

**Rancang Bangun Aplikasi Web E-Checksheet
untuk Optimalisasi Proses Quality Control pada
PT XYZ**

Proyek Akhir

Disusun Oleh:

Steve Yosafat Marpaung

3312301101



PROGRAM STUDI TEKNIK INFORMATIKA

JURUSAN TEKNIK INFORMATIKA

POLITEKNIK NEGERI BATAM

2026

KATA PENGANTAR

Puji dan syukur atas kehadiran Tuhan Yesus Kristus atas segala berkat, karunia, dan petunjuk-Nya sehingga penulis dapat menyelesaikan tugas akhir yang berjudul “Rancang Bangun Aplikasi Web *E-Checksheet* untuk Optimalisasi Proses *Quality Control* pada PT XYZ” dengan baik dan tepat waktu. Tugas akhir ini disusun sebagai salah satu syarat untuk memperoleh gelar D3 Teknik Informatika pada Politeknik Negeri Batam.

Dalam proses penyusunan tugas akhir ini, penulis menyadari bahwa tidak akan mungkin dapat terselesaikan tanpa bantuan, bimbingan, serta dukungan dari berbagai pihak. Oleh karena itu, dengan segala kerendahan hati, penulis ingin menyampaikan terima kasih yang sebesar-besarnya kepada:

1. Tuhan Yesus Kristus yang telah mencurahkan Berkatnya hingga detik ini.
2. Ayah dan Ibu tercinta atas doa, dukungan, dan cinta kasih sayang yang tiada hentinya.
3. Bapak Ir. Bambang Hendrawan, S.T., M.S.M., CIPMP., CISCIP. Selaku Direktur Politeknik Negeri Batam.
4. Bapak Ahmad Hamim Thohari, S.S.T., M.T. selaku Kepala Jurusan Teknik Informatika
5. Ibu Yeni Rokhayati, S.Si., M.Sc, selaku Kepala Program Studi Teknik Informatika
6. Ibu Cyntia Lasmi Andesti, S.Kom., M. Kom, selaku dosen pembimbing Penulis. Terimakasih karena telah sangat banyak meluangkan waktu, tenaga, dan kesabaran dalam membimbing. Serta kebijaksanaan dalam memberikan saran dan nasihat yang sangat bermanfaat bagi penulis dalam mengerjakan proposal tugas akhir ini.
7. Seluruh dosen dan staf Teknik Informatika yang telah memberikan pendidikan serta ilmu yang bermanfaat bagi penulis selama mengampu pendidikan di Politeknik Negeri Batam.

Penulis menyadari bahwa laporan ini jauh dari sempurna. Oleh karena itu, kritik dan saran yang membangun akan penulis harapkan demi penyempurnaan di masa-masa yang akan mendatang. Besar harapan penulis, tugas akhir ini dapat memberikan manfaat dan informasi bagi pembaca dan menjadi tambahan pengetahuan dalam bidang Teknik Informatika.

Akhir kata, semoga segala bantuan dan dukungan yang diberikan mendapat berkat yang berlipat-lipat ganda dari Tuhan Yesus Kristus.

DAFTAR ISI

KATA PENGANTAR.....	2
DAFTAR ISI.....	3
DAFTAR TABEL.....	5
DAFTAR GAMBAR.....	6
ABSTRAK.....	8
BAB I PENDAHULUAN.....	9
1.1 Latar Belakang.....	9
1.2 Rumusan Masalah.....	10
1.3 Tujuan.....	11
1.4 Batasan Masalah.....	11
1.5 Manfaat.....	11
BAB II TINJAUAN PUSTAKA.....	13
2.1 Penelitian Terkait.....	13
2.2 Landasan Teori.....	16
2.3 Metode Pengembangan Produk.....	18
2.4 Studi Kasus: Kondisi Awal Proses Quality Control di PT XYZ.....	22
BAB III ANALISIS DAN PERANCANGAN.....	25
3.1 Analisis Kebutuhan.....	25
3.2 Perancangan.....	29
BAB IV IMPLEMENTASI DAN PENGUJIAN.....	53
4.1 Hasil Implementasi.....	53
4.2 Pengujian Black Box Testing.....	63
BAB V KESIMPULAN.....	72
5.1 Kesimpulan.....	72

5.2 Saran.....	73
DAFTAR PUSTAKA.....	74
DAFTAR LAMPIRAN.....	77
Lampiran A: Dokumen Proses Pengumpulan Requirement.....	77
Lampiran B: Link Produk / GitHub.....	77
Lampiran C: Dokumen Pengujian Black Box Testing.....	77

DAFTAR TABEL

Table 2.1 Perbandingan Penelitian Sebelumnya.....	18
Table 3.1 Ringkasan Teknologi yang Digunakan.....	32
Table 3.2 Kebutuhan Fungsional.....	33
Table 3.3 Kebutuhan Non-Fungsional.....	34
Tabel 3.4 Skenario Use Case Login.....	36
Tabel 3.5 Skenario Use Case Pengisian Checksheet.....	36
Tabel 3.6 Skenario Use Case Approve/Reject Checksheet.....	37
Tabel 3.7 Skenario Use Case Kelola Pengguna.....	38
Tabel 3.8 Skenario Use Case Kelola Form dan Jadwal Inspeksi.....	38
Tabel 3.9 Skenario Use Case Monitor Checksheet.....	39
Tabel 3.10 Skenario Use Case Lihat Summary.....	40
Tabel 3.11 Skenario Use Case Notifikasi Otomatis.....	40
Tabel 4.1 Implementasi Lingkungan Pengembangan.....	58
Tabel 4.3 Hasil Pengujian Black Box Testing - Login dan Autentikasi (FR-01).....	69
Tabel 4.4 Hasil Pengujian Black Box Testing - Manajemen Pengguna (FR-02).....	71
Tabel 4.5 Hasil Pengujian Black Box Testing - Manajemen Form dan Jadwal Inspeksi (FR-03).....	71
Tabel 4.6 Hasil Pengujian Black Box Testing - Pengisian Checksheet oleh Petugas QC (FR-07).....	73
Tabel 4.7 Hasil Pengujian Black Box Testing - Approve dan Reject oleh Supervisor (FR-04).....	74
Tabel 4.8 Hasil Pengujian Black Box Testing - Monitoring Real-Time oleh Supervisor (FR-05).....	75

Tabel 4.9 Hasil Pengujian Black Box Testing - Notifikasi Otomatis (FR-08 dan FR-09).....	75
Tabel 4.10 Hasil Pengujian Black Box Testing - Summary dan Laporan (FR-06).....	76
Tabel 4.11 Hasil Pengujian Black Box Testing - Kebutuhan Non-Fungsional (NFR-01 s.d. NFR-05).....	77
Tabel 4.12 Distribusi Jawaban Responden Kuesioner SUS.....	78
Tabel 4.13 Hasil Pengujian System Usability Scale (SUS).....	80

DAFTAR GAMBAR

Gambar 2.1 Metode Agile-Scrum.....	21
Gambar 3.1 Gambaran Umum Sistem.....	30
Gambar 3.2 Diagram Use Case.....	35
Gambar 3.3 Activity Diagram Login.....	42
Gambar 3.4 Activity Diagram Admin Kelola Pengguna.....	43
Gambar 3.5 Activity Diagram Admin Kelola Checksheet.....	45
Gambar 3.6 Activity Diagram Supervisor Monitor Checksheet.....	45
Gambar 3.7 Activity Diagram Supervisor Dashboard.....	46
Gambar 3.8 Activity Diagram Summary.....	46
Gambar 3.9 Activity Diagram QC Checksheet.....	47
Gambar 3.10 Activity Diagram Notifikasi QC Checksheet.....	48
Gambar 3.11 Activity Diagram Notifikasi QC Checksheet Ditolak.....	48
Gambar 3.12 Entity Relationship Diagram (ERD).....	49
Gambar 3.13 Wireframe Halaman Login.....	50
Gambar 3.14 Wireframe Dashboard.....	51
Gambar 3.15 Wireframe Monitor Checksheet Supervisor.....	51
Gambar 3.16 Wireframe Summary Supervisor.....	52
Gambar 3.17 Wireframe Dashboard QC.....	52
Gambar 3.18 Wireframe Isi Checksheet QC.....	53
Gambar 3.19 Wireframe Summary QC.....	53
Gambar 3.20 Wireframe Notifikasi Sidebar.....	54
Gambar 3.21 Wireframe Dashboard Admin.....	54

Gambar 3.22 Wireframe Manajemen Pengguna.....	55
Gambar 3.23 Wireframe Form Inspeksi.....	55
Gambar 3.24 Wireframe Jadwal Inspeksi.....	56
Gambar 4.1 Halaman Login.....	59
Gambar 4.2 Dashboard Admin.....	60
Gambar 4.3 Halaman Manajemen Pengguna.....	61
Gambar 4.4 Halaman Form Inspeksi.....	62
Gambar 4.5 Halaman Jadwal Inspeksi.....	62
Gambar 4.6 Dashboard Petugas QC.....	63
Gambar 4.7 Halaman Isi Checksheet.....	64
Gambar 4.8 Dashboard Supervisor.....	65
Gambar 4.9 Halaman Monitor Checksheet.....	66
Gambar 4.10 Fitur Notifikasi Sistem.....	67
Gambar 4.11 Halaman Summary.....	68.

ABSTRAK

PT XYZ masih menggunakan sistem *checksheet* berbasis kertas dalam proses quality control, yang dinilai tidak efisien, rentan rusak, dan menyulitkan dalam pelacakan data. Tujuan proyek ini adalah mengembangkan aplikasi web *e-checksheet* untuk mendigitalisasi proses quality control, sehingga meningkatkan efisiensi, akurasi, dan kemudahan pelaporan. Metode pengembangan yang digunakan adalah Agile-Scrum dengan framework Laravel dan Quasar Vue.js. Hasil implementasi berupa sistem *e-checksheet* yang bisa membantu petugas QC melakukan pengisian *checksheet*, supervisor memantau status pengisian secara real-time, sistem memberikan notifikasi pengingat otomatis jika *checksheet* belum diisi, serta menghasilkan laporan *summary* harian dan mingguan. Pengujian menggunakan metode *Black Box Testing* menunjukkan seluruh 51 skenario pengujian berhasil, membuktikan bahwa sistem berjalan sesuai kebutuhan pengguna. Sistem ini diharapkan dapat memberikan kontribusi nyata dalam optimalisasi proses quality control di PT XYZ.

Kata Kunci: *E-Checksheet, Quality Control, Vue.js, Quasar, Agile-Scrum*

BAB I PENDAHULUAN

1.1 Latar Belakang

Dalam industri manufaktur, *Quality Control* menempati posisi krusial karena bertugas menjaga agar mutu produk yang dihasilkan tetap konsisten. Khususnya pada industri elektronik, tiap tahapan produksi wajib diawasi secara ketat agar hasilnya sesuai dengan standar kualitas yang ditetapkan perusahaan. Salah satu instrumen yang lazim dipakai untuk mendukung pengawasan mutu tersebut adalah *checksheet*, yakni formulir pencatatan hasil pemeriksaan atas parameter-parameter tertentu sepanjang proses produksi berjalan.

Idealnya, sistem *checksheet* sudah berbentuk digital agar pencatatan, penyimpanan, dan pelaporan data dapat berlangsung lebih efisien dan akurat. Digitalisasi pada proses *quality control* juga mempermudah penelusuran data, mempercepat penyusunan laporan, dan memungkinkan manajemen melakukan pengawasan secara *real-time*. Hal ini sejalan dengan temuan Zhang et al. (2023) dalam *Journal of Intelligent Manufacturing*, yang mengungkapkan bahwa penerapan digitalisasi pada manajemen mutu dapat mendorong efisiensi produksi sekaligus menekan kesalahan manusia (*human error*). Sejalan dengan itu, Hofmann et al. (2023) dalam jurnal *Buildings* memperlihatkan bahwa sistem *quality control* berbasis digital dapat mempercepat proses inspeksi di lapangan dan meningkatkan akurasi dokumentasi mutu.

Kondisi tersebut berbeda dengan praktik yang berjalan di PT XYZ, di mana *checksheet* yang dipakai masih berupa formulir kertas yang diisi manual oleh petugas *quality control*. Setiap harinya, petugas perlu mengisi formulir untuk sejumlah parameter pemeriksaan lalu menyerahkannya untuk diarsipkan. Cara kerja manual semacam ini memunculkan sejumlah kendala operasional, di antaranya risiko dokumen rusak atau hilang akibat faktor lingkungan seperti kelembapan, robek, maupun tumpahan cairan, serta sulitnya menelusuri data historis karena sistem pengarsipan yang kurang tertata. Di sisi lain, supervisor

tidak memiliki akses untuk memantau pengisian *checksheet* secara *real-time* sehingga potensi masalah produksi cenderung terlambat terdeteksi. Ketiadaan pengingat otomatis pun membuat sebagian petugas lupa mengisi *checksheet* sesuai jadwal yang ditentukan, sementara penyusunan laporan menjadi kurang efisien karena data dari *checksheet* kertas harus dipindahkan ulang secara manual ke komputer.

Bertolak dari permasalahan dan kesenjangan tersebut, penelitian ini diarahkan untuk merancang serta membangun sistem *e-checksheet* berbasis web bagi PT XYZ sebagai upaya digitalisasi proses *quality control*. Sistem yang dirancang memungkinkan petugas mengisi data inspeksi secara digital, menyimpan hasil pemeriksaan pada basis data terpusat, mengirimkan notifikasi pengingat otomatis, menyediakan fitur monitoring *real-time* untuk supervisor, sekaligus menghasilkan laporan secara otomatis. Melalui penerapan sistem ini, proses *quality control* di PT XYZ diharapkan menjadi lebih efisien, akurat, transparan, dan mudah diawasi, sehingga turut mendukung peningkatan kualitas produk secara berkelanjutan.

Istilah "optimalisasi" yang dipakai pada judul penelitian ini merujuk pada definisi dalam Kamus Besar Bahasa Indonesia (KBBI), yakni proses, cara, atau tindakan mengoptimalkan sesuatu agar mencapai kondisi terbaik, tertinggi, atau paling menguntungkan. Oleh sebab itu, cakupan optimalisasi pada penelitian ini tidak hanya terbatas pada pengalihan *checksheet* dari format kertas ke berbasis web, tetapi juga meliputi upaya memperbaiki kinerja proses *quality control* agar lebih efektif dan efisien dibandingkan kondisi sebelumnya. Hal ini selaras dengan tujuan penelitian, yaitu menghadirkan sistem *E-checksheet* yang dapat mempersingkat waktu pengisian dan penyusunan laporan, meningkatkan akurasi pencatatan data, memperkuat transparansi lewat fitur monitoring *real-time*, serta memudahkan pengawasan oleh supervisor. Dengan demikian, tingkat optimalisasi dalam penelitian ini diukur dari sejauh mana capaian-capaian tersebut mengalami perbaikan dibandingkan proses *checksheet* manual yang sebelumnya berjalan di

PT XYZ, sehingga proses *quality control* dapat berlangsung secara lebih optimal secara menyeluruh.

1.2 Rumusan Masalah

Mengacu pada uraian latar belakang tersebut, berikut adalah beberapa permasalahan yang dihadapi PT XYZ dalam pelaksanaan *quality control* dengan sistem *checksheet* manual:

1. Bagaimana merancang serta membangun sistem *E-checksheet* berbasis web yang mampu menggantikan proses pengisian *checksheet* manual di PT XYZ?
2. Bagaimana sistem *E-checksheet* dapat dilengkapi fitur notifikasi pengingat otomatis guna memastikan pengisian *checksheet* berlangsung tepat waktu?
3. Bagaimana sistem *E-checksheet* dapat menghadirkan dashboard monitoring real-time sehingga supervisor mampu memantau status pengisian dan hasil pemeriksaan secara langsung?

1.3 Tujuan

Penelitian ini bertujuan mengembangkan sekaligus mengimplementasikan sistem *E-checksheet* berbasis web guna meningkatkan efektivitas kegiatan *quality control* di PT XYZ. Secara khusus, tujuan penelitian ini adalah sebagai berikut:

1. Merancang serta membangun sistem *E-checksheet* berbasis web untuk mendigitalisasi proses pengisian *checksheet* pada kegiatan *quality control* di PT XYZ.
2. Menerapkan fitur notifikasi pengingat otomatis agar petugas *quality control* mampu mengisi *checksheet* sesuai jadwal yang ditentukan.
3. Menghadirkan dashboard monitoring real-time bagi supervisor agar dapat memantau status pengisian dan hasil pemeriksaan secara cepat dan akurat.

1.4 Batasan Masalah

Supaya penelitian ini tetap terarah dan selaras dengan tujuan yang hendak dicapai, ditetapkan sejumlah batasan masalah sebagai berikut:

1. Penelitian ini berfokus pada pengembangan sistem *E-checksheet* berbasis web untuk mendukung kegiatan quality control di PT XYZ.
2. Sistem yang dibangun hanya mencakup proses pengisian, penyimpanan, monitoring, dan pelaporan checksheet, serta tidak mencakup kontrol mesin produksi maupun pengambilan keputusan otomatis.
3. Data yang dipakai dalam sistem bersifat simulasi atau data uji coba yang disesuaikan dengan kebutuhan studi kasus di PT XYZ.

1.5 Manfaat

Di luar tujuan yang telah diuraikan, penelitian ini juga diharapkan memberi kontribusi baik dari sisi keilmuan maupun penerapannya secara langsung di lapangan, sebagaimana dijabarkan berikut ini:

a. Manfaat Teoritis

Dari sisi akademis, hasil penelitian ini diharapkan mampu memberikan kontribusi berikut:

1. Memperkaya khazanah keilmuan di bidang teknologi informasi, khususnya terkait penerapan sistem informasi berbasis web untuk mendukung kegiatan quality control pada sektor industri manufaktur.
2. Dapat menjadi bahan referensi bagi penelitian selanjutnya yang mengangkat topik digitalisasi lini produksi maupun otomatisasi sistem pengawasan mutu.
3. Turut mendukung penerapan konsep Industry 4.0 melalui sistem digital yang terintegrasi dalam proses pengendalian kualitas.

b. Manfaat Praktis

Secara praktis, penelitian ini diharapkan memberikan manfaat sebagai berikut:

1. Membantu PT XYZ meningkatkan efisiensi dan akurasi proses quality control dengan menggantikan sistem checksheet manual menggunakan sistem digital berbasis web.
2. Memudahkan supervisor dan manajemen memantau secara real-time aktivitas pemeriksaan serta hasil inspeksi kualitas produk.
3. Menekan risiko hilang atau rusaknya data checksheet karena seluruh data tersimpan dalam basis data digital yang aman.
4. Mempercepat penyusunan laporan pemeriksaan berkat adanya fitur pelaporan otomatis.
5. Memudahkan petugas quality control melalui notifikasi pengingat otomatis agar kegiatan inspeksi berlangsung tepat waktu.

BAB II TINJAUAN PUSTAKA

2.1 Penelitian Terkait

Sejumlah penelitian terdahulu memperlihatkan perkembangan sistem digitalisasi quality control sekaligus menjadi acuan bagi kebaruan yang diusung penelitian ini.

Hofmann et al. (2023) membangun *Digital Quality Control System* berbasis web untuk proyek konstruksi yang berhasil mempercepat proses inspeksi lapangan sekaligus meningkatkan akurasi dokumentasi mutu. Meski demikian, sistem tersebut belum dilengkapi fitur notifikasi otomatis, padahal keberadaan fitur tersebut berpotensi menekan kekurangan pada proses inspeksi secara signifikan.

Zhang et al. (2023) membangun platform digital yang mengusung pendekatan *data-driven quality management* untuk lingkungan manufaktur bertingkat. Platform ini terbukti mampu mendongkrak efisiensi serta menekan *human error*, namun belum dilengkapi dashboard monitoring real-time. Sistem tersebut memanfaatkan analitik data berlapis untuk memetakan keterkaitan sebab-akibat antar tahapan produksi sehingga sumber defect dapat ditelusuri secara lebih presisi. Selain itu, data dari berbagai lini produksi diintegrasikan ke dalam satu basis data terpusat melalui pendekatan multi-level data fusion, yang memungkinkan manajer kualitas mengenali pola kegagalan berulang dan mengambil tindakan korektif berdasarkan data. Kendati begitu, sistem ini dirancang khusus untuk lingkungan manufaktur bertingkat yang telah memiliki infrastruktur sensor, sehingga belum menyediakan antarmuka monitoring real-time yang dapat langsung dipakai oleh supervisor di lantai produksi tanpa memerlukan keahlian analitik khusus.

Santoso dan Ramadhan (2024) mengembangkan *website checksheet walk-by inspection* berbasis React Native dan Firebase untuk mendukung

condition-based monitoring. Pemanfaatan Firebase Realtime Database pada sistem ini membuat data checklist tersimpan dan tersinkronisasi secara langsung tanpa jeda, sehingga teknisi di lokasi yang berbeda dapat memantau status kondisi peralatan pada waktu yang bersamaan. Antarmuka React Native yang digunakan turut mendukung pengisian inspeksi lewat perangkat mobile sehingga fleksibilitas petugas di lapangan meningkat. Penelitian tersebut berhasil menekan waktu pencatatan manual dan meningkatkan efisiensi inspeksi, kendati cakupan pengujiannya masih terbatas pada satu unit kerja dan belum dilengkapi fitur pelaporan otomatis yang bisa diakses langsung oleh manajemen.

Suryawan (2024) membangun Dashboard Digital Quality Assurance di PT Dharma Polimetal Tbk untuk memudahkan supervisor dalam memantau hasil inspeksi. Dashboard berbasis teknologi web dengan MySQL dan PHP ini menyajikan rekapitulasi hasil quality assurance secara visual melalui grafik dan tabel ringkasan, sehingga manajemen mampu mengenali tren kualitas produk lebih cepat dibandingkan laporan berbasis kertas. Tersedia pula fitur filter berdasarkan lini produksi dan periode waktu guna mendukung analisis komparatif antar periode. Meski demikian, sistem tersebut belum dilengkapi modul checklist digital harian yang memungkinkan petugas QC mengisi data inspeksi secara langsung, sehingga data pada dashboard masih bergantung pada proses input manual sebelumnya.

Rinaldi dan Purnomo (2022) membangun *Progressive Web Application* untuk preventive maintenance dengan menerapkan metode Agile. Arsitektur *Progressive Web Application* yang diadopsi membuat aplikasi dapat diakses melalui browser tanpa instalasi khusus, sekaligus tetap berfungsi dalam kondisi jaringan terbatas di lingkungan industri. Melalui sistem ini, teknisi dapat mencatat jadwal dan riwayat perawatan mesin secara digital, dilengkapi notifikasi berbasis jadwal yang mengingatkan teknisi sebelum batas waktu perawatan tiba. Penerapan pendekatan Agile memungkinkan fitur ditambahkan secara bertahap sesuai masukan pengguna, sehingga sistem dapat terus menyesuaikan diri dengan kebutuhan operasional yang berubah. Kendati demikian, fokus sistem ini terletak

pada manajemen perawatan mesin, bukan pengendalian mutu produk, sehingga belum mencakup fitur checklist inspeksi kualitas maupun mekanisme approve/reject yang diperlukan dalam proses quality control.

Berdasarkan tinjauan di atas, terlihat bahwa penelitian-penelitian sebelumnya belum menggabungkan checklist digital, notifikasi otomatis, dashboard monitoring real-time, dan pelaporan terotomatisasi dalam satu sistem yang terpadu. Kondisi inilah yang menjadi celah sekaligus kebaruan proyek ini, yaitu pengembangan aplikasi web e-checksheet berbasis Laravel dan Quasar Vue.js untuk mendukung digitalisasi proses quality control di PT XYZ.

Tabel berikut menyajikan perbandingan penelitian-penelitian sebelumnya yang menjadi dasar penelitian ini.

Table 2.1 Perbandingan Penelitian Sebelumnya

Peneliti/Tahun	Judul/Topik Penelitian	Metode/Teknologi	Kelebihan	Kekurangan/Celah
Hofmann et al. (2023)	Digital Quality Control System for Construction Projects	Web-based Inspection System, Digital Form Integration	Meningkatkan akurasi dan efisiensi inspeksi lapangan	Tidak memiliki notifikasi otomatis
Zhang et al. (2023)	Data-Driven Digital Quality Management in Multi-Level Manufacturing Systems	Data Analytics, Web Platform	Mengurangi human error, mendukung keputusan berbasis data	Tidak menyediakan dashboard real-time
Santoso & Ramadhan (2024)	Development of <i>checksheet</i> Walk-by Inspection Website	React Native, Firebase	Efisien untuk condition-based monitoring	Pengujian terbatas, belum ada laporan otomatis

Peneliti/Tahun	Judul/Topik Penelitian	Metode/Teknologi	Kelebihan	Kekurangan/Celah
Suryawan (2024)	Perancangan Dashboard Digital Quality Assurance Berbasis Web	Web Dashboard, MySQL	Mempermudah pengawasan hasil inspeksi	Tidak memiliki modul <i>checksheet</i> digital
Rinaldi & Purnomo (2022)	Progressive Web Application for Preventive Maintenance	Agile Method, Laravel, MySQL	Mendukung multi-user, efisiensi pengembangan	Fokus pada perawatan mesin
Penelitian ini (2025)	Rancang Bangun Aplikasi Web <i>E-Checksheet</i> untuk Optimalisasi Proses QC pada PT XYZ	Laravel, Quasar Vue.js, Agile-Scrum	Integrasi <i>checksheet</i> digital, notifikasi otomatis, dashboard real-time, laporan otomatis	Menjawab celah penelitian sebelumnya dengan sistem <i>e-checksheet</i> terintegrasi

2.2 Landasan Teori

2.2.1 Quality Control (QC)

Quality Control merupakan proses sistematis yang ditujukan untuk memastikan produk atau jasa memenuhi standar yang telah ditetapkan. Besterfield et al. (2022) menguraikan bahwa QC meliputi rangkaian kegiatan inspeksi, evaluasi, dan tindakan korektif terhadap hasil produksi guna mendeteksi penyimpangan dari standar yang berlaku. Pada industri manufaktur, QC berperan strategis dalam menjaga konsistensi mutu, keandalan produk, dan kepuasan pelanggan secara berkelanjutan (Besterfield et al., 2022; Hofmann et al., 2023). Besterfield et al. (2022) berpandangan bahwa sistem pengendalian kualitas yang efektif tidak sebatas mendeteksi cacat, melainkan juga mencegahnya melalui penerapan kontrol proses statistik (Statistical Process Control/SPC), pemantauan berkala terhadap parameter kritis, serta penggunaan alat dasar kualitas seperti

checksheet, diagram Pareto, dan diagram sebab-akibat. Melalui pendekatan Total Quality Management (TQM), Besterfield et al. (2022) menegaskan bahwa keterlibatan seluruh elemen organisasi, dari operator produksi hingga manajemen puncak, menjadi kunci keberlanjutan sistem QC. Prinsip tersebut relevan dengan sistem *E-checksheet* pada penelitian ini, di mana ketiga peran pengguna — Admin, Petugas QC, dan Supervisor — memikul tanggung jawab masing-masing dalam menjaga kualitas data inspeksi secara real-time.

2.2.2 *Checksheet* dalam Quality Control

Checksheet termasuk salah satu alat dasar pengendalian mutu (Seven Basic Tools of Quality) yang berfungsi mencatat hasil pemeriksaan atas parameter tertentu. Sejumlah keterbatasan *checksheet* kertas, seperti risiko kehilangan data dan sulitnya pengarsipan, dapat diatasi melalui digitalisasi *checksheet* berbasis web yang mendukung pencatatan sekaligus pemantauan secara real-time (Santoso & Ramadhan, 2024).

2.2.3 Digitalisasi Quality Control dalam Industri 4.0

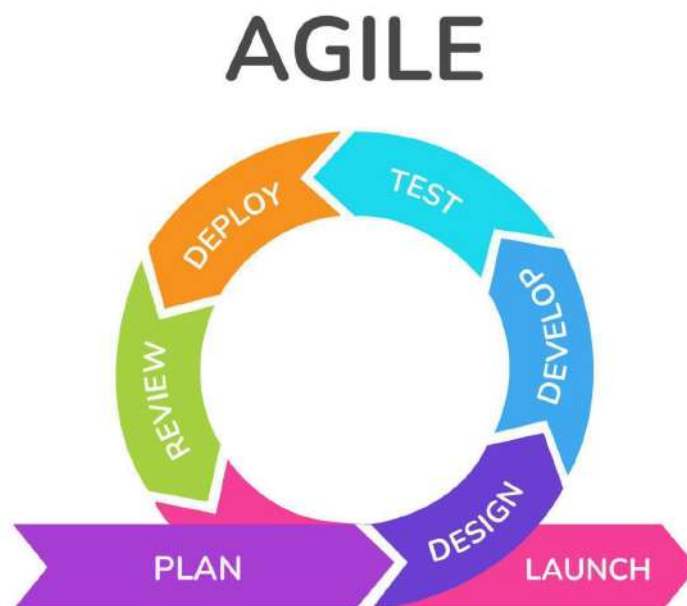
Industry 4.0 mengintegrasikan IoT, *Big Data Analytics*, dan AI untuk mendukung otomatisasi produksi (Lee et al., 2021). Lee et al. (2021) menerangkan bahwa dalam kerangka Industry 4.0, keterhubungan sistem Cyber-Physical memungkinkan data produksi dikumpulkan secara kontinu dari lantai pabrik menuju lapisan analitik berbasis cloud, sehingga keputusan terkait kualitas dapat diambil secara real-time berdasarkan data aktual, bukan sekadar estimasi manual. Digitalisasi pada proses QC pun membuka ruang bagi pengawasan berbasis data serta pengambilan keputusan yang lebih cepat (Zhang et al., 2023). Zhang et al. (2023) menambahkan bahwa penerapan platform manajemen mutu digital pada manufaktur bertingkat mampu menekan tingkat cacat produk secara signifikan melalui analitik prediktif yang mendeteksi potensi penyimpangan sebelum cacat produk terbentuk. Hal ini relevan dengan pengembangan sistem E-Checksheet yang menjadikan digitalisasi pencatatan

inspeksi sebagai langkah awal menuju pengendalian kualitas berbasis data di PT XYZ.

2.2.4 *E-Checksheet* Berbasis Web

E-checksheet merupakan bentuk pengembangan checksheet manual menjadi format digital berbasis web yang mendukung pencatatan, notifikasi otomatis, dan dashboard interaktif. Suryawan (2024) membuktikan bahwa penerapan E-checksheet mampu mendongkrak efisiensi hingga 30%. Adapun framework Laravel dan Quasar Vue.js dipilih karena mendukung antarmuka modern serta integrasi database secara real-time (Kim & Park, 2022).

2.3 Metode Pengembangan Produk



Ilustrasi berikut menggambarkan metode Agile-Scrum yang diterapkan dalam pengembangan sistem ini.

Gambar 2.1 Metode Agile-Scrum

Sumber: Krasamo

Pengembangan sistem ini menerapkan metode Agile Scrum, yakni kerangka kerja Agile yang mendukung pengembangan perangkat lunak secara iteratif dan inkremental melalui siklus yang disebut *sprint*. Tiap sprint menghasilkan *increment* produk yang bisa langsung digunakan dan dievaluasi oleh pengguna, sehingga tim pengembang memperoleh umpan balik dengan cepat dan dapat melakukan perbaikan secara berkelanjutan. Scrum juga memiliki sejumlah aktivitas inti — Sprint Planning, Daily Scrum, Sprint Review, dan Sprint Retrospective — yang diarahkan untuk memperkuat kolaborasi, transparansi, serta kemampuan tim beradaptasi terhadap perubahan kebutuhan selama pengembangan berlangsung. Selain itu, Scrum menetapkan *Definition of Done (DoD)* sebagai standar kualitas yang memastikan setiap increment telah memenuhi kriteria penyelesaian yang disepakati sebelum diserahkan kepada pengguna (Sassa et al., 2023).

Metode Agile Scrum dipilih dalam penelitian ini karena dinilai mampu mendukung pengembangan sistem E-Checksheet secara bertahap, memungkinkan evaluasi kebutuhan pengguna dilakukan secara berkala, serta memudahkan penyesuaian fitur apabila terjadi perubahan kebutuhan di tengah proses pengembangan.

Penerapan metode Agile-Scrum pada proyek ini mencakup tahapan berikut:

1. Product Backlog Creation

Tahap awal ini difokuskan pada identifikasi seluruh kebutuhan fungsional dan non-fungsional sistem berdasarkan hasil observasi serta wawancara dengan pihak PT XYZ. Kebutuhan yang terkumpul selanjutnya dikonversi menjadi user stories dan disusun sesuai prioritas bisnis ke dalam product backlog. Sebagai contoh, fitur pengisian checksheet digital, pengingat otomatis, pelaporan, dan dashboard real-time dimasukkan sebagai backlog utama (Verwijis & Russo, 2023).

2. Sprint Planning

Tim bersama *Product Owner* menentukan fitur yang akan dikerjakan pada satu sprint berdurasi 2–4 minggu. Pada tahap ini pula ditetapkan Definition of Done (DoD) yang jelas untuk tiap backlog agar kriteria penyelesaian pekerjaan dapat diukur (Verwijis & Russo, 2023). Verwijis & Russo (2023) menegaskan bahwa Sprint Planning yang efektif perlu memastikan setiap item Product Backlog yang dipilih memiliki kriteria penerimaan yang terukur dan dapat diverifikasi, sehingga tim pengembang dan Product Owner memiliki kesamaan pemahaman mengenai definisi "selesai". Pada proyek ini, Sprint Planning dilaksanakan sebelum setiap sprint dimulai untuk menetapkan fitur yang akan dikembangkan — seperti modul pengisian checksheet, sistem notifikasi, atau dashboard monitoring — beserta target fungsionalitas yang harus tercapai pada akhir sprint.

Melalui penelitian selama tujuh tahun terhadap lebih dari 2.000 tim Scrum, Verwijis & Russo (2023) menyimpulkan bahwa efektivitas tim Scrum sangat ditentukan oleh lima faktor utama: responsiveness, stakeholder concern, continuous improvement, team autonomy, dan management support. Masih dalam konteks Sprint Planning, Verwijis & Russo (2023) menyoroti pentingnya Definition of Done (DoD) sebagai komitmen bersama tim yang menghadirkan standar penyelesaian terukur dan transparan. Konsistensi penerapan DoD membantu tim menghindari penumpukan technical debt sekaligus memastikan setiap backlog item yang masuk ke dalam sprint benar-benar siap diserahkan kepada pengguna pada akhir sprint. Prinsip tersebut diterapkan pada pengembangan sistem E-Checksheet, di mana suatu fitur dinyatakan selesai apabila telah lolos pengujian fungsional, terintegrasi ke branch utama, dan dapat berjalan di lingkungan lokal tanpa error.

3. Sprint Execution & Daily Stand-up

Implementasi fitur berlangsung sepanjang sprint, dengan tim mengadakan Daily Stand-up Meeting setiap hari selama ± 15 menit untuk membahas progres, kendala, dan rencana kerja berikutnya. Pendekatan ini terbukti memperkuat

transparansi dan kolaborasi tim (Moe, Smite, & Ågerfalk, 2021). Moe, Smite, dan Ågerfalk (2021) menekankan bahwa sinkronisasi harian lewat Daily Stand-up menjadi mekanisme koordinasi yang sangat efektif, baik pada tim terdistribusi maupun kolokasi, karena setiap anggota dapat mengetahui perkembangan pekerjaan rekan lain dan segera mengenali hambatan teknis sebelum berdampak pada jadwal sprint. Pada pengembangan sistem E-Checksheet, Daily Stand-up dijalankan secara mandiri oleh pengembang dengan mencatat progres harian, kendala yang dijumpai (seperti masalah integrasi API atau validasi form), serta rencana kerja untuk hari berikutnya.

4. Sprint Review

Begitu sprint berakhir, hasil increment sistem didemonstrasikan kepada pengguna guna memperoleh umpan balik secara langsung. Supervisor dan staf QC PT XYZ menilai kesesuaian fitur yang dikembangkan dengan kebutuhan mereka, sejalan dengan pandangan Verwijns & Russo (2023) bahwa Sprint Review menjadi mekanisme kunci yang menjembatani tim pengembang dengan pemangku kepentingan agar produk berkembang sesuai kebutuhan nyata. Verwijns & Russo (2023) juga menemukan bahwa tim yang konsisten menggelar Sprint Review dan memperoleh umpan balik nyata dari pengguna cenderung menghasilkan tingkat kepuasan pengguna akhir yang lebih tinggi dibandingkan tim yang melewatkan atau mempersingkat tahap tersebut, karena Sprint Review membantu mencegah kesenjangan antara ekspektasi pengguna dan produk yang dibangun. Pada proyek ini, Sprint Review dilaksanakan bersama pembimbing sebagai simulasi evaluasi oleh pengguna representatif dari PT XYZ, guna memastikan setiap fitur yang dirilis benar-benar memenuhi kebutuhan operasional yang sesungguhnya.

Verwijns & Russo (2023) turut menegaskan bahwa Sprint Review berfungsi sebagai mekanisme umpan balik berkelanjutan; tim Scrum yang secara rutin melibatkan pemangku kepentingan dalam Sprint Review cenderung memiliki kepuasan pengguna yang lebih tinggi dan lebih jarang menghadapi perubahan besar menjelang akhir proyek. Pada penelitian ini, Sprint Review dilaksanakan bersama pembimbing selaku representasi pengguna, di mana tiap fitur

E-Checksheet yang rampung didemonstrasikan langsung untuk memperoleh masukan. Umpan balik yang terkumpul selanjutnya dipakai untuk memperbarui product backlog dan merumuskan sprint goal berikutnya, sehingga pengembangan sistem senantiasa berorientasi pada kebutuhan aktual pengguna di PT XYZ.

5. Sprint Retrospective

Tahap ini menjadi ruang refleksi internal tim untuk membahas hal-hal yang telah berjalan baik, aspek yang perlu diperbaiki, dan langkah peningkatan ke depan. Praktik retrospektif semacam ini membantu tim terus menyempurnakan proses kerja dan menekan hambatan teknis (Verwijis & Russo, 2023). Pada pengembangan sistem E-Checksheet, Sprint Retrospective dilaksanakan di setiap akhir sprint untuk mengevaluasi efektivitas proses pengembangan sekaligus mengidentifikasi area perbaikan pada sprint selanjutnya.

6. Product Increment & Release

Hasil tiap sprint kemudian digabungkan menjadi increment produk akhir. Setelah seluruh fitur utama rampung, sistem diuji melalui *Black Box Testing* oleh pengguna akhir, dan produk dirilis sebagai versi siap pakai di lingkungan produksi PT XYZ apabila seluruh kriteria telah terpenuhi (Verwijis & Russo, 2023). Verwijis & Russo (2023) menekankan bahwa konsistensi penerapan prinsip Potentially Shippable Increment mendorong tim menjaga kualitas kode dan dokumentasi pada tiap sprint, sebab produk mesti selalu berada dalam kondisi siap rilis tanpa memerlukan pekerjaan tambahan yang berarti. Praktik ini juga mendisiplinkan tim agar tidak menunda pengujian hingga akhir proyek, melainkan menjalankan pengujian fungsional secara inkremental pada setiap sprint. Pada proyek E-Checksheet, prinsip tersebut diwujudkan dengan memastikan setiap sprint menghasilkan modul yang telah diuji secara fungsional di lingkungan XAMPP lokal sebelum sprint berikutnya dimulai.

2.4 Studi Kasus: Kondisi Awal Proses Quality Control di PT XYZ

PT XYZ adalah perusahaan manufaktur elektronik yang memproduksi komponen presisi untuk kebutuhan otomotif dan perangkat digital. Setiap lini produksinya wajib menjalani pemeriksaan kualitas (quality control/QC) secara rutin menggunakan lembar checksheet, yang memuat parameter mutu produk seperti dimensi, nilai toleransi, jumlah cacat (defect), dan hasil inspeksi visual yang harus diisi petugas QC pada setiap shift. Kendati demikian, hingga penelitian ini dilakukan, PT XYZ masih mengandalkan sistem checksheet berbasis kertas untuk mendokumentasikan seluruh hasil inspeksinya.

2.4.1 Permasalahan Utama Sistem Manual di PT XYZ

Berikut permasalahan utama yang timbul dari sistem manual di PT XYZ:

1. Kehilangan dan kerusakan data. Dokumen kertas rentan terpengaruh faktor lingkungan seperti kelembapan, tumpahan cairan, atau sobek akibat penanganan sehari-hari.
2. Pemborosan waktu dan tenaga. Tiap shift menghasilkan rata-rata 50–70 lembar checksheet yang seluruhnya harus direkap secara manual, memakan waktu hingga 2–3 jam per hari.
3. Kendala dalam analisis dan pelaporan. Supervisor sulit melakukan analisis tren secara cepat karena data masih tersimpan dalam bentuk fisik.
4. Minimnya monitoring real-time. Sistem manual tidak memberi ruang bagi supervisor untuk mengetahui status pengisian checksheet secara langsung.
5. Absennya sistem pengingat otomatis. Operator tidak menerima notifikasi saat jadwal inspeksi tiba, sehingga sejumlah parameter kritis kerap terlewat.

2.4.2 Dampak Terhadap Efisiensi Perusahaan

Hasil wawancara internal menunjukkan bahwa sistem manual memicu ketidaktepatan pelaporan hingga $\pm 15\%$ akibat kesalahan input maupun dokumen

yang hilang. Proses audit internal oleh bagian Quality Assurance pun menjadi kurang efisien karena staf harus menelusuri dokumen dari ribuan lembar arsip.

2.4.3 Kesimpulan Studi Kasus

Uraian tersebut memperlihatkan bahwa penggunaan checksheet kertas di PT XYZ menimbulkan berbagai persoalan terkait efisiensi, keandalan data, dan kecepatan pengawasan. Karena itu, digitalisasi melalui sistem E-checksheet berbasis web menjadi kebutuhan strategis untuk meningkatkan transparansi, akurasi, dan efektivitas proses QC. Manfaat digitalisasi ini pun tidak berhenti pada departemen QC semata, melainkan turut mendorong efisiensi pelaporan manajemen, mempermudah audit internal oleh bagian Quality Assurance, serta mempercepat pengambilan keputusan berbasis data di seluruh lini produksi PT XYZ.

BAB III ANALISIS DAN PERANCANGAN

3.1 Analisis Kebutuhan

Analisis kebutuhan merupakan tahap awal dalam pengembangan sistem yang bertujuan untuk mengidentifikasi dan mendokumentasikan seluruh kebutuhan yang harus dipenuhi oleh sistem yang akan dibangun. Pada tahap ini dilakukan pengumpulan informasi melalui observasi langsung terhadap proses quality control yang berjalan di PT XYZ, serta wawancara dengan pihak-pihak yang terlibat, meliputi petugas QC, supervisor, dan pihak manajemen. Hasil analisis ini kemudian digunakan sebagai dasar perancangan sistem *E-Checksheet* agar solusi yang dikembangkan benar-benar sesuai dengan kebutuhan operasional pengguna di lapangan.

3.1.1 Proses Bisnis Berjalan

PT XYZ saat ini masih menggunakan sistem *checksheet* berbasis kertas dalam kegiatan *quality control*. Setiap petugas QC mengisi formulir pemeriksaan manual untuk setiap batch produksi, kemudian menyerahkan lembar tersebut kepada bagian administrasi untuk diarsipkan. Proses ini menimbulkan beberapa permasalahan utama yaitu:

1. Inefisiensi waktu dan tenaga, karena staf harus menginput ulang data *checksheet* ke sistem Excel untuk pelaporan.
2. Kesulitan penelusuran data historis (*traceability*) akibat sistem arsip manual.
3. Tingginya risiko kehilangan atau kerusakan dokumen akibat faktor lingkungan dan human error.
4. Tidak adanya monitoring *real-time*, sehingga supervisor tidak dapat mengetahui status pengisian *checksheet* secara langsung.
5. Tidak terdapat sistem notifikasi otomatis yang mengingatkan petugas QC untuk mengisi *checksheet* sesuai jadwal inspeksi.

3.1.2 Proses Bisnis yang Diusulkan

Sebelum implementasi sistem *E-Checksheet*, proses bisnis quality control di PT XYZ berjalan secara manual: petugas QC mengisi formulir kertas secara manual untuk setiap shift, kemudian lembar tersebut diserahkan ke bagian administrasi untuk diarsipkan dan direkap secara manual ke spreadsheet. Proses ini membutuhkan waktu 2–3 jam per hari dan rentan terhadap kehilangan data serta keterlambatan pelaporan kepada supervisor. Setelah implementasi sistem *E-Checksheet*, alur proses bisnis berubah menjadi sepenuhnya digital: petugas QC mengisi form *checksheet* melalui antarmuka web, data tersimpan langsung ke basis data terpusat, supervisor menerima notifikasi real-time dan dapat melakukan approve atau reject secara langsung, serta laporan summary harian dan mingguan dihasilkan secara otomatis tanpa proses rekap manual.

Sebagai solusi terhadap permasalahan tersebut, dikembangkan Aplikasi Web *E-Checksheet* yang akan menggantikan sistem kertas menjadi sistem digital berbasis web. Sistem ini memungkinkan:

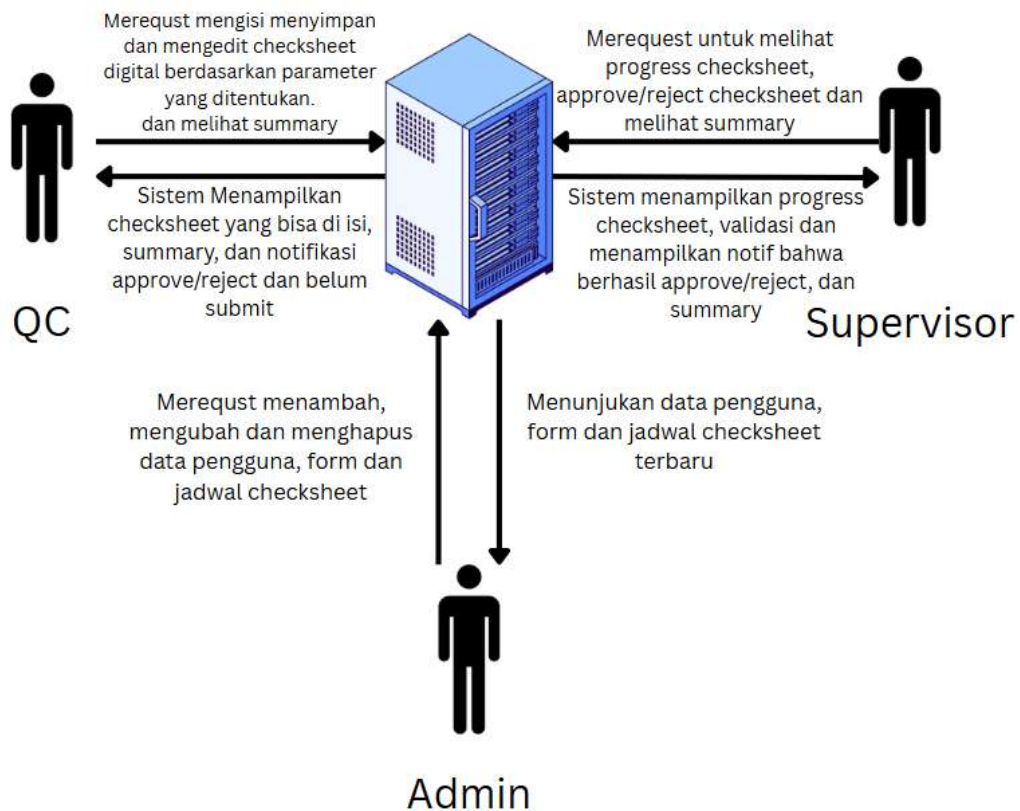
1. Petugas QC melakukan pengisian data inspeksi secara digital melalui form online.
2. Supervisor memantau status pengisian *checksheet* secara real-time.
3. Sistem memberikan notifikasi otomatis apabila jadwal inspeksi belum dilakukan.
4. Data hasil pemeriksaan tersimpan langsung ke basis data terpusat.

3.1.3 Gambaran Umum Sistem

Secara umum, sistem *E-Checksheet* terdiri atas tiga peran pengguna utama yaitu:

1. Admin – mengelola data pengguna, jadwal inspeksi, dan parameter QC.
2. Petugas QC (Operator) – mengisi form *checksheet* digital harian sesuai shift.

3. *Supervisor* – memantau hasil pengisian, memberikan persetujuan (approve/reject), dan memantau laporan.



Berikut ini adalah gambaran umum arsitektur sistem *E-Checksheet* beserta ketiga peran pengguna yang terlibat.

Gambar 3.1 Gambaran Umum Sistem

3.1.4 Teknologi yang Digunakan

Pengembangan sistem *E-Checksheet* ini menggunakan sejumlah teknologi yang dipilih berdasarkan kesesuaian dengan kebutuhan fungsional sistem, kemudahan integrasi antar komponen, serta ketersediaan dukungan komunitas yang aktif.

a. PHP (*Hypertext Preprocessor*)

PHP merupakan bahasa pemrograman sisi server (server-side scripting language) yang banyak digunakan dalam pengembangan aplikasi web dinamis. PHP memiliki kemampuan untuk berinteraksi dengan berbagai sistem basis data, mendukung pengembangan aplikasi berbasis web secara cepat, serta memiliki komunitas pengembang yang luas sehingga memudahkan proses pemeliharaan dan pengembangan sistem. Pada penelitian ini, PHP digunakan sebagai bahasa utama dalam pengembangan backend sistem *E-Checksheet* (Pratama & Wijaya, 2023).

b. Laravel (Backend Framework)

Laravel merupakan framework berbasis PHP yang menerapkan arsitektur Model-View-Controller (MVC) untuk memisahkan logika bisnis, antarmuka pengguna, dan pengelolaan data. Laravel menyediakan berbagai fitur seperti routing, middleware, autentikasi, ORM Eloquent, serta dukungan RESTful API yang dapat mempercepat proses pengembangan aplikasi web. Oleh karena itu, Laravel digunakan sebagai framework backend dalam pengembangan sistem *E-Checksheet* karena mampu menghasilkan kode yang lebih terstruktur, mudah dipelihara, dan aman (Saputra et al., 2023).

c. Vue.js dan Quasar Framework (Frontend)

Vue.js merupakan framework JavaScript yang digunakan untuk membangun antarmuka pengguna yang interaktif dan reaktif. Vue.js memungkinkan pembaruan data secara real-time pada tampilan tanpa perlu memuat ulang halaman secara keseluruhan. Dalam penelitian ini, Vue.js dikombinasikan dengan Quasar Framework yang menyediakan berbagai komponen antarmuka siap pakai sehingga mempercepat proses pengembangan aplikasi web dengan tampilan yang responsif pada berbagai ukuran perangkat. Kombinasi kedua teknologi tersebut mendukung pembangunan sistem

E-Checksheet yang modern dan mudah digunakan oleh pengguna (Hidayat & Nugroho, 2023).

d. MySQL (Database Management System)

MySQL merupakan sistem manajemen basis data relasional (Relational Database Management System/RDBMS) yang digunakan untuk menyimpan, mengelola, dan memproses data secara terstruktur. MySQL mendukung pengelolaan data dalam jumlah besar, memiliki performa yang baik, serta mendukung integrasi dengan berbagai bahasa pemrograman dan framework pengembangan web. Pada sistem *E-Checksheet*, MySQL digunakan sebagai media penyimpanan data pengguna, data inspeksi, jadwal pemeriksaan, hasil pengisian *checksheet*, dan data notifikasi sistem (Rahman & Setiawan, 2023).

Ringkasan seluruh teknologi yang digunakan dalam pengembangan sistem *E-Checksheet* dapat dilihat pada tabel berikut.

Table 3.1 Ringkasan Teknologi yang Digunakan

Komponen	Teknologi	Versi	Fungsi dalam Sistem
Bahasa Pemrograman	PHP	8.2	Bahasa utama pengembangan backend server-side
Backend Framework	Laravel	10	RESTful API, autentikasi, notifikasi terjadwal, manajemen hak akses
Frontend Framework	Vue.js + Quasar	3 / 2	Antarmuka pengguna SPA, dashboard monitoring, form <i>checksheet</i> digital
Database	MySQL	8.0	Penyimpanan data pengguna, <i>checksheet</i> , hasil inspeksi, jadwal, log notifikasi

Komponen	Teknologi	Versi	Fungsi dalam Sistem
Autentikasi	Laravel Sanctum	-	Token-based authentication, session management, role-based access control

3.2 Perancangan

3.2.1 Kebutuhan Fungsional

Adapun kebutuhan fungsional yang akan dibuat sebagai berikut:

Table 3.2 Kebutuhan Fungsional

Kode	Role	Deskripsi Kebutuhan Fungsional
FR-01	Semua	Sistem memungkinkan pengguna melakukan login dan logout dengan hak akses sesuai peran (Admin, QC, Supervisor).
FR-02	Admin	Admin dapat menambah, mengubah, dan menghapus data pengguna.
FR-03	Admin	Admin dapat menambah, mengubah, dan menghapus form serta jadwal inspeksi.
FR-04	Supervisor	Supervisor dapat melakukan Approve/Reject terhadap submission dari petugas QC.

Kode	Role	Deskripsi Kebutuhan Fungsional
FR-05	Supervisor	Supervisor dapat memantau status pengisian <i>checksheet</i> secara real-time melalui dashboard.
FR-06	QC	QC dan Supervisor dapat melihat summary harian dan mingguan.
FR-07	QC	Petugas QC dapat mengisi, menyimpan, dan mengedit <i>checksheet</i> digital berdasarkan parameter yang ditentukan.
FR-08	Sistem	Sistem mengirimkan notifikasi otomatis kepada petugas QC jika <i>checksheet</i> belum diisi sesuai jadwal.
FR-09	Sistem	Sistem mengirimkan notifikasi otomatis kepada petugas QC jika <i>checksheet</i> direject dan harus direvisi.

3.2.2 Kebutuhan Non-Fungsional

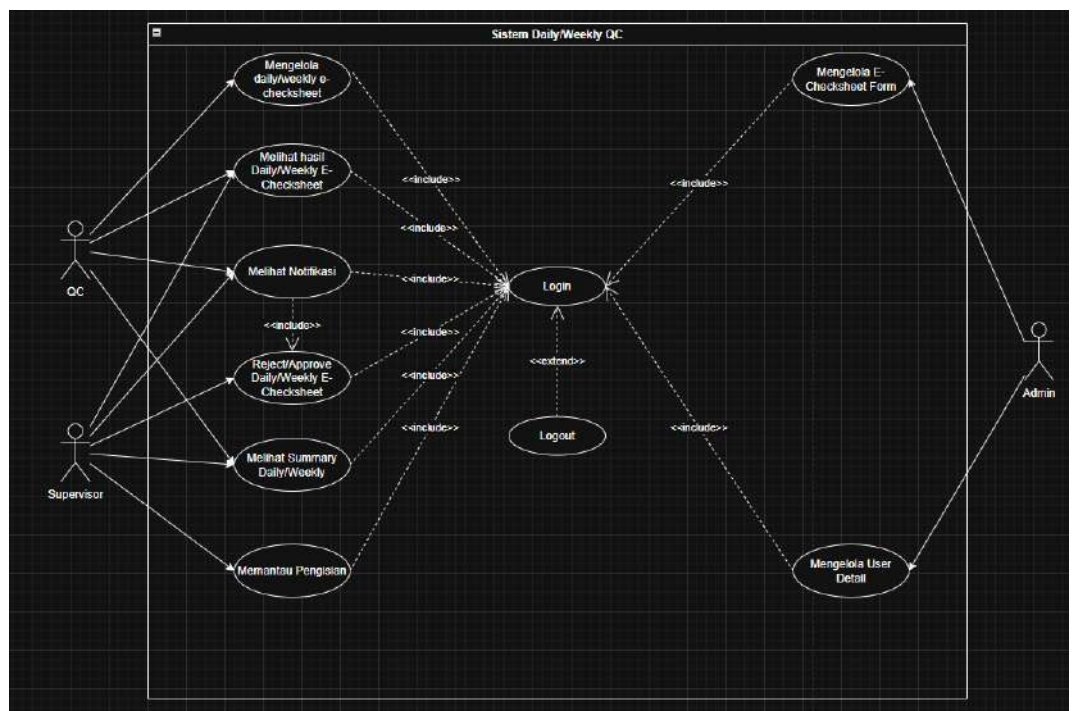
Adapun kebutuhan non-fungsional sistem yang harus dipenuhi adalah sebagai berikut.

Table 3.3 Kebutuhan Non-Fungsional

Kode	Deskripsi Kebutuhan Non-Fungsional
NFR-01	Seluruh teks antarmuka sistem (label, tombol, pesan kesalahan, dan notifikasi) menggunakan Bahasa Indonesia formal sesuai PUEBI.
NFR-02	Waktu respon sistem maksimal 3 detik per permintaan untuk setiap operasi pengisian, penyimpanan, dan pemuatan halaman.

Kode	Deskripsi Kebutuhan Non-Fungsional
NFR-03	Sistem harus menerapkan autentikasi berbasis token menggunakan Laravel Sanctum. Sesi pengguna otomatis berakhir (expired) setelah 8 jam tidak aktif.
NFR-04	Pengguna baru (petugas QC non-teknis) harus dapat menyelesaikan pengisian <i>checksheet</i> digital untuk pertama kalinya tanpa pelatihan khusus dalam waktu tidak lebih dari 10 menit.
NFR-05	Sistem harus dapat berjalan dengan baik pada browser Google Chrome, Mozilla Firefox , dan Microsoft Edge.

3.2.3 Diagram Use Case



Berikut ini adalah Diagram Use Case yang menggambarkan interaksi antara aktor dan sistem *E-Checksheet*.

Gambar 3.2 Diagram Use Case

Use Case Diagram pada sistem *E-Checksheet* melibatkan tiga aktor utama, yaitu Admin, Petugas QC, dan Supervisor. Admin bertanggung jawab dalam mengelola data pengguna, form inspeksi, serta jadwal inspeksi. Petugas QC bertugas mengisi dan mengirimkan *checksheet* digital berdasarkan jadwal yang telah ditentukan. Supervisor bertanggung jawab melakukan pemantauan pengisian

checksheet, memberikan persetujuan (approve) atau penolakan (reject), serta melihat laporan dan ringkasan hasil inspeksi. Selain ketiga aktor tersebut, sistem juga menjalankan fungsi otomatis berupa pengiriman notifikasi pengingat dan notifikasi hasil review *checksheet*.

3.2.4 Skenario Use Case

Berikut adalah skenario use case utama pada sistem *E-Checksheet*:

a. Skenario Use Case Login

Table 3.4 Skenario Use Case Login

Komponen	Keterangan
Nama Use Case	Login
Aktor	Admin, Petugas QC, Supervisor
Deskripsi	Pengguna melakukan autentikasi untuk masuk ke sistem
Kondisi Awal	Pengguna belum login, berada di halaman login
Alur Utama	1. Pengguna membuka halaman login 2. Memasukkan email dan password 3. Sistem memvalidasi kredensial 4. Sistem menghasilkan token autentikasi 5. Pengguna diarahkan ke dashboard sesuai peran
Alur Alternatif	Jika email atau password salah, sistem menampilkan pesan kesalahan
Kondisi Akhir	Pengguna berhasil masuk dan dapat mengakses fitur sesuai perannya

b. Skenario Use Case Pengisian *Checksheet*

Table 3.5 Skenario Use Case Pengisian *Checksheet*

Komponen	Keterangan
Nama Use Case	Isi <i>Checksheet</i>
Aktor	Petugas QC
Deskripsi	Petugas QC mengisi form <i>checksheets</i> digital sesuai jadwal shift
Kondisi Awal	Petugas QC telah login dan ada jadwal aktif hari ini
Alur Utama	1. Petugas QC membuka halaman Isi <i>Checksheets</i> 2. Sistem menampilkan daftar jadwal hari ini 3. Petugas memilih jadwal dan mengisi nilai aktual tiap parameter 4. Petugas memilih hasil (OK/NG/NA) untuk setiap parameter 5. Petugas menyimpan draft atau langsung submit ke supervisor
Alur Alternatif	Petugas dapat menyimpan sebagai draft jika belum selesai dan melanjutkan kemudian
Kondisi Akhir	<i>Checksheets</i> tersimpan dengan status Draft atau Submitted

c. Skenario Use Case Approve/Reject *Checksheets*

Table 3.6 Skenario Use Case Approve/Reject *Checksheets*

Komponen	Keterangan
Nama Use Case	Approve/Reject <i>Checksheets</i>
Aktor	Supervisor
Deskripsi	Supervisor mereview dan memberikan keputusan terhadap <i>checksheets</i> yang disubmit
Kondisi Awal	Ada <i>checksheets</i> dengan status Submitted
Alur Utama	1. Supervisor membuka Monitor <i>Checksheets</i> 2. Memilih <i>checksheets</i> yang perlu direview 3. Melihat detail parameter dan nilai aktual 4. Memberikan keputusan Approve atau Reject 5. Sistem mengirim notifikasi ke petugas QC

Komponen	Keterangan
Alur Alternatif	Jika Reject, supervisor wajib mengisi catatan alasan penolakan
Kondisi Akhir	Status <i>checksheet</i> berubah menjadi Approved atau Rejected, QC mendapat notifikasi

d. Skenario Use Case Kelola Pengguna

Table 3.7 Skenario Use Case Kelola Pengguna

Komponen	Keterangan
Nama Use Case	Kelola Pengguna
Aktor	Admin
Deskripsi	Admin menambah, mengubah peran, mengaktifkan, atau menonaktifkan akun pengguna sistem
Kondisi Awal	Admin telah login dan berada di halaman Manajemen Pengguna
Alur Utama	1. Admin membuka halaman Manajemen Pengguna 2. Sistem menampilkan daftar seluruh pengguna beserta peran dan statusnya 3. Admin memilih tindakan: Tambah, Edit, Aktifkan, atau Nonaktifkan 4. Admin mengisi atau mengubah data yang diperlukan 5. Admin menyimpan perubahan 6. Sistem memperbarui data dan menampilkan konfirmasi
Alur Alternatif	Jika email yang dimasukkan sudah terdaftar, sistem menampilkan pesan kesalahan dan tidak menyimpan data
Kondisi Akhir	Data pengguna berhasil ditambah, diubah, atau statusnya diperbarui sesuai tindakan Admin

e. Skenario Use Case Kelola Form dan Jadwal Inspeksi

Table 3.8 Skenario Use Case Kelola Form dan Jadwal Inspeksi

Komponen	Keterangan
Nama Use Case	Kelola Form dan Jadwal Inspeksi
Aktor	Admin
Deskripsi	Admin membuat template form inspeksi beserta parameter pemeriksaan, kemudian menetapkan jadwal inspeksi per shift
Kondisi Awal	Admin telah login dan berada di halaman Form Inspeksi atau Jadwal Inspeksi
Alur Utama	1. Admin membuka halaman Form Inspeksi 2. Admin menambahkan form baru dengan nama, lini produksi, dan parameter 3. Admin menyimpan form 4. Admin membuka halaman Jadwal Inspeksi 5. Admin memilih form, shift, jam mulai, jam selesai, dan jam pengingat 6. Admin menyimpan jadwal 7. Sistem menyimpan jadwal dan scheduler otomatis terkonfigurasi
Alur Alternatif	Jika jam selesai lebih awal dari jam mulai, sistem menampilkan validasi kesalahan dan tidak menyimpan jadwal
Kondisi Akhir	Form inspeksi dan jadwal berhasil tersimpan; sistem scheduler akan mengirimkan notifikasi pengingat sesuai jam yang ditetapkan

f. Skenario Use Case Monitor *Checksheets***Table 3.9 Skenario Use Case Monitor *Checksheets***

Komponen	Keterangan
Nama Use Case	Monitor <i>Checksheets</i>
Aktor	Supervisor

Komponen	Keterangan
Deskripsi	Supervisor memantau seluruh <i>checksheet</i> yang telah disubmit oleh petugas QC secara real-time
Kondisi Awal	Supervisor telah login dan terdapat <i>checksheet</i> dengan status Menunggu Review atau status lainnya
Alur Utama	- Supervisor membuka halaman Monitor <i>Checksheet</i> - Sistem menampilkan daftar seluruh <i>checksheet</i> beserta status masing-masing - Supervisor dapat menggunakan filter berdasarkan status atau tanggal - Supervisor memilih <i>checksheet</i> untuk melihat detail parameter pemeriksaan
Alur Alternatif	Jika belum ada <i>checksheet</i> yang disubmit, halaman menampilkan pesan data kosong
Kondisi Akhir	Supervisor dapat melihat status dan detail seluruh <i>checksheet</i> secara real-time

g. Skenario Use Case Lihat Summary

Table 3.10 Skenario Use Case Lihat Summary

Komponen	Keterangan
Nama Use Case	Lihat Summary
Aktor	Petugas QC, Supervisor
Deskripsi	Pengguna melihat rekapitulasi statistik hasil inspeksi harian dan mingguan
Kondisi Awal	Pengguna telah login dan terdapat data <i>checksheet</i> yang telah disubmit
Alur Utama	- Pengguna membuka halaman Summary - Sistem menampilkan statistik total <i>checksheet</i>, jumlah disetujui, ditolak, dan menunggu review

Komponen	Keterangan
	3. Sistem menampilkan distribusi hasil inspeksi OK/NG/NA dalam bentuk progress bar 4. Sistem menampilkan grafik tren 7 hari terakhir
Alur Alternatif	Jika belum ada data <i>checksheet</i> , grafik dan statistik menampilkan nilai nol
Kondisi Akhir	Pengguna dapat melihat rekapitulasi data inspeksi harian dan mingguan secara akurat

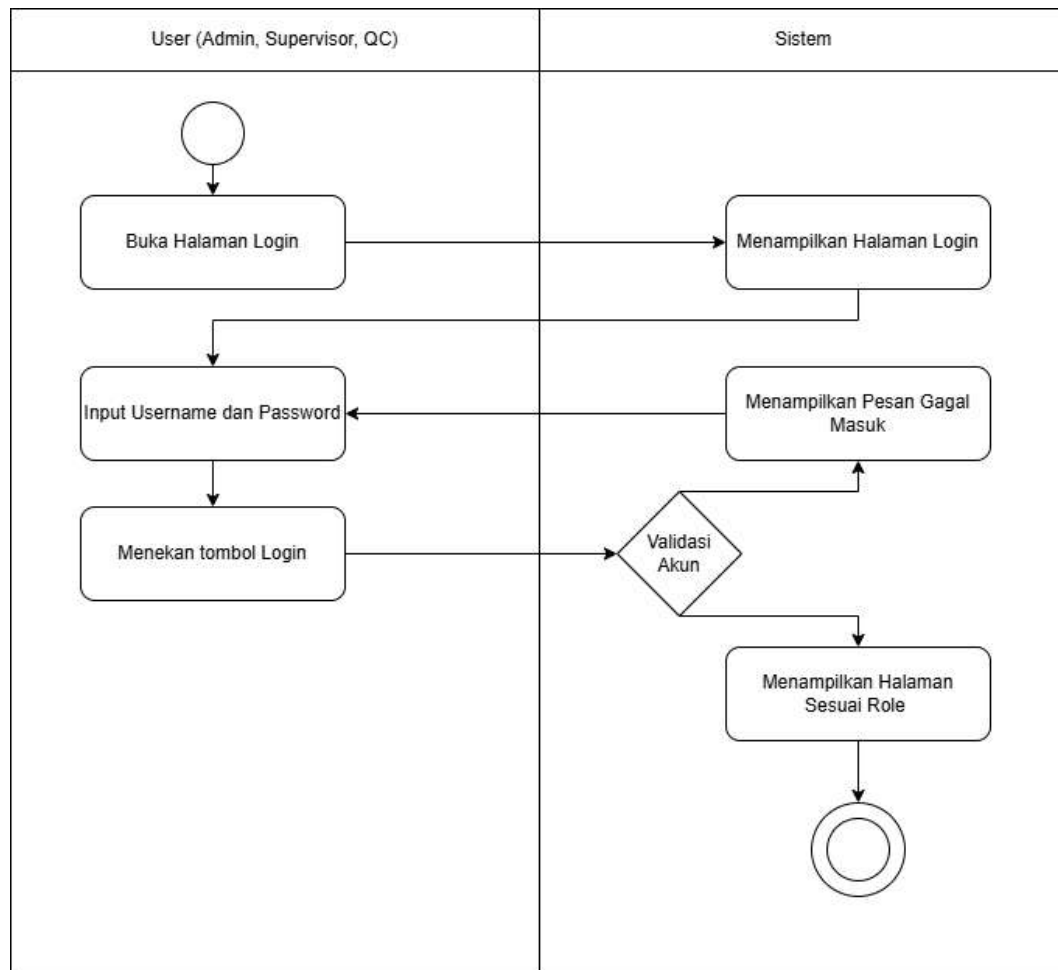
h. Skenario Use Case Notifikasi Otomatis

Tabel 3.11 Skenario Use Case Notifikasi Otomatis

Komponen	Keterangan
Nama Use Case	Notifikasi Otomatis
Aktor	Sistem (Laravel Scheduler)
Deskripsi	Sistem secara otomatis mengirimkan notifikasi pengingat kepada petugas QC jika <i>checksheet</i> belum diisi saat jam pengingat tiba, serta notifikasi hasil review dari supervisor
Kondisi Awal	Jadwal inspeksi telah dikonfigurasi dan jam pengingat telah tiba; atau supervisor telah memberikan keputusan approve/reject
Alur Utama	1. Laravel Scheduler memeriksa jadwal aktif pada jam pengingat yang ditentukan 2. Sistem mengidentifikasi petugas QC yang belum mengisi <i>checksheet</i> untuk jadwal tersebut 3. Sistem mengirimkan notifikasi pengingat ke akun QC yang bersangkutan 4. Notifikasi ditampilkan pada ikon lonceng di navbar dengan badge merah
Alur Alternatif	Jika <i>checksheet</i> sudah diisi sebelum jam pengingat, sistem tidak mengirimkan notifikasi. Jika supervisor melakukan approve/reject,

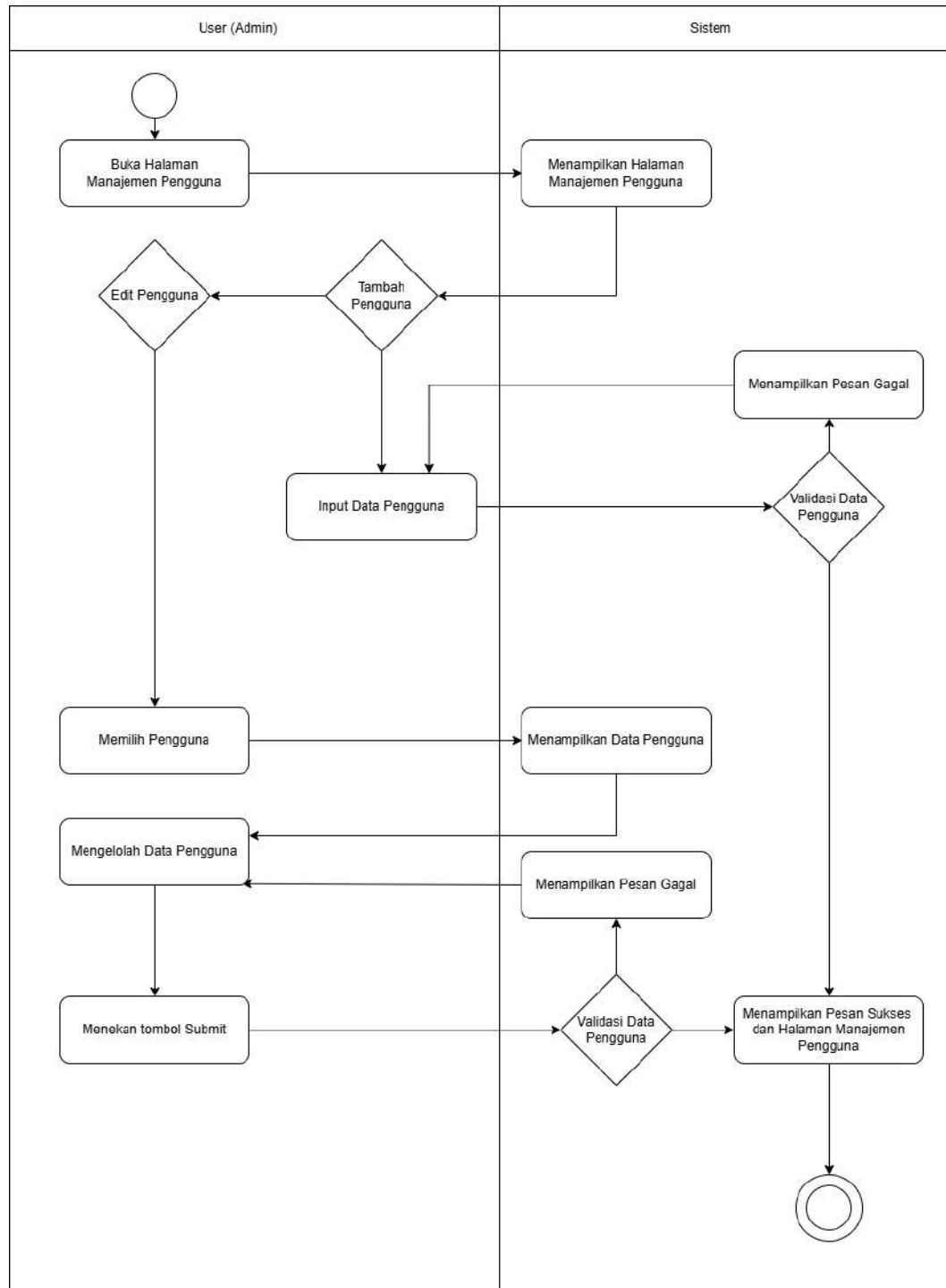
Komponen	Keterangan
	sistem mengirimkan notifikasi hasil review beserta catatan alasan (jika reject)
Kondisi Akhir	Petugas QC menerima notifikasi pengingat atau notifikasi hasil review; badge merah pada ikon lonceng menunjukkan jumlah notifikasi yang belum dibaca

3.2.5 Activity Diagram



Gambar 3.3 Activity Diagram Login

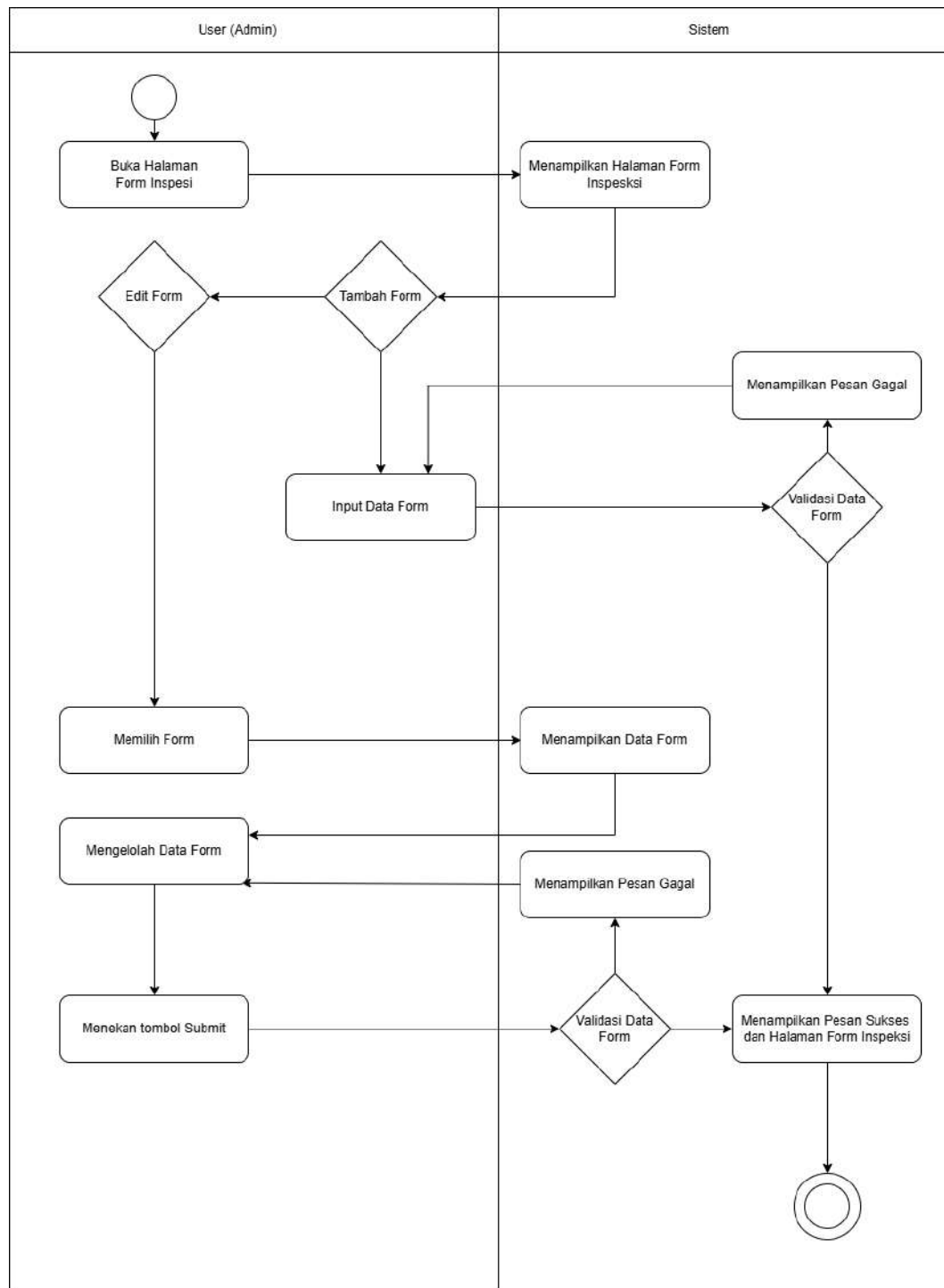
Activity Diagram Login menggambarkan alur proses autentikasi pengguna ke dalam sistem. Setelah memasukkan kredensial yang valid, sistem memverifikasi data pengguna dan menentukan hak akses berdasarkan peran (role) yang ditetapkan. Pengguna dengan peran Admin akan diarahkan ke Dashboard Admin, Petugas QC ke Dashboard QC, dan Supervisor ke Dashboard Supervisor. Apabila kredensial tidak valid, sistem menampilkan pesan kesalahan dan pengguna diminta memasukkan ulang data login.



Gambar 3.4 Activity Diagram Admin Kelola Pengguna

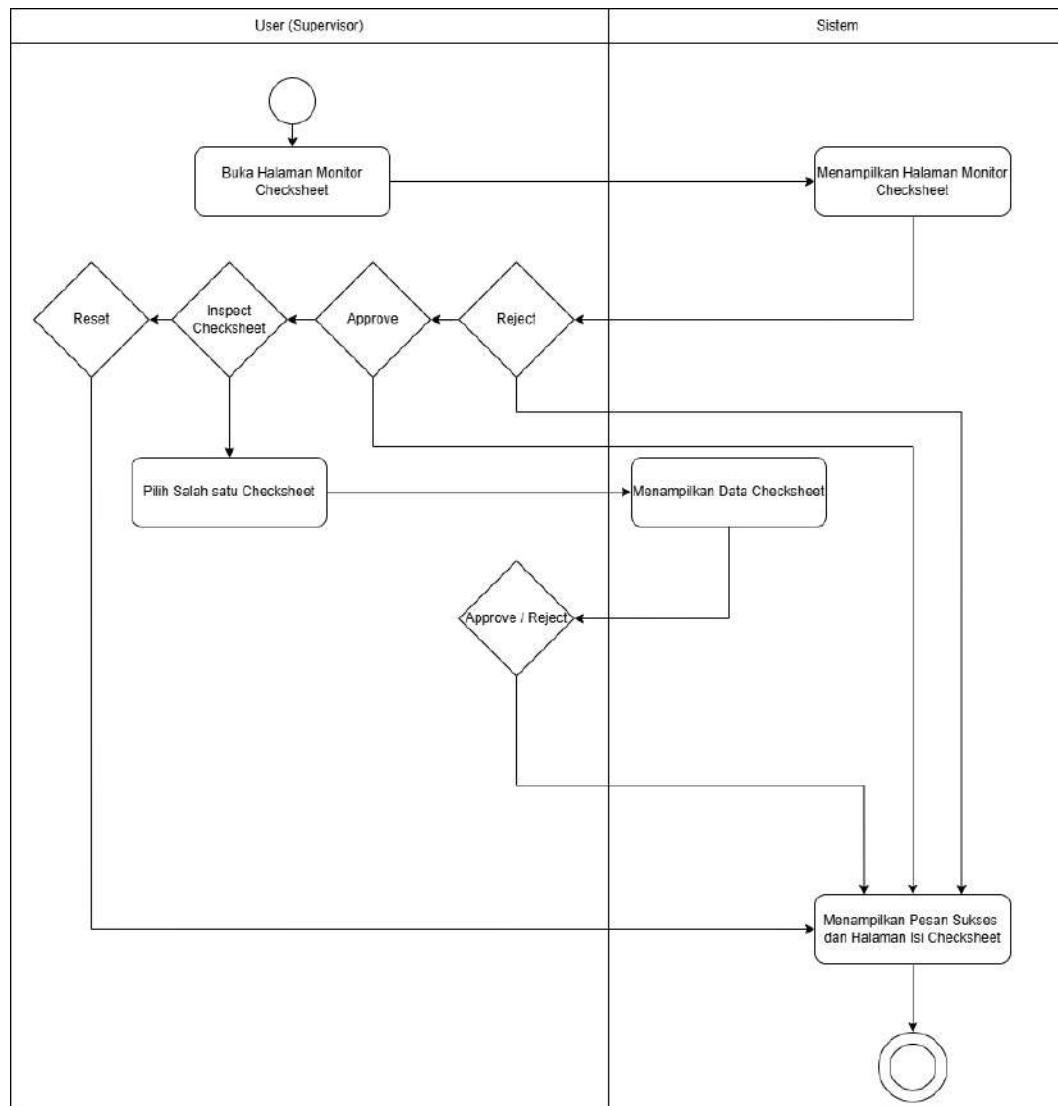
Activity Diagram Admin Kelola Pengguna menggambarkan alur proses pengelolaan akun pengguna oleh Admin. Admin dapat menambahkan pengguna baru dengan mengisi data nama, email, password, dan peran (role), mengubah

peran pengguna yang sudah ada, serta mengaktifkan atau menonaktifkan akun pengguna sesuai kebutuhan operasional sistem.



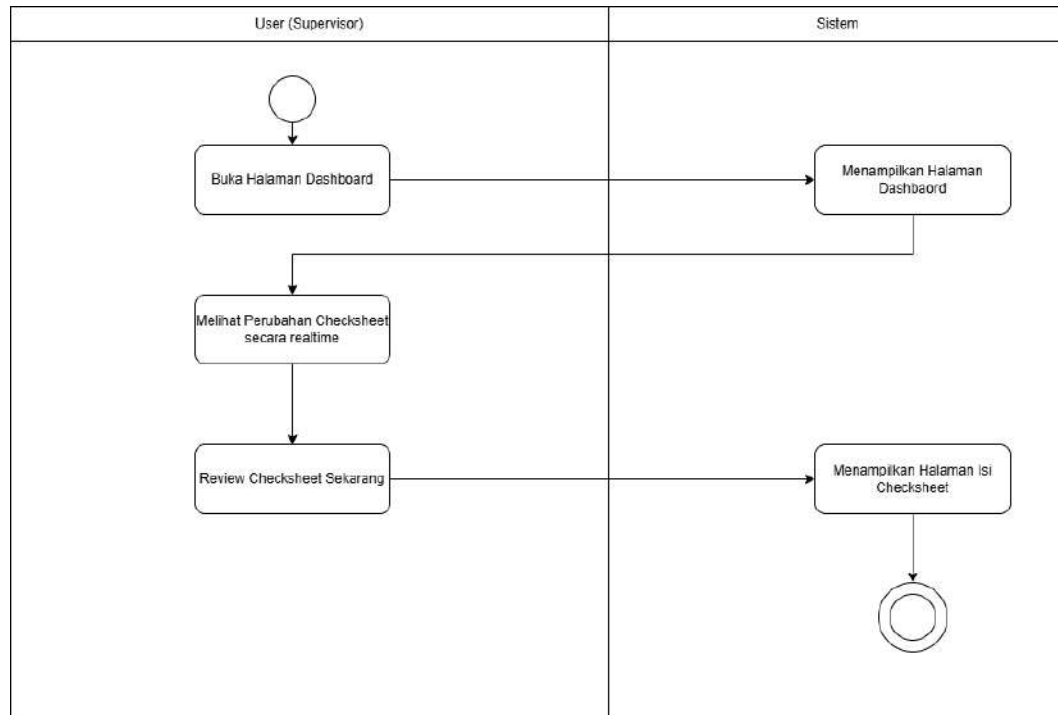
Activity Diagram Admin Kelola *Checksheet* menggambarkan alur proses yang dilakukan Admin dalam membuat, mengubah, dan menghapus form serta jadwal inspeksi pada sistem.

Gambar 3.5 Activity Diagram Admin Kelola *Checksheet*



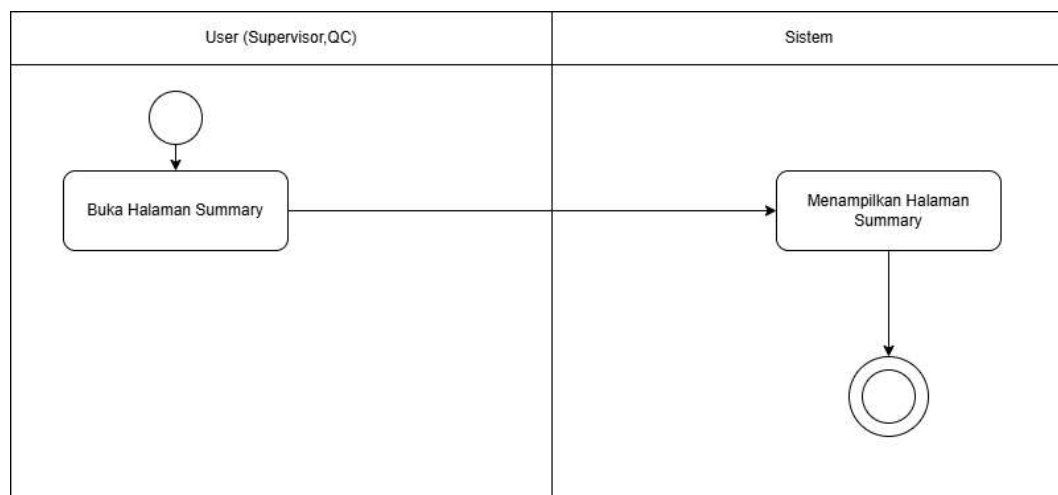
Activity Diagram Supervisor Monitor *Checksheet* menggambarkan alur proses pemantauan dan pemberian keputusan approve atau reject terhadap *checksheet* yang disubmit oleh petugas QC.

Gambar 3.6 Activity Diagram Supervisor Monitor *Checksheet*



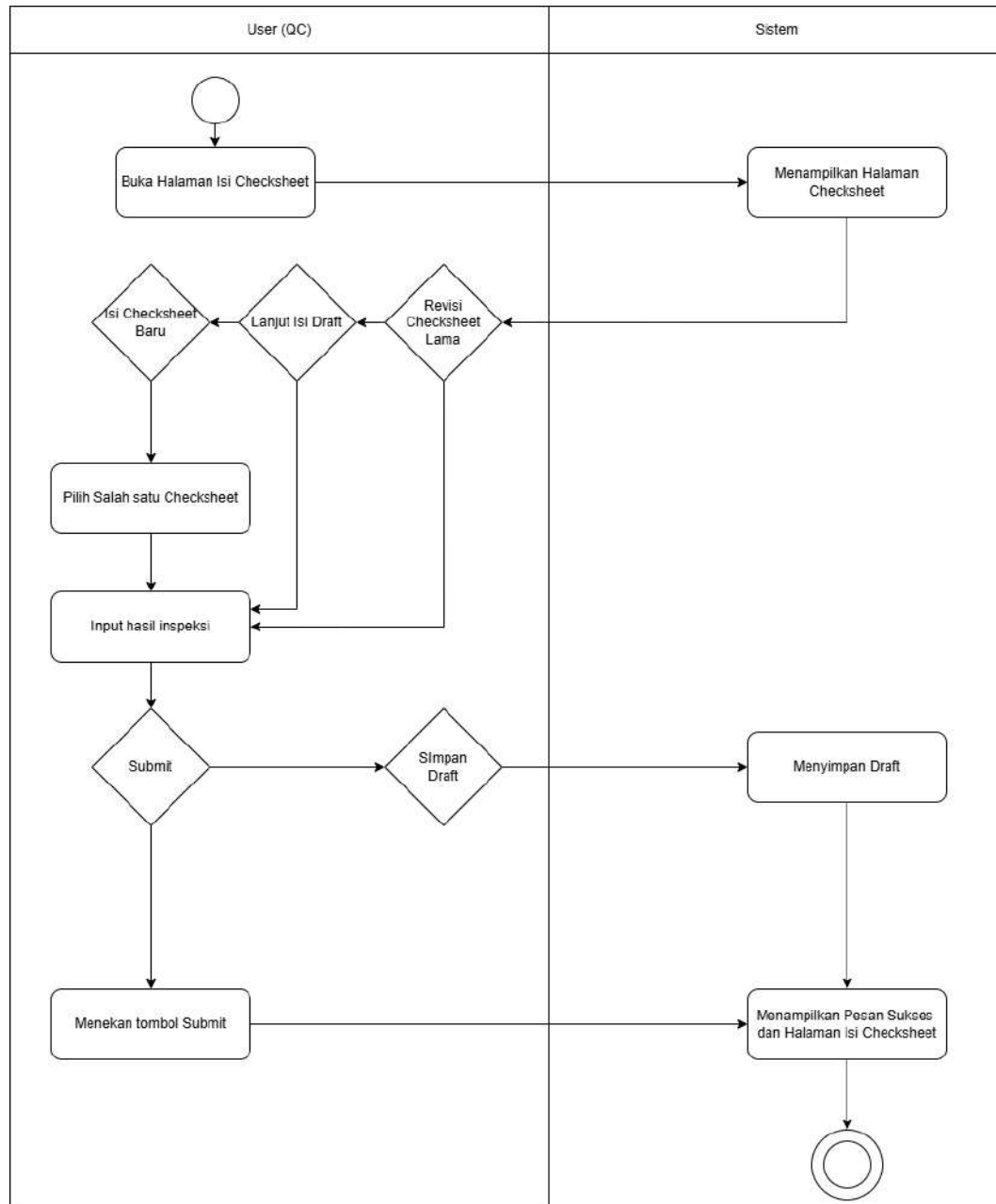
Activity Diagram Supervisor Dashboard menggambarkan alur proses akses dan navigasi *Supervisor* pada halaman dashboard untuk memantau statistik pengisian *checksheet* secara real-time.

Gambar 3.7 Activity Diagram Supervisor Dashboard



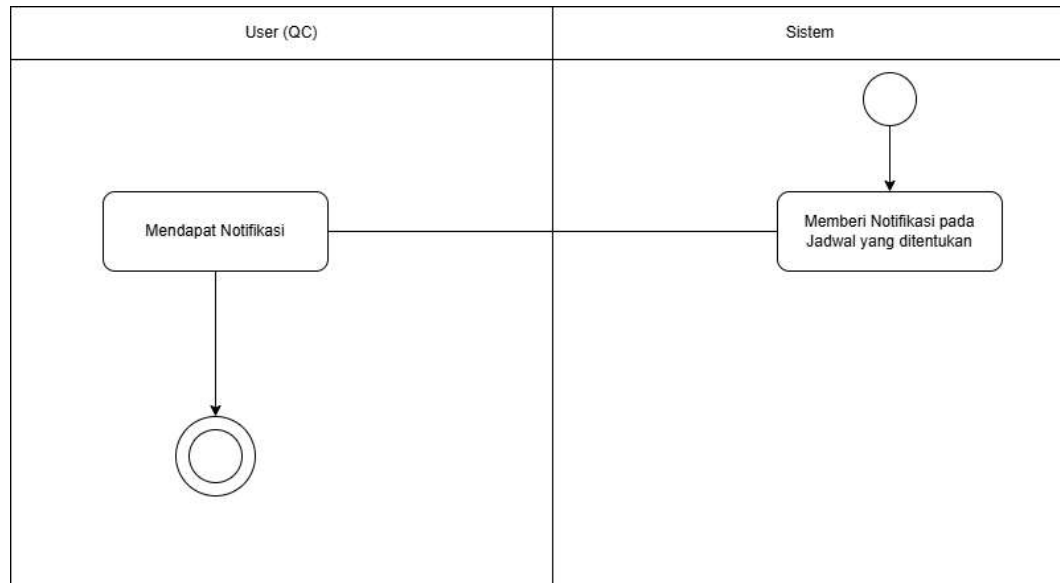
Activity Diagram Summary menggambarkan alur proses pengguna dalam mengakses rekapitulasi data inspeksi harian dan mingguan pada halaman Summary.

Gambar 3.8 Activity Diagram Summary



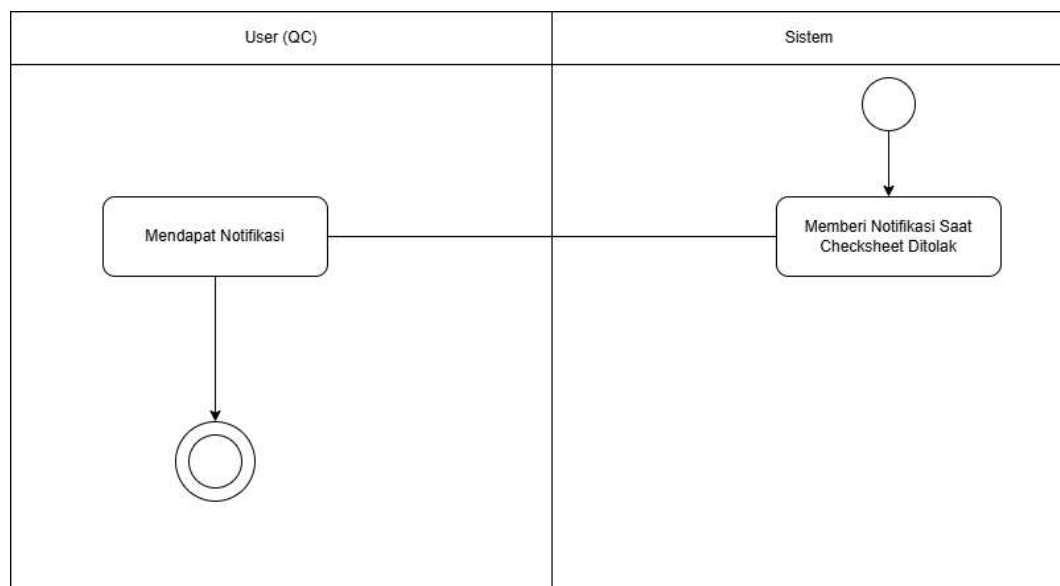
Activity Diagram QC *Checksheet* menggambarkan alur proses pengisian, penyimpanan draft, dan pengiriman *checksheet* digital oleh Petugas QC kepada Supervisor.

Gambar 3.9 Activity Diagram QC *Checksheet*



Activity Diagram Notifikasi QC *Checksheet* menggambarkan alur pengiriman notifikasi pengingat otomatis kepada petugas QC apabila *checksheet* belum diisi sesuai jadwal.

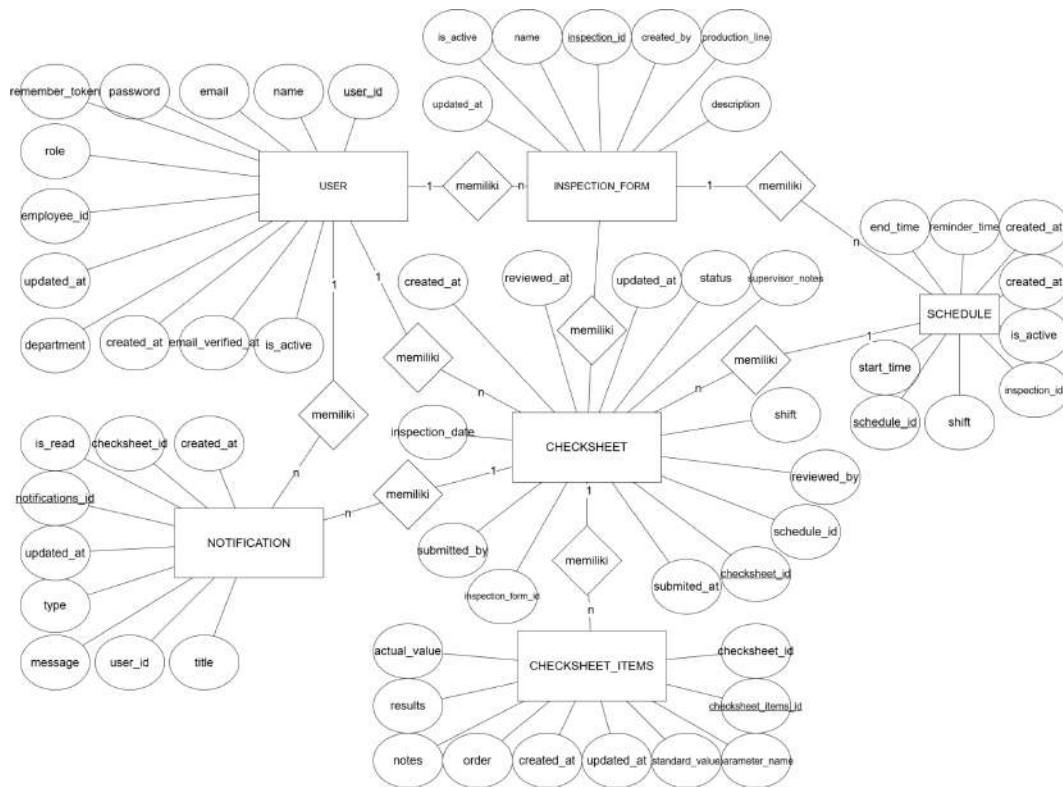
Gambar 3.10 Activity Diagram Notifikasi QC *Checksheet*



Gambar 3.11 Activity Diagram Notifikasi QC *Checksheet* ditolak

Activity Diagram Notifikasi QC *Checksheet* Ditolak menggambarkan alur pengiriman notifikasi kepada petugas QC ketika *checksheet* yang disubmit ditolak oleh Supervisor beserta catatan alasan penolakan.

3.2.6 Entity Relationship Diagram (ERD)



Berikut ini adalah Entity Relationship Diagram (ERD) yang menggambarkan struktur basis data dan hubungan antar entitas dalam sistem *E-Checksheet*.

Gambar 3.12 Entity Relationship Diagram (ERD)

Entity Relationship Diagram (ERD) digunakan untuk menggambarkan struktur basis data yang digunakan dalam sistem *E-Checksheet* beserta hubungan antar entitas. Entitas utama pada sistem terdiri dari pengguna (users), form inspeksi (inspection_forms), jadwal inspeksi (inspection_schedules), *checksheet*, detail *checksheet*, dan notifikasi. Hubungan antar entitas dirancang untuk memastikan integritas data serta mendukung proses pengisian *checksheet*, monitoring, dan pengiriman notifikasi secara terintegrasi. Dengan adanya ERD,

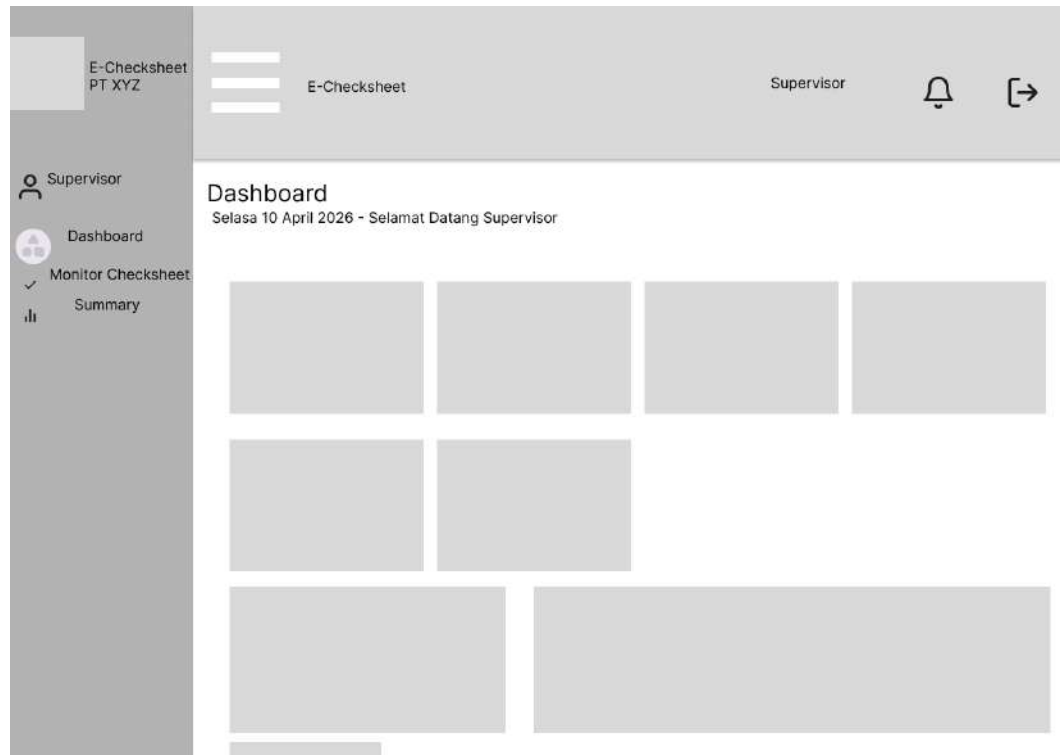
struktur penyimpanan data sistem dapat dirancang secara lebih terorganisir dan mudah dikembangkan pada tahap implementasi.

3.2.7 Perancangan Antarmuka (Wireframe)



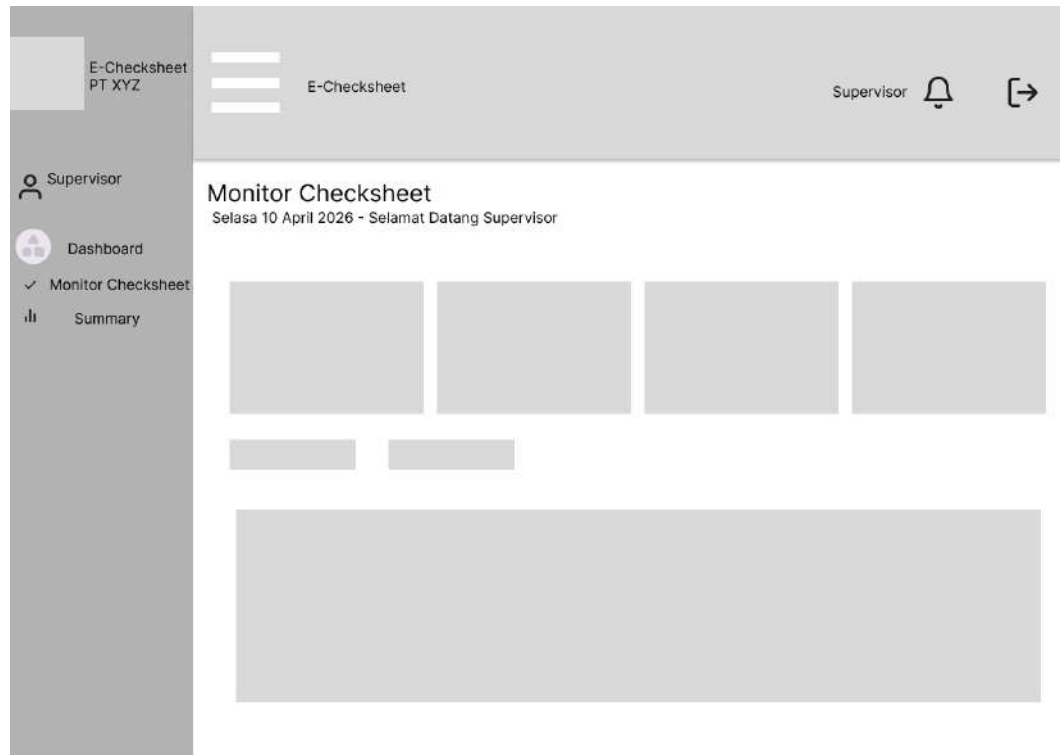
Gambar 3.13 Wireframe Halaman Login

Berikut ini adalah rancangan antarmuka (wireframe) halaman login sebagai pintu masuk utama sistem *E-Checksheets*.



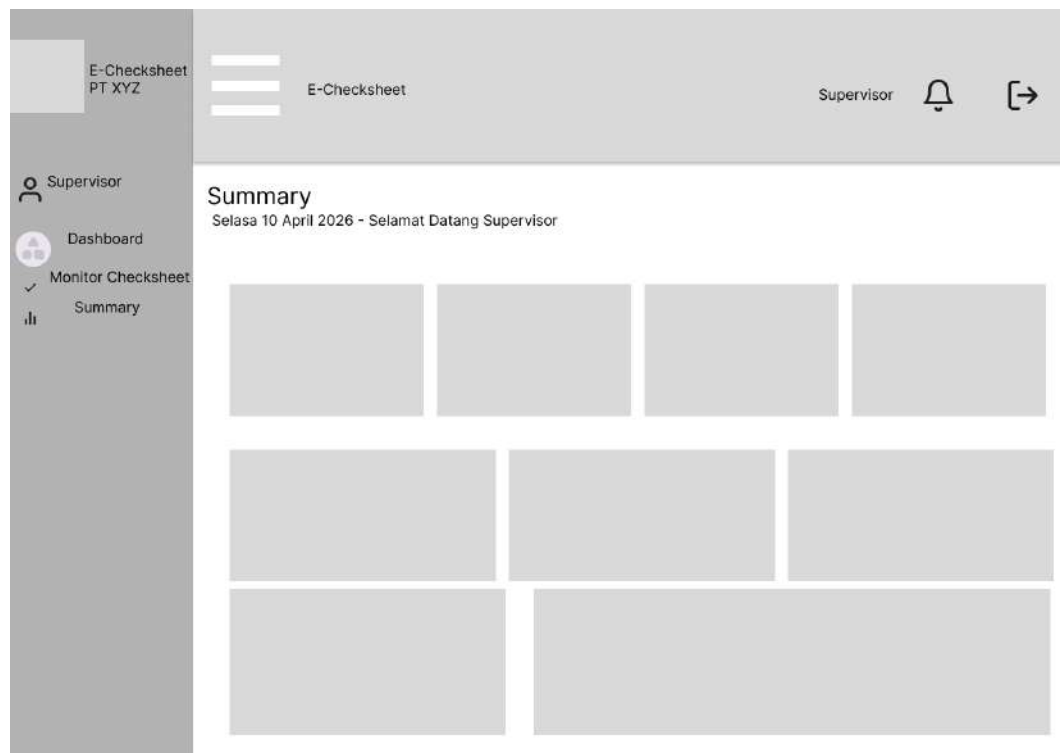
Gambar 3.14 Wireframe Dashboard Supervisor

Berikut ini adalah rancangan antarmuka halaman Dashboard Supervisor yang menampilkan statistik pengisian *checksheet* secara real-time.



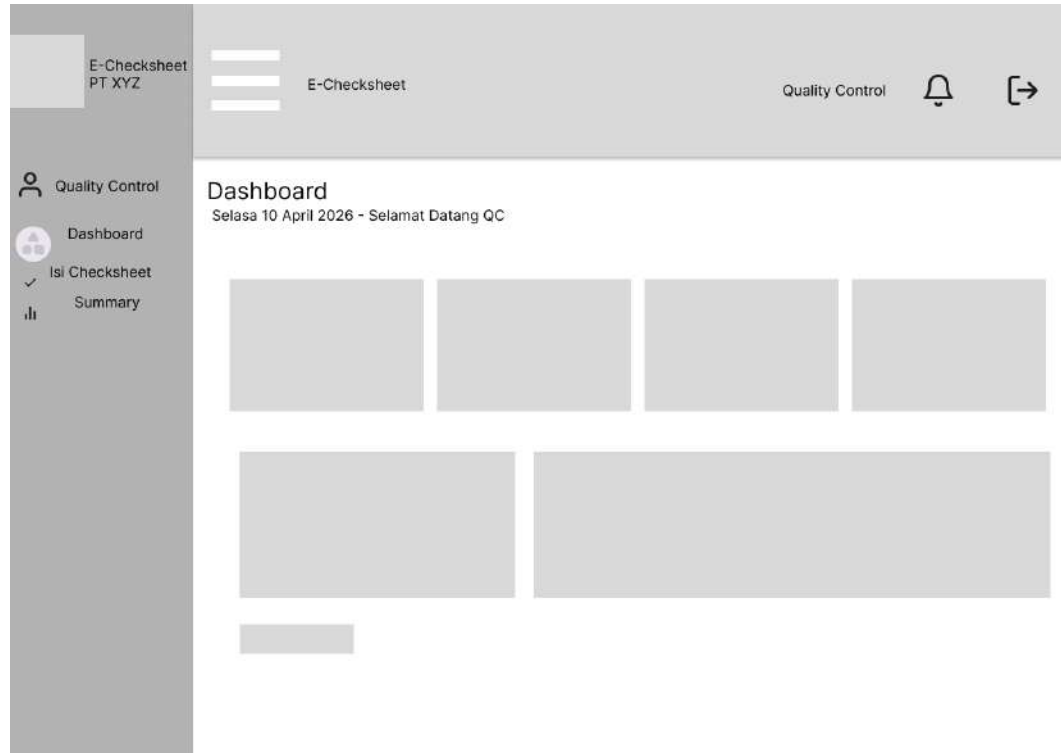
Gambar 3.15 Wireframe Monitor *Checksheet* Supervisor

Berikut ini adalah rancangan antarmuka halaman Monitor *Checksheet* yang digunakan Supervisor untuk memantau dan meninjau seluruh *checksheet* yang disubmit.



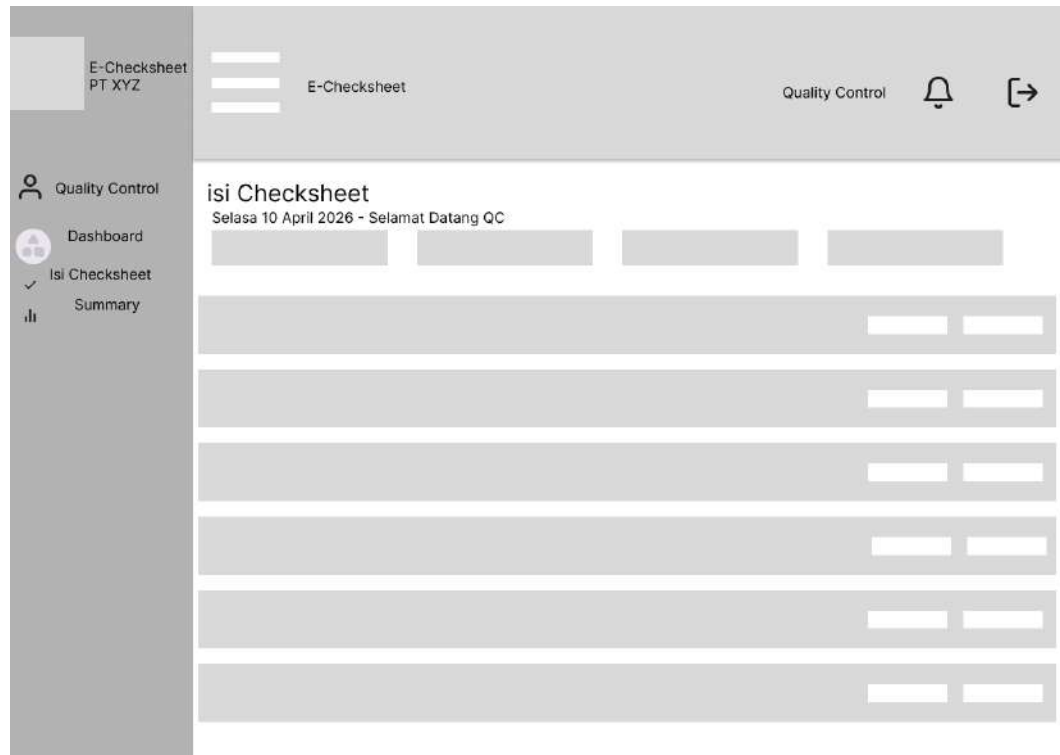
Gambar 3.16 Wireframe Summary Supervisor

Berikut ini adalah rancangan antarmuka halaman Summary Supervisor yang menampilkan rekapitulasi data inspeksi harian dan mingguan.



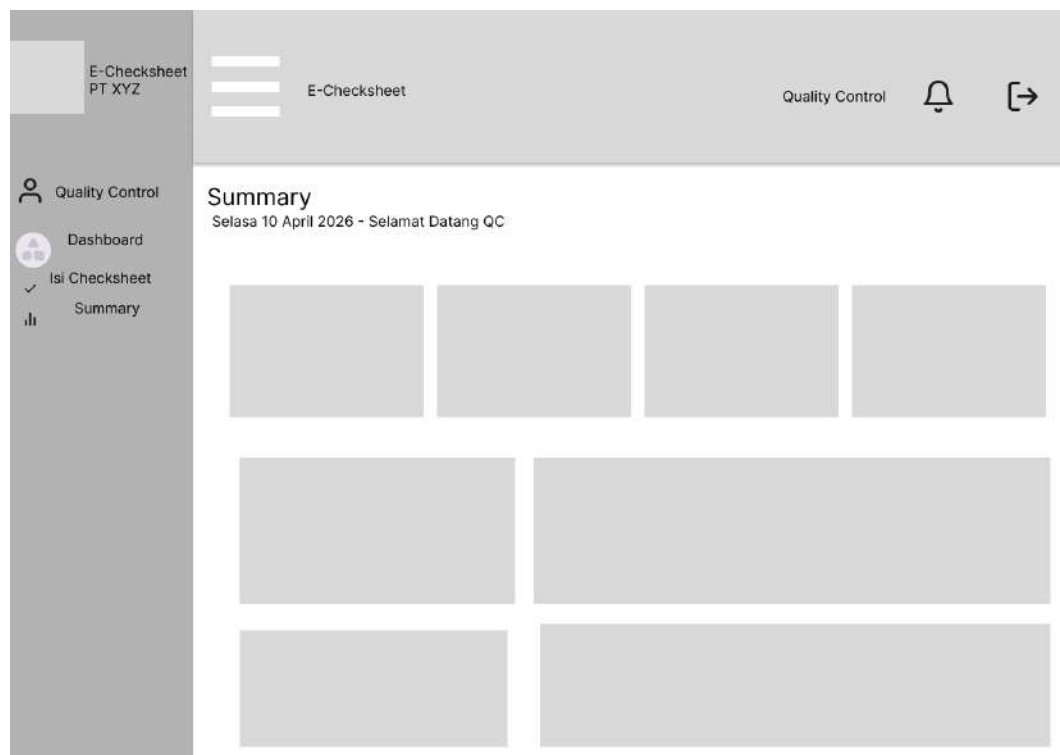
Gambar 3.17 Wireframe Dashboard QC

Berikut ini adalah rancangan antarmuka halaman Dashboard Petugas QC yang menampilkan statistik personal pengisian *checksheet*.



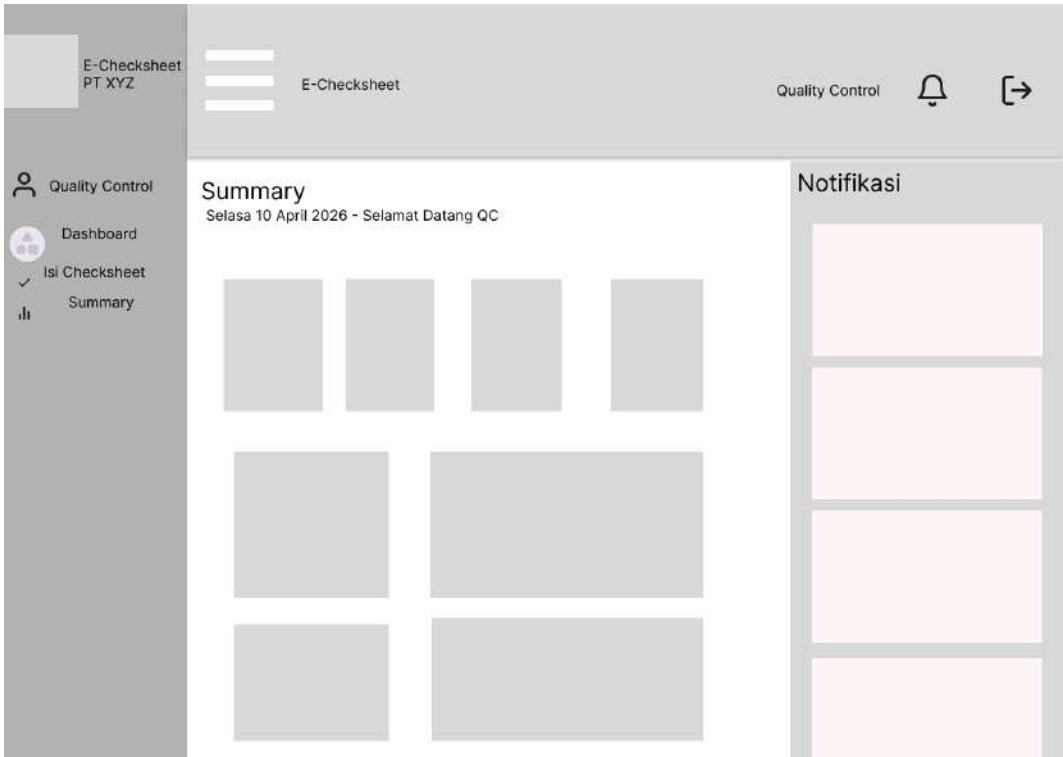
Gambar 3.18 Wireframe Isi *Checksheet* QC

Berikut ini adalah rancangan antarmuka halaman Isi *Checksheet* yang digunakan Petugas QC untuk mengisi data inspeksi secara digital.



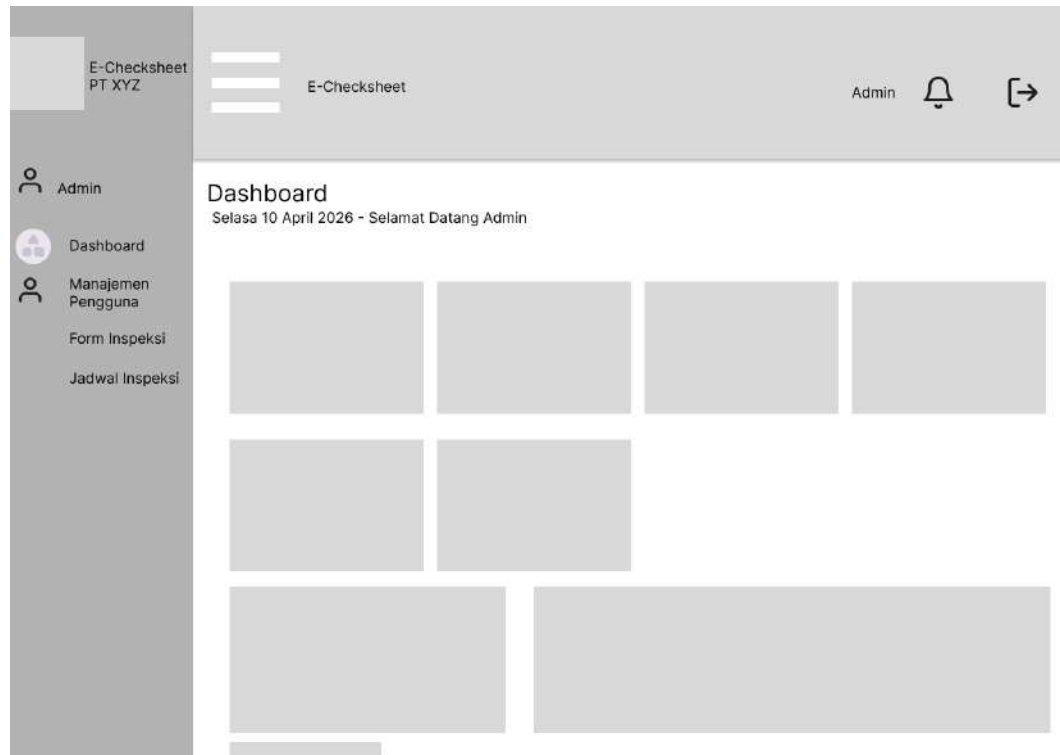
Gambar 3.19 Wireframe Summary QC

Berikut ini adalah rancangan antarmuka halaman Summary QC yang menampilkan rekapitulasi hasil inspeksi milik Petugas QC.



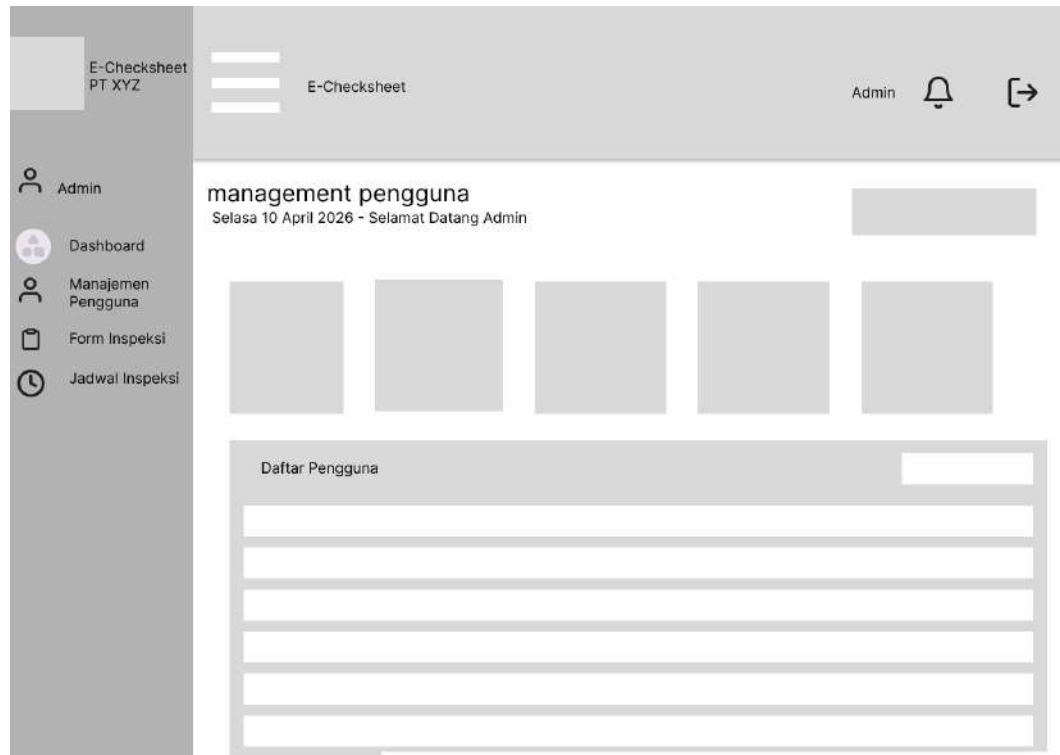
Berikut ini adalah rancangan antarmuka komponen notifikasi sidebar yang menampilkan daftar notifikasi sistem kepada pengguna.

Gambar 3.20 Wireframe Notifikasi Sidebar



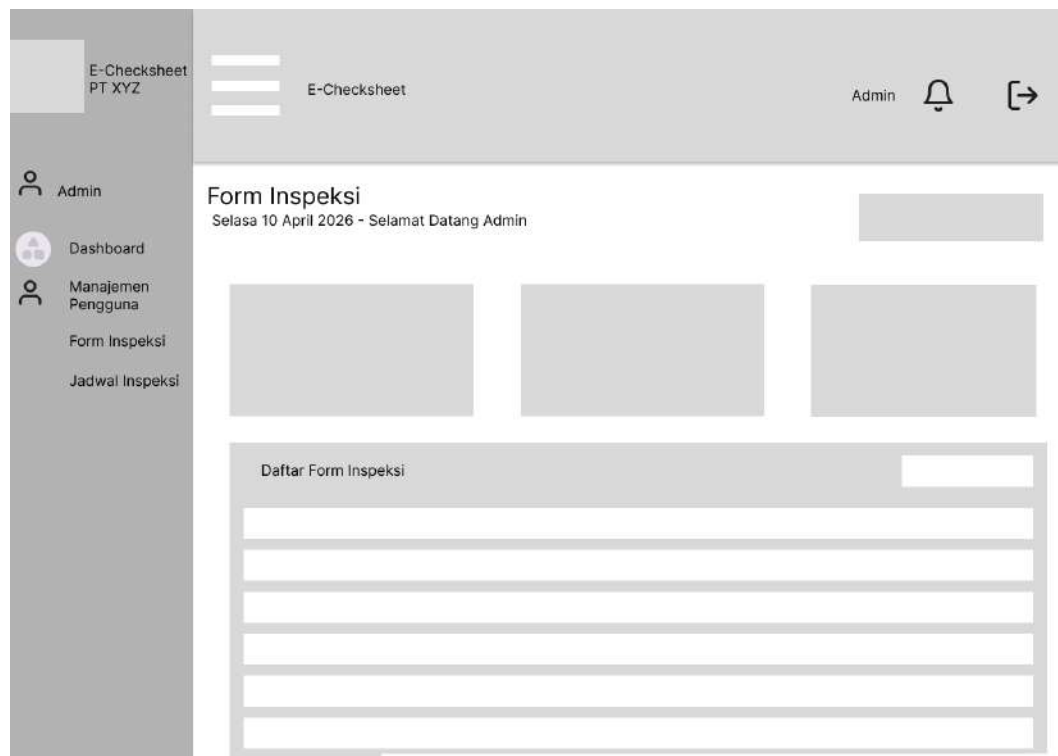
Gambar 3.21 Wireframe Dashboard Admin

Berikut ini adalah rancangan antarmuka halaman Dashboard Admin yang menampilkan ringkasan statistik dan akses cepat ke fitur pengelolaan sistem.



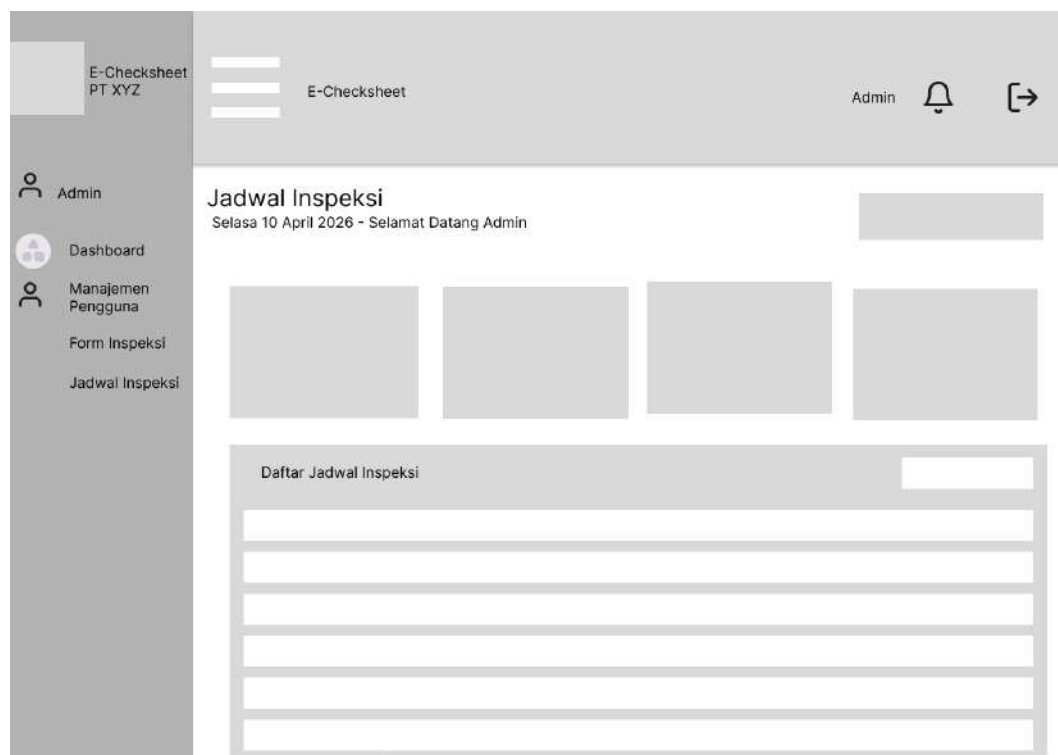
Gambar 3.22 Wireframe Manajemen Pengguna

Berikut ini adalah rancangan antarmuka halaman Manajemen Pengguna yang memungkinkan Admin mengelola seluruh akun pengguna sistem.



Gambar 3.23 Wireframe Form Inspeksi

Berikut ini adalah rancangan antarmuka halaman Form Inspeksi yang digunakan Admin untuk membuat dan mengelola template form pemeriksaan QC.



Gambar 3.24 Wireframe Jadwal Inspeksi

Berikut ini adalah rancangan antarmuka halaman Jadwal Inspeksi yang digunakan Admin untuk menetapkan jadwal pemeriksaan beserta jam pengingat otomatis.

BAB IV IMPLEMENTASI DAN PENGUJIAN

4.1 Hasil Implementasi

Tahap implementasi merupakan proses penerapan hasil perancangan sistem ke dalam bentuk aplikasi yang dapat digunakan. Sistem yang dibangun pada Proyek Akhir ini berupa Aplikasi Web *E-Checksheet* untuk Optimalisasi Proses Quality Control pada PT XYZ, menggunakan framework Laravel sebagai backend dan Quasar Vue.js sebagai frontend.

4.1.1 Deskripsi dan Proses Implementasi

Sistem *E-Checksheet* dikembangkan menggunakan dua komponen utama, yaitu *backend* yang dibangun dengan *framework* Laravel berbasis bahasa pemrograman PHP, serta *frontend* yang dikembangkan menggunakan Vue.js dengan *framework* Quasar. *Backend* terhubung dengan *database* MySQL untuk menyimpan seluruh data sistem. Proses implementasi dilakukan melalui beberapa tahapan sebagai berikut.

4. Melakukan *cloning* repositori proyek dari GitHub menggunakan perintah *git clone*.
5. Melakukan konfigurasi file *environment* (.env) untuk koneksi *database* MySQL yang dijalankan melalui XAMPP.
6. Membuat *database* MySQL yang akan digunakan oleh sistem melalui phpMyAdmin pada XAMPP.
7. Menginstal seluruh dependensi *backend* Laravel menggunakan perintah *composer install*, serta dependensi *frontend* Quasar Vue.js menggunakan perintah *npm install*.
8. Menjalankan migrasi *database* menggunakan perintah *php artisan migrate* untuk membuat struktur tabel pada *database*.
9. Menjalankan server *backend* Laravel menggunakan perintah *php artisan serve*, serta menjalankan server *frontend* Quasar menggunakan perintah *quasar dev*.

Spesifikasi lingkungan pengembangan yang digunakan dalam implementasi sistem ini dapat dilihat pada Tabel 4.1 berikut.

Tabel 4.1 Implementasi Lingkungan Pengembangan

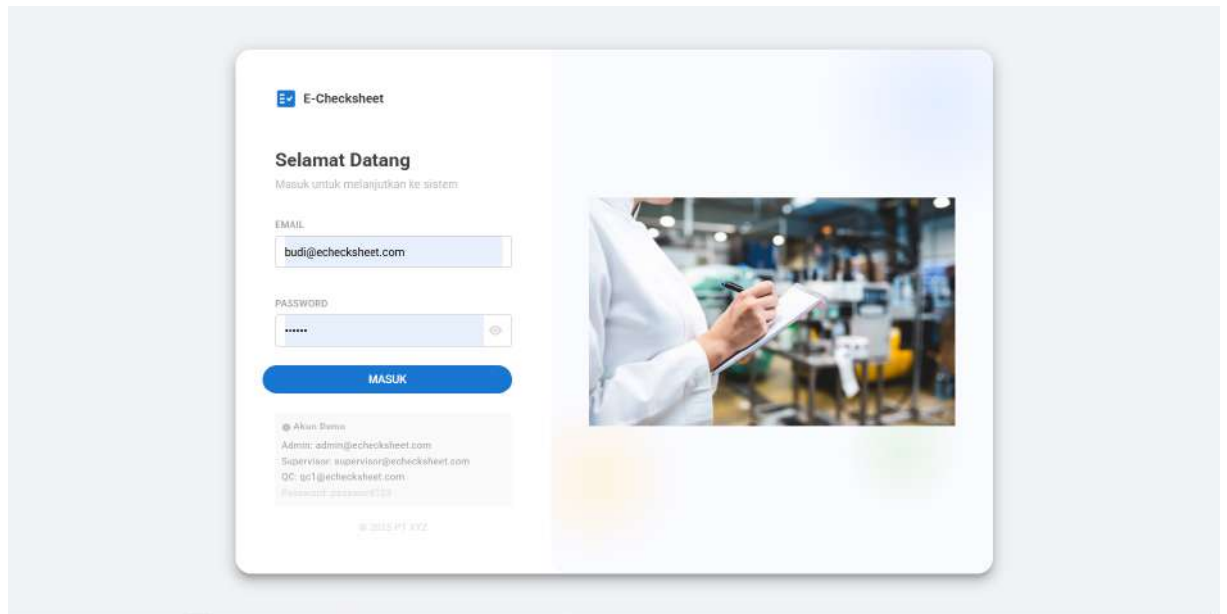
Komponen	Spesifikasi
Framework Backend	Laravel versi 10
Bahasa Pemrograman Backend	PHP 8.2
Framework Frontend	Vue.js 3 + Quasar Framework 2
Database	MySQL 8.0
Web Server (Development)	XAMPP (Apache + MySQL)
Autentikasi	Laravel Sanctum (Token-based)
Node.js	v22+
Package Manager	Composer (PHP), npm (JavaScript)
Code Editor	Visual Studio Code

4.1.2 Implementasi Fitur Sistem

Implementasi sistem dibagi berdasarkan peran pengguna: Admin, Petugas QC, dan Supervisor.

a. Halaman Login

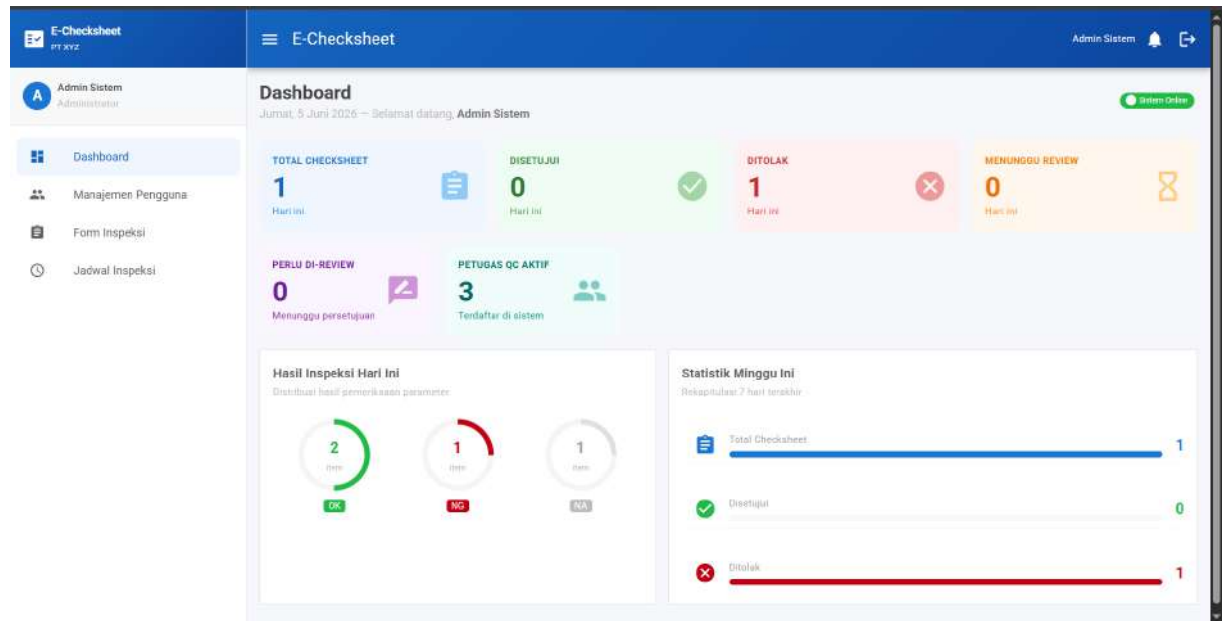
Halaman login merupakan pintu masuk utama sistem *E-Checksheets*. Pengguna memasukkan email dan password untuk autentikasi. Sistem menggunakan Laravel Sanctum untuk menghasilkan token autentikasi yang disimpan di browser dan digunakan pada setiap request API berikutnya. Halaman ini juga menampilkan deskripsi singkat fitur-fitur sistem di panel kanan sebagai informasi awal bagi pengguna.



Gambar 4.1 Halaman Login

b. Dashboard Admin

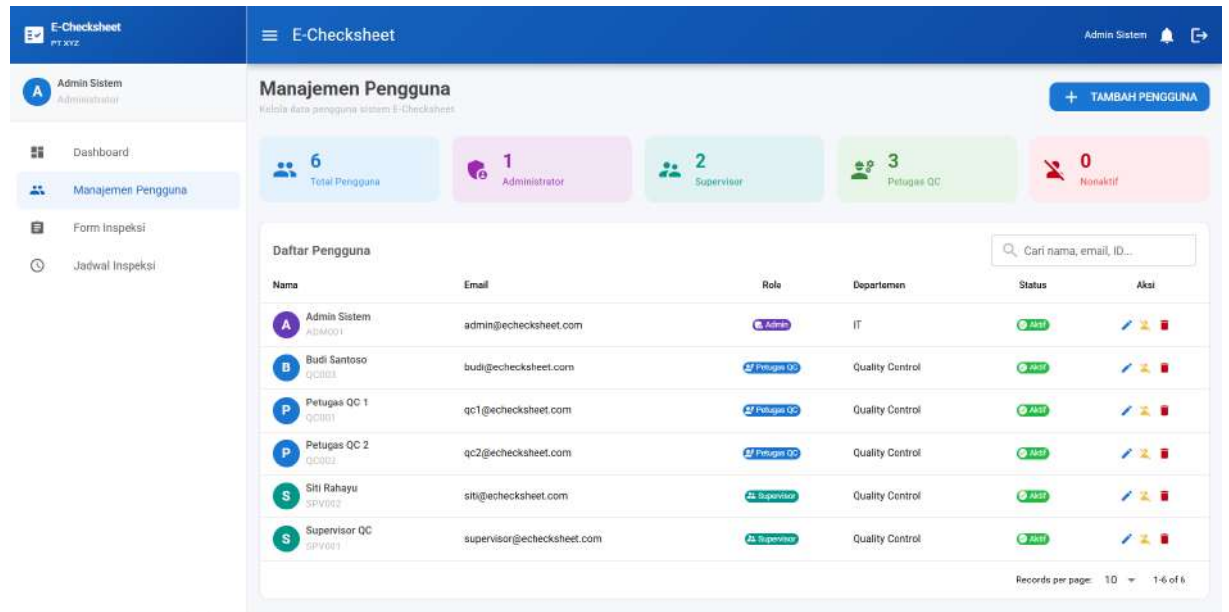
Dashboard Admin menampilkan ringkasan statistik sistem secara real-time, meliputi total *checksheets* hari ini, jumlah yang disetujui, ditolak, dan menunggu review. Terdapat juga informasi jumlah petugas QC aktif dan *checksheets* yang perlu direview. Grafik hasil inspeksi OK/NG/NA ditampilkan dalam bentuk circular progress untuk memudahkan pemantauan kualitas.



Gambar 4.2 Dashboard Admin

c. Halaman Manajemen Pengguna

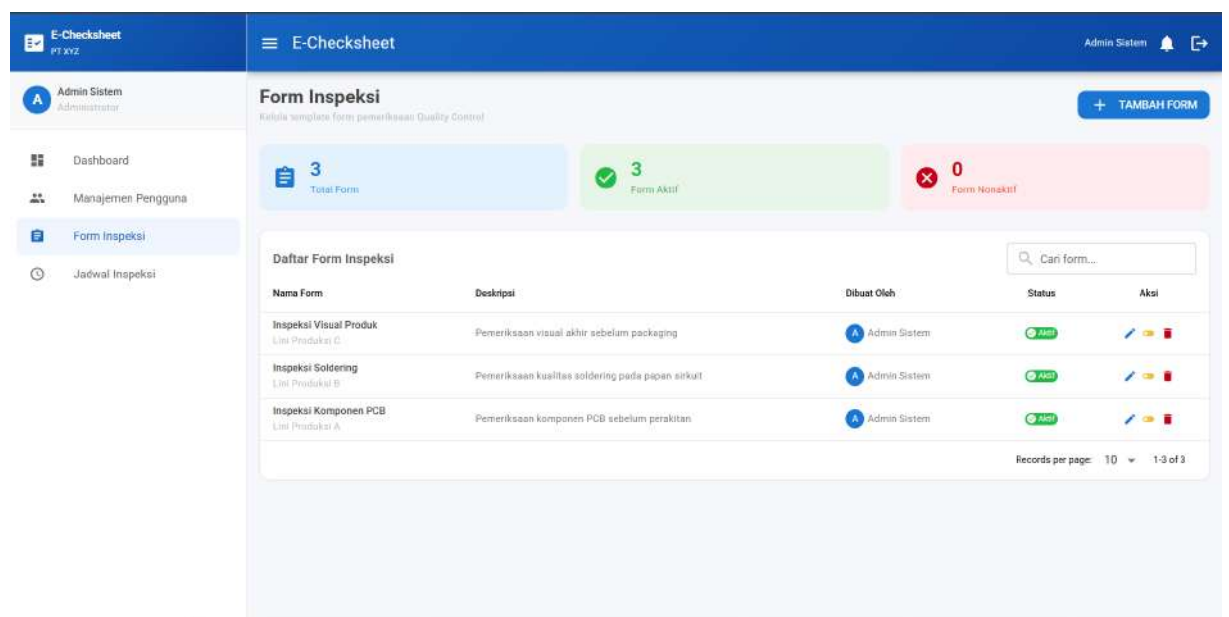
Halaman Manajemen Pengguna memungkinkan Admin untuk menambah, mengedit, mengaktifkan/menonaktifkan, dan menghapus akun pengguna. Setiap pengguna memiliki peran (Admin, Supervisor, atau Petugas QC) yang menentukan hak akses di seluruh sistem. Tabel menampilkan avatar dengan inisial nama, badge warna berbeda untuk setiap peran, dan chip status aktif/nonaktif.



Gambar 4.3 Halaman Manajemen Pengguna

d. Halaman Form Inspeksi

Halaman Form Inspeksi digunakan Admin untuk mengelola template form pemeriksaan QC. Setiap form berisi nama form, lini produksi, dan deskripsi. Form yang aktif akan muncul sebagai pilihan saat membuat jadwal inspeksi dan menjadi acuan parameter yang diisi petugas QC.



Gambar 4.4 Halaman Form Inspeksi

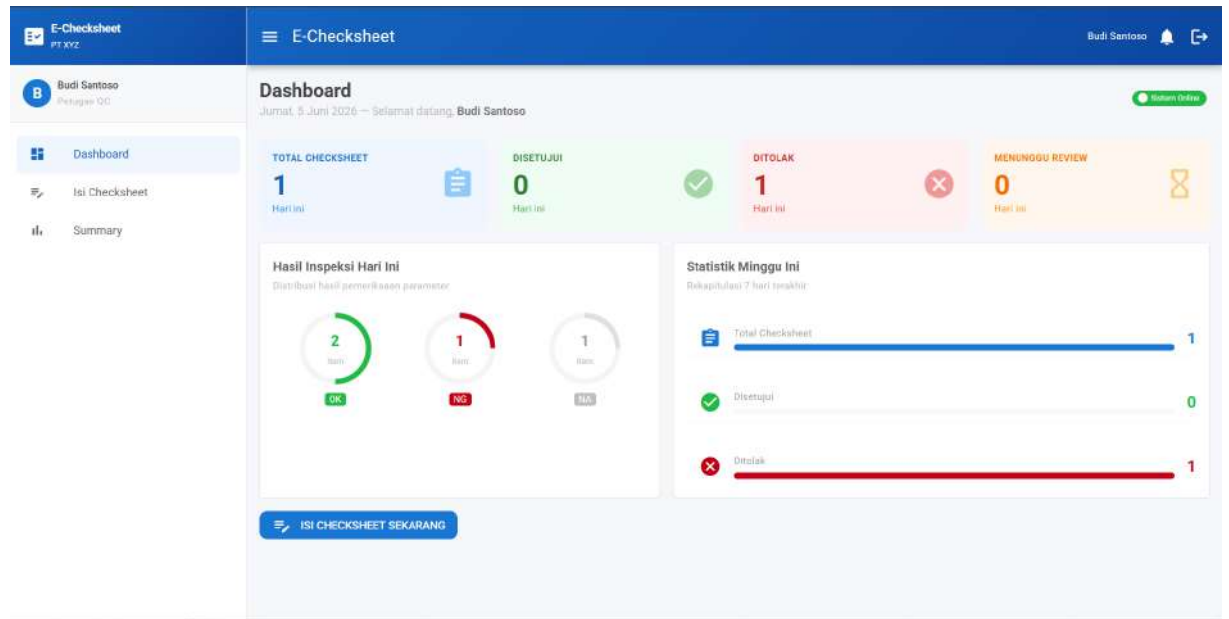
e. Halaman Jadwal Inspeksi

Halaman Jadwal Inspeksi memungkinkan Admin untuk menetapkan jadwal pemeriksaan QC berdasarkan form inspeksi yang telah dibuat. Setiap jadwal memiliki shift (pagi/siang/malam), jam mulai, jam selesai, dan jam pengingat. Jam pengingat digunakan oleh sistem scheduler untuk mengirimkan notifikasi otomatis ke seluruh petugas QC aktif apabila *checksheet* belum diisi pada waktu tersebut.

Gambar 4.5 Halaman Jadwal Inspeksi

f. Dashboard Petugas QC

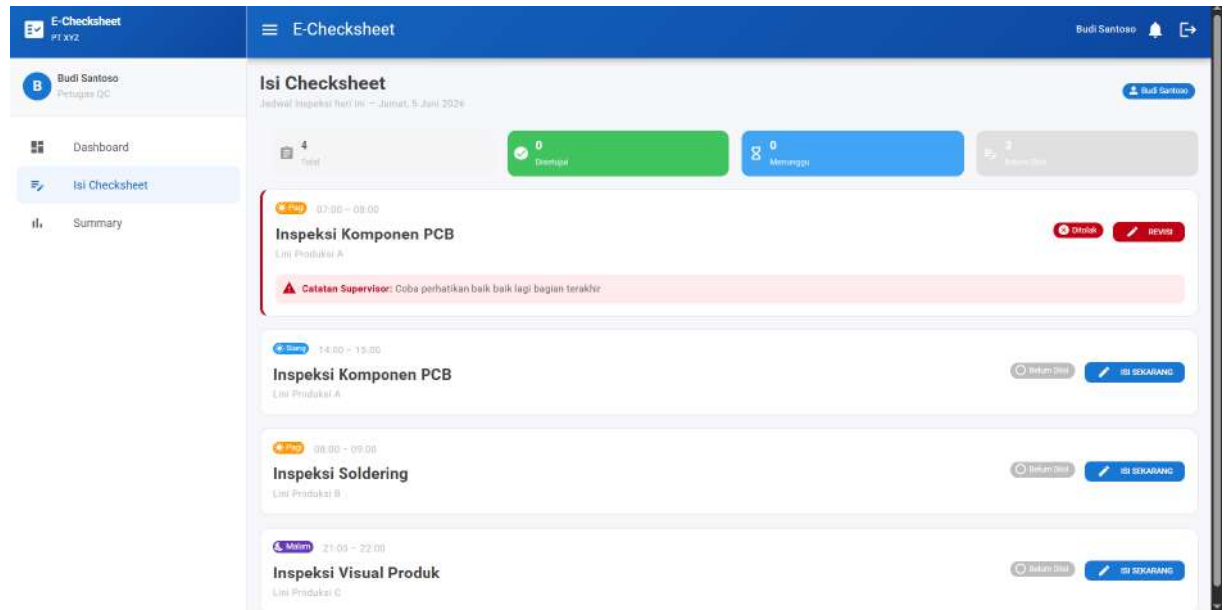
Dashboard Petugas QC menampilkan statistik personal pengisian *checksheet* hari ini dan minggu ini, meliputi jumlah *checksheet* yang disetujui, ditolak, dan menunggu review. Terdapat tombol pintasan “Isi *Checksheet* Sekarang” untuk memudahkan navigasi ke halaman pengisian. Grafik hasil inspeksi OK/NG/NA juga ditampilkan untuk pemantauan kualitas mandiri.



Gambar 4.6 Dashboard Petugas QC

g. Halaman Isi *Checksheet*

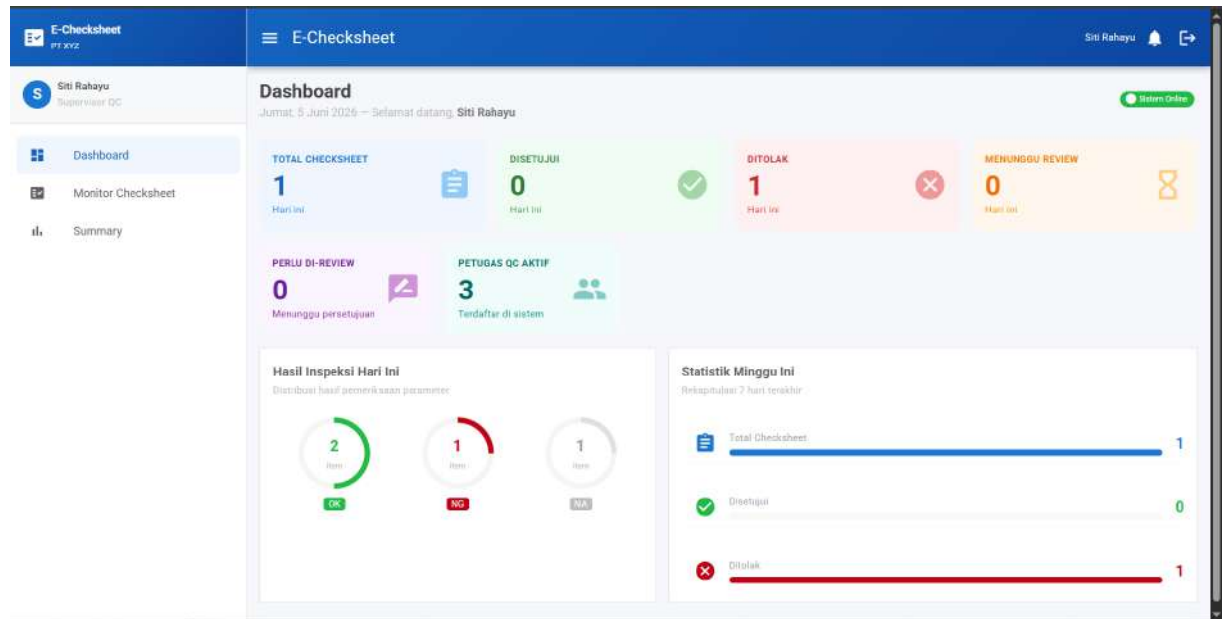
Halaman Isi *Checksheet* merupakan fitur utama bagi petugas QC. Halaman ini menampilkan daftar jadwal inspeksi hari ini beserta status pengisian masing-masing (Belum Diisi, Draft, Menunggu Review, Disetujui, atau Ditolak). Petugas QC dapat mengisi nilai aktual untuk setiap parameter, memilih hasil pemeriksaan (OK/NG/NA), dan menambahkan catatan. Tersedia tombol Simpan Draft untuk menyimpan sementara dan Submit ke Supervisor untuk mengirim *checksheet* yang telah selesai.



Gambar 4.7 Halaman Isi *Checksheet*

h. Dashboard Supervisor

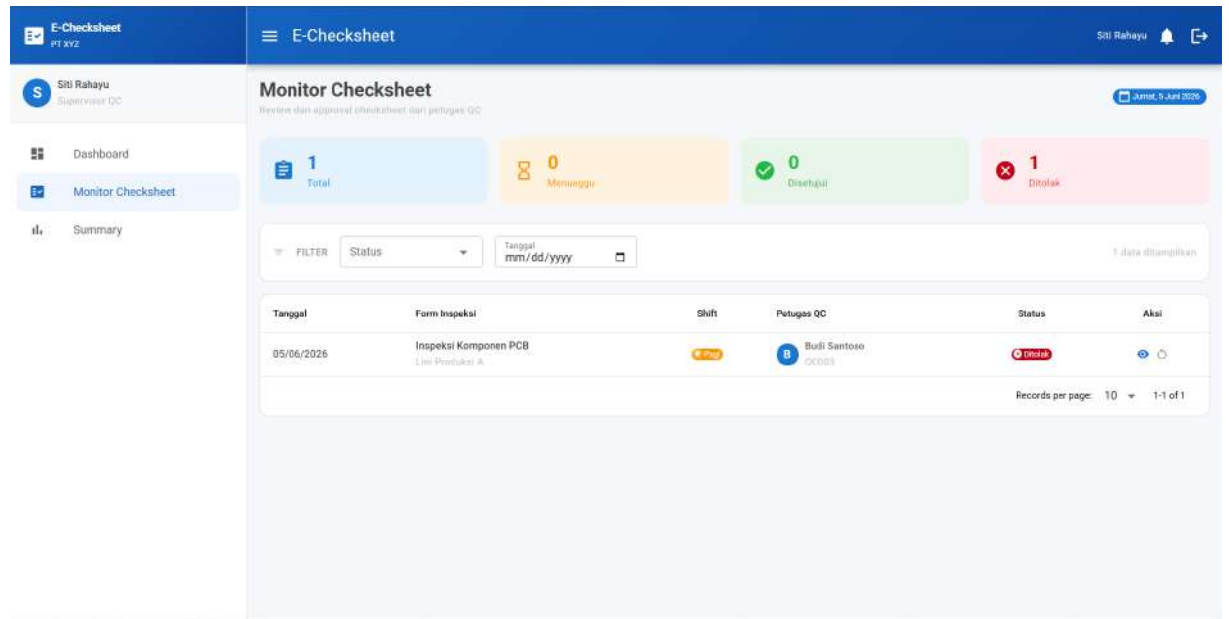
Dashboard Supervisor menampilkan informasi statistik menyeluruh termasuk total *checksheet*, jumlah yang perlu direview segera, dan jumlah petugas QC aktif. Terdapat tombol pintasan menuju halaman Monitor *Checksheet* apabila terdapat *checksheet* yang menunggu persetujuan, sehingga supervisor dapat segera mengambil tindakan.



Gambar 4.8 Dashboard Supervisor

i. Halaman Monitor *Checksheets*

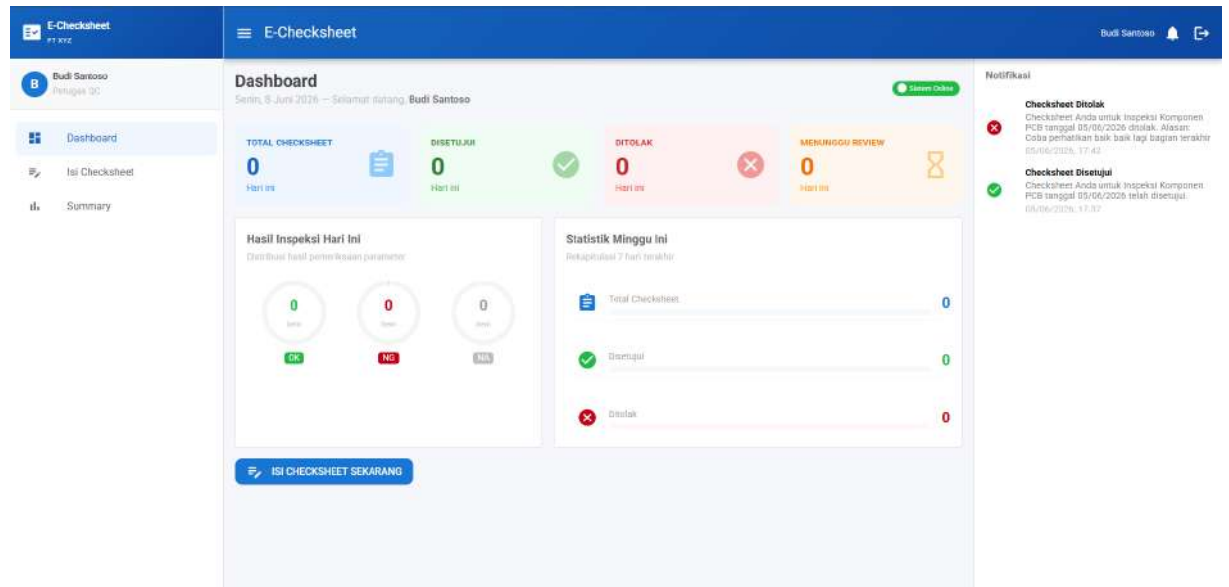
Halaman Monitor *Checksheets* memungkinkan Supervisor untuk melihat seluruh *checksheets* yang disubmit oleh petugas QC. Tersedia filter berdasarkan status dan tanggal untuk memudahkan pencarian. Supervisor dapat melihat detail parameter pemeriksaan, kemudian memberikan keputusan Approve (menyetujui) atau Reject (menolak dengan catatan alasan). Setiap keputusan secara otomatis mengirimkan notifikasi kepada petugas QC yang bersangkutan.



Gambar 4.9 Halaman Monitor *Checksheet*

j. Fitur Notifikasi Sistem

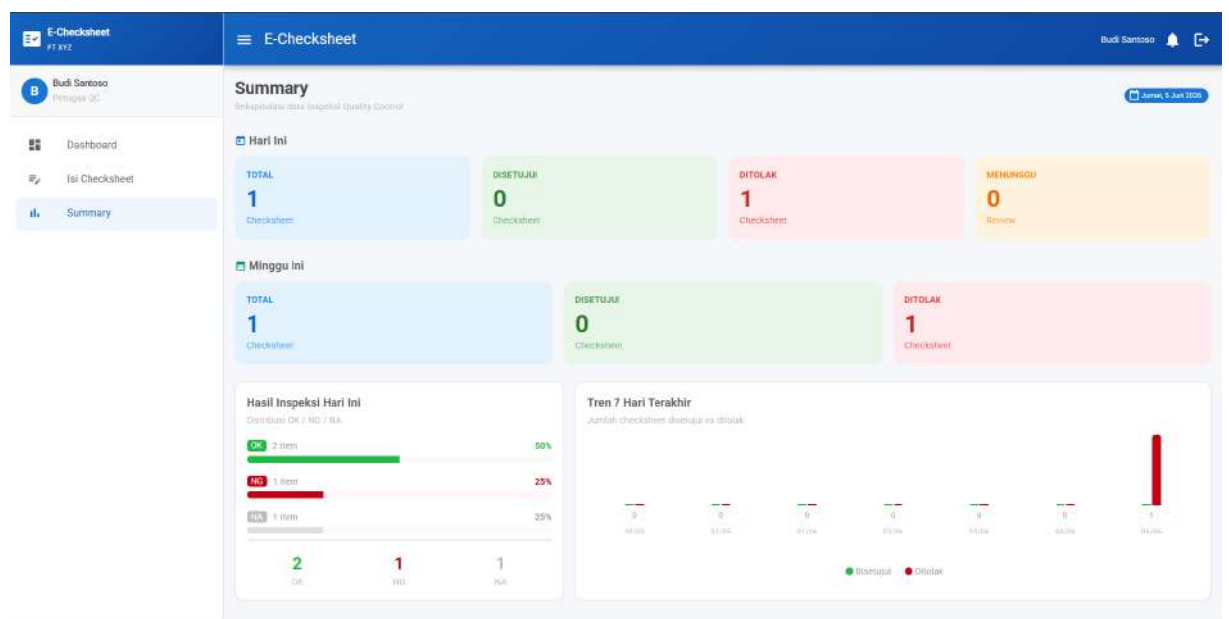
Sistem notifikasi ditampilkan melalui ikon lonceng di navbar dengan badge merah yang menunjukkan jumlah notifikasi yang belum dibaca. Terdapat dua jenis notifikasi: (1) Notifikasi reminder otomatis yang dikirim oleh scheduler Laravel ketika jadwal inspeksi telah tiba namun *checksheets* belum diisi, dan (2) Notifikasi hasil review yang dikirim ketika supervisor menyetujui atau menolak *checksheets*. Pengguna dapat menandai notifikasi sebagai sudah dibaca secara individual maupun sekaligus.



Gambar 4.10 Fitur Notifikasi Sistem

k. Halaman Summary

Halaman Summary menyediakan rekapitulasi data inspeksi harian dan mingguan. Ditampilkan statistik total *checksheet*, jumlah disetujui, ditolak, dan menunggu review. Hasil inspeksi OK/NG/NA divisualisasikan dengan progress bar. Tren 7 hari terakhir ditampilkan dalam bentuk grafik batang yang memudahkan analisis kecenderungan kualitas produksi.



Gambar 4.11 Halaman Summary

4.1.3 Alur Penggunaan Aplikasi

4.1.3.1 Alur Penggunaan Aplikasi Admin

Alur penggunaan aplikasi oleh Admin dimulai dengan masuk ke sistem menggunakan kredensial yang valid, kemudian Admin akan diarahkan ke *dashboard* yang menampilkan ringkasan statistik sistem secara *real-time*. Admin dapat mengelola akun pengguna, termasuk menambahkan, mengubah peran, mengaktifkan, dan menonaktifkan pengguna. Selanjutnya, Admin dapat membuat *template* form inspeksi yang berisi parameter pemeriksaan, kemudian menetapkan jadwal inspeksi beserta jam pengingat otomatis untuk setiap *shift*. Setelah semua konfigurasi selesai, sistem akan berjalan secara otomatis mengirimkan notifikasi pengingat kepada Petugas QC sesuai jadwal yang telah ditetapkan.

4.1.3.2 Alur Penggunaan Aplikasi Petugas QC

Petugas QC memulai penggunaan aplikasi dengan masuk ke sistem menggunakan akun yang telah dibuat oleh Admin. Setelah masuk, Petugas QC dapat melihat daftar jadwal inspeksi hari ini beserta status pengisiannya melalui halaman Isi *Checksheet*. Petugas QC mengisi nilai aktual setiap parameter pemeriksaan, memilih hasil (OK/NG/NA), dan dapat menyimpan sebagai *draft* sebelum mengirimkan ke Supervisor. Apabila *checksheet* ditolak oleh Supervisor, Petugas QC akan menerima notifikasi beserta catatan alasan penolakan dan dapat melakukan revisi sebelum mengirim ulang.

4.1.3.3 Alur Penggunaan Aplikasi Supervisor

Supervisor masuk ke sistem dan dapat langsung memantau status pengisian *checksheet* melalui *dashboard*. Pada halaman Monitor *Checksheet*, Supervisor dapat melihat seluruh *checksheet* yang telah disubmit, memeriksa detail parameter pemeriksaan, kemudian memberikan keputusan *approve* atau *reject* disertai catatan. Setiap keputusan secara otomatis mengirimkan notifikasi kepada Petugas QC yang bersangkutan. Selain itu, Supervisor dapat melihat

rekapitulasi data inspeksi harian dan mingguan melalui halaman *Summary* untuk mendukung pengambilan keputusan terkait kualitas produksi.

4.2 Pengujian Black Box Testing

Pengujian sistem dilakukan menggunakan metode Black Box Testing. Metode ini berfokus pada pengujian fungsi perangkat lunak berdasarkan masukan (input) dan keluaran (output) yang dihasilkan tanpa memperhatikan struktur kode program yang digunakan. Tujuan dari pengujian Black Box Testing adalah untuk memastikan bahwa setiap fitur sistem dapat berjalan sesuai dengan kebutuhan fungsional yang telah ditentukan serta mampu menghasilkan keluaran yang sesuai dengan harapan pengguna. Metode ini banyak digunakan dalam pengujian aplikasi karena efektif dalam mengidentifikasi kesalahan fungsi, validasi data, dan kesesuaian proses bisnis yang diimplementasikan pada sistem (Saravanos & Curinga, 2023).

Tabel 4.3 Hasil Pengujian Black Box Testing - Login dan Autentikasi (FR-01)

No	Skenario Pengujian	Input	Expected Output	Actual Output	Status
BBT-01-01	Login kredensial valid - Admin	Email dan password Admin benar	Redirect ke Dashboard Admin	Redirect ke Dashboard Admin	Berhasil
BBT-01-02	Login kredensial valid - QC	Email dan password QC benar	Redirect ke Dashboard QC	Redirect ke Dashboard QC	Berhasil
BBT-01-03	Login kredensial valid - Supervisor	Email dan password	Redirect ke Dashboard Supervisor	Redirect ke Dashboard Supervisor	Berhasil

No	Skenario Pengujian	Input	Expected Output	Actual Output	Status
		Supervisor benar			
BBT-01-04	Login password salah	Email benar, password salah	Pesan error, akses ditolak	Pesan error tampil	Berhasil
BBT-01-05	Login email tidak terdaftar	Email tidak ada di sistem	Pesan error, akses ditolak	Pesan error tampil	Berhasil
BBT-01-06	Login field kosong	Email dan password dikosongkan	Validasi wajib isi tampil	Validasi tampil	Berhasil
BBT-01-07	Login akun nonaktif	Email akun yang dinonaktifkan	Akses ditolak, pesan nonaktif tampil	Akses ditolak	Berhasil
BBT-01-08	Logout dari sistem	Klik tombol Logout	Token dihapus, redirect ke Login	Redirect ke Login	Berhasil
BBT-01-09	Akses halaman Admin oleh QC	Login QC, akses URL halaman Admin	Akses ditolak / error 403	Akses ditolak	Berhasil
BBT-01-10	Akses setelah logout (back button)	Klik Back browser setelah logout	Redirect ke Login	Redirect ke Login	Berhasil

Tabel 4.4 Hasil Pengujian Black Box Testing - Manajemen Pengguna (FR-02)

No	Skenario Pengujian	Input	Expected Output	Actual Output	Status
BBT-02-01	Tambah pengguna data lengkap	Nama, email, password, role diisi	Pengguna tersimpan, tampil di tabel	Data tersimpan	Berhasil
BBT-02-02	Tambah pengguna email duplikat	Email yang sudah terdaftar	Pesan error email sudah digunakan	Pesan error tampil	Berhasil
BBT-02-03	Tambah pengguna field wajib kosong	Salah satu field kosong	Validasi wajib isi tampil	Validasi tampil	Berhasil
BBT-02-04	Ubah role pengguna	Ganti role QC ke Supervisor	Role berhasil diperbarui	Role berubah di tabel	Berhasil
BBT-02-05	Nonaktifkan pengguna aktif	Klik nonaktifkan, konfirmasi	Status berubah menjadi Nonaktif	Status Nonaktif tampil	Berhasil
BBT-02-06	Aktifkan pengguna nonaktif	Klik aktifkan	Status berubah menjadi Aktif	Status Aktif tampil	Berhasil

Tabel 4.5 Hasil Pengujian Black Box Testing - Manajemen Form dan Jadwal Inspeksi (FR-03)

No	Skenario Pengujian	Input	Expected Output	Actual Output	Status
BBT-03-01	Tambah form inspeksi data lengkap	Nama, lini produksi	Form tersimpan dan tampil di tabel	Data form tersimpan	Berhasil

No	Skenario Pengujian	Input	Expected Output	Actual Output	Status
		parameter diisi			
BBT-03-02	Tambah form field wajib kosong	Field nama form kosong	Validasi wajib isi tampil	Validasi tampil	Berhasil
BBT-03-03	Ubah data form inspeksi	Edit nama dan lini produksi	Data berhasil diperbarui	Data terbaru tampil	Berhasil
BBT-03-04	Hapus form inspeksi	Klik hapus, konfirmasi	Form terhapus dari tabel	Form tidak tampil	Berhasil
BBT-03-05	Tambah jadwal inspeksi	Pilih form, shift, jam mulai, selesai, pengingat	Jadwal tersimpan dengan info shift dan jam	Jadwal tampil di tabel	Berhasil
BBT-03-06	Jadwal - jam selesai lebih awal dari mulai	Jam selesai kurang dari jam mulai	Validasi kesalahan jam tampil	Validasi tampil	Berhasil
BBT-03-07	Ubah jam pengingat jadwal	Edit jam pengingat	Jadwal diperbarui, scheduler pakai jam baru	Data terbaru tersimpan	Berhasil
BBT-03-08	Hapus jadwal inspeksi	Klik hapus jadwal, konfirmasi	Jadwal terhapus dari tabel	Jadwal tidak tampil	Berhasil

Tabel 4.6 Hasil Pengujian Black Box Testing - Pengisian *Checksheet* oleh Petugas QC (FR-07)

No	Skenario Pengujian	Input	Expected Output	Actual Output	Status
BBT-04-01	Isi semua parameter dan submit	Semua nilai aktual dan OK/NG/NA diisi, klik Submit	Status berubah menjadi Menunggu Review	Status berubah	Berhasil
BBT-04-02	Simpan sebagai draft	Sebagian parameter diisi, klik Simpan Draft	<i>Checksheet</i> tersimpan dengan status Draft	Status Draft tampil	Berhasil
BBT-04-03	Submit dengan parameter wajib kosong	Field parameter wajib kosong, klik Submit	Validasi tampil, tidak bisa disubmit	Validasi tampil	Berhasil
BBT-04-04	Edit <i>checksheet</i> Draft	Buka draft, ubah nilai parameter	Perubahan tersimpan	Data terbaru tersimpan	Berhasil
BBT-04-05	Revisi setelah ditolak Supervisor	Buka ditolak, ubah nilai, submit ulang	Status kembali Menunggu Review	Status berubah ke Menunggu Review	Berhasil

No	Skenario Pengujian	Input	Expected Output	Actual Output	Status
BBT -04- 06	Akses <i>checksheet</i> Disetujui (proteksi)	Coba edit <i>checksheet</i> berstatus Disetujui	Sistem tidak izinkan perubahan	Form tidak bisa diedit	Berhasil

Tabel 4.7 Hasil Pengujian Black Box Testing - Approve dan Reject oleh Supervisor (FR-04)

No	Skenario Pengujian	Input	Expected Output	Actual Output	Status
BBT -05- 01	Approve <i>checksheet</i> Menunggu Review	Klik Setujui pada <i>checksheet</i>	Status Disetujui, notifikasi ke QC	Status Disetujui tampil, QC menerima notifikasi	Berhasil
BBT -05- 02	Reject <i>checksheet</i> dengan catatan	Klik Tolak, isi alasan, konfirmasi	Status Ditolak, notifikasi dan catatan ke QC	Status Ditolak, QC menerima notifikasi beserta catatan	Berhasil
BBT -05- 03	Reject tanpa alasan	Klik Tolak, field alasan kosong	Validasi wajib isi alasan	Validasi tampil	Berhasil
BBT -05- 04	Approve yang sudah Disetujui (proteksi)	Coba approve <i>checksheet</i> Disetujui	Tombol tidak aktif / aksi tidak diproses	Aksi tidak dapat dilakukan	Berhasil

Tabel 4.8 Hasil Pengujian Black Box Testing - Monitoring Real-Time oleh Supervisor (FR-05)

No	Skenario Pengujian	Input	Expected Output	Actual Output	Status
BBT-06-01	Lihat daftar seluruh <i>checksheet</i>	Login Supervisor, buka Monitor <i>Checksheet</i>	Semua <i>checksheet</i> tampil dengan statusnya	Daftar <i>checksheet</i> tampil	Berhasil
BBT-06-02	Lihat detail parameter <i>checksheet</i>	Klik detail pada <i>checksheet</i>	Nilai parameter dan OK/NG/NA tampil lengkap	Detail tampil	Berhasil
BBT-06-03	Filter <i>checksheet</i> berdasarkan status	Pilih filter Menunggu Review	Hanya <i>checksheet</i> Menunggu Review yang tampil	Hasil filter sesuai	Berhasil

Tabel 4.9 Hasil Pengujian Black Box Testing - Notifikasi Otomatis (FR-08 dan FR-09)

No	Skenario Pengujian	Input	Expected Output	Actual Output	Status
BBT-07-01	Notifikasi reminder belum isi <i>checksheet</i>	Jam pengingat tiba, <i>checksheet</i> belum diisi	Notifikasi otomatis muncul di bell icon QC	Notifikasi diterima sesuai jam	Berhasil
BBT-07-02	Tidak ada notifikasi jika sudah diisi	<i>Checksheet</i> diisi sebelum jam pengingat	Scheduler tidak mengirim notifikasi	Notifikasi tidak dikirim	Berhasil
BBT-07-03	Notifikasi saat <i>checksheet</i> ditolak	Supervisor reject <i>checksheet</i>	QC menerima notifikasi beserta catatan alasan	Notifikasi dan catatan diterima	Berhasil

No	Skenario Pengujian	Input	Expected Output	Actual Output	Status
BBT-07-04	Notifikasi saat <i>checksheet</i> disetujui	Supervisor approve <i>checksheet</i>	QC menerima notifikasi persetujuan	Notifikasi diterima	Berhasil
BBT-07-05	Tandai notifikasi sebagai dibaca	Klik notifikasi atau Tandai Semua Dibaca	Badge merah berkurang, status menjadi dibaca	Badge berubah, status terbaru tampil	Berhasil

Tabel 4.10 Hasil Pengujian Black Box Testing - Summary dan Laporan (FR-06)

No	Skenario Pengujian	Input	Expected Output	Actual Output	Status
BBT-08-01	Lihat Summary sebagai Supervisor	Login Supervisor, buka menu Summary	Statistik dan grafik harian/mingguan tampil	Data summary tampil akurat	Berhasil
BBT-08-02	Lihat Summary sebagai Petugas QC	Login QC, buka menu Summary	Data summary sesuai role QC tampil	Data summary tampil	Berhasil
BBT-08-03	Verifikasi data Summary konsisten	Bandingkan summary dengan <i>checksheet</i> disubmit	Angka summary konsisten dengan data aktual	Data konsisten	Berhasil

**Tabel 4.11 Hasil Pengujian Black Box Testing - Kebutuhan Non-Fungsional
(NFR-01 s.d. NFR-05)**

No	Skenario Pengujian	Input	Expected Output	Actual Output	Status
BBT-09-01	Waktu respon maksimal 3 detik (NFR-02)	Submit <i>checksheet</i> / load halaman dashboard	Respon kurang dari atau sama dengan 3 detik	Respon dalam batas 3 detik	Berhasil
BBT-09-02	Kompatibilitas Google Chrome (NFR-05)	Akses sistem via Chrome	Seluruh fungsi berjalan normal	Berjalan normal	Berhasil
BBT-09-03	Kompatibilitas Mozilla Firefox (NFR-05)	Akses sistem via Firefox	Seluruh fungsi berjalan normal	Berjalan normal	Berhasil
BBT-09-04	Kompatibilitas Microsoft Edge (NFR-05)	Akses sistem via Edge	Seluruh fungsi berjalan normal	Berjalan normal	Berhasil
BBT-09-05	Antarmuka Bahasa Indonesia formal (NFR-01)	Navigasi seluruh halaman	Semua label, tombol, pesan menggunakan Bahasa Indonesia	Teks sesuai PUEBI	Berhasil
BBT-09-06	Sesi expired setelah 8 jam idle (NFR-03)	Biarkan idle 8 jam	Sistem otomatis logout, redirect ke Login	Sesi berakhir otomatis	Berhasil

Berdasarkan hasil pengujian Black Box Testing di atas, seluruh 51 skenario pengujian menunjukkan hasil Berhasil. Sistem *E-Checksheet* telah terbukti menghasilkan output yang sesuai dengan expected output pada setiap fungsi yang diuji, mencakup skenario valid maupun skenario negatif (input tidak valid, akses tidak sah, dan kondisi batas).

4.3 Pengujian System Usability Scale (SUS)

Selain pengujian fungsional menggunakan Black Box Testing, dilakukan pula pengujian usabilitas menggunakan metode *System Usability Scale* (SUS). SUS merupakan kuesioner standar yang terdiri atas 10 pernyataan dengan skala Likert 1–5. Metode ini banyak digunakan dalam evaluasi usabilitas perangkat lunak karena bersifat sederhana, cepat, dan telah teruji validitasnya secara luas dalam berbagai konteks penerapan sistem digital (Vlachogianni & Tselios, 2022). Skor SUS dihitung menggunakan rumus berikut: untuk pernyataan bernomor ganjil (Q1, Q3, Q5, Q7, Q9), kontribusi skor = nilai responden – 1; sedangkan untuk pernyataan bernomor genap (Q2, Q4, Q6, Q8, Q10), kontribusi skor = 5 – nilai responden. Total kontribusi seluruh 10 pernyataan kemudian dikalikan 2,5 untuk menghasilkan skor SUS dalam skala 0–100.

Kuesioner SUS disebarkan kepada 16 responden yang merupakan pengguna langsung sistem *E-Checksheets* di PT XYZ, dengan komposisi: 5 orang Admin (31,3%), 4 orang Supervisor (25%), dan 7 orang Petugas Quality Control (43,8%). Distribusi jawaban responden per pernyataan ditampilkan pada Tabel 4.12 berikut.

Tabel 4.12 Distribusi Jawaban Responden Kuesioner SUS

No	Pernyataan	SS (1)	S (2)	N (3)	TS (4)	STS (5)	Total Skor	Kontri busi SUS	+/-
Q1	Saya pikir akan sering menggunakan sistem ini	0	0	1	6	9	72	56	+

No	Pernyataan	SS (1)	S (2)	N (3)	TS (4)	STS (5)	Total Skor	Kontri busi SUS	+/-
Q2	Sistem terasa terlalu rumit	1	3	2	10	0	55	27	-
Q3	Sistem mudah digunakan	0	0	0	2	14	78	62	+
Q4	Saya perlu bantuan orang teknis untuk menggunakan sistem	10	4	2	0	0	22	56	-
Q5	Berbagai fungsi dalam sistem terintegrasi dengan baik	0	0	0	4	12	76	60	+
Q6	Terdapat terlalu banyak inkonsistensi dalam sistem	12	3	1	0	0	20	59	-
Q7	Sebagian besar orang dapat belajar menggunakan sistem ini dengan cepat	0	0	1	3	12	75	59	+
Q8	Sistem terasa membingungkan saat digunakan	12	4	0	0	0	20	60	-

No	Pernyataan	SS (1)	S (2)	N (3)	TS (4)	STS (5)	Total Skor	Kontri busi SUS	+/-
Q9	Saya merasa percaya diri menggunakan sistem ini	0	0	0	3	13	77	61	+
Q10	Saya perlu banyak belajar sebelum dapat menggunakan sistem	10	5	1	0	0	21	57	-
—	Total Kontribusi SUS							557	

Keterangan: SS = Sangat Setuju, S = Setuju, N = Netral, TS = Tidak Setuju, STS = Sangat Tidak Setuju. Tanda (+) = pernyataan positif (ganjil), tanda (-) = pernyataan negatif (genap).

Berdasarkan distribusi jawaban di atas, perhitungan skor SUS dilakukan dengan menjumlahkan seluruh nilai kontribusi per pertanyaan, kemudian menerapkan rumus: **Skor SUS = (Σ Kontribusi / N) $\times$ 2,5**, dengan N = 16 responden. Total kontribusi dari seluruh 10 pertanyaan adalah 557, sehingga: Skor SUS = $(557 / 16) \times 2,5 = 34,81 \times 2,5 = 92,19$.

Tabel 4.13 Hasil Pengujian System Usability Scale (SUS)

Jumlah Responden	Skor SUS	Grade	Adjective Rating
16 orang	92,19	A	Excellent / Best Imaginable

Berdasarkan skala penilaian SUS yang telah divalidasi secara luas dalam literatur evaluasi usabilitas (Vlachogianni & Tselios, 2022; Deshmukh & Chalmeta, 2024), skor 92,19 berada pada rentang $\geq 90,9$ yang dikategorikan sebagai ***Best Imaginable*** dengan Grade A. Hasil ini menunjukkan bahwa sistem *E-Checksheet* dinilai memiliki tingkat usabilitas yang sangat tinggi oleh seluruh kelompok pengguna. Responden dari ketiga peran pengguna (Admin, Supervisor, dan Petugas QC) secara konsisten menilai bahwa sistem mudah dipelajari, nyaman digunakan, dan tidak memerlukan bantuan teknis yang signifikan untuk pengoperasiannya. Dengan demikian, hasil pengujian SUS mengonfirmasi bahwa sistem *E-Checksheet* tidak hanya memenuhi kebutuhan fungsional (sebagaimana dibuktikan oleh Black Box Testing), tetapi juga memenuhi aspek kualitas non-fungsional yang berkaitan dengan kemudahan penggunaan (NFR-01 s.d. NFR-04).

BAB V KESIMPULAN

5.1 Kesimpulan

Berdasarkan hasil perancangan, implementasi, dan pengujian yang telah dilakukan, dapat disimpulkan bahwa Proyek Akhir ini telah berhasil mencapai tujuan yang ditetapkan. Berikut ini merupakan jawaban atas ketiga rumusan masalah yang telah dirumuskan pada Bab I. Sistem Aplikasi Web *E-Checksheets* untuk Optimalisasi Proses *Quality Control* pada PT XYZ berhasil dibangun menggunakan framework Laravel sebagai backend REST API dan Quasar Vue.js sebagai frontend Single Page Application (SPA), dengan MySQL sebagai basis data.

Berdasarkan hasil perancangan, implementasi, dan pengujian yang telah dilakukan, penelitian ini berhasil menjawab ketiga rumusan masalah yang telah ditetapkan pada Bab I, yaitu:

1. Menjawab rumusan masalah pertama, sistem *E-checksheet* berbasis web berhasil dirancang dan dibangun untuk menggantikan proses pengisian *checksheet* manual berbasis kertas di PT XYZ, sehingga mengurangi risiko kehilangan data dan meningkatkan efisiensi pencatatan.
2. Menjawab rumusan masalah kedua, sistem *E-checksheet* menyediakan fitur notifikasi pengingat otomatis berbasis Laravel Scheduler yang mengirimkan notifikasi kepada petugas QC apabila *checksheet* belum diisi sesuai jadwal yang ditetapkan, sehingga memastikan pengisian dilakukan tepat waktu.

3. Menjawab rumusan masalah ketiga, sistem *E-checksheet* menyediakan dashboard monitoring *real-time* bagi Supervisor untuk memantau status pengisian *checksheet* dan hasil pemeriksaan secara langsung, melakukan *approve* atau *reject*, serta melihat laporan *summary* harian dan mingguan.

Berdasarkan hasil pengujian menggunakan metode Black box Testing, seluruh 51 skenario pengujian menunjukkan hasil berhasil. Tidak ditemukan kesalahan kritis yang menghambat fungsi utama sistem, sehingga dapat disimpulkan bahwa sistem telah layak untuk digunakan dalam mendukung kegiatan quality control di PT XYZ.

5.2 Saran

Meskipun sistem *E-Checksheet* yang dikembangkan telah berhasil memenuhi seluruh kebutuhan fungsional yang ditetapkan, terdapat beberapa saran untuk pengembangan lebih lanjut:

1. Pengembangan fitur ekspor laporan dalam format PDF atau Excel agar supervisor dapat mencetak atau mendistribusikan laporan inspeksi secara lebih mudah.
2. Penambahan fitur manajemen parameter *checksheet* yang dapat dikonfigurasi secara dinamis oleh Admin, sehingga parameter pemeriksaan dapat disesuaikan tanpa perlu mengubah kode program.

DAFTAR PUSTAKA

Verwijns, C., & Russo, D. (2023). A theory of Scrum team effectiveness. *ACM Transactions on Software Engineering and Methodology*, 32(3), Article 74. <https://doi.org/10.1145/3571849>

Saravanos, A., & Curinga, M. (2023). Simulating the software development lifecycle: The waterfall model. *ArXiv Preprint ArXiv:2306.04360*. <https://doi.org/10.48550/arXiv.2306.04360>

Hofmann, M., Weber, C., & Braun, J. (2023). Digital quality control system for construction projects. *Buildings*, 13(5), 1210. <https://doi.org/10.3390/buildings13051210>

Kim, D., & Park, J. (2022). Web-based manufacturing quality control using integrated dashboard visualization. *Applied Sciences*, 12(9), 4567.

Lee, J., Bagheri, B., & Kao, H. A. (2021). Industry 4.0 technologies for manufacturing quality improvement. *Procedia CIRP*, 104, 47–52.

Moe, N. B., Smite, D., & Ågerfalk, P. J. (2021). Understanding distributed agile teams: A case study of collaboration in global software development. *Information and Software Technology*, 134, 106561.

Rinaldi, A., & Purnomo, D. (2022). Progressive web application for preventive maintenance using Agile methodology. *Jurnal Informatika Poltek GT*, 6(2), 77–85.

Santoso, R., & Ramadhan, F. (2024). Development of *checksheet* walk-by inspection website for condition-based monitoring. Prosiding Seminar Nasional Teknologi Manufaktur, 5(1), 221–230.

Besterfield, D. H., Besterfield-Michna, C., & Besterfield, G. (2022). Total quality management (5th ed.). Pearson.

Suryawan, D. (2024). Perancangan dashboard digital quality assurance berbasis web di PT Dharma Polimetal Tbk. Jurnal Venus, 13(2), 101–112.

Zhang, H., Liu, Y., & Chen, X. (2023). Data-driven digital quality management in multi-level manufacturing systems. Journal of Intelligent Manufacturing, 34(7)

Pratama, R., & Wijaya, S. (2023). Pengembangan aplikasi web dinamis menggunakan PHP dalam sistem informasi berbasis server. Jurnal Teknologi Informasi dan Komunikasi, 11(2), 88–97.

Saputra, D., Nugroho, H., & Kurniawan, A. (2023). Implementasi framework Laravel dalam pengembangan sistem informasi berbasis RESTful API. Jurnal Ilmu Komputer dan Informatika, 8(1), 34–45.

Hidayat, F., & Nugroho, P. (2023). Pengembangan antarmuka pengguna responsif menggunakan Vue.js dan Quasar Framework pada aplikasi web berbasis SPA. Jurnal Sistem Informasi dan Teknologi, 5(3), 112–121.

Rahman, A., & Setiawan, B. (2023). Optimasi pengelolaan basis data relasional menggunakan MySQL pada sistem informasi manajemen industri. Jurnal Rekayasa Perangkat Lunak, 10(2), 67–76.

Vlachogianni, P., & Tselios, N. (2022). Perceived usability evaluation of educational technology using the System Usability Scale (SUS): A systematic review. Journal of Research on Technology in Education, 54(3), 392–409. <https://doi.org/10.1080/15391523.2020.1867938>

Deshmukh, A. M., & Chalmeta, R. (2024). Validation of system usability scale as a usability metric to evaluate voice user interfaces. *PeerJ Computer Science*, 10, e1918. <https://doi.org/10.7717/peerj-cs.1918>

DAFTAR LAMPIRAN

Lampiran A: Dokumen Proses Pengumpulan Requirement

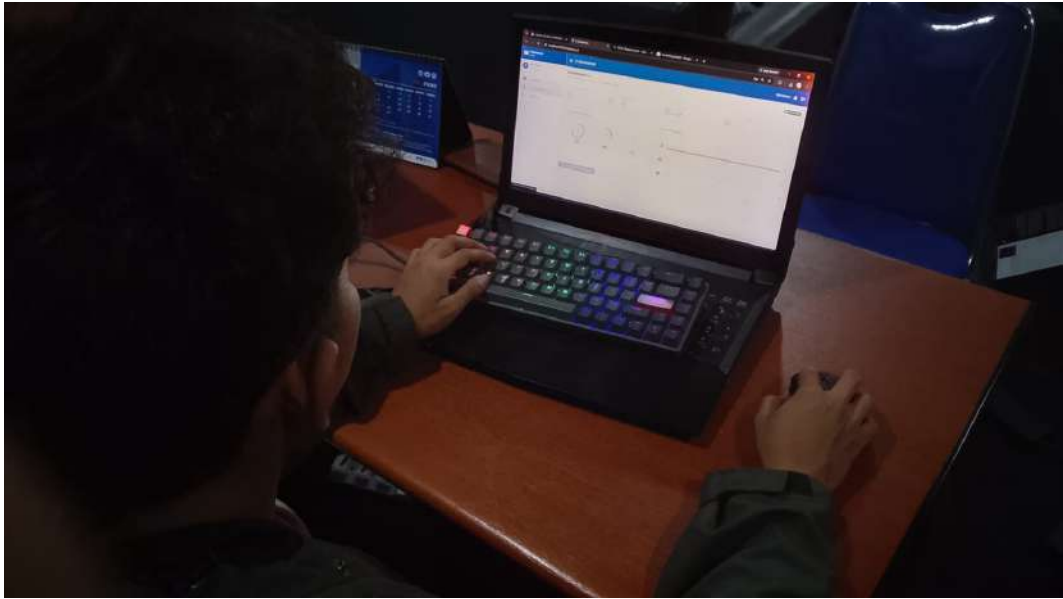
PT SAT NUSAPERSADA Tbk
FACILITY ENGG. & UTILITY DEPT

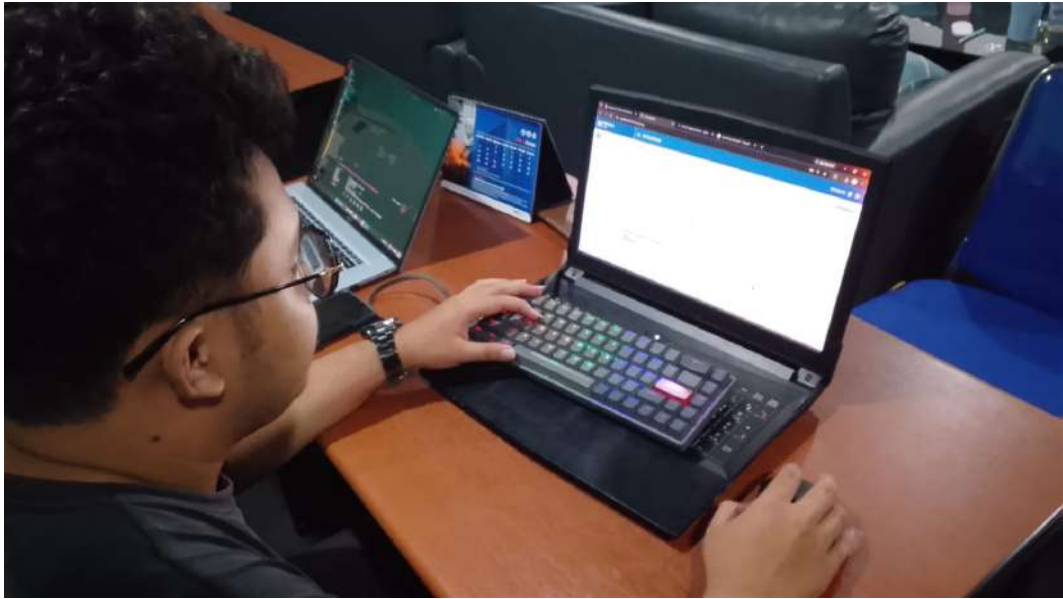
Factory :
Dept :

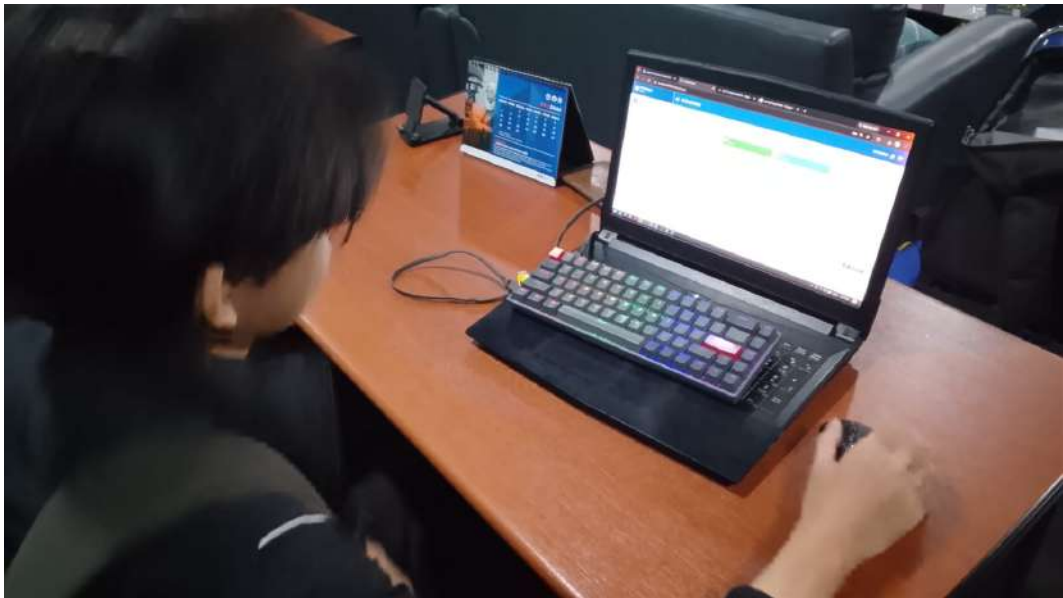
AC Model / Kapasitas :
AC Area :

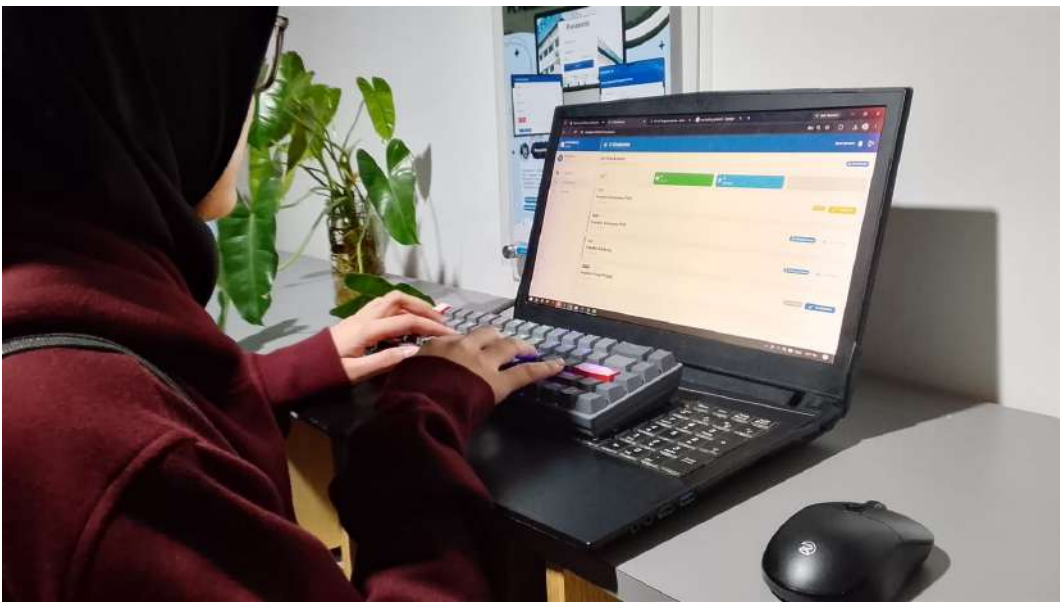
No.	Service (At the ...)	Location (At the ...)	Remarks
1	...	...	...
2	...	...	...
3	...	...	...
4	...	...	...
5	...	...	...
6	...	...	...
7	...	...	...
8	...	...	...
9	...	...	...
10	...	...	...
11	...	...	...
12	...	...	...
13	...	...	...
14	...	...	...
15	...	...	...
16	...	...	...
17	...	...	...
18	...	...	...
19	...	...	...
20	...	...	...
21	...	...	...
22	...	...	...
23	...	...	...
24	...	...	...
25	...	...	...
26	...	...	...
27	...	...	...
28	...	...	...
29	...	...	...
30	...	...	...
31	...	...	...
32	...	...	...
33	...	...	...
34	...	...	...
35	...	...	...
36	...	...	...
37	...	...	...
38	...	...	...
39	...	...	...
40	...	...	...
41	...	...	...
42	...	...	...
43	...	...	...
44	...	...	...
45	...	...	...
46	...	...	...
47	...	...	...
48	...	...	...
49	...	...	...
50	...	...	...

Form No : FM-SN-FAC-018 Rev-00











Lampiran B: Link Produk / GitHub

<https://github.com/IceyStevie/E-Checksheet>

Lampiran C: Dokumen Pengujian Black Box Testing

[Template UAA Informatika Steve](#)- Tanda Tangan