

BAB II

TINJAUAN PUSTAKA

2.1 Studi Sebelumnya

Sumarudin dkk. merancang Sistem Informasi Penelitian Internal di POLINDRA untuk menutup celah proses manual via email pada pengajuan, penilaian, hingga pelaporan usulan penelitian. Penulis memilih basis web karena fleksibel dan mudah dioperasikan, memakai Laravel dan MariaDB, serta menguji fungsionalitas dengan black-box testing yang menunjukkan fitur berjalan baik. Studi ini menerapkan Kanban untuk merespons perubahan kebutuhan dengan cepat, menata pekerjaan dalam papan yang terdiri dari kolom *Backlog*, To-Do, Doing, Testing, dan Done. Penulis juga menetapkan batas WIP agar tim fokus dan aliran kerja terjaga. Hasilnya, proses pengusulan hingga persetujuan menjadi lebih cepat, terintegrasi, dan mudah dipantau oleh dosen maupun pengelola penelitian. Selain itu, artikel menegaskan struktur tiga papan Kanban utama di fase requirement, develop, dan testing sehingga jalur pengembangan jelas dan risiko cacat terdeteksi dini [6].

Syah dkk. mengatasi persoalan presensi, izin cuti, dan penggajian yang masih manual di Royal Laundry Pontianak. Permasalahan meliputi pencatatan kertas untuk presensi, slip gaji, rekap bulanan, serta pengelolaan cuti yang menyulitkan karyawan dan rawan salah. Solusi yang dibangun adalah sistem informasi presensi dan penggajian berbasis web dengan metode Kanban karena siklus pengembangan yang cepat dan efektif. Hasil penelitian menghasilkan sistem yang digunakan di Royal Laundry. Penerapan Kanban dijelaskan rinci, mulai dari penetapan tahapan *Backlog*, To-Do, Doing, Testing, hingga Done. Tiap siklus berfokus pada pengembangan per fitur dan evaluasi pada setiap putaran, sehingga prioritas dan timeline bisa dipantau. Penulis juga menerangkan karakter Kanban sebagai sistem visual untuk meminimalkan kemacetan dan membatasi pekerjaan berjalan agar produktivitas meningkat [7].

Lubis dkk. memotret layanan padat di kantor imigrasi dan masalah pengelolaan keluhan publik yang belum memadai. Keluhan sulit ditelusuri kembali karena pencatatan belum terintegrasi. Penelitian merancang sistem informasi helpdesk berbasis web agar data keluhan terkumpul rapi, penerimaan dan pencatatan lebih cepat, serta laporan

keluhan bisa menjadi dasar perbaikan prosedur. Kanban dipilih untuk meningkatkan manajemen keluhan melalui pengawasan alur kerja secara visual dan efisien. Lubis dkk. juga menegaskan langkah penerapan Kanban mulai dari identifikasi alur, pembuatan papan dan kartu, pembatasan WIP, pemindahan kartu per fase, hingga monitoring berkala untuk menemukan bottleneck [8].

Widjaja dkk. merancang Sistem Manajemen Proyek berbasis web untuk Divisi Digital Learning PT Kurnia Ciptamoda Gemilang. Sistem dikembangkan dengan Next.js, menerapkan Kanban untuk manajemen tugas, dan memakai model SDLC Waterfall pada proses pengembangannya. Fitur mencakup workspace, manajemen anggota, proyek dan tugas, prioritas, tenggat, unggahan hasil kerja, komentar, revisi, notifikasi, serta visualisasi Kanban dan kalender. Evaluasi memakai Black Box Testing dan seluruh fitur berjalan sesuai spesifikasi. Paper ini menekankan alasan pemilihan Waterfall karena kebutuhan jelas sejak awal, sementara Kanban digunakan di sisi pengguna untuk mengelola alur kerja harian. Studi juga menyoroti gap riset. Penerapan Kanban banyak dikaji di tim teknis, sementara konteks non-teknis seperti tim pelatihan digital masih minim. Kontribusi paper adalah memfokuskan rancangan pada kebutuhan unit kerja tersebut [9].

Arribe dkk. mengembangkan fitur absensi pada aplikasi OPS PEDIA di PT Nielsen dengan metode Waterfall. Masalah awal adalah proses absensi manual yang rawan salah dan berpotensi kecurangan. Peneliti menyusun tahapan berurutan mulai requirement, desain, implementasi, verifikasi, hingga pemeliharaan. Pengumpulan data meliputi observasi, studi pustaka, analisis kebutuhan, dan permodelan UML. Hasilnya. Fitur absensi meningkatkan efisiensi dan akurasi pencatatan. Potensi kecurangan turun. Perusahaan memperoleh sistem terintegrasi yang lebih andal untuk pengelolaan kehadiran karyawan. Studi ini menegaskan kekuatan Waterfall saat kebutuhan stabil dan dokumentasi awal kuat, namun juga menyoroti sifat sekuensial yang kurang ramah perubahan mendadak [10].

Sofiati menerapkan metode Rapid Application Development untuk membangun Sistem Informasi Bimbingan Konseling di SMKN 1 Sijunjung. Layanan BK sebelumnya mengandalkan tatap muka dengan waktu terbatas dan pencatatan manual. Peneliti merancang sistem web dengan PHP dan MySQL. Fitur meliputi manajemen data siswa,

jurusan, jenis pelanggaran, konsultasi, serta pencetakan surat laporan. Pengujian memakai Black Box Testing. Tingkat kepuasan pengguna tercatat 80 persen. Studi ini menunjukkan RAD efektif mempercepat pengembangan dengan workshop desain intensif, iterasi cepat, dan kolaborasi dekat dengan pengguna. Kelebihan ini relevan untuk layanan sekolah yang butuh respons cepat dan perbaikan berulang [11].

Aprillisa dkk. merancang Sistem Informasi Inventory di CV Multikom dengan metode Prototype. Proses bisnis awal masih manual. Stok masuk dan keluar dicatat di kertas lalu diketik ulang. Data pengiriman dan pelanggan belum terintegrasi. Peneliti memilih Prototype agar kebutuhan pengguna cepat terklasifikasi lewat evaluasi berulang. Pemodelan terstruktur memakai Data Flow Diagram dan perbaikan proses ditopang flowchart alur barang. Pengujian fungsional per menu valid. Sistem menampilkan grafik untuk membantu pengambilan keputusan. Studi ini memperlihatkan Prototype efektif untuk mengurangi ambiguitas kebutuhan dan untuk menguji rancangan cepat sebelum implementasi penuh [12].

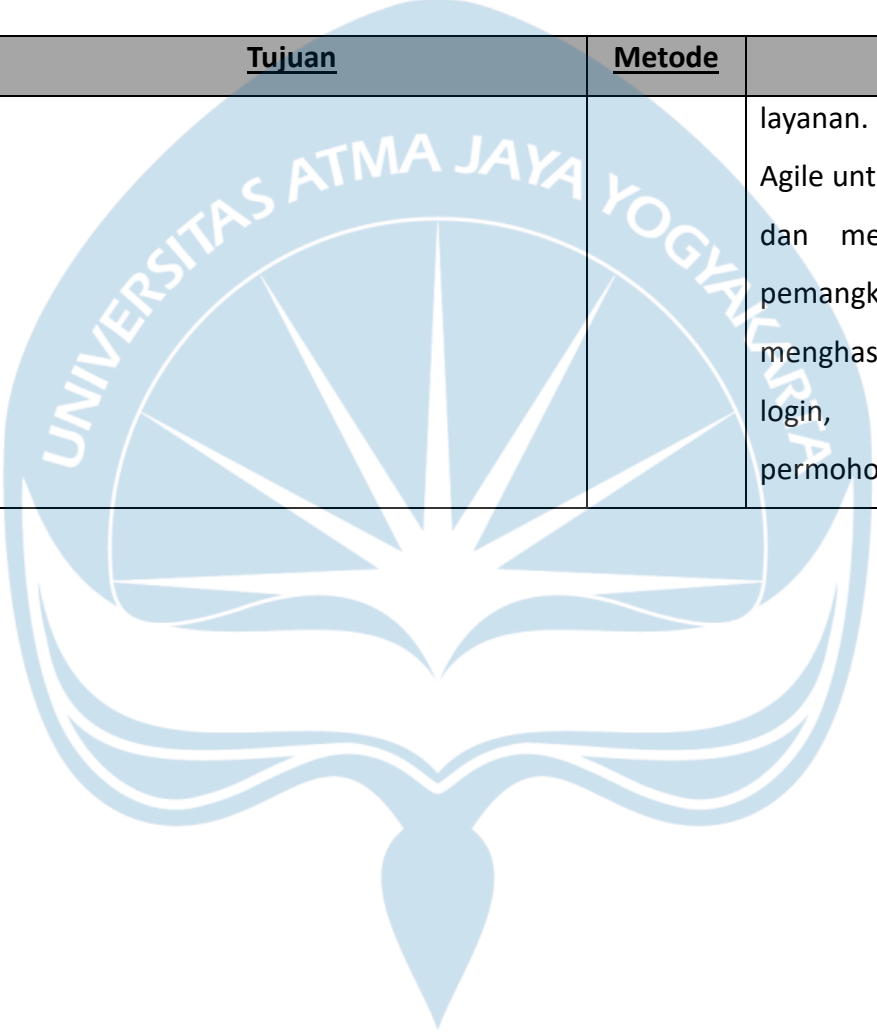
Atallah dkk. menerapkan Agile Scrum pada Sistem Informasi Surat Izin Penelitian di Bakesbangpol Lombok Tengah. Peneliti memakai pendekatan deskriptif kualitatif dengan observasi, wawancara, dan studi pustaka. Proses pengembangan dimulai dari analisis kebutuhan, desain dengan UML, implementasi, hingga pengujian. Hasil menunjukkan Scrum efektif meningkatkan efisiensi dan kualitas. Sistem web mempercepat penerbitan izin dan memperbaiki layanan. Studi ini memperkuat manfaat praktik Agile saat kebutuhan berubah cepat dan koordinasi lintas peran intensif [13]. Tabel perbandingan studi sebelumnya dapat dilihat pada tabel 2.1.

Tabel 2.1 Studi Sebelumnya

<u>No.</u>	<u>Penulis</u>	<u>Tahun</u>	<u>Tujuan</u>	<u>Metode</u>	<u>Hasil</u>
1.	Sumarudin dkk.[6]	2021	Membangun Sistem Informasi Penelitian Internal POLINDRA untuk mengelola usulan penelitian dari pengajuan hingga pelaporan agar proses lebih cepat dan terintegrasi.	Kanban	Fitur berjalan baik berdasarkan black box. Alur kerja lebih terstruktur. Pengusulan dan persetujuan lebih cepat dan mudah dipantau.
2.	Syah dkk. [7]	2024	Mengatasi presensi, cuti, dan penggajian Royal Laundry yang masih manual agar pengelolaan data karyawan lebih efisien dan akurat.	Kanban	Sistem web presensi dan penggajian digunakan di Royal Laundry. Mendukung presensi selfie dan rekap data. Memudahkan monitoring banyak cabang. Kanban efektif untuk skala kecil.
3.	Lubis dkk. [8]	2023	Merancang Sistem Informasi Helpdesk Imigrasi berbasis web untuk mengelola keluhan secara terintegrasi agar pencatatan lebih cepat dan mudah ditelusuri.	Kanban	Menghasilkan rancangan alur kerja Kanban, daftar tugas, dan model kebutuhan aktor sebagai dasar implementasi. Menyediakan fungsi cetak laporan pengaduan.
4.	Widjaja dkk. [9]	2025	Merancang sistem manajemen proyek berbasis web untuk Divisi Digital Learning PT Kurnia Ciptamoda Gemilang agar	Waterfall	Tersedia fitur workspace, anggota, proyek, tugas, prioritas, tenggat, papan Kanban, dan kalender. Fitur lulus uji black box. Efisiensi

<u>No.</u>	<u>Penulis</u>	<u>Tahun</u>	<u>Tujuan</u>	<u>Metode</u>	<u>Hasil</u>
			pemantauan proyek lebih terstruktur dan tepat waktu.		kerja meningkat dan miskomunikasi berkurang.
5.	Arribe dkk. [10]	2023	Merancang Sistem Informasi Absensi pada PT Nielsen agar pencatatan kehadiran lebih efisien dan akurat.	Waterfall	Implementasi fitur kehadiran dalam aplikasi OPS PEDIA berhasil mengatasi masalah ketidakakuratan dan penipuan. Fitur baru ini tidak hanya meningkatkan efisiensi dan akurasi dalam pencatatan absensi tetapi juga mengurangi biaya operasional yang terkait dan mengeliminasi potensi penipuan absensi.
6.	Sofiati [11]	2024	Membangun Sistem Informasi Bimbingan dan Konseling di SMKN 1 Sijunjung agar layanan konsultasi lebih cepat dan terdata.	RAD	Sistem web mencakup manajemen data siswa, jurusan, jenis pelanggaran, konsultasi, dan cetak surat laporan. Pengujian black box menunjukkan fungsi berjalan sesuai skenario. Pengguna melaporkan tingkat kepuasan 80 persen. Sistem memudahkan admin, guru, dan siswa dalam proses layanan BK yang sebelumnya terbatas waktu dan masih manual.

<u>No.</u>	<u>Penulis</u>	<u>Tahun</u>	<u>Tujuan</u>	<u>Metode</u>	<u>Hasil</u>
7.	Aprillisa dkk. [12]	2024	Merancang Sistem Informasi Inventory Barang pada CV Multikom agar pendataan stok, transaksi, dan pengiriman terintegrasi.	Prototype	Sistem menggantikan proses manual pada pendataan barang masuk dan keluar, data pengiriman, dan data pelanggan yang sebelumnya tersebar. Pemodelan terstruktur menggunakan DFD. Pengujian black box per menu menunjukkan seluruh fungsi valid, termasuk dashboard, barang masuk, barang keluar, transaksi, surat jalan, dan logout. Sistem membantu karyawan gudang serta pimpinan melalui penyederhanaan proses dan mengurangi penumpukan berkas. Informasi disajikan dalam bentuk grafik untuk mendukung pengambilan keputusan.
8.	Atallah dkk. [13]	2024	Merancang Sistem Informasi Surat Izin Penelitian di Bakesbangpol Lombok Tengah agar layanan perizinan lebih cepat dan berkualitas.	Agile Scrum	Proses pengembangan dimulai dari analisis kebutuhan, desain menggunakan UML, dilanjutkan implementasi dan pengujian. Penerapan diharapkan mempercepat penerbitan surat izin dan memperbaiki



<u>No.</u>	<u>Penulis</u>	<u>Tahun</u>	<u>Tujuan</u>	<u>Metode</u>	<u>Hasil</u>
					layanan. Peneliti menekankan kemampuan Agile untuk mengakomodasi perubahan cepat dan mendorong kolaborasi tim dengan pemangku kepentingan. Hasil perancangan menghasilkan prototipe halaman inti seperti login, registrasi, dashboard admin, permohonan masuk, dan pelayanan surat izin.

2.1.1 Analisis Gap dan Kebaruan Penelitian

Berdasarkan pemetaan studi pada Tabel 2.1, terdapat beberapa kesenjangan yang relevan dengan konteks otomasi pelaporan teknisi lapangan. Sumarudin dkk., Syah dkk., dan Lubis dkk. menitikberatkan otomasi proses administratif di lingkungan instansi pendidikan, jasa laundry, dan layanan helpdesk imigrasi. Otomasi diwujudkan melalui formulir web, pengelolaan data karyawan, serta pencatatan keluhan pelanggan [6], [7], [8]. Namun ketiga penelitian tersebut belum secara eksplisit menangani skenario pelaporan pekerjaan teknisi di lapangan yang memerlukan pengelolaan dokumentasi berupa foto kondisi awal dan akhir, bukti penggantian komponen, serta kronologi penanganan sebagai dasar penyusunan Berita Acara Serah Terima (BAST).

Widjaja dkk. dan Arribe dkk. berfokus pada manajemen proyek dan pencatatan kehadiran internal organisasi. Kedua sistem menyediakan fitur pelacakan tugas, tenggat waktu, dan absensi terpusat sehingga pengelolaan sumber daya menjadi lebih terstruktur dan akuntabel. Meskipun demikian, orientasi keduanya tetap pada pemantauan aktivitas internal, bukan pada integrasi dokumentasi kerja lapangan ke dalam sistem pelaporan formal yang menjadi kewajiban perusahaan jasa di hadapan mitra. Struktur data yang diolah masih didominasi informasi tekstual seperti deskripsi tugas, jadwal, dan status kehadiran, bukan paket bukti pekerjaan teknis yang lengkap [9], [10].

Studi Sofiati, Aprillisa dkk., dan Atallah dkk. menunjukkan variasi metode pengembangan seperti Rapid Application Development, Prototype, dan Agile Scrum untuk membangun sistem informasi layanan sekolah, inventory barang, serta perizinan penelitian. Konteks domain dan jenis layanan yang diotomasi berbeda dengan kebutuhan industri jasa pemeliharaan teknis. Pola input pada penelitian tersebut masih didominasi entri data teks dan angka yang dimasukkan langsung ke antarmuka web oleh pengguna. Tidak ditemukan mekanisme yang secara khusus merancang alur integrasi dokumentasi lapangan dalam bentuk foto dan keterangan teknis dari kanal komunikasi populer seperti WhatsApp ke basis data terpusat yang kemudian diolah menjadi laporan formal [11], [12], [13].

Dari tinjauan tersebut dapat disimpulkan bahwa belum ada penelitian yang secara eksplisit mengintegrasikan input foto dan dokumentasi teknisi lapangan langsung ke dalam sistem otomasi laporan yang menghasilkan dokumen Berita Acara Serah Terima siap pakai. Celah lain terletak pada belum adanya rancangan arsitektur yang memanfaatkan alur multiplatform, misalnya pemanfaatan WhatsApp sebagai kanal penerimaan dokumentasi di sisi teknisi dan aplikasi web sebagai antarmuka pengolahan serta pencetakan laporan di sisi admin dan manajemen. Selain itu, belum ada model yang secara spesifik diarahkan untuk memenuhi tuntutan pelaporan cepat dan akurat pada perusahaan jasa pemeliharaan teknis yang terikat perjanjian layanan dengan mitra.

Kebaruan penelitian ini terletak pada perancangan dan implementasi sistem otomasi pelaporan teknisi berbasis web yang mengintegrasikan basis data terpusat dengan kanal input multiplatform. Dokumentasi pekerjaan teknisi dikirim melalui WhatsApp dalam bentuk foto dan keterangan singkat, kemudian dikonversi menjadi data terstruktur di aplikasi web. Sistem memanfaatkan template laporan yang terotomasi berdasarkan jenis pekerjaan, misalnya instalasi baru, perbaikan, atau pemeliharaan berkala. Dengan demikian data teknisi, perangkat, lokasi, temuan di lapangan, dan tindakan korektif dapat langsung terpetakan ke susunan laporan yang sesuai standar perusahaan dan kebutuhan mitra.

Dari sisi domain, penelitian ini memfokuskan otomasi pada industri jasa pemeliharaan teknis yang menuntut pembuatan Berita Acara Serah Terima secara cepat, konsisten, dan terdokumentasi. Kombinasi integrasi kanal WhatsApp ke aplikasi web, pemanfaatan basis data terpusat, serta otomasi template pelaporan menjadikan rancangan sistem berbeda dengan studi sebelumnya yang umumnya hanya mengotomasi entri formulir administratif. Kontribusi praktis penelitian ini adalah model otomasi pelaporan teknisi yang dapat direplikasi pada perusahaan jasa sejenis. Kontribusi teoretisnya berupa kerangka integrasi dokumentasi lapangan ke dalam sistem pelaporan terpusat

yang menggabungkan konsep sistem informasi otomasi, basis data terintegrasi, dan manajemen proses bisnis berbasis Kanban.

2.2 Dasar Teori

2.2.1 Sistem Informasi Otomisasi

Sistem informasi otomasi dapat dipahami sebagai sistem informasi yang dirancang untuk mengotomatisasi prosedur kerja yang berulang melalui kombinasi perangkat lunak, basis data, dan aturan bisnis terstruktur sehingga intervensi manual dapat diminimalkan dan standar layanan dapat ditegakkan secara konsisten. Otomasi tidak hanya menggantikan aktivitas fisik, tetapi juga menertibkan alur persetujuan, pelacakan status, serta dokumentasi bukti kerja agar seluruh proses tercatat dan dapat diaudit [1], [14].

Pada lingkungan organisasi, sistem informasi otomasi diimplementasikan untuk menggantikan atau menyederhanakan aktivitas manual yang berulang sehingga aliran kerja menjadi lebih cepat, konsisten, dan terukur. Penerapannya sering memanfaatkan integrasi antarmuka web, modul autentikasi, basis data terpusat, serta alur persetujuan daring. Tujuannya agar proses operasional seperti check in dan check out, pencatatan transaksi, atau alur surat menyurat dapat berjalan secara swalayang dengan jejak audit yang rapi. Penelitian pada layanan asrama perguruan tinggi menunjukkan bahwa perancangan layanan check in dan check out berbasis web mampu menstandarkan prosedur, mengurangi antrean, serta meningkatkan akurasi pencatatan karena data diproses secara real time pada satu sumber kebenaran [14].

Otomasi juga membantu meningkatkan kemampuan pelaporan instansi. Sistem otomatisasi laporan mendukung penggabungan data dari berbagai unit kerja, seperti penelitian dan pengabdian masyarakat, memperkuat proses validasi data, serta menghasilkan laporan yang konsisten sesuai dengan indikator kinerja tanpa perlu menghabiskan waktu untuk mengolah data secara manual. Pendekatan bertahap yang mencakup analisis kebutuhan, perencanaan, penerapan, dan evaluasi terbukti efektif dalam mengurangi kesalahan saat mengisi data, mempercepat pembuatan laporan ringkasan, serta meningkatkan

kemampuan pimpinan dalam memantau perkembangan dan tren pencapaian. Dari sisi pengguna, desain antarmuka yang sederhana dan terarah membuat proses pembelajaran lebih mudah, sehingga pengisian dan verifikasi bukti menjadi lebih cepat dan efisien [1].

Dalam bidang manajemen administrasi kantor, konsep e-Office menunjukkan cara otomatisasi mengurangi ketergantungan pada dokumen fisik dengan fitur seperti registrasi pengguna, pengelolaan arsip, template dokumen, penomoran otomatis, dan pelacakan status secara online. Penggunaan sistem ini menerapkan proses penyelesaian tugas lebih cepat, mengurangi risiko hilangnya dokumen, serta memudahkan pencarian kembali berdasarkan data metadata. Dengan diintegrasikan ke dalam kebijakan peran dan hak akses, sistem ini mampu menjaga keamanan informasi tanpa mengurangi kecepatan pelayanan, sekaligus menyediakan catatan aktivitas untuk keperluan audit internal [15].

2.2.2 Database

Database (basis data) merupakan bagian penting dari sistem informasi yang bertugas menyimpan, mengatur, dan menampilkan data secara terorganisir agar lebih mudah diakses dan dikelola. Dalam satu dekade terakhir, kebutuhan akan kemampuan untuk berkembang dan selalu tersedia mendorong perusahaan menggunakan basis data terdistribusi, di mana data disimpan di beberapa titik yang saling terhubung. Informasi terbaru menunjukkan bahwa pendekatan terdistribusi meningkatkan ketahanan terhadap kegagalan dan fleksibilitas dalam menangani beban kerja, tetapi memerlukan desain yang matang dalam hal replikasi, partisi, serta konsistensi berdasarkan model CAP yang dipilih [16].

Dalam perbandingan model database, basis data relasional (SQL) menyimpan data dalam bentuk tabel dua dimensi (baris dan kolom) dengan skema yang ditentukan sejak awal, mendukung transaksi ACID, bersifat skalabel secara vertikal, serta menyediakan bahasa kueri yang kuat. Kelebihannya mencakup kemudahan dalam penggunaan, keamanan, konsistensi, dan kemampuan pemulihan yang baik, serta cocok untuk struktur data kompleks.

Namun, kekurangannya antara lain memerlukan ruang penyimpanan yang relatif besar, kurang cocok untuk menangani data tak terstruktur yang berukuran besar, serta membutuhkan perancangan skema yang matang. Sebaliknya, NoSQL dirancang untuk model data tertentu seperti key-value, dokumen, kolom-lebar, dan graf dengan skema yang fleksibel, mendorong penskalaan secara horizontal, lebih menekankan throughput dan ketersediaan, serta memberikan fleksibilitas dan performa tinggi untuk pola akses tertentu. Namun, dukungan administrasi bisa lebih terbatas dan memerlukan keahlian teknis yang cukup. Perbedaan lainnya adalah SQL lebih unggul dalam mengelola transaksi multi-baris dan menjaga konsistensi tinggi, sedangkan NoSQL lebih tepat untuk menangani himpunan data besar dan skenario real-time. Performa SQL sangat bergantung pada optimasi kueri, indeks, dan struktur tabel, sedangkan performa NoSQL lebih dipengaruhi oleh ukuran klaster, latensi jaringan, dan beban aplikasi [17].

Pendekatan menggunakan basis data terdistribusi juga penting ketika kebutuhan operasional yang menuntut tingkat ketersediaan tinggi dan beban akses terdistribusi di berbagai lokasi. Implementasi RESTful API sebagai lapisan akses dengan pertukaran data dalam format JSON membantu memisahkan aplikasi klien dari mesin basis data utama, sehingga memudahkan proses replikasi basis data saat terjadi gangguan serta mendukung integrasi layanan seperti katalog daring dan pelaporan. Dalam hal kinerja, rancangan tersebut terbukti mempercepat akses data sekitar 30% dibandingkan sistem terpusat, mengurangi waktu sinkronisasi menjadi sekitar 2 detik per transaksi, serta memangkas beban server pusat hingga 70% karena permintaan terdistribusi ke lokasi terdekat. Selain itu, strategi *disaster recovery* mengurangi risiko kehilangan data tanpa mengorbankan konsistensi operasional. Prabowo dkk. menegaskan bahwa perpindahan dari basis data terpusat ke arsitektur terdistribusi yang dikembangkan secara iteratif, misalnya dengan pendekatan RAD dapat meningkatkan efisiensi, reliabilitas, dan pengalaman pengguna pada sistem informasi perpustakaan modern [18].

Dalam konteks penelitian ini, basis data terintegrasi merujuk pada satu sumber data terpusat yang menggabungkan informasi teknis, perangkat,

pekerjaan lapangan, dan status Berita Acara Serah Terima. Integrasi dicapai melalui Application Programming Interface yang menghubungkan kanal input seperti WhatsApp Gateway dengan aplikasi web, sehingga data foto, teks, dan metadata pekerjaan dapat tersimpan dalam skema yang seragam. Mekanisme sinkronisasi data secara real time memastikan perubahan pada satu modul segera tercermin pada modul lain, misalnya ketika teknisi mengunggah dokumentasi baru maka status pekerjaan dan ketersediaan laporan di sisi admin ikut diperbarui [16], [17]. Pendekatan ini sejalan dengan prinsip database terdistribusi dan pemanfaatan layanan web untuk meningkatkan ketersediaan, konsistensi operasional, serta keandalan sistem.

2.2.3 Agile

Agile merupakan pendekatan pengembangan perangkat lunak yang menekankan kolaborasi intensif, perubahan yang cepat direspons, dan penyampaian produk secara inkremental. Metode ini muncul sebagai kritik terhadap model tradisional yang sangat kaku terhadap perubahan kebutuhan. Agile memecah pembangunan sistem ke dalam siklus pendek sehingga setiap iterasi menghasilkan *increment* yang siap dievaluasi. Dengan memprioritaskan komunikasi antara pengembang dan pengguna, kebutuhan yang belum jelas di awal dapat dikonfirmasi secara bertahap. Pendekatan ini terbukti meningkatkan kepuasan pengguna, mempercepat *time-to-market*, serta mengurangi risiko kesalahan konseptual pada tahap akhir pengembangan [19].

Kerangka kerja dalam Agile memiliki variasi, namun yang paling populer adalah Scrum dan Kanban. Scrum terstruktur dengan peran dan seremonial yang baku seperti Sprint, Daily Scrum, Sprint Review, dan Retrospective, yang menjaga ritme kerja dan meningkatkan transparansi progres. Sebaliknya, Kanban menitikberatkan pada visualisasi alur kerja serta pembatasan pekerjaan berjalan (WIP Limit) sehingga tim fokus menyelesaikan tugas tanpa multitasking berlebih. Studi literatur menunjukkan bahwa pemilihan metode hendaknya menyesuaikan sifat proyek. Scrum ideal ketika perubahan dapat dijadwalkan per iterasi, sedangkan Kanban lebih unggul pada alur kerja kontinu yang sangat dinamis.

Tidak jarang, organisasi mengadopsi hybrid Agile dengan memadukan keunggulan keduanya [19], [20].

Pada penelitian ini, Agile tidak diadopsi sebagai kerangka kerja formal yang lengkap seperti Scrum. Prinsip dasarnya dimanfaatkan untuk mendukung pengembangan sistem, antara lain iterasi bertahap, kolaborasi intensif dengan pemangku kepentingan, dan kemampuan merespons perubahan kebutuhan [2], [19]. Nilai nilai tersebut diimplementasikan melalui penggunaan Kanban sebagai metode pengelolaan tugas tim pengembang. Dengan demikian, pembahasan Agile pada dasar teori berfungsi sebagai payung konseptual yang menjelaskan alasan pemilihan Kanban dan pendekatan iteratif, bukan sebagai metode pengembangan terpisah yang dimodelkan secara rinci di dalam sistem.

2.2.4 Sistem Pelaporan

Sistem pelaporan merupakan subsistem dari sistem informasi yang berfungsi mengumpulkan, mengolah, dan menyajikan data menjadi informasi terstruktur dalam bentuk laporan yang dapat digunakan sebagai dasar pengambilan keputusan [1], [3]. Laporan biasanya tersusun dalam format yang sudah distandardkan sehingga memudahkan pembacaan dan perbandingan dari waktu ke waktu. Kualitas sistem pelaporan ditentukan oleh akurasi data, ketepatan waktu, tingkat kelengkapan informasi, dan kemampuan melacak sumber data yang digunakan.

Dalam lingkungan organisasi, sistem pelaporan modern tidak hanya menghasilkan dokumen untuk kebutuhan internal, tetapi juga mendukung pemenuhan kewajiban eksternal seperti laporan kepada regulator, mitra bisnis, atau klien [1], [5]. Otomasi pelaporan memanfaatkan basis data terpusat dan aturan bisnis yang jelas agar angka dan narasi yang disajikan konsisten dengan fakta lapangan. Penggunaan template laporan, penomoran otomatis, dan fitur pencarian historis membantu mengurangi kesalahan pengetikan manual dan mempercepat proses penyusunan dokumen [3], [5].

Penelitian ini menempatkan sistem pelaporan sebagai komponen inti karena keluaran utama yang dihasilkan berupa laporan pekerjaan teknisi dan

Berita Acara Serah Terima bagi mitra perusahaan. Sistem dirancang untuk menarik data teknisi, pekerjaan, dan dokumentasi lapangan secara langsung dari basis data lalu memetakannya ke struktur laporan yang telah ditentukan. Dengan demikian, setiap entri pekerjaan lapangan secara otomatis memiliki jejak dokumentasi dan laporan yang dapat dilacak kembali tanpa proses penyalinan ulang di luar sistem.

2.2.5 Arsitektur Aplikasi Berbasis Web

Aplikasi berbasis web umumnya dibangun di atas arsitektur client server yang memisahkan lapisan presentasi di sisi klien dari logika aplikasi dan pengelolaan data di sisi server. Pada pola tiga lapis, antarmuka pengguna berjalan di browser, server aplikasi menangani aturan bisnis dan koordinasi proses, sedangkan server basis data menyimpan dan mengambil data secara terpusat. Pemisahan ini memudahkan pemeliharaan, penskalaan, dan pengamanan komponen sesuai perannya masing masing [14].

Perkembangan teknologi web mendorong penggunaan Application Programming Interface berbasis HTTP yang memungkinkan berbagai klien berinteraksi dengan server yang sama, termasuk aplikasi mobile dan sistem pihak ketiga. Pertukaran data biasanya menggunakan format ringan seperti JavaScript Object Notation sehingga mudah diolah dan diintegrasikan. Dengan arsitektur ini, validasi input, kontrol akses, dan mekanisme pencatatan aktivitas dapat ditempatkan pada lapisan server sehingga perilaku aplikasi lebih konsisten dan mudah diuji [16], [18].

2.2.6 Otomasi Workflow atau Proses Bisnis

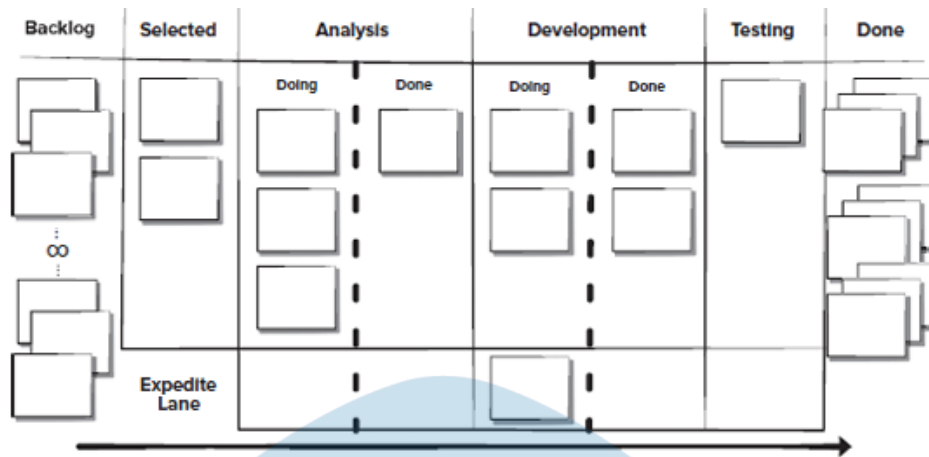
Otomasi workflow atau proses bisnis merujuk pada penggunaan sistem informasi untuk menjalankan rangkaian aktivitas kerja sesuai aturan yang telah dimodelkan sehingga perpindahan tugas dari satu peran ke peran lain terjadi secara terkontrol dan dapat dipantau. Setiap langkah dalam proses diwakili oleh status, formulir, atau tugas yang harus diselesaikan. Sistem mengatur pemicu perpindahan status, misalnya setelah dokumen disetujui maka tugas otomatis berpindah ke unit berikutnya tanpa perlu instruksi manual [1], [15].

Otomasi workflow yang baik memerlukan pemodelan proses yang jelas, penentuan peran dan hak akses, serta mekanisme eskalasi ketika terjadi hambatan. Pendekatan ini membantu mengurangi ketergantungan pada individu tertentu, meminimalkan risiko pekerjaan terlewat, dan menyediakan data historis untuk evaluasi kinerja. Integrasi workflow dengan basis data terpusat memungkinkan organisasi menganalisis waktu siklus, beban kerja, dan titik kemacetan untuk perbaikan berkelanjutan [15], [19].

Penelitian ini menerapkan otomasi workflow pada proses pelaporan teknisi mulai dari pencatatan pekerjaan di lapangan, pengunggahan dokumentasi, validasi oleh admin, hingga penerbitan laporan dan Berita Acara Serah Terima. Setiap tahap direpresentasikan sebagai status yang tercatat di sistem, sedangkan aturan perpindahan status diterapkan melalui logika aplikasi. Integrasi dengan konsep Kanban dalam pengelolaan pekerjaan pengembangan membantu memastikan bahwa perubahan pada proses bisnis tidak hanya terdokumentasi, tetapi juga dapat diimplementasikan secara iteratif dan terkontrol [2], [19].

2.2.7 Kanban

Kanban merupakan metode pengelolaan pekerjaan yang menekankan visualisasi alur kerja, pembatasan pekerjaan yang sedang berlangsung (*Work-In-Progress/WIP limit*), dan mekanisme *pull* agar beban tim tetap seimbang serta waktu tunggu dapat ditekan. Dalam praktik pengembangan sistem, Kanban *Board* dipecah menjadi kolom-kolom status, setiap kartu merepresentasikan unit kerja yang bergerak lintas kolom ketika kriteria selesai terpenuhi. Dengan demikian, tim memperoleh transparansi progres secara *real-time*, meminimalkan multitugas berlebih, dan lebih mudah mengidentifikasi kemacetan proses [6]. Visualisasi Kanban *Board* dapat dilihat pada gambar 2.1.



Gambar 2.1 Kanban Board [21]

1. Backlog

Backlog merupakan kumpulan pekerjaan potensial yang telah diidentifikasi dan belum diputuskan kapan akan dikerjakan. Isi *Backlog* dapat berupa kebutuhan baru, perbaikan, peningkatan kualitas, maupun ide awal yang masih memerlukan penajaman. Pada tahap ini deskripsi pekerjaan tidak harus rinci, yang utama adalah masalah dan tujuan utama sudah tercatat sehingga tidak hilang. *Backlog* perlu ditinjau secara berkala untuk mengelompokkan item yang serupa, memecah pekerjaan yang terlalu luas, serta menghapus pekerjaan yang sudah tidak relevan. Suatu item dapat dipindahkan dari *Backlog* apabila telah memiliki prioritas yang jelas dan ruang lingkup yang cukup untuk diproses pada tahap berikutnya.

2. Selected

Selected adalah antrian pendek yang berisi pekerjaan yang telah diprioritaskan dan disepakati sebagai fokus jangka pendek tim. Perpindahan dari *Backlog* ke selected dilakukan setelah mempertimbangkan urgensi, dampak terhadap tujuan organisasi, serta ketergantungan dengan pekerjaan lain. Tahap ini mengikuti prinsip *pull-based*, yaitu kartu hanya berpindah ke selected ketika kapasitas di tahap analisis dan pengembangan tersedia. Jumlah pekerjaan di selected dijaga terbatas agar perhatian tim tidak terpecah. Cara ini akan membuat aliran kerja menjadi lebih stabil dan risiko penumpukan pekerjaan di tengah proses dapat dikurangi.

3. Analysis – Doing

Analysis–Doing menggambarkan tahap ketika suatu pekerjaan sedang dianalisis secara aktif. Tujuan utama tahap ini adalah mengubah kebutuhan yang masih umum menjadi spesifikasi yang operasional. Kegiatan di dalamnya meliputi klarifikasi tujuan, penentuan batasan ruang lingkup, perumusan kriteria penerimaan, identifikasi risiko dan dependensi, serta penyusunan gambaran awal skenario uji. Hasil dari analisis tidak hanya menjelaskan apa yang harus dibangun, tetapi juga bagaimana cara menilai bahwa pekerjaan tersebut telah berhasil. Pendekatan ini membantu mengurangi salah tafsir antara pemangku kepentingan dan tim pelaksana serta menekan kebutuhan perubahan di tahap implementasi.

4. *Analysis – Done*

Analysis–Done merupakan penanda bahwa suatu pekerjaan telah selesai dianalisis dan memenuhi kriteria siap dikerjakan. Pada tahap ini, cakupan pekerjaan sudah terdefinisi dengan jelas, kriteria penerimaan telah disepakati, dan tidak ada isu besar yang belum terjawab. Keberadaan tahap ini berfungsi sebagai penyangga antara analisis dan pengembangan. Tim pengembang dapat menarik pekerjaan kapan saja dari *Analysis–Done* tanpa perlu menunggu klarifikasi tambahan.

5. *Development – Doing*

Development–Doing merupakan tahap realisasi solusi berdasarkan spesifikasi yang telah disusun pada fase analisis. Pada tahap ini dilakukan kegiatan implementasi seperti penulisan kode, perancangan konfigurasi, penyesuaian basis data, atau pekerjaan teknis lain sesuai konteks organisasi. Pembatasan jumlah pekerjaan yang sedang berjalan diterapkan agar setiap anggota tim dapat memfokuskan energi pada penyelesaian tugas, bukan pada penyebaran usaha ke banyak pekerjaan sekaligus.

6. *Development – Done*

Development–Done menandai bahwa kegiatan implementasi terhadap suatu pekerjaan telah selesai dan stabil dari sudut pandang tim pengembang. Solusi sudah terintegrasi dengan komponen terkait dan telah melewati pemeriksaan internal yang diperlukan. Pada titik ini, artefak pendukung

seperti catatan perubahan, dokumentasi singkat teknis, atau paket build disiapkan agar tahap pengujian dapat berlangsung tanpa hambatan.

7. *Testing*

Testing adalah tahap verifikasi yang bertujuan memastikan bahwa hasil kerja telah memenuhi kriteria penerimaan serta tidak menimbulkan cacat yang mengganggu. Pengujian dapat mencakup pengujian fungsional dari sudut pandang pengguna, pengecekan alur proses, serta pemeriksaan dasar terhadap kinerja dan keandalan. Temuan selama pengujian harus dicatat dengan jelas dan dikomunikasikan kembali ke tim pengembang untuk ditindaklanjuti. Suatu pekerjaan dinyatakan lulus pengujian apabila seluruh skenario yang ditetapkan terpenuhi dan tidak ada masalah terbuka yang berdampak signifikan. Dengan demikian, tahap *Testing* berfungsi sebagai mekanisme pengendalian mutu sebelum hasil kerja dilepas ke lingkungan penggunaan yang lebih luas.

8. *Done*

Done merepresentasikan keadaan akhir ketika suatu pekerjaan telah menyelesaikan seluruh rangkaian proses, mulai dari perencanaan hingga pengujian, dan dinilai layak untuk digunakan atau diarsipkan. Pada tahap ini dilakukan aktivitas penutup seperti pembaruan dokumentasi, pencatatan hasil akhir, dan komunikasi singkat kepada pemangku kepentingan mengenai perubahan yang telah dilakukan. Status *Done* tidak hanya berfungsi sebagai tanda selesai, tetapi juga sebagai jejak historis yang menunjukkan bagaimana sebuah pekerjaan bergerak melalui sistem. Informasi ini dapat dimanfaatkan untuk evaluasi kinerja, analisis waktu siklus, maupun perbaikan proses di masa mendatang.

9. *Expedite Lane*

Expedite Lane adalah jalur khusus bagi pekerjaan yang dikategorikan sangat mendesak dan berdampak tinggi terhadap organisasi, sehingga perlu diproses dengan prioritas di atas aliran reguler. Penggunaan jalur ini diatur oleh kebijakan yang ketat. Misalnya pembatasan jumlah pekerjaan yang boleh berjalan sekaligus serta kebutuhan persetujuan dari penanggung jawab tertentu. Sebelum suatu pekerjaan dimasukkan ke *Expedite Lane*,

perlu dilakukan peninjauan singkat untuk memastikan bahwa permintaan tersebut benar benar bersifat kritis, bukan akibat perencanaan yang kurang baik. Pendekatan ini menjaga keseimbangan antara kemampuan merespons keadaan darurat dan stabilitas sistem kerja secara keseluruhan, serta mencegah terbentuknya budaya “selalu darurat” yang dapat merusak disiplin prioritas.

Dalam penelitian ini, Kanban diposisikan sebagai metode pengelolaan pekerjaan selama proses pengembangan sistem, bukan sebagai fitur yang disediakan bagi pengguna akhir pada aplikasi pelaporan teknis. Pemilihan Kanban didasarkan pada karakteristik proyek yang menuntut respons cepat terhadap perubahan kebutuhan dari manajemen, admin, dan teknisi selama proses pengembangan. Selain itu, penelitian ini dikerjakan oleh tim kecil dengan durasi pengerjaan yang terbatas sehingga diperlukan metode yang ringan, namun tetap menjaga keterlacakan pekerjaan melalui visualisasi alur kerja dan pembatasan Work In Progress (WIP) untuk mencegah penumpukan tugas. Kanban juga sesuai karena pengembangan dilakukan per fitur secara bertahap dan kontinyu, sehingga prioritas dapat ditarik berdasarkan kapasitas tim dan urgensi bisnis, misalnya kebutuhan percepatan penyusunan laporan dan Berita Acara Serah Terima (BAST). Kanban Board digunakan oleh tim pengembang untuk memvisualisasikan Backlog, mengatur prioritas, dan memantau progres pekerjaan hingga rilis. Antarmuka yang dihadapi teknisi dan admin berfokus pada formulir input data, pengelolaan dokumentasi lapangan, serta penyusunan laporan tanpa menyajikan papan Kanban di dalam sistem operasional. Penjelasan ini sekaligus memperjelas bahwa Kanban adalah pendekatan manajemen kerja pada tim pengembang, bukan modul yang diakses mitra atau teknisi. Tabel perbandingan metode pengembangan dapat dilihat pada tabel 2.2.

Tabel 2.2 Perbandingan Metode Pengembangan

Aspek Pembanding	Kanban	Waterfall	RAD	Agile Scrum
Karakteristik	• Visualisasi alur kerja (board/kartu) • Fokus pada flow & WIP limit • Cocok untuk kerja kontinu	• Fase berurutan & kaku • Mulai dari analisis → desain → coding → testing • Cocok jika kebutuhan stabil	• Prototipe cepat • Iteratif & fokus ke kecepatan • Cocok untuk proyek kecil/menengah	• Sprint time-boxed • Ada peran: PO, SM, Dev Team • Iteratif & inkremental
Alur Kerja dan Dokumentasi	• Kolom: To Do – In Progress – Done • Dokumentasi ringan	• Dokumentasi lengkap di tiap fase • Sangat terstruktur	• Dokumentasi minim di awal • Fokus ke prototipe	• Artefak: Product Backlog, Sprint Backlog, Increment • Dokumentasi bertahap
Keterlibatan Pengguna	Terlibat lewat prioritas tugas	Terlibat di awal & akhir saja	Sangat aktif via prototipe	Aktif lewat Product Owner & Sprint Review
Penanganan Perubahan Kebutuhan	Sangat fleksibel	Sulit berubah di tengah jalan	Bisa berubah lewat iterasi	Perubahan masuk sprint berikutnya

Aspek Pembandingan	Kanban	Waterfall	RAD	Agile Scrum
Kelebihan	Visual, fleksibel, mudah diadopsi	Jelas, rapi, cocok untuk audit	Cepat, feedback awal	Terstruktur, adaptif, kolaboratif
Kekurangan	Butuh disiplin tinggi	Tidak adaptif	Bergantung pada keterlibatan user	Berat untuk tim sangat kecil

2.2.8 User Acceptance Testing

User Acceptance Testing (UAT) merupakan tahap krusial dalam siklus pengembangan sistem informasi yang bertujuan memastikan sistem memenuhi kebutuhan bisnis dan pengguna akhir sebelum implementasi penuh. Menurut Hermansah dkk. UAT didefinisikan sebagai pengujian oleh end-user atau pemilik proses untuk memverifikasi bahwa aplikasi beroperasi sesuai spesifikasi fungsional dan skenario dunia nyata, berbeda dari pengujian teknis sebelumnya. Dalam konteks Business Process Reengineering (BPR) sistem pembayaran digital seperti integrasi Midtrans, UAT memvalidasi alur transaksi end-to-end termasuk error handling, keamanan, dan kepatuhan regulasi keuangan [22].

Pelaksanaan UAT mengikuti siklus terstruktur untuk hasil optimal:

1. Perencanaan: Identifikasi scope, tujuan bisnis, tim penguji (5-10 end-user representatif), jadwal, dan kriteria sukses/gagal.
2. Persiapan: Buat test case berdasarkan use case (end-to-end, bukan isolated), data realistis (anonim produksi), environment staging mirip live, serta dokumen (script, defect log template).
3. Pelatihan Penguji: Briefing user tentang proses, akses, dan ekspektasi; pilih yang paham proses bisnis saat ini.
4. Eksekusi Pengujian: User jalankan skenario (2-4 jam/blok), catat hasil (pass/fail, bug, usability issue), check-in harian; gunakan data kontinu antar use case.
5. Pelaporan dan Perbaikan: Masukkan hasil ke matriks tracking, prioritaskan defect (severity), developer fix, retest siklus hingga threshold (e.g., 95% pass rate).
6. Sign-off dan Transisi: Stakeholder tandatangani dokumen approval formal; rencanakan training dan rollout.