenario <i>Use case</i> Mengelola Data Aset Statis	26
Tabel 3. 6 Skenario <i>Use case</i> Mengelola Data <i>Equipment</i>	28
Tabel 3. 7 Skenario <i>Use case</i> Mengelola Data <i>Inventory</i>	29
Tabel 3. 8 Skenario <i>Use case</i> Mengelola Data <i>Location</i>	30
Tabel 3. 9 Skenario <i>Use case</i> Mengelola Data <i>Category</i>	31
Tabel 3. 10 Skenario <i>Use case</i> Mengelola Data <i>User</i>	32
Tabel 3. 11 Skenario <i>Use case Generate QR Code</i>	33
Tabel 3. 12 Skenario <i>Use case</i> Meminjam <i>Equipment</i> (dengan <i>QR Scanner</i>)	34
Tabel 3. 13 Skenario <i>Use case</i> Mengembalikan <i>Equipment</i> (dengan <i>QR Scanner</i>)	35
Tabel 3. 14 Skenario <i>Use case</i> Mengajukan <i>Complaint</i>	37
Tabel 3. 15 Skenario <i>Use case</i> Melihat <i>History</i>	39
Tabel 3. 16 Skenario <i>Use case</i> Mencatat Transaksi <i>Inventory In/Out</i>	40
Tabel 3. 17 Skenario <i>Use case</i> Memberikan Respon <i>Complaint</i>	41
Tabel 3. 18 Skenario <i>Use case</i> Mengunduh Laporan Aset	42
Tabel 3. 19 Skenario <i>Use case Logout</i>	43
Tabel 4. 1 <i>Blackbox Testing Login</i>	95
Tabel 4. 2 <i>Blackbox Testing Dashboard</i>	96
Tabel 4. 3 <i>Blackbox Testing</i> Mengelola Data Aset Statis	98
Tabel 4. 4 <i>Blackbox Testing</i> Mengelola Data <i>Equipment</i>	104
Tabel 4. 5 <i>Blackbox Testing</i> Mengelola Data <i>Inventory</i>	107
Tabel 4. 6 <i>Blackbox Testing</i> Mengelola Data <i>Location</i>	110
Tabel 4. 7 <i>Blackbox Testing</i> Mengelola Data <i>Category</i>	112
Tabel 4. 8 <i>Blackbox Testing</i> Mengelola Data <i>User</i>	114

Tabel 4. 9 <i>Blackbox Testing</i> Mengelola dan Mengunduh <i>QR Code</i>	117
Tabel 4. 10 <i>Blackbox Testing</i> Peminjaman <i>Equipment</i>	118
Tabel 4. 11 <i>Blackbox Testing</i> Pengembalian <i>Equipment</i>	121
Tabel 4. 12 <i>Blackbox Testing</i> Mengajukan <i>Complaint</i>	123
Tabel 4. 13 <i>Blackbox Testing</i> Melihat Riwayat Peminjaman	125
Tabel 4. 14 <i>Blackbox Testing</i> Mencatat Transaksi <i>Inventory In/Out</i>	127
Tabel 4. 15 <i>Blackbox Testing</i> Memberikan Respon Komplain	130
Tabel 4. 16 <i>Blackbox Testing</i> Mengunduh Laporan PDF	132
Tabel 4. 17 <i>Blackbox Testing</i> Logout.....	134
Tabel 4. 18 <i>Security Testing</i>	136
Tabel 4. 19 <i>Performance Testing</i>	139
Tabel 4. 20 <i>Usability Testing</i>	141
Tabel 4. 21 <i>Portability Testing</i>	142

DAFTAR GAMBAR

Gambar 2. 1 Metode <i>Waterfall</i>	13
Gambar 3. 1 Gambaran Umum Sistem	19
Gambar 3. 2 <i>Usecase</i> Diagram.....	22
Gambar 3. 3 <i>Activity</i> Diagram <i>Login</i>	45
Gambar 3. 4 <i>Activity</i> Diagram Melihat <i>Dashboard</i>	46
Gambar 3. 5 <i>Activity</i> Diagram Mengelola Master Data Aset Statis.....	47
Gambar 3. 6 <i>Activity</i> Diagram Mengelola Data <i>Equipment</i>	48
Gambar 3. 7 <i>Activity</i> Diagram <i>Inventory</i>	49
Gambar 3. 8 <i>Activity</i> Diagram Mengelola <i>Location</i>	50
Gambar 3. 9 <i>Activity</i> Diagram Mengelola <i>Category</i>	50
Gambar 3. 10 <i>Activity</i> Diagram <i>User Data</i>	51
Gambar 3. 11 <i>Activity</i> Diagram <i>Generate QR Code</i>	52
Gambar 3. 12 <i>Activity</i> Diagram <i>Borrowing Equipment</i>	53
Gambar 3. 13 <i>Activity</i> Diagram <i>Return Equipment</i>	54
Gambar 3. 14 <i>Activity</i> Diagram <i>User Complaint</i>	55
Gambar 3. 15 <i>Activity</i> Diagram <i>History</i>	56
Gambar 3. 16 <i>Activity</i> Diagram <i>Inventory In/Out</i>	56
Gambar 3. 17 <i>Activity</i> Diagram Admin Merespon <i>Complaint</i>	57
Gambar 3. 18 <i>Activity</i> Diagram Mengunduh Laporan PDF.....	58
Gambar 3. 19 <i>Activity</i> Diagram <i>Logout</i>	58
Gambar 3. 20 ER Diagram.....	59
Gambar 3. 21 Tampilan <i>Login</i>	62
Gambar 3. 22 Tampilan <i>Dashboard</i> Admin	62
Gambar 3. 23 Tampilan Data <i>Asset company</i>	63
Gambar 3. 24 Tampilan <i>Inventory In/Out</i>	63
Gambar 3. 25 Tampilan <i>Location</i>	64
Gambar 3. 26 Tampilan <i>Equipment Borrowing</i>	65
Gambar 3. 27 Tampilan <i>Equipment Returning</i>	65
Gambar 3. 28 Tampilan <i>History</i> Admin	66

Gambar 4. 1 Implementasi Basis Data	68
Gambar 4. 2 Struktur Direktori MVC pada Sistem.....	70
Gambar 4. 3 Implementasi Koneksi Basis Data dan Keamanan <i>Backend</i>	71
Gambar 4. 4 Implementasi Integrasi <i>Library</i> TCPDF pada <i>Controller</i>	71
Gambar 4. 5 Implementasi UI <i>Login</i>	72
Gambar 4. 6 Implementasi UI <i>Dashboard</i> Admin.....	73
Gambar 4. 7 Implementasi UI <i>Office Equipment</i> Data	75
Gambar 4. 8 Implementasi UI <i>Equipment</i> Data	77
Gambar 4. 9 Implementasi UI <i>Inventory</i>	79
Gambar 4. 10 Implementasi UI <i>Land & Building</i> Data	80
Gambar 4. 11 Implementasi UI <i>Vehicle</i> Data	81
Gambar 4. 12 Implementasi UI <i>Intangible Assets</i> Data.....	82
Gambar 4. 13 Implementasi UI <i>Location</i>	82
Gambar 4. 14 Implementasi UI <i>Category</i>	83
Gambar 4. 15 Implementasi UI <i>User</i> Data.....	85
Gambar 4. 16 Implementasi UI <i>History</i> Admin.....	86
Gambar 4. 17 Implementasi UI <i>Manage Complaint</i>	87
Gambar 4. 18 Implementasi UI <i>Setting</i>	88
Gambar 4. 19 Implementasi UI <i>Dashboard User</i>	89
Gambar 4. 20 Implementasi UI <i>Borrow</i>	91
Gambar 4. 21 Implementasi UI <i>Return</i>	93
Gambar 4. 22 Implementasi UI <i>History User</i>	93

DAFTAR LAMPIRAN

Lampiran 1 Dokumen Proses Pengumpulan <i>Requirement</i>	148
Lampiran 2 Dokumentasi Perancangan Sistem.....	148
Lampiran 3 <i>Link</i> Produk.....	148
Lampiran 4 Dokumen Pengujian UAT	148
Lampiran 5 Demo Aplikasi.....	148

ABSTRAK

Pengelolaan aset di PT Dalaz Teknik Utama saat ini masih dilakukan secara manual menggunakan *spreadsheet* dan catatan kertas, sehingga menimbulkan permasalahan seperti tingginya potensi kesalahan pencatatan, keterlambatan dalam pencarian data, serta sulitnya memantau status ketersediaan peralatan inspeksi secara akurat. Proyek Akhir ini bertujuan untuk merancang dan membangun Sistem Manajemen Aset berbasis web yang terintegrasi dengan teknologi *QR Code* guna mengoptimalkan pengelolaan logistik di PT Dalaz Teknik Utama. Sistem dikembangkan menggunakan bahasa pemrograman PHP dengan pola arsitektur *Model-View-Controller* (MVC) dan basis data MySQL, serta menerapkan metodologi pengembangan *Waterfall*. Fitur yang dihasilkan mencakup pengelolaan enam kategori aset (*Office Equipment, Equipment, Inventory, Vehicle, Land & Building, dan Intangible Assets*), proses peminjaman dan pengembalian khusus kategori *Equipment* berbasis pemindaian *QR Code*, pencatatan transaksi *Inventory In/Out*, manajemen *complaint*, serta penyediaan laporan dan *dashboard* statistik. Hasil pengujian fungsional menggunakan metode *Black Box Testing* menunjukkan bahwa seluruh fitur sistem berjalan 100% sesuai dengan spesifikasi kebutuhan. Hasil pengujian non-fungsional membuktikan bahwa sistem memiliki keamanan autentikasi yang memadai, antarmuka yang responsif dan intuitif, serta performa tinggi dengan rata-rata waktu respons di bawah 250 ms pada simulasi 20 pengguna secara bersamaan. Dengan diimplementasikannya sistem ini, PT Dalaz Teknik Utama dapat meminimalkan kesalahan pencatatan manual serta meningkatkan efisiensi dan akurasi pengelolaan logistik perusahaan.

Kata Kunci: Manajemen Aset, *QR Code*, Sistem Berbasis Web, PHP MVC, Metode *Waterfall*

BAB I

PENDAHULUAN

1.1. Latar Belakang

Dalam era transformasi digital, banyak perusahaan yang tergerak untuk memadukan teknologi informasi dalam aktivitas bisnis mereka. Perusahaan-perusahaan sedang mendigitalkan proses-proses manual, termasuk distribusi barang, pengendalian persediaan, dan pemantauan logistik (Oktawijaya dan Agustin, 2024). Selain meningkatkan efisiensi operasional, proses digitalisasi ini membuka peluang baru untuk inovasi, yang membantu perusahaan berkembang di tengah persaingan yang semakin ketat di industri logistik Indonesia, yang diperkirakan akan mencapai USD 4,25 miliar pada tahun 2028 (Ken Research, 2025).

Salah satu fokus utama dalam peningkatan operasional tersebut terletak pada manajemen aset. Manajemen aset merupakan bagian penting dalam operasional sebuah perusahaan yang bertujuan untuk mengelola dan mengoptimalkan aset sehingga mengurangi resiko kerusakan atau kehilangan aset (Kusumojati dan Mediawati, 2024).

Namun, banyak perusahaan masih mengandalkan proses manual ketika mengelola aset, yang dapat menyebabkan kelalaian akibat kelalaian manusia (Rijadi dan Suakanto, 2024). Penggunaan *spreadsheet* dan catatan kertas untuk mengelola aset dapat menyebabkan masalah operasional yang serius. Masalah ini meliputi kesulitan dalam melacak aset secara *real-time*, proses pelaporan yang lambat, penginputan data berulang, dan risiko kehilangan informasi penting, yang dapat memengaruhi operasional perusahaan dalam pengambilan keputusan (Noguerra, 2023).

Kondisi ini sejalan dengan tren transformasi digital saat ini di industri logistik, di mana banyak perusahaan secara aktif mencari solusi teknologi untuk mengatasi kesulitan dalam mengelola aset dan persediaan (Oyeniya *et al.*, 2024). Teknologi *QR Code* telah terbukti efektif dalam meningkatkan efisiensi dalam

pengelolaan aset dengan memungkinkan identifikasi dan pelacakan aset yang cepat, akurat, dan minim kesalahan (Grandynarta *et al.*, 2024).

Masalah serupa juga muncul dalam pengelolaan aset di PT Dalaz Teknik Utama, sebuah perusahaan yang menyediakan layanan inspeksi peralatan berat. Perusahaan memiliki enam kategori aset yang perlu dikelola: *Office Equipment* (aset tetap seperti meja dan kursi), *Land & Building*, *Vehicle*, dan *Intangible Assets* yang dicatat menggunakan *spreadsheet*, *Equipment* (peralatan inspeksi yang dapat dipinjam) dan *Inventory* (barang habis pakai) yang masih dicatat secara manual menggunakan kertas. Teknik pencatatan manual yang ada menyebabkan sejumlah masalah operasional, seperti kesalahan dalam data inventaris, keterlambatan dalam proses pencarian, dan kesulitan melacak lokasi aset.

Untuk meningkatkan efektivitas proses logistik di PT Dalaz Teknik Utama, penelitian ini bertujuan untuk merancang dan mengembangkan sistem manajemen aset gudang berbasis kode QR. Hal ini disebabkan oleh urgensi masalah dan potensi keunggulan teknologi kode QR, yang telah terbukti dalam berbagai implementasi.

Untuk meningkatkan efektivitas proses logistik di PT Dalaz Teknik Utama, diperlukan sistem manajemen aset berbasis *QR Code*. Implementasi sistem manajemen aset berbasis *QR Code* memberikan berbagai keuntungan signifikan, termasuk peningkatan akurasi pencatatan data hingga 95%, pengurangan waktu pencarian barang hingga 60%, serta penyediaan akses data *real-time* yang meningkatkan transparansi operasional (Fajar *et al.*, 2025). Sistem berbasis web memungkinkan akses data secara terpusat, mengurangi risiko kehilangan dokumen, serta memudahkan proses pelaporan dan audit (Kusumojati dan Mediawati, 2024).

Dengan demikian, sistem yang dirancang ini diharapkan dapat mempercepat proses peminjaman dan pengembalian *Equipment*, mengotomatisasi proses pencatatan aset, *monitoring* status *Equipment* dan stok *Inventory*, sehingga meningkatkan efisiensi operasional dan produktivitas PT Dalaz Teknik Utama dalam mendukung layanan *Quality Inspection* untuk klien.

1.2. Rumusan Masalah

Berdasarkan latar belakang yang telah diuraikan, rumusan masalah dalam penelitian ini adalah :

1. Bagaimana merancang dan membangun sistem informasi manajemen aset berbasis web untuk mengelola enam kategori aset (*Office Equipment, Equipment, Inventory, Vehicle, Land & Building, dan Intangible Assets*) di PT Dalaz Teknik Utama?
2. Bagaimana sistem dapat menyajikan status ketersediaan aset yang langsung diperbarui pada *dashboard* setelah transaksi disimpan?

1.3. Tujuan

Mengacu pada rumusan masalah di atas, tujuan dari Proyek Akhir ini adalah sebagai berikut :

1. Membangun sistem informasi berbasis web untuk mengelola data enam kategori aset di PT Dalaz Teknik Utama.
2. Menyediakan fitur laporan dan *dashboard* yang menampilkan status ketersediaan aset secara langsung berdasarkan data transaksi terakhir yang tersimpan dalam basis data.

1.4. Batasan Masalah

Agar proyek ini lebih terfokus dan dapat diselesaikan sesuai waktu yang ditentukan, maka ruang lingkup permasalahan dibatasi pada hal-hal berikut:

1. Sistem mengelola pendataan enam kategori aset: *Office Equipment, Equipment, Inventory, Vehicle, Land & Building, dan Intangible Assets*.
2. Fitur peminjaman (*borrow*) dan pengembalian (*return*) menggunakan *QR Code Scanner* hanya difokuskan untuk aset kategori *Equipment*.
3. Lima kategori aset lainnya hanya mencakup pendataan inventaris dan riwayat keluar-masuk barang, tanpa adanya alur peminjaman.

4. Sistem berbasis web ini dioperasikan menggunakan pemindai *QR Code*, baik melalui kamera perangkat (*smartphone/laptop*) maupun perangkat keras tambahan (*USB/Wireless Scanner*).
5. Hak akses dibedakan menjadi dua tingkat:
 - Administrator: Mengelola seluruh pendataan aset, merespon komplain kerusakan, dan mencetak laporan.
 - *User/Inspector*: Hanya dapat memindai peminjaman, memindai pengembalian, dan melaporkan kondisi/kerusakan khusus untuk kategori *Equipment*.
6. Parameter *real-time* dibatasi pada pembaruan status di dalam basis data (*database*), di mana ketersediaan barang di *dashboard* langsung berubah setelah pengguna menyimpan transaksi.
7. Pengembangan sistem menggunakan metode *Waterfall*. Tahap pemeliharaan (*maintenance*) dibatasi hanya sampai masa magang penulis di PT Dalaz Teknik Utama selesai.
8. Pengujian sistem menggunakan metode *Black Box Testing* untuk fungsi aplikasi dan *User Acceptance Testing* (UAT) yang dibuktikan dengan dokumen *User Acceptance Approval* dari perusahaan.
9. Sistem tidak mencakup proses pengadaan barang (*purchasing*), penyusutan nilai aset, maupun integrasi sistem akuntansi.
10. Sistem diasumsikan berjalan berdampingan dengan Standar Operasional Prosedur (SOP) gudang, di mana setiap transaksi pemindaian *QR Code* (peminjaman dan pengembalian) dilakukan di area penyimpanan dan diawasi langsung secara fisik oleh Administrator, sehingga sistem tidak memerlukan fitur persetujuan (*approval*) digital bertingkat.

1.5. Manfaat

Hasil dari Proyek Akhir ini diharapkan dapat memberikan beberapa manfaat sebagai berikut :

1. Memberikan referensi praktis terkait penerapan *QR Code* dalam pengembangan sistem informasi manajemen aset.

2. Mempermudah dan mempercepat proses pendataan, peminjaman, serta pengembalian peralatan inspeksi secara digital.
3. Mengurangi risiko *human error* berkat pemindaian kode aset secara otomatis menggunakan *QR Code*.
4. Memudahkan pihak manajemen memantau stok dan ketersediaan aset perusahaan.
5. Menjadi wadah bagi penulis untuk menerapkan ilmu pengembangan perangkat lunak (*software development*) pada kasus industri yang nyata.

Selain manfaat-manfaat tersebut, proyek ini juga memberikan kontribusi terhadap pencapaian Tujuan Pembangunan Berkelanjutan (*Sustainable Development Goals/SDGs*), khususnya SDG 9: *Industry, Innovation and Infrastructure*. Dengan merancang dan membangun sistem manajemen aset berbasis *QR Code*, proyek ini mendukung pembangunan infrastruktur informasi yang andal dan mendorong inovasi dalam pengelolaan logistik industri. Digitalisasi proses pencatatan dan pemantauan aset yang sebelumnya dilakukan secara manual meningkatkan efisiensi operasional, mengurangi pemborosan waktu dan sumber daya, serta mendukung industrialisasi yang lebih berkelanjutan di PT Dalaz Teknik Utama sebagai perusahaan penyedia layanan inspeksi dan sertifikasi di sektor industri.

1.6. Kolaborasi dan Sinergi dalam Pelaksanaan Proyek

Proyek Akhir ini dilaksanakan secara individu selama kegiatan magang di PT Dalaz Teknik Utama. Meskipun dikerjakan secara mandiri, pelaksanaan proyek tetap melibatkan kolaborasi aktif dengan berbagai pihak guna memastikan kualitas dan relevansi sistem yang dihasilkan. Bentuk kolaborasi tersebut antara lain:

1. Diskusi rutin dengan pembimbing lapangan, Bapak M. Dzikri Fadilah selaku IT *Supervisor* di PT Dalaz Teknik Utama, dilakukan secara berkala untuk memantau progres, mendapatkan arahan teknis, serta memastikan sistem yang dibangun sesuai dengan kebutuhan dan standar operasional perusahaan.

2. Koordinasi dengan rekan kerja di divisi terkait dilakukan dalam hal pengumpulan data aset, verifikasi alur kerja peminjaman dan pengembalian peralatan, serta pengujian sistem secara langsung di lingkungan kerja.
3. Konsultasi dengan dosen pembimbing, Ibu Alena Uperiati, S.T., M.Cs., dilakukan secara rutin untuk memastikan kesesuaian metodologi, kelengkapan dokumentasi, dan kualitas sistem sesuai standar akademik Politeknik Negeri Batam.
4. Pengumpulan umpan balik dari calon pengguna dilakukan melalui sesi *User Acceptance Testing* (UAT) bersama staf terkait di PT Dalaz Teknik Utama, guna memastikan sistem mudah digunakan dan memenuhi kebutuhan operasional secara nyata.

Bentuk kolaborasi ini menunjukkan bahwa meskipun proyek dilaksanakan secara individu, terdapat sinergi yang kuat dengan berbagai pemangku kepentingan. Keterlibatan aktif dari pembimbing lapangan, rekan kerja, dosen pembimbing, dan calon pengguna menjadi bagian penting dalam menjamin keberhasilan dan kualitas proyek secara keseluruhan.

BAB II

TINJAUAN PUSTAKA

2.1. Tinjauan Solusi dan Sistem Terkait

Bagian ini memberikan gambaran tentang sistem dan penelitian yang ada terkait dengan pengembangan sistem informasi manajemen aset dan inventaris. Tujuannya adalah untuk memahami posisi sistem yang dibangun dibandingkan dengan solusi lain, serta untuk menunjukkan pendekatan aplikatif apa yang ditawarkan untuk menyelesaikan masalah di PT Dalaz Teknik Utama.

Sistem "*Optimization of Inventory Management with QR Code Integration*" (Fajar *et al.*, 2025) memperkenalkan manajemen persediaan berbasis *QR Code* untuk kantor pajak. Sistem ini menggunakan *framework* Laravel dan MySQL, serta mengintegrasikan algoritma pencarian untuk mempercepat pengambilan data. Sistem ini terbukti meningkatkan akurasi pencatatan hingga 95% dan mengurangi waktu pencarian barang hingga 60%. Namun, solusi ini terbatas pada pendataan barang inventaris dalam satu kategori saja dan tidak mencakup pengelolaan aset multi-kategori yang memiliki karakteristik berbeda.

Sistem "*Implementation Web-Based QR-Code Information System in Warehouse Inventory Management*" (Anggara *et al.*, 2024) mengembangkan sistem informasi berbasis web untuk manajemen persediaan gudang. Dibangun menggunakan PHP dan MySQL, sistem ini memanfaatkan *QR Code* untuk mengurangi kesalahan pencatatan dan menyediakan akses data secara *real-time* yang meningkatkan transparansi operasional. Keterbatasan utama dari sistem ini adalah tidak tersedianya fitur peminjaman (*borrowing*) peralatan, karena hanya difokuskan pada stok masuk dan keluar.

Sistem "*Implementasi Warehouse Management System*" (Rahim *et al.*, 2025) mengembangkan sistem manajemen gudang berbasis web untuk meningkatkan efisiensi operasional perusahaan. Implementasi sistem ini mampu meningkatkan transparansi data, kecepatan proses, dan utilisasi ruang gudang secara optimal menggunakan PHP dan MySQL. Keterbatasannya terletak pada

identifikasi barang yang masih konvensional, di mana sistem ini belum mengimplementasikan teknologi *QR Code* untuk melacak aset.

Sistem "*Optimizing Inventory Receipt Processes Through a Mobile Application Utilizing QR Code*" (Saputra *et al.*, 2024) dirancang untuk meningkatkan proses penerimaan barang dengan memanfaatkan pemindaian *QR Code*. Sistem ini efektif meminimalkan kesalahan input data manual dan mempercepat akses informasi. Namun, solusi ini dikembangkan sebagai aplikasi *mobile (smartphone)*, sedangkan operasional di PT Dalaz Teknik Utama membutuhkan platform berbasis web yang terpusat dan dapat diakses fleksibel melalui PC atau laptop di area kantor.

Berdasarkan tinjauan terhadap solusi-solusi di atas, dapat disimpulkan bahwa mayoritas sistem masih berfokus pada manajemen persediaan gudang biasa, belum ada yang mengelola aset multi-kategori secara bersamaan, dan ada yang belum memanfaatkan *QR Code*. Sistem Manajemen Aset yang dikembangkan dalam proyek akhir ini hadir untuk mengisi celah tersebut dengan mengelola enam kategori aset sekaligus (*Office Equipment, Equipment, Inventory, Vehicle, Land & Building, dan Intangible Assets*), serta menyediakan fitur transaksi peminjaman khusus untuk *Equipment* menggunakan pemindai *QR Code* fisik.

Berikut adalah ringkasan perbandingan berbagai sistem sebelumnya yang relevan beserta kelebihan dan keterbatasannya yang disajikan pada Tabel 2.1.

Tabel 2. 1 Tinjauan Solusi dan Sistem Terkait

Peneliti/ Tahun	Nama Sistem/ Aplikasi	Fitur Utama	Teknologi	Kelebihan	Keterbatasan
Fajar <i>et al.</i> (2025)	<i>Optimization of Inventory Management with QR Code</i>	Manajemen persediaan, pencarian data	Web, Laravel, MySQL	Akurasi pencatatan tinggi (95%), pencarian barang sangat cepat.	Hanya berfokus pada satu kategori inventaris dasar.
Anggara <i>et al.</i> (2024)	<i>Web-Based QR-Code Info System in</i>	Pencatatan inventaris gudang	Web, PHP, MySQL	Mengurangi <i>human error</i> ,	Tidak memiliki fitur peminjaman

	<i>Warehouse Inventory</i>			akses data transparan.	(<i>borrowing</i>) peralatan.
Rahim et al. (2025)	<i>Warehouse Management System</i>	Tata kelola ruang gudang dan stok	Web, PHP, MySQL	Meningkatkan transparansi dan kecepatan operasional.	Belum mengimplementasikan teknologi pemindaian <i>QR Code</i> .
Saputra et al. (2024)	<i>Optimizing Inventory Receipt Using QR Code</i>	Penerimaan barang masuk	Aplikasi <i>Mobile</i>	Pemindaian cepat, meminimalkan kesalahan input.	Berbasis <i>mobile</i> , tidak sesuai untuk admin yang bekerja menggunakan PC/Laptop.
Proyek Ini (Yunita, 2026)	Sistem Manajemen Aset Berbasis <i>QR Code</i>	Kelola 6 kategori aset, peminjaman <i>Equipment</i> , ekspor PDF	Web, PHP Native (MVC), MySQL	Mengelola multi-kategori aset, fitur peminjaman <i>Equipment</i> yang spesifik, dan <i>dashboard</i> terpusat.	Fokus pada operasional internal, pemeliharaan sistem dibatasi selama masa pasca-implementasi.

Melalui analisis komparatif dari berbagai sistem terkait di atas, terlihat jelas bahwa keandalan dan fleksibilitas arsitektur perangkat lunak sangat menentukan keberhasilan sistem di lingkungan industri. Oleh karena itu, untuk mendukung realisasi dari sistem manajemen aset yang diusulkan di PT Dalaz Teknik Utama, subbab berikut akan membahas mengenai landasan teori dan konsep-konsep dasar teknologi yang digunakan sebagai fondasi pembangunan perangkat lunak ini.

2.2. Landasan Teori

Bagian ini memuat konsep-konsep dasar yang mendukung pengembangan sistem manajemen aset berbasis *QR Code* di PT Dalaz Teknik Utama. Teori yang dijelaskan berkaitan langsung dengan topik dan teknologi yang digunakan dalam proyek ini.

2.2.1 Sistem Informasi

Sistem informasi terdiri dari bagian-bagian yang terhubung dan berfungsi bersama untuk mencapai tujuan tertentu (Lim *et al.*, 2021). Sistem ini terdiri dari elemen-elemen yang terintegrasi seperti individu, teknologi, data, dan proses yang bekerja sama untuk mengumpulkan, mengelola, menyimpan, dan berbagi informasi (Sumberajar, 2023). Dalam studi ini, sistem informasi diterapkan untuk mengelola data aset di PT Dalaz Teknik Utama.

2.2.2 Manajemen Aset

Manajemen aset adalah pendekatan strategis yang melibatkan pengelolaan, pemantauan, dan penggunaan aset secara efektif dan efisien untuk membantu mencapai tujuan organisasi. Proses ini mencakup beberapa tahap kunci seperti perencanaan, pengadaan, pengoperasian, dan akhirnya peremajaan atau pembuangan aset (PPM School, 2024). Di PT Dalaz Teknik Utama, manajemen aset digunakan untuk mengelola enam kategori aset, yaitu *Office Equipment*, *Equipment*, *Inventory*, *Vehicle*, *Land & Building*, dan *Intangible Assets*.

2.2.3 QR Code (Quick Response Code)

Kode QR adalah kode batang dua dimensi yang menyimpan informasi dalam pola persegi (Pangaila *et al.*, 2022). Kode-kode ini dapat menyimpan berbagai jenis data, dengan kapasitas maksimum 7.089 karakter numerik atau 4.296 karakter alfanumerik (Anggraeni, 2025). Dalam sistem manajemen aset ini, *QR Code* berfungsi sebagai penanda identitas unik (*unique identifier*) yang diimplementasikan secara khusus untuk kategori *Office Equipment* dan *Equipment*, guna mempermudah proses identifikasi, pencatatan, dan pelacakan aset secara efisien dan akurat (Pangaila *et al.*, 2022).

2.2.4 Website

Website adalah kumpulan halaman web yang saling terhubung dan dapat diakses melalui internet menggunakan web *browser*. Sistem manajemen aset yang dikembangkan merupakan *website* dinamis yang memungkinkan pengguna untuk mengelola data aset, memproses pemindaian *QR Code*, dan melakukan transaksi

peminjaman *Equipment*. Sistem ini menyediakan fitur laporan dan *dashboard* yang menampilkan status ketersediaan aset secara langsung berdasarkan data transaksi terakhir yang tersimpan dalam basis data.

2.2.5 PHP (*Hypertext Preprocessor*)

PHP (*Hypertext Preprocessor*) adalah bahasa pemrograman sisi server yang umum digunakan untuk mengembangkan aplikasi web dinamis. PHP bersifat *open-source*, ramah pengguna, dan didukung oleh komunitas yang kuat (Achour *et al.*, 2022). PHP mampu bekerja dengan berbagai sistem basis data dan kompatibel dengan berbagai kerangka kerja dan gaya arsitektur seperti MVC.

2.2.6 Pola Arsitektur MVC (*Model-View-Controller*)

Model-View-Controller (MVC) adalah pendekatan desain perangkat lunak yang membagi aplikasi menjadi tiga bagian utama: *Model*, *View*, dan *Controller*. *Model* menangani data dan aturan yang mengatur cara data diproses, *View* bertanggung jawab untuk menampilkan data kepada pengguna, dan *Controller* berfungsi sebagai penghubung antara *Model* dan *View* (Dicoding, 2021). Sistem ini menggunakan arsitektur MVC untuk memisahkan logika aplikasi, cara aplikasi ditampilkan kepada pengguna, dan pengelolaan data satu sama lain.

2.2.7 MySQL

MySQL adalah sistem manajemen basis data relasional yang bersifat *open-source* dan banyak digunakan untuk aplikasi web dan menggunakan SQL sebagai bahasa untuk mengelola data. Dalam sistem manajemen aset ini, MySQL digunakan sebagai *database* untuk menyimpan semua data aset termasuk data *Office Equipment*, *Equipment*, *Inventory*, *Vehicle*, *Land & Building*, *Intangible Assets*, data pengguna, dan data transaksi peminjaman.

2.2.8 PT Dalaz Teknik Utama

PT Dalaz Teknik Utama adalah perusahaan yang bergerak di bidang layanan inspeksi peralatan berat. Perusahaan ini memiliki berbagai jenis aset yang harus dikelola dengan baik untuk memastikan kelancaran operasional bisnis. Aset-aset ini dibagi menjadi enam kategori utama yaitu *Office Equipment*, *Equipment*, *Inventory*,

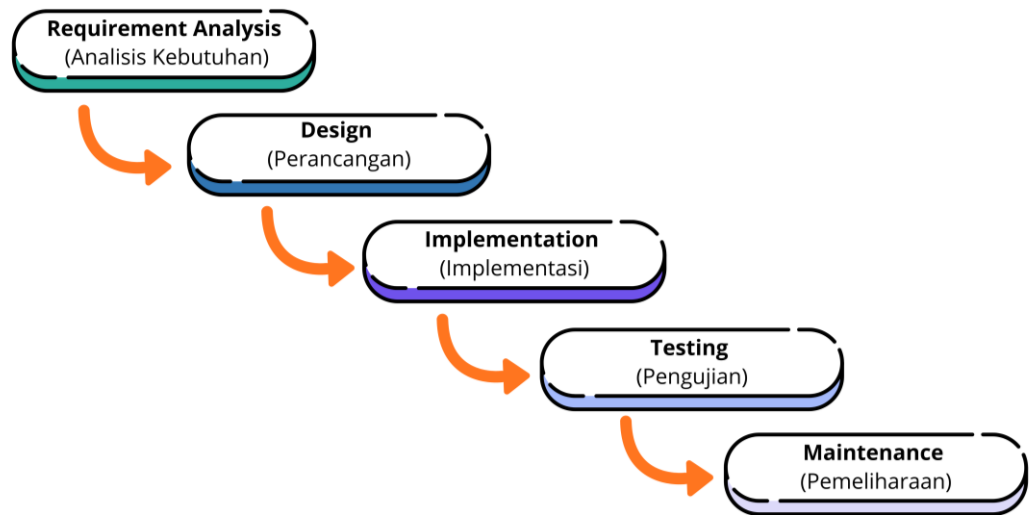
Vehicle, Land & Building, dan *Intangible Assets*. Sebelumnya, proses inventarisasi dicatat secara manual menggunakan *spreadsheet* dan kertas. Metode manual ini menyebabkan kesulitan dalam pelacakan aset, keterlambatan pelaporan, dan kesalahan pencatatan, terutama pada kategori *Equipment* yang sering dipinjam oleh *inspector* ke lapangan. Oleh karena itu, diperlukan sistem manajemen aset berbasis *QR Code* yang dapat mengintegrasikan keenam pendataan aset tersebut secara terpadu, serta mengotomatisasi alur peminjaman khusus untuk *Equipment* agar pelacakan logistik lebih akurat.

2.3. Metode Pengembangan Produk

Metode pengembangan yang digunakan dalam proyek akhir ini adalah *Waterfall*. Model *Waterfall* adalah proses pengembangan perangkat lunak bertahap yang berlangsung secara linier, dari satu fase ke fase berikutnya, mirip dengan air yang mengalir turun dari air terjun (Bassil, 2012). Metode ini pertama kali diusulkan oleh Dr. Winston W. Royce pada tahun 1970 untuk mengatasi kebutuhan akan cara yang lebih terorganisir dan terdokumentasi dengan baik dalam pengembangan perangkat lunak (Dicoding, 2025).

Menurut Bassil (2012), metode *waterfall* terdiri dari lima tahap yang harus diselesaikan secara berurutan, satu demi satu. Tahap-tahap tersebut adalah *Requirements Analysis* (Analisis Kebutuhan), *Design* (Perancangan), *Implementation* (Implementasi), *Testing* (Pengujian), dan *Maintenance* (Pemeliharaan). Metode *waterfall* menawarkan pendekatan yang sederhana dan terstruktur, yang membantu dalam mengelola proyek dan dokumentasi (Telkom University, 2024).

Berikut adalah penjelasan setiap tahapan metode *waterfall* beserta rencana kegiatan yang akan dilakukan dalam konteks proyek sistem manajemen aset berbasis *QR Code* di PT Dalaz Teknik Utama :



Gambar 2. 1 Metode *Waterfall*

2.3.1. *Requirements Analysis* (Analisis Kebutuhan)

Menurut Bassil (2012), tahap ini dimulai dengan memahami kebutuhan dan tujuan perangkat lunak yang akan dikembangkan. Tim pengembangan akan menganalisis persyaratan pengguna dan menentukan fitur dan fungsi yang diperlukan (Telkom University, 2024). Selama fase ini, semua persyaratan proyek dikumpulkan dari klien atau pemangku kepentingan untuk memastikan semua pihak yang terlibat memiliki pemahaman yang jelas dan sama tentang apa yang akan menjadi produk akhir (Dicoding, 2025).

Rencana kegiatan yang akan dilakukan:

1. Melakukan wawancara dengan pihak PT Dalaz Teknik Utama untuk mengidentifikasi kebutuhan sistem manajemen aset.
2. Menganalisis proses bisnis yang sedang berjalan terkait pengelolaan *Office Equipment, Equipment, Inventory, Vehicle, Land & Building*, dan *Intangible Assets*.
3. Mengidentifikasi permasalahan yang dihadapi dalam pencatatan manual menggunakan *spreadsheet* dan kertas.
4. Menyusun kebutuhan fungsional dan non-fungsional sistem.

5. Menentukan fitur-fitur sistem seperti manajemen data aset, peminjaman *Equipment*, *scanning QR Code*, dan pelaporan.

2.3.2. Design (Perancangan)

Setelah memahami kebutuhan, tahap selanjutnya adalah mengembangkan arsitektur, desain, dan spesifikasi teknis perangkat lunak (Telkom University, 2024). Proses ini juga mencakup pembuatan diagram alur dan desain antarmuka pengguna (Telkom University, 2024). Seperti yang disebutkan oleh Bassil (2012), fase ini menghasilkan dokumen desain sistem yang berfungsi sebagai panduan selama tahap implementasi.

Rencana kegiatan yang akan dilakukan:

1. Merancang arsitektur sistem menggunakan pola MVC (*Model-View-Controller*).
2. Membuat desain basis data dengan *Entity Relationship Diagram* (ERD).
3. Merancang *use case* diagram untuk menggambarkan interaksi pengguna dengan sistem.
4. Membuat desain antarmuka pengguna (*wireframe*) untuk setiap fitur sistem.
5. Merancang alur kerja sistem peminjaman dan pengembalian *Equipment* berbasis *QR Code*.
6. Menentukan teknologi yang akan digunakan: PHP *Native* (MVC), MySQL, dan *library QR Code*.

2.3.3. Implementation (Implementasi)

Proses implementasi meliputi penulisan kode program dan pengujian untuk memastikan kualitas perangkat lunak yang sedang dikembangkan (Telkom University, 2024). Pada fase ini, desain yang telah ditetapkan pada tahap sebelumnya diubah menjadi kode program yang sebenarnya (Bassil, 2012).

Rencana kegiatan yang akan dilakukan :

1. Membuat struktur folder sesuai dengan arsitektur MVC (*Model, View, Controller*).

2. Mengimplementasikan *database* MySQL berdasarkan ERD yang telah dirancang.
3. Mengimplementasikan fitur *generate* dan *scanning QR Code* untuk identifikasi aset.
4. Membuat fitur peminjaman dan pengembalian *Equipment*.
5. Mengimplementasikan sistem autentikasi dan otorisasi pengguna (Administrator dan *User/Inspector*).
6. Membuat fitur pelaporan aset dengan *export* ke PDF.

2.3.4. Testing (Pengujian)

Setelah kode program dikembangkan sepenuhnya, tahap pengujian dilakukan untuk memastikan bahwa perangkat lunak berfungsi sesuai dengan yang diharapkan dan memenuhi kebutuhan pengguna (Telkom University, 2024). Seperti yang disebutkan oleh Bassil (2012), tahap ini dirancang untuk mengidentifikasi dan menyelesaikan kesalahan atau masalah sebelum sistem digunakan secara aktual.

Rencana kegiatan yang akan dilakukan :

1. Melakukan pengujian *Black Box Testing* untuk menguji fungsionalitas setiap fitur sistem dan pengujian *User Acceptance Testing* (UAT).
2. Menguji fitur manajemen data aset (tambah, *edit*, hapus, lihat) untuk semua kategori aset.
3. Menguji fitur *generate* dan *scanning QR Code* untuk memastikan identifikasi aset berjalan dengan baik.
4. Menguji fitur peminjaman dan pengembalian *Equipment* termasuk validasi data.
5. Menguji fitur pelaporan dan *export* data ke PDF.
6. Menguji sistem autentikasi dan hak akses pengguna.
7. Memperbaiki *bug* atau kesalahan yang ditemukan selama pengujian.

2.3.5. *Maintenance* (Pemeliharaan)

Proses pemeliharaan dilakukan setelah produk tersedia bagi pengguna. Tim pengembangan terus bekerja pada peningkatan, pembaruan, dan fitur perangkat lunak baru berdasarkan kebutuhan pengguna (Telkom University, 2024). Seperti yang disebutkan oleh Bassil (2012), tahap ini berlanjut sepanjang siklus hidup perangkat lunak untuk memastikan sistem tetap berfungsi dan beroperasi secara efektif. Rencana kegiatan yang akan dilakukan :

1. Melakukan *deployment* sistem ke server PT Dalaz Teknik Utama.
2. Memberikan pelatihan kepada karyawan tentang cara menggunakan sistem.
3. Menyediakan dokumentasi pengguna dan dokumentasi teknis.
4. Melakukan *monitoring* terhadap kinerja sistem setelah *deployment*.
5. Menerima *feedback* dari pengguna untuk perbaikan atau penambahan fitur di masa mendatang.
6. Melaksanakan pemeliharaan (*maintenance*) sistem secara berkala dengan batasan waktu penyelesaian selama 6 bulan setelah sistem berhasil di-*hosting* dan diserahkan kepada PT Dalaz Teknik Utama.

BAB III

ANALISIS DAN PERANCANGAN

3.1. Analisis Kebutuhan

3.1.1 Proses Bisnis Berjalan

PT Dalaz Teknik Utama merupakan perusahaan yang bergerak di bidang layanan inspeksi peralatan berat. Dalam operasionalnya, perusahaan memiliki berbagai jenis aset yang perlu dikelola secara efektif untuk mendukung kegiatan inspeksi. Aset-aset tersebut dikategorikan menjadi enam jenis utama, yaitu *Office Equipment* (peralatan kantor), *Equipment* (peralatan inspeksi khusus lapangan), *Inventory* (persediaan/barang habis pakai), *Vehicle* (kendaraan operasional), *Land & Building* (tanah dan bangunan), serta *Intangible Assets* (aset tak berwujud seperti lisensi).

Saat ini, proses pengelolaan aset di PT Dalaz Teknik Utama masih dilakukan secara manual, dengan metode yang berbeda untuk setiap jenis aset. Pencatatan inventaris untuk *Office Equipment*, *Vehicle*, *Land & Building*, dan *Intangible Assets* saat ini dikelola menggunakan *spreadsheet*. Sementara itu, pencatatan transaksi masuk-keluar untuk *inventory* serta *Equipment* dicatat secara manual di atas kertas.

Metode pencatatan data secara manual ini menyebabkan beberapa masalah operasional seperti :

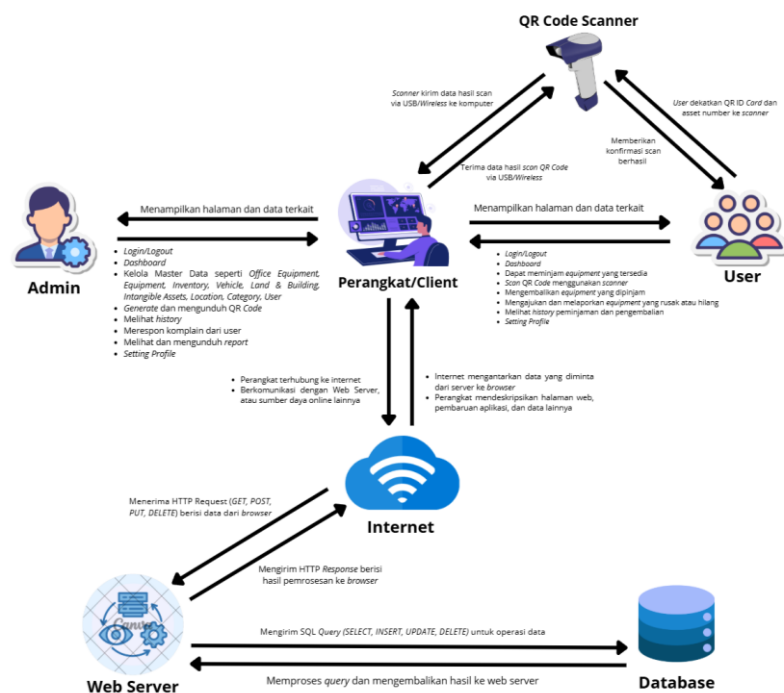
1. Sulit untuk melacak lokasi aset pada waktu tertentu, terutama untuk peralatan yang digunakan oleh *inspector* di lapangan.
2. Ada keterlambatan saat mencari informasi aset karena harus membuka file *spreadsheet* atau memeriksa dokumen kertas.
3. Data sering terulang dan informasi mungkin berbeda antara entri *spreadsheet* dan catatan tertulis.
4. Ada risiko kehilangan data jika dokumen kertas rusak atau file *spreadsheet* menjadi korup.

5. Kesalahan manusia selama penginputan manual dapat memengaruhi akurasi data inventaris.
6. Kesulitan dalam membuat laporan aset secara cepat untuk keperluan audit atau pengambilan keputusan manajemen.
7. Tidak ada sistem yang memadai untuk memantau bagaimana peralatan dan bahan habis pakai digunakan.

3.1.2 Gambaran Umum Sistem

Untuk mengatasi tantangan dalam proses bisnis yang ada, dirancang sebuah Sistem Manajemen Aset berbasis *QR Code* yang terintegrasi dan terkomputerisasi. Sistem ini akan menggantikan metode pencatatan manual dengan platform berbasis web yang dapat digunakan oleh Administrator dan Pengguna (*Inspector*), masing-masing dengan tingkat akses yang berbeda.

Sistem ini terdiri dari tiga bagian utama yaitu *Client* (Admin dan *User*), *Web Server*, dan *Database*. Proses dimulai ketika *User* meminjam *Equipment* dengan memindai Kode QR pada kartu karyawan mereka dan kode QR yang terpasang pada *asset Equipment* menggunakan *QR Code Scanner*. Data yang dipindai kemudian dikirim melalui internet ke server web untuk diproses. Server web memeriksa data, menyimpan detail peminjaman di *database*, dan mengirim konfirmasi kembali ke klien.



Gambar 3. 1 Gambaran Umum Sistem

Gambar 3.1 menunjukkan arsitektur sistem, yang terdiri dari beberapa bagian utama yang bekerja sama untuk membentuk sistem manajemen aset. Administrator memiliki kendali penuh atas berbagai bagian sistem. Mereka dapat mengelola proses *login* dan *logout*, melihat dasbor dengan statistik terkait aset, mengelola data master keenam kategori aset (*Office Equipment, Equipment, Inventory, Vehicle, Land & Building, Intangible Assets*), *Location, Category*, dan *User Data*, membuat kode QR untuk setiap aset dan pengguna, melihat catatan lengkap peminjaman peralatan, menanggapi keluhan pengguna, mencatat transaksi masuk/keluar barang habis pakai, mengunduh laporan dalam format PDF, dan mengatur profil pengguna.

Untuk pengguna, seperti *inspector* memiliki akses terbatas. Mereka dapat *login* dan *logout*, melihat *dashboard*, meminjam peralatan dengan memindai kode QR, mengembalikan peralatan dan melaporkan kondisinya, mengajukan komplain barang rusak atau hilang, memeriksa riwayat peminjaman mereka sendiri, dan mengelola pengaturan profil mereka.

QR Code scanner adalah perangkat fisik berupa pemindai *QR Code* yang terhubung ke komputer melalui USB atau *wireless*. Perangkat ini digunakan untuk memindai QR nomor karyawan dan *Equipment*. Setelah pemindaian selesai, pemindai mengirimkan data ke sistem dan memberikan umpan balik, seperti bunyi bip atau lampu LED, untuk memberitahu pengguna bahwa pemindaian berhasil, memastikan pengalaman yang lancar dan responsif.

Perangkat atau klien adalah laptop atau komputer yang terhubung ke sistem melalui *browser* web seperti Chrome, Firefox, atau Edge. Perangkat ini dapat menerima masukan dari pemindai QR atau secara manual oleh pengguna. Kemudian, perangkat tersebut mengirimkan permintaan HTTP ke server, yang memproses permintaan tersebut dan mengembalikan respons berupa halaman web atau data.

Internet berfungsi sebagai jembatan antara klien dan server, membantu mengirim dan menerima permintaan dan respons HTTP. Hal ini memastikan bahwa data ditransmisikan dengan aman dan efisien.

Server web beroperasi menggunakan PHP *Native* dan mengikuti pola desain MVC (*Model-View-Controller*). Server ini menangani berbagai tugas seperti memproses logika bisnis, autentikasi pengguna menggunakan *bcrypt*, mengelola sesi dengan kontrol akses berbasis peran, menghasilkan kode QR menggunakan *library phpqrcode*, dan memvalidasi masukan untuk mencegah injeksi SQL.

Database yang digunakan adalah sistem manajemen *database* relasional bernama MySQL. *Database* ini menyimpan semua data sistem dalam tabel terstruktur, termasuk data pengguna, data master keenam kategori aset (*Office Equipment, Equipment, Inventory, Vehicle, Land & Building, Intangible Assets*), lokasi, kategori, catatan peminjaman, riwayat, dan keluhan.

3.2. Perancangan

3.2.1 Kebutuhan Fungsional

Kebutuhan Fungsional yang terdapat pada Sistem Manajemen Aset dapat dilihat pada tabel berikut :

Tabel 3. 1 Kebutuhan Fungsional

Kode	Kebutuhan Fungsional
FR-01	Admin dan <i>User</i> dapat melakukan <i>Login</i>
FR-02	Admin dan <i>User</i> dapat melihat <i>Dashboard</i> dengan statistik aset
FR-03	Admin dapat mengelola data <i>Office Equipment, Vehicle, Land & Building, dan Intangible Assets</i>
FR-04	Admin dapat mengelola data <i>Equipment</i>
FR-05	Admin dapat mengelola data <i>Inventory</i>
FR-06	Admin dapat mengelola data <i>Location</i>
FR-07	Admin dapat mengelola data <i>Category</i>
FR-08	Admin dapat mengelola data <i>User</i>
FR-09	Admin dapat melakukan <i>generate</i> dan mengunduh (<i>download</i>) <i>QR Code</i> pada kategori <i>Office Equipment, Equipment, serta User</i>
FR-10	<i>User</i> dapat melakukan peminjaman <i>Equipment</i> dengan <i>scan QR Code</i>
FR-11	<i>User</i> dapat melakukan pengembalian <i>Equipment</i> dengan <i>scan QR Code</i>
FR-12	<i>User</i> dapat mengajukan <i>Complaint</i> barang rusak/hilang saat pengembalian, disertai unggahan foto bukti khusus untuk kondisi barang rusak.
FR-13	Admin dan <i>User</i> dapat melihat <i>History</i> peminjaman dengan detail
FR-14	Admin dapat mencatat transaksi <i>Inventory In/Out</i>
FR-15	Admin dapat memberikan respon <i>Complaint</i> dari <i>User</i>
FR-16	Admin dapat mengunduh laporan data khusus untuk kategori <i>Equipment</i> dan laporan kerusakan (<i>Defect Report</i>) pada <i>Complaint</i> dalam format PDF.
FR-17	Admin dan <i>User</i> dapat melakukan <i>Logout</i>

3.2.2 Kebutuhan Non-Fungsional

Kebutuhan Non-Fungsional yang terdapat pada Sistem Manajemen Aset dapat dilihat pada tabel berikut :

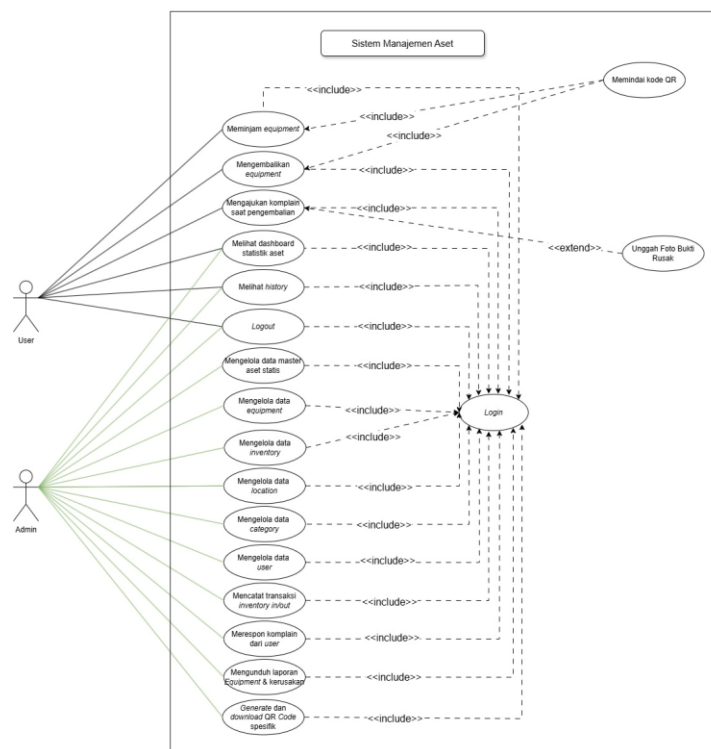
Tabel 3. 2 Kebutuhan Non-Fungsional

Kode	Parameter	Kebutuhan Non-Fungsional
NFR-01	<i>Security</i>	Sistem harus memiliki keamanan dengan autentikasi <i>login, hashing bcrypt</i> untuk <i>password</i> , dan <i>role-based access control</i>

NFR-02	<i>Performance</i>	Sistem dapat memproses transaksi, pencarian data, dan verifikasi <i>QR Code</i> dalam waktu maksimal 5 detik pada kondisi jaringan internet stabil.
NFR-03	<i>Usability</i>	Sistem memiliki tampilan (UI/UX) yang rapi, terstruktur, <i>user-friendly</i> , intuitif, dan <i>responsive</i> .
NFR-04	<i>Portability</i>	Bahasa yang digunakan pada antarmuka sistem adalah Bahasa Inggris, kecuali pada dokumen laporan yang diekspor dalam format PDF yang menggunakan Bahasa Indonesia sesuai kebutuhan pelaporan perusahaan

3.2.3 Diagram Use case

Diagram *use case* menggambarkan interaksi antara aktor (Admin dan *User*) dengan sistem manajemen aset berbasis *QR Code*. Sistem memiliki dua aktor utama yaitu Administrator yang memiliki akses penuh terhadap semua fitur sistem, dan *User (Inspector)* yang memiliki akses terbatas untuk melakukan peminjaman dan pengembalian *Equipment*



Gambar 3.2 Usecase Diagram

Gambar 3.2 memvisualisasikan interaksi antara aktor dengan sistem, di mana Admin memiliki wewenang manajerial yang meliputi : melihat statistik *dashboard*, mengelola data master aset statis (*Office Equipment, Vehicle, Land & Building, Intangible Assets*), mengelola data *Equipment* dan *Inventory*, mengatur lokasi dan kategori, mengelola akun *User*, membuat dan mengunduh *QR Code* spesifik, mencatat mutasi stok *Inventory In/Out*, memberikan respon komplain, melihat riwayat peminjaman, serta mengunduh laporan logistik (PDF). Sementara itu, aktor *User* (sebagai *Inspector*) memiliki akses operasional yang meliputi : memantau *dashboard*, melakukan peminjaman dan pengembalian *Equipment*, mengajukan komplain (*Complaint*), serta melihat riwayat transaksi pribadi. Kedua aktor memiliki fungsionalitas dasar yang sama untuk melakukan *Login* dan *Logout*.

Diagram tersebut juga memetakan prosedur sistem melalui relasi `<<include>>` dan `<<extend>>`. Relasi `<<include>>` menunjukkan sebuah prasyarat wajib; di mana *use case Login* harus dilakukan terlebih dahulu sebelum aktor dapat mengakses fitur-fitur utama, dan *use case* "Memindai Kode QR" wajib dieksekusi oleh sistem setiap kali *User* melakukan peminjaman maupun pengembalian *Equipment*. Sedangkan relasi `<<extend>>` menunjukkan fungsi situasional, di mana *use case* "Mengunggah Foto Bukti Rusak" bertindak sebagai perluasan dari aktivitas komplain yang hanya akan aktif jika *User* mengembalikan barang dalam kondisi cacat atau rusak.

3.2.4 Skenario *Use case*

Login merupakan fitur utama yang menjadi pintu masuk bagi semua pengguna untuk mengakses sistem. Admin dan *User* harus melakukan autentikasi terlebih dahulu sebelum dapat menggunakan fitur-fitur sistem.

Tabel 3. 3 Skenario *Use case Login*

Aspek	Deskripsi
Nomor	UC-01
Nama	Melakukan <i>Login</i>
Deskripsi	Proses autentikasi pengguna sebelum mengakses sistem
<i>Primary Actor</i>	Admin dan <i>User</i>
<i>Preconditions</i>	Pengguna sudah memiliki akun terdaftar di sistem
<i>Postconditions</i>	Pengguna berhasil <i>login</i> dan diarahkan ke <i>dashboard</i> sesuai <i>role</i>
<i>Main Success Scenario</i>	1. Pengguna membuka halaman <i>login</i> 2. Sistem menampilkan form <i>login</i> (<i>Username</i> dan <i>password</i>) 3. Pengguna memasukkan <i>Username</i> dan <i>password</i> 4. Pengguna menekan tombol <i>Login</i> 5. Sistem memvalidasi kredensial pengguna dengan <i>database</i> 6. Sistem membuat session dengan informasi <i>User_id</i>, <i>role</i> 7. Jika <i>role</i> Admin, sistem mengarahkan ke <i>Dashboard Admin</i> 8. Jika <i>role User</i>, sistem mengarahkan ke <i>Dashboard User</i>
<i>Alternative Flow</i>	5a. Jika ada <i>field</i> yang kosong: - Sistem tampilkan "<i>Username and password must be filled in.</i>" 5b. Jika <i>Username/password</i> salah: - Sistem menampilkan pesan "<i>Invalid Username or password.</i>" - Kembali ke <i>Step 2</i>

Dashboard merupakan halaman utama yang menampilkan ringkasan statistik aset secara visual untuk memudahkan Admin dan *User* dalam memonitor kondisi aset perusahaan terkini.

Tabel 3. 4 Skenario *Use case* Melihat *Dashboard* Statistik Aset

Aspek	Deskripsi
Nomor	UC-02
Nama	Melihat <i>Dashboard</i> Statistik Aset
Deskripsi	Admin dapat melihat statistik dan <i>monitoring</i> aset terkini
<i>Primary Actor</i>	Admin
<i>Preconditions</i>	Admin dan <i>User</i> sudah <i>login</i> ke sistem
<i>Postconditions</i>	<i>Dashboard</i> menampilkan data statistik aset terkini
<i>Main Success Scenario</i>	<i>Dashboard Admin:</i> 1. Admin <i>login</i> berhasil 2. Sistem mengarahkan ke <i>Dashboard Admin</i> 3. Sistem menampilkan data statistik aset 5. <i>Dashboard</i> ter-update otomatis setiap ada perubahan data <i>Dashboard User:</i> 1. <i>User login</i> berhasil 2. Sistem mengarahkan ke <i>Dashboard User</i> 3. Sistem menampilkan data statistik untuk <i>User</i> tersebut 4. Sistem menampilkan <i>section</i>: - <i>Notifications</i> (pemberitahuan) - <i>Recent History</i> (riwayat peminjaman terbaru) - <i>Upcoming Returns</i> (<i>Equipment</i> yang harus segera dikembalikan) - <i>Popular Assets</i> (<i>Equipment</i> yang sering dipinjam) 5. <i>Dashboard</i> terupdate setiap <i>User</i> melakukan transaksi
<i>Alternative Flow</i>	3a. Jika tidak ada data (Admin): - Sistem menampilkan "<i>No asset data available</i>" 3b. Jika <i>User</i> belum pernah meminjam (<i>User</i>): - Sistem menampilkan "<i>No borrowing activity yet</i>"

Mengelola Data Aset Statis merupakan fitur untuk Admin dalam menambah, mengubah, dan menghapus data aset *Office equipment, Land & Building, Vehicle, dan Intangible Assets*.

Tabel 3. 5 Skenario Use case Mengelola Data Aset Statis

Aspek	Deskripsi
Nomor	UC-03
Nama	Mengelola Master Data Aset Statis
Deskripsi	Admin dapat menambah (manual atau <i>Import CSV</i>), mengubah, dan menghapus data <i>Office Equipment, Vehicle, Land & Building, dan Intangible Assets</i> .
<i>Primary Actor</i>	Admin
<i>Preconditions</i>	Admin sudah <i>login</i> ke sistem
<i>Postconditions</i>	Data aset statis tersimpan/ter-update/terhapus di <i>database</i>
<i>Main Success Scenario</i>	Menambah <i>Asset</i> (Manual): 1. Admin klik menu <i>Asset Data</i> > (Pilih salah satu dari 4 kategori statis). 2. Sistem menampilkan tabel data aset 3. Admin klik tombol <i>Add</i> 4. Sistem menampilkan modal form input 5. Admin mengisi data: <i>Asset Name, Category</i> (pilih salah satu dari 4 kategori statis), <i>Location, Quantity, Description, Pictures</i> 6. Admin klik tombol <i>Save</i> 7. Sistem validasi input 8. Sistem <i>generate QR Code</i> (khusus untuk <i>Office Equipment</i>). 9. Sistem simpan data ke <i>database</i> 10. Sistem menampilkan pesan "<i>Asset Added successfully</i>" 11. Sistem <i>refresh</i> tabel dan menampilkan data baru Tambah via <i>Import CSV</i>: 1. Admin klik tombol "<i>Import CSV</i>"

Aspek	Deskripsi
	2. Sistem menampilkan <i>modal Import</i> dengan petunjuk pengisian template CSV 3. Admin memilih <i>file</i> CSV dari perangkat dan klik "<i>Upload & Import</i>" 4. Sistem memproses baris data, otomatis men-<i>generate QR Code</i> (khusus untuk <i>Office Equipment</i>), dan menampilkan laporan jumlah data yang sukses/gagal diimpor. <i>Edit Asset:</i> 1. Admin klik tombol <i>Edit</i> pada <i>row</i> data 2. Sistem tampilkan modal form dengan data terisi 3. Admin ubah data yang diperlukan 4. Admin klik <i>Update</i> 5. Sistem simpan perubahan 6. Sistem tampilkan pesan "<i>Data updated successfully</i>" <i>Hapus Asset:</i> 1. Admin klik tombol Delete 2. Sistem tampilkan konfirmasi "<i>Are you sure you want to delete this asset?</i>" 3. Admin klik <i>Yes</i> 4. Sistem hapus data dari <i>database</i> 5. Sistem tampilkan pesan "<i>Data deleted successfully</i>"
<i>Alternative Flow</i>	7a. Jika ada <i>field</i> yang kosong: - Sistem tampilkan "<i>All fields are required</i>" 7b. Jika nama aset sudah ada: - Sistem tampilkan "<i>Asset with this Name already exists</i>"

Mengelola Data *Equipment* memungkinkan Admin untuk menambah, mengubah, dan menghapus data peralatan inspeksi yang dapat dipinjam oleh *inspector* dengan detail spesifikasi yang lengkap.

Tabel 3. 6 Skenario *Use case* Mengelola Data *Equipment*

Aspek	Deskripsi
Nomor	UC-04
Nama	Mengelola Data <i>Equipment</i>
Deskripsi	Admin dapat menambah (manual atau <i>Import CSV</i>), mengubah, dan menghapus data peralatan inspeksi (<i>Equipment</i>).
<i>Primary Actor</i>	Admin
<i>Preconditions</i>	Admin sudah <i>login</i> ke sistem
<i>Postconditions</i>	Data <i>Equipment</i> tersimpan/terupdate/terhapus di <i>database</i>
<i>Main Success Scenario</i>	Menambah <i>Equipment</i> (Manual) : 1. Admin klik menu <i>Asset Data > Equipment</i> 2. Sistem menampilkan tabel data <i>Equipment</i> 3. Admin klik tombol <i>Add New Equipment</i> 4. Sistem menampilkan modal form lengkap 5. Admin mengisi form <i>Equipment</i> 6. Admin klik tombol <i>Save</i> 7. Sistem validasi dan <i>generate QR Code</i> 8. Sistem simpan data 9. Sistem tampilkan pesan "<i>Equipment Added successfully</i>" Tambah via <i>Import CSV</i>: 1. Admin klik tombol "<i>Import CSV</i>". 2. Sistem menampilkan <i>modal Import</i> dengan instruksi CSV dan tautan unduh template. 3. Admin memilih <i>file</i> CSV yang sudah diisi dan klik "<i>Upload & Import</i>". 4. Sistem memproses seluruh baris data, otomatis membuat <i>QR Code</i> untuk setiap <i>Equipment</i> baru, dan menampilkan notifikasi sukses/gagal.
<i>Alternative Flow</i>	7a. Jika <i>Asset Number</i> sudah ada: - Sistem tampilkan "<i>Asset Number already in use</i>"

Mengelola Data *Inventory* adalah fitur untuk Admin dalam menambah, mengubah, dan menghapus master data untuk barang-barang persediaan/habis pakai sebelum transaksinya dicatat.

Tabel 3. 7 Skenario *Use case* Mengelola Data *Inventory*

Aspek	Deskripsi
Nomor	UC-05
Nama	Mengelola Data <i>Inventory</i>
Deskripsi	Admin dapat menambah (manual atau <i>Import CSV</i>), mengubah, dan menghapus master data barang persediaan.
<i>Primary Actor</i>	Admin
<i>Preconditions</i>	Admin sudah <i>login</i> ke sistem
<i>Postconditions</i>	Data <i>Inventory</i> tersimpan di <i>database</i>
<i>Main Success Scenario</i>	Tambah <i>Inventory</i> (Manual): 1. Admin klik menu <i>Asset Data > Inventory Data</i> 2. Sistem menampilkan tabel data beserta tombol "+ Add" dan "<i>Import CSV</i>" 3. Admin klik tombol "+ Add" 4. Sistem menampilkan <i>modal form</i> 5. Admin mengisi <i>General Information: Asset Name, Asset Code, Merk, Responsible Person, User, Unit, Minimum Order, Quantity, Location, Category, Supporting Vendor</i>, dan <i>Condition</i> 6. Admin mengisi <i>Documentation: Upload Photos</i> (bisa lebih dari satu) dan <i>Supporting Document</i> (file PDF maks 2MB) 7. Admin klik "<i>Save</i>" 8. Sistem memvalidasi dan menyimpan data ke <i>database</i> beserta <i>file</i> unggahan 9. Sistem menampilkan pesan: "<i>Successfully added</i>" Tambah via <i>Import CSV</i>:

Aspek	Deskripsi
	1. Admin klik tombol <i>"Import CSV"</i> 2. Sistem menampilkan <i>modal Import</i> dengan petunjuk pengisian 3. Admin memilih <i>file</i> CSV dan klik <i>"Upload & Import"</i> 4. Sistem memproses baris data dan menampilkan pesan sukses/gagal
<i>Alternative Flow</i>	8a. Jika <i>Asset Name</i> atau <i>Asset Code</i> kosong: - Sistem menampilkan pesan: <i>"Asset Name & Code must be filled"</i> 8b. Jika <i>Asset Code</i> sudah terdaftar di <i>database</i> : - Sistem menampilkan pesan: <i>"Asset Code already exists"</i>

Mengelola Data *Location* adalah pengelolaan master data lokasi yang digunakan untuk mencatat penempatan aset dan *equipment* di perusahaan.

Tabel 3. 8 Skenario *Use case* Mengelola Data *Location*

Aspek	Deskripsi
Nomor	UC-06
Nama	Mengelola Data <i>Location</i>
Deskripsi	Admin dapat CRUD data lokasi penempatan aset
<i>Primary Actor</i>	Admin
<i>Preconditions</i>	Admin sudah <i>login</i>
<i>Postconditions</i>	Data <i>Location</i> tersimpan di <i>database</i>
<i>Main Success Scenario</i>	1. Admin klik menu <i>Location Data</i> 2. Sistem tampilkan tabel data <i>Location</i> 3. Admin klik <i>Add Location</i> 4. Admin isi form: <i>Location Name</i> 5. Admin klik <i>Save</i>

Aspek	Deskripsi
	6. Sistem simpan data 7. Data muncul di tabel
<i>Alternative Flow</i>	5a. Jika <i>field</i> kosong: - Sistem tampilkan " <i>Location Name</i> " 5b. <i>Edit/Hapus</i> sama seperti UC-03

Mengelola Data *Category* adalah pengelolaan master data kategori yang digunakan untuk mengklasifikasikan aset dan *equipment* berdasarkan jenisnya.

Tabel 3. 9 Skenario *Use case* Mengelola Data *Category*

Aspek	Deskripsi
Nomor	UC-07
Nama	Mengelola Data <i>Category</i>
Deskripsi	Admin dapat CRUD data kategori aset
<i>Primary Actor</i>	Admin
<i>Preconditions</i>	Admin sudah <i>login</i>
<i>Postconditions</i>	Data <i>Category</i> tersimpan di <i>database</i>
<i>Main Success Scenario</i>	1. Admin klik menu <i>Category Data</i> 2. Sistem tampilkan tabel data <i>Category</i> 3. Admin klik <i>Add Category</i> 4. Admin isi form: <i>Category Name</i> 5. Admin klik <i>Save</i> 6. Sistem simpan data 7. Data muncul di tabel
<i>Alternative Flow</i>	5a. Jika <i>field</i> kosong: - Sistem tampilkan " <i>Category Name is required</i> "

Aspek	Deskripsi
	5b. <i>Edit/Hapus</i> sama seperti UC-03

Mengelola Data *User* memungkinkan Admin untuk menambah, mengubah, dan menghapus data pengguna sistem serta mengatur hak akses mereka (Admin atau *User/Inspector*).

Tabel 3. 10 Skenario *Use case* Mengelola Data *User*

Aspek	Deskripsi
Nomor	UC-08
Nama	Mengelola Data <i>User</i>
Deskripsi	Admin dapat CRUD data pengguna sistem
<i>Primary Actor</i>	Admin
<i>Preconditions</i>	Admin sudah <i>login</i>
<i>Postconditions</i>	Data <i>User</i> tersimpan di <i>database</i> dan <i>QR Code</i> ter-generate
<i>Main Success Scenario</i>	1. Admin klik menu <i>User Data</i> 2. Sistem tampilkan tabel data <i>User</i> 3. Admin klik <i>Add User</i> 4. Admin isi form: <i>Full Name, Position, Employee Number, Role</i> (Admin/<i>User</i>) 5. Admin klik <i>Save</i> 6. Sistem simpan data dan generate <i>QR Code</i> untuk <i>User</i> 7. Data muncul di tabel dengan <i>QR Code</i>
<i>Alternative Flow</i>	6a. Jika <i>Employee Number</i> sudah ada: - Sistem tampilkan " <i>Employee Number already registered</i> "

Generate QR Code adalah proses otomatis yang dilakukan sistem untuk membuat kode QR unik bagi setiap *Office Equipment*, *Equipment*, dan *User* yang ditambahkan ke dalam sistem.

Tabel 3. 11 Skenario Use case Generate QR Code

Aspek	Deskripsi
Nomor	UC-09
Nama	<i>Generate dan Download QR Code untuk Aset dan User</i>
Deskripsi	Sistem otomatis melakukan <i>generate QR Code</i> saat tambah/edit data, dan Admin dapat mengunduhnya
<i>Primary Actor</i>	Admin
<i>Preconditions</i>	Admin sedang menambah/edit <i>Office Equipment/Equipment/User</i>
<i>Postconditions</i>	<i>QR Code</i> tersimpan dan berhasil diunduh
<i>Main Success Scenario</i>	1. Admin simpan data <i>Asset/Equipment/User</i> baru 2. Sistem <i>generate QR Code</i> berisi ID unik 3. Sistem <i>update</i> kolom <i>qr_code</i> di <i>database</i> 4. Sistem simpan <i>file QR Code</i> di direktori sistem 5. <i>QR Code</i> ditampilkan di tabel dengan tombol "<i>Download</i>" 6. Admin mengklik tombol "<i>Download</i>" pada <i>QR Code</i> yang diinginkan 7. <i>File QR Code</i> berhasil terunduh ke perangkat Admin untuk dicetak
<i>Alternative Flow</i>	-

Meminjam *Equipment* adalah proses peminjaman peralatan inspeksi oleh *User* menggunakan *QR Code Scanner* untuk identifikasi *User* dan *Equipment* secara cepat dan akurat.

Tabel 3. 12 Skenario *Use case* Meminjam *Equipment* (dengan *QR Scanner*)

Aspek	Deskripsi
Nomor	UC-10
Nama	Meminjam <i>Equipment</i> dengan <i>Scan QR Code</i>
Deskripsi	<i>User</i> meminjam <i>Equipment</i> menggunakan <i>QR Code Scanner</i>
<i>Primary Actor</i>	<i>User</i>
<i>Preconditions</i>	- <i>User</i> sudah <i>login</i> - <i>User</i> memiliki <i>ID Card</i> dengan <i>QR Code</i> - <i>Equipment</i> tersedia
<i>Postconditions</i>	- Data peminjaman tersimpan - Status <i>Equipment</i> berubah jadi "<i>BORROWED</i>" - <i>History</i> tercatat
<i>Main Success Scenario</i>	Step 1: <i>Scan ID Card</i> 1. <i>User</i> membuka menu <i>Borrow</i> 2. Sistem menampilkan "<i>Step 1: Identify Yourself</i>" dan kolom input <i>Employee ID</i> 3. <i>User</i> mengarahkan <i>QR Scanner hardware</i> ke <i>ID Card</i> 4. Alat <i>scanner</i> otomatis mengisi kolom input teks dengan ID karyawan dan mengirimkan data 5. Sistem memvalidasi <i>User</i> dan otomatis pindah ke Step 2 Step 2: <i>Job Information</i> 6. Sistem menampilkan form "<i>Step 2: Job Details</i>" 7. <i>User</i> mengisi form: <i>No JD, Client, Location, Working Days</i> lalu mengklik "<i>Next</i>" Step 3: <i>Scan Equipment</i> 8. Sistem menampilkan "<i>Step 3: Scan Equipment</i>" 9. <i>User</i> mengarahkan <i>QR Scanner</i> ke stiker QR pada <i>Equipment</i>

Aspek	Deskripsi
	10. Alat <i>scanner</i> otomatis mengisi kolom input <i>Asset Code</i> dan mengirimkan data 11. Sistem mengecek ketersediaan dan memunculkan <i>Equipment</i> tersebut di tabel " <i>Equipment Scanned</i> " 12. <i>User</i> mengklik tombol " <i>Confirm Borrowing</i> " 13. Sistem menampilkan <i>Modal Confirm Borrowing</i> untuk memastikan jumlah barang 14. <i>User</i> mengklik tombol " <i>Yes, Borrow</i> " 15. Sistem menyimpan data ke tabel <i>BORROWED</i> dan memperbarui status <i>Equipment</i> 16. Sistem menampilkan pesan " <i>Borrowing Successful!</i> "
<i>Alternative Flow</i>	4a. Jika <i>ID Card</i> tidak terdaftar: - Sistem menampilkan pesan: " <i>User not registered</i> " 10a. Jika <i>Equipment</i> tidak tersedia atau tidak valid: - Sistem menampilkan pesan: " <i>Equipment not available</i> "

Mengembalikan *Equipment* adalah proses pengembalian peralatan inspeksi oleh *User* dengan melaporkan kondisi *Equipment* (*Good*, *Lost* atau *Damaged*) menggunakan *QR Code Scanner*.

Tabel 3. 13 Skenario *Use case* Mengembalikan *Equipment* (dengan *QR Scanner*)

Aspek	Deskripsi
Nomor	UC-11
Nama	Mengembalikan <i>Equipment</i> dengan <i>Scan QR Code</i>
Deskripsi	<i>User</i> mengembalikan <i>Equipment</i> dan melaporkan kondisinya
<i>Primary Actor</i>	<i>User</i>
<i>Preconditions</i>	- <i>User</i> sudah <i>login</i> - <i>User</i> memiliki peminjaman aktif

Aspek	Deskripsi
<i>Postconditions</i>	- Status <i>Equipment</i> kembali "<i>available</i>" atau "<i>Damaged</i>" - Data pengembalian dan <i>History</i> tersimpan
<i>Main Success Scenario</i>	Step 1: Scan <i>ID Card</i> 1. <i>User</i> membuka menu <i>Return</i> 2. Sistem menampilkan "<i>Step 1: Identify Borrower</i>" dan kolom input <i>Employee ID</i> 3. <i>User</i> mengarahkan <i>QR Scanner</i> ke <i>ID Card</i> 4. Alat <i>scanner</i> otomatis mengisi input teks dan mengirimkan data 5. Sistem memvalidasi <i>User</i> dan pindah ke Step 2 Step 2: <i>Return Items</i> 6. Sistem menampilkan tabel <i>Borrowed Items List</i> milik <i>User</i> tersebut 7. <i>User</i> mengarahkan alat <i>scanner</i> ke <i>Equipment</i> fisik (otomatis mengisi kolom <i>Asset QR Code</i>) 8. Sistem memverifikasi aset dan meminta <i>User</i> memilih <i>Condition</i> (<i>Good / Damaged / Lost</i>) 9. Jika <i>User</i> pilih <i>Good</i>: - Sistem mengatur status <i>Equipment</i> menjadi "<i>available</i>" 10. Jika <i>User</i> pilih <i>Damaged</i>: - Sistem memunculkan <i>Modal Defect Report</i> - <i>User</i> wajib mengisi <i>Defect Cause / Chronology</i> dan wajib mengunggah <i>Photos of Defect</i> - <i>User</i> mengklik "<i>Save Report</i>". Sistem menyimpan data ke <i>COMPLAINTS</i> dan mengatur status jadi "<i>Damaged</i>" 11. Jika <i>User</i> pilih <i>Lost</i>: - Sistem memunculkan <i>Modal Defect Report</i> - <i>User</i> wajib mengisi <i>Defect Cause / Chronology</i> saja (tanpa foto) - <i>User</i> mengklik "<i>Save Report</i>". Sistem menyimpan data ke <i>COMPLAINTS</i> dan mengatur status jadi "<i>Lost</i>" 12. <i>User</i> mengklik tombol "<i>Confirm Return</i>" pada halaman utama 13. Sistem menampilkan <i>Modal Confirm Return</i> 14. <i>User</i> mengklik "<i>Yes, Return</i>" 15. Sistem memperbarui tabel <i>RETURN</i>, mencatat <i>History</i>, dan menampilkan pesan "<i>Return Successful!</i>"

Aspek	Deskripsi
<i>Alternative Flow</i>	7a. Jika <i>Equipment</i> yang di-scan bukan milik <i>User</i> tersebut: - Sistem menampilkan pesan: "<i>This Equipment is not borrowed by you</i>" 10a. Jika memilih <i>Damaged</i> tapi tidak mengunggah foto / kronologi: - Sistem menahan proses simpan dan meminta <i>User</i> melengkapinya 11a. Jika memilih <i>Lost</i> tapi tidak mengisi kronologi: - Sistem menahan proses simpan dan meminta <i>User</i> mengisi kronologi

Mengajukan *Complaint* adalah fitur bagi *User* untuk melaporkan kerusakan atau masalah pada *Equipment* kepada Admin untuk mendapatkan tindakan perbaikan atau penggantian.

Tabel 3. 14 Skenario Use case Mengajukan Complaint

Aspek	Deskripsi
Nomor	UC-12
Nama	Mengajukan <i>Complaint</i> (via <i>Defect Report</i>)
Deskripsi	<i>User</i> melaporkan <i>Equipment</i> yang mengalami kerusakan (<i>Damaged</i>) atau kehilangan (<i>Lost</i>) dengan mengisi kronologi kejadian dan mengunggah foto bukti (khusus barang rusak) secara langsung terintegrasi pada saat proses pengembalian barang (<i>Return</i>).
<i>Primary Actor</i>	<i>User</i>
<i>Preconditions</i>	- <i>User</i> sudah login ke sistem. - <i>User</i> sedang memproses pengembalian barang di menu <i>Return</i> dan berada pada antarmuka Step 2: <i>Return Items</i>. - Terdapat data <i>Equipment</i> pada tabel <i>Borrowed Items List</i>.
<i>Postconditions</i>	Data <i>Defect Report</i> tersimpan otomatis ke dalam tabel <i>COMPLAINTS</i> dengan status ' <i>waiting response</i> ' dan diteruskan ke halaman Admin bersamaan dengan selesainya proses konfirmasi pengembalian (<i>Confirm Return</i>).

Aspek	Deskripsi
Main Success Scenario	Aktivasi <i>Modal Defect Report</i> 1. User memilih kondisi "<i>Damaged</i>" atau "<i>Lost</i>" pada <i>dropdown Condition</i> untuk salah satu <i>Equipment</i> di tabel <i>Borrowed Items List</i>. 2. Sistem secara otomatis memunculkan <i>pop-up Modal Defect Report</i>. Pengisian Form Sesuai Kondisi 3. Jika kondisi <i>Damaged</i>: User wajib mengetikkan kronologi pada <i>field Defect Cause / Chronology</i> dan mengunggah file foto pada <i>field Photos of Defect</i>. 4. Jika kondisi <i>Lost</i>: User hanya wajib mengetikkan kronologi pada <i>field Defect Cause / Chronology</i> (tanpa <i>field</i> unggah foto). Penyimpanan & Eksekusi 5. User mengklik tombol "<i>Save Report</i>" pada <i>Modal</i>. 6. Sistem memvalidasi kelengkapan form, menutup <i>Modal</i>, dan menyimpan data laporan sementara ke antrean <i>Return</i>. 7. User mengklik tombol "<i>Confirm Return</i>" pada halaman utama. 8. Sistem memproses pengembalian sekaligus menyimpan rincian <i>Defect Report</i> ke tabel <i>COMPLAINTS</i> dengan status '<i>waiting response</i>'. 9. Sistem menampilkan pesan "<i>Return Successful!</i>" (menggabungkan kesuksesan proses <i>Return</i> dan pengajuan <i>Complaint</i>).
Alternative Flow	Validasi <i>Modal Defect Report (Damaged)</i> 3a. Jika User memilih <i>Damaged</i> tetapi tidak mengunggah foto: - Sistem memblokir instruksi <i>Save Report</i> dan menampilkan peringatan: "<i>Please upload at least 1 evidence photo</i>". 3b. Jika User memilih <i>Damaged</i> tetapi mengosongkan kronologi: - Sistem memblokir instruksi <i>Save Report</i> dan menampilkan peringatan: "<i>Defect Cause / Chronology is required</i>". Validasi <i>Modal Defect Report (Lost)</i> 4a. Jika User memilih <i>Lost</i> tetapi mengosongkan kronologi: - Sistem memblokir instruksi <i>Save Report</i> dan menampilkan peringatan: "<i>Defect Cause / Chronology is required</i>".

Melihat *History* Peminjaman memungkinkan Admin dan *User* untuk mengakses riwayat peminjaman *Equipment* dengan detail informasi lengkap untuk keperluan *monitoring* data.

Tabel 3. 15 Skenario *Use case* Melihat *History*

Aspek	Deskripsi
Nomor	UC-13
Nama	Melihat <i>History</i> Peminjaman
Deskripsi	Admin dan <i>User</i> dapat melihat riwayat peminjaman
<i>Primary Actor</i>	Admin dan <i>User</i>
<i>Preconditions</i>	Sudah <i>login</i>
<i>Postconditions</i>	Sistem menampilkan data <i>History</i>
<i>Main Success Scenario</i>	<i>User</i>: 1. <i>User</i> klik menu <i>History</i> 2. Sistem tampilkan tabel "<i>History</i>" 3. Sistem tampilkan <i>History</i> peminjaman <i>User</i> tersebut saja 4. Sistem tampilkan <i>pagination</i> (<i>Prev</i>, 1, <i>Next</i>) Admin: 1. Admin klik menu <i>History</i> 2. Sistem tampilkan halaman "<i>Equipment History</i>" 3. Sistem tampilkan tabel <i>Equipment</i> 4. Admin klik tombol "Detail" pada tabel <i>action</i> 5. Sistem tampilkan halaman "<i>EQUIPMENT DETAIL HISTORY</i>" 6. Sistem tampilkan informasi <i>Equipment</i> 7. Sistem tampilkan <i>search box</i> dan <i>date filter</i> 8. Sistem tampilkan tabel detail <i>Equipment History</i> 9. Admin bisa <i>search</i> atau filter berdasarkan tanggal 10. Sistem tampilkan tombol "<i>Back to List</i>" untuk kembali ke list <i>Equipment</i>

Aspek	Deskripsi
<i>Alternative Flow</i>	2a. Jika belum ada <i>History (User)</i> : - Sistem tampilkan " <i>No borrowing History available</i> " 3a. Jika <i>Equipment</i> belum pernah dipinjam (Admin): - Sistem tampilkan " <i>No borrowing History for this Equipment</i> "

Mencatat Transaksi *Inventory In/Out* adalah proses pencatatan barang persediaan yang masuk (*In*) atau keluar (*Out*) untuk menjaga akurasi stok dan *monitoring* penggunaan barang secara sistematis.

Tabel 3. 16 Skenario *Use case* Mencatat Transaksi *Inventory In/Out*

Aspek	Deskripsi
Nomor	UC-14
Nama	Mencatat Transaksi <i>Inventory In/Out</i>
Deskripsi	Admin mencatat barang persediaan yang masuk atau keluar
<i>Primary Actor</i>	Admin
<i>Preconditions</i>	Admin sudah <i>login</i>
<i>Postconditions</i>	Transaksi tercatat dan <i>stock</i> ter-update
<i>Main Success Scenario</i>	1. Admin klik menu " <i>Asset Data</i> " → " <i>Inventory Data</i> " 2. Admin klik tombol " <i>In/Out Stock</i> " 3. Sistem tampilkan halaman log transaksi 4. Admin klik tombol untuk menambah transaksi baru 5. Sistem tampilkan form: <i>Item Code</i> , <i>Item Name</i> , <i>Quantity</i> , <i>Date</i> , <i>Remark</i> , <i>Status (In/Out)</i> 6. Admin isi form dan pilih status 7. (Opsional) Admin mengunggah dokumen bukti transaksi dalam format PDF

Aspek	Deskripsi
	8. Admin klik <i>"Save"</i> 9. Sistem validasi 10. Jika Status = <i>"In"</i> : Tambah <i>stock</i> . Jika Status = <i>"Out"</i> : Kurangi <i>stock</i> 11. Sistem <i>update</i> tabel <i>Inventory</i> dan simpan log transaksi 12. Sistem tampilkan pesan: <i>"Stock Successfully Updated"</i>
<i>Alternative Flow</i>	9a. Jika <i>Out</i> melebihi <i>stock</i> : - Sistem tampilkan pesan: <i>"Stock not sufficient. Remaining: [jumlah sisa stok]"</i> .

Memberikan respon *Complaint* adalah fitur bagi Admin untuk menanggapi keluhan *User* terkait *Equipment* rusak dengan memberikan solusi perbaikan, penggantian, atau tindakan lainnya.

Tabel 3. 17 Skenario *Use case* Memberikan Respon *Complaint*

Aspek	Deskripsi
Nomor	UC-15
Nama	Memberikan Respon <i>Complaint</i>
Deskripsi	Admin merespon <i>Complaint</i> dari <i>User</i>
<i>Primary Actor</i>	Admin
<i>Preconditions</i>	- Admin sudah <i>login</i> - Ada <i>Complaint</i> dengan status <i>"waiting response"</i>
<i>Postconditions</i>	- <i>Complaint</i> terupdate dengan <i>response</i> Admin - Status berubah jadi <i>"Repair/Replaced/Disposal"</i>
<i>Main Success Scenario</i>	1. Admin klik menu <i>Complaint</i> 2. Sistem tampilkan halaman <i>"Complaints"</i>

Aspek	Deskripsi
	3. Sistem tampilkan tabel <i>Complaint</i> 4. Sistem tampilkan <i>Complaints</i> dengan berbagai status 5. Admin klik tombol <i>Action</i> (icon/button) pada <i>Complaint</i> yang ingin direspon 6. Sistem tampilkan modal " <i>Admin Response</i> " 7. Admin isi form 8. Admin klik tombol " <i>Save</i> " 9. Sistem <i>update</i> table <i>COMPLAINTS</i> 10. Sistem <i>close</i> modal 11. Sistem <i>update</i> tabel <i>Complaints</i> 12. Sistem tampilkan " <i>Response Saved successfully</i> " 13. <i>User</i> bisa lihat <i>response</i> di kolom " <i>Admin Response</i> " pada tabel <i>Complaint</i>
<i>Alternative Flow</i>	8a. Jika <i>field</i> wajib kosong: - Sistem tampilkan " <i>All required fields must be filled</i> " 8b. Admin klik <i>Cancel</i> : - Sistem <i>close</i> modal tanpa menyimpan

Mengunduh Laporan Aset memungkinkan Admin untuk *export* data aset dalam format PDF untuk keperluan dokumentasi, audit, atau presentasi kepada manajemen.

Tabel 3. 18 Skenario *Use case* Mengunduh Laporan Aset

Aspek	Deskripsi
Nomor	UC-16
Nama	Mengunduh Laporan PDF (<i>Equipment & Defect Report</i>)
Deskripsi	Admin dapat mengunduh laporan daftar <i>Equipment</i> dan laporan detail kerusakan (<i>Defect Report</i>) dalam format PDF.

Aspek	Deskripsi
Primary Actor	Admin
Preconditions	Admin sudah <i>login</i>
Postconditions	File PDF laporan berhasil diunduh ke perangkat Admin.
Main Success Scenario	Laporan <i>Equipment</i>: 1. Admin membuka menu <i>Asset Data > Equipment</i> 2. Sistem menampilkan tabel data <i>Equipment</i> dan tombol "<i>Export PDF</i>" 3. Admin mengklik tombol "<i>Export PDF</i>" 4. Sistem memproses data dan otomatis mengunduh file PDF daftar <i>Equipment</i> Laporan Kerusakan (<i>Defect Report</i>): 1. Admin membuka menu <i>Complaint</i> 2. Sistem menampilkan tabel komplain yang sudah direspon (<i>Resolved</i>) 3. Admin mengklik ikon PDF pada baris komplain yang diinginkan 4. Sistem memproses data yang mencakup rincian komplain, kronologi kerusakan, serta respon Admin 5. Sistem otomatis mengunduh file PDF "<i>Defect Report</i>"
Alternative Flow	3a. Jika data tidak tersedia atau gagal diproses: - Sistem menampilkan pesan: "<i>Data not found</i>"

Logout adalah fitur untuk mengakhiri sesi pengguna dan keluar dari sistem dengan aman untuk menjaga keamanan data dan mencegah akses tidak sah.

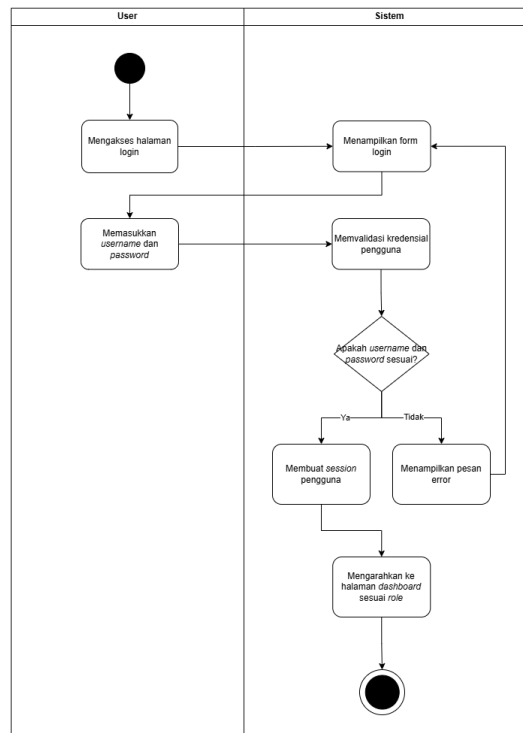
Tabel 3. 19 Skenario *Use case Logout*

Aspek	Deskripsi
Nomor	UC-17
Nama	<i>Logout</i> dari Sistem

Aspek	Deskripsi
Deskripsi	Pengguna keluar dari sistem
Primary Actor	Admin dan <i>User</i>
Preconditions	Pengguna sudah <i>login</i>
Postconditions	Session dihapus, pengguna keluar dari sistem
Main Success Scenario	1. Pengguna klik tombol "<i>Logout</i>" (di <i>header/profile menu</i>) 2. Sistem hapus <i>session</i> pengguna 3. Sistem <i>redirect</i> ke halaman <i>Login</i>
Alternative Flow	-

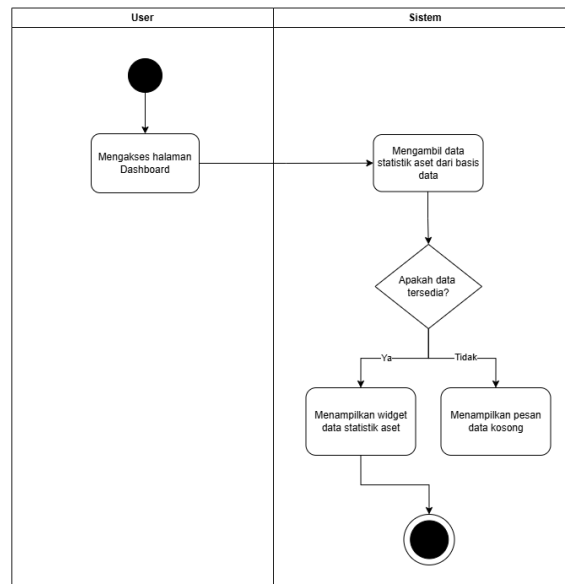
3.2.5 Activity Diagram

Activity Diagram menggambarkan alur aktivitas yang terjadi dalam sistem, menunjukkan urutan aktivitas satu ke aktivitas lainnya. *Activity* diagram digunakan untuk memodelkan proses bisnis dan alur kerja sistem secara detail, mulai dari titik awal hingga titik akhir dari suatu aktivitas. Diagram ini membantu memahami bagaimana sistem berinteraksi dengan pengguna dan bagaimana data mengalir di dalam sistem. Berikut adalah penjelasan *activity* diagram untuk setiap fitur utama sistem :



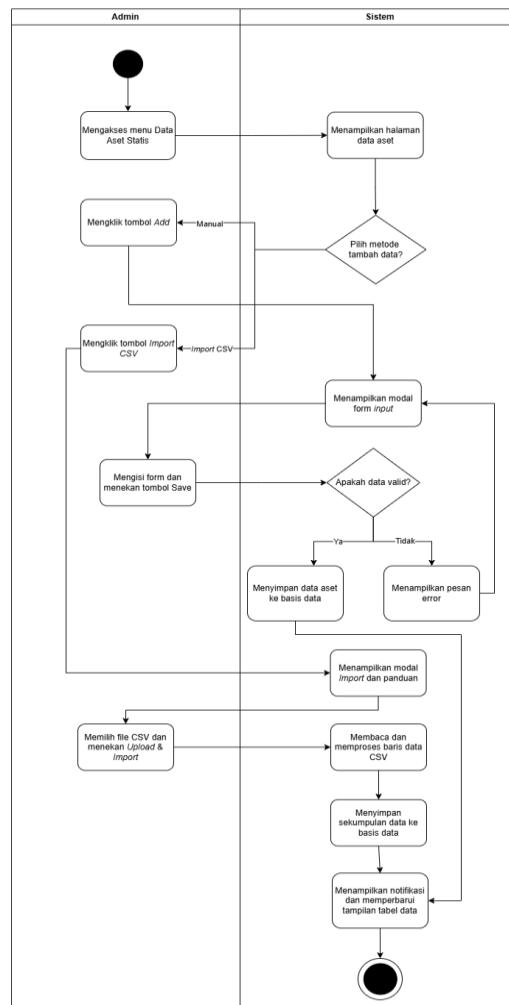
Gambar 3.3 Activity Diagram Login

Gambar 3.3 merupakan *Activity Diagram Login* yang menggambarkan alur proses autentikasi pengguna. Alur dimulai saat pengguna mengakses halaman *login*, kemudian sistem menampilkan formulir autentikasi. Pengguna memasukkan *Username* dan *password*, lalu sistem melakukan validasi kredensial terhadap basis data. Jika data tidak sesuai, sistem akan menampilkan pesan *error* dan mengembalikan pengguna ke formulir *login*. Sebaliknya, jika data valid, sistem akan membuat *session* dan mengarahkan pengguna ke halaman *dashboard* sesuai dengan hak akses (*role*) masing-masing.



Gambar 3. 4 Activity Diagram Melihat *Dashboard*

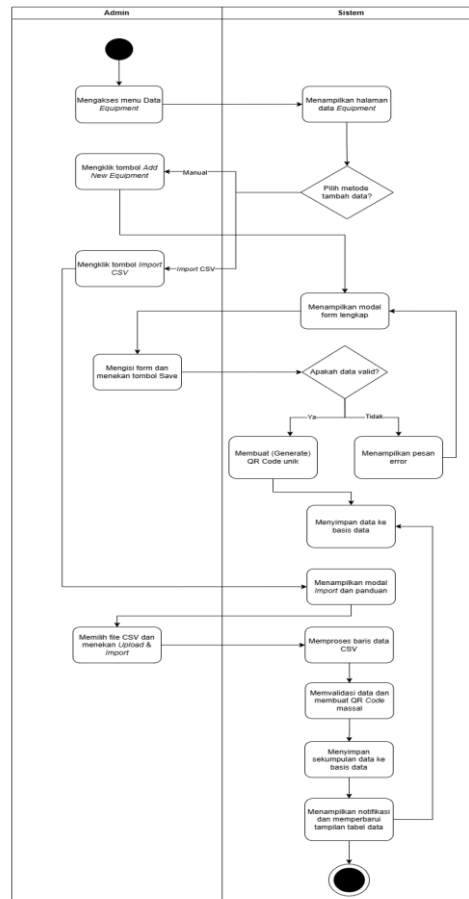
Gambar 3.4 merupakan *Activity* Diagram Melihat *Dashboard* yang menjelaskan alur sistem dalam menampilkan ringkasan data untuk Admin dan *User*. Proses ini dimulai saat pengguna mengakses halaman *dashboard*. Selanjutnya, sistem akan mengambil data riwayat transaksi dan ketersediaan aset dari basis data. Jika data tersebut tersedia, sistem akan menampilkannya dalam bentuk angka statistik pada layar pengguna. Namun, jika basis data masih kosong atau belum ada transaksi yang tercatat, sistem akan memberikan pesan informasi bahwa data belum tersedia.



Gambar 3.5 Activity Diagram Mengelola Master Data Aset Statis

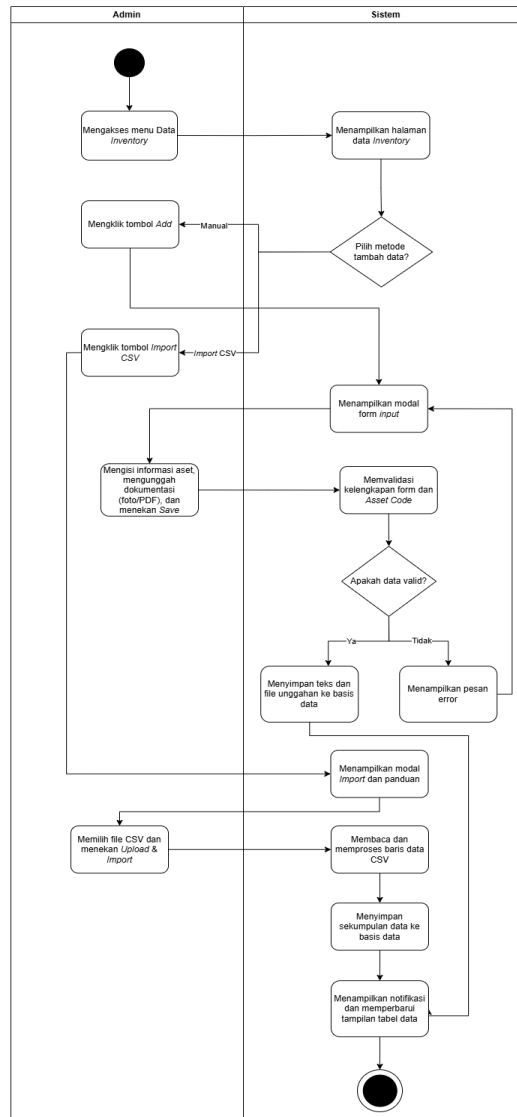
Gambar 3.5 merupakan *Activity Diagram* Mengelola Master Data Aset Statis yang menunjukkan alur penambahan data melalui dua metode: manual dan *Import CSV*. Proses diawali ketika Admin mengakses menu data aset statis. Pada metode manual, Admin mengisi formulir yang disediakan sistem. Sistem kemudian memvalidasi data tersebut; jika gagal, sistem memunculkan pesan *error*, namun jika valid, data langsung disimpan ke basis data. Pada metode *Import CSV*, Admin mengunggah berkas format CSV dan sistem akan memproses serta menyimpan seluruh baris data secara massal. Kedua alur metode ini diakhiri dengan langkah

sistem yang menampilkan notifikasi keberhasilan dan memperbarui antarmuka tabel data secara otomatis.



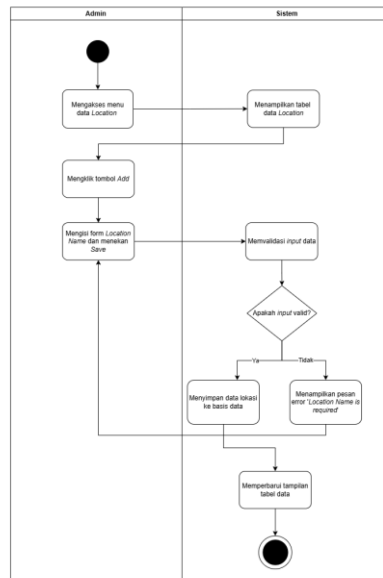
Gambar 3. 6 Activity Diagram Mengelola Data *Equipment*

Gambar 3.6 menunjukkan *Activity Diagram* penambahan data *Equipment* melalui metode manual dan *Import CSV*. Pada metode manual, sistem memvalidasi *Asset Number* dari formulir isian dan otomatis membuat *QR Code* jika valid. Pada metode *Import CSV*, sistem memvalidasi dan membuat *QR Code* secara massal dari berkas unggahan. Kedua metode tersebut diakhiri dengan penyimpanan data, notifikasi keberhasilan, dan pembaruan tabel otomatis.



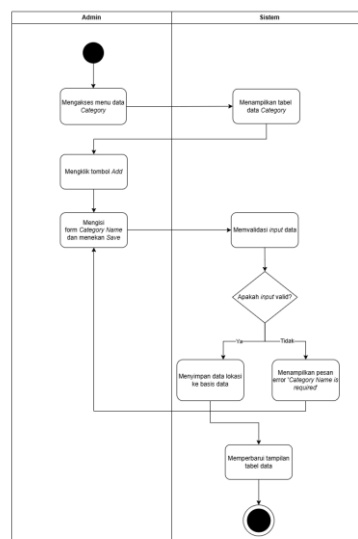
Gambar 3. 7 Activity Diagram Inventory

Gambar 3.7 menunjukkan *Activity Diagram* penambahan data *Inventory* melalui metode manual dan *Import CSV*. Pada metode manual, Admin mengisi formulir dan mengunggah dokumen pendukung, lalu sistem memvalidasi *Asset Code* sebelum menyimpan data. Pada metode *Import CSV*, sistem langsung memproses dan menyimpan data dari berkas secara massal. Kedua alur diakhiri dengan notifikasi sukses dan pembaruan tabel otomatis.



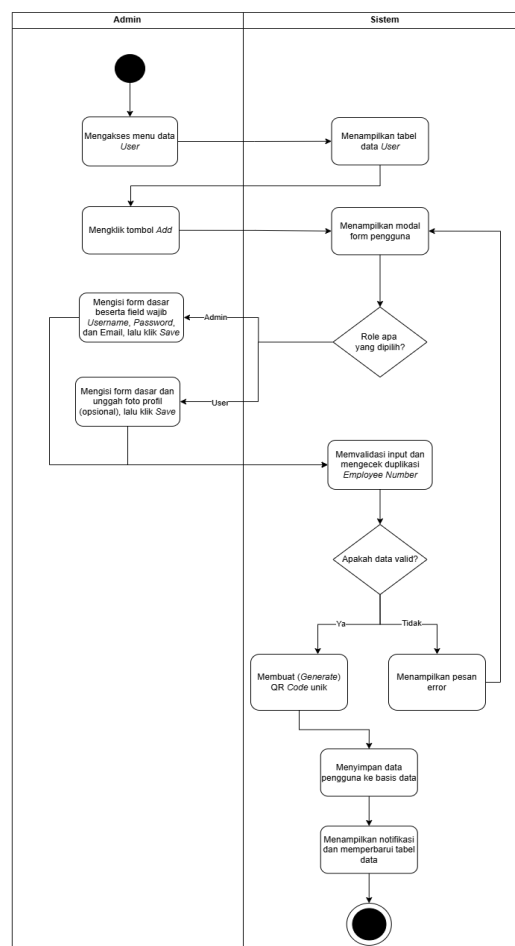
Gambar 3.8 Activity Diagram Mengelola Location

Gambar 3.8 merupakan *Activity Diagram Mengelola Location Data* yang menggambarkan alur pengelolaan data lokasi. Saat Admin mengklik menu *Location Data*, sistem menampilkan tabel data *Location*. Dari halaman ini, Admin dapat menambah, mengubah, atau menghapus data. Setiap perubahan akan divalidasi oleh sistem; jika tidak sesuai akan muncul pesan *error*, namun jika valid, sistem akan menyimpannya ke basis data dan memperbarui tabel secara otomatis.



Gambar 3.9 Activity Diagram Mengelola Category

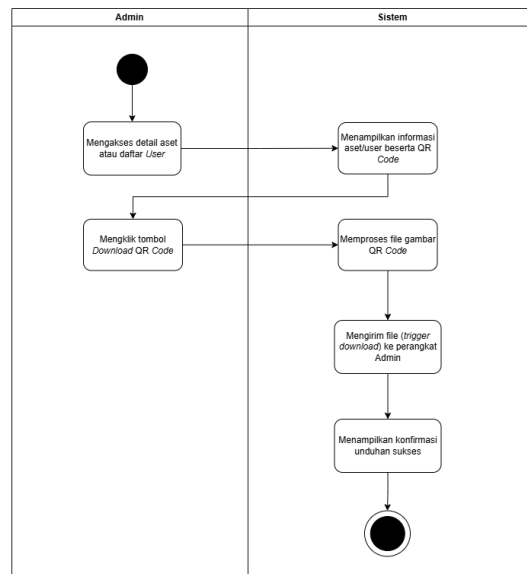
Gambar 3.9 merupakan *Activity Diagram Mengelola Category Data*. Saat Admin mengakses menu tersebut, sistem menampilkan daftar kategori yang ada. Untuk menambah data, Admin wajib mengisi formulir nama kategori. Sistem akan memvalidasi isian tersebut; menampilkan pesan *error* jika kosong, atau menyimpannya langsung ke basis data jika valid, lalu diakhiri dengan pembaruan antarmuka tabel.



Gambar 3. 10 Activity Diagram User Data

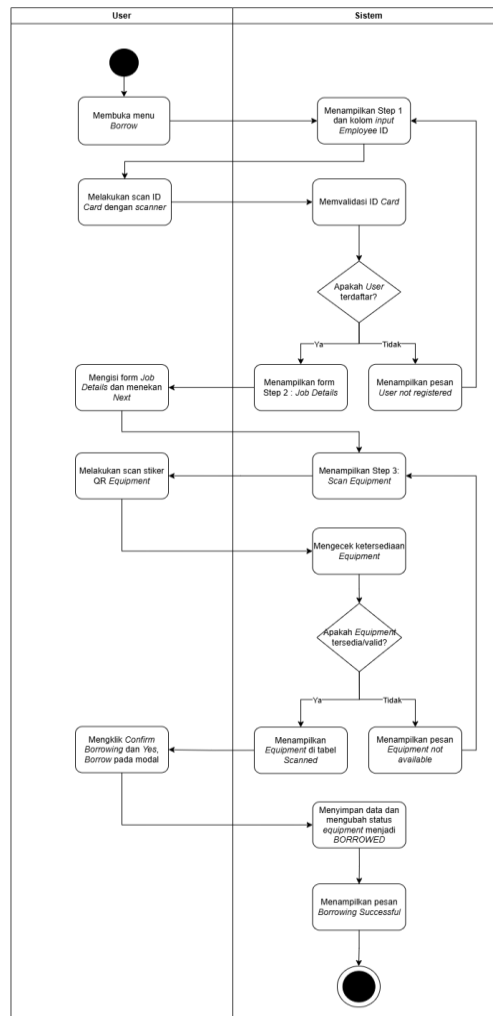
Gambar 3.10 merupakan *Activity Diagram Mengelola User Data*. Saat Admin mengakses menu dan mengisi formulir pengguna, sistem akan menyesuaikan kolom isian berdasarkan *role* yang dipilih. Setelah tombol *Save* ditekan, sistem memvalidasi data dan secara otomatis membuat *QR Code* jika input

valid. Data pengguna beserta *QR Code* tersebut kemudian disimpan ke basis data dan ditampilkan secara langsung pada tabel pengguna.



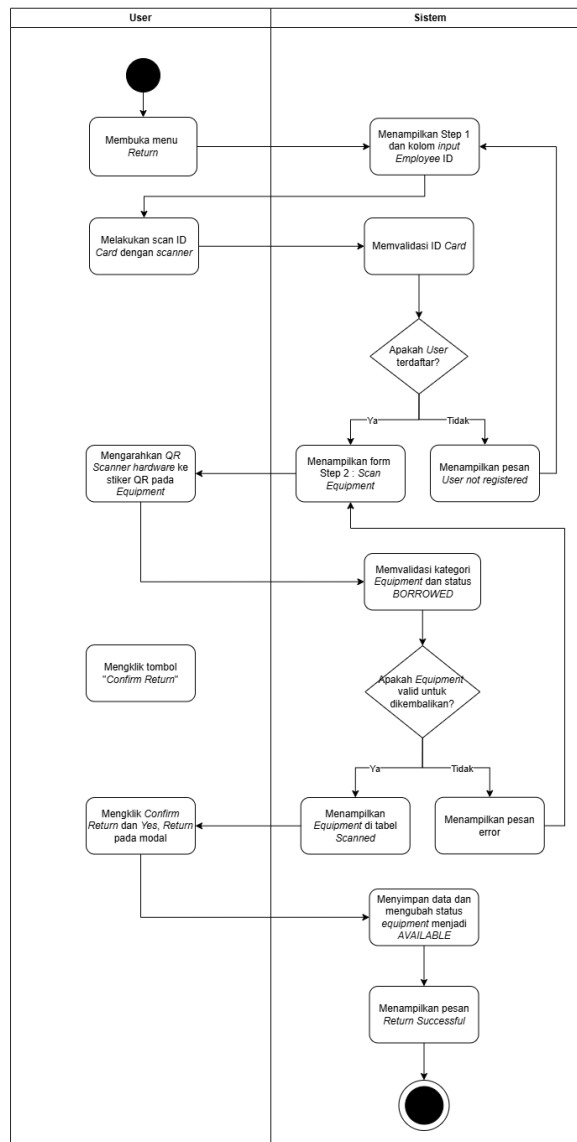
Gambar 3. 11 Activity Diagram Generate QR Code

Gambar 3.11 merupakan *Activity Diagram* fitur *Generate* dan *Download QR Code*. Setelah sistem menampilkan detail aset atau pengguna beserta *QR Code* yang sudah dibuat, Admin dapat melakukan unduhan berkas secara langsung. Saat tombol *download* diklik, sistem akan memproses berkas gambar tersebut dan mengirimkannya ke perangkat Admin. Proses diakhiri dengan munculnya notifikasi bahwa berkas telah berhasil diunduh.



Gambar 3. 12 Activity Diagram Borrowing Equipment

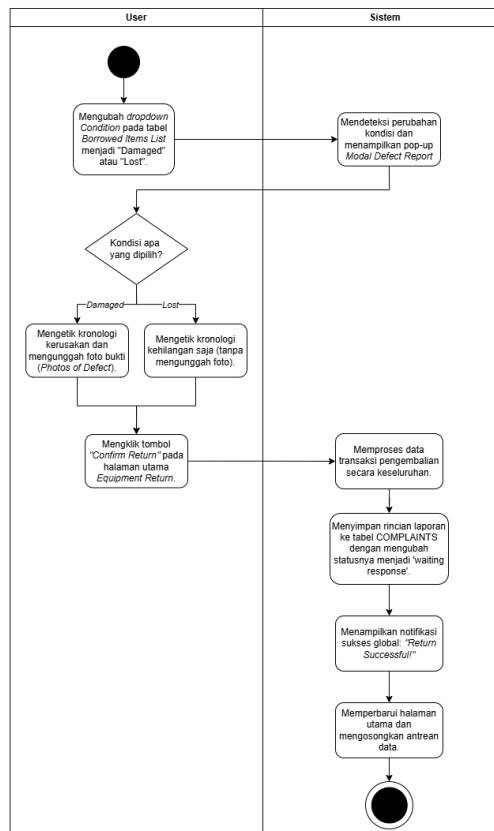
Gambar 3.12 merupakan *Activity Diagram* proses meminjam *Equipment* menggunakan *QR Scanner hardware*. Alur ini terbagi menjadi tiga tahapan utama. Pada tahap pertama, *User* memindai *ID Card* untuk divalidasi oleh sistem. Pada tahap kedua, *User* mengisi rincian informasi pekerjaan (*Job Details*). Pada tahap ketiga, *User* memindai *QR Code* pada peralatan. Pada setiap proses pemindaian, sistem akan melakukan validasi; jika data tidak terdaftar atau tidak tersedia, sistem menampilkan pesan *error*. Namun, jika seluruh tahapan valid dan *User* melakukan konfirmasi, sistem akan mencatat transaksi peminjaman, mengubah status peralatan menjadi '*BORROWED*', dan menampilkan notifikasi keberhasilan.



Gambar 3. 13 Activity Diagram Return Equipment

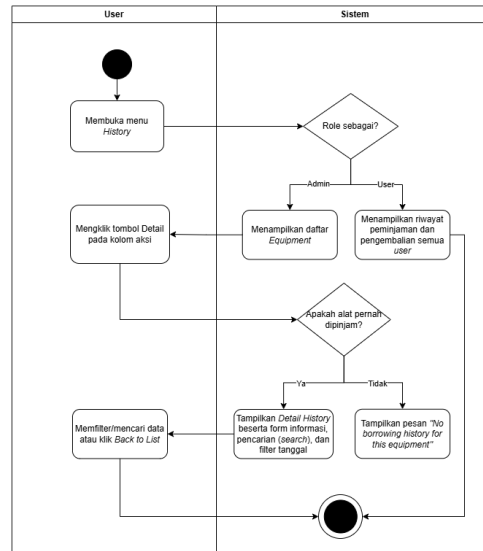
Gambar 3.13 merupakan *Activity Diagram* proses pengembalian (*Return Equipment*) menggunakan pemindai perangkat keras. Alur ini dibagi menjadi dua tahapan utama. Pada tahap pertama, *User* memindai *ID Card* untuk verifikasi sistem. Pada tahap kedua, *User* memindai *QR Code* pada peralatan. Sistem akan memvalidasi bahwa aset tersebut adalah *Equipment* yang sedang berstatus dipinjam (*BORROWED*). Jika terdapat ketidaksesuaian, sistem memunculkan pesan *error*. Jika seluruh tahapan valid dan *User* menyetujui konfirmasi pengembalian, sistem

akan memproses transaksi, memperbarui status peralatan menjadi tersedia (*AVAILABLE*), dan memunculkan notifikasi berhasil.



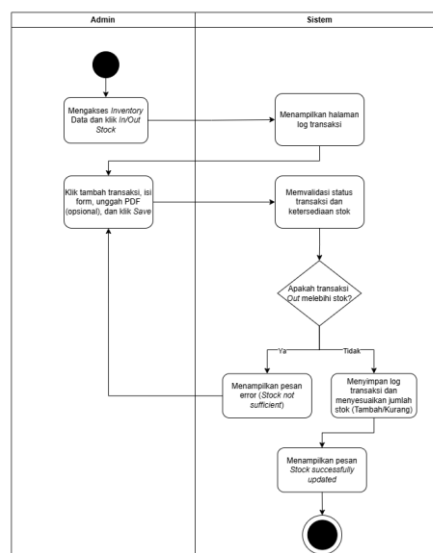
Gambar 3. 14 Activity Diagram User Complaint

Gambar 3.14 merupakan *Activity Diagram* pengajuan *Complaint* oleh *User*. *User* wajib melengkapi formulir dengan memilih *Equipment*, mengisi deskripsi, dan mengunggah foto bukti. Jika validasi kelengkapan gagal, sistem memunculkan pesan *error*. Jika valid, sistem langsung menyimpan laporan tersebut dengan status '*waiting response*' beserta notifikasi keberhasilan.



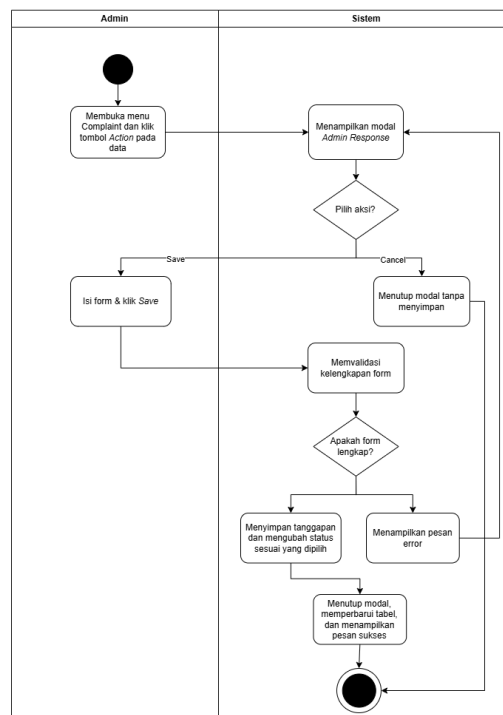
Gambar 3. 15 Activity Diagram History

Gambar 3.15 merupakan *Activity Diagram* Melihat *History* Peminjaman. Jika diakses oleh *User*, sistem akan menampilkan tabel riwayat peminjaman dari keseluruhan pengguna. Jika diakses oleh Admin, sistem menampilkan daftar *Equipment* terlebih dahulu. Admin kemudian dapat memilih detail untuk melihat riwayat spesifik suatu alat yang dilengkapi fitur pencarian dan filter tanggal. Sistem otomatis memunculkan pesan informatif jika riwayat peminjaman masih kosong pada kedua alur tersebut.



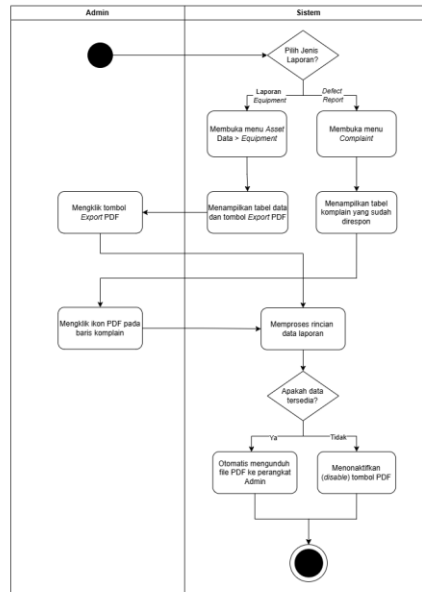
Gambar 3. 16 Activity Diagram Inventory In/Out

Gambar 3.16 merupakan *Activity Diagram* Mencatat Transaksi *Inventory In/Out*. Admin mengisi formulir barang masuk atau keluar (*In/Out*) beserta unggahan bukti dokumen PDF (opsional). Sistem kemudian memvalidasi data; jika transaksi *Out* melebihi jumlah stok yang ada, sistem memunculkan pesan *error*. Jika data valid, sistem langsung menyesuaikan jumlah stok (*update*), menyimpan log transaksi, dan menampilkan pesan keberhasilan.



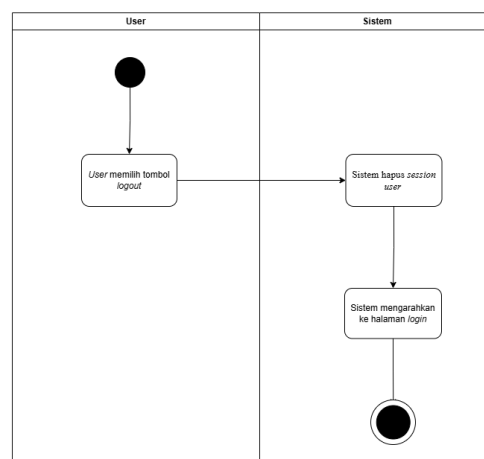
Gambar 3. 17 Activity Diagram Admin Merespon Complaint

Gambar 3.17 merupakan *Activity Diagram* Admin merespon *Complaint*. Admin mengisi form tanggapan dan memilih status penyelesaian yang baru (misal: *Repair* atau *Replaced* atau *Disposal*). Jika divalidasi berhasil, sistem akan menyimpan respon, memperbarui status pengaduan sesuai pilihan tersebut, dan memunculkan notifikasi sukses. Proses juga dapat dibatalkan tanpa menyimpan perubahan.



Gambar 3. 18 Activity Diagram Mengunduh Laporan PDF

Gambar 3.18 merupakan *Activity Diagram* mengunduh laporan PDF untuk *Equipment* dan *Defect Report*. Saat Admin mengakses menu laporan, sistem akan mengecek ketersediaan data terlebih dahulu. Jika data kosong, sistem akan menampilkan pesan '*Data not found*' dan tombol unduhan tidak dapat ditekan (*disabled*). Namun, jika data tersedia, tombol akan aktif; saat diklik, sistem langsung memproses dan mengunduh berkas PDF ke perangkat Admin.

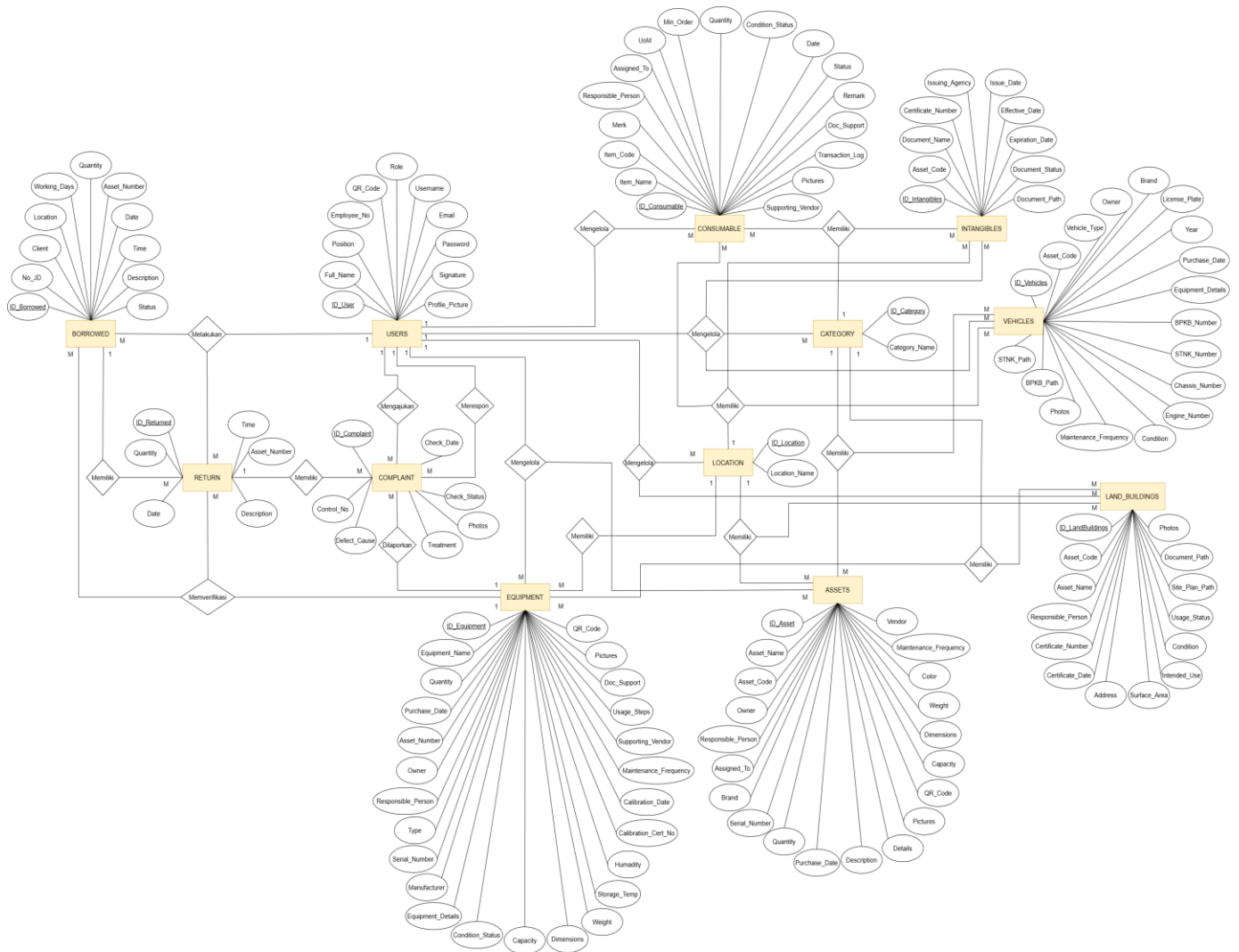


Gambar 3. 19 Activity Diagram Logout

Gambar 3.19 merupakan *Activity Diagram Logout* yang menggambarkan proses keluar dari sistem. *User* memilih tombol *logout*, sistem menghapus *session User*, kemudian sistem mengarahkan ke halaman *login*.

3.2.6 ER Diagram

Desain *database* adalah proses perencanaan dan pengorganisasian struktur *database* untuk memastikan bahwa data yang disimpan dapat diakses, dikelola, dan diperbarui. Salah satu alat yang sering digunakan dalam proses ini adalah *Entity Relationship Diagram* (ERD).



Gambar 3. 20 ER Diagram

Gambar ERD di atas menunjukkan rancangan struktur basis data pada sistem manajemen aset. Data pengguna di dalam sistem (entitas *USER*) dibagi menjadi dua peran utama, yaitu *ADMIN* dan *INSPECTOR*. Entitas *USER* menyimpan informasi dasar seperti nama lengkap, jabatan, nomor pegawai, dan foto profil. Secara khusus, sub-entitas *ADMIN* menyimpan data untuk akses masuk aplikasi (*login*) seperti *Username* dan *password*, sedangkan sub-entitas *INSPECTOR* memiliki data tanda tangan (*signature*) untuk keperluan pengesahan dokumen di lapangan.

Sistem ini mengelompokkan data aset ke dalam enam tabel yang berbeda sesuai jenisnya, yaitu *ASSETS* (peralatan kantor), *VEHICLES* (kendaraan), *LAND_BUILDINGS* (tanah dan bangunan), *INTANGIBLES* (aset tak berwujud), *CONSUMABLE* (barang habis pakai), dan *EQUIPMENT* (peralatan kerja). Semua tabel aset ini terhubung dengan data master *CATEGORY* dan *LOCATION*. Hubungan ini bersifat Satu-ke-Banyak (1:M), yang berarti satu kategori atau satu titik lokasi dapat menampung banyak unit aset.

Khusus untuk data kendaraan (*VEHICLES*) serta tanah dan bangunan (*LAND_BUILDINGS*), terdapat hubungan tanggung jawab secara langsung dengan pengguna. Satu *ADMIN* atau *INSPECTOR* dapat ditugaskan untuk bertanggung jawab atas beberapa aset pada kedua tabel tersebut. Aset-aset ini sifatnya hanya didata sebagai inventaris dan tidak masuk ke dalam alur peminjaman.

Di sisi lain, entitas *EQUIPMENT* (peralatan kerja) menjadi pusat kegiatan operasional karena alat ini digunakan bergantian di lapangan. Oleh karena itu, tabel peralatan ini terhubung langsung dengan tiga entitas transaksi utama, dengan alur sebagai berikut:

- Peminjaman (*BORROWED*): Mencatat data alat yang dipinjam beserta informasi pendukungnya. Hubungannya bersifat Banyak-ke-Satu (M:1) dengan *INSPECTOR* dan *EQUIPMENT*, artinya satu orang pengguna dapat meminjam banyak alat, dan satu alat bisa memiliki banyak riwayat peminjaman.
- Pengembalian (*RETURN*): Menyimpan rekam jejak alat yang telah selesai dipinjam. Hubungan antara peminjaman dan pengembalian bersifat Satu-

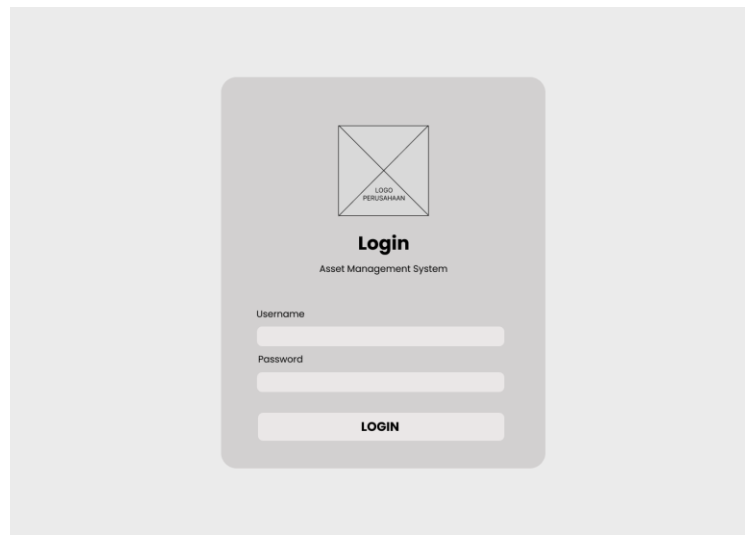
ke-Banyak (1:M). Hal ini dirancang agar pengguna dapat mengembalikan alat secara bertahap atau dicicil dari satu nomor peminjaman.

- Pelaporan (*COMPLAINT*): Menyimpan laporan jika terdapat alat yang rusak saat dikembalikan. Hubungan dari pengembalian ke pelaporan juga bersifat Satu-ke-Banyak (1:M), sehingga pengguna dapat merinci kerusakan dengan membuat beberapa tiket komplain yang berbeda dari satu kali proses pengembalian alat.

Secara keseluruhan, rancangan ERD ini menunjukkan bahwa basis data telah disusun dengan sangat teratur. Pemisahan yang jelas antara aset tetap, barang habis pakai, dan peralatan kerja membuat sistem mampu menyesuaikan diri dengan fungsi masing-masing barang. Hubungan antar tabel juga dirancang dengan baik untuk mencegah penumpukan data yang berulang, sehingga informasi selalu akurat mulai dari pengaturan master data hingga ke pencatatan transaksi harian di lapangan.

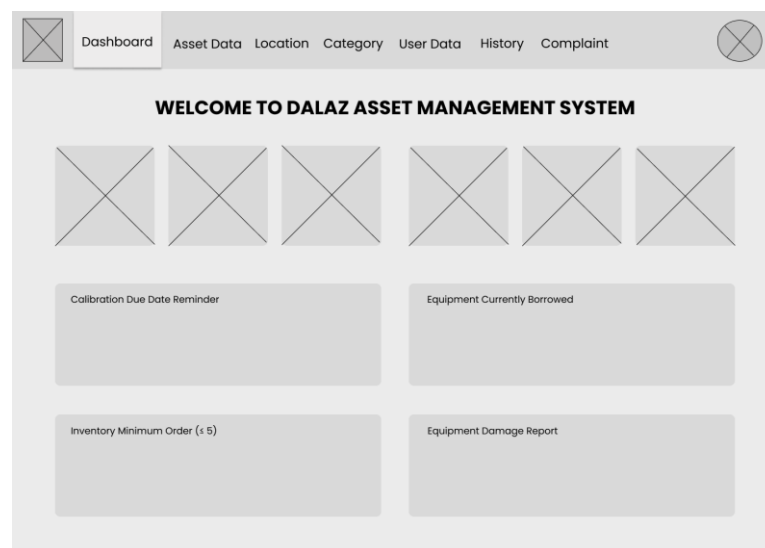
3.2.7 Perancangan Antarmuka Pengguna

Perancangan antarmuka pengguna (*User Interface*) dilakukan untuk memberikan pengalaman pengguna yang intuitif, mudah digunakan, dan efisien dalam mengoperasikan sistem manajemen aset berbasis *QR Code*. Antarmuka dirancang dengan mempertimbangkan kebutuhan Admin dan *User* dengan hak akses yang berbeda. Berikut adalah deskripsi antarmuka utama sistem :



Gambar 3. 21 Tampilan *Login*

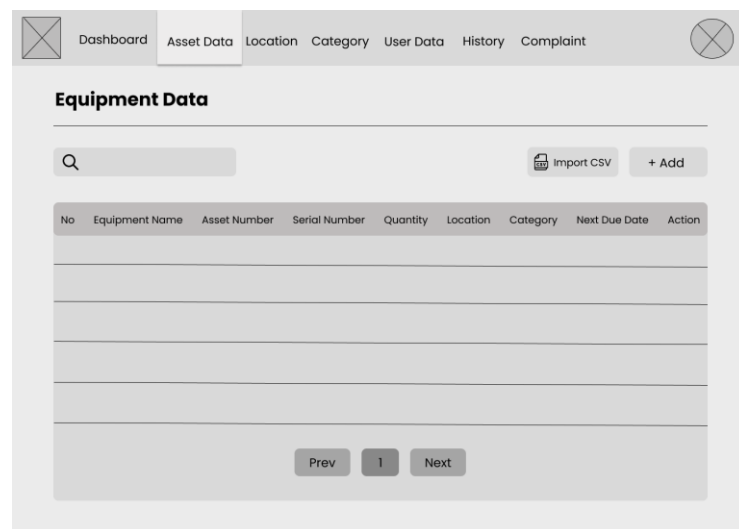
Gambar 3.21 merupakan halaman *Login*. Halaman ini menampilkan logo perusahaan, kolom isian untuk *Username* dan *Password*, serta tombol *Login* yang berfungsi untuk memproses autentikasi akses pengguna ke dalam sistem.



Gambar 3. 22 Tampilan *Dashboard* Admin

Gambar 3.22 merupakan halaman *Dashboard* Admin. Halaman ini memuat menu navigasi utama di bagian atas dan teks sambutan. Selain itu, terdapat empat kartu informasi utama yang memantau status operasional, yaitu pengingat jadwal kalibrasi (*Calibration Due Date Reminder*), peralatan yang sedang dipinjam

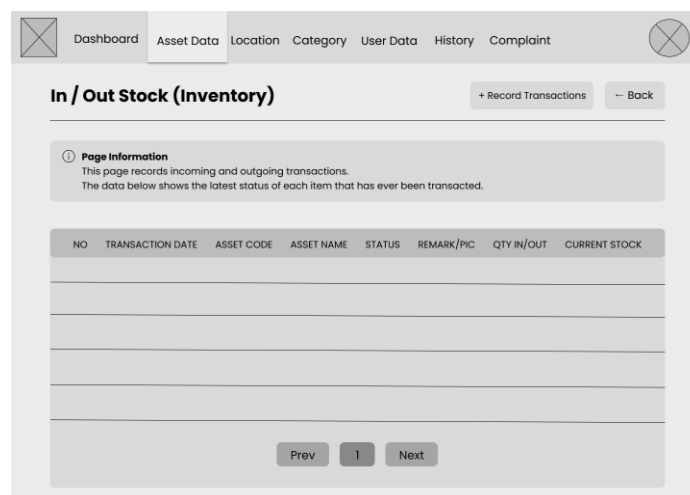
(*Equipment Currently Borrowed*), peringatan batas minimum stok (*Inventory Minimum Order*), laporan kerusakan alat (*Equipment Damage Report*), dan pengembalian yang terlambat (*Late Returning Borrowing*).



No	Equipment Name	Asset Number	Serial Number	Quantity	Location	Category	Next Due Date	Action

Gambar 3. 23 Tampilan Data *Asset company*

Gambar 3.23 merupakan halaman *Equipment Data*. Halaman ini menyediakan tabel komprehensif yang memuat daftar peralatan kerja. Untuk memudahkan pengelolaan, halaman ini dilengkapi dengan kolom pencarian (*Search*), tombol *Import CSV* untuk memasukkan data secara massal, serta tombol *+ Add* untuk menambahkan data alat baru secara manual.



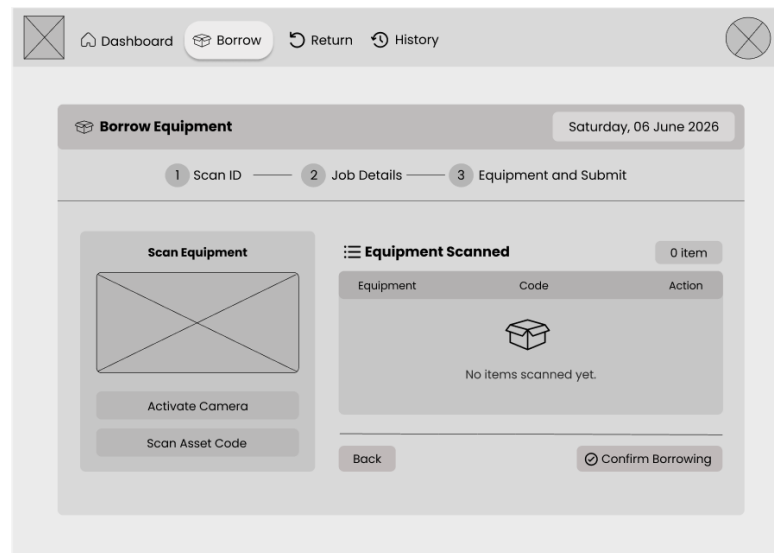
NO	TRANSACTION DATE	ASSET CODE	ASSET NAME	STATUS	REMARK/PIC	QTY IN/OUT	CURRENT STOCK

Gambar 3. 24 Tampilan *Inventory In/Out*

Gambar 3.24 merupakan halaman *In / Out Stock (Inventory)* yang berfungsi untuk merekam riwayat pergerakan barang habis pakai. Halaman ini menampilkan informasi petunjuk halaman, tombol + *Record Transactions* untuk menambah data keluar-masuk barang, serta tabel log transaksi yang memuat detail riwayat dan pembaruan jumlah stok saat ini.

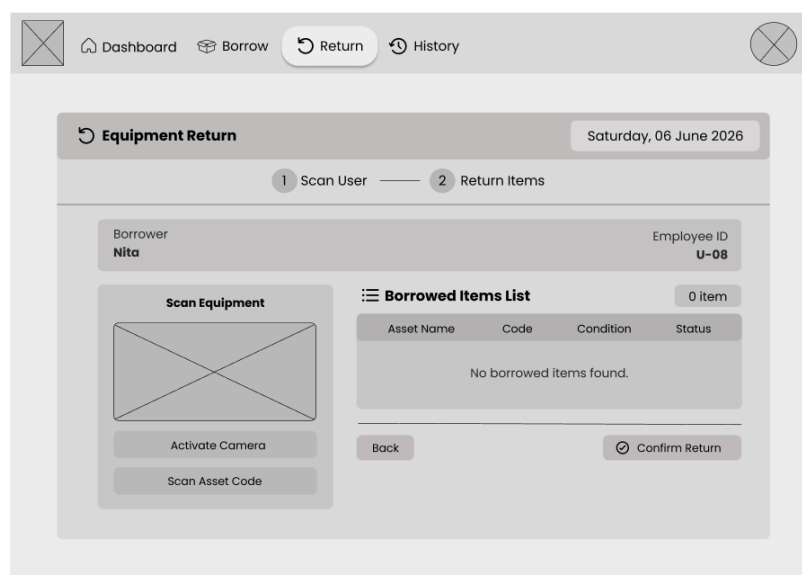
Gambar 3. 25 Tampilan *Location*

Gambar 3.25 merupakan halaman *Location Data* yang terbagi menjadi dua fitur utama. Bagian atas menampilkan tabel daftar lokasi fisik perusahaan beserta tombol + *Add* untuk menambah lokasi baru. Bagian bawah menyediakan fitur pencarian spesifik (*Search assets by location*) untuk melihat daftar aset yang berada pada titik lokasi tertentu.



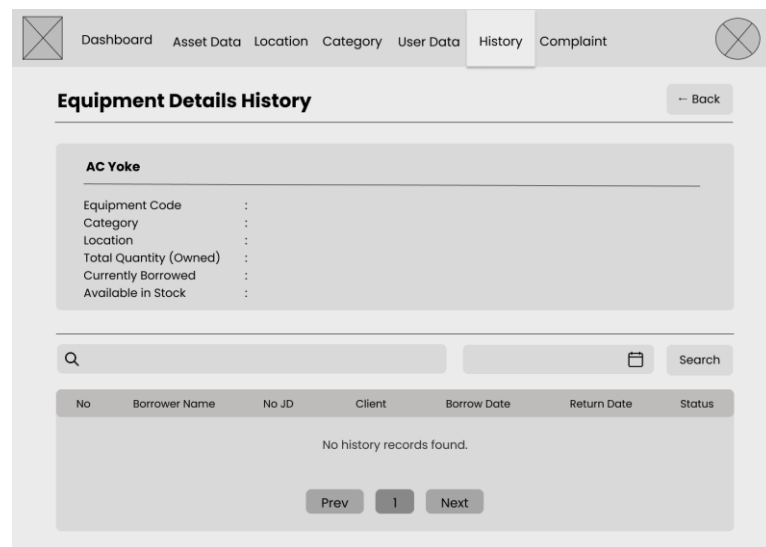
Gambar 3. 26 Tampilan *Equipment Borrowing*

Gambar 3.26 merupakan halaman *Borrow Equipment* untuk memproses transaksi peminjaman alat. Proses ini dibagi menjadi tiga tahapan alur: *Scan ID*, pengisian *Job Details*, dan penyelesaian pada tahap *Equipment and Submit*. Pada tahap penyelesaian, sistem menyediakan antarmuka kamera dan tombol *Scan Asset Code* untuk memindai alat, tabel daftar alat yang berhasil dipindai, serta tombol *Confirm Borrowing* untuk memvalidasi transaksi.



Gambar 3. 27 Tampilan *Equipment Returning*

Gambar 3.27 merupakan halaman *Equipment Return* untuk memproses pengembalian alat yang dipinjam. Alur pengembalian terdiri dari dua tahapan: *Scan User* dan *Return Items*. Halaman ini menampilkan kotak identitas peminjam (*Borrower*), area pemindaian *QR Code* alat, tabel daftar barang yang dikembalikan (*Borrowed Items List*) yang mencakup kolom kondisi alat, serta tombol *Confirm Return*.



Gambar 3. 28 Tampilan *History* Admin

Gambar 3.28 merupakan halaman *Equipment Details History* yang menampilkan rincian riwayat penggunaan dari satu alat yang spesifik. Halaman ini memuat kotak informasi detail alat (seperti total kuantitas dan ketersediaan stok), fitur penyaringan berdasarkan kata kunci dan tanggal, serta tabel yang berisi rekam jejak peminjaman dan pengembalian alat oleh pengguna.

BAB IV

IMPLEMENTASI DAN PENGUJIAN

2.1. Hasil Implementasi

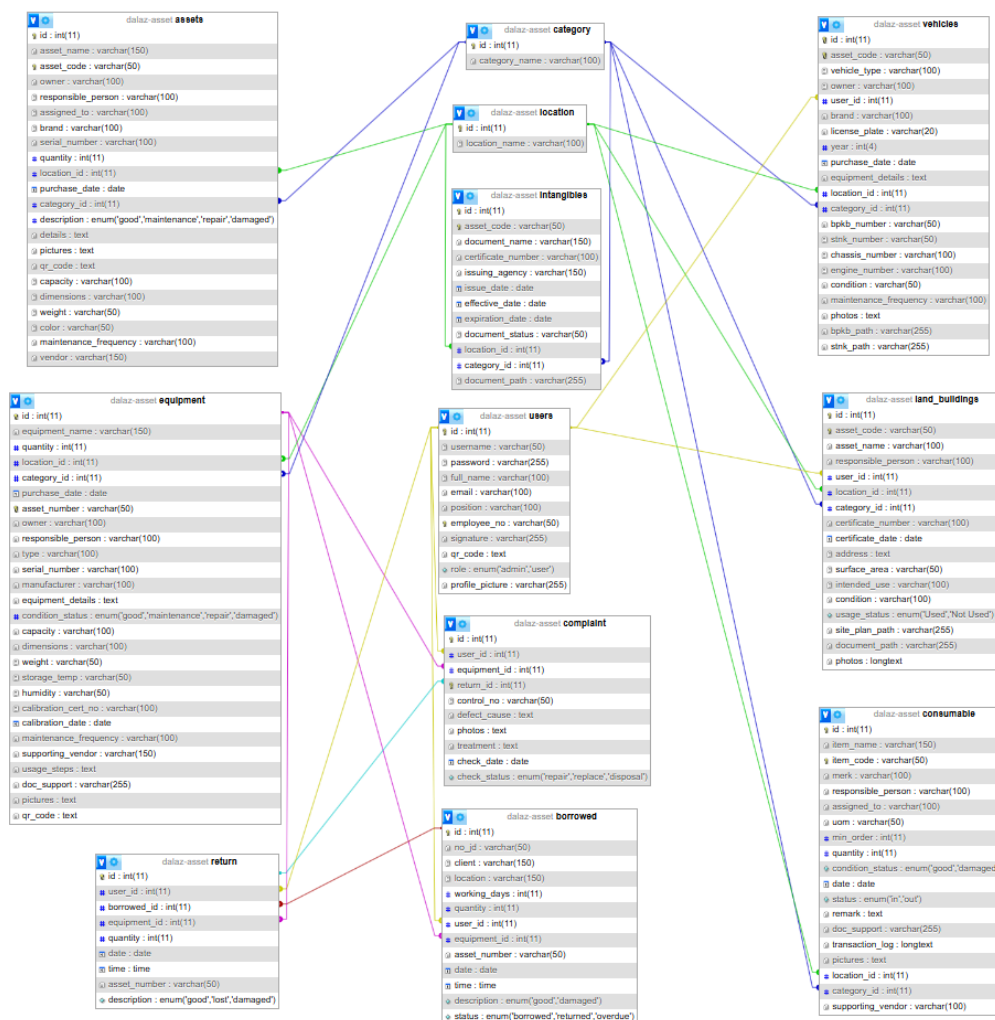
Implementasi pada proyek akhir ini dilakukan setelah tahap analisis kebutuhan dan perancangan sistem selesai dilaksanakan sesuai dengan metode pengembangan yang digunakan. Tahap implementasi proyek akhir dilaksanakan setelah seluruh analisis kebutuhan dan perancangan sistem selesai. Langkah pertama adalah membuat basis data menggunakan MySQL. Basis data tersebut mencakup data pengguna serta catatan untuk enam kategori master data aset (*Office Equipment, Equipment, Inventory, Vehicle, Land & Building, dan Intangible Assets*), serta log transaksi peminjaman, pengembalian, dan pelaporan kerusakan (*complaints*). Tabel-tabel dalam basis data dibuat sesuai dengan rancangan *Entity Relationship Diagram* (ERD). Hal ini bertujuan agar hubungan antar-data tersusun rapi, mencegah terjadinya duplikasi data, dan memastikan kinerja operasional sistem berjalan lancar.

Tahap implementasi selanjutnya dilakukan pada sisi *backend* menggunakan bahasa pemrograman PHP *Native* dengan menerapkan pola arsitektur *Model-View-Controller* (MVC). Logika *backend* yang dikembangkan mencakup proses autentikasi pengguna dengan enkripsi *bcrypt*, fungsi dasar pengelolaan data CRUD untuk pengelolaan data aset, serta pencatatan stok keluar-masuk (*In/Out*) pada inventaris. Pada tahap ini juga dilakukan pengintegrasian *library* pihak ketiga, yaitu *Endroid QR Code* untuk *generate* kode QR unik secara otomatis saat aset baru ditambahkan, serta *library TCPDF* untuk memproses dan mengeksport data menjadi dokumen laporan *Equipment* dan *Defect Report* dalam format PDF.

Pada sisi *frontend*, implementasi menggunakan HTML, CSS, dan *JavaScript* agar tampilannya responsif, interaktif, dan mudah digunakan oleh pihak Admin maupun *User*. Halaman-halaman utama seperti menu pengelolaan aset (termasuk fitur *Import* data CSV), dan tabel riwayat transaksi dirancang dengan tampilan yang intuitif. Formulir peminjaman (*Borrow*) dan pengembalian (*Return*) dirancang

secara khusus untuk terintegrasi dengan perangkat keras *QR Code Scanner*, di mana sistem dapat menerima input teks otomatis dari alat pemindai sehingga proses validasi aset di lapangan menjadi jauh lebih cepat. Sebagai langkah akhir, seluruh sistem yang telah selesai dikonfigurasi kemudian dijalankan menggunakan *web server* Apache dan MySQL melalui perangkat lunak XAMPP agar siap digunakan di lingkungan operasional internal PT. Dalaz Teknik Utama.

4.1.1 Implementasi Basis Data



Gambar 4. 1 Implementasi Basis Data

Gambar 4.1 menunjukkan implementasi skema relasi basis data yang digunakan pada sistem manajemen aset PT. Dalaz Teknik Utama. Basis data ini

terdiri dari dua belas tabel utama yang saling terhubung satu sama lain untuk mendukung pengelolaan aset perusahaan secara menyeluruh.

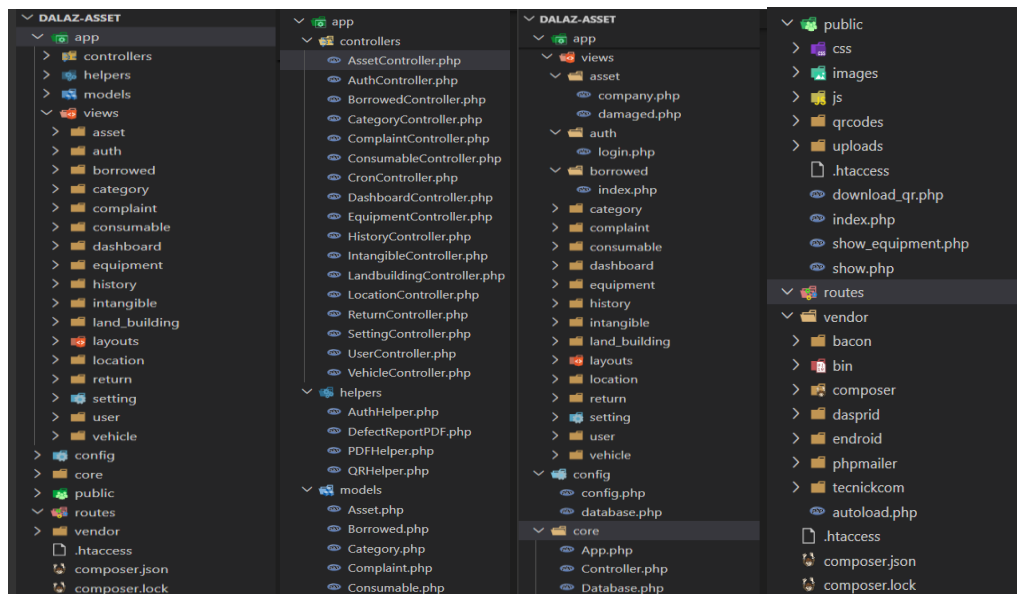
Tabel utama dalam sistem ini meliputi *assets*, *equipment*, *Vehicles*, *land_buildings*, *intangibles*, dan *consumable*, yang masing-masing menyimpan data aset sesuai kategorinya. Setiap tabel aset memiliki referensi ke tabel *category* dan *location*, sehingga setiap aset dapat dikelompokkan berdasarkan jenis dan diketahui lokasi penempatannya. Tabel *Users* berperan sebagai pengelola sistem, di mana setiap aktivitas seperti peminjaman, pengembalian, maupun pengaduan dapat dilacak berdasarkan pengguna yang bertanggung jawab.

Proses peminjaman aset dikelola melalui tabel *borrowed*, yang mencatat informasi seperti peminjam, lokasi peminjaman, jumlah, dan status pengembalian. Tabel *return* kemudian mencatat data pengembalian aset yang telah dipinjam, dengan referensi langsung ke tabel *borrowed* dan *Equipment*. Selain itu, tabel *complaint* digunakan untuk mencatat laporan kerusakan atau kendala pada aset, termasuk penyebab kerusakan, tindakan perbaikan, serta status penanganannya.

Secara keseluruhan, skema basis data ini dirancang untuk memastikan setiap aset dapat dipantau secara lengkap, mulai dari data kepemilikan, kondisi fisik, lokasi penempatan, riwayat peminjaman, hingga penanganan keluhan. Hubungan antar tabel menggunakan kunci asing (*foreign key*) yang memastikan integritas dan konsistensi data di seluruh sistem.

4.1.2 Implementasi Sistem

Sistem Manajemen Aset PT. Dalaz Teknik Utama dikembangkan menggunakan bahasa pemrograman PHP *Native* dengan menerapkan pola arsitektur *Model-View-Controller* (MVC). Arsitektur ini memisahkan logika pemrosesan data dari tampilan antarmuka, sehingga kode program menjadi lebih terstruktur dan mudah dikelola.



Gambar 4. 2 Struktur Direktori MVC pada Sistem

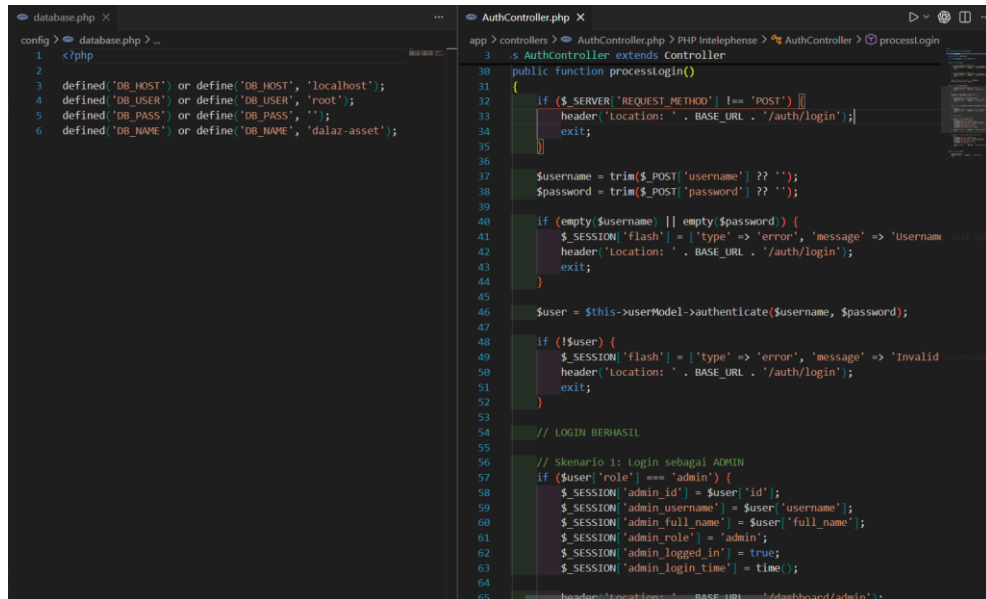
Lingkungan pengembangan sistem dikonfigurasi dengan spesifikasi sebagai berikut :

- Perangkat Keras (*Hardware*): Laptop digunakan sebagai server lokal, didukung oleh perangkat *QR Code Scanner* fisik untuk mempercepat pemindaian identitas aset di lapangan.
- Perangkat Lunak (*Software*): Visual Studio Code sebagai code editor, XAMPP sebagai web server lokal berbasis Apache sekaligus manajemen basis data melalui *phpMyAdmin*, dan *Google Chrome* sebagai peramban utama.
- Basis Data: MySQL digunakan untuk menyimpan seluruh data master aset, data pengguna, serta riwayat transaksi operasional.

Pada sisi *backend*, setiap permintaan dari pengguna diproses melalui lapisan *Controller*. Lapisan ini bertanggung jawab menjalankan aturan bisnis dan validasi data, seperti memastikan kelengkapan formulir pelaporan kerusakan aset dan mencegah transaksi peminjaman apabila stok tidak mencukupi.

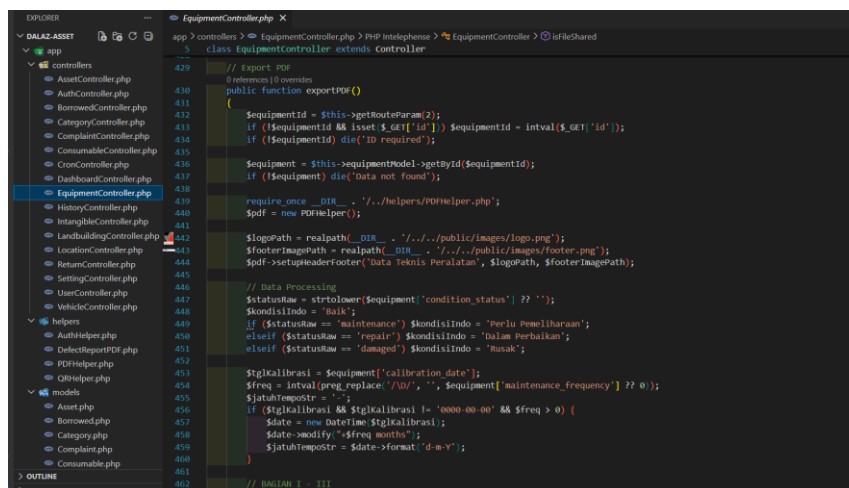
Koneksi antara aplikasi dan basis data MySQL dibangun menggunakan ekstensi *PHP Data Objects* (PDO) untuk menjamin keamanan lalu lintas data. Keamanan sistem juga diperkuat dengan penerapan *Role-Based Access Control* (RBAC)

berbasis sesi untuk membedakan hak akses antara Admin dan *User*, serta enkripsi kata sandi menggunakan algoritma *bcrypt*.



Gambar 4. 3 Implementasi Koneksi Basis Data dan Keamanan *Backend*

Sistem turut mengintegrasikan dua library pihak ketiga untuk mendukung fitur tambahan. Pertama, *library Endroid QR Code* digunakan untuk menghasilkan kode QR secara otomatis setiap kali data aset atau pegawai baru ditambahkan ke dalam sistem. Kedua, *library TCPDF* digunakan untuk menghasilkan dokumen laporan berformat PDF, seperti daftar aset dan formulir laporan kerusakan, yang tampilannya telah disesuaikan dengan format resmi perusahaan.



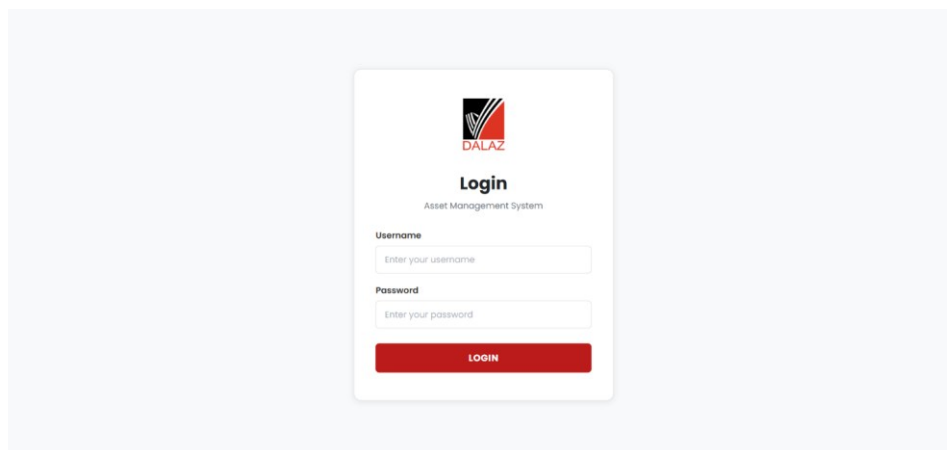
Gambar 4. 4 Implementasi Integrasi *Library TCPDF* pada *Controller*

4.1.3 Implementasi Antarmuka

Implementasi antarmuka merupakan tahap di mana rancangan tampilan aplikasi diwujudkan ke dalam bentuk nyata sehingga dapat digunakan langsung oleh pengguna. Seluruh antarmuka ini dirancang untuk memastikan sistem tidak hanya fungsional, tetapi juga mudah dioperasikan (*User-friendly*). Implementasi antarmuka ini disesuaikan dengan kebutuhan operasional di lapangan, baik untuk peran admin yang mengelola data secara menyeluruh, maupun peran *User/inspector* yang melakukan transaksi harian. Berikut adalah rincian hasil implementasi antarmuka dari sistem yang telah dibangun :

1. Halaman Login

Halaman *login* merupakan halaman pertama yang ditampilkan saat pengguna mengakses sistem. Halaman ini berfungsi sebagai gerbang autentikasi untuk memastikan bahwa hanya pengguna yang terdaftar dapat masuk ke dalam sistem. Terdapat dua *field* input, yaitu *Username* dan *Password*, serta tombol LOGIN untuk memproses autentikasi. Jika *Username* atau *password* yang dimasukkan salah, sistem akan menampilkan pesan *error* bahwa data tidak valid. Namun, apabila data yang dimasukkan benar dan otentikasi berhasil, sistem akan langsung mengarahkan pengguna ke halaman *Dashboard* sesuai dengan hak aksesnya masing-masing, yaitu sebagai Admin atau *User/Inspector*.

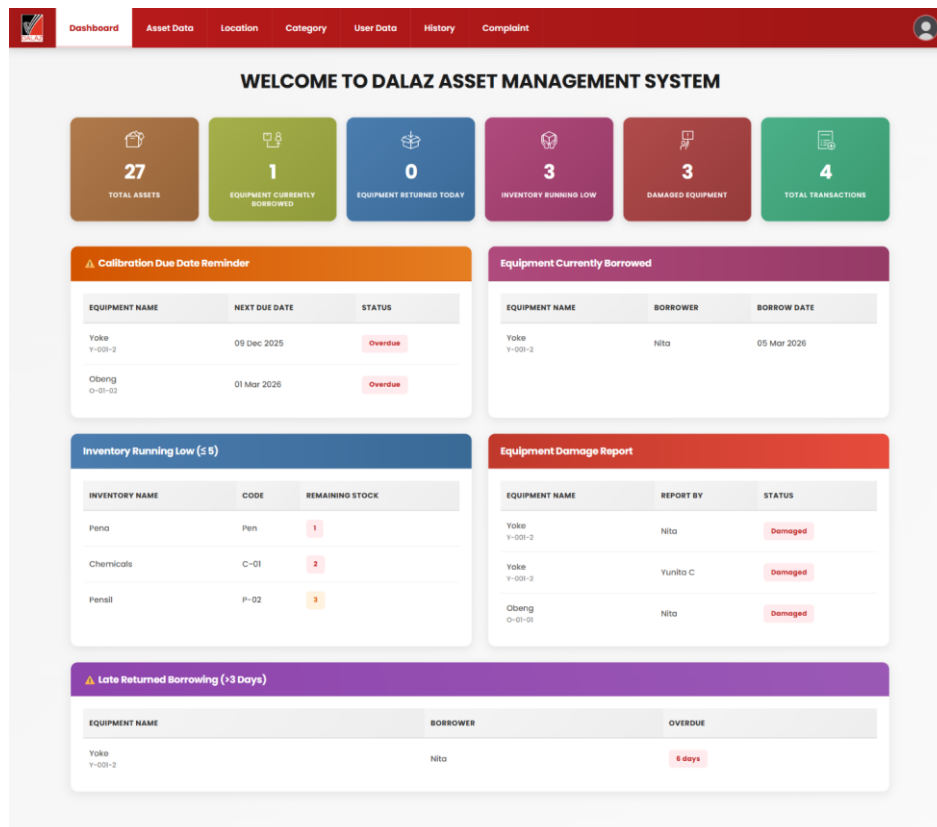


Gambar 4. 5 Implementasi UI *Login*

2. Halaman *Dashboard* Admin

Halaman *Dashboard* Admin adalah tampilan utama yang muncul setelah Admin berhasil melakukan *login*. Halaman ini menyajikan ringkasan kondisi aset secara keseluruhan melalui enam kartu statistik, yaitu *Total Assets*, *Equipment Currently Borrowed*, *Equipment Returned Today*, *Inventory Running Low*, *Damaged Equipment*, dan *Total Transactions*.

Selain kartu statistik, terdapat lima bagian notifikasi yang menampilkan informasi lebih rinci, yaitu *Calibration Due Date Reminder* untuk pengingat jadwal kalibrasi yang telah melewati batas waktu, *Equipment Currently Borrowed* untuk daftar aset yang sedang dipinjam, *Inventory Running Low* untuk stok inventaris yang tersisa lima unit atau kurang, *Equipment Damage Report* untuk laporan kerusakan aset, serta *Late Returned Borrowing* untuk aset yang terlambat dikembalikan lebih dari satu hari.



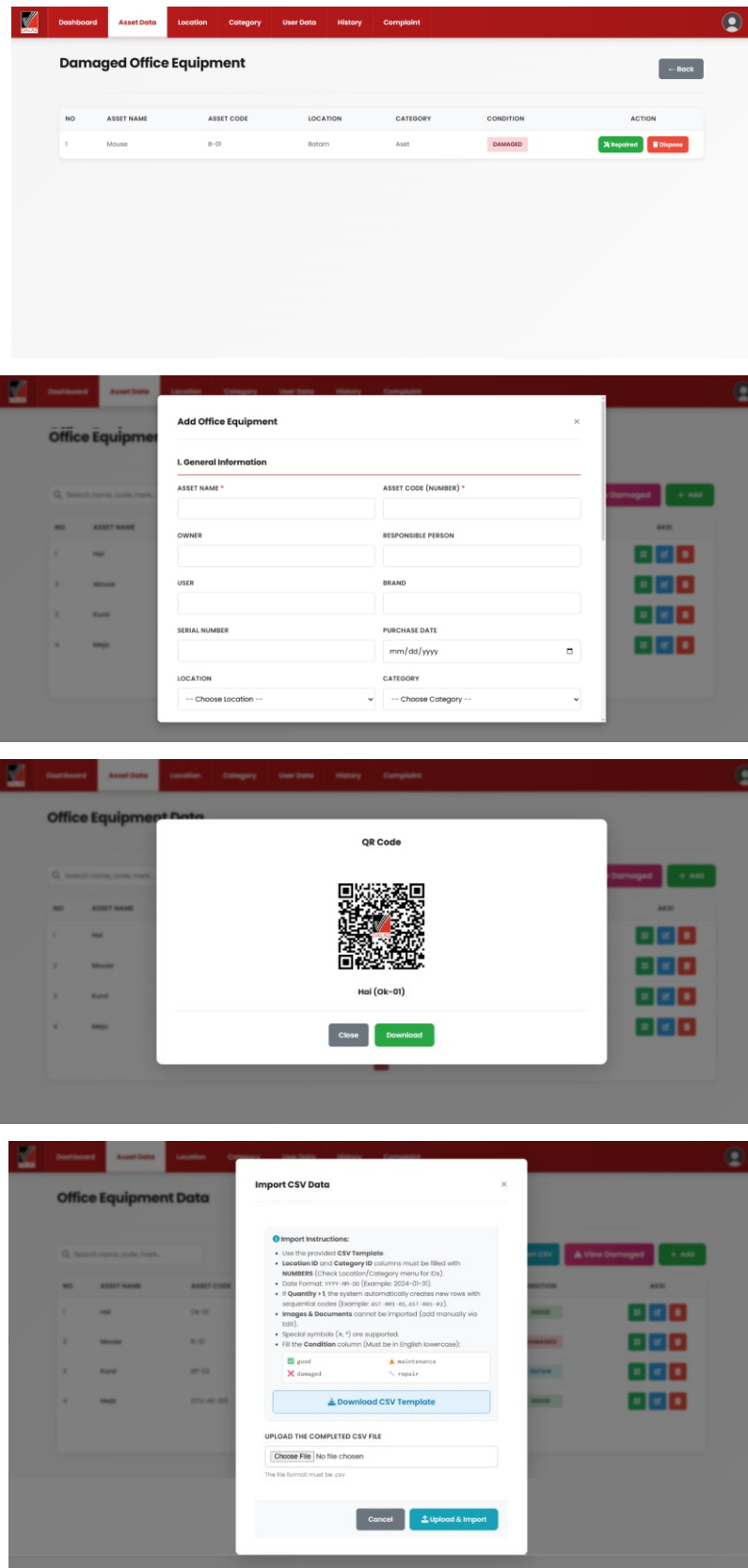
Gambar 4. 6 Implementasi UI *Dashboard* Admin

3. Halaman *Office Equipment Data*

Halaman *Office Equipment Data* menampilkan daftar seluruh aset peralatan kantor yang tercatat dalam sistem. Data ditampilkan dalam bentuk tabel dengan kolom *No*, *Asset Name*, *Asset Code*, *Quantity*, *Location*, *Category*, dan *Condition* yang menunjukkan kondisi aset saat ini, yaitu *Good*, *Damaged*, atau *Repair*. Tersedia *search bar* untuk mencari aset, serta tiga tombol utama yaitu *Import CSV* untuk menambah data secara massal, *View Damaged* untuk melihat daftar aset rusak, dan *Add* untuk menambah aset baru secara manual melalui formulir yang terbagi menjadi tiga bagian, yaitu *General Information*, *Technical Specifications*, dan *Maintenance*.

Pada setiap baris terdapat tombol *QR Code* untuk menampilkan dan mengunduh kode QR aset, tombol edit untuk mengubah data, serta tombol hapus yang dilengkapi *pop-up* konfirmasi sebelum data dihapus secara permanen. Tombol *View Damaged* akan mengarahkan Admin ke halaman *Damaged Office Equipment* yang menampilkan daftar aset rusak beserta dua tombol aksi, yaitu *Repaired* untuk menandai aset telah diperbaiki dan *Dispose* untuk menghapus aset yang sudah tidak dapat digunakan.

NO	ASSET NAME	ASSET CODE	QUANTITY	LOCATION	CATEGORY	CONDITION	AKSI
1	Hal	OK-01	1 Unit	-	-	GOOD	  
2	Mouse	B-01	1 Unit	Botam	Aset	DAMAGED	  
3	Kursi	BP-02	1 Unit	-	-	REPAIR	  
4	Meja	DTU-AK-001	1 Unit	Botam	Aset	GOOD	  



Gambar 4. 7 Implementasi UI *Office Equipment Data*

4. Halaman *Equipment Data*

Halaman *Equipment Data* menampilkan daftar aset peralatan yang memerlukan kalibrasi berkala. Data ditampilkan dalam bentuk tabel dengan kolom *No*, *Equipment Name*, *Asset Number*, *Serial Number*, *Quantity*, *Location*, *Category*, dan *Next Due Date* yang menunjukkan jadwal kalibrasi berikutnya. Kolom *Next Due Date* akan ditampilkan dalam warna merah apabila jadwal kalibrasi sudah dekat atau telah terlewat sebagai peringatan bagi Admin.

Pada bagian atas halaman terdapat *search bar* untuk mencari data berdasarkan nama atau nomor aset, serta dua tombol utama yaitu *Import CSV* dan *Add*. Saat tombol *Add* diklik, akan muncul formulir *Add New Equipment* yang terbagi menjadi lima bagian, yaitu *General Information* yang mencakup nama, kode, pemilik, penanggung jawab, merek, nomor seri, produsen, tanggal pembelian, lokasi, dan kategori; *Technical Specifications*; *Maintenance Information* yang mencakup data kalibrasi dan vendor; *Usage Steps* untuk panduan penggunaan alat; serta *Documentation* untuk mengunggah foto dan dokumen pendukung. Saat tombol *Import CSV* diklik, akan muncul modal yang menampilkan petunjuk pengisian beserta tombol *Download CSV Template* untuk mengunduh format yang benar, dan tombol *Upload & Import* untuk mengunggah file CSV yang telah diisi.

Pada setiap baris data terdapat empat tombol aksi, yaitu tombol PDF berwarna merah untuk mengunduh laporan aset dalam format PDF, tombol *QR Code* berwarna hijau untuk menampilkan kode QR aset, tombol edit berwarna biru untuk mengubah data, serta tombol hapus berwarna merah yang dilengkapi *pop-up* konfirmasi sebelum data dihapus secara permanen dari sistem.

NO	EQUIPMENT NAME	ASSET NUMBER	SERIAL NUMBER	QUANTITY	LOCATION	CATEGORY	NEXT DUE DATE	ACTION
1	Obeng	O-01-02	aso	1 Unit	Botom	Equipment	01 March 2026	[Icons]
2	Obeng	O-01-01	aso	1 Unit	Botom	Equipment	18 October 2026	[Icons]
3	Yoke	Y-001-2	Y-01	0 Unit	Botom	Equipment	09 December 2025	[Icons]

Add New Equipment

I. General Information

EQUIPMENT NAME *

ASSET NUMBER (CODE) *

OWNER

RESPONSIBLE PERSON

MERK / TYPE

SERIAL NUMBER

MANUFACTURER

PURCHASE DATE
mm/dd/yyyy

LOCATION
-- Select Location --

CATEGORY
-- Select Category --

Import Data CSV

Filling Instructions:

- Use the provided **CSV Template**.
- Location ID & Category ID** must be filled with numbers.
- Date format must be **YYYY-MM-DD** (Example: 2024-12-31).
- If **Quantity > 1**, the system automatically creates a new row with a 2-digit suffix (Example: eq-001-01).
- Usage steps are in the last column (can be plain text or HTML).
- Filling in the **Condition** column (English required).

good

damaged

🔧 maintenance

🔧 repair

[Download CSV Template](#)

UPLOAD CSV FILE

Choose File | No file chosen

Cancel | Upload & Import

Gambar 4. 8 Implementasi UI *Equipment Data*

5. Halaman *Inventory Data*

Halaman *Inventory Data* menampilkan daftar seluruh barang habis pakai (*consumable*) yang tercatat dalam sistem. Data ditampilkan dalam bentuk tabel dengan kolom *No*, *Asset Name*, *Code*, *Stock*, *Unit*, *Location*,

Category, dan *Condition*. Pada bagian atas halaman terdapat *search bar* untuk mencari item berdasarkan nama atau kode, serta tiga tombol utama yaitu *In/Out Stock* berwarna kuning untuk mengelola transaksi stok, *Import CSV* untuk menambah data secara massal, dan *Add* untuk menambah item baru secara manual. Pada setiap baris terdapat tiga tombol aksi, yaitu tombol detail berwarna biru muda untuk melihat rincian item, tombol edit berwarna biru untuk mengubah data, dan tombol hapus berwarna merah.

Tombol *In/Out Stock* akan mengarahkan Admin ke halaman *In/Out Stock (Inventory)* yang mencatat seluruh riwayat transaksi stok masuk dan keluar. Tabel pada halaman ini memuat kolom *Transaction Date*, *Asset Code*, *Asset Name*, *Status*, *Remark/PIC*, *Qty In/Out*, dan *Current Stock*. Untuk mencatat transaksi baru, Admin dapat mengklik tombol *Record Transactions* yang akan menampilkan formulir *Record New Stock*. Formulir ini mencakup *field* pencarian item, jumlah stok saat ini, kuantitas, tanggal, aksi (*IN* atau *OUT*), keterangan, serta unggahan dokumen pendukung.

Tombol detail pada halaman *Inventory Data* akan mengarahkan Admin ke halaman *Detail Inventory & History* yang menampilkan informasi lengkap suatu item, meliputi *Asset Code*, *Category*, *Location*, *Current Stock*, *Min. Order Alert*, dan *Condition*, beserta riwayat seluruh transaksi stok yang pernah dilakukan, mencakup tanggal, status *Stock In* atau *Stock Out*, perubahan jumlah, stok terkini, dan keterangan

NO	ASSET NAME	CODE	STOCK	UNIT	LOCATION	CATEGORY	CONDITION	ACTION
1	Pens	P-01	3	Box	Batam	Consumable	Good	[Detail] [Edit] [Delete]
2	Masker	M-01	0	Box	Batam	APD	Good	[Detail] [Edit] [Delete]
3	Chemicals	C-01	2	Box	Batam	Asset	Good	[Detail] [Edit] [Delete]
4	Pensil	P-02	3	Pcs	Batam	ATK	Good	[Detail] [Edit] [Delete]
5	Pena	PEN	1	Box	-	ATK	Good	[Detail] [Edit] [Delete]


Dashboard
Asset Data
Location
Category
User Data
History
Complaint



Detail Inventory & History

Pena

Asset Code

: P-001

Category

: ATK (Alat Tulis Kantor)

Location

: Batam

Current Stock

: 8 Pcs

Min. Order Alert

: 6

Condition

: Good

Date

mm/dd/yyyy

Status

All Status

Search Remark / PIC

Type remark keyword...

Clear

NO	TRANSACTION DATE	STATUS	QTY CHANGE	CURRENT STOCK	REMARK
1	17 Jun 2026	STOCK OUT	-2	8	Yunita
2	16 Jun 2026	STOCK OUT	-2	10	Yuni
3	16 Jun 2026	STOCK OUT	-3	12	Yunita

< Prev

1

Next >


Dashboard
Asset Data
Location
Category
User Data
History
Complaint



In / Out Stock (Inventory)

Record Transactions

Back

Page Information

This page records incoming and outgoing transactions.
The data below shows the **latest status** of each item that has ever been transacted.

Search Asset

Search by Code or Name...

Filter by Date

mm/dd/yyyy

Filter by Status

All Status

Clear

NO	TRANSACTION DATE	ASSET CODE	ASSET NAME	STATUS	REMARK/PIC	QTY IN/OUT	CURRENT STOCK
1	2026-06-17	P-001	Pena	OUT	Yunita	2	8
2	2026-06-16	P-001	Pena	OUT	Yuni	2	10
3	2026-06-16	P-001	Pena	OUT	Yunita	3	12
4	2026-06-16	M-01	Masker	IN	Yunita	3	3
5	2026-01-08	C-01	Chemicals	IN	Dilempar dari CSV	2	2
6	2026-01-08	P-02	Pensil	IN	Stok Awal	3	3

< Prev

1

Next >


Dashboard
Asset Data
Location
Category
User Data
History
Complaint



In / Out Stock (Inventory)

Record Transactions

Back

Page Information

This page records incoming and outgoing transactions.
The data below shows the **latest status** of each item that has ever been transacted.

Search Asset

Search by Code or Name...

Filter by Date

mm/dd/yyyy

Filter by Status

All Status

Clear

NO	TRANSACTION DATE	ASSET CODE	ASSET NAME	STATUS	REMARK/PIC	QTY IN/OUT	CURRENT STOCK
1	2026-06-17	P-001	Pena	OUT	Yunita	2	8
2	2026-06-16	P-001	Pena	OUT	Yuni	2	10
3	2026-06-16	P-001	Pena	OUT	Yunita	3	12
4	2026-06-16	M-01	Masker	IN	Yunita	3	3
5	2026-01-08	C-01	Chemicals	IN	Dilempar dari CSV	2	2
6	2026-01-08	P-02	Pensil	IN	Stok Awal	3	3

< Prev

1

Next >

Record New Stock

SEARCH ITEM (CODE OR NAME) *

Type the asset code or name...

CURRENT STOCK

QUANTITY *

DATE *

06/12/2026

ACTION *

-- Select --

REMARK

SUPPORTING DOCUMENT (pdf)

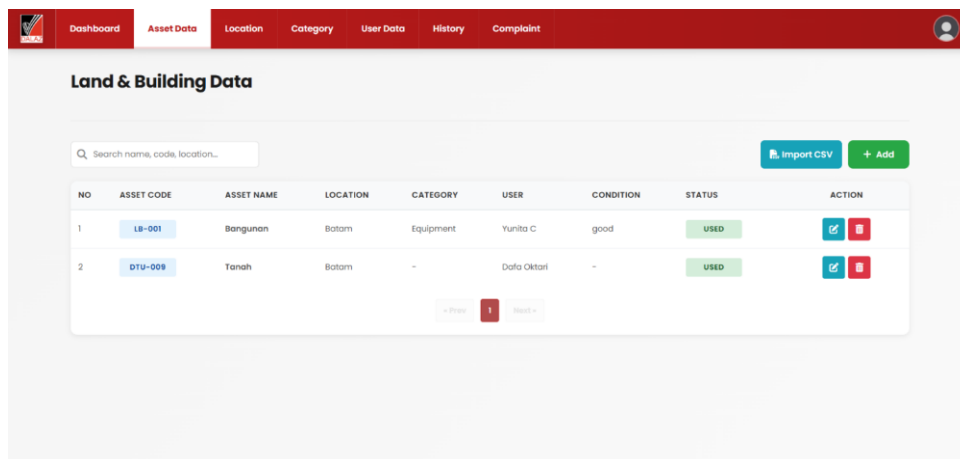
Choose File | No file chosen

Optional. Max 5MB.

Gambar 4. 9 Implementasi UI *Inventory*

6. Halaman *Land & Building Data*

Halaman *Land & Building Data* menampilkan daftar seluruh aset tanah dan bangunan yang dimiliki perusahaan. Data ditampilkan dalam bentuk tabel dengan kolom *No*, *Asset Code*, *Asset Name*, *Location*, *Category*, *User*, *Condition*, dan *Status* yang menunjukkan status penggunaan aset. Pada bagian atas halaman terdapat *search bar* untuk mencari aset berdasarkan nama, kode, atau lokasi, serta dua tombol utama yaitu *Import CSV* untuk menambah data secara massal dan *Add* untuk menambah aset baru secara manual. Pada setiap baris terdapat dua tombol aksi, yaitu tombol edit berwarna biru untuk mengubah data dan tombol hapus berwarna merah yang dilengkapi *pop-up* konfirmasi sebelum data dihapus secara permanen dari sistem.

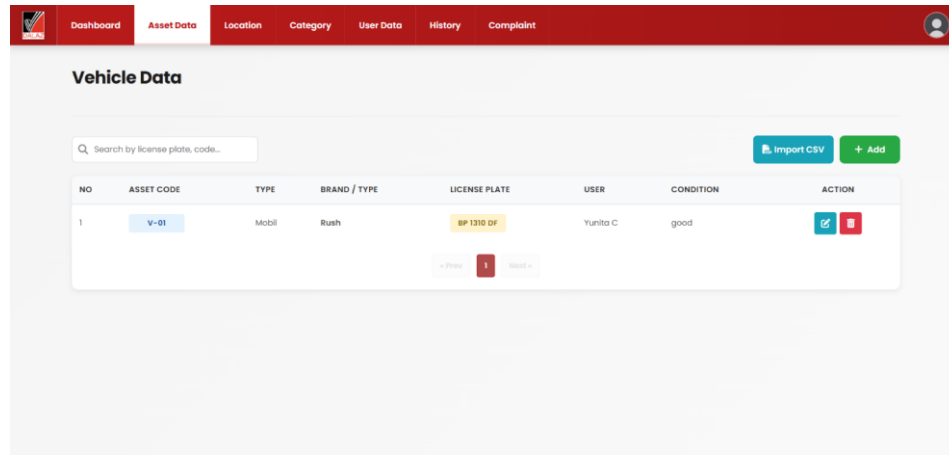


Gambar 4. 10 Implementasi UI *Land & Building Data*

7. Halaman *Vehicle Data*

Halaman *Vehicle Data* menampilkan daftar seluruh aset kendaraan yang dimiliki perusahaan. Data ditampilkan dalam bentuk tabel dengan kolom *No*, *Asset Code*, *Type*, *Brand/Type*, *License Plate*, *User*, dan *Condition*. Pada bagian atas halaman terdapat *search bar* untuk mencari kendaraan berdasarkan nomor plat atau kode aset, serta dua tombol utama yaitu *Import CSV* untuk menambah data secara massal dan *Add* untuk menambah data kendaraan baru secara manual. Pada setiap baris terdapat dua tombol aksi, yaitu tombol edit berwarna biru untuk mengubah data dan tombol hapus

berwarna merah yang dilengkapi *pop-up* konfirmasi sebelum data dihapus secara permanen dari sistem.

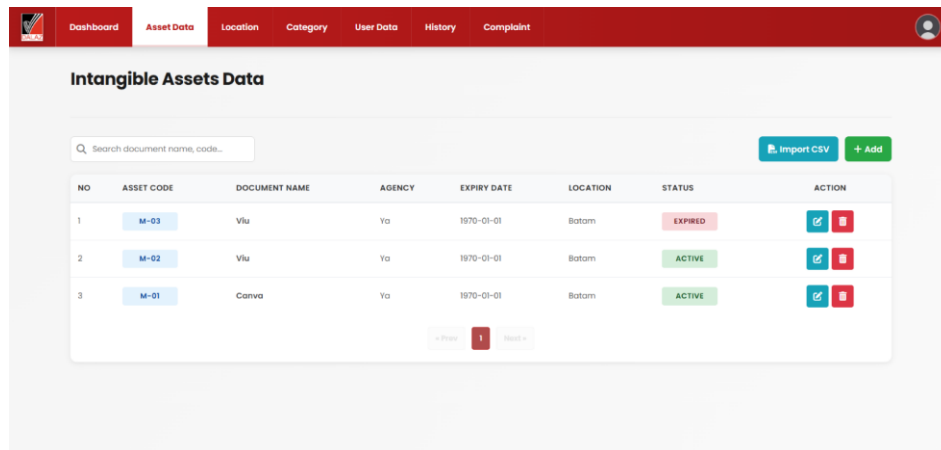


Gambar 4. 11 Implementasi UI *Vehicle Data*

8. Halaman *Intangible Assets Data*

Halaman *Intangible Assets Data* menampilkan daftar seluruh aset tidak berwujud milik perusahaan, seperti lisensi perangkat lunak dan dokumen legal. Data ditampilkan dalam bentuk tabel dengan kolom No, *Asset Code*, *Document Name*, *Agency*, *Expiry Date*, *Location*, dan *Status* yang menunjukkan apakah dokumen masih *Active* atau telah *Expired*. Pada bagian atas halaman terdapat *search bar* untuk mencari dokumen berdasarkan nama atau kode, serta dua tombol utama yaitu *Import CSV* dan *Add*.

Saat tombol *Add* diklik, akan muncul formulir *Add New Document* yang terbagi menjadi tiga bagian. Bagian *General Information* mencakup *Asset Code* dan *Document Name* sebagai *field* wajib, serta *Location* dan *Category* yang dipilih melalui *dropdown*. Bagian *Document Details* mencakup *Certificate Number*, *Issuing Agency*, *Issue Date*, dan *Effective Date*. Bagian *Status & Upload* mencakup *Expiration Date*, *Document Status* (*Active* atau *Expired*), dan unggahan file dokumen dalam format PDF dengan ukuran maksimal 2MB. Pada setiap baris terdapat tombol edit berwarna biru untuk mengubah data dan tombol hapus berwarna merah yang dilengkapi *pop-up* konfirmasi sebelum data dihapus secara permanen.

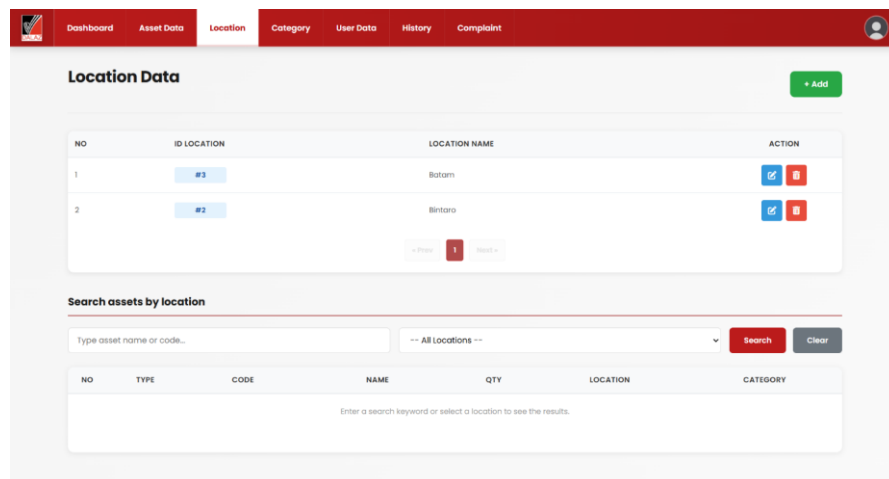


Gambar 4. 12 Implementasi UI *Intangible Assets Data*

9. Halaman *Location*

Halaman *Location Data* menampilkan daftar lokasi tempat aset perusahaan berada. Data ditampilkan dalam bentuk tabel dengan kolom No, *ID Location*, dan *Location Name*. Pada pojok kanan atas terdapat tombol *Add* untuk menambah lokasi baru, sedangkan pada setiap baris terdapat tombol edit berwarna biru dan tombol hapus berwarna merah.

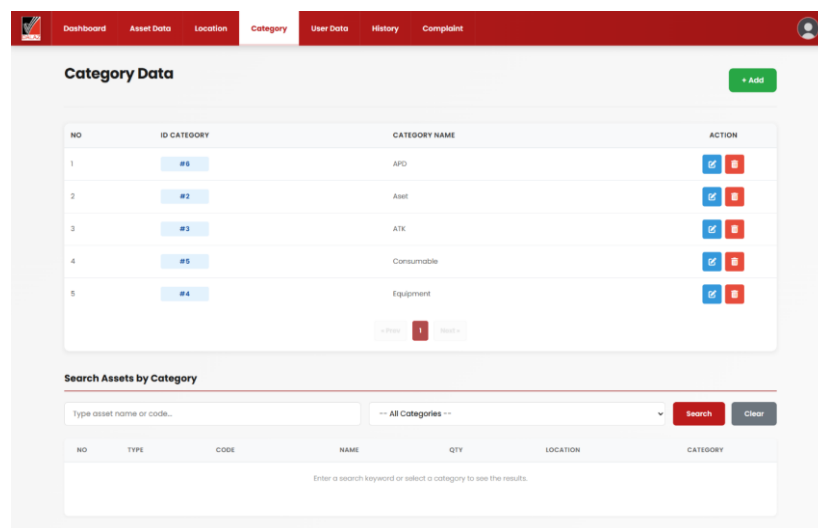
Selain tabel data lokasi, halaman ini juga dilengkapi fitur *Search Assets by Location* di bagian bawah yang memungkinkan Admin mencari aset berdasarkan nama atau kode aset dengan memilih lokasi tertentu melalui *dropdown*. Hasil pencarian akan ditampilkan dalam tabel dengan kolom No, *Type*, *Code*, *Name*, *Qty*, *Location*, dan *Category*.



Gambar 4. 13 Implementasi UI *Location*

10. Halaman *Category*

Halaman *Category Data* menampilkan daftar kategori aset yang digunakan untuk pengelompokan aset. Data ditampilkan dalam bentuk tabel dengan kolom No, *ID Category*, dan *Category Name*. Pada pojok kanan atas terdapat tombol *Add* untuk menambah kategori baru, sedangkan pada setiap baris terdapat tombol edit berwarna biru dan tombol hapus berwarna merah. Sama seperti halaman *Location*, halaman ini juga dilengkapi fitur *Search Assets by Category* di bagian bawah yang memungkinkan Admin mencari aset berdasarkan nama atau kode dengan memilih kategori tertentu melalui *dropdown*. Hasil pencarian akan ditampilkan dalam tabel dengan kolom No, *Type*, *Code*, *Name*, *Qty*, *Location*, dan *Category*.



Gambar 4. 14 Implementasi UI *Category*

11. Halaman *User Data*

Halaman *User Data* menampilkan daftar seluruh pengguna yang terdaftar pada sistem. Data ditampilkan dalam bentuk tabel dengan kolom No, *User ID*, *Employee Number*, *Full Name*, *Position*, dan *Role* yang menunjukkan peran pengguna, yaitu *User* atau *Admin*. Pada pojok kanan atas terdapat tombol *Add* untuk menambah pengguna baru.

Saat tombol *Add* diklik, akan muncul formulir *Add New User* yang memuat *field* wajib berupa *Full Name*, *Employee Number*, dan *Role*.

Apabila *role* yang dipilih adalah *User*, formulir hanya menampilkan *field* nama, nomor karyawan, jabatan, *role*, dan foto profil. Sedangkan apabila *role* yang dipilih adalah Admin, formulir akan menampilkan tambahan *field* berupa *Username*, *Email Address*, dan *Password* yang diperlukan untuk keperluan *login*. Pada setiap baris terdapat tiga tombol aksi, yaitu tombol *QR Code* berwarna hijau untuk menampilkan kode QR pengguna, tombol edit berwarna biru untuk mengubah data, dan tombol hapus berwarna merah yang dilengkapi *pop-up* konfirmasi sebelum data dihapus secara permanen.

NO	USER ID	EMPLOYEE NUMBER	FULL NAME	POSITION	ROLE	ACTION
1	#10	M-002	Dafa Oktari	Inspector	USER	[QR] [Edit] [Delete]
2	#11	U-08	Nita	Manager	USER	[QR] [Edit] [Delete]
3	#9	M-001	Yunita C	Admin	ADMIN	[QR] [Edit] [Delete]

Add New User

FULL NAME *

POSITION

EMPLOYEE NUMBER *

ROLE *

PROFILE PICTURE (OPTIONAL) No file chosen

SIGNATURE (OPTIONAL) No file chosen

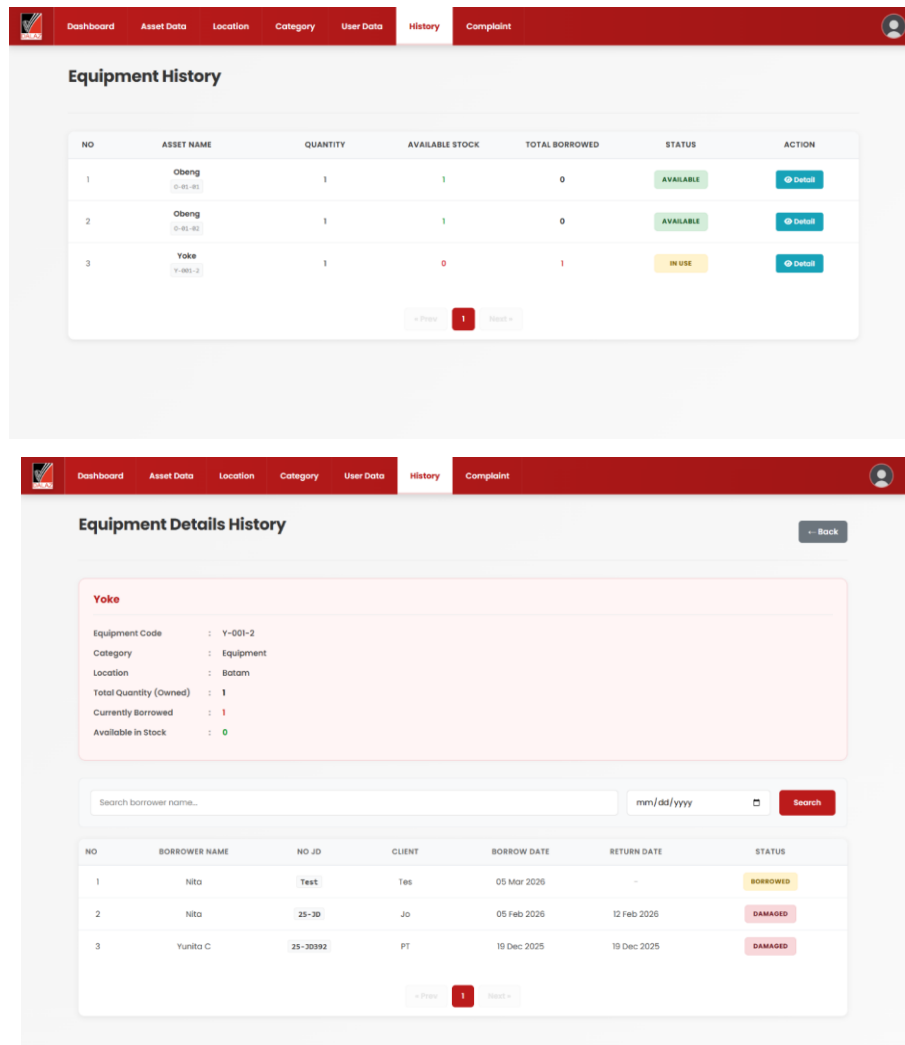
* Recommended PNG file with transparent background for reports.

Gambar 4. 15 Implementasi UI *User Data*

12. Halaman *History Admin*

Halaman *Equipment History* menampilkan rekap seluruh aset *Equipment* beserta status ketersediaannya. Data ditampilkan dalam bentuk tabel dengan kolom No, *Asset Name*, *Quantity*, *Available Stock*, *Total Borrowed*, dan *Status* yang menunjukkan kondisi aset saat ini, yaitu *Available* atau *In Use*. Pada setiap baris terdapat tombol *Detail* untuk melihat rincian riwayat peminjaman aset tersebut.

Saat tombol *Detail* diklik, Admin akan diarahkan ke halaman *Equipment Details History* yang menampilkan informasi lengkap aset meliputi *Equipment Code*, *Category*, *Location*, *Total Quantity (Owned)*, *Currently Borrowed*, dan *Available in Stock*. Di bawah informasi tersebut terdapat tabel riwayat peminjaman yang memuat kolom *Borrower Name*, *No JD*, *Client*, *Borrow Date*, *Return Date*, dan *Status* yang menunjukkan status peminjaman seperti *Borrowed* atau *Damaged*. Tersedia pula fitur pencarian berdasarkan nama peminjam dan filter tanggal untuk memudahkan Admin dalam menelusuri riwayat transaksi.



Gambar 4. 16 Implementasi UI *History* Admin

13. Halaman *Manage Complaint*

Halaman *Manage Complaints* menampilkan daftar seluruh laporan kerusakan aset yang dikirimkan oleh pengguna. Data ditampilkan dalam bentuk tabel dengan kolom *No*, *Asset Info*, *Reported By*, *Description*, *Date*, *Status*, dan *Response*. Kolom *Status* menunjukkan status penanganan keluhan, yaitu *Pending Response*, *Repair*, atau *Disposal*.

Apabila keluhan belum ditanggapi, tombol *Response* akan muncul pada kolom *Action* untuk membuka formulir *Admin Response Form*. Formulir ini menampilkan informasi keluhan secara otomatis, meliputi *Asset Number*, *Equipment Name*, *Reported By*, *Date of Complaint*, *Time of*

Complaint, Defect Cause/Chronology, dan Evidence Photos. Admin kemudian mengisi *field* wajib berupa *Control No, Date of Inspection, Treatment/Action* untuk mencatat tindakan yang diambil, serta *Status* untuk menentukan hasil penanganan. Apabila keluhan sudah ditanggapi, tombol aksi akan berubah menjadi tombol *Edit* untuk mengubah respons dan tombol hapus untuk menghapus data keluhan.

The image shows a web application interface for managing complaints. The top navigation bar includes links for Dashboard, Asset Data, Location, Category, User Data, History, and Complaint. The main section is titled 'Manage Complaints' and displays a table of complaints. Below the table, there is a modal form titled 'Admin Response Form' for adding or updating a response.

NO	ASSET INFO	REPORTED BY	DESCRIPTION	DATE	STATUS	RESPONSE	ACTION
1	Obeng (0-01-01)	Nita	retak saat di lapangan	05 Mar 2026	PENDING RESPONSE	-	Response
2	Yoke (Y-005-2)	Nita	barat	12 Feb 2026	REPAIR	Iya	edit
3	Yoke (Y-005-2)	Yunita C	losses	19 Dec 2025	DISPOSAL	okee akan saya diperbaiki	edit

Admin Response Form

ASSET NUMBER: 0-01-01 EQUIPMENT NAME: Obeng

REPORTED BY: Nita DATE OF COMPLAINT: 2026-03-05

TIME OF COMPLAINT: 11:44:04

DEFECT CAUSE / CHRONOLOGY: retak saat di lapangan

EVIDENCE PHOTOS:

CONTROL NO * DATE OF INSPECTION *
 mm/dd/yyyy

TREATMENT / ACTION *
 Describe the treatment/action taken...

STATUS *
 -- Select Status --

Buttons: Cancel, Save Response

Gambar 4. 17 Implementasi UI *Manage Complaint*

14. Halaman *Settings*

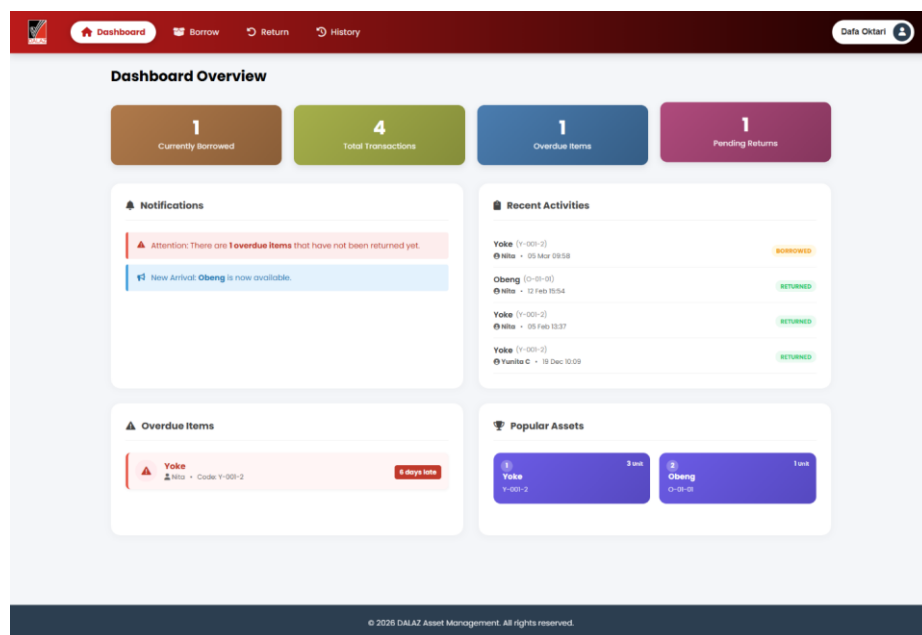
Halaman *Settings* tersedia untuk Admin maupun *User*, dengan tampilan yang sama. Halaman ini memungkinkan pengguna untuk mengelola profil dan keamanan akun, yang terbagi menjadi dua bagian. Bagian *Profile* memuat *field Full Name* yang dapat diubah, *Username* yang bersifat tetap dan tidak dapat diubah, *Employee Number* yang dikelola oleh Administrator, serta *Position*. Bagian *Password* bersifat opsional dan hanya diisi apabila pengguna ingin mengganti kata sandi, dengan mengisi *Current Password*, *New Password*, dan *Confirm New Password*. Terdapat tombol *Save Changes* untuk menyimpan perubahan dan tombol *Cancel* untuk membatalkan.

Gambar 4. 18 Implementasi UI Setting

15. Halaman *Dashboard User*

Halaman *Dashboard User* merupakan halaman utama yang ditampilkan setelah *User* berhasil *login*. Berbeda dengan *Dashboard Admin*, tampilan ini lebih sederhana dan berfokus pada aktivitas peminjaman pengguna. Halaman ini menyajikan empat kartu statistik, yaitu *Currently Borrowed* untuk jumlah aset yang sedang dipinjam, *Total Transactions* untuk total seluruh transaksi, *Overdue Items* untuk aset yang terlambat dikembalikan, dan *Pending Returns* untuk aset yang menunggu konfirmasi pengembalian.

Selain kartu statistik, terdapat empat bagian informasi tambahan, yaitu *Notifications* yang menampilkan peringatan aset terlambat dikembalikan dan informasi aset baru yang tersedia; *Recent Activities* yang menampilkan riwayat aktivitas peminjaman dan pengembalian terkini beserta nama peminjam dan waktu transaksi; *Overdue Items* yang menampilkan daftar aset yang belum dikembalikan melebihi batas waktu beserta informasi keterlambatannya; serta *Popular Assets* yang menampilkan aset paling sering dipinjam oleh pengguna.



Gambar 4. 19 Implementasi UI *Dashboard User*

16. Halaman *Borrow Equipment*

Halaman *Borrow Equipment* merupakan halaman yang digunakan *User* untuk melakukan proses peminjaman aset. Proses peminjaman terbagi menjadi tiga tahap yang ditampilkan secara berurutan melalui indikator langkah di bagian atas halaman.

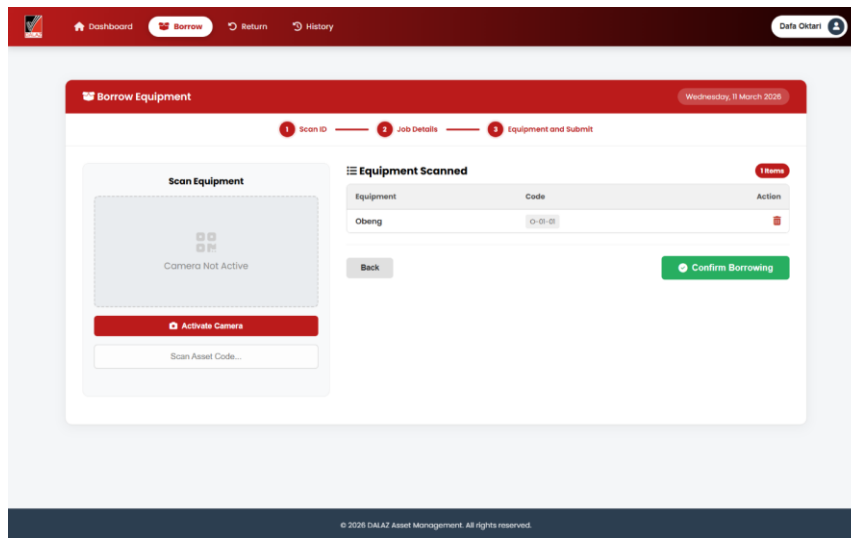
Tahap pertama adalah *Scan ID*, di mana *User* diminta untuk mengidentifikasi diri dengan mengarahkan alat *QR scanner* ke kartu karyawan. Kode akan terbaca secara otomatis melalui *field* yang tersedia dan sistem akan langsung melanjutkan ke tahap berikutnya. Tahap kedua adalah *Job Details*, di mana *User* mengisi informasi pekerjaan berupa *No*

JD (Job Description), *Client*, *Location*, dan *Working Days* sebagai data wajib sebelum melanjutkan ke tahap berikutnya. Tahap ketiga adalah *Equipment and Submit*, di mana *User* mengarahkan alat *QR scanner* ke kode QR aset yang akan dipinjam melalui *field* yang tersedia. Aset yang berhasil terbaca akan ditampilkan secara otomatis dalam tabel *Equipment Scanned* beserta kodenya. Apabila terdapat aset yang terpindai secara keliru, *User* dapat menghapusnya dengan mengklik tombol hapus berwarna merah pada baris aset tersebut. Tombol *Confirm Borrowing* hanya akan aktif apabila minimal satu aset telah berhasil dipindai, dan *User* dapat mengkliknya untuk menyelesaikan proses peminjaman.

The image displays two screenshots of a web application titled "Borrow Equipment".

The top screenshot shows the "Identify Yourself" step. It features a progress bar with three steps: "1 Scan ID", "2 Job Details", and "3 Equipment and Submit". The current step is "1 Scan ID". Below the progress bar, there is a section titled "Identify Yourself" with a camera icon and the text "Camera Not Active". Below this, there is a red button labeled "Activate Camera" and a text input field labeled "Scan employee ID...". The date "Wednesday, 11 March 2020" is displayed in the top right corner.

The bottom screenshot shows the "Job Details" step. It features a progress bar with the same three steps, but the current step is "2 Job Details". Below the progress bar, there is a blue banner with a user icon and the text "Welcome! Welcome, Dafa Oktari". Below the banner, there is a section titled "Job Details" with four input fields: "NO JD (JOB DESCRIPTION)", "CLIENT", "LOCATION", and "WORKING DAYS". The "WORKING DAYS" field has the value "1" entered. Below the input fields, there are two buttons: "Back" and "Next". The date "Sunday, 14 June 2020" is displayed in the top right corner.



Gambar 4. 20 Implementasi UI *Borrow*

17. Halaman *Equipment Return*

Halaman *Equipment Return* merupakan halaman yang digunakan *User* untuk melakukan proses pengembalian aset. Proses pengembalian terbagi menjadi dua tahap yang ditampilkan melalui indikator langkah di bagian atas halaman.

Tahap pertama adalah *Scan User*, di mana *User* diminta mengidentifikasi peminjam dengan mengarahkan alat *QR scanner* ke *QR Code* pada kartu karyawan. ID akan terbaca secara otomatis melalui *field* yang tersedia, kemudian sistem akan memuat data pinjaman aktif dan melanjutkan ke tahap berikutnya. Apabila *scanner* tidak dapat membaca, *Employee ID* dapat diketik secara manual.

Tahap kedua adalah *Return Items*, di mana sistem menampilkan daftar aset yang sedang dipinjam oleh peminjam tersebut. Apabila peminjam tidak memiliki pinjaman aktif, tabel akan menampilkan pesan "*No borrowed items found*". Jika terdapat pinjaman aktif, *User* mengarahkan *scanner* ke *QR Code* aset yang akan dikembalikan. Aset yang berhasil dipindai akan muncul dalam tabel dengan status *Ready*. *User* kemudian memeriksa kondisi fisik aset dan menyesuaikan *dropdown Condition* pada setiap baris, yaitu *Good* apabila aset dalam kondisi baik, atau *Lost/Damaged* apabila terdapat kerusakan.

Apabila kondisi aset diubah menjadi selain *Good*, formulir *Defect Report* akan muncul secara otomatis. *User* wajib mengisi kronologi kerusakan pada kolom *Defect Cause/Chronology* dan mengunggah minimal satu foto bukti kerusakan sebelum pengembalian dapat diproses. Laporan ini akan diteruskan kepada Admin untuk ditindaklanjuti. Setelah semua kondisi aset ditentukan, *User* mengklik tombol *Confirm Return* untuk menyelesaikan proses pengembalian.

The interface consists of a top navigation bar with links to Dashboard, Borrow, Return, and History. The main content area is titled 'Equipment Return' and shows the date 'Wednesday, 8 March 2028'.

Step 1: Scan User

Identify Borrower
Scan User QR Code to load borrowed items.

Camera Inactive

Activate Camera

Scan Employee ID

Step 2: Return Items

BORROWER: Nita
EMPLOYEE ID: U-08

Scan Item to Return

Camera Off

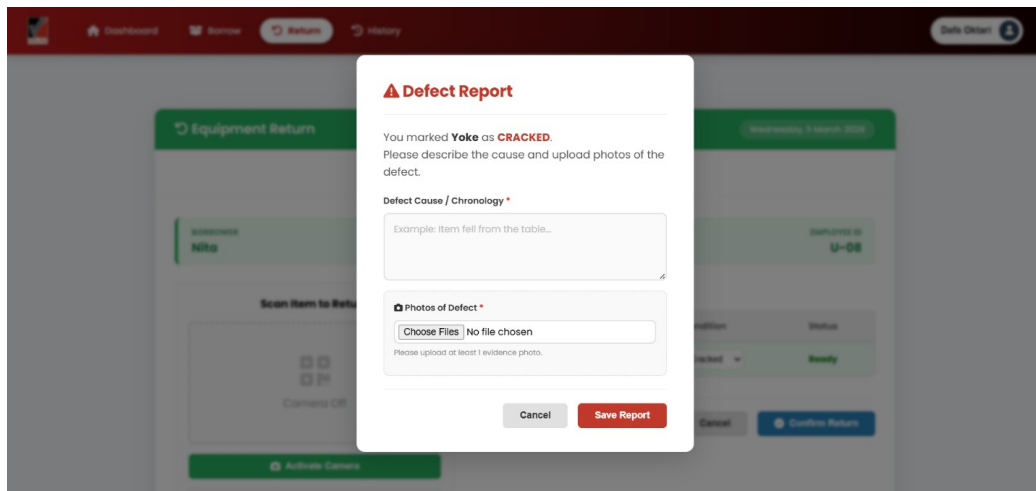
Activate Camera

Scan Asset QR Code here

Borrowed Items List

Asset Name	Code	Condition	Status
Yoke	Y-001-2	Good	Ready

Cancel Confirm Return



Gambar 4. 21 Implementasi UI *Return*

18. Halaman *History User*

Halaman *History of All Transactions* menampilkan seluruh riwayat transaksi peminjaman dan pengembalian aset dari semua pengguna. Data ditampilkan dalam bentuk tabel dengan kolom No, *Borrower*, *Asset Name*, *Asset Code*, *Borrow Date*, *Return Date*, dan *Status*. Kolom *Return Date* ditampilkan dalam warna hijau untuk membedakannya dari tanggal peminjaman, sedangkan kolom *Status* menunjukkan status transaksi seperti *Returned* atau *Borrowed*. Pada pojok kanan atas halaman terdapat informasi *Total Records* yang menampilkan jumlah keseluruhan transaksi yang tercatat.

NO	BORROWER	ASSET NAME	ASSET CODE	BORROW DATE	RETURN DATE	STATUS
1	Nita ID: U-08	Obeng	Test	07 Apr 2026	07 Apr 2026	RETURNED
2	Nita ID: U-08	Yoke	Test	05 Mar 2026	09 Apr 2026	RETURNED
3	Nita ID: U-08	Obeng	25-301	12 Feb 2026	05 Mar 2026	RETURNED
4	Nita ID: U-08	Yoke	25-30	05 Feb 2026	12 Feb 2026	RETURNED
5	Yunita C ID: M-001	Yoke	25-30392	19 Dec 2025	19 Dec 2025	RETURNED

Gambar 4. 22 Implementasi UI *History User*

4.2. Pengujian *User Acceptance Testing* (UAT)

Pengujian *User Acceptance Testing* (UAT) dilakukan untuk memastikan sistem yang telah dikembangkan dapat diterima oleh pengguna akhir dan memenuhi seluruh kebutuhan yang telah disepakati. Fokus utama pengujian ini adalah mengevaluasi kemampuan sistem dalam menyelesaikan skenario operasional secara nyata.

4.2.1 Pengujian Perangkat Lunak (*Functional Testing*)

Pengujian perangkat lunak pada sistem ini dilakukan menggunakan metode *Black Box Testing*. Metode ini bertujuan untuk mengevaluasi fungsionalitas aplikasi berdasarkan masukan (*input*) dan keluaran (*output*) pada antarmuka, tanpa melihat struktur kode program di baliknya.

Pengujian ini bertujuan untuk membuktikan bahwa ke-17 (tujuh belas) Kebutuhan Fungsional (FR) yang telah dirancang dapat beroperasi secara normal, mampu menangani kesalahan (*error handling*), dan memberikan respons yang sesuai berdasarkan peran pengguna, baik Admin maupun *User*. Berikut adalah rincian hasil pengujian fungsionalitas sistem menggunakan metode *Black Box Testing* :

1. Pengujian Skenario *Use Case Login*

Pengujian *scenario use case* Login dilakukan berdasarkan *Use Case Scenario Login* (Tabel 4.1). *Use case scenario* mendefinisikan proses pengguna dalam melakukan *login* ke dalam sistem DALAZ Asset Management, namun pengujian dilakukan secara lebih rinci untuk memvalidasi setiap kondisi input serta respons sistem terhadap proses autentikasi pengguna. Hal ini bertujuan untuk memastikan mekanisme *login* berjalan sesuai dengan kebutuhan sistem dan skenario bisnis yang telah dirancang.

Pengujian fungsionalitas sistem, baik dari sisi Administrator maupun Pengguna, dilakukan oleh Angelia Dwi Shella selaku Lead QHSE & Legal PT Dalaz Teknik Utama, karena departemen tersebut merupakan divisi utama yang akan menggunakan dan mengelola sistem

ini. Seluruh hasil pengujian kemudian ditinjau langsung oleh M. Dzikri Fadilah selaku IT Supervisor sekaligus Pembimbing Magang untuk memastikan sistem beroperasi sesuai standar teknologi perusahaan.

Tabel 4. 1 *Blackbox Testing Login*

Test Case ID	Skenario	Luaran yang Diharapkan	Luaran yang Dihasilkan	Hasil	
				B	G
TC01	<i>Username dan Password valid, role Admin</i>	<i>Login berhasil divalidasi dan sistem mengarahkan pengguna ke Dashboard Admin</i>	<i>Login divalidasi dengan benar dan redirect ke Dashboard Admin berhasil</i>	✓	
TC02	<i>Username dan Password valid, role User</i>	<i>Login berhasil divalidasi dan sistem mengarahkan pengguna ke Dashboard User</i>	<i>Kredensial berhasil divalidasi dan sistem mengarahkan pengguna ke Dashboard User dengan tepat</i>	✓	
TC03	<i>Username benar, Password salah</i>	<i>Login tidak berhasil dan sistem menampilkan pesan error: "Invalid Username or password."</i>	<i>Sistem menolak akses dan menampilkan pesan error: "Invalid Username or password."</i>	✓	
TC04	<i>Username tidak terdaftar, Password benar/salah</i>	<i>Login tidak berhasil dan sistem menampilkan pesan error: "Invalid Username or password."</i>	<i>Sistem menolak akses dan menampilkan pesan error: "Invalid Username or password."</i>	✓	
TC05	<i>Username kosong, Password terisi</i>	<i>Login tidak berhasil dan sistem menampilkan pesan: "Username and password must be filled in."</i>	<i>Sistem mencegah proses login dan menampilkan pesan: "Username and password must be filled in."</i>	✓	
TC06	<i>Username terisi, Password kosong</i>	<i>Login tidak berhasil dan sistem menampilkan pesan: "Username and password must be filled in."</i>	<i>Sistem mencegah proses login dan menampilkan pesan: "Username and password must be filled in."</i>	✓	
TC07	<i>Seluruh field kosong</i>	<i>Login tidak berhasil dan sistem menampilkan pesan: "Username and password must be filled in."</i>	<i>Sistem mencegah proses login dan menampilkan pesan: "Username and password must be filled in."</i>		

Hasil Analisis :

1. Persentase *test case* yang lolos = $7/7 = 100\%$
2. Persentase *test case* yang gagal = $0/7 = 0\%$
3. Persentase *test case* yang telah diuji = $7/7 = 100\%$
4. Persentase *test case* yang belum diuji = $0/7 = 0\%$

Jumlah total *test case* dalam pengujian = 7

2. Pengujian Skenario *Use Case Dashboard*

Pengujian *scenario use case Dashboard* dilakukan berdasarkan *Use Case Scenario Dashboard* (Tabel 4.2). Pengujian ini bertujuan untuk memvalidasi bahwa informasi yang disajikan pada halaman *dashboard* sesuai dengan hak akses masing-masing pengguna, baik Admin maupun *User*, serta menguji respons sistem saat pangkalan data berisi data maupun saat dalam kondisi kosong.

Sama halnya dengan pengujian sebelumnya, proses validasi antarmuka *dashboard* ini dilakukan secara langsung oleh Angelia Dwi Shella selaku *Lead QHSE & Legal*. Pengujian dilakukan dari kedua sisi hak akses (*Admin* dan *User*) untuk memastikan bahwa data ringkasan yang disajikan benar-benar relevan dan membantu kebutuhan pantauan harian departemen tersebut. Seluruh hasil dari pengujian ini kemudian ditinjau (*review*) oleh M. Dzikri Fadilah selaku *IT Supervisor* PT Dalaz Teknik Utama guna memastikan sistem berjalan dengan stabil dan sesuai spesifikasi.

Tabel 4. 2 Blackbox Testing Dashboard

Test Case ID	Skenario	Luaran yang Diharapkan	Luaran yang Dihasilkan	Hasil	
				B	G
TC01	<i>Dashboard</i> Admin ditampilkan dengan data tersedia	Sistem berhasil menampilkan seluruh <i>widget</i> statistik dan daftar pemantauan aset (<i>Calibration Due Date Reminder, Equipment</i>)	Tampilan <i>dashboard</i> Admin berhasil memuat <i>widget</i> dan rincian pemantauan aset secara akurat dan mutakhir	✓	

		<i>Currently Borrowed</i> , <i>Inventory Minimum Order</i> , dan <i>Equipment Damage Report</i>) dengan angka yang akurat dan menyesuaikan kondisi data terbaru			
TC02	<i>Dashboard Admin</i> ditampilkan saat data kosong	Sistem menampilkan angka 0 pada seluruh <i>widget</i> statistik dan memunculkan pesan " <i>No asset data available</i> " pada bagian daftar informasi	Sistem berhasil menangani kondisi data kosong dengan menampilkan pesan " <i>No asset data available</i> "	✓	
TC03	<i>Dashboard User</i> ditampilkan dengan riwayat transaksi	Sistem menampilkan halaman <i>dashboard User</i> yang memuat <i>Notifications</i> , <i>Recent History</i> , <i>Upcoming Returns</i> , dan <i>Popular Assets</i> sesuai dengan riwayat aktivitas pengguna	Tampilan <i>dashboard User</i> berhasil memuat rekam jejak personal (<i>Recent History</i> dan <i>Upcoming Returns</i>) dengan tepat	✓	
TC04	<i>Dashboard User</i> ditampilkan pada akun baru tanpa riwayat transaksi	Sistem merespons kondisi ketiadaan riwayat transaksi dengan memunculkan pesan " <i>No borrowing activity yet</i> " pada layar pengguna	Sistem berhasil menangani akun <i>User</i> baru dengan menampilkan pesan " <i>No borrowing activity yet</i> "	✓	

Hasil Analisis:

1. Persentase *test case* yang lolos = $4/4 = 100\%$
2. Persentase *test case* yang gagal = $0/4 = 0\%$
3. Persentase *test case* yang telah diuji = $4/4 = 100\%$
4. Persentase *test case* yang belum diuji = $0/4 = 0\%$

Jumlah total *test case* dalam pengujian = 4

3. Pengujian Skenario *Use Case* Mengelola Data Aset Statis

Pengujian fungsionalitas pengelolaan data aset (kategori *Office Equipment*, *Vehicle*, *Land & Building*, dan *Intangible Assets*) dilakukan secara lebih rinci untuk memvalidasi setiap kondisi masukan (input) serta respons sistem terhadap aksi Admin. Pengujian ini bertujuan untuk

memastikan mekanisme penambahan manual, impor CSV, pengubahan, penghapusan, pencarian, aturan pembuatan *QR Code* spesifik kategori, serta validasi data ganda berjalan sesuai dengan kebutuhan sistem dan spesifikasi yang telah dirancang.

Mengingat kelengkapan dan keakuratan pendataan aset ini sangat penting bagi perusahaan, pengujian fungsi-fungsi *Admin* tersebut dilakukan secara langsung oleh Angelia Dwi Shella selaku *Lead QHSE & Legal*. Keterlibatan ini memastikan bahwa sistem pencatatan yang dibangun benar-benar sesuai dengan alur kerja departemen terkait. Selanjutnya, seluruh hasil pengujian ini ditinjau (*review*) oleh M. Dzikri Fadilah selaku *IT Supervisor* sekaligus Pembimbing Magang di PT Dalaz Teknik Utama, guna menjamin bahwa pengelolaan data berjalan dengan stabil, aman, dan memenuhi standar teknologi perusahaan.

Tabel 4. 3 Blackbox Testing Mengelola Data Aset Statis

Test Case ID	Skenario	Luaran yang Diharapkan	Luaran yang Dihasilkan	Hasil	
				B	G
TC01	Admin menambah data <i>Office Equipment</i> secara manual.	Data aset baru tersimpan, sistem mencetak <i>QR Code</i> , dan menampilkan pesan " <i>Asset Added successfully</i> ".	Data berhasil ditambahkan dan <i>QR Code</i> tercetak secara otomatis.	✓	
TC02	Admin menambah data <i>Office Equipment</i> melalui impor CSV.	Data dari CSV tersimpan, sistem mencetak <i>QR Code</i> , dan menampilkan pesan " <i>Asset Added successfully</i> ".	Data massal berhasil diimpor dan <i>QR Code</i> tercetak secara otomatis.	✓	
TC03	Admin mengubah (<i>Edit</i>) kuantitas <i>Office Equipment</i> .	Kuantitas berhasil diperbarui dan sistem menampilkan pesan " <i>Data updated successfully</i> ".	Kuantitas berhasil diperbarui dan notifikasi sukses muncul.	✓	
TC04	Admin menghapus data <i>Office Equipment</i> .	Data aset terhapus secara permanen dan sistem menampilkan pesan " <i>Data deleted successfully</i> ".	Data berhasil dihapus dari basis data dengan aman.	✓	

TC05	Admin menambah data <i>Land & Building</i> secara manual.	Data aset baru tersimpan, sistem tidak mencetak <i>QR Code</i> , dan menampilkan pesan " <i>Asset Added successfully</i> ".	Data berhasil ditambahkan tanpa ada pencetakan <i>QR Code</i> .	✓	
TC06	Admin menambah data <i>Land & Building</i> melalui impor CSV.	Data dari CSV tersimpan, sistem tidak mencetak <i>QR Code</i> , dan menampilkan pesan " <i>Asset Added successfully</i> ".	Data massal berhasil diimpor tanpa pencetakan <i>QR Code</i> .	✓	
TC07	Admin mengubah (<i>Edit</i>) lokasi <i>Land & Building</i> .	Lokasi berhasil diperbarui dan sistem menampilkan pesan " <i>Data updated successfully</i> ".	Lokasi berhasil diperbarui dan notifikasi sukses muncul.	✓	
TC08	Admin menghapus data <i>Land & Building</i> .	Data aset terhapus secara permanen dan sistem menampilkan pesan " <i>Data deleted successfully</i> ".	Data berhasil dihapus dari basis data dengan aman.	✓	
TC09	Admin menambah data <i>Vehicle</i> secara manual.	Data aset baru tersimpan, sistem tidak mencetak <i>QR Code</i> , dan menampilkan pesan " <i>Asset Added successfully</i> ".	Data berhasil ditambahkan tanpa ada pencetakan <i>QR Code</i> .	✓	
TC10	Admin menambah data <i>Vehicle</i> melalui impor CSV.	Data dari CSV tersimpan, sistem tidak mencetak <i>QR Code</i> , dan menampilkan pesan " <i>Asset Added successfully</i> ".	Data massal berhasil diimpor tanpa pencetakan <i>QR Code</i> .	✓	
TC11	Admin mengubah (<i>Edit</i>) tipe kendaraan <i>Vehicle</i> .	Tipe kendaraan diperbarui dan sistem menampilkan pesan " <i>Data updated successfully</i> ".	Tipe kendaraan berhasil diperbarui dan notifikasi sukses muncul.	✓	
TC12	Admin menghapus data <i>Vehicle</i> .	Data aset terhapus secara permanen dan sistem menampilkan pesan " <i>Data deleted successfully</i> ".	Data berhasil dihapus dari basis data dengan aman.	✓	
TC13	Admin menambah data <i>Intangible Assets</i> secara manual.	Data aset baru tersimpan, sistem tidak mencetak <i>QR Code</i> , dan menampilkan pesan " <i>Asset Added successfully</i> ".	Data berhasil ditambahkan tanpa ada pencetakan <i>QR Code</i> .	✓	
TC14	Admin menambah data <i>Intangible Assets</i> melalui impor CSV.	Data dari CSV tersimpan, sistem tidak mencetak <i>QR Code</i> , dan menampilkan pesan " <i>Asset Added</i>	Data massal berhasil diimpor tanpa pencetakan <i>QR Code</i> .	✓	

		<i>successfully</i> ".			
TC15	Admin mengubah (<i>Edit</i>) status dokumen <i>Intangible Assets</i> .	Status dokumen diperbarui dan sistem menampilkan pesan " <i>Data updated successfully</i> ".	Status dokumen berhasil diperbarui dan notifikasi sukses muncul.	✓	
TC16	Admin menghapus data <i>Intangible Assets</i> .	Data aset terhapus secara permanen dan sistem menampilkan pesan " <i>Data deleted successfully</i> ".	Data berhasil dihapus dari basis data dengan aman.	✓	
TC17	Admin menambah <i>Office Equipment</i> dengan mengosongkan <i>Asset Name</i> .	Sistem menahan simpan data dan menampilkan pesan " <i>All fields are required</i> ".	Validasi berhasil memblokir instruksi simpan akibat kekosongan <i>Asset Name</i> .	✓	
TC18	Admin menambah <i>Office Equipment</i> dengan mengosongkan <i>Asset Code</i> .	Sistem menahan simpan data dan menampilkan pesan " <i>All fields are required</i> ".	Validasi berhasil memblokir instruksi simpan akibat kekosongan <i>Asset Code</i> .	✓	
TC19	Admin menambah <i>Office Equipment</i> dengan mengosongkan <i>Quantity</i> (opsional).	Sistem menyimpan aksi tambah dan menampilkan pesan " <i>Asset Added successfully</i> ".	Data berhasil disimpan (karena <i>Quantity</i> bersifat opsional).	✓	
TC20	Admin menambah <i>Land & Building</i> dengan mengosongkan <i>Asset Name</i> .	Sistem menahan simpan data dan menampilkan pesan " <i>All fields are required</i> ".	Validasi berhasil memblokir instruksi simpan akibat kekosongan <i>Asset Name</i> .	✓	
TC21	Admin menambah <i>Land & Building</i> dengan mengosongkan <i>Asset Code</i> .	Sistem menahan simpan data dan menampilkan pesan " <i>All fields are required</i> ".	Validasi berhasil memblokir instruksi simpan akibat kekosongan <i>Asset Code</i> .	✓	
TC22	Admin menambah <i>Land & Building</i> dengan mengosongkan <i>Location</i> (opsional).	Sistem menyimpan aksi tambah dan menampilkan pesan " <i>Asset Added successfully</i> ".	Data berhasil disimpan (karena <i>Location</i> bersifat opsional).	✓	
TC23	Admin menambah <i>Vehicle</i> dengan mengosongkan <i>Asset Name</i> .	Sistem menahan simpan data dan menampilkan pesan " <i>All fields are required</i> ".	Validasi berhasil memblokir instruksi simpan akibat kekosongan <i>Asset Name</i> .	✓	
TC24	Admin menambah <i>Vehicle</i> dengan mengosongkan <i>Asset Code</i> .	Sistem menahan simpan data dan menampilkan pesan " <i>All fields are required</i> ".	Validasi berhasil memblokir instruksi simpan akibat kekosongan <i>Asset Code</i> .	✓	
TC25	Admin menambah <i>Vehicle</i> dengan mengosongkan <i>Vehicle</i>	Sistem menyimpan aksi tambah dan menampilkan pesan " <i>Asset</i>	Data berhasil disimpan (karena <i>Vehicle Type</i>	✓	

	Type (opsional).	Added successfully".	bersifat opsional).		
TC26	Admin menambah <i>Intangible Assets</i> dengan mengosongkan <i>Asset Name</i> .	Sistem menahan simpan data dan menampilkan pesan " <i>All fields are required</i> ".	Validasi berhasil memblokir instruksi simpan akibat kekosongan <i>Asset Name</i> .	✓	
TC27	Admin menambah <i>Intangible Assets</i> dengan mengosongkan <i>Asset Code</i> .	Sistem menahan simpan data dan menampilkan pesan " <i>All fields are required</i> ".	Validasi berhasil memblokir instruksi simpan akibat kekosongan <i>Asset Code</i> .	✓	
TC28	Admin menambah <i>Intangible Assets</i> dengan mengosongkan <i>Document Status</i> (opsional).	Sistem menyimpan aksi tambah dan menampilkan pesan " <i>Asset Added successfully</i> ".	Data berhasil disimpan (karena <i>Document Status</i> bersifat opsional).	✓	
TC29	Admin memasukkan format huruf pada kolom <i>Quantity</i> .	Sistem menolak huruf dan menampilkan pesan " <i>Quantity must be a number</i> ".	Validasi format kolom angka berfungsi dengan baik.	✓	
TC30	Admin mengunggah file impor CSV dengan format tidak valid (contoh: .pdf atau .xlsx).	Sistem menolak dokumen dan menampilkan pesan " <i>Invalid file format, please upload a CSV file</i> ".	Validasi ekstensi unggahan berhasil menolak file non-CSV.	✓	
TC31	Admin menambah data <i>Office Equipment</i> dengan <i>Asset Name</i> yang duplikat.	Sistem menyimpan aksi tambah dan menampilkan pesan " <i>Asset Added successfully</i> ".	Data berhasil disimpan (Nama aset diizinkan duplikat).	✓	
TC32	Admin menambah data <i>Land & Building</i> dengan <i>Asset Name</i> yang duplikat.	Sistem menyimpan aksi tambah dan menampilkan pesan " <i>Asset Added successfully</i> ".	Data berhasil disimpan (Nama aset diizinkan duplikat).	✓	
TC33	Admin menambah data <i>Vehicle</i> dengan <i>Asset Name</i> yang duplikat.	Sistem menyimpan aksi tambah dan menampilkan pesan " <i>Asset Added successfully</i> ".	Data berhasil disimpan (Nama aset diizinkan duplikat).	✓	
TC34	Admin menambah data <i>Intangible Assets</i> dengan <i>Asset Name</i> yang duplikat.	Sistem menyimpan aksi tambah dan menampilkan pesan " <i>Asset Added successfully</i> ".	Data berhasil disimpan (Nama aset diizinkan duplikat).	✓	
TC35	Admin mengubah (<i>Edit</i>) <i>Asset Name Office Equipment</i> menjadi nama yang duplikat.	Sistem menyimpan aksi pembaruan (<i>edit</i>) dan menampilkan pesan " <i>Asset updated successfully</i> ".	Data berhasil diperbarui (Nama aset diizinkan duplikat).	✓	
TC36	Admin mengubah (<i>Edit</i>) <i>Asset Name Land & Building</i> menjadi nama yang duplikat.	Sistem menyimpan aksi pembaruan (<i>edit</i>) dan menampilkan pesan " <i>Asset updated successfully</i> ".	Data berhasil diperbarui (Nama aset diizinkan duplikat).	✓	

TC37	Admin mengubah (<i>Edit</i>) <i>Asset Name Vehicle</i> menjadi nama yang duplikat.	Sistem menyimpan aksi pembaruan (<i>edit</i>) dan menampilkan pesan " <i>Asset updated successfully</i> ".	Data berhasil diperbarui (Nama aset diizinkan duplikat).	✓	
TC38	Admin mengubah (<i>Edit</i>) <i>Asset Name Intangible Assets</i> menjadi nama yang duplikat.	Sistem menyimpan aksi pembaruan (<i>edit</i>) dan menampilkan pesan " <i>Asset updated successfully</i> ".	Data berhasil diperbarui (Nama aset diizinkan duplikat).	✓	
TC39	Admin menambah data <i>Office Equipment</i> dengan <i>Asset Code</i> yang sudah terdaftar.	Sistem menolak aksi tambah dan menampilkan pesan " <i>Asset Code already exists</i> ".	Deteksi <i>Asset Code</i> duplikat berhasil pada form tambah <i>Office Equipment</i> .	✓	
TC40	Admin menambah data <i>Land & Building</i> dengan <i>Asset Code</i> yang sudah terdaftar.	Sistem menolak aksi tambah dan menampilkan pesan " <i>Asset Code already exists</i> ".	Deteksi <i>Asset Code</i> duplikat berhasil pada form tambah <i>Land & Building</i> .	✓	
TC41	Admin menambah data <i>Vehicle</i> dengan <i>Asset Code</i> yang sudah terdaftar.	Sistem menolak aksi tambah dan menampilkan pesan " <i>Asset Code already exists</i> ".	Deteksi <i>Asset Code</i> duplikat berhasil pada form tambah <i>Vehicle</i> .	✓	
TC42	Admin menambah data <i>Intangible Assets</i> dengan <i>Asset Code</i> yang sudah terdaftar.	Sistem menolak aksi tambah dan menampilkan pesan " <i>Asset Code already exists</i> ".	Deteksi <i>Asset Code</i> duplikat berhasil pada form tambah <i>Intangible Assets</i> .	✓	
TC43	Admin mengubah (<i>Edit</i>) <i>Asset Code Office Equipment</i> menjadi kode yang sudah terdaftar.	Sistem menolak aksi pembaruan (<i>edit</i>) dan menampilkan pesan " <i>Asset Code already exists</i> ".	Deteksi <i>Asset Code</i> duplikat berhasil pada form <i>edit Office Equipment</i> .	✓	
TC44	Admin mengubah (<i>Edit</i>) <i>Asset Code Land & Building</i> menjadi kode yang sudah terdaftar.	Sistem menolak aksi pembaruan (<i>edit</i>) dan menampilkan pesan " <i>Asset Code already exists</i> ".	Deteksi <i>Asset Code</i> duplikat berhasil pada form <i>edit Land & Building</i> .	✓	
TC45	Admin mengubah (<i>Edit</i>) <i>Asset Code Vehicle</i> menjadi kode yang sudah terdaftar.	Sistem menolak aksi pembaruan (<i>edit</i>) dan menampilkan pesan " <i>Asset Code already exists</i> ".	Deteksi <i>Asset Code</i> duplikat berhasil pada form <i>edit Vehicle</i> .	✓	
TC46	Admin mengubah (<i>Edit</i>) <i>Asset Code Intangible Assets</i> menjadi kode yang sudah terdaftar.	Sistem menolak aksi pembaruan (<i>edit</i>) dan menampilkan pesan " <i>Asset Code already exists</i> ".	Deteksi <i>Asset Code</i> duplikat berhasil pada form <i>edit Intangible Assets</i> .	✓	
TC47	Admin mengimpor CSV <i>Office Equipment</i> yang berisi <i>Asset Code</i> yang duplikat.	Sistem menolak baris data di CSV dan menampilkan keterangan gagal karena " <i>Asset Code already exists</i> ".	Deteksi <i>Asset Code</i> duplikat berhasil pada baris impor CSV <i>Office Equipment</i> .	✓	

TC48	Admin mengimpor CSV <i>Land & Building</i> yang berisi <i>Asset Code</i> yang duplikat.	Sistem menolak baris data di CSV dan menampilkan keterangan gagal karena " <i>Asset Code already exists</i> ".	Deteksi <i>Asset Code</i> duplikat berhasil pada baris impor CSV <i>Land & Building</i> .	✓	
TC49	Admin mengimpor CSV <i>Vehicle</i> yang berisi <i>Asset Code</i> yang duplikat.	Sistem menolak baris data di CSV dan menampilkan keterangan gagal karena " <i>Asset Code already exists</i> ".	Deteksi <i>Asset Code</i> duplikat berhasil pada baris impor CSV <i>Vehicle</i> .	✓	
TC50	Admin mengimpor CSV <i>Intangible Assets</i> yang berisi <i>Asset Code</i> yang duplikat.	Sistem menolak baris data di CSV dan menampilkan keterangan gagal karena " <i>Asset Code already exists</i> ".	Deteksi <i>Asset Code</i> duplikat berhasil pada baris impor CSV <i>Intangible Assets</i> .	✓	
TC51	Admin melakukan pencarian berdasarkan <i>Asset Name</i> yang valid.	Sistem menyaring tabel dan hanya menampilkan baris data aset yang memuat kata kunci pencarian pada kolom <i>Asset Name</i> .	Tabel secara otomatis menyaring baris data sesuai huruf atau kata yang diketikkan pada kolom pencarian tanpa perlu memuat ulang halaman.	✓	
TC52	Admin melakukan pencarian berdasarkan <i>Asset Code</i> yang valid.	Sistem menyaring tabel dan hanya menampilkan baris data aset yang memiliki kata kunci pencarian pada kolom <i>Asset Code</i> .	Tabel secara otomatis menyaring baris data sesuai huruf atau kata yang diketikkan pada kolom pencarian tanpa perlu memuat ulang halaman.	✓	
TC53	Admin melakukan pencarian dengan kata kunci yang tidak ada di <i>database</i> .	Jika kata kunci tidak valid, sistem menampilkan tabel kosong atau memunculkan pesan "Data tidak ditemukan" (<i>No data available in table</i>).	Sistem mengosongkan baris tabel dan menampilkan teks " <i>No data available in table</i> " atau "Data tidak ditemukan".	✓	

Hasil Analisis:

1. Persentase *test case* yang lolos = $53/53 = 100\%$
2. Persentase *test case* yang gagal = $0/53 = 0\%$
3. Persentase *test case* yang telah diuji = $53/53 = 100\%$
4. Persentase *test case* yang belum diuji = $0/53 = 0\%$ Jumlah total *test case* dalam pengujian = 53

4. Pengujian Skenario Mengelola Data *Equipment*

Pengujian fungsionalitas pengelolaan master data peralatan kerja (*Equipment*) dilakukan secara menyeluruh berdasarkan 22 skenario uji. Pengujian ini difokuskan pada entitas yang menjadi pusat transaksi operasional, mencakup validasi penambahan data (manual & CSV), pembaruan (*edit*), penghapusan, fungsi pencarian, pembuatan *QR Code* otomatis, hingga pengujian mendalam terhadap 6 kondisi *field* kosong dan pencegahan duplikasi *Asset Number*.

Untuk memastikan sistem yang dibangun tepat sasaran dan mudah digunakan, tahap pengujian ini dieksekusi oleh Ibu Angelia Dwi Shella sebagai *Lead QHSE & Legal*, yakni pihak yang akan secara langsung mengelola data aset ini. Selanjutnya, seluruh hasil pengujian tersebut diperiksa dan dievaluasi oleh M. Dzikri Fadilah, selaku *IT Supervisor* sekaligus Pembimbing Magang di PT Dalaz Teknik Utama, guna menjamin kualitas dan keamanan sistem.

Tabel 4. 4 Blackbox Testing Mengelola Data *Equipment*

Test Case ID	Skenario	Luaran yang Diharapkan	Luaran yang Dihasilkan	Hasil	
				B	G
TC01	Admin menambah data <i>Equipment</i> secara manual dengan seluruh form terisi valid.	Data aset baru tersimpan, sistem mencetak <i>QR Code</i> otomatis, dan menampilkan pesan sukses.	Data berhasil ditambahkan dan <i>QR Code</i> tercetak secara otomatis.	✓	
TC02	Admin mengimpor data <i>Equipment</i> melalui unggahan berkas CSV berformat valid.	Data dari CSV tersimpan, sistem mencetak <i>QR Code</i> untuk setiap baris, dan menampilkan pesan sukses.	Data massal berhasil diimpor dan <i>QR Code</i> tercetak secara otomatis.	✓	
TC03	Admin mengubah (<i>Edit</i>) nilai pada kolom <i>Quantity</i> data yang sudah ada.	Kuantitas berhasil diperbarui dan sistem menampilkan notifikasi data berhasil diubah.	Kuantitas berhasil diperbarui dan notifikasi sukses muncul.	✓	
TC04	Admin mengubah (<i>Edit</i>) lokasi penempatan pada data <i>Equipment</i> .	Lokasi berhasil diperbarui pada pangkalan data dan tabel langsung menampilkan perubahan.	Lokasi berhasil diperbarui dan notifikasi sukses muncul.	✓	

TC05	Admin menghapus data <i>Equipment</i> dan menyetujui konfirmasi peringatan (<i>Yes</i>).	Data aset terhapus secara permanen dari pangkalan data beserta pesan suksesnya.	Data sukses tereksekusi dengan konfirmasi keamanan.	✓	
TC06	Admin menekan tombol hapus, namun memilih batal (<i>Cancel</i>) pada <i>pop-up</i> konfirmasi.	Sistem membatalkan aksi hapus dan data tetap aman berada di dalam tabel.	Aksi pembatalan berfungsi dengan baik menjaga keamanan data.	✓	
TC07	Admin menambah data dengan hanya mengisi <i>Equipment Name</i> dan <i>Asset Number</i> (kolom opsional dibiarkan kosong).	Sistem tetap memproses penyimpanan karena kebutuhan form wajib telah terpenuhi.	Sistem sukses menyimpan data yang hanya menyertakan <i>Equipment Name</i> dan <i>Asset Number</i> .	✓	
TC08	Admin mengubah data (<i>Edit</i>) dengan mengosongkan nilai pada kategori atau lokasi.	Sistem memperbarui rekaman data dengan mengosongkan parameter opsional tersebut.	Sistem sukses memperbarui data meskipun pengguna mengosongkan kategori atau lokasi opsional.	✓	
TC09	Admin mencoba menyimpan form tambah dengan mengosongkan <i>Equipment Name</i> .	Sistem menahan proses simpan dan memunculkan peringatan wajib isi (<i>required</i>).	Validasi sukses memblokir form tambah jika <i>Equipment Name</i> dikosongkan.	✓	
TC10	Admin mencoba menyimpan form tambah dengan mengosongkan <i>Asset Number</i> .	Sistem menahan proses simpan dan memunculkan peringatan wajib isi (<i>required</i>).	Validasi sukses memblokir form tambah jika <i>Asset Number</i> dikosongkan.	✓	
TC11	Admin mengubah data (<i>Edit</i>) dengan sengaja menghapus teks pada <i>Equipment Name</i> .	Sistem menahan proses edit dan meminta pengguna melengkapi data wajib.	Validasi sukses memblokir form edit yang kehilangan data <i>Equipment Name</i> .	✓	
TC12	Admin mengubah data (<i>Edit</i>) dengan sengaja menghapus teks pada <i>Asset Number</i> .	Sistem menahan proses <i>edit</i> dan meminta pengguna melengkapi data wajib.	Validasi sukses memblokir form <i>edit</i> yang kehilangan data <i>Asset Number</i> .	✓	
TC13	Admin memasukkan teks huruf (misal: "Lima") pada kolom angka (<i>Quantity</i>).	Sistem menolak format input dan memunculkan pesan peringatan tipe data.	Perlindungan tipe data (angka) pada kolom kuantitas berfungsi dengan akurat.	✓	
TC14	Admin mengunggah fail dengan ekstensi yang dilarang (misal: PDF atau Excel) pada fitur Import.	Sistem menolak unggahan dan meminta fail dalam format CSV yang benar.	Perlindungan ekstensi unggahan berhasil menolak dokumen PDF/Excel.	✓	

TC15	Admin menambah data baru menggunakan <i>Equipment Name</i> yang sama persis dengan alat lain.	Sistem mengizinkan dan menyimpan data (karena nama alat boleh sama/banyak).	Sistem sukses mematuhi aturan untuk mengizinkan kesamaan nama alat saat penambahan baru.	✓	
TC16	Admin mengubah nama alat menjadi sama dengan nama alat lain yang sudah ada.	Sistem menerima perubahan dan menyimpannya (nama duplikat diizinkan).	Sistem sukses mematuhi aturan untuk mengizinkan kesamaan nama alat saat pembaruan data.	✓	
TC17	Admin menambah data baru menggunakan <i>Asset Number</i> yang sudah terdaftar di sistem.	Sistem memblokir penyimpanan mutlak dengan pesan error " <i>Asset Number Already Exists</i> ".	Proteksi sukses memblokir upaya mendaftarkan nomor aset yang sudah eksis di tabel.	✓	
TC18	Admin mengubah <i>Asset Number</i> milik suatu alat menjadi <i>Asset Number</i> milik alat lain.	Sistem memblokir pembaruan data untuk mencegah data alat tertukar (<i>error duplication</i>).	Proteksi sukses memblokir upaya mengganti nomor aset menjadi milik alat lain.	✓	
TC19	Admin mengimpor berkas CSV yang mengandung <i>Asset Number</i> yang sudah ada di basis data.	Sistem menolak pemrosesan baris data tersebut dan menampilkan status gagal pada log impor.	Pengecekan CSV sukses menyaring dan memblokir baris data yang nomor asetnya duplikat.	✓	
TC20	Admin mencari data dengan mengetikkan nama alat yang valid pada kolom pencarian (<i>Search</i>).	Tabel langsung disaring untuk menampilkan baris yang mengandung nama alat tersebut.	Tabel secara otomatis menyaring baris data sesuai huruf atau kata yang diketikkan pada kolom pencarian tanpa perlu memuat ulang (<i>reload</i>) halaman.	✓	
TC21	Admin mencari data dengan mengetikkan nomor aset (<i>Asset Number</i>) yang spesifik.	Tabel langsung disaring dan hanya menampilkan satu baris data alat yang sesuai dengan nomor tersebut.	Tabel secara otomatis menyaring baris data sesuai huruf atau kata yang diketikkan pada kolom pencarian tanpa perlu memuat ulang (<i>reload</i>) halaman.	✓	
TC22	Admin mengetikkan kata kunci pencarian acak yang tidak ada di dalam database.	Tabel dikosongkan secara visual dan memunculkan keterangan bahwa data tidak ditemukan.	Sistem mengosongkan baris tabel dan menampilkan teks " <i>No data available in table</i> " atau "Data tidak ditemukan".	✓	

Hasil Analisis:

1. Persentase *test case* yang lolos = $22/22 = 100\%$

2. Persentase *test case* yang gagal = $0/22 = 0\%$
3. Persentase *test case* yang telah diuji = $22/22 = 100\%$
4. Persentase *test case* yang belum diuji = $0/22 = 0\%$ Jumlah total *test case* dalam pengujian = 22

5. Testing Skenario Mengelola Data *Inventory*

Pengujian fungsionalitas pengelolaan master data barang persediaan (*Inventory / Consumable*) dilakukan secara menyeluruh berdasarkan 18 skenario uji. Pengujian ini berfokus pada validasi penambahan data (manual dan CSV), pembaruan (*edit*), penghapusan, pencarian data, serta validasi ketat terhadap sistem penguncian kolom kuantitas (*Quantity*) untuk mencegah manipulasi stok secara manual di luar fitur transaksi. Selain itu, pengujian juga mencakup pencegahan form kosong dan penolakan duplikasi *Asset Code*.

Sebagai pihak yang berkepentingan langsung dengan pengelolaan stok harian, Angelia Dwi Shella selaku *Lead QHSE & Legal* bertindak sebagai eksekutor utama dalam pengujian modul ini. Hasil dari seluruh skenario pengujian tersebut kemudian diperiksa dan dievaluasi oleh M. Dzikri Fadilah, selaku *IT Supervisor* sekaligus Pembimbing Magang di PT Dalaz Teknik Utama, guna menjamin fungsi sistem terimplementasi dengan stabil dan aman.

Tabel 4. 5 Blackbox Testing Mengelola Data *Inventory*

Test Case ID	Skenario	Luaran yang Diharapkan	Luaran yang Dihasilkan	Hasil	
				B	G
TC01	Admin menambah data Inventory secara manual dengan seluruh <i>field</i> terisi lengkap.	Data barang baru tersimpan di basis data dan sistem menampilkan pesan sukses penambahan.	Data berhasil ditambahkan dan notifikasi " <i>Successfully added</i> " muncul.	✓	
TC02	Admin mengimpor data Inventory melalui unggahan berkas CSV berformat valid.	Data massal dari CSV tersimpan ke basis data dan sistem menampilkan pesan sukses impor.	Data massal berhasil diimpor dan notifikasi " <i>Successfully added</i> " muncul.	✓	

TC03	Admin mengubah (<i>Edit</i>) data <i>Inventory</i> dengan membiarkan <i>field</i> opsional (misal: <i>Location</i>) dikosongkan.	Sistem menerima pembaruan data dengan <i>field</i> kosong tersebut dan menampilkan pesan sukses.	Data berhasil diperbarui meskipun <i>field</i> opsional dikosongkan, notifikasi " <i>Successfully Updated</i> " muncul.	✓	
TC04	Admin menghapus data <i>Inventory</i> dan menyetujui peringatan konfirmasi (<i>Yes</i>).	Data barang terhapus secara permanen dari tabel dan basis data.	Data berhasil dihapus dari basis data dengan aman.	✓	
TC05	Admin membuka form pembaruan (<i>Edit</i>) dan mencoba mengubah nilai pada kolom <i>Quantity</i> .	Kolom <i>Quantity</i> dalam keadaan terkunci (<i>disabled / readonly</i>) dan tidak bisa diketik manual.	Aturan sistem berfungsi, kolom <i>Quantity</i> terbukti terkunci dan tidak bisa di- <i>edit</i> .	✓	
TC06	Admin mencoba menyimpan form tambah dengan sengaja mengosongkan <i>Asset Name</i> .	Sistem menahan proses penyimpanan dan memunculkan peringatan wajib isi (<i>required</i>).	Validasi berhasil memblokir instruksi simpan akibat kekosongan <i>Asset Name</i> .	✓	
TC07	Admin mencoba menyimpan form tambah dengan sengaja mengosongkan <i>Asset Code</i> .	Sistem menahan proses penyimpanan dan memunculkan peringatan wajib isi (<i>required</i>).	Validasi berhasil memblokir instruksi simpan akibat kekosongan <i>Asset Code</i> .	✓	
TC08	Admin mengubah data (<i>Edit</i>) dan sengaja menghapus teks pada <i>Asset Name</i> yang sudah ada.	Sistem menahan proses pembaruan dan meminta pengguna melengkapi nama barang.	Validasi berhasil memblokir instruksi pembaruan akibat <i>Asset Name</i> dihapus.	✓	
TC09	Admin mengubah data (<i>Edit</i>) dan sengaja menghapus teks pada <i>Asset Code</i> yang sudah ada.	Sistem menahan proses pembaruan dan meminta pengguna melengkapi kode barang.	Validasi berhasil memblokir instruksi pembaruan akibat <i>Asset Code</i> dihapus.	✓	
TC10	Admin mengunggah <i>file</i> dengan ekstensi yang tidak diizinkan (misal: dokumen PDF) pada menu Impor.	Sistem menolak proses impor dan menampilkan pesan kesalahan format fail.	Validasi ekstensi unggahan berhasil menolak <i>file</i> berformat PDF.	✓	
TC11	Admin menambah data baru menggunakan <i>Asset Name</i> yang sama persis dengan nama barang lain.	Sistem mengizinkan dan menyimpan data (karena nama barang stok bisa memiliki tipe/merek berbeda).	Data duplikat nama berhasil disimpan sesuai aturan sistem.	✓	

TC12	Admin mengubah nama barang (<i>Edit</i>) menjadi sama dengan nama barang lain yang sudah ada.	Sistem menerima perubahan dan tetap menyimpannya dengan aman.	Pembaruan data dengan nama duplikat berhasil disimpan sesuai aturan sistem.	✓	
TC13	Admin menambah data baru menggunakan <i>Asset Code</i> yang sudah terdaftar di sistem.	Sistem mutlak memblokir proses simpan dan memunculkan pesan error "Kode Sudah Terdaftar".	Deteksi <i>Asset Code</i> duplikat berhasil pada form tambah data.	✓	
TC14	Admin mengubah <i>Asset Code</i> suatu barang menjadi <i>Asset Code</i> milik barang lain yang sudah eksis.	Sistem memblokir pembaruan data untuk mencegah data inventaris tertukar.	Deteksi <i>Asset Code</i> duplikat berhasil pada form <i>Edit</i> data.	✓	
TC15	Admin mengimpor berkas CSV yang di dalamnya mengandung <i>Asset Code</i> yang sudah digunakan.	Sistem menolak baris data duplikat tersebut pada proses impor dan meloloskan yang valid.	Deteksi <i>Asset Code</i> duplikat berhasil menyaring dan memblokir baris impor CSV.	✓	
TC16	Admin mencari data dengan mengetikkan nama barang (<i>Asset Name</i>) pada kolom <i>Search</i> .	Tabel secara otomatis disaring untuk menampilkan baris yang mengandung nama barang tersebut.	Tabel secara otomatis menyaring baris data sesuai huruf atau kata yang diketikkan pada kolom pencarian tanpa perlu memuat ulang (<i>reload</i>) halaman.	✓	
TC17	Admin mencari data dengan mengetikkan kode unik barang (<i>Asset Code</i>) secara spesifik.	Tabel secara otomatis disaring dan hanya menampilkan baris data alat yang sesuai dengan kode itu.	Tabel secara otomatis menyaring baris data sesuai huruf atau kata yang diketikkan pada kolom pencarian tanpa perlu memuat ulang (<i>reload</i>) halaman.	✓	
TC18	Admin mengetikkan kata kunci pencarian acak yang tidak ada di dalam basis data.	Tabel dikosongkan secara visual dan memunculkan keterangan data tidak ditemukan.	Sistem mengosongkan baris tabel dan menampilkan teks " <i>No data available in table</i> " atau "Data tidak ditemukan".	✓	

Hasil Analisis:

1. Persentase *test case* yang lolos = $18/18 = 100\%$
2. Persentase *test case* yang gagal = $0/18 = 0\%$
3. Persentase *test case* yang telah diuji = $18/18 = 100\%$

4. Persentase *test case* yang belum diuji = $0/18 = 0\%$ Jumlah total *test case* dalam pengujian = 18

6. Pengujian Skenario Mengelola Data *Location*

Pengujian fungsionalitas pengelolaan data referensi Lokasi (*Location*) dilakukan berdasarkan 10 skenario uji. Pengujian ini sangat penting karena data lokasi menjadi acuan master (*master data*) bagi penempatan aset. Fokus pengujian mencakup penambahan lokasi, pembaruan (*edit*), penghapusan, fitur pencarian, serta validasi ketat terhadap form wajib dan pencegahan duplikasi nama lokasi agar tidak terjadi kebingungan penempatan aset di lapangan.

Untuk memastikan pencatatan lokasi sesuai dengan kebutuhan operasional, proses pengujian dieksekusi secara langsung oleh Angelia Dwi Shella selaku *Lead QHSE & Legal*. Seluruh hasil pengujian ini kemudian dievaluasi dan ditinjau oleh M. Dzikri Fadilah selaku *IT Supervisor* sekaligus Pembimbing Magang di PT Dalaz Teknik Utama, guna menjamin bahwa sistem pengelolaan referensi lokasi telah berjalan stabil dan terintegrasi dengan baik.

Tabel 4. 6 Blackbox Testing Mengelola Data *Location*

<i>Test Case ID</i>	Skenario	Luaran yang Diharapkan	Luaran yang Dihasilkan	Hasil	
				B	G
TC01	Admin menambah data lokasi baru secara manual dengan <i>Location Name</i> dan <i>Description</i> yang valid.	Data lokasi baru tersimpan di basis data dan sistem menampilkan pesan sukses.	Data berhasil ditambahkan dan notifikasi sukses muncul.	✓	
TC02	Admin mengubah (<i>Edit</i>) informasi <i>Description</i> pada data lokasi yang sudah ada.	Deskripsi lokasi berhasil diperbarui dan sistem menampilkan pesan sukses.	Data berhasil diperbarui dan notifikasi sukses muncul.	✓	
TC03	Admin menghapus data lokasi dari tabel dan menyetujui konfirmasi penghapusan.	Data lokasi terhapus secara permanen dan sistem menampilkan pesan sukses.	Data berhasil dihapus secara permanen dengan aman.	✓	

TC04	Admin mencoba menyimpan form tambah lokasi dengan sengaja mengosongkan <i>Location Name</i> .	Sistem menahan proses penyimpanan dan memunculkan pesan " <i>Location Name is required</i> ".	Validasi berhasil memblokir instruksi simpan akibat kekosongan <i>Location Name</i> .	✓	
TC05	Admin menambah lokasi baru dengan mengisi <i>Location Name</i> namun mengosongkan <i>Description</i> (opsional).	Sistem memproses penyimpanan data dan menampilkan pesan sukses.	Data berhasil disimpan dengan <i>field</i> opsional dibiarkan kosong.	✓	
TC06	Admin menambah lokasi baru menggunakan <i>Location Name</i> yang sudah terdaftar sebelumnya di sistem.	Sistem mutlak memblokir proses simpan dan menampilkan pesan " <i>Location Name already exists</i> ".	Sistem memblokir duplikasi nama lokasi pada form tambah data.	✓	
TC07	Admin mengubah (<i>Edit</i>) <i>Location Name</i> suatu lokasi menjadi nama lokasi lain yang sudah ada.	Sistem memblokir pembaruan data dan menampilkan pesan " <i>Location Name already exists</i> ".	Sistem memblokir duplikasi nama lokasi pada form <i>edit</i> data.	✓	
TC08	Admin mencari data dengan mengetikkan nama lokasi (<i>Location Name</i>) yang valid pada kolom <i>Search</i> .	Tabel secara otomatis disaring untuk menampilkan baris yang mengandung nama lokasi tersebut.	Tabel otomatis menyaring baris data sesuai huruf atau kata yang diketikkan tanpa perlu <i>reload</i> .	✓	
TC09	Admin mencari data dengan mengetikkan deskripsi (<i>Description</i>) lokasi yang valid pada kolom <i>Search</i> .	Tabel secara otomatis disaring untuk menampilkan baris yang mengandung deskripsi tersebut.	Tabel otomatis menyaring baris data sesuai huruf atau kata yang diketikkan tanpa perlu <i>reload</i> .	✓	
TC10	Admin mengetikkan kata kunci pencarian acak yang tidak ada di dalam pangkalan data.	Tabel dikosongkan secara visual dan memunculkan pesan " <i>No data available in table</i> ".	Sistem mengosongkan baris tabel dan menampilkan teks " <i>No data available in table</i> ".	✓	

Hasil Analisis:

1. Persentase *test case* yang lolos = $10/10 = 100\%$
2. Persentase *test case* yang gagal = $0/10 = 0\%$
3. Persentase *test case* yang telah diuji = $10/10 = 100\%$
4. Persentase *test case* yang belum diuji = $0/10 = 0\%$ Jumlah total *test case* dalam pengujian = 10

7. Pengujian Skenario Mengelola Data *Category*

Pengujian fungsionalitas pengelolaan data referensi Kategori (*Category*) dilakukan berdasarkan 10 skenario uji. Sama halnya dengan data lokasi, kategori merupakan *master data* yang sangat esensial karena digunakan untuk mengelompokkan jenis aset di seluruh sistem. Pengujian ini berfokus pada validasi kelancaran penambahan data, pembaruan (*edit*), penghapusan, fungsi pencarian, serta pengujian ketat terhadap validasi form wajib dan pencegahan duplikasi nama kategori guna menjaga konsistensi pangkalan data.

Pengujian ini dilaksanakan secara langsung oleh Angelia Dwi Shella selaku *Lead QHSE & Legal* untuk memastikan pengelompokan aset sesuai dengan kebutuhan manajemen di lapangan. Seluruh hasil pengujian kemudian ditinjau dan dievaluasi oleh M. Dzikri Fadilah selaku *IT Supervisor* sekaligus Pembimbing Magang di PT Dalaz Teknik Utama, sebagai langkah verifikasi akhir agar sistem yang dikembangkan selaras dengan standar operasional perusahaan.

Tabel 4. 7 Blackbox Testing Mengelola Data *Category*

Test Case ID	Skenario	Luaran yang Diharapkan	Luaran yang Dihasilkan	Hasil	
				B	G
TC01	Admin menambah data kategori baru secara manual dengan <i>Category Name</i> dan <i>Description</i> yang valid.	Data kategori baru tersimpan di basis data dan sistem menampilkan pesan sukses penambahan.	Data berhasil ditambahkan dan notifikasi sukses muncul.	✓	
TC02	Admin mengubah (<i>Edit</i>) informasi <i>Description</i> pada data kategori yang sudah ada.	Deskripsi kategori berhasil diperbarui dan sistem menampilkan pesan sukses.	Data berhasil diperbarui dan notifikasi sukses muncul.	✓	
TC03	Admin menghapus data kategori dari tabel dan menyetujui peringatan konfirmasi (<i>Yes</i>).	Data kategori terhapus secara permanen dari basis data dan sistem menampilkan pesan sukses.	Data berhasil dihapus secara permanen dengan aman.	✓	
TC04	Admin mencoba menyimpan form tambah kategori dengan sengaja mengosongkan <i>Category</i>	Sistem menahan proses penyimpanan dan memunculkan pesan	Validasi berhasil memblokir instruksi simpan akibat kekosongan <i>Category Name</i> .	✓	

	<i>Name.</i>	"Category Name is required".			
TC05	Admin menambah kategori baru dengan mengisi <i>Category Name</i> namun mengosongkan <i>Description</i> (opsional).	Sistem memproses penyimpanan data dan menampilkan pesan sukses karena <i>Description</i> opsional.	Data berhasil disimpan dengan <i>field</i> opsional dibiarkan kosong.	✓	
TC06	Admin menambah kategori baru menggunakan <i>Category Name</i> yang sudah terdaftar sebelumnya di sistem.	Sistem mutlak memblokir proses simpan dan menampilkan pesan " <i>Category Name already exists</i> ".	Sistem memblokir duplikasi nama kategori pada form tambah data.	✓	
TC07	Admin mengubah (<i>Edit</i>) <i>Category Name</i> suatu kategori menjadi nama kategori lain yang sudah eksis.	Sistem memblokir pembaruan data dan menampilkan pesan " <i>Category Name already exists</i> ".	Sistem memblokir duplikasi nama kategori pada <i>form</i> edit data.	✓	
TC08	Admin mencari data dengan mengetikkan nama kategori (<i>Category Name</i>) yang valid pada kolom pencarian.	Tabel secara otomatis disaring untuk menampilkan baris yang mengandung nama kategori tersebut.	Tabel otomatis menyaring baris data sesuai huruf atau kata yang diketikkan tanpa perlu <i>reload</i> .	✓	
TC09	Admin mencari data dengan mengetikkan deskripsi (<i>Description</i>) kategori yang valid pada kolom pencarian.	Tabel secara otomatis disaring untuk menampilkan baris yang mengandung deskripsi tersebut.	Tabel otomatis menyaring baris data sesuai huruf atau kata yang diketikkan tanpa perlu <i>reload</i> .	✓	
TC10	Admin mengetikkan kata kunci pencarian acak yang tidak ada di dalam pangkalan data.	Tabel dikosongkan secara visual dan memunculkan pesan " <i>No data available in table</i> ".	Sistem mengosongkan baris tabel dan menampilkan teks " <i>No data available in table</i> ".	✓	

Hasil Analisis:

1. Persentase *test case* yang lolos = $10/10 = 100\%$
2. Persentase *test case* yang gagal = $0/10 = 0\%$
3. Persentase *test case* yang telah diuji = $10/10 = 100\%$
4. Persentase *test case* yang belum diuji = $0/10 = 0\%$ Jumlah total *test case* dalam pengujian = 10

8. Pengujian Skenario Mengelola Data *User*

Pengujian fungsionalitas pengelolaan data pengguna (*User*) dilakukan secara menyeluruh berdasarkan 15 skenario uji. Pengujian ini berfokus pada

validasi kelancaran penambahan data pengguna (baik manual maupun impor CSV) beserta pembuatan *QR Code* otomatis yang berfungsi sebagai identitas karyawan saat meminjam barang. Selain itu, pengujian juga mencakup proses pembaruan (*edit*), penghapusan, fitur pencarian, serta validasi ketat terhadap form wajib isi dan pencegahan duplikasi ID Karyawan maupun *Username* guna menjaga keamanan akses sistem.

Untuk memastikan fungsionalitas ini praktis digunakan dalam pendataan karyawan, pengujian dilakukan langsung oleh Angelia Dwi Shella selaku *Lead QHSE & Legal*. Hasil dari pengujian tersebut selanjutnya ditinjau oleh M. Dzikri Fadilah sebagai *IT Supervisor* dan Pembimbing Magang, sebagai langkah verifikasi akhir sebelum sistem diterapkan pada lingkungan operasional perusahaan.

Tabel 4. 8 Blackbox Testing Mengelola Data User

Test Case ID	Skenario	Luaran yang Diharapkan	Luaran yang Dihasilkan	Hasil	
				B	G
TC01	Admin menambah data <i>User</i> baru secara manual dengan data (Nama, ID, <i>Role</i> , <i>Username</i> , <i>Password</i>) yang valid.	Data pengguna tersimpan, sistem mencetak <i>QR Code</i> secara otomatis, dan menampilkan pesan sukses.	Data berhasil ditambahkan dan <i>QR Code</i> tercetak secara otomatis.	✓	
TC02	Admin mengimpor data <i>User</i> melalui unggahan berkas CSV berformat valid.	Data massal tersimpan, sistem mencetak <i>QR Code</i> untuk tiap pengguna, dan menampilkan pesan sukses.	Data massal berhasil diimpor dan <i>QR Code</i> tercetak secara otomatis.	✓	
TC03	Admin mengubah (<i>Edit</i>) informasi pengguna (misal: penempatan departemen atau hak akses/ <i>Role</i>).	Data pengguna berhasil diperbarui dan sistem menampilkan pesan sukses pembaruan.	Data berhasil diperbarui dan notifikasi sukses muncul.	✓	
TC04	Admin menghapus data pengguna dari tabel dan menyetujui konfirmasi penghapusan.	Data pengguna terhapus secara permanen dari basis data dan sistem menampilkan pesan sukses.	Data berhasil dihapus secara permanen dengan aman.	✓	

TC05	Admin mencoba menyimpan form tambah pengguna dengan mengosongkan Nama Karyawan (<i>Name</i>).	Sistem menahan proses penyimpanan dan memunculkan peringatan wajib isi (<i>required</i>).	Validasi berhasil memblokir instruksi simpan akibat kekosongan Nama.	✓	
TC06	Admin mencoba menyimpan form tambah pengguna dengan mengosongkan ID Karyawan .	Sistem menahan proses penyimpanan dan memunculkan peringatan wajib isi (<i>required</i>).	Validasi berhasil memblokir instruksi simpan akibat kekosongan ID Karyawan.	✓	
TC07	Admin mencoba menyimpan form tambah pengguna dengan mengosongkan <i>Username</i> atau <i>Password</i> .	Sistem menahan proses penyimpanan untuk mencegah pembuatan akun tanpa akses masuk.	Validasi berhasil memblokir instruksi simpan akibat kekosongan <i>Username/Password</i> .	✓	
TC08	Admin mencoba menyimpan form dengan mengosongkan <i>Role</i> (hak akses Admin/ <i>User</i>).	Sistem menahan proses penyimpanan karena <i>Role</i> menentukan jenis <i>dashboard</i> pengguna.	Validasi berhasil memblokir instruksi simpan akibat kekosongan <i>Role</i> .	✓	
TC09	Admin menambah data pengguna baru dengan menggunakan ID Karyawan yang sudah terdaftar.	Sistem mutlak memblokir proses simpan dan menampilkan pesan <i>error</i> "ID Karyawan sudah terdaftar".	Sistem memblokir duplikasi ID Karyawan pada form tambah data.	✓	
TC10	Admin menambah pengguna baru dengan <i>Username</i> yang sudah digunakan oleh akun lain.	Sistem mutlak memblokir proses simpan dan menampilkan pesan <i>error</i> " <i>Username</i> sudah digunakan".	Sistem memblokir duplikasi <i>Username</i> pada form tambah data.	✓	
TC11	Admin mengubah (<i>Edit</i>) ID Karyawan atau <i>Username</i> menjadi milik pengguna lain yang sudah ada.	Sistem memblokir pembaruan data dan menampilkan pesan peringatan duplikasi.	Sistem memblokir duplikasi data identitas pada form <i>edit</i> data.	✓	
TC12	Admin mengimpor berkas CSV yang mengandung ID Karyawan atau <i>Username</i> yang duplikat.	Sistem menolak baris data duplikat tersebut pada proses impor dan memunculkan pesan gagal.	Deteksi duplikasi berhasil menyaring dan memblokir baris impor CSV yang bermasalah.	✓	
TC13	Admin mencari data dengan mengetikkan Nama Karyawan yang valid pada kolom pencarian (<i>Search</i>).	Tabel secara otomatis disaring untuk menampilkan baris yang mengandung Nama Karyawan tersebut.	Tabel otomatis menyaring baris data sesuai huruf atau kata yang diketikkan tanpa perlu <i>reload</i> .	✓	

TC14	Admin mencari data dengan mengetikkan ID Karyawan atau <i>Role</i> secara spesifik pada kolom pencarian.	Tabel secara otomatis disaring untuk menampilkan baris data pengguna yang sesuai dengan ID/ <i>Role</i> itu.	Tabel otomatis menyaring baris data sesuai huruf atau kata yang diketikkan tanpa perlu <i>reload</i> .	✓	
TC15	Admin mengetikkan kata kunci pencarian acak yang tidak ada di dalam pangkalan data.	Tabel dikosongkan secara visual dan memunculkan pesan " <i>No data available in table</i> ".	Sistem mengosongkan baris tabel dan menampilkan teks " <i>No data available in table</i> ".	✓	

Hasil Analisis:

1. Persentase *test case* yang lolos = $15/15 = 100\%$
2. Persentase *test case* yang gagal = $0/15 = 0\%$
3. Persentase *test case* yang telah diuji = $15/15 = 100\%$
4. Persentase *test case* yang belum diuji = $0/15 = 0\%$ Jumlah total *test case* dalam pengujian = 15

9. Pengujian Mengelola dan Mengunduh *QR Code*

Pengujian fungsionalitas pembuatan (*generate*) dan pengunduhan (*download*) *QR Code* dilakukan berdasarkan 10 skenario uji. Fitur ini merupakan komponen paling vital dalam sistem otomatisasi logistik karena *QR Code* akan dicetak sebagai label fisik barang dan kartu identitas karyawan. Pengujian ini berfokus untuk memastikan bahwa sistem hanya menghasilkan *QR Code* pada kategori yang diizinkan (*Office Equipment*, *Equipment*, dan *User*), fungsi unduh berjalan lancar, dan hasil kode yang diunduh dapat terbaca secara akurat oleh pemindai (*scanner*) fisik untuk kebutuhan operasional di lapangan.

Simulasi peran Admin dalam pengujian ini dilakukan secara langsung oleh Angelia Dwi Shella selaku *Lead QHSE & Legal*, mengingat departemen ini yang memegang kendali penuh atas administrasi pelabelan aset di lapangan. Hasil pengujian kemudian ditinjau secara menyeluruh oleh M. Dzikri Fadilah selaku *IT Supervisor* dan Pembimbing Magang di PT Dalaz Teknik Utama, sebagai bentuk verifikasi akhir kelayakan sistem.

Tabel 4. 9 Blackbox Testing Mengelola dan Mengunduh QR Code

Test Case ID	Skenario	Luaran yang Diharapkan	Luaran yang Dihasilkan	Hasil	
				B	G
TC01	Admin memeriksa baris data pada tabel kategori <i>Office Equipment</i> .	Sistem telah melakukan generate dan menampilkan gambar <i>QR Code</i> secara otomatis untuk aset tersebut.	<i>QR Code</i> berhasil dibuat otomatis oleh sistem saat data ditambahkan.	✓	
TC02	Admin menekan tombol aksi unduh (<i>download</i>) <i>QR Code</i> pada salah satu data <i>Office Equipment</i> .	Sistem berhasil mengunduh gambar <i>QR Code</i> dengan format yang sesuai ke penyimpanan lokal komputer.	Berkas gambar <i>QR Code</i> berhasil diunduh ke dalam perangkat komputer.	✓	
TC03	Admin memeriksa baris data pada tabel kategori <i>Equipment</i> .	Sistem telah melakukan generate dan menampilkan gambar <i>QR Code</i> secara otomatis untuk peralatan kerja tersebut.	<i>QR Code</i> alat kerja berhasil dibuat otomatis oleh sistem.	✓	
TC04	Admin menekan tombol aksi unduh (<i>download</i>) <i>QR Code</i> pada salah satu data <i>Equipment</i> .	Sistem berhasil mengunduh gambar <i>QR Code</i> alat kerja tersebut ke penyimpanan lokal untuk dicetak.	Berkas gambar <i>QR Code Equipment</i> berhasil diunduh ke perangkat komputer.	✓	
TC05	Admin memeriksa baris data pada tabel <i>User</i> (Karyawan).	Sistem telah melakukan <i>generate</i> dan menampilkan gambar <i>QR Code</i> yang berfungsi sebagai identitas peminjam.	<i>QR Code</i> identitas karyawan berhasil dibuat otomatis oleh sistem.	✓	
TC06	Admin menekan tombol aksi unduh (<i>download</i>) <i>QR Code</i> pada salah satu data <i>User</i> .	Sistem berhasil mengunduh gambar <i>QR Code</i> pengguna tersebut ke penyimpanan lokal untuk dicetak menjadi <i>ID Card</i> .	Berkas gambar <i>QR Code</i> Karyawan berhasil diunduh ke perangkat komputer.	✓	
TC07	Admin memeriksa baris data pada tabel kategori <i>Vehicle, Land & Building</i> , dan <i>Intangible Assets</i> .	Sistem tidak menghasilkan atau menampilkan tombol <i>QR Code</i> pada ketiga kategori statis ini.	Aturan sistem memblokir pembuatan <i>QR Code</i> di luar kategori yang diizinkan.	✓	

Hasil Analisis:

1. Persentase *test case* yang lolos = $7/7 = 100\%$
2. Persentase *test case* yang gagal = $0/7 = 0\%$
3. Persentase *test case* yang telah diuji = $7/7 = 100\%$

4. Persentase *test case* yang belum diuji = $0/7 = 0\%$ Jumlah total *test case* dalam pengujian = 7

10. Pengujian Skenario Peminjaman *Equipment*

Pengujian fungsionalitas proses peminjaman *Equipment* dilakukan berdasarkan 14 skenario uji yang melibatkan penggunaan perangkat keras pemindai (*Hardware QR Scanner*). Pengujian ini memvalidasi alur peminjaman yang terdiri dari tiga tahapan utama: identifikasi pengguna (*Step 1: ID Scan*), pengisian detail pekerjaan (*Step 2: Job Details*), dan pemindaian barang (*Step 3: Equipment Scan & Submit*). Fokus pengujian mencakup validasi data karyawan, ketersediaan aset, manipulasi tabel antrean *scan*, serta penahanan formulir (*form validation*) apabila terdapat data wajib yang tidak diisi sebelum transaksi dikonfirmasi.

Mengingat proses peminjaman ini merupakan aktivitas operasional yang paling sering dilakukan, pengujian dieksekusi secara langsung oleh Angelia Dwi Shella selaku *Lead QHSE & Legal* untuk memastikan alur kerja sistem sesuai dengan ritme di lapangan. Seluruh hasil pengujian kemudian ditinjau oleh M. Dzikri Fadilah selaku *IT Supervisor* sekaligus Pembimbing Magang di PT Dalaz Teknik Utama, sebagai langkah verifikasi kelancaran integrasi antara sistem web dengan perangkat keras pemindai fisik.

Tabel 4. 10 Blackbox Testing Peminjaman *Equipment*

<i>Test Case ID</i>	Skenario	Luaran yang Diharapkan	Luaran yang Dihasilkan	Hasil	
				B	G
TC01	<i>User</i> memindai <i>QR Code ID Card</i> karyawan yang valid dan terdaftar (<i>Step 1</i>).	ID otomatis terbaca, divalidasi sistem, dan otomatis memindahkan antarmuka ke <i>Step 2</i> .	Identifikasi sukses, sistem beralih ke halaman <i>Step 2</i> .	✓	
TC02	<i>User</i> memindai <i>QR Code ID Card</i> acak atau yang tidak terdaftar.	Sistem menggagalkan proses dan menampilkan alert " <i>User not registered</i> ".	Alert " <i>User not registered</i> " muncul, <i>User</i> tertahan.	✓	
TC03	<i>User</i> mengisi seluruh formulir	Sistem menerima input form	Form berhasil diisi, sistem	✓	

	<i>Job Details</i> (No JD, <i>Client</i> , <i>Location</i> , <i>Working Days</i>) dengan data valid (Step 2).	Step 2 tanpa kendala, dan memuluskan transisi ke layar Step 3.	menyimpan data dan lanjut ke Step 3.		
TC04	<i>User</i> mencoba melakukan konfirmasi peminjaman (<i>submit final</i>) dengan kondisi form No JD dikosongkan.	Saat <i>submit final</i> , sistem memblokir eksekusi dan memunculkan peringatan bahwa No JD wajib diisi.	Sistem mencegah <i>submit</i> dan menampilkan peringatan No JD kosong.	✓	
TC05	<i>User</i> mencoba melakukan konfirmasi peminjaman dengan kondisi form <i>Client</i> dikosongkan.	Saat <i>submit final</i> , sistem memblokir eksekusi dan memunculkan peringatan bahwa <i>Client</i> wajib diisi.	Sistem mencegah <i>submit</i> dan menampilkan peringatan <i>Client</i> kosong.	✓	
TC06	<i>User</i> mencoba melakukan konfirmasi peminjaman dengan kondisi form <i>Location</i> dikosongkan.	Saat <i>submit final</i> , sistem memblokir eksekusi dan memunculkan peringatan bahwa <i>Location</i> wajib diisi.	Sistem mencegah <i>submit</i> dan menampilkan peringatan <i>Location</i> kosong.	✓	
TC07	<i>User</i> mencoba melakukan konfirmasi peminjaman dengan kondisi form <i>Working Days</i> dikosongkan.	Saat <i>submit final</i> , sistem memblokir eksekusi dan memunculkan peringatan bahwa <i>Working Days</i> wajib diisi.	Sistem mencegah <i>submit</i> dan menampilkan peringatan <i>Working Days</i> kosong.	✓	
TC08	<i>User</i> memindai <i>QR Code Equipment</i> yang valid dan berstatus <i>Available</i> (Step 3).	Kode aset terekam, divalidasi ketersediaannya, dan masuk ke tabel daftar pinjam.	Data aset sukses ditambahkan ke dalam tabel antrean <i>scan</i> .	✓	
TC09	<i>User</i> memindai <i>QR Code</i> barang acak atau yang tidak ada di dalam database.	Sistem mendeteksi barang tidak valid dan memunculkan alert " <i>Equipment not available</i> ".	Alert " <i>Equipment not available</i> " muncul, data tertolak.	✓	
TC10	<i>User</i> memindai <i>QR Code Equipment</i> yang saat ini sedang dipinjam oleh orang lain (<i>Borrowed</i>).	Sistem mendeteksi konflik status barang dan menampilkan " <i>Equipment not available</i> ".	Alert " <i>Equipment not available</i> " muncul, data tertolak.	✓	
TC11	<i>User</i> menghapus satu baris data aset dari dalam tabel <i>Equipment Scanned</i> yang berisi beberapa barang.	Sistem menghapus satu baris data aset dari tabel tanpa mengganggu baris aset lainnya.	Aset spesifik sukses dihapus dari tabel antrean <i>scan</i> .	✓	

TC12	User menghapus seluruh baris data aset yang ada di dalam tabel <i>Equipment Scanned</i> .	Sistem berhasil membersihkan seluruh baris data aset, mengembalikan tabel ke status awal.	Seluruh aset sukses dihapus, tabel <i>scan</i> bersih.	✓	
TC13	User menekan tombol " <i>Confirm Borrowing</i> " saat tabel <i>Equipment Scanned</i> dalam keadaan kosong.	Sistem memblokir instruksi konfirmasi peminjaman dengan tabel/antrian kosong.	Sistem sukses menahan eksekusi tombol konfirmasi.	✓	
TC14	User mengonfirmasi peminjaman dengan Step 1 valid, Step 2 lengkap, dan tabel Step 3 berisi aset.	Setelah konfirmasi pada modal, pesan " <i>Borrowing Successful!</i> " tampil, status aset menjadi <i>BORROWED</i> .	Transaksi selesai. Status aset ter-update dan <i>History</i> tercatat.	✓	

Hasil Analisis:

1. Persentase *test case* yang lolos = $14/14 = 100\%$
2. Persentase *test case* yang gagal = $0/14 = 0\%$
3. Persentase *test case* yang telah diuji = $14/14 = 100\%$
4. Persentase *test case* yang belum diuji = $0/14 = 0\%$ Jumlah total *test case* dalam pengujian = 14

11. Pengujian Skenario Pengembalian *Equipment*

Pengujian fungsionalitas proses pengembalian *Equipment* dilakukan berdasarkan 14 skenario uji menggunakan pemindai perangkat keras (*Hardware QR Scanner*). Pengujian ini memvalidasi dua langkah utama: identifikasi peminjam (*Step 1: Identify Borrower*) dan pemindaian barang yang dikembalikan (*Step 2: Return Items*). Fokus pengujian mencakup validasi kepemilikan pinjaman, pelaporan kondisi barang (*Good, Damaged, Lost*), validasi wajib isi pada *Modal Defect Report*, serta berbagai skenario bisnis lanjutan seperti *Full Return, Partial Return*, dan *Combo Return*.

Proses pengujian fungsional ini dijalankan secara langsung oleh Angelia Dwi Shella selaku *Lead QHSE & Legal* untuk memastikan akurasi pelacakan dan kondisi aset yang dikembalikan ke gudang. Selanjutnya, seluruh hasil pengujian ditinjau dan dievaluasi oleh M. Dzikri Fadilah selaku *IT Supervisor* sekaligus Pembimbing Magang di PT Dalaz Teknik

Utama, sebagai langkah verifikasi akhir bahwa sistem siap diimplementasikan secara operasional.

Tabel 4. 11 Blackbox Testing Pengembalian Equipment

Test Case ID	Skenario	Luaran yang Diharapkan	Luaran yang Dihasilkan	Hasil	
				B	G
TC01	User memindai QR Code ID Card valid yang memiliki tanggungan pinjaman (Step 1).	ID otomatis terbaca, sistem memvalidasi tanggungan pinjaman, dan beralih ke Step 2.	Identifikasi sukses, sistem beralih ke halaman Step 2 (<i>Borrowed List</i>).	✓	
TC02	User memindai QR Code ID Card acak atau tidak terdaftar di sistem.	Sistem menolak instruksi dan menampilkan peringatan "User not registered".	Alert muncul, proses tertahan di Step 1.	✓	
TC03	User memindai QR Code ID Card yang valid namun tidak memiliki pinjaman aktif.	Sistem mendeteksi tidak ada tagihan pinjaman dan menampilkan status kosong/peringatan.	Sistem menginformasikan User bebas dari pinjaman aktif.	✓	
TC04	User memindai QR Code Equipment yang benar-benar ada di daftar pinjaman miliknya (Step 2).	Kode QR tervalidasi, sistem mencocokkan kepemilikan, dan meminta pemilihan Condition.	Scanner sukses mengenali barang, dropdown Condition terbuka.	✓	
TC05	User memindai QR Code Equipment yang sedang dipinjam orang lain atau masih tersedia di gudang.	Sistem memblokir pemindaian dan secara spesifik menampilkan pesan "This Equipment is not borrowed by you".	Alert "This Equipment is not borrowed by you" muncul.	✓	
TC06	User memindai barang valid dan memilih opsi kondisi "Good".	Sistem mencatat status barang siap dikembalikan secara normal tanpa memicu Modal Defect.	Kondisi Good tersimpan di memori antrean pengembalian.	✓	
TC07	User memilih kondisi "Damaged", namun mengosongkan teks kronologi atau tidak mengunggah foto, lalu klik "Save Report".	Sistem mencegah proses Save Report karena persyaratan wajib (<i>Chronology & Photo</i>) tidak terpenuhi.	Validasi form memblokir tombol simpan pada Modal Damaged.	✓	

TC08	<i>User</i> memilih kondisi "Damaged", mengisi kronologi lengkap, mengunggah multi-foto, lalu klik "Save Report".	Sistem menyimpan data <i>Damaged</i> beserta <i>file</i> unggahan jamak ke dalam memori sesi sementara.	Foto berhasil diunggah, <i>Defect Report (Damaged)</i> tersimpan sementara.	✓	
TC09	<i>User</i> memilih kondisi "Lost", namun mengosongkan teks kronologi kehilangan, lalu klik "Save Report".	Sistem mencegah proses Save Report karena <i>Defect Cause / Chronology</i> kosong.	Validasi form memblokir tombol simpan pada Modal <i>Lost</i> .	✓	
TC10	<i>User</i> memilih kondisi "Lost", mengisi teks kronologi tanpa unggahan foto, lalu klik "Save Report".	Sistem menyimpan data <i>Lost</i> (hanya butuh teks) ke dalam memori sesi sementara.	<i>Defect Report (Lost)</i> berhasil tersimpan tanpa perlu paksaan unggah foto.	✓	
TC11	<i>User</i> melakukan <i>Full Return</i> dengan men-scan seluruh barang tanggungannya (kondisi <i>Good</i>), lalu menekan "Confirm Return".	Sistem mengeksekusi pengembalian utuh. Menampilkan pesan "Return Successful!". Status seluruh aset ter-update dan tanggungan <i>User</i> lunas.	<i>Full Return</i> berhasil dieksekusi. Tanggungan aset <i>User</i> menjadi bersih.	✓	
TC12	<i>User</i> melakukan <i>Partial Return</i> dengan hanya men-scan sebagian barang dari daftar tanggungannya, lalu menekan "Confirm Return".	Sistem berhasil mengeksekusi pengembalian sebagian. Menampilkan pesan "Return Successful!". Sisa barang tidak terpengaruh.	<i>Partial Return</i> berhasil dieksekusi. Barang yang tidak di-scan tetap aktif dipinjam.	✓	
TC13	<i>User</i> melakukan <i>Combo Return</i> (satu barang diatur <i>Damaged</i> , satu barang diatur <i>Lost</i>), mengisi form keduanya, lalu menekan "Confirm Return".	Sistem berhasil mengeksekusi transaksi silang (Combo). Menampilkan "Return Successful!". Data masuk ke tabel <i>COMPLAINTS</i> .	Eksekusi 2 kondisi berbeda sukses. Tabel laporan cacat (<i>COMPLAINTS</i>) ter-update.	✓	
TC14	<i>User</i> menekan tombol "Confirm Return" tanpa men-scan satu pun barang ke dalam tabel antrian.	Sistem menahan instruksi konfirmasi karena tidak logis mengembalikan 0 barang dari tabel.	Sistem berhasil memblokir tombol konfirmasi saat tidak ada item yang disiapkan.	✓	

Hasil Analisis:

1. Persentase *test case* yang lolos = $14/14 = 100\%$
2. Persentase *test case* yang gagal = $0/14 = 0\%$
3. Persentase *test case* yang telah diuji = $14/14 = 100\%$

4. Persentase *test case* yang belum diuji = $0/14 = 0\%$ Jumlah total *test case* dalam pengujian = 14

12. Pengujian Skenario Mengajukan *Complaint / Defect Report*

Pengujian fungsionalitas dan validasi form pada *Modal Defect Report* dilakukan berdasarkan 6 skenario uji. Pengujian ini berfokus pada form pelaporan yang otomatis muncul ketika *User* mengubah status barang menjadi *Damaged* (atau *Cracked/Broken*) dan *Lost* pada tahapan pengembalian barang. Fokus utama pengujian ini adalah memastikan sistem mewajibkan pengisian teks kronologi dan unggahan foto bukti untuk kondisi rusak, serta memvalidasi pengecualian aturan unggah foto untuk kondisi barang hilang.

Untuk menjamin kesesuaian alur pelaporan keluhan ini dengan standar manajemen perusahaan, pengujian dilakukan langsung oleh Angelia Dwi Shella selaku *Lead QHSE & Legal*. Rekam jejak pengujian ini selanjutnya dievaluasi dan ditinjau oleh M. Dzikri Fadilah sebagai *IT Supervisor* dan Pembimbing Magang, guna memastikan logika validasi form berjalan sempurna sebelum sistem diimplementasikan.

Tabel 4. 12 Blackbox Testing Mengajukan *Complaint*

Test Case ID	Skenario	Luaran yang Diharapkan	Luaran yang Dihasilkan	Hasil	
				B	G
TC01	<i>User</i> memilih kondisi <i>Damaged</i> , mengisi <i>Defect Cause / Chronology</i> , dan mengunggah <i>Photos of Defect</i> secara lengkap.	Form tervalidasi sempurna. Modal tertutup dan sistem siap memproses status barang sebagai <i>Damaged/Cracked</i> pada konfirmasi akhir.	Data <i>Defect Report</i> (teks & foto) sukses tersimpan di antrean pengembalian.	✓	
TC02	<i>User</i> memilih kondisi <i>Damaged</i> , mengisi <i>Defect Cause</i> , namun tidak mengunggah foto bukti apapun.	Sistem menolak aksi <i>Save Report</i> dan menampilkan pesan peringatan: " <i>Please upload at least 1 evidence photo</i> ".	Validasi sukses menahan eksekusi akibat <i>field</i> lampiran foto kosong.	✓	
TC03	<i>User</i> memilih kondisi <i>Damaged</i> , mengosongkan kolom <i>Defect</i>	Sistem menolak aksi <i>Save Report</i> dan menampilkan	Validasi sukses menahan eksekusi akibat teks area	✓	

	<i>Cause</i> , namun tetap mengunggah foto bukti.	pesan peringatan: " <i>Defect Cause / Chronology is required</i> ".	kronologi dikosongkan.		
TC04	<i>User</i> memilih kondisi <i>Damaged</i> , membiarkan kolom teks kosong dan tidak mengunggah foto sama sekali.	Sistem menolak aksi <i>Save Report</i> secara menyeluruh dan meminta <i>User</i> melengkapi kedua <i>field</i> wajib tersebut.	Validasi sukses menahan form karena <i>User</i> tidak mengisi data apa pun.	✓	
TC05	<i>User</i> memilih kondisi <i>Lost</i> , dan hanya mengisi kronologi kehilangan tanpa mengunggah foto.	Sistem menyesuaikan logika (foto tidak wajib), menyimpan teks kronologi <i>Lost</i> , menutup Modal, dan memperbarui status tabel.	Data kehilangan sukses terekam hanya dengan modal teks kronologi tanpa <i>error</i> foto.	✓	
TC06	<i>User</i> memilih kondisi <i>Lost</i> , dan membiarkan kolom <i>Defect Cause / Chronology</i> dalam keadaan kosong.	Sistem menolak aksi <i>Save Report</i> untuk kondisi hilang, dan memunculkan peringatan: " <i>Defect Cause / Chronology is required</i> ".	Validasi sukses menahan form akibat <i>User</i> tidak menjelaskan kronologi kehilangan.	✓	

Hasil Analisis:

1. Persentase *test case* yang lolos = $6/6 = 100\%$
2. Persentase *test case* yang gagal = $0/6 = 0\%$
3. Persentase *test case* yang telah diuji = $6/6 = 100\%$
4. Persentase *test case* yang belum diuji = $0/6 = 0\%$ Jumlah total *test case* dalam pengujian = 6

13. Pengujian Skenario Melihat Riwayat Peminjaman

Pengujian fungsionalitas halaman Riwayat Peminjaman (*History*) dilakukan berdasarkan 11 skenario uji yang dibedakan berdasarkan hak akses (*Role Base*). Untuk pengguna reguler (*User*), pengujian difokuskan pada validasi keamanan data untuk memastikan pengguna hanya dapat melihat riwayat peminjamannya sendiri. Sedangkan untuk Admin, pengujian memvalidasi kemampuan sistem dalam menampilkan seluruh daftar *Equipment*, memuat halaman detail riwayat per peralatan, serta memastikan fungsionalitas fitur pencarian teks dan penyaringan tanggal (*Date filter*) beroperasi dengan presisi.

Proses validasi pembatasan akses dan pelacakan riwayat ini dilakukan langsung oleh Angelia Dwi Shella selaku *Lead QHSE & Legal* guna memastikan fungsionalitas monitoring beroperasi dengan baik. Selanjutnya, rekam jejak pengujian ditinjau secara menyeluruh oleh M. Dzikri Fadilah sebagai *IT Supervisor* dan Pembimbing Magang, untuk memverifikasi keandalan sistem sesuai dengan standar operasional perusahaan.

Tabel 4. 13 Blackbox Testing Melihat Riwayat Peminjaman

Test Case ID	Skenario	Luaran yang Diharapkan	Luaran yang Dihasilkan	Hasil	
				B	G
TC01	<i>User reguler</i> (yang memiliki riwayat transaksi) melakukan login dan membuka menu <i>History</i> .	Halaman memuat antarmuka " <i>History</i> " dan menampilkan tabel transaksi sukses.	Tabel " <i>History</i> " termuat sempurna di antarmuka <i>User</i> .	✓	
TC02	Memeriksa kolom nama/ID peminjam pada tabel <i>History</i> milik <i>User</i> tersebut.	Filter akses tingkat data beroperasi; <i>User</i> hanya dapat melihat riwayat dengan <i>Employee ID</i> miliknya sendiri.	Tidak ada kebocoran data. <i>User</i> hanya melihat histori peminjamannya sendiri.	✓	
TC03	<i>User reguler</i> baru (yang belum pernah meminjam) melakukan login dan membuka menu <i>History</i> .	Sistem menangani status kosong secara elegan dengan memunculkan indikator " <i>No borrowing History available</i> ".	Tampilan <i>empty state</i> muncul sesuai ekspektasi tanpa menyebabkan layar <i>blank</i> .	✓	
TC04	<i>User</i> mengeklik tombol paginasi (<i>Prev</i> , Angka Halaman, <i>Next</i>) pada tabel riwayat yang melebihi limit baris per halaman.	Navigasi <i>Next</i> , <i>Prev</i> , dan angka halaman menggeser baris data tabel secara dinamis.	Fitur paginasi beroperasi lancar untuk riwayat transaksi yang panjang.	✓	
TC05	Admin melakukan <i>login</i> dan membuka menu <i>History</i> .	Halaman memuat antarmuka " <i>Equipment History</i> " berisi daftar <i>Equipment</i> yang ada di pangkalan data operasional.	Daftar seluruh <i>Equipment</i> tampil rapi pada tabel antarmuka Admin.	✓	
TC06	Admin mengeklik tombol "Detail" pada kolom aksi untuk salah satu aset yang sering dipinjam di halaman <i>Equipment</i>	Sistem memindahkan halaman ke " <i>EQUIPMENT DETAIL HISTORY</i> ", memuat <i>header</i> informasi aset, dan tabel	Data detail aset dan tabel riwayat siklus hidupnya terekstrak dengan benar.	✓	

	<i>History.</i>	riwayatnya.			
TC07	Admin mengklik tombol "Detail" pada peralatan baru yang belum pernah dipinjam.	Detail aset terbuka, tabel tidak <i>error</i> , dan secara spesifik menampilkan pesan " <i>No borrowing History for this Equipment</i> ".	Pesan status kosong khusus untuk <i>Equipment</i> sukses ditampilkan.	✓	
TC08	Admin mengklik tombol " <i>Back to List</i> " saat berada di halaman <i>EQUIPMENT DETAIL HISTORY</i> .	Tombol " <i>Back to List</i> " sukses mengembalikan Admin ke halaman menu sebelumnya tanpa kehilangan sesi aktif.	Routing kembali ke daftar <i>Equipment History</i> bekerja dengan responsif.	✓	
TC09	Admin mengetikkan nama/ID karyawan tertentu (contoh: "Nita") di kotak pencarian (<i>Search Box</i>) pada halaman detail.	Sistem mengevaluasi kata kunci pada kotak pencarian dan menyaring data di tabel detail secara spesifik.	<i>Search box</i> sukses menemukan data transaksi berdasarkan teks pencarian.	✓	
TC10	Admin memilih rentang tanggal (<i>Date filter</i>) yang diketahui terdapat riwayat transaksi.	<i>Date filter</i> mengeksekusi parameter waktu dan menyortir tabel detail riwayat secara presisi sesuai rentang terpilih.	Filter rentang tanggal sukses memilah riwayat berdasarkan periode waktu operasi.	✓	
TC11	Admin memilih rentang tanggal di masa depan atau periode di mana tidak ada transaksi sama sekali pada <i>Date filter</i> .	Tabel detail menampilkan status kosong secara aman ketika filter waktu diarahkan ke rentang tak bertransaksi.	Validasi sistem sukses menghapus data dari layar (<i>empty state</i>) tanpa memunculkan error.	✓	

Hasil Analisis:

1. Persentase *test case* yang lolos = $11/11 = 100\%$
2. Persentase *test case* yang gagal = $0/11 = 0\%$
3. Persentase *test case* yang telah diuji = $11/11 = 100\%$
4. Persentase *test case* yang belum diuji = $0/11 = 0\%$ Jumlah total *test case* dalam pengujian = 11

14. Pengujian Skenario Mencatat Transaksi *Inventory In/Out*

Pengujian fungsionalitas pencatatan transaksi keluar-masuk barang persediaan (*Inventory In/Out*) dilakukan berdasarkan 10 skenario uji. Pengujian ini mencakup validasi fitur pencarian barang yang secara

otomatis mengunci stok saat ini (*Current Stock*), kalkulasi penambahan stok untuk aksi masuk (*IN*), pengurangan stok untuk aksi keluar (*OUT*), validasi pencegahan stok minus jika jumlah keluar melebihi stok yang ada, pengecekan form wajib, serta validasi lampiran dokumen pendukung berformat PDF dan batasan ukuran berkasnya.

Guna menjamin akurasi pengelolaan stok harian, pengujian fitur ini dilakukan langsung oleh Angelia Dwi Shella selaku *Lead QHSE & Legal*. Rekam jejak pengujian tersebut selanjutnya ditinjau secara komprehensif oleh M. Dzikri Fadilah sebagai *IT Supervisor* dan Pembimbing Magang, sebagai bentuk validasi akhir sebelum sistem diterapkan pada lingkungan operasional perusahaan.

Tabel 4. 14 Blackbox Testing Mencatat Transaksi Inventory In/Out

Test Case ID	Skenario	Luaran yang Diharapkan	Luaran yang Dihasilkan	Hasil	
				B	G
TC01	Admin mengetikkan nama barang pada <i>field SEARCH ITEM</i> ("Masker").	Pencarian barang interaktif menemukan ID barang dan otomatis menampilkan nilai <i>Current Stock</i> yang terkunci secara statis.	Fitur pencarian sukses; <i>Current Stock</i> terisi otomatis dan tidak bisa dimanipulasi manual.	✓	
TC02	Admin menginput transaksi dengan <i>Action "IN"</i> , <i>Quantity</i> 10, dan mengunggah dokumen pendukung PDF berukuran valid (2MB).	Sistem menambahkan <i>Quantity</i> ke <i>Current Stock</i> (kalkulasi <i>IN</i> berjalan), menyimpan lampiran PDF, dan menampilkan pesan sukses.	Transaksi <i>IN</i> berhasil. Kalkulasi presisi, file tersimpan, badge hijau tampil di tabel.	✓	
TC03	Admin menginput transaksi dengan <i>Action "OUT"</i> , <i>Quantity</i> 1, dan sengaja tidak melampirkan dokumen pendukung (dibiarkan kosong).	Sistem mengurangi <i>Quantity</i> (kalkulasi <i>OUT</i>), memproses transaksi dengan aman meskipun tanpa lampiran dokumen, dan menampilkan pesan sukses.	Transaksi <i>OUT</i> berhasil tanpa lampiran. Kalkulasi presisi, badge merah tampil di tabel.	✓	
TC04	Admin mencoba melakukan penarikan barang (<i>Action "OUT"</i>)	Sistem secara ketat mendeteksi potensi stok	Validasi logis berfungsi; sistem menggagalkan aksi	✓	

	dengan kuantitas yang melebihi jumlah stok yang tersedia.	minus, menolak transaksi, dan memunculkan notifikasi " <i>Stock not sufficient</i> ".	penarikan melebihi persediaan.		
TC05	Admin mencoba menyimpan transaksi tanpa memilih barang pada <i>field SEARCH ITEM</i> .	Sistem memblokir perintah simpan karena absennya identitas barang pada form <i>Search Item</i> (memunculkan peringatan).	Validasi <i>required field</i> sukses menahan form akibat <i>Search Item</i> kosong.	✓	
TC06	Admin mencoba menyimpan transaksi dengan membiarkan form <i>QUANTITY</i> dalam keadaan kosong.	Sistem memblokir perintah simpan karena absennya besaran transaksi pada form <i>Quantity</i> (memunculkan peringatan).	Validasi <i>required field</i> sukses menahan form akibat <i>Quantity</i> kosong.	✓	
TC07	Admin mencoba menyimpan transaksi dengan menghapus atau mengosongkan nilai pada form <i>DATE</i> .	Sistem memblokir perintah simpan karena ketiadaan tanggal pencatatan pada form <i>Date</i> (memunculkan peringatan).	Validasi <i>required field</i> sukses menahan form akibat <i>Date</i> dikosongkan.	✓	
TC08	Admin mencoba menyimpan transaksi dengan membiarkan form <i>ACTION</i> pada opsi bawaan (-- <i>Select</i> --).	Sistem memblokir perintah simpan karena jenis pergerakan belum dipilih pada <i>dropdown Action</i> (memunculkan peringatan).	Validasi <i>required field</i> sukses menahan form akibat opsi <i>Action</i> belum dipilih.	✓	
TC09	Admin mengunggah dokumen pendukung PDF dengan ukuran file melebihi batas maksimal (contoh: 8MB).	Sistem menahan proses upload karena ukuran file melanggar batas maksimal memori, memunculkan pesan peringatan batas 5MB.	Validasi ukuran file berfungsi; file > 5MB ditolak oleh sistem.	✓	
TC10	Admin mengunggah dokumen pendukung dengan format yang salah (contoh: file gambar .jpg atau dokumen .docx).	Sistem menahan proses upload karena ekstensi MIME-type tidak sesuai, memunculkan pesan peringatan yang mewajibkan format .pdf.	Validasi ekstensi file berfungsi; dokumen non-PDF ditolak oleh sistem.	✓	
TC11	Admin mengetikkan kata kunci berupa kode atau nama barang pada kolom pencarian (<i>search bar</i>) tabel riwayat.	Tabel menyortir data secara otomatis dan hanya menampilkan riwayat transaksi yang sesuai dengan	Fitur pencarian tabel berfungsi akurat; data riwayat langsung tersortir sesuai input kode atau nama.	✓	

		kode atau nama barang tersebut.			
TC12	Admin memilih tanggal spesifik pada fitur filter tanggal di atas tabel riwayat.	Tabel menyortir data riwayat secara spesifik dan hanya menampilkan rekam jejak transaksi <i>In/Out</i> yang terjadi pada rentang tanggal yang dipilih.	Filter tanggal berfungsi dengan baik; tabel berhasil menampilkan deretan riwayat sesuai dengan tanggal masukan.	✓	
TC13	Admin memilih opsi status " <i>IN</i> " atau " <i>OUT</i> " pada <i>dropdown</i> filter status di tabel.	Tabel menyortir data secara dinamis berdasarkan jenis pergerakan, hanya menampilkan deretan baris berstatus " <i>IN</i> " (Masuk) atau " <i>OUT</i> " (Keluar).	Filter status berfungsi dengan sangat baik; penyortiran data <i>IN</i> dan <i>OUT</i> berjalan akurat.	✓	

Hasil Analisis:

1. Persentase *test case* yang lolos = $13/13 = 100\%$
2. Persentase *test case* yang gagal = $0/13 = 0\%$
3. Persentase *test case* yang telah diuji = $13/13 = 100\%$
4. Persentase *test case* yang belum diuji = $0/13 = 0\%$ Jumlah total *test case* dalam pengujian = 13

15. Pengujian Skenario Memberikan Respon Komplain

Pengujian fungsionalitas bagi Admin untuk menanggapi (*Response*) laporan kerusakan/kehilangan (*Complaint*) yang diajukan oleh *User* dilakukan berdasarkan 11 skenario uji. Pengujian ini mencakup validasi keamanan form pada Modal "*Admin Response*", perubahan status antarmuka pada tabel secara *real-time* (tombol berubah menjadi *Edit*), serta pengujian keakuratan *generate* dokumen PDF *Defect Equipment Report* berdasarkan respons perbaikan atau pembuangan (*Disposal*) yang diberikan.

Untuk memastikan alur penanganan kerusakan aset ini berjalan sesuai prosedur perusahaan, simulasi pengujian dilakukan langsung oleh Angelia Dwi Shella selaku *Lead QHSE & Legal*. Rekam jejak pengujian tersebut

selanjutnya ditinjau secara komprehensif oleh M. Dzikri Fadilah sebagai *IT Supervisor* dan Pembimbing Magang, guna memastikan integritas data dan fungsionalitas sistem telah memenuhi standar kebutuhan operasional.

Tabel 4. 15 Blacbox Testing Memberikan Respon Komplain

Test Case ID	Skenario	Luaran yang Diharapkan	Luaran yang Dihasilkan	Hasil	
				B	G
TC01	Admin mengakses menu <i>Complaint</i> dan membuka halaman <i>Manage Complaints</i> .	Halaman <i>Manage Complaints</i> termuat dengan struktur tabel dan label status warna-warni yang jelas.	Antarmuka memuat tabel daftar komplain beserta status operasionalnya dengan baik.	✓	
TC02	Admin membuka modal " <i>Response</i> ", lalu menekan tombol " <i>Cancel</i> ".	Tombol " <i>Cancel</i> " berfungsi menutup modal dan membatalkan sesi pengisian tanpa menyimpan data.	Modal tertutup secara aman dan halaman kembali ke mode <i>view</i> tabel.	✓	
TC03	Admin mencoba menyimpan respons (<i>Save Response</i>) dengan mengosongkan kolom <i>CONTROL NO</i> .	Sistem mendeteksi kekosongan <i>Control No</i> dan memunculkan peringatan " <i>All required fields must be filled</i> ".	Validasi sukses memblokir eksekusi akibat <i>field Control No</i> kosong.	✓	
TC04	Admin mencoba menyimpan respons dengan mengosongkan kolom <i>DATE OF INSPECTION</i> .	Sistem mendeteksi kekosongan <i>Date of Inspection</i> dan memunculkan peringatan wajib isi.	Validasi sukses memblokir eksekusi akibat form <i>Date</i> kosong.	✓	
TC05	Admin mencoba menyimpan respons dengan mengosongkan teks area <i>TREATMENT / ACTION</i> .	Sistem mendeteksi kekosongan <i>Treatment / Action</i> dan memunculkan peringatan wajib isi.	Validasi sukses memblokir eksekusi akibat area teks <i>Treatment</i> kosong.	✓	
TC06	Admin mencoba menyimpan respons tanpa memilih opsi pada <i>dropdown STATUS</i> .	Sistem mendeteksi tidak adanya status final yang dipilih, menahan instruksi <i>Save Response</i> .	Validasi sukses memblokir eksekusi karena <i>Dropdown Status</i> tidak dipilih.	✓	
TC07	Admin mengisi form respons dengan valid dan memilih status " <i>Repaired</i> " (<i>REPAIR</i>).	Sistem menyimpan respons Admin ke basis data, tabel ter-update, badge berubah biru (<i>REPAIR</i>), dan muncul	Respon <i>REPAIR</i> tersimpan sempurna. Antarmuka tabel diperbarui.	✓	

		notifikasi sukses.			
TC08	Admin mengisi form respons dengan valid dan memilih status "Disposal".	Sistem menyimpan respons Admin, tabel ter-update, badge berubah merah (DISPOSAL).	Respon DISPOSAL tersimpan sempurna. Visualisasi status bekerja dengan akurat.	✓	
TC09	Admin melihat perubahan baris data pada tabel untuk komplain yang baru saja diberikan respons.	Logika UI mendeteksi komplain terselesaikan, lalu mengganti mode tombol action ke mode Edit dan mengaktifkan cetak PDF.	Perubahan adaptif tombol UI (Response -> Edit) berfungsi sesuai aturan sistem.		
TC10	Admin menekan tombol "Edit" pada komplain yang sudah direspons dan mengubah isinya.	Sistem memuat ulang data existing ke dalam modal, memproses perubahan baru, dan menyimpannya.	Fungsionalitas "Edit" sukses memperbarui tanggapan Admin di database.	✓	
TC11	Admin mengeklik ikon dokumen PDF berwarna merah untuk mengunduh laporan.	Sistem berhasil mengonversi format data menjadi Defect Equipment Report berformat PDF dengan relasi status (Repair/Replaced/Disposal) yang akurat.	File PDF berhasil terunduh dengan tata letak dokumen formal dan isi data yang presisi.	✓	

Hasil Analisis:

1. Persentase *test case* yang lolos = $11/11 = 100\%$
2. Persentase *test case* yang gagal = $0/11 = 0\%$
3. Persentase *test case* yang telah diuji = $11/11 = 100\%$
4. Persentase *test case* yang belum diuji = $0/11 = 0\%$ Jumlah total *test case* dalam pengujian = 11

16. Pengujian Skenario Mengunduh Laporan PDF

Pengujian fungsionalitas pencetakan dan pengunduhan laporan ekspor berformat PDF pada sistem dilakukan berdasarkan 10 skenario uji. Pengujian ini difokuskan pada dua modul utama: ekspor massal daftar aset pada halaman *Equipment*, dan ekspor satuan untuk lembar formulir resmi *Defect Equipment Report* milik PT DALAZ pada halaman *Manage*

Complaints. Fokus pengujian mencakup keakuratan *render* dokumen, validasi pencegahan cetak pada status yang belum direspons, sinkronisasi data riil ke dalam lembar cetak, hingga keamanan proteksi URL (*Direct URL Bypass*).

Untuk menjamin standar dokumentasi laporan sesuai dengan standar operasional perusahaan, simulasi pengujian dilakukan langsung oleh Angelia Dwi Shella selaku *Lead QHSE & Legal*. Rekam jejak pengujian tersebut selanjutnya ditinjau secara komprehensif oleh M. Dzikri Fadilah sebagai *IT Supervisor* dan Pembimbing Magang, guna memastikan bahwa fitur ekspor data telah berjalan dengan aman sebelum diimplementasikan.

Tabel 4. 16 Blackbox Testing Mengunduh Laporan PDF

Test Case ID	Skenario	Luaran yang Diharapkan	Luaran yang Dihasilkan	Hasil	
				B	G
TC01	Admin mengeklik tombol "Export PDF" pada halaman <i>Equipment</i> saat data aset tersedia di tabel.	Sistem berhasil mengonversi baris tabel menjadi dokumen PDF formal dan otomatis mengunduh file secara utuh ke penyimpanan lokal.	Dokumen daftar aset sukses terunduh dalam format berkas PDF dengan aman.	✓	
TC02	Admin menerapkan filter pencarian pada tabel <i>Equipment</i> , lalu mengeklik "Export PDF".	Dokumen PDF yang terunduh bersifat dinamis; hanya memuat baris data yang lolos penyaringan filter, bukan seluruh isi <i>database</i> .	Dokumen PDF sukses tercetak secara spesifik mengikuti parameter filter aktif pada web.	✓	
TC03	Admin mengeklik "Export PDF" saat pangkalan data kosong atau pencarian menghasilkan nihil.	Sistem menolak proses perenderan dokumen kosong, menghentikan unduhan, dan menampilkan notifikasi kesalahan "Data not found".	Pesan "Data not found" muncul di layar, mencegah <i>print error</i> pada <i>file</i> kosong.	✓	
TC04	Admin mencoba mengeklik ikon dokumen PDF pada baris komplain yang masih berstatus <i>WAITING RESPONSE</i> .	Logika UI mendeteksi komplain belum ditindaklanjuti; ikon dokumen berwarna abu-abu dan terkunci (<i>disabled</i>) sehingga tidak bisa	Ikon terkunci secara aman, mencegah pencetakan formulir yang belum diinspeksi oleh Admin.	✓	

		diklik.			
TC05	Admin mengeklik ikon PDF merah pada baris data komplain berstatus <i>REPAIR</i> .	Sistem membaca status <i>REPAIR</i> , memicu perintah unduh berkas PDF lembar formulir resmi <i>Defect Equipment Report</i> milik PT DALAZ.	File formulir cetak untuk data CTR-02 sukses terunduh dengan respons cepat.	✓	
TC06	Admin mengeklik ikon PDF merah pada baris data komplain berstatus <i>DISPOSAL</i> .	Sistem membaca status <i>DISPOSAL</i> , memicu perintah unduh berkas PDF lembar formulir resmi <i>Defect Equipment Report</i> milik PT DALAZ.	File formulir cetak untuk data pembuangan barang sukses terunduh tanpa kendala tautan.	✓	
TC07	Memvalidasi <i>layout</i> dokumen PDF hasil ekspor massal daftar <i>Equipment</i> .	Dokumen PDF menampilkan <i>layout</i> tabel yang presisi, memiliki koping nama perusahaan PT DALAZ, dan teks tidak terpotong margin kertas.	Struktur visual dokumen laporan daftar aset tercetak rapi sesuai standar format perusahaan.	✓	
TC08	Memvalidasi kelengkapan data sinkron pada dokumen satuan <i>Defect_Report_CTR-02.pdf</i> .	Lembar PDF memuat data sinkron dari sistem: Kronologi = "bare", <i>Treatment</i> = "iya", dan tabel kotak pengecekan otomatis tercentang [X] pada baris " <i>Repair</i> ".	Konten dokumen tercetak dengan sangat akurat sesuai isi data riil pada <i>database</i> web.	✓	
TC09	Logout dari Admin, lalu mencoba melakukan <i>bypass</i> akses dengan mengetikkan langsung URL <i>download</i> PDF di <i>browser</i> .	Sistem mendeteksi pelanggaran hak akses sesi, menolak proses pemuatan PDF, dan otomatis mengalihkan pengguna kembali ke halaman <i>Login</i> .	Keamanan <i>backend</i> URL berfungsi; instruksi <i>bypass link</i> berhasil ditolak oleh sistem.		
TC10	Mengunduh file PDF secara berurutan dan membukanya pada software pihak ketiga (<i>Adobe Acrobat/Chrome</i>).	File PDF lolos pengujian <i>integrity check</i> ; tidak rusak, tidak <i>corrupt</i> , dan seluruh elemen grafis (garis tabel & teks) dirender sempurna.	Berkas PDF stabil dan dapat dibuka dengan baik di berbagai aplikasi pembaca PDF.	✓	

Hasil Analisis:

1. Persentase *test case* yang lolos = $10/10 = 100\%$
2. Persentase *test case* yang gagal = $0/10 = 0\%$
3. Persentase *test case* yang telah diuji = $10/10 = 100\%$
4. Persentase *test case* yang belum diuji = $0/10 = 0\%$ Jumlah total *test case* dalam pengujian = 10

17. Pengujian Skenario *Logout*

Pengujian fungsionalitas pengakhiran sesi (*Logout*) dilakukan berdasarkan 4 skenario uji untuk pengguna tingkat Admin maupun *User*. Pengujian ini tidak hanya berfokus pada keberhasilan pengalihan (*redirect*) ke halaman *Login*, tetapi juga secara ketat memvalidasi keamanan penghancuran sesi (*session destruction*). Hal ini bertujuan untuk memastikan sistem kebal terhadap upaya akses ilegal pasca-*logout*, seperti penggunaan tombol *Back* pada *browser*, manipulasi tautan URL langsung, hingga pencegahan celah keamanan pada akses *multi-tab*.

Proses pengujian keamanan sesi ini dilaksanakan secara langsung oleh Angelia Dwi Shella selaku *Lead QHSE & Legal* guna memastikan integritas keamanan akun karyawan tetap terjaga. Seluruh hasil pengujian kemudian dievaluasi dan ditinjau oleh M. Dzikri Fadilah selaku *IT Supervisor* sekaligus Pembimbing Magang di PT Dalaz Teknik Utama, sebagai langkah verifikasi akhir bahwa celah keamanan akses telah tertutup rapat.

Tabel 4. 17 Blackbox Testing Logout

Test Case ID	Skenario	Luaran yang Diharapkan	Luaran yang Dihasilkan	Hasil	
				B	G
TC01	Pengguna yang sedang aktif mengeklik menu profil di <i>header</i> dan memilih opsi " <i>Logout</i> ".	Modul menghancurkan sesi aktif (<i>session_destroy()</i>) di pangkalan data secara instan dan mengalihkan pengguna ke antarmuka otentikasi awal (<i>Login page</i>).	Fitur <i>Logout</i> berjalan lancar, sesi sukses dihancurkan, rute kembali ke <i>Login</i> .	✓	

TC02	Pengguna menekan tombol "Back" pada web browser setelah berhasil <i>Logout</i> dan berada di halaman <i>Login</i> .	Konfigurasi kontrol tombolok (<i>cache-control headers</i>) mencegah <i>browser</i> menampilkan halaman <i>Dashboard</i> sebelumnya, memaksa pengguna tetap di gerbang <i>Login</i> .	Keamanan <i>cache</i> sukses menahan instruksi <i>Back browser</i> , memblokir akses data tanpa <i>login</i> .	✓	
TC03	Pengguna yang telah <i>Logout</i> mengetikkan rute tautan terlindungi secara manual di <i>address bar browser</i> (contoh: akses langsung ke <i>dashboard</i>).	Pengecekan sesi tingkat sistem (<i>session validation middleware</i>) mendeteksi tidak adanya ID otorisasi, lalu menolak permintaan rute paksa (<i>URL Bypass</i>).	Keamanan sesi sukses menolak akses masuk via ketikan URL langsung.	✓	
TC04	Pengguna membuka sistem di dua tab (A dan B), melakukan <i>Logout</i> di Tab A, lalu mencoba memuat ulang atau mengeklik menu di Tab B.	Penghancuran sesi di Tab pertama bersifat global untuk seluruh siklus <i>browser</i> . Interaksi apa pun di Tab kedua otomatis terdeteksi sebagai akses ilegal dan dialihkan ke <i>Login</i> .	Validasi <i>multi-tab</i> sukses, eksekusi aksi di tab yang tersisa langsung digagalkan dan dialihkan.	✓	

Hasil Analisis:

1. Persentase *test case* yang lolos = $4/4 = 100\%$
2. Persentase *test case* yang gagal = $0/4 = 0\%$
3. Persentase *test case* yang telah diuji = $4/4 = 100\%$
4. Persentase *test case* yang belum diuji = $0/4 = 0\%$ Jumlah total *test case* dalam pengujian = 4

4.2.2 Pengujian Perangkat Lunak (*Non-Functional Testing*)

Pengujian non-fungsional bertujuan untuk mengevaluasi kualitas, stabilitas, dan keamanan sistem. Berbeda dengan pengujian fungsional yang berfokus pada fitur, pengujian ini menilai kinerja sistem dalam kondisi operasional nyata. Aspek utama yang diuji meliputi perlindungan data dari akses tidak sah, serta kecepatan respons sistem saat memproses banyak transaksi secara bersamaan.

1. *Security Testing*

Pada tahap ini dilakukan pengujian non-fungsional untuk memastikan bahwa sistem memenuhi kebutuhan pada aspek keamanan web. Pengujian ini difokuskan pada kemampuan aplikasi dalam melindungi data kredensial melalui algoritma enkripsi *Bcrypt*, mekanisme autentikasi masuk (*login*), serta pencegahan akses tidak sah melalui penerapan *Role-Based Access Control* (RBAC) untuk memisahkan hak akses antara Administrator dan Karyawan. Selain itu, dilakukan pemindaian kerentanan menggunakan perangkat OWASP ZAP untuk memastikan ketahanan sistem terhadap serangan injeksi seperti *SQL Injection* dan *Cross-Site Scripting* (XSS). Pengujian ini disusun berdasarkan kebutuhan non-fungsional NFR-01, yaitu sistem harus memiliki perlindungan autentikasi dan enkripsi dasar yang memadai untuk menjaga kerahasiaan data, serta memastikan hanya pengguna yang memiliki otorisasi sah yang dapat mengakses halaman manajemen aset.

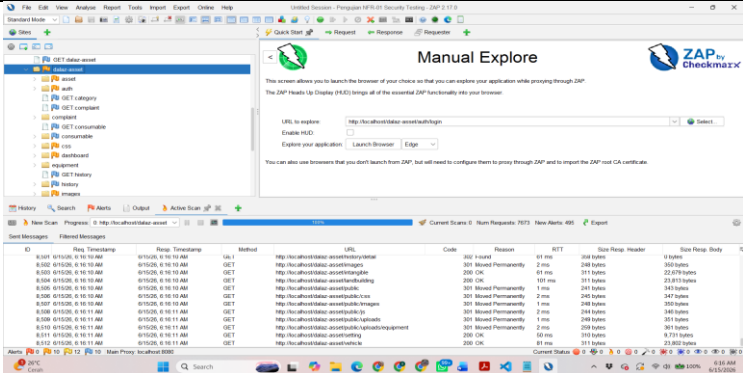
Tabel 4. 18 *Security Testing*

Komponen	Rincian
<i>Test Case ID</i>	TC-NF01
<i>Type</i>	<i>Security Testing</i>
Tujuan	Melakukan pengujian untuk memastikan sistem memiliki perlindungan terhadap berbagai ancaman keamanan web, memvalidasi enkripsi <i>password</i> (<i>Bcrypt</i>), serta mencegah akses yang tidak sah melalui penerapan <i>Role-Based Access Control</i> (RBAC).
Lingkungan	Pengujian dilakukan pada <i>server</i> pengembangan lokal (<i>localhost</i> / XAMPP) menggunakan <i>database testing</i> dengan akses terbatas untuk menghindari kerusakan data utama.
Alat	OWASP ZAP (<i>Zed Attack Proxy</i>) versi 2.17.0 & <i>phpMyAdmin</i>
Metode	1. Melakukan pengujian keamanan otomatis (<i>Active Scan</i>) menggunakan pendekatan <i>penetration testing</i> pada seluruh <i>endpoint</i> sistem. 2. Mengidentifikasi potensi kerentanan terhadap injeksi (<i>SQL Injection</i> , <i>XSS</i>) secara terkontrol untuk menguji ketahanan sistem. 3. Menganalisis respon <i>server</i> dan <i>header</i> keamanan dari aplikasi PHP.
Ekspektasi	Sistem tidak memiliki celah keamanan tingkat tinggi (<i>High Risk</i>), sistem <i>login</i> terlindungi dari serangan injeksi paksa (<i>brute force</i> / <i>bypass</i>), serta akses fitur dikontrol sesuai otorisasi pengguna.

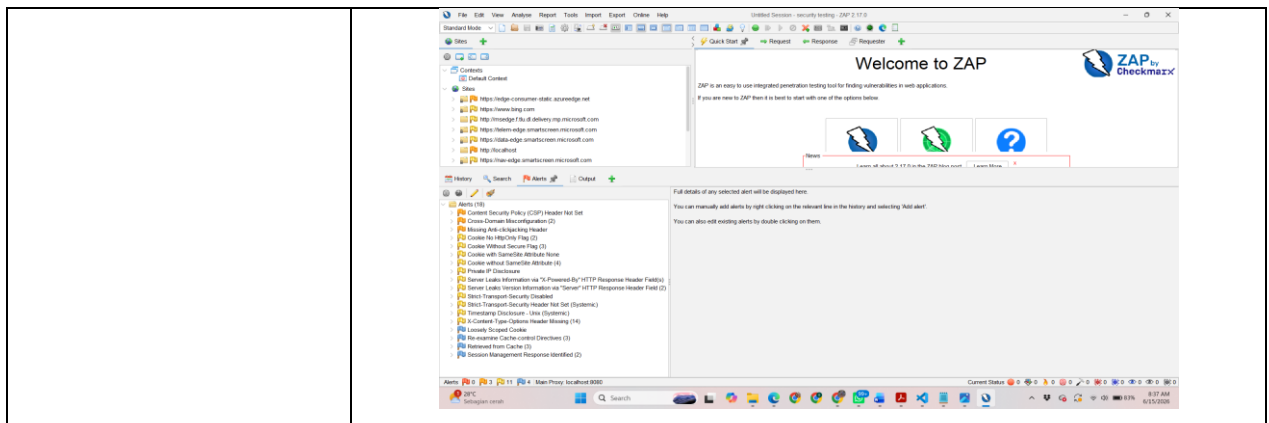
Hasil Uji			
Jumlah Alert Keamanan		12	
No	Temuan	Tingkat Resiko	Penjelasan
1	<i>Absence of Anti-CSRF Tokens</i>	Menengah	Formulir HTML di dalam sistem PHP tidak menyertakan token Anti-CSRF secara menyeluruh. Hal ini dapat memungkinkan penyerang membuat permintaan palsu atas nama pengguna yang sedang masuk (login).
2	<i>Content Security Policy (CSP) Header Not Set</i>	Menengah	Aplikasi belum mengirimkan <i>header Content-Security-Policy</i> pada respons HTTP. Kondisi ini dapat meningkatkan risiko serangan <i>Cross Site Scripting (XSS)</i> karena <i>browser</i> tidak membatasi sumber skrip yang dapat dijalankan.
3	<i>Missing Anti-clickjacking Header</i>	Menengah	Aplikasi tidak menerapkan <i>header</i> keamanan <i>X-Frame-Options</i> atau <i>frame-ancestors</i> . Hal ini memungkinkan tampilan antarmuka sistem (seperti form <i>login</i>) disisipkan ke dalam <i>iframe</i> oleh pihak lain dan berpotensi memicu serangan <i>clickjacking</i> .
4	<i>Directory Browsing</i>	Menengah	Server XAMPP masih mengizinkan <i>Directory Browsing</i> , sehingga pengguna dapat melihat daftar fail di dalam direktori publik (seperti folder <i>uploads</i> atau <i>css</i>) jika tidak ada <i>fail index.php</i> .
5	<i>Cookie Without Secure Flag</i>	Rendah	Aplikasi berjalan di lingkungan <i>localhost</i> (HTTP biasa) sehingga <i>cookie</i> sesi tidak dikirimkan menggunakan jalur yang dienkripsi (HTTPS). Hal ini wajar terjadi pada tahap pengembangan lokal.
6	<i>Server Leaks Version Information via "Server"</i>	Rendah	Server merespons dengan menampilkan informasi versi perangkat lunak (Apache/PHP) pada <i>header</i> HTTP. Informasi ini dapat mempermudah penyerang melakukan <i>profiling</i> terhadap infrastruktur sistem.
7	<i>Modern Web Application</i>	Informational	Aplikasi teridentifikasi sebagai aplikasi web modern. Pemindaian standar berhasil menelusuri struktur sistem.

8	Re-examine Cache-control Directives	Informational	Header Cache-Control belum dikonfigurasi secara spesifik. Hal ini memungkinkan browser menyimpan data halaman yang mungkin sensitif di dalam cache lokal pengguna.
9	Retrieved from Cache	Informational	Beberapa respons dokumen berhasil diambil langsung dari cache browser. Evaluasi konfigurasi caching disarankan untuk data yang bersifat real-time.
10	Session Management Response Identified	Informational	ZAP mendeteksi adanya mekanisme respons manajemen sesi (fitur Otentikasi/Login) yang aktif untuk melindungi data sistem.
11	Authentication Request Identified	Informational	Sistem terdeteksi meminta proses autentikasi (kredensial) untuk mengakses halaman Dashboard. Ini mengonfirmasi bahwa NFR-01 terkait fungsi perlindungan Login berjalan dengan baik.
12	User Agent Fuzzer	Informational	Pemindai aktif (active scanner) melakukan pengujian dengan berbagai modifikasi User Agent untuk melihat respons server. Sistem tidak menunjukkan perilaku anomali.

Hasil Uji Security Testing



The screenshot displays the ZAP web application security tool interface. The left pane shows a tree view of discovered resources under 'http://localhost:8080'. The main window is titled 'Manual Explore' and shows a table of HTTP requests and responses. The table includes columns for ID, Request Timestamp, Response Timestamp, Method, URL, Code, Reason, RTT, Size Resp, Header, and Size Body. The first row shows a GET request to http://localhost:8080/asset/building.html with a 200 OK status and a response size of 248 bytes. Subsequent rows show various other requests to different assets like asset/public.js, asset/public.css, and asset/public.js?v=1, all returning 200 OK or 204 No Content status codes.

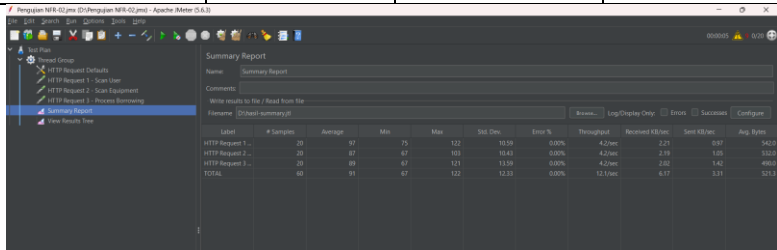
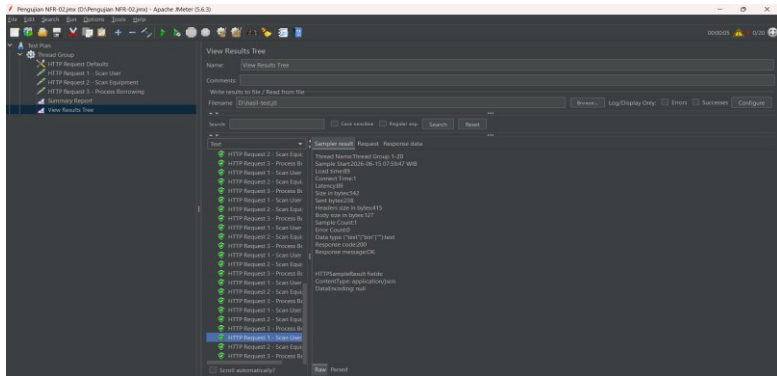


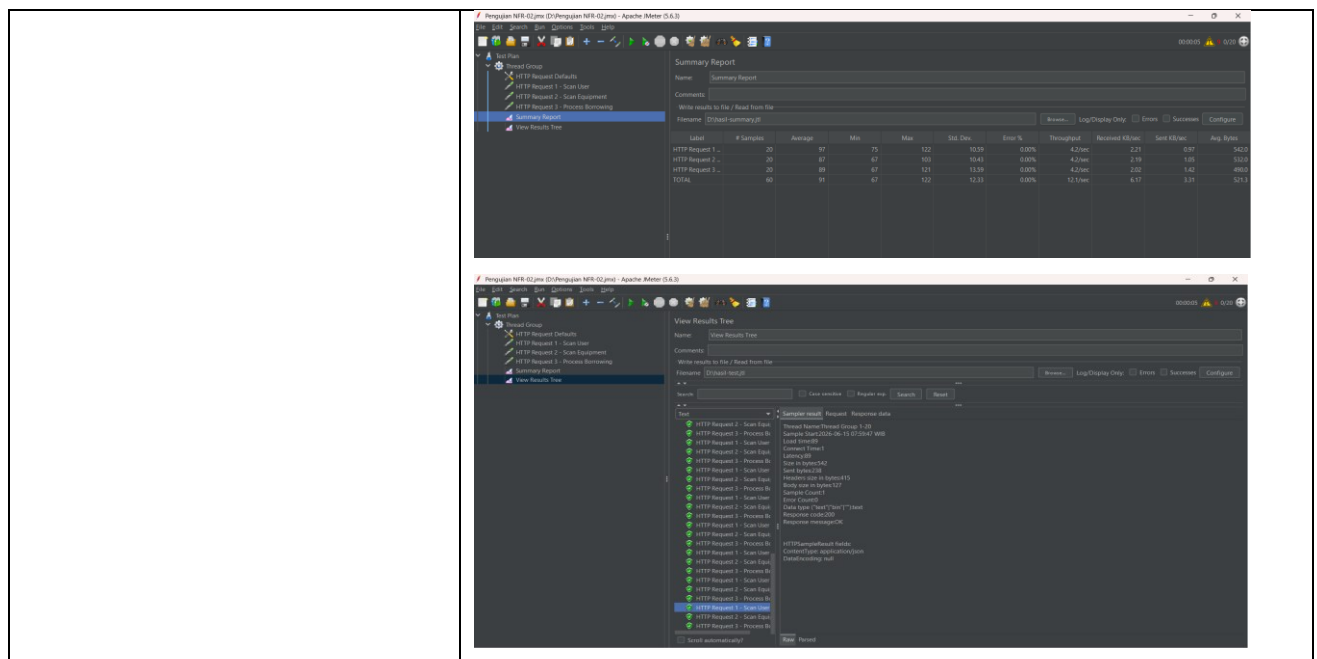
2. Performance Testing

Tahap pengujian non-fungsional selanjutnya difokuskan pada aspek performa dan stabilitas sistem di bawah beban akses yang tinggi (*Load Testing*). Pengujian ini disusun berdasarkan Kebutuhan Non-Fungsional NFR-02, yaitu sistem harus mampu memproses transaksi dan pencarian data dalam waktu maksimal 5 detik (5.000 milidetik). Evaluasi ini sangat krusial untuk memastikan bahwa sistem berbasis web tetap responsif dan tidak mengalami kegagalan proses saat diakses oleh banyak pengguna secara bersamaan. Pengujian dilakukan menggunakan Apache JMeter dengan mensimulasikan 20 pengguna virtual yang mengakses sistem secara serentak dalam periode *ramp-up* 5 detik, mencakup endpoint transaksi Peminjaman (*Borrowing*) dan Pengembalian (*Return*). Hasil pengujian disajikan pada tabel berikut.

Tabel 4. 19 Performance Testing

Komponen	Rincian
Test Case ID	TC-NF02
Type	Performance / Load Testing
Tujuan	Mengukur waktu respons rata-rata (<i>Average Response Time</i>) sistem dalam memproses transaksi untuk memastikan kinerjanya tidak melebihi batas toleransi 5 detik (5000 ms) dalam kondisi banyak akses.
Lingkungan	Pengujian dilakukan pada server pengembangan lokal (localhost / XAMPP Apache & MariaDB).
Alat	Apache JMeter

Metode	Mensimulasikan 20 pengguna virtual (<i>threads</i>) yang melakukan request secara serentak (<i>ramp-up</i> period 5 detik) pada jalur pemrosesan data (<i>POST</i>) secara berulang.				
Ekspektasi	Rata-rata waktu respons sistem untuk setiap endpoint berada di bawah 5000 ms, dan persentase transaksi yang gagal (<i>Error Rate</i>) adalah 0%.				
Hasil Uji					
Skenario Pengujian (<i>Endpoint</i>)	Jumlah Permintaan (<i>Samples</i>)	Rata-rata Waktu Respons (<i>Average</i>)	Waktu Maksimal (<i>Max</i>)	Persentase <i>Error</i>	Status
Scan User - Borrow Transaction	20	97 ms	122 ms	0.00%	Lulus (< 5 Detik)
Scan Equipment - Borrow Transaction	20	87 ms	103 ms	0.00%	Lulus (< 5 Detik)
Process - Borrow Transaction	20	89 ms	121 ms	0.00%	Lulus (< 5 Detik)
Scan User - Return Transaction	20	231 ms	1044 ms	0.00%	Lulus (< 5 Detik)
Scan Equipment - Return Transaction	20	88 ms	115 ms	0.00%	Lulus (< 5 Detik)
Process - Return Transaction	20	130 ms	232 ms	0.00%	Lulus (< 5 Detik)
Hasil Uji <i>Performance Testing</i>	 				



3. Usability Testing

Pengujian *usability* dilakukan untuk mengevaluasi apakah tampilan antarmuka sistem memenuhi aspek kerapian, keterbacaan, kemudahan penggunaan, serta responsivitas pada berbagai ukuran layar. Pengujian ini dilakukan secara observasional dengan mengakses langsung fitur-fitur utama sistem menggunakan *browser* pada perangkat desktop dan *mobile*.

Tabel 4. 20 Usability Testing

Aspek yang Dinilai	Kriteria	Hasil Observasi	Status
Kerapian Tampilan (UI)	Tata letak elemen konsisten dan terstruktur	Tata letak header, menu navigasi, tabel, dan form tersusun rapi dan konsisten di setiap halaman	Lulus
Kemudahan Penggunaan (UX)	Alur penggunaan mudah dipahami tanpa panduan khusus	Alur peminjaman dan pengembalian menggunakan step <i>indicator</i> yang jelas (Step 1 → 2 → 3)	Lulus
Intuitif	Ikon dan label mudah dipahami	Penggunaan ikon dan label teks pada setiap tombol dan menu mencerminkan fungsinya dengan jelas	Lulus
Responsivitas	Tampilan menyesuaikan ukuran layar	Tampilan sistem menyesuaikan ukuran layar desktop dan <i>mobile</i> tanpa mengalami pergeseran elemen	Lulus

4. *Portability Testing*

Pengujian portabilitas dilakukan untuk memverifikasi bahwa seluruh antarmuka dan konten sistem menggunakan Bahasa Inggris sebagai bahasa utama, sehingga sistem dapat digunakan secara konsisten tanpa hambatan bahasa.

Tabel 4. 21 *Portability Testing*

Aspek yang Dinilai	Kriteria	Hasil Observasi	Status
Label & Navigasi	Seluruh menu dan tombol menggunakan Bahasa Inggris	Menu navigasi dan seluruh tombol aksi menggunakan Bahasa Inggris	Lulus
Formulir & Input	Seluruh label <i>field</i> formulir menggunakan Bahasa Inggris	Label pada menggunakan Bahasa Inggris	Lulus
Pesan Sistem	Notifikasi dan pesan sistem menggunakan Bahasa Inggris	Pesan sukses, pesan <i>error</i> , dan konfirmasi sistem seluruhnya ditampilkan dalam Bahasa Inggris	Lulus
Konten Halaman	Seluruh teks konten menggunakan Bahasa Inggris	Judul halaman, deskripsi, dan keterangan tabel seluruhnya menggunakan Bahasa Inggris	Lulus
Ekspor PDF	Dokumen laporan yang diekspor menggunakan bahasa yang sesuai kebutuhan pelaporan perusahaan	Dokumen laporan diekspor dalam Bahasa Indonesia sesuai kebutuhan pelaporan perusahaan	Lulus

BAB V

KESIMPULAN

5.1 Kesimpulan

Berdasarkan hasil perancangan, pembangunan, dan pengujian yang telah dilakukan, seluruh tujuan Proyek Akhir ini dapat dinyatakan tercapai. Kesimpulan yang diperoleh adalah sebagai berikut :

1. Sistem informasi manajemen aset berbasis web berhasil dibangun untuk mengelola data enam kategori aset di PT Dalaz Teknik Utama, meliputi *Office Equipment, Equipment, Inventory, Vehicle, Land & Building*, dan *Intangible Assets*, dengan antarmuka yang terstruktur dan mudah dioperasikan.
2. Fitur *dashboard* dan laporan berhasil disediakan untuk menampilkan status ketersediaan aset secara langsung berdasarkan data transaksi terakhir yang tersimpan dalam basis data, sehingga mendukung pengambilan keputusan yang lebih cepat dan akurat.
3. Hasil pengujian non-fungsional menunjukkan bahwa sistem telah memenuhi seluruh kebutuhan yang ditetapkan, mencakup keamanan autentikasi dan *role-based access control* (NFR-01), performa rata-rata di bawah 250 ms dengan *Error Rate* 0% (NFR-02), antarmuka yang rapi dan intuitif pada berbagai ukuran layar (NFR-03), serta konsistensi penggunaan Bahasa Inggris pada antarmuka sistem (NFR-04).

Dengan terealisasinya seluruh tujuan tersebut, sistem manajemen aset berbasis *QR Code* yang dikembangkan mampu mendukung proses pengelolaan logistik PT Dalaz Teknik Utama secara lebih efisien, akurat, dan terdigitalisasi, sehingga mengurangi ketergantungan terhadap pencatatan manual yang selama ini menjadi sumber utama kesalahan operasional.

5.2 Saran

Berdasarkan keterbatasan yang ditemukan selama proses pengembangan dan pengujian, berikut beberapa saran yang direkomendasikan untuk pengembangan sistem lebih lanjut.

1. Peningkatan Keamanan Sistem

Sistem dapat ditingkatkan dengan menerapkan mekanisme autentikasi yang lebih kuat seperti *two-factor authentication* (2FA) serta enkripsi data pada level basis data, guna melindungi data aset perusahaan dari akses yang tidak sah.

2. Pengembangan Aplikasi *Mobile*

Untuk meningkatkan kemudahan akses, sistem dapat dikembangkan menjadi aplikasi *mobile* berbasis Android atau iOS, sehingga proses pemindaian *QR Code* dan pemantauan aset dapat dilakukan lebih fleksibel di lapangan.

3. Pengelolaan Multi-Cabang

PT Dalaz Teknik Utama memiliki lebih dari satu lokasi operasional, sehingga sistem dapat dikembangkan untuk mendukung pengelolaan aset lintas lokasi secara terpusat dan memungkinkan seluruh aset di setiap cabang dipantau dalam satu platform yang terintegrasi.

Dengan diterapkannya saran-saran tersebut, diharapkan sistem manajemen aset ini dapat terus berkembang menjadi solusi pengelolaan logistik yang semakin optimal, adaptif, dan mampu mendukung kebutuhan operasional PT Dalaz Teknik Utama secara berkelanjutan.

DAFTAR PUSTAKA

- Achour, M., Betz, F., Dovgal, A., Lopes, N., Magnusson, H., Richter, G., Seguy, D., & Vrana, J. (2022). *PHP manual*. The PHP Group. <https://www.php.net/manual/en/>
- Anggraeni, R. (2025). QR Code-based information system Designed to display gallery and information museum. *Buana Teknologi*, 8(1), 25-32. <https://doi.org/10.24002/jbt.v8i1.2165>
- Anggara, B., Anshor, A. H., & Hadikristanto, W. (2024). Implementation web-based QR-Code information system Design in warehouse inventory management system using Rapid Application Development (RAD) method at PT Dharma Precision Parts. *Formosa Journal of Computer and Information Science*, 3(2), 81-90. <https://doi.org/10.55927/fjcis.v3i2.10117>
- Bassil, Y. (2012). A simulation Model for the waterfall software development life cycle. *International Journal of Engineering and Technology (iJET)*, 2(5), 1-7. <https://arxiv.org/abs/1205.6904>
- Dicoding. (2025, Mei 14). *Waterfall Model: Pahami 5 tahap dasarnya untuk sukseskan proyek IT*. <https://www.dicoding.com/blog/waterfall-Model-pahami-5-tahap-dasarnya-untuk-sukseskan-proyek-it/>
- Fajar, M., Rahman, A., & Susanto, B. (2025). Optimization of inventory management with QR Code integration and sequential search algorithm: A case study in a regional revenue Office. *Journal of Applied Informatics and Computing*, 9(2), 245-260. <https://doi.org/10.30871/jaic.v9i2.8919>
- Grandynarta, S. M., Sudarma, M., & Arthana, I. K. R. (2024). Sistem manajemen aset sekolah dengan QR Code dan website. *Jurnal Mahasiswa Aplikasi Teknologi Komputer dan Informasi (JMApTeKsi)*, 6(1), 28-34. <https://doi.org/10.35145/jmapteksi.v6i1.4134>
- Ken Research. (2025). *Indonesia warehousing market outlook to 2028*. <https://www.kenresearch.com/industry-reports/indonesia-warehousing-market-outlook-to-2028>

- Kusumojati, P. P., & Mediawati, E. (2024). Web-based asset management information systems in higher education. *International Journal of Business, Law, and Education*, 5(1), 398-411. <https://doi.org/10.56442/ijble.v5i1.382>
- Lim, M., Setiawan, A., & Hartono, B. (2021). Perancangan sistem informasi berbasis web untuk manajemen data akademik. *Jurnal Comasie*, 4(2), 52-60. <https://doi.org/10.31539/comasie.v4i2.3173>
- Noguerra, C. P. (2023). Design and evaluation of an innovative mobile solution: QR Code-based inventory monitoring system. *International Journal of Advanced Engineering and Management Research*, 8(3), 231-242. <https://doi.org/10.51505/ijaemr.2023.8317>
- Oktawijaya, J. D., & Agustin, S. (2024). Optimalisasi sistem digitalisasi penerimaan dan pengeluaran barang dengan Laravel 11 pada PT Swadaya Graha Gresik. *Jurnal RESTIKOM (Riset Teknik Informatika dan Komputer)*, 6(3), 489-500.
- Oyenyi, L. D., Ugochukwu, C. E., & Mhlongo, N. Z. (2024). IoT applications in asset management: A review of accounting and tracking techniques. *International Journal of Science and Research Archive*, 11(2), 1510-1525. <https://doi.org/10.30574/ijjsra.2024.11.2.0640>
- Pangaila, F. S., Saerang, D. P., & Tirayoh, V. Z. (2022). QR Code labeling Design for fixed asset management in accounting information system PT Cipta Jaya Perkasa. *Coopetition: Jurnal Ilmiah Manajemen*, 15(6), 901-912.
- PPM School. (2024, November 30). *Manajemen aset: Pengertian, fungsi dan tahapan*. <https://www.ppmschool.ac.id/manajemen-aset/>
- Rahim, A., Santoso, H., & Wibowo, A. (2025). Implementasi warehouse management system pada PT XYZ. *Jurnal Sistem Informasi Bisnis*, 5(1), 78-92.
- Rijadi, S. C. R., & Suakanto, S. (2024). Development of an information system for asset management. *Jurnal Inovtek Polbeng - Seri Informatika*, 9(2), 940-952.

- Saputra, D., Wijaya, K., & Kurniawan, R. (2024). Optimizing inventory receipt processes through a mobile application utilizing QR Code technology. *International Journal of Mobile Computing and Application*, 11(4), 201-215.
- Sumberajar. (2023, Desember 31). *Pengertian sistem informasi: Komponen, jenis, proses, dan fungsinya dalam organisasi*. <https://sumberajar.com/post/pengertian-sistem-informasi-komponen-jenis-proses-dan-fungsinya-dalam-organisasi>
- Telkom University. (2024, Januari 8). *Metode Waterfall dalam pengembangan perangkat lunak*. <https://it.telkomuniversity.ac.id/metode-waterfall-dalam-pengembangan-perangkat-lunak/>

DAFTAR LAMPIRAN

Lampiran 1 Dokumen Proses Pengumpulan *Requirement*

[Link Dokumen Work Process](#)

Lampiran 2 Dokumentasi Perancangan Sistem

[Link Dokumentasi Perancangan Sistem](#)

Lampiran 3 *Link* Produk

[Link Github](#)

Lampiran 4 Dokumen Pengujian UAT

Security Testing

[Link Security Testing](#)

Performance Testing

[Link Performance Testing](#)

Dokumen *User Acceptance Testing* (UAT)

[Link Dokumen UAT](#)

Lampiran 5 Demo Aplikasi

[Link Demo Aplikasi](#)