

RANCANG BANGUN WEBSITE TOKO TAMBUNAN PUPUK MENGUNAKAN MIRCROSERVICE

Proyek Pengembangan Aplikasi Terdistribusi

KELOMPOK 10



Richard Aritonang
11322010



Agustin A Tambunan
11322026



Cristian N Tambunan
11322040



Nita H K Simangunsong
11322056

DESKRIPSI

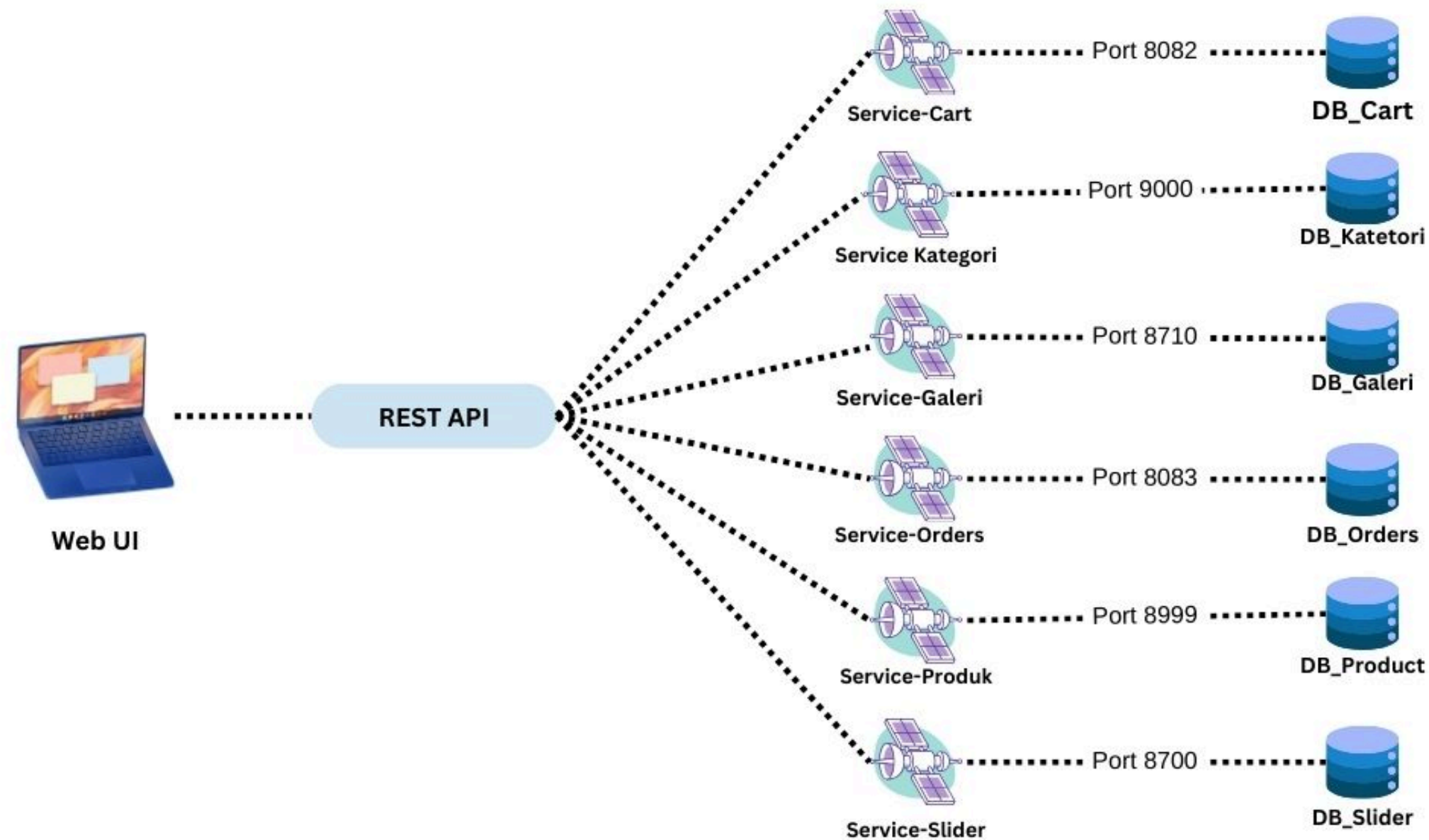
Aplikasi berbasis web ini dirancang untuk sistem layanan jual beli online yang memungkinkan admin dan pelanggan untuk menjual dan membeli barang. Dalam aplikasi ini, admin bertugas menambahkan produk yang dijual, sementara pelanggan dapat membeli produk tersebut.

Website ini dikembangkan menggunakan dua bahasa pemrograman yang berbeda untuk bagian back-end dan front-end. Bagian back-end dibangun menggunakan bahasa Go dan framework Laravel, sedangkan bagian front-end menggunakan bahasa PHP dengan framework Laravel.

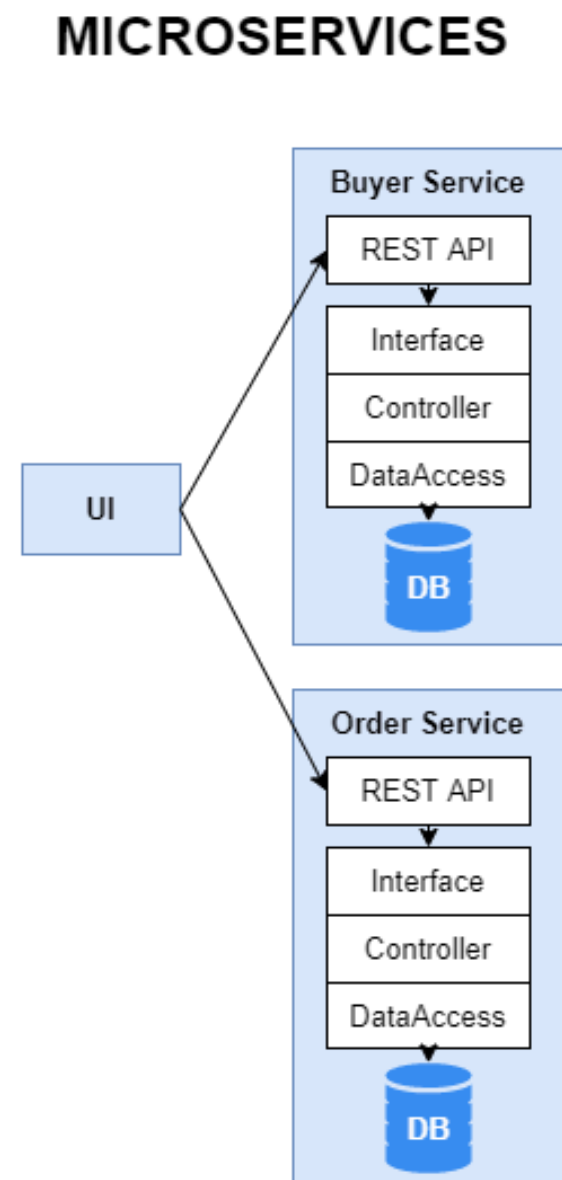
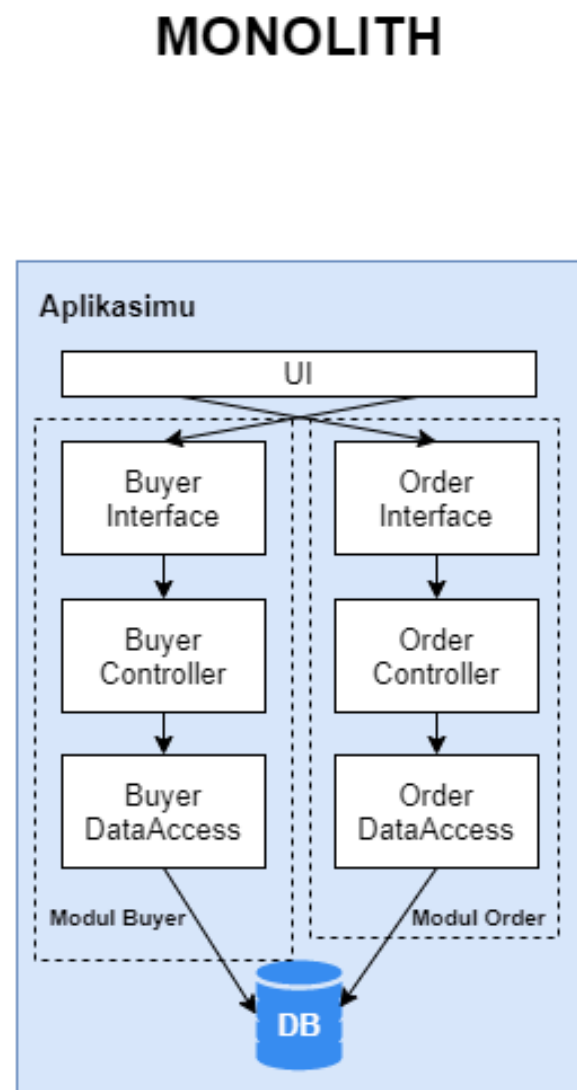
Website ini mengadopsi arsitektur microservice, yang membagi sistem menjadi layanan-layanan yang lebih kecil dan independen. Setiap layanan memiliki database sendiri. Layanan yang ada dalam website ini meliputi Service_cart, Service_Category, Service_Product, Service_Galeri, Service_Slider, dan Service_Orders. Setiap layanan berjalan secara independen, sehingga satu layanan tidak akan mempengaruhi layanan lainnya. Namun, meskipun mandiri, setiap layanan tetap saling terhubung dan berkomunikasi satu sama lain menggunakan HTTP request dengan standar REST API.

ARSITEKTUR MICROSERVICE

Arsitektur microservice yang ditunjukkan pada diagram di samping menggambarkan bagaimana berbagai layanan (service) dalam website Toko Tambunan Pupuk berfungsi secara independen namun tetap terhubung melalui REST API.



PERBEDAAN ARSITEKTUR MICROSERVICE DAN MONOLITH



Pada arsitektur monolitik, seluruh komponen aplikasi seperti antarmuka pengguna (UI) terintegrasi dalam satu kesatuan aplikasi dan berbagi satu basis data tunggal. Semua bagian saling bergantung satu sama lain, sehingga jika ada satu bagian yang gagal, bisa mempengaruhi keseluruhan sistem.

Sementara pada arsitektur microservices, aplikasi dipecah menjadi beberapa layanan kecil yang independen, masing-masing dengan fungsi spesifik. Setiap layanan memiliki antarmuka, controller, dan akses data sendiri serta dapat memiliki basis data terpisah. Layanan-layanan ini berkomunikasi satu sama lain melalui API (REST API). Pendekatan ini memungkinkan fleksibilitas yang lebih besar, skalabilitas, dan ketahanan yang lebih baik, karena kegagalan pada satu layanan tidak akan mempengaruhi layanan lainnya.

FUNGSI

1. Fungsi Autentikasi
2. Fungsi Melihat Produk
3. Fungsi Melihat Detail Produk
4. Fungsi Menambah Produk ke keranjang
5. Fungsi Melihat Halaman Keranjang
6. Fungsi Menghapus Produk dari keranjang
7. Fungsi CRUD Produk
8. Fungsi CRUD Kategori
9. Fungsi CRUD Galeri
10. Fungsi Melihat Daftar Pesanan

USECASE DIAGRAM





DEMO PROYEK

KESIMPULAN

Dengan menggunakan microservice, setiap service pada website yang dibangun dapat berjalan secara independent. Artinya, ketika service lain sedang down, hal ini tidak akan memengaruhi service lainnya.



**TERIMA
KASIH**