

## **LAPORAN AKHIR PRAKTIKUM**

Mata Praktikum : Rekayasa Perangkat Lunak 2  
Kelas : 4IA18  
Praktikum ke- : 1  
Tanggal : 18 Oktober 2025  
Materi : Pengenalan RPL  
NPM : 50422215  
Nama : Andi Purnama  
Ketua Asisten : Haikal Abizar  
Paraf Asisten : -  
Nama Asisten : 1.  
2.  
3.  
Jumlah Lembar : 3 Lembar



**LABORATORIUM INFORMATIKA**  
**UNIVERSITAS GUNADARMA**

**2025**

## Soal

---

**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 adalah pendekatan pengembangan perangkat lunak yang membagi sistem menjadi bagian-bagian kecil yang disebut komponen, di mana setiap komponen memiliki fungsi tertentu dan dapat digunakan kembali di berbagai aplikasi.

Pendekatan ini membantu mengurangi kompleksitas karena pengembang tidak perlu membangun seluruh sistem sekaligus. Setiap komponen bisa dikembangkan, diuji, dan diperbaiki secara terpisah tanpa memengaruhi bagian lain. Selain itu, karena komponen bisa digunakan kembali, proses pengembangan menjadi lebih cepat, efisien, dan hemat biaya.

Sebagai contoh, komponen login pengguna yang sudah dibuat untuk satu aplikasi bisa digunakan kembali pada aplikasi lain seperti e-commerce, sistem sekolah, atau portal berita. Dengan cara ini, desain berbasis komponen membantu menciptakan perangkat lunak yang modular, mudah dipelihara, dan fleksibel untuk dikembangkan di masa depan.

**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:** Secara keseluruhan, aplikasi terdistribusi adalah sistem yang komponennya dijalankan di beberapa komputer atau server yang saling terhubung melalui jaringan, dengan tujuan membagi beban kerja agar sistem menjadi lebih efisien, cepat, dan mudah dikembangkan. Dalam penerapannya, arsitektur aplikasi terdistribusi umumnya menggunakan model 2-tier atau 3-tier.

Pada arsitektur 2-tier, client berinteraksi langsung dengan database. Struktur ini sederhana dan cocok untuk aplikasi berskala kecil, namun memiliki kelemahan dari sisi keamanan dan skalabilitas karena seluruh logika bisnis berada di sisi client. Sementara itu, arsitektur 3-tier menambahkan satu lapisan tambahan di antara client dan database, yaitu application layer yang menangani logika bisnis. Pendekatan ini memisahkan tanggung jawab tiap lapisan — tampilan, logika, dan data — sehingga membuat sistem lebih terstruktur, aman, serta mudah dikelola.

Contoh penerapan 3-tier dapat ditemukan pada aplikasi web modern seperti e-commerce (Tokopedia, Shopee, Lazada), sistem akademik online, atau aplikasi perbankan digital. Dalam sistem seperti ini, lapisan presentasi berada di sisi pengguna (browser atau aplikasi mobile), lapisan logika bisnis berjalan di server aplikasi (misalnya menggunakan Node.js, Laravel, atau Spring Boot), dan lapisan data dikelola oleh database server (seperti MySQL atau PostgreSQL).

Dengan arsitektur 3-tier, aplikasi memperoleh berbagai keuntungan seperti peningkatan keamanan karena client tidak langsung berhubungan dengan database, kemudahan

pengembangan dan pemeliharaan karena perubahan bisa dilakukan tanpa mempengaruhi seluruh sistem, skalabilitas yang lebih baik karena setiap lapisan bisa dikembangkan secara terpisah, serta kemampuan integrasi yang lebih fleksibel dengan layanan lain.

Singkatnya, arsitektur 3-tier memberikan pondasi yang kuat, aman, dan fleksibel untuk membangun aplikasi terdistribusi modern yang dapat berkembang sesuai kebutuhan pengguna dan teknologi masa kini.

**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 terpadu yang digunakan perusahaan untuk mengelola berbagai aktivitas bisnis seperti keuangan, produksi, penjualan, dan SDM dalam satu platform. Dengan ERP, data antarbagian saling terhubung, sehingga proses bisnis menjadi lebih efisien, transparan, dan mudah dikontrol.

Sistem ini membantu perusahaan meningkatkan koordinasi, mempercepat pengambilan keputusan, dan mengurangi kesalahan akibat data yang terpisah. Namun, tantangan yang sering muncul meliputi biaya implementasi yang tinggi, penyesuaian proses bisnis, dan kurangnya kesiapan pengguna.

Untuk mengatasinya, perusahaan perlu melakukan perencanaan yang baik, memberikan pelatihan kepada karyawan, dan memilih solusi ERP yang sesuai dengan kebutuhan serta kapasitas organisasi.

**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 berperan sebagai *jembatan penghubung* antara front-end (bagian tampilan pengguna) dan back-end (logika bisnis serta database) dalam aplikasi terdistribusi berbasis komponen. Ia memastikan komunikasi antarbagian sistem berjalan lancar, aman, dan efisien — misalnya dengan menangani pertukaran data, autentikasi, atau pengiriman permintaan dari client ke server.

Secara sederhana, middleware membuat berbagai komponen aplikasi yang berbeda teknologi tetap bisa “berbicara” satu sama lain.

Contoh nyata penerapannya adalah Express.js pada aplikasi web berbasis Node.js, yang berfungsi mengelola permintaan HTTP dari front-end ke back-end. Contoh lainnya adalah message broker seperti RabbitMQ atau Kafka, yang membantu mengatur pengiriman pesan antar layanan pada sistem berbasis microservices.