

BAB II

LANDASAN TEORI

2.1 Konsep Dasar Sistem

Sistem dapat ditemukan dalam berbagai aspek kehidupan, termasuk dalam tubuh manusia. Contohnya, sistem pencernaan berperan dalam memproses makanan dan minuman yang kita konsumsi. Pada dasarnya, sistem adalah suatu kesatuan yang terdiri atas berbagai komponen berbeda. Meskipun setiap komponen memiliki fungsi khusus, mereka saling berhubungan dan bekerja sama untuk mencapai tujuan bersama.

Konsep dasar sistem menekankan bahwa suatu sistem bukan hanya sekadar kumpulan bagian, melainkan suatu kesatuan yang terorganisir di mana antar-komponen saling bergantung. Dalam konteks ini, input yang masuk akan diproses melalui interaksi antar-komponen, kemudian menghasilkan output yang diharapkan (Soufitri, 2023). Prinsip ini berlaku universal, mulai dari sistem biologis seperti peredaran darah, hingga sistem buatan manusia seperti jaringan komputer atau tata kelola organisasi. Pemahaman terhadap konsep ini menjadi fondasi penting untuk menganalisis fungsi, efisiensi, serta hubungan antara sistem dengan lingkungan sekitarnya.

2.1.1 Analisis Sistem

Analisis sistem merupakan suatu proses evaluasi yang komprehensif terhadap sistem yang sedang berjalan. Tujuannya bukan hanya untuk mengidentifikasi kelemahan dan masalah, tetapi juga untuk menilai kelayakan pengembangan sistem baru serta merumuskan kebutuhan yang harus dipenuhi.

Menurut (Soufitri, 2023), analisis sistem adalah proses menganalisis, mempelajari, dan mengevaluasi sistem yang telah ada untuk mengidentifikasi permasalahan, kelayakan, dan kebutuhan sehingga dapat dirancang sistem baru yang lebih efektif dan efisien.

2.1.2 Sistem Informasi

Sistem Informasi merupakan suatu entitas terintegrasi yang tersusun atas berbagai elemen saling terkait. Elemen-elemen ini bekerja sama melakukan pengolahan data guna menghasilkan informasi yang bernilai. Komponen pembentuknya mencakup perangkat keras (*hardware*), perangkat lunak (*software*), data, prosedur kerja, serta sumber daya manusia (*brainware*). Informasi yang diperoleh kemudian berperan sebagai landasan untuk proses pengambilan keputusan, pelaksanaan koordinasi, dan aktivitas pengendalian di dalam suatu organisasi (Soufitri, 2023).

2.2 Pengertian Rancang Bangun

Rancang bangun merupakan suatu proses sistematis dalam pengembangan sistem yang diawali dari tahap analisis kebutuhan, kemudian diterjemahkan ke dalam bentuk perancangan teknis dan direalisasikan menjadi paket perangkat lunak yang dapat diimplementasikan. Proses ini tidak hanya mencakup pembuatan sistem baru, tetapi juga dapat berupa pengembangan atau penyempurnaan terhadap sistem yang telah ada agar sesuai dengan kebutuhan pengguna dan tujuan organisasi (Sitanggang, Dachi, & Manurung, 2022). Tahapan ini melibatkan proses penerjemahan kebutuhan pengguna menjadi model atau rancangan sistem yang

dapat diimplementasikan, seperti diagram alur data, perancangan basis data, serta rancangan antarmuka pengguna. Dengan demikian, rancang bangun tidak hanya berfokus pada pembuatan program, tetapi juga pada bagaimana sistem tersebut dapat bekerja secara efektif dan efisien sesuai dengan kebutuhan yang telah dianalisis sebelumnya.

2.3 Metode Pendukung

Metode pendukung merupakan pendekatan yang digunakan untuk menunjang proses pengembangan dan implementasi sistem agar sesuai dengan kebutuhan operasional. Pada penelitian ini, metode pendukung yang digunakan meliputi Rapid Application Development (RAD) sebagai pendekatan pengembangan sistem serta Routing-Based Work In Process (WIP) Tracking sebagai pendekatan dalam pengelolaan dan pelacakan alur proses produksi.

2.3.1 Rapid Application Development (RAD)

Rapid Application Development (RAD) merupakan pendekatan pengembangan perangkat lunak yang menekankan kecepatan pengembangan sistem melalui proses iteratif, penggunaan prototipe, serta keterlibatan pengguna secara intensif. RAD dirancang untuk mengurangi waktu pengembangan dengan cara memecah sistem ke dalam modul-modul fungsional yang dapat dikembangkan dan diuji secara paralel (Daru, Adhiwibowo, & Anggara, 2021).



Gambar 2. 1 Siklus Rapid Application Development (RAD)

Sumber : (Hendri, Virgo, Komarudin, Afni, & Maulana, 2024)

2.3.2 Routing-Based WIP Tracking

Routing-Based WIP Tracking adalah metodologi pelacakan yang memanfaatkan struktur alur produksi yang telah ditetapkan (*fixed routing*) sebagai kerangka referensi utama untuk memantau pergerakan dan status setiap unit produksi (Springer, 2021).

Berdasarkan struktur rute tetap tersebut, setiap unit produksi dapat dilacak pergerakannya secara real-time dengan mencocokkan lokasi aktualnya terhadap tahapan yang telah ditentukan dalam routing. Sistem ini biasanya mengandalkan identifikasi unik, sehingga menciptakan jejak audit digital yang akurat. Akibatnya, manajemen tidak hanya mengetahui di mana suatu item berada, tetapi juga dapat menganalisis waktu tinggal (*dwell time*), mengidentifikasi kemacetan (*bottleneck*), dan memastikan kepatuhan terhadap urutan operasi yang telah direncanakan. Dengan demikian, metodologi ini mentransformasi routing dari sekadar dokumen prosedural menjadi kerangka dinamis untuk visibilitas, kontrol, dan pengambilan keputusan yang lebih baik di rantai produksi.

2.4 Aplikasi Web

Aplikasi web adalah pengembangan lebih lanjut dari website statis, dimana pengguna dapat berinteraksi secara dinamis dengan sistem. Berbeda dengan website konvensional yang hanya menampilkan informasi, aplikasi web memungkinkan pertukaran data dua arah antara pengguna dan server, serta dapat memproses input dari pengguna untuk menghasilkan output yang sesuai.

Menurut Muhammad Ibnu Sa'ad (2023), website atau aplikasi web merupakan kumpulan halaman yang saling terhubung dan dapat diakses melalui internet, yang memungkinkan interaksi dua arah antara pengguna dengan sistem. Aplikasi web modern tidak hanya menampilkan informasi statis, tetapi juga dapat memproses data input dari pengguna dan menghasilkan output yang dinamis sesuai kebutuhan.

2.5 Teknologi Pendukung

2.5.1 Laravel Framework

Laravel merupakan salah satu framework PHP yang menggunakan pendekatan Model-View-Controller dalam pengembangan aplikasi web dan menawarkan fitur-fitur yang dapat mendukung pengembangan aplikasi web seperti, sintaks yang mudah dipahami sehingga mempercepat proses pembuatan aplikasi berbasis web serta modul-modul tambahan sebagai alat bantu (Susanto, Widyandayani, & Pramono, 2024).

Penggunaan Laravel sebagai basis perancangan sistem informasi pada penelitian ini bertujuan untuk memudahkan dalam proses perancangan sistem karena Laravel memiliki stuktur yang mudah dipahami, sistematis, juga performa

yang baik untuk pelacakan proses produksi karena memiliki banyak fitur dan *packages-packages* gratis yang dapat mendukung penelitian ini.

2.5.2 Laragon

Laragon adalah sebuah *local development environment* yang menyediakan paket terintegrasi dari komponen-komponen perangkat lunak yang dibutuhkan untuk membangun, menjalankan, dan menguji aplikasi web secara lokal pada komputer pengembang. Laragon menggabungkan web server (seperti Apache atau Nginx), bahasa pemrograman server-side (seperti PHP), dan sistem basis data (seperti MySQL/MariaDB) dalam satu lingkungan kerja yang telah dikonfigurasi sebelumnya sehingga pengembang tidak perlu melakukan *setup* server manual yang kompleks (Sommerfeld, 2024).

Laragon dirancang dengan filosofi yang ringan, cepat, dan mudah digunakan sehingga ideal untuk mendukung proses *development* dan *testing* sebelum aplikasi di-*deploy* ke lingkungan produksi. Lingkungan kerja ini mendukung berbagai *framework* dan aplikasi web populer seperti Laravel, CodeIgniter, WordPress, dan juga mendukung ekosistem Node.js, sehingga memberikan fleksibilitas bagi pengembang dalam bekerja dengan berbagai teknologi.

2.5.3 MySQL

MySQL adalah sistem manajemen basis data relasional (*Relational Database Management System / RDBMS*) yang bersifat *open-source* dan menggunakan bahasa kueri terstruktur (*Structured Query Language / SQL*) untuk

mengelola data. MySQL dikembangkan oleh MySQL AB (sekarang dimiliki oleh Oracle Corporation) dan menjadi salah satu DBMS paling populer karena kehandalan, kecepatan, dan kemudahan penggunaannya. MySQL mendukung arsitektur *client-server*, di mana beberapa *client* dapat mengakses dan memanipulasi data yang tersimpan di dalam *server* basis data secara bersamaan.

Beberapa keunggulan MySQL antara lain adalah kemampuannya menangani basis data yang besar, mendukung transaksi dengan ACID (*Atomicity, Consistency, Isolation, Durability*), serta memiliki sistem keamanan yang baik dengan mekanisme *user privilege* dan *password encryption*. MySQL juga kompatibel dengan berbagai platform sistem operasi seperti Windows, Linux, dan macOS, serta dapat diintegrasikan dengan berbagai bahasa pemrograman seperti PHP, Python, dan Java. Selain itu, MySQL mendukung berbagai jenis *storage engine*, seperti InnoDB dan MyISAM, yang dapat dipilih sesuai kebutuhan.

Dalam rekayasa perangkat lunak, MySQL sering digunakan sebagai solusi penyimpanan data untuk aplikasi berbasis web maupun desktop. Dengan dukungan terhadap kueri kompleks, *stored procedure*, *trigger*, dan *view*, MySQL memungkinkan pengembang untuk merancang dan mengelola basis data yang terstruktur. Kombinasi MySQL dengan bahasa pemrograman seperti PHP dan framework seperti Laravel telah menjadi standar dalam pengembangan aplikasi modern.

2.6 Perangkat Analisis Sistem

2.6.1 Pengenalan UML

UML (*Unified Modeling Language*) merupakan sebuah bahasa standar dalam pemodelan yang digunakan untuk merepresentasikan struktur dan perilaku suatu sistem atau proses. Bahasa ini dikembangkan untuk menyatukan berbagai metodologi pengembangan berbasis objek yang sebelumnya berbeda. UML berfungsi sebagai alat untuk memvisualisasikan, mendefinisikan, serta mendokumentasikan rancangan sistem perangkat lunak. Dalam praktiknya, UML mencakup sejumlah diagram yang masing-masing memiliki fungsi dan tujuan spesifik sesuai kebutuhan pemodelan (Nurshadrina & Voutama, 2022).

2.6.2 Use Case Diagram

Use Case diagram adalah model UML yang digunakan untuk mendeskripsikan fungsionalitas sistem dari perspektif pengguna (aktor). Diagram ini menggambarkan interaksi antara aktor dengan sistem yang direpresentasikan sebagai fungsi atau layanan yang disediakan oleh sistem.

Use case adalah suatu alat yang digunakan untuk mendeskripsikan fungsionalitas suatu sistem dari sudut pandang pengguna akhir. Melalui *use case*, dapat ditentukan proses serta komponen-komponen apa saja yang akan ditangani oleh sistem. *Use case* dioperasionalkan dengan menggunakan skenario, yaitu deskripsi langkah demi langkah yang menggambarkan interaksi timbal balik antara pengguna dan sistem (Setiyani, 2021).

2.6.3 Skenario

Menurut Kurniawan (2020), yang menyatakan bahwa skenario *use case* adalah bagian penting dari *use case* yang menggambarkan interaksi antara aktor dan sistem dalam bentuk deskripsi kejadian, baik secara normal maupun untuk kemungkinan variasi alur. Skenario ini mencakup *main success scenario*, alur alternatif, dan kondisi pengecualian, yang semuanya diperlukan untuk memberikan pemahaman menyeluruh terhadap perilaku sistem terhadap masukan dari aktor.

2.6.4 Activity Diagram

Activity diagram merupakan salah satu jenis diagram yang digunakan dalam pemodelan sistem berbasis Unified Modeling Language (UML). Diagram ini berfungsi untuk menggambarkan alur aktivitas atau proses bisnis secara terstruktur, mulai dari awal hingga akhir suatu proses. Menurut Hamid Kurniawan dkk. (2020), “Activity diagram adalah gambaran aktifitas dari suatu proses bisnis dan urutan aktifitasnya dalam suatu proses.”.

Diagram ini berguna untuk memvisualisasikan bagaimana aktivitas-aktivitas dalam sistem berjalan dan berinteraksi, baik secara berurutan maupun paralel. Dalam pengembangan sistem informasi, activity diagram sering digunakan untuk menjelaskan proses kerja suatu fitur atau modul dari sudut pandang fungsional.