

LAPORAN AKHIR PRAKTIKUM

Mata Praktikum : Rekayasa Perangkat Lunak 2
Kelas : 4IA07
Praktikum ke- : 1
Tanggal : 18 Oktober 2025
Materi :
NPM : 51422084
Nama : Muhammad Izzuddin Almansyur
Ketua Asisten : Haikal Abizar
Jumlah Lembar : 5



LABORATORIUM INFORMATIKA
UNIVERSITAS GUNADARMA
2025

1. Jelaskan apa yang dimaksud dengan desain perangkat lunak berbasis komponen dan bagaimana pendekatan ini membantu dalam mengurangi kompleksitas pengembangan perangkat lunak? Sertakan contoh bagaimana komponen dapat digunakan kembali dalam berbagai aplikasi.

Jawab:

Desain perangkat lunak berbasis komponen (Component-Based Software Engineering/CBSE) adalah pendekatan pengembangan perangkat lunak yang menekankan pada penyusunan sistem dari komponen-komponen yang sudah ada. Komponen di sini adalah unit perangkat lunak yang mandiri, dapat digunakan kembali (reusable), dan memiliki antarmuka (interface) yang jelas

Pendekatan desain perangkat lunak berbasis komponen membantu mengurangi kompleksitas pengembangan dengan cara memecah sistem besar menjadi unit-unit kecil yang lebih mudah dipahami, diuji, dan dipelihara. Setiap komponen memiliki fungsi spesifik dan antarmuka yang jelas, sehingga pengembang dapat fokus pada satu bagian tanpa harus memahami keseluruhan sistem secara detail. Selain itu, penggunaan kembali komponen yang sudah ada mempercepat proses pengembangan karena tidak perlu membangun fitur dari awal. Komponen yang telah teruji juga meningkatkan kualitas perangkat lunak secara keseluruhan, sebab stabilitas dan reliabilitasnya sudah terbukti dalam berbagai konteks.

Sebagai contoh, sebuah komponen autentikasi pengguna yang berfungsi untuk login dan logout dapat digunakan kembali pada berbagai aplikasi, mulai dari sistem e-commerce hingga platform e-learning, tanpa perlu ditulis ulang. Demikian pula, komponen pembayaran daring dapat dipakai baik pada aplikasi marketplace maupun aplikasi pemesanan hotel, karena keduanya membutuhkan mekanisme transaksi yang serupa. Komponen lain seperti notifikasi email atau SMS juga dapat digunakan lintas domain, misalnya untuk mengirim kode OTP pada aplikasi perbankan maupun sebagai pengingat jadwal konsultasi pada aplikasi kesehatan.

2. Jelaskan konsep aplikasi terdistribusi dan bandingkan arsitektur 2-tier dan 3-tier. Berikan contoh aplikasi yang dapat diimplementasikan dengan arsitektur 3-tier dan jelaskan keuntungannya.

Jawab:

Aplikasi terdistribusi adalah sistem perangkat lunak yang komponennya dijalankan pada beberapa komputer atau server yang saling terhubung melalui jaringan.

Tujuan utama dari pendekatan ini adalah membagi beban kerja agar sistem lebih efisien, skalabel, dan andal. Dalam praktiknya, aplikasi terdistribusi memisahkan proses antarmuka pengguna, logika bisnis, dan penyimpanan data ke dalam node yang berbeda, sehingga setiap bagian dapat bekerja secara independen namun tetap berkoordinasi melalui protokol komunikasi.

Arsitektur 2-tier terdiri dari dua lapisan utama, yaitu client dan database server. Pada model ini, client tidak hanya menampilkan antarmuka pengguna, tetapi juga menjalankan sebagian besar logika aplikasi, sementara database server bertugas menyimpan dan mengelola data. Kelebihan dari arsitektur ini adalah kesederhanaannya, sehingga mudah diimplementasikan untuk aplikasi kecil. Namun, kelemahannya terletak pada keterbatasan skalabilitas dan keamanan, karena client berinteraksi langsung dengan database.

Sebaliknya, arsitektur 3-tier menambahkan lapisan application server di antara client dan database. Lapisan ini bertugas menangani logika bisnis, sehingga client hanya berfokus pada presentasi dan database hanya berfokus pada penyimpanan data. Pendekatan ini membuat sistem lebih modular, aman, dan mudah diskalakan, meskipun desain awalnya lebih kompleks dibandingkan 2-tier.

Salah satu contoh implementasi arsitektur 3-tier adalah aplikasi e-commerce. Pada lapisan presentasi, pengguna berinteraksi melalui website atau aplikasi mobile untuk mencari produk dan melakukan transaksi. Lapisan logika bisnis berada di server aplikasi yang menangani validasi pesanan, perhitungan ongkos kirim, serta integrasi dengan layanan pembayaran. Sementara itu, lapisan data dikelola oleh server basis data yang menyimpan informasi produk, pengguna, dan riwayat transaksi. Keuntungan dari pendekatan ini adalah keamanan yang lebih baik karena client tidak langsung mengakses database, skalabilitas yang tinggi karena server aplikasi dapat ditambah sesuai kebutuhan, serta kemudahan pemeliharaan karena perubahan pada logika bisnis tidak memengaruhi antarmuka pengguna maupun struktur database.

3. Jelaskan apa itu ERP (Enterprise Resource Planning) dan bagaimana sistem ini dapat membantu perusahaan mengelola operasional mereka. Berikan penjelasan tentang tantangan yang mungkin dihadapi perusahaan saat mengimplementasikan ERP dan bagaimana tantangan tersebut dapat diatasi.

Jawab:

ERP (Enterprise Resource Planning) adalah sistem perangkat lunak terintegrasi yang dirancang untuk menyatukan berbagai fungsi bisnis utama dalam satu platform. Sistem ini memungkinkan perusahaan mengelola keuangan, akuntansi, produksi, inventaris, penjualan, pemasaran, hingga sumber daya manusia secara terpusat. Dengan adanya ERP, setiap departemen tidak lagi bekerja secara terpisah, melainkan saling terhubung melalui basis data yang sama, sehingga informasi yang dihasilkan lebih konsisten, real-time, dan dapat diakses lintas divisi.

ERP membantu perusahaan dengan cara meningkatkan efisiensi operasional melalui otomatisasi proses yang sebelumnya dilakukan secara manual. Sistem ini juga menyediakan data real-time yang dapat digunakan manajemen untuk mengambil keputusan strategis dengan lebih cepat dan tepat. Selain itu, ERP mengurangi risiko kesalahan data karena semua departemen menggunakan sumber informasi yang sama. Kolaborasi antar divisi pun menjadi lebih lancar, misalnya tim penjualan dapat langsung mengetahui ketersediaan stok dari gudang tanpa harus menunggu laporan manual.

Meskipun menawarkan banyak manfaat, implementasi ERP sering menghadapi sejumlah tantangan. Salah satunya adalah biaya investasi yang tinggi, baik untuk lisensi perangkat lunak, infrastruktur, maupun pelatihan karyawan. Selain itu, resistensi dari karyawan juga kerap muncul karena perubahan sistem kerja yang berbeda dari kebiasaan lama. Kompleksitas integrasi dengan sistem lama (legacy systems) dapat menjadi hambatan, terutama jika data yang ada tidak seragam. Tantangan lain adalah risiko kustomisasi berlebihan yang membuat sistem sulit dipelihara.

4. Apa peran middleware dalam aplikasi terdistribusi berbasis komponen? Jelaskan bagaimana middleware berfungsi sebagai penghubung antara front-end dan back-end serta memberikan contoh nyata dari implementasi middleware dalam suatu sistem.

Jawab:

Middleware adalah perangkat lunak perantara yang berfungsi sebagai “jembatan” antara berbagai komponen dalam aplikasi terdistribusi. Dalam konteks sistem berbasis komponen, middleware memungkinkan komunikasi, koordinasi, dan integrasi antarbagian aplikasi yang berjalan pada lingkungan berbeda. Perannya sangat penting karena ia menyembunyikan kompleksitas teknis jaringan, protokol komunikasi, maupun heterogenitas platform, sehingga pengembang dapat lebih fokus pada logika bisnis tanpa harus memikirkan detail teknis konektivitas.

Dalam arsitektur aplikasi terdistribusi, front-end biasanya berfungsi sebagai lapisan presentasi yang berinteraksi langsung dengan pengguna, sementara back-end menangani logika bisnis dan pengelolaan data. Middleware berada di antara keduanya untuk mengatur aliran data dan permintaan. Misalnya, ketika pengguna mengirimkan permintaan melalui antarmuka web atau aplikasi mobile, middleware akan menerima permintaan tersebut, menerjemahkannya ke dalam format yang dapat dipahami oleh server aplikasi atau basis data, lalu mengembalikan hasilnya ke front-end.

Salah satu contoh nyata implementasi middleware adalah Express.js dalam ekosistem Node.js.. Express.js berfungsi sebagai middleware yang menghubungkan permintaan dari client (front-end) dengan logika bisnis dan basis data di back-end. Ia dapat menangani routing, autentikasi, validasi input, hingga pengelolaan session.