

BAB I

PENDAHULUAN

A. Latar Belakang

Website telah menjadi bagian penting dalam kehidupan sehari-hari. Saat ini, jumlah *website* terus bertambah dengan lebih dari 1,13 miliar *website* terdaftar pada tahun 2024, meskipun hanya sekitar 18% yang aktif [1]. Peran *website* semakin kuat dengan meningkatnya penetrasi *internet* global yang telah mencapai 67,1% (5,37 miliar orang), di mana 96% pengguna *internet* mengaksesnya melalui perangkat *mobile*. Selain itu, 53% lalu lintas *web* global berasal dari pencarian organik, menunjukkan pentingnya optimasi *website* dalam dunia *digital* [2].

Perkembangan teknologi, seperti adopsi jaringan 5G yang telah menjangkau 45% populasi global pada 2023 dan diproyeksikan mencapai 85% pada tahun 2029, memberikan koneksi lebih cepat dan stabil bagi layanan berbasis *web*. Kemajuan ini memungkinkan berbagai sistem berbasis *web*, termasuk sistem informasi dalam dunia bisnis, untuk bekerja lebih efisien dan diakses secara *online* [3].

Seiring dengan pesatnya perkembangan *website* dan meningkatnya akses *internet* global, sistem informasi berbasis *web* menjadi solusi yang semakin diandalkan dalam berbagai bidang, termasuk dunia bisnis. Dengan dukungan teknologi jaringan yang semakin cepat dan stabil, sistem informasi berbasis *web* memungkinkan akses *online*, integrasi yang lebih baik, serta efisiensi dalam pengolahan data dan pengambilan keputusan. Dalam dunia bisnis, sistem ini memainkan peran penting dalam mengelola transaksi, stok, laporan keuangan, dan visualisasi data pelanggan secara otomatis [4].

Meskipun manfaat sistem informasi berbasis *website* sudah banyak dirasakan, masih banyak industri kuliner yang belum memiliki sistem *digital* dan masih menggunakan pencatatan manual [5]. Hal ini menyebabkan berbagai permasalahan operasional, seperti ketidakakuratan dalam pengelolaan stok dan

transaksi, yang dapat mengakibatkan stok berlebih atau kekurangan bahan baku, menghambat operasional, serta meningkatkan risiko kerugian yang tidak tercatat [6], [7]. Tanpa sistem terintegrasi, pemilik usaha juga kesulitan memantau keuangan secara *online* dan lebih rentan terhadap kecurangan karyawan, seperti manipulasi laporan atau penyalahgunaan bahan baku, yang dapat menghambat evaluasi bisnis dan pengambilan keputusan strategis [8].

Pada objek penelitian ini, Waroeng Bebek Geseng menghadapi berbagai kendala operasional sebagaimana dijelaskan sebelumnya. Kendala utama terletak pada tidak adanya sistem pencatatan secara *online*, yang menyebabkan banyak data transaksi terlewat atau tidak terdokumentasi dengan baik. Akibatnya, terjadi ketidakseuaian dalam pengelolaan stok, keuangan, dan transaksi yang berpotensi menimbulkan kerugian bagi usaha. Tanpa sistem terintegrasi, pemilik usaha kesulitan memantau operasional secara efektif, sehingga evaluasi bisnis dan pengambilan keputusan strategis menjadi kurang optimal.

Untuk mengatasi permasalahan tersebut, maka dibuatlah sistem informasi dan transaksi berbasis *website* yang dapat mencatat data secara *online* dan meningkatkan efisiensi operasional. Sistem ini dibangun menggunakan React untuk *front-end* dan Laravel untuk *back-end*. React dipilih karena kemampuannya dalam membangun antarmuka yang dinamis dan responsif, dengan fitur *reusable components* serta *Virtual DOM* yang meningkatkan efisiensi aplikasi [9]. Sementara itu, Laravel sebagai *framework back-end* menawarkan arsitektur *Model-View-Controller (MVC)*, pengelolaan basis data yang lebih baik, serta fitur keamanan yang tinggi. Dengan struktur kode yang terorganisir dan skalabilitas yang baik [10], Laravel telah digunakan oleh lebih dari 1,2 juta situs *web* aktif di seluruh dunia, menjadikannya salah satu *framework* PHP terpopuler [11], [12]. Melalui kombinasi teknologi ini, sistem yang dibangun diharapkan mampu memberikan solusi yang terintegrasi, akurat, dan mudah diakses guna meningkatkan efisiensi pengelolaan transaksi, stok, serta laporan keuangan di Waroeng Bebek Geseng.

Sistem ini akan dilengkapi dengan *dashboard* deskriptif untuk memberikan wawasan mendalam tentang tren pelanggan, data transaksi, dan performa bisnis [13]. *Dashboard* ini memungkinkan pengelola bisnis untuk melihat laporan penjualan, stok bahan baku, serta perilaku pelanggan secara *online*, sehingga keputusan bisnis dapat dibuat berdasarkan data yang lebih akurat [14]. Implementasi *dashboard* dilakukan menggunakan pustaka *Chart.js* atau *ApexCharts* di React, yang menghasilkan visualisasi data yang interaktif dan informatif [15].

Selain meningkatkan efisiensi pencatatan transaksi dan manajemen stok, sistem ini juga memastikan integrasi yang optimal antara *front-end* dan *back-end*. Siklus permintaan dan respons digunakan untuk integrasi, di mana *front-end* mengirimkan permintaan data, *back-end* memprosesnya, lalu mengirimkan respons kembali. Dengan integrasi yang baik, pengalaman pengguna akan lebih optimal, sementara operasional tempat makan dapat berjalan lebih efisien dan transparan [16].

B. Rumusan Masalah

Pada Studi kasus penelitian, di salah satu industri kuliner diketahui segala proses pencatatan masih dilakukan secara manual dan belum ada sistem informasi dan transaksi berbasis *website* yang mampu mengelola stok, bahan baku, transaksi penjualan, dan laporan keuangan secara otomatis.

C. Batasan Masalah

1. Sistem informasi yang dikembangkan berbasis *website* dan tidak mencakup pengembangan aplikasi *mobile*.
2. Sistem hanya menangani pengelolaan transaksi penjualan, manajemen stok bahan baku, dan laporan keuangan, tanpa mencakup fitur seperti integrasi pembayaran *digital*.
3. Data yang digunakan dalam pengujian sistem berbasis studi kasus pada satu tempat makan tertentu, sehingga implementasi di bisnis lain memerlukan penyesuaian lebih lanjut.

D. Tujuan Penelitian

1. Merancang dan membangun sistem informasi dan transaksi berbasis *website* yang dapat menggantikan proses bisnis manual, meningkatkan efisiensi operasional melalui pencatatan transaksi otomatis, pengelolaan stok yang akurat, serta pelaporan keuangan yang sistematis.
2. Menyediakan fitur *dashboard* deskriptif dan *membership* yang dapat membantu pemilik usaha dalam pengambilan keputusan strategis serta membangun loyalitas pelanggan melalui informasi yang *online* dan interaktif.

E. Metode Penelitian

Penelitian ini menggunakan pendekatan *Waterfall* (lihat Gambar 1.1), di mana setiap tahapan dilakukan secara linear dan harus diselesaikan sepenuhnya sebelum melanjutkan ke tahap berikutnya. Tahapan metode ini meliputi:

1. Analisis Kebutuhan
 - Melakukan studi Pustaka dengan mengkaji penelitian sebelumnya yang relevan dengan sistem informasi, seperti yang dijelaskan pada Bab II
 - Melakukan wawancara langsung dengan pemilik Waroeng Bebek Geseng untuk mengidentifikasi masalah utama akibat proses manual dan memahami kebutuhan fungsional serta non-fungsional dari sistem yang diharapkan .
 - Menghasilkan dokumen spesifikasi kebutuhan sistem yang mencakup kebutuhan fungsional dan non-fungsional, yang menjadi acuan untuk tahap perancangan.
2. Perancangan Sistem
 - Membuat desain sistem secara mendetail, meliputi:
 - o Merancang arsitektur sistem menggunakan pola *Model-View-Controller (MVC)* yang terintegrasi melalui REST API.

Arsitektur ini memisahkan antara antarmuka (*View*), logika bisnis (*Controller*), dan pengelolaan data (*Model*).

- Membuat diagram *Use Case* untuk menggambarkan interaksi antara aktor (*Customer*, Kasir, Admin/Owner) dengan fungsionalitas sistem.
- Merancang struktur basis data menggunakan Entity-Relationship Diagram (ERD) untuk mendefinisikan entitas dan relasinya, seperti *Customer*, Transaksi, Produk, dan Karyawan.
- Perancangan antarmuka dengan membuat desain antarmuka dalam bentuk *wireframe* atau prototype *low-fidelity* dengan menggunakan Figma dan dilanjutkan dengan pengembangan prototype *high-fidelity* yang interaktif untuk pengujian awal terhadap calon pengguna.

3. Implementasi

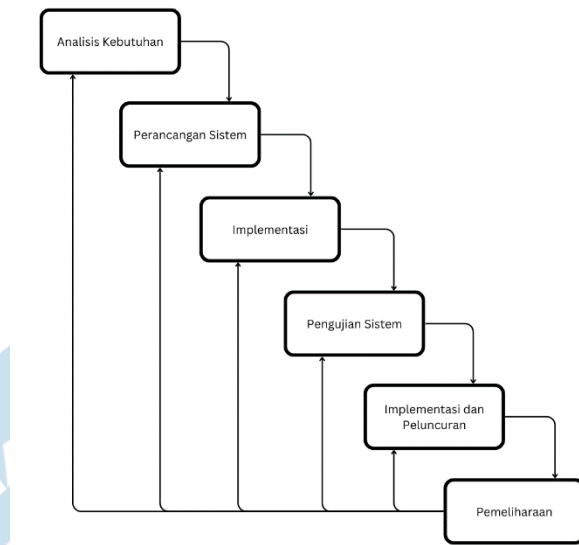
- Mengembangkan sistem berdasarkan desain yang telah dibuat. Implementasi dilakukan dalam dua bagian utama:
 - *Front-end*: Dibangun menggunakan React untuk menciptakan antarmuka yang responsif dan interaktif.
 - *Back-end*: Dibangun menggunakan Laravel untuk menangani logika aplikasi, otentikasi, dan pengelolaan database.
- Menggunakan MySQL sebagai sistem manajemen basis data (DBMS) untuk menyimpan seluruh data sistem.
- Menghubungkan *front-end* dan *back-end* menggunakan *RESTful API* untuk memastikan pertukaran data yang efisien antara sisi klien dan *server*.

4. Pengujian Sistem

- Melakukan pengujian untuk memastikan sistem berjalan sesuai dengan kebutuhan yang telah ditentukan, meliputi:

- Pengujian Prototype: Melakukan pengujian terhadap prototipe high-fidelity yang dibuat di Figma kepada 6 responden menggunakan platform Maze. Tujuannya adalah untuk mengevaluasi kemudahan penggunaan (usability) dan mendapatkan umpan balik awal sebelum implementasi penuh.
 - Pengujian Fungsionalitas (Black Box testing): Menguji setiap fungsi pada sistem yang sudah jadi untuk memastikan bahwa *output* yang dihasilkan sesuai dengan *input* yang diberikan, tanpa melihat struktur kode internalnya. Hasil pengujian ini didokumentasikan dalam tabel pengujian fungsionalitas.
5. Implementasi dan Peluncuran (Belum dijalankan)
- Mengimplementasikan sistem pada lingkungan operasional yang terbatas untuk uji coba pengguna (*User Acceptance Testing*).
 - Setelah sistem disetujui, dilakukan peluncuran secara penuh ke lingkungan produksi.
6. Pemeliharaan (Belum dijalankan)
- Memantau sistem untuk memastikan kinerja tetap optimal dan melakukan perbaikan atau pembaruan jika diperlukan.
 - Tahap ini mencakup pemeliharaan preventif, korektif, dan adaptif berdasarkan kebutuhan pengguna dan lingkungan.

Gambar 1.1 merupakan visualisasi dari metode penelitian dengan metode *Waterfall*. Pendekatan dengan metode *Waterfall* dipilih karena setiap tahapan dapat terdokumentasi dengan jelas dan rapi, memastikan bahwa kebutuhan sistem, desain, dan implementasi telah diselesaikan secara mendetail sebelum melanjutkan ke tahap berikutnya. Pendekatan ini cocok untuk proyek dengan kebutuhan yang telah jelas dan stabil di awal, seperti pengembangan sistem informasi berbasis *website*.



Gambar 1. 1 Diagram Metode *Waterfall*

F. Sistematika Penulisan

BAB I PENDAHULUAN

Bab ini menyajikan pengantar mengenai penelitian yang dilakukan. Bagian ini diawali dengan latar belakang yang membahas pentingnya sistem informasi berbasis *website* dalam mengatasi permasalahan operasional di industri kuliner, khususnya studi kasus pada Waroeng Bebek Geseng. Selanjutnya, dirumuskan rumusan masalah, batasan masalah untuk membatasi ruang lingkup sistem, serta tujuan penelitian yang ingin dicapai. Bab ini juga menjelaskan metode penelitian *Waterfall* yang digunakan sebagai pendekatan pengembangan sistem dan diakhiri dengan sistematika penulisan ini.

BAB II - TINJAUAN PUSTAKA

Bab ini berisi ulasan dan analisis terhadap penelitian-penelitian sebelumnya yang relevan dengan pengembangan sistem informasi. Setiap penelitian dibandingkan dari sisi teknologi yang digunakan, fitur yang dikembangkan, serta kelebihan dan kekurangannya untuk memposisikan penelitian ini dan mengidentifikasi celah yang dapat diisi.

BAB III LANDASAN TEORI

Bab ini menguraikan dasar-dasar teori yang menjadi fondasi dalam perancangan dan pembangunan sistem. Teori yang dibahas mencakup konsep-konsep mengenai *Website*, Sistem Informasi, Transaksi, Restoran, serta teknologi yang digunakan seperti *Framework* React untuk *front-end*, Laravel untuk *back-end*, integrasi antar *framework*, dan sistem manajemen basis data MySQL.

BAB IV - ANALISIS DAN PERANCANGAN SISTEM

Bab ini memaparkan hasil analisis terhadap kebutuhan untuk sistem yang akan dibangun. Berdasarkan analisis tersebut, dilakukan perancangan sistem secara mendetail yang meliputi perancangan alur proses dengan *use case diagram*, penjelasan *use case*, perancangan basis data dengan *Entity-Relationship Diagram* (ERD), perancangan arsitektur, serta perancangan antarmuka pengguna (*user interface*).

BAB V - IMPLEMENTASI DAN PENGUJIAN SISTEM

Bab ini akan menjelaskan proses realisasi dari perancangan sistem ke dalam kode program yang fungsional. Bagian ini akan merinci implementasi pada sisi *front-end* menggunakan React dan *back-end* menggunakan Laravel. Selanjutnya, akan dipaparkan hasil pengujian sistem, yang mencakup pengujian integrasi dan pengujian fungsional untuk memastikan semua fitur berjalan sesuai dengan spesifikasi yang telah ditentukan.

BAB VI - PENUTUP

Bab terakhir ini berisi kesimpulan dari keseluruhan hasil penelitian, yang menjawab tujuan penelitian yang telah ditetapkan. Selain itu, pada bab ini juga akan disampaikan saran-saran yang dapat digunakan sebagai acuan untuk pengembangan sistem di masa mendatang agar menjadi lebih baik dan komprehensif.