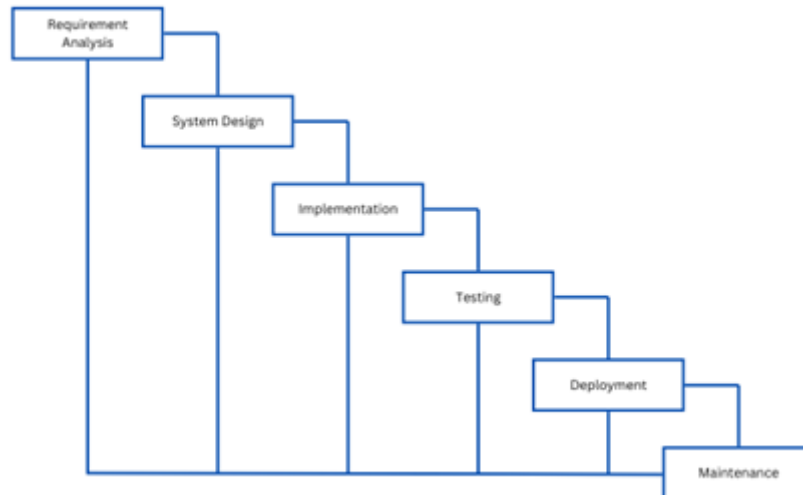


BAB III



Gambar 1. Metode Waterfall

Penelitian ini menggunakan pendekatan rekayasa perangkat lunak dengan model pengembangan sistem waterfall yang terdiri dari enam tahapan utama, yaitu analisis kebutuhan, perancangan sistem, implementasi, pengujian, deployment, dan pemeliharaan[7]. Model ini dipilih karena memiliki struktur yang sistematis dan sesuai untuk pengembangan aplikasi berbasis web dengan kebutuhan yang telah terdefinisi secara jelas. Fokus penelitian ini adalah merancang dan mengimplementasikan sistem pendukung keputusan untuk pemantauan stok dan pengiriman restok produk secara otomatis[8].

Penelitian ini menggunakan metode kuantitatif dengan pendekatan Moving Average dalam sistem Decision Support System (DSS) untuk menentukan rekomendasi jumlah stok produk[9]. Metode Moving Average dipilih karena mampu menangkap pola tren penjualan harian secara sederhana dan memberikan hasil prediksi yang cukup akurat untuk kebutuhan stok harian atau mingguan.

2.1 Analisa Kebutuhan

Data yang digunakan dalam penelitian ini terdiri dari data primer dan sekunder. Data primer diperoleh melalui observasi langsung terhadap alur distribusi dan pencatatan stok di outlet, serta wawancara dengan pengelola gudang dan petugas pengiriman. Data sekunder berupa catatan transaksi penjualan, data stok harian, serta rekap pengiriman produk sebelumnya yang diperoleh dari dokumentasi internal perusahaan. Data-data ini digunakan untuk merancang logika pengambilan keputusan sistem dan untuk kebutuhan pengujian efektivitas sistem dalam memberikan rekomendasi restok[10]. Hasil analisis ini digunakan untuk merumuskan

kebutuhan sistem secara fungsional, seperti kebutuhan input data produk, pemantauan stok harian, notifikasi kebutuhan restok, dan laporan rekomendasi pengiriman ulang. Selain itu, kebutuhan non-fungsional seperti keamanan, kecepatan akses, dan kemudahan penggunaan juga didefinisikan secara rinci[11].

2.2 Perancangan Sistem

Sistem dirancang menggunakan pendekatan Model-View-Controller (MVC) pada Laravel Framework untuk memisahkan logika bisnis, tampilan, dan pengelolaan data secara terstruktur. Laravel memanfaatkan Eloquent ORM untuk interaksi database dan Blade Template Engine untuk membangun antarmuka yang dinamis dan mudah dikembangkan. Basis data dirancang dengan Entity Relationship Diagram (ERD) yang mencakup entitas produk, outlet, stok, dan pengiriman. Pada frontend, digunakan HTML, CSS dengan framework Bootstrap untuk tampilan responsif, serta JavaScript dan Chart.js untuk interaktivitas dan visualisasi data. Rekomendasi restok dihitung secara otomatis di backend menggunakan metode Moving Average berdasarkan rata-rata penjualan harian dalam periode tertentu, sehingga sistem dapat memberikan prediksi kebutuhan stok yang akurat dan real-time[12] -[14].

Rumus Moving Average:

$$MA_t = \frac{P_{t-1} + P_{t-2} + \dots + P_{t-n}}{n}$$

MA_t = Prediksi penjualan hari ke-t

P_{t-n} = Penjualan hari ke-(t-n)

n = Jumlah hari yang digunakan untuk perhitungan rata-rata

2.3 Implementasi

Proses pembangunan sistem sesuai desain yang telah disusun. Sistem dibangun menggunakan Laravel sebagai kerangka kerja backend, dengan MySQL sebagai sistem manajemen basis data. Antarmuka pengguna dikembangkan menggunakan Blade Template dari Laravel dan framework Bootstrap untuk tampilan yang responsif. Beberapa fitur utama yang diimplementasikan meliputi pengelolaan data produk dan outlet, pemantauan stok harian, perhitungan otomatis rekomendasi restok, serta penyajian laporan distribusi dalam bentuk grafik dan tabel interaktif. Sistem juga dilengkapi dengan autentikasi pengguna dan pembatasan akses berdasarkan peran.

2.4 Pengujian

Pengujian sistem dilakukan dengan metode Black Box Testing untuk memastikan setiap fungsi berjalan sesuai spesifikasi dan User Acceptance Testing (UAT) yang melibatkan

pengguna asli dari gudang dan manajemen outlet. Uji coba menggunakan data real untuk mengukur akurasi rekomendasi restok. Hasil pengujian dianalisis untuk mengevaluasi efektivitas dan mengidentifikasi perbaikan yang diperlukan.

2.5 Deployment

Setelah pengujian selesai dan sistem dinyatakan stabil, tahap deployment dilakukan dengan mengunggah sistem ke server hosting yang telah dikonfigurasi untuk menjalankan aplikasi Laravel dan database MySQL. Tahapan ini mencakup migrasi struktur database, pengaturan domain, serta pengujian sistem secara menyeluruh untuk memastikan performa dan fungsionalitas tetap terjaga saat diakses secara publik.

2.6 Pemeliharaan

Pemeliharaan sistem dilakukan secara berkala untuk memastikan performa dan keandalan tetap optimal . Aktivitas meliputi pemantauan server, pencadangan data rutin, perbaikan bug berdasarkan laporan pengguna, serta pengembangan fitur tambahan bila diperlukan. Sistem juga terus diperiksa dari sisi keamanan dan kompatibilitas dengan perangkat dan browser yang digunakan oleh outlet dan pengelola gudang .