

BAB II

TINJAUAN PUSTAKA

2.1 Teori Terkait dan Penelitian Sebelumnya (Research Gap)

Penelitian mengenai sistem informasi manajemen proyek dan pemantauan *lead time* telah banyak dikaji oleh peneliti terdahulu dalam berbagai *cluster* industri. Analisis terhadap penelitian-penelitian serupa sangat penting dilakukan guna merancang metodologi yang tepat, menemukan celah penelitian (*research gap*), serta menghindari unsur plagiarisme dalam rekayasa perangkat lunak.

Sebuah penelitian mengenai analisis sistem informasi manajemen proyek berbasis tinjauan literatur sistematis membuktikan bahwa pemanfaatan teknologi berbasis web mampu meningkatkan transparansi, struktur organisasi, dan efisiensi pengambilan keputusan taktis secara signifikan dibandingkan penggunaan pelaporan konvensional. Dalam konteks visualisasi data, penelitian mengenai perancangan sistem pemantauan operasional menegaskan bahwa visualisasi status berbasis indikator warna terbukti mampu mempercepat respon manajemen terhadap kendala di lapangan karena mempermudah pelacakan aset secara *real-time*. Sementara itu, dalam aspek pengendalian waktu, penelitian mengenai tinjauan penjadwalan menekankan bahwa penggunaan *time schedule* yang terstruktur sangat krusial dalam mengorganisasi kegiatan proyek, memitigasi

risiko pembengkakan durasi, serta mengidentifikasi area kerja yang membutuhkan intervensi dini [1].

Meskipun demikian, berdasarkan kajian dari berbagai penelitian sebelumnya tersebut, terdapat celah penelitian (*research gap*) yang belum banyak dieksplorasi secara komprehensif, di antaranya belum adanya otomatisasi pembacaan berkas Excel secara massal yang terintegrasi langsung dengan penghentian kalkulasi otomatis pada status final, serta kebutuhan khusus untuk mengawasi operasional mitra kerja di lingkungan PT Telkom Akses. Perbedaan fundamental antara penelitian-penelitian terdahulu dengan penelitian rancang bangun yang dilakukan saat ini terletak pada objek industri, arsitektur sistem, dan kedalaman integrasi fiturnya. Sistem DMS (*Data Management System*) yang dirancang dalam skripsi ini dikembangkan menggunakan arsitektur monolitik ringan berbasis *framework* CodeIgniter 3 dan Tailwind CSS, di mana komunikasi data antarmuka dilakukan secara sinkron menggunakan AJAX untuk mengoptimalkan performa pemrosesan masal (*bulk upload*) berkas Excel. Selain itu, sistem ini tidak sekadar menampilkan grafik statis, melainkan menggabungkan modul analisis keuangan (akumulasi nilai BOQ berdasarkan status proyek), pelacakan performa vendor secara dinamis, penghentian kalkulasi otomatis *lead time* saat kondisi final status, hingga pengawasan aspek keamanan data yang lebih ketat melalui mekanisme *Native PHP Session Timeout* otomatis selama 3 jam dan perekaman *Audit Trail* yang terperinci ke dalam basis data MySQL.

2.2 Landasan Teori

2.2.1 Sistem Manajemen dan Rancang Bangun

Rancang bangun sistem manajemen merupakan proses menerjemahkan kebutuhan bisnis dan fungsional ke dalam representasi perangkat lunak yang siap dieksekusi. Sistem informasi manajemen berfungsi mengumpulkan, memproses, menyimpan, dan menganalisis informasi untuk mendukung proses pengambilan keputusan taktis di dalam organisasi. Aktivitas rancang bangun ini melibatkan berbagai tahapan terstruktur, mulai dari analisis kebutuhan pengguna (*requirements gathering*), pemodelan sistem menggunakan diagram alir data, hingga perancangan arsitektur basis data yang kokoh. Melalui pendekatan rancang bangun yang sistematis, sebuah platform digital dapat dikembangkan secara presisi untuk memitigasi kesalahan operasional, mengotomatiskan tugas-tugas repetitif, serta menyediakan struktur pelaporan data yang konsisten dan terintegrasi di seluruh lini manajemen perusahaan [2].

Dalam konteks manajemen modern, kehadiran Sistem Informasi Manajemen (SIM) yang diimplementasikan melalui DMS (*Data Management System*) bertindak sebagai infrastruktur krusial yang menjembatani kesenjangan komunikasi operasional antara level staf dan pihak manajemen perusahaan. Aplikasi DMS ini tidak sekadar berfungsi sebagai tempat penyimpanan data digital pasif, melainkan sebagai mesin pengolah data mentah transaksional menjadi informasi

bernilai strategis yang mendukung fungsi perencanaan, pengorganisasian, dan pengawasan proyek konstruksi. Melalui pengolahan data yang dinamis dan terpusat, DMS mampu meminimalkan risiko redundansi data, mempercepat proses distribusi laporan operasional, serta meningkatkan akuntabilitas kinerja setiap elemen organisasi. Dengan demikian, penerapan sistem DMS ini secara langsung berkontribusi pada peningkatan efisiensi biaya operasional, reduksi kesalahan manusia (*human error*), serta penguatan daya saing organisasi dalam menghadapi dinamika industri yang kompetitif [3].

2.2.2 Monitoring Lead Time dan Konsep Visual Management

Lead time adalah total waktu yang dihabiskan sejak sebuah pesanan atau pekerjaan dibuat hingga pekerjaan tersebut dinyatakan selesai sepenuhnya. Dalam Visual Management, penggunaan warna (Hijau, Kuning, Merah) bertujuan untuk memberikan stimulus kognitif instan kepada pembaca agar dapat langsung membedakan antara kondisi normal, kondisi waspada, dan kondisi kritis (*overdue*) tanpa harus membaca data teks secara mendalam. Pemantauan metrik ini secara terus menerus sangat penting untuk mengidentifikasi adanya hambatan atau pemborosan waktu (*bottleneck*) dalam rantai pasok maupun siklus pelaksanaan proyek. Melalui visualisasi data yang adaptif, manajemen dapat dengan cepat menangkap anomali

durasi kerja operasional, mengevaluasi efektivitas alur kerja, serta melakukan intervensi korektif secara dini sebelum keterlambatan tersebut berdampak luas pada kepuasan pelanggan ataupun membengkaknya anggaran biaya [4].

Penerapan konsep Visual Management di dalam sistem monitoring digital mentransformasikan data angka yang rumit menjadi representasi grafis yang intuitif dan mudah dipahami oleh seluruh tingkatan pengguna. Pendekatan berbasis indikator visual ini memanfaatkan kecenderungan alami manusia yang lebih cepat merespons simbol visual dan warna dibandingkan baris teks naratif yang padat. Ketika suatu pekerjaan beralih status warna dari hijau menjadi kuning atau merah, sistem secara otomatis memberikan sinyal urgensi yang memaksa penanggung jawab untuk segera mengambil tindakan responsif. Selain meningkatkan transparansi kinerja di lapangan, metodologi manajemen visual ini juga mempererat koordinasi antar tim, menyederhanakan proses peninjauan berkala oleh pihak eksekutif, serta menumbuhkan budaya kerja yang proaktif dan berorientasi pada ketepatan waktu.

2.2.3 Arsitektur Perangkat Lunak Monolitik Dinamis

1. CodeIgniter 3 (CI3)

CodeIgniter 3 merupakan sebuah kerangka kerja (*framework*) pengembangan aplikasi web berbasis PHP yang mengadopsi pola

arsitektur Model-View-Controller (MVC). Melalui struktur MVC yang tertata, CodeIgniter secara tegas memisahkan antara logika manipulasi data (*Model*), logika bisnis aplikasi (*Controller*), dan tampilan antarmuka (*View*). Keunggulan utama dari kerangka kerja ini terletak pada performanya yang cepat (*lightweight*), konfigurasi yang minimal, serta dokumentasi yang matang. Dalam pengembangan aplikasi DMS, CodeIgniter 3 bertindak di sisi server (*backend server*) yang bertanggung jawab penuh dalam mengelola sesi pengguna, menyediakan gerbang validasi data, serta menangani seluruh logika pemrosesan kalkulasi matematika operasional proyek [4].

2. Tailwind CSS

Tailwind CSS adalah sebuah kerangka kerja CSS modern yang mengusung pendekatan *utility-first*. Berbeda dengan kerangka kerja konvensional yang menyediakan komponen siap pakai, Tailwind CSS menyediakan kumpulan kelas utilitas tingkat rendah (*low-level utility classes*) seperti pengaturan tata letak, warna, tipografi, dan responsivitas yang dapat disematkan langsung ke dalam kode HTML. Pendekatan ini memungkinkan pengembang untuk membangun antarmuka dasbor (*dashboard interface*) yang sangat kustom, modern, dan adaptif tanpa menghasilkan ukuran berkas CSS yang membengkak. Penggunaan Tailwind CSS pada aplikasi DMS menjamin

visualisasi data logistik dan finansial proyek dapat tersaji secara rapi dan responsif di berbagai ukuran perangkat [5].

3. AJAX (Asynchronous JavaScript and XML)

Asynchronous JavaScript and XML (AJAX) merupakan teknik rekayasa web yang memungkinkan aplikasi untuk mengirim dan menerima data dari peladen secara asinkron (di latar belakang) tanpa harus memuat ulang keseluruhan halaman web (*page refresh*). Dengan memanfaatkan objek XMLHttpRequest atau pustaka berbasis JavaScript, AJAX menjembatani komunikasi instan antara antarmuka pengguna (*View*) dan pengontrol (*Controller*) pada CodeIgniter 3. Implementasi AJAX dalam aplikasi DMS sangat krusial untuk menciptakan pengalaman pengguna yang interaktif, terutama pada proses pengunggahan massal (*bulk upload*) berkas Excel, pembaruan status proyek secara seketika, serta kalkulasi visual *lead time order* secara dinamis [6].

2.2.4 Keamanan Sistem, RBAC, dan Audit Trail

1. Role-Based Access Control (RBAC)

Role-Based Access Control (RBAC) merupakan pendekatan keamanan basis data dan pembatasan hak akses sistem kepada pengguna yang sah berdasarkan peran (*role*) atau jabatan mereka di dalam organisasi. Melalui penerapan konsep RBAC pada

aplikasi DMS, pengguna yang masuk ke dalam sistem diklasifikasikan secara ketat ke dalam dua peran utama, yaitu Admin dan Staf. Setiap peran hanya diberikan wewenang khusus untuk menjalankan fungsi-fungsi spesifik yang relevan dengan deskripsi pekerjaan mereka. Mekanisme ini secara efektif mengeliminasi risiko terjadinya tumpang tindih fungsi kerja, memitigasi potensi kebocoran data sensitif proyek kepada pihak yang tidak berwenang, serta mempermudah pengelolaan kredensial pengguna secara terpusat oleh administrator [7].

2. Audit Trail (Log Audit)

Audit Trail atau log audit merupakan catatan kronologis berbasis waktu yang direkam sistem untuk menangkap setiap aktivitas manipulasi data oleh pengguna (mencakup informasi tentang siapa, aktivitas apa, waktu eksekusi, serta baris data proyek terkait). Keberadaan rekaman ini memegang peranan vital sebagai instrumen transparansi, akuntabilitas, dan pengawasan internal terhadap operasi data. Jika di kemudian hari ditemukan kesalahan atau tindakan yang tidak sah, log audit ini akan bertindak sebagai bukti forensik digital yang akurat untuk melacak akar permasalahan sekaligus menjaga integritas data proyek konstruksi.

2.2.5 Teori Pembuatan dan Pengujian Sistem

1. Tahapan Pembuatan Perangkat Lunak

Pembuatan perangkat lunak atau rekayasa sistem dilakukan melalui pendekatan terstruktur agar aplikasi yang dihasilkan sesuai dengan kebutuhan fungsional dunia nyata. Proses ini mencakup identifikasi kebutuhan pengguna, perancangan basis data, implementasi kode program, hingga integrasi fitur-fitur utama seperti pengolahan data Excel dan sistem keamanan.

2. Black-Box Testing (Pengujian Kotak Hitam)

Black-box testing adalah metode pengujian fungsionalitas perangkat lunak dari sudut pandang pengguna luar tanpa harus menguji kode pemrograman secara internal. Pengujian ini berfokus pada kesesuaian antara data masukan (*input*) dengan keluaran (*output*) sistem, seperti memastikan fitur unggah berkas Excel, tombol navigasi dasbor, pembaruan status proyek, serta pembatasan hak akses berjalan dengan benar sesuai fungsinya.

3. User Acceptance Testing (UAT)

User Acceptance Testing (UAT) merupakan tahap pengujian akhir di mana sistem dioperasikan langsung oleh calon pengguna atau pihak manajemen terkait untuk memastikan bahwa aplikasi yang dibangun telah memenuhi seluruh kebutuhan operasional dan layak untuk digunakan secara nyata di dalam perusahaan.