

BAB I PENDAHULUAN

1.1. Gambaran Umum

1.1.1. Latar Belakang

Perkembangan teknologi informasi telah mendorong berbagai organisasi, termasuk PT Ramayana Lestari Sentosa Tbk, untuk mengoptimalkan sistem pengelolaan data agar lebih efisien, terintegrasi, dan mudah diakses. Pemanfaatan sistem informasi berbasis *web* terbukti mampu meningkatkan kecepatan akses data serta mengurangi kesalahan dalam pengelolaan persediaan barang, khususnya pada perusahaan ritel dengan volume transaksi yang tinggi[1]. Selama masa magang, penulis terlibat langsung dalam pengelolaan data *stock* dan menemukan bahwa ketersediaan informasi *stock* yang akurat menjadi faktor penting dalam mendukung pengambilan keputusan manajerial.

Permasalahan yang ditemukan di PT Ramayana Lestari Sentosa Tbk adalah belum tersedianya sistem *Kartu Stock DC* yang dapat menampilkan riwayat transaksi *stock* secara terperinci dan *real-time*. Proses monitoring *stock* masih dilakukan manual melalui *query database* langsung, sehingga membutuhkan waktu lama dan rawan kesalahan input. Kondisi ini sejalan dengan penelitian sebelumnya yang menyebutkan bahwa sistem *inventory* yang tidak terintegrasi dapat menyebabkan ketidaksinkronan data serta menurunkan efektivitas operasional perusahaan[2][3]. Diperlukan sebuah sistem laporan *stock* yang mampu mengintegrasikan data antar unit kerja secara terpusat dan *real-time*.

Teknologi yang digunakan dalam sistem ini mencakup kombinasi pengelolaan *backend* dan *frontend*. Framework *Laravel* berperan sebagai pengatur logika *backend* dan manajemen *database*, sementara *Blade Template* dan *Bootstrap* mempermudah pembuatan antarmuka yang responsif dan *user-friendly*. Integrasi teknologi ini memungkinkan laporan *stock* diperbarui secara otomatis, dapat diakses oleh manajemen maupun unit operasional, dan mendukung.

Fitur *Kartu Stock DC* yang dikembangkan memiliki keunggulan dalam

menyajikan data transaksi *stock* secara lengkap mulai dari saldo awal, detail transaksi penerimaan dan pengeluaran, hingga perhitungan saldo akhir secara otomatis. Sistem juga dilengkapi dengan filter dinamis dan fitur *export* ke format *PDF* dan *Excel* yang memudahkan proses pelaporan. Inovasi ini tidak hanya meningkatkan visibilitas dan keteraturan data, tetapi juga mendukung proses analisis yang lebih komprehensif melalui struktur informasi yang sistematis. Framework *Laravel* memberikan fleksibilitas dalam membangun modul-modul tambahan seperti notifikasi stok minimum dan pemetaan alur distribusi barang, sehingga sistem dapat terus dikembangkan sesuai kebutuhan perusahaan.

Tujuan dari pengembangan fitur *Kartu Stock DC* ini adalah untuk membantu tim operasional *Distribution Center* dalam melakukan monitoring transaksi *stock* secara *real-time* dan akurat. Sistem menyediakan informasi lengkap mulai dari saldo awal, detail transaksi penerimaan dan pengeluaran, hingga perhitungan saldo akhir secara otomatis, sehingga mengurangi proses pencarian data manual yang memakan waktu. Dengan adanya filter dinamis berdasarkan *DC*, *sku*, lokasi, dan periode, pengguna dapat dengan cepat melacak riwayat pergerakan *stock* spesifik dan mengidentifikasi pola transaksi untuk kebutuhan analisis atau *audit* internal. Fitur *export* ke *PDF* dan *Excel* juga mempermudah proses dokumentasi dan pelaporan kepada manajemen. Dengan sistem yang terstruktur dan responsif ini, diharapkan perusahaan dapat meningkatkan efisiensi operasional gudang, meminimalkan kesalahan pencatatan stok, serta mendukung pengambilan keputusan yang lebih cepat dan tepat dalam pengelolaan persediaan *Distribution Center*.

1.1.2. Deskripsi Kegiatan

Selama saya magang di perusahaan Ramayana Lestari Sentosa *role* saya itu sebagai *fullstack developer* di bagian *web*, yang dimana sebagai *fullstack developer* pastinya *handle* yang nama nya *frontend*, *backend* dan juga *UI* nya.

Tabel 1.1 Jobdesk Pengembangan Fitur Kartu Stock DC

No	Jobdesk / Tugas	Deskripsi
1.	Identifikasi Kebutuhan Sistem	Melakukan observasi terhadap sistem pelaporan stok yang sudah berjalan dan mengidentifikasi kendala terkait ketiadaan laporan kartu <i>stock DC</i> .
2.	Analisis Kebutuhan Pengguna	Mengumpulkan kebutuhan pengguna internal terkait pelaporan stok berdasarkan struktur organisasi dan lokasi operasional.
3.	Perancangan Struktur Data	Mendesain struktur data yang mendukung penyajian transaksi stok lengkap (saldo awal, transaksi harian, saldo akhir) dari tabel <i>PRODBAL</i> , <i>PRODHIIST</i> , dan <i>TRCODE</i> .
4.	Perancangan Alur Logika Aplikasi	Membuat alur logika untuk menampilkan data stok secara <i>real-time</i> dan terstruktur.
5.	Pengembangan Backend	Mengembangkan <i>Controller</i> , <i>Service</i> , dan <i>Repository</i> menggunakan <i>Laravel</i> untuk mengeksekusi <i>query 3 UNION ALL</i> (Saldo Awal + Transaksi + Saldo Akhir) dari <i>database Oracle</i> .
6.	Pengembangan Frontend	Membuat <i>form filter</i> (<i>DC, SKU, Location, Periode</i>) dan tabel transaksi menggunakan <i>Blade Template</i> , <i>Bootstrap</i> , dan <i>AJAX/Query</i> agar data dapat tampil secara dinamis dan <i>real-time</i> tanpa <i>reload</i> halaman.
7.	Integrasi Frontend-Backend	Menghubungkan <i>API backend</i> dengan tampilan <i>frontend</i> untuk memastikan data <i>stock</i> dapat ditampilkan secara akurat dan <i>real-time</i> .
8.	Pengembangan Fitur Pencarian	Menambahkan fitur pencarian dinamis untuk mempermudah pengguna dalam menemukan stok berdasarkan lokasi atau kategori.
9.	Pengujian Sistem	Melakukan pengujian <i>fungsionalitas</i> untuk memastikan fitur laporan <i>stock</i> berjalan stabil, akurat, dan sesuai kebutuhan pengguna.

No	Jobdesk / Tugas	Deskripsi
10.	Pemeliharaan & Penyempurnaan	Melakukan perbaikan dan penyempurnaan berdasarkan hasil uji dan masukan pengguna.
11.	Dokumentasi & Implementasi	Menyusun dokumentasi fitur serta memastikan sistem siap digunakan pada lingkungan produksi.

Tabel 1.1 menjelaskan rangkaian tugas dalam pengembangan fitur Kartu Stok DC, dimulai dari identifikasi kebutuhan sistem, perancangan struktur data dan alur logika aplikasi, pengembangan **backend** menggunakan Laravel dan **frontend** yang dinamis, integrasi keduanya, hingga pengujian sistem dan implementasi ke lingkungan produksi.

1.2. Rumusan Masalah

Berdasarkan latar belakang yang telah diuraikan di atas, maka rumusan masalah dalam penelitian ini adalah sebagai berikut:

1. Bagaimana merancang dan mengimplementasikan fitur Kartu Stock DC yang dapat menampilkan riwayat transaksi *stock* secara *real-time* pada sistem *RIS* menggunakan pendekatan *Fullstack Development*?

1.3. Batasan Masalah

Batasan masalah dari pengembangan web ris.rumavina.co.id adalah sebagai berikut :

1. Pengembangan sistem difokuskan hanya pada fitur Kartu Stock DC di dalam sistem aplikasi internal *RIS*, tidak mencakup proses transaksi *stock* (*receiving, transfer, adjustment*), manajemen gudang, maupun modul pengadaan barang
2. Sistem menampilkan data transaksi *stock* berdasarkan filter DC (*RDC1/RDC6*), *SKU Number*, *Location*, dan *Periode* (maksimal 1 bulan) sesuai dengan data yang tersedia dalam tabel *PRODBAL*, *PRODHIST*, dan *TRCODE* pada schema *FASHION2016 database Oracle* internal perusahaan.
3. Pengembangan dilakukan dengan pendekatan *Fullstack Development*

menggunakan *framework Laravel* untuk sisi *backend* dan *Blade Template*, *Bootstrap*, serta *AJAX jQuery* untuk sisi *frontend*.

4. Basis data yang digunakan adalah *Oracle Database* dengan koneksi ke *environment MARISDEV* (untuk *development*) dan *MARISPROD* (*read-only* untuk *production*), tanpa melakukan perubahan terhadap skema database utama atau struktur tabel yang sudah ada.
5. Implementasi sistem dilakukan dalam lingkungan internal perusahaan (*dev-ris* dan *production*), sehingga tidak mencakup aspek keamanan data eksternal, integrasi *API* pihak ketiga, maupun akses publik dari luar jaringan perusahaan.
6. Akses fitur laporan dibatasi hanya untuk pengguna internal yang telah *ter-autentikasi* melalui sistem *login RIS*, dengan hak akses sesuai *role user* (*IT Technical*, *Warehouse Staff*, *Management*) yang telah ditentukan oleh sistem *otorisasi* perusahaan.

1.4. Tujuan

Tujuan yang akan dicapai dalam pengembangan ini adalah:

1. Mengembangkan fitur *Kartu Stock DC* yang mampu menampilkan riwayat transaksi *stock* secara lengkap (*saldo awal*, *transaksi harian*, *saldo akhir*) dengan filter dinamis berdasarkan *DC*, *sku*, lokasi, dan periode pada sistem *RIS*.
2. Menerapkan pendekatan *Fullstack Development* dalam proses pengembangan fitur laporan stok agar integrasi antara sisi *frontend* dan *backend* dapat berjalan secara efisien dan responsif.
3. Merancang dan mengimplementasikan tampilan antarmuka laporan yang interaktif dan dinamis, yang memungkinkan pengguna melakukan filter pencarian (*DC*, *sku*, lokasi, periode), menampilkan data secara *real-time* menggunakan *AJAX*, serta mengeksport laporan ke format *PDF* dan *Excel*.