

IMPLEMENTASI ARSITEKTUR MICROSERVICES PADA RANCANG BANGUN APLIKASI ONLINE SHOP BERBASIS WEB

Achmad Buhori¹, Diva Lutfiando², Fachry Syah Alam³

Program Studi Teknologi Informasi, Fakultas Teknik Informatika dan Pariwisata,
Institut Bisnis dan Informatika Kesatuan Bogor. Jl. Ranga Gading No.01,
Gudang, Kecamatan Bogor Tengah, Kota Bogor, Jawa Barat 16123.

202310064@student.ibik.ac.id¹, 202310028@student.ibik.ac.id²,
202310065@student.ibik.ac.id³

ABSTRAK

Kemajuan Teknologi Informasi saat ini sangat membantu sekali kegiatan manusia dalam melakukan jual dan beli yang bisa dilakukan secara online melalui Toko Online ataupun Marketplace. Untuk memenuhi kebutuhan konsumen sangat dibutuhkan peranan bidang teknologi dalam proses penjualan, pembelian, hingga efisiensi dan efektifitas. Terdapat banyak sekali toko-toko *sparepart* motor yang tersebar luas di berbagai daerah, dalam hal tersebut menimbulkan masalah yaitu membutuhkan banyak waktu untuk berbelanja kebutuhan kendaraan dan juga menjadi masalah bagi konsumen yang bertempat tinggal jauh dari keramaian kota. Aplikasi berbasis *web* yang akan dibangun guna untuk mempermudah konsumen dalam berbelanja kebutuhan *sparepart* ataupun *accessories* kendaraannya. Aplikasi ini nantinya akan dibuat berbasis *Online Shop* dan menggunakan arsitektur *Microservices*. *Microservices* merupakan arsitektur yang membagi sistem menjadi service-service kecil. Penggunaan arsitektur tersebut dapat mengatasi masalah pada arsitektur monolith yaitu penggunaan resources yang besar pada server. Untuk proses pengolahan data pada tiap-tiap service menggunakan application programming interface (Api) dan menggunakan javascript web token (Jwt) sebagai otorisasi untuk menggunakan end point yang disediakan. Untuk mendukung arsitektur tersebut dapat menggunakan docker dalam proses manajemen service. Pada penelitian ini terdiri dari 3 service yaitu *API-gateway*, *Product Service*, *Account Service* yang berjalan dalam *docker container*. Berdasarkan uji coba yang telah dilakukan pada penelitian ini, bahwa aplikasi ini memiliki tingkat keberhasilan 99% pada pengujian end point dalam mengolah data pada service-service yang terdapat dalam *docker container*, lalu pengujian juga dilakukan pada *system* secara fungsional dan menghasilkan nilai 100% dengan berbagai macam skenario.

Kata kunci : Online Shop, Sparepart motor, Docker, Microservices

ABSTRACT

Advances in Information Technology are currently very helpful for human activities in buying and selling which can be done online through Online Stores or Marketplaces. To meet consumer needs, the role of technology is needed in the process of selling, purchasing, to efficiency and effectiveness. There are lots of motorbike spare parts shops that are widespread in various regions, in this case it creates a problem, namely it takes a lot of time to shop for vehicle needs and is also a problem for consumers who live far from the hustle and bustle of the city. A web-based application will be built to make it easier for consumers to shop for spare parts or accessories for their vehicles. This application will be made based on the Online Shop and uses the Microservices architecture. Microservices is an architecture that divides the system into small services. The use of this architecture can overcome the problem of the monolith architecture, namely the use of large resources on the server. For data processing, each service uses an application programming interface (Api) and uses a javascript web token (Jwt) as authorization to use the end point provided. To support this architecture, you can use docker in the service management process. In this study, it consists of 3 services, namely API-gateway, Product Service, Service Account that runs in a docker container. Based on the trials that have been carried out in this study, this application has a 99% success rate in end point testing in processing data on services contained in the docker container, then testing is also carried out on the system functionally and produces a value of 100% with various kinds of scenarios.

Keywords: Online Shop, Motor spareparts, Docker, Microservices

I. PENDAHULUAN

Dalam zaman teknologi dan informasi seperti sekarang ini, manusia sangat terbantu sekali dengan adanya perkembangan teknologi yang sangat pesat, contohnya yaitu dalam bidang Bisnis. Dalam bidang tersebut peranan Teknologi dan Informasi menjadi hal sangat penting sekali untuk mendukung pengambilan keputusan.

Perkembangan teknologi juga terjadi pada arsitektur aplikasi, yaitu *Microservices Architecture*. *Microservices Architecture* semakin populer dan pengadopsi menikmati kesuksesan yang luar biasa. Laporan “*Microservices Adoption in 2020*”, berdasarkan jajak pendapat 1.500 insinyur perangkat lunak, sistem dan arsitek teknis, insinyur dan pembuat keputusan, menyatakan bahwa lebih

dari tiga perempat (77%) bisnis ini telah mengadopsi layanan mikro. Mereka yang mengadopsi *Microservices Architecture*, hampir semua (95%) melaporkan tingkat keberhasilan yang sangat tinggi. Lebih jauh lagi, kini sebagian besar bisnis sudah berbasis *Online*. Yang tentu saja mereka bertaruh sangat besar pada teknologi, Toko *Online* kini sudah merangkap semua jenis produk dimulai dari produk tradisional hingga teknologi terbaru. Sehingga organisasi IT kini mulai berlomba untuk menyediakan aplikasi Toko *Online* berbasis layanan bagi masyarakat untuk memudahkan dalam memenuhi kebutuhannya.

Microservices Architecture memiliki konsep aplikasi dibagi menjadi bagian-bagian kecil yang berfungsi spesifik dan tidak bergantung pada komponen program lainnya, konsep tersebut bertujuan agar sistem yang dibangun bisa menangani kegagalan total jika terdapat satu aplikasi yang bermasalah. Kelebihan arsitektur *microservices* yaitu memudahkan developer dalam proses development sistem, memungkinkan setiap service bisa dikembangkan secara independent, tidak ada hambatan dalam

menggunakan teknologi baru Arsitektur ini digunakan untuk mengganti penggunaan arsitektur monolitik yang merupakan sebuah arsitektur yang semua komponennya menjadi satu kesatuan. Kekurangan arsitektur monolitik yaitu menggunakan banyak resources, sulit melakukan *maintenance*, dan jika terjadi *error* pada satu fungsi maka akan mempengaruhi seluruh aplikasi, hal ini dikarenakan karena menggunakan *single* database.

II. LANDASAN TEORI

A. Arsitektur Microservices

Microservice berarti membagi aplikasi menjadi layanan yang lebih kecil dan saling terhubung tidak seperti aplikasi monolitik. Arsitektur *microservice* merupakan alternatif arsitektur yang lebih terukur dan lebih fleksibel. Dengan pendekatan ini, suatu sistem informasi akan terdiri dari beberapa *service* yang dapat dikelola dan didistribusikan secara independent, hal ini akan lebih memudahkan sistem untuk beradaptasi terhadap perubahan kebutuhan.

B. Rest API

REST adalah singkatan dari (***Representational State Transfer***) dan merupakan salah satu

tipe arsitektur *software*. *REST* mewakili serangkaian batasan yang harus diikuti saat Anda mengembangkan aplikasi di internet. *API* akan disebut menjadi *RESTful API* apabila mampu memenuhi ketentuan dan Batasan yang ditetapkan oleh gaya arsitektur ini.^[3]

C. JSON

JSON (JavaScript Object Notation) adalah format pertukaran data ringan, mudah dibaca dan ditulis oleh manusia, serta mudah diterjemahkan dan dibuat (*generate*) oleh komputer. Standar ECMA-262 Edisi ke-3- Desember 1999. *JSON* merupakan format teks yang tidak bergantung pada bahasa pemrograman apapun karena menggunakan gaya bahasa yang umum digunakan oleh programmer keluarga C termasuk C, C++, C#, Java, JavaScript, Perl, Python.^[4]

D. Toko Online

Toko *online* didefinisikan sebagai sebuah sistem informasi pada sebuah toko yang memungkinkan pembeli untuk berpartisipasi untuk bertukar informasi tentang harga dan produk yang akan dibeli. Di toko *Online* ini hanya menampilkan atau menyajikan produk dari toko terkait atau sebuah perusahaan yang mengembangkan

tokonya menjadi berbasis Aplikasi Web/menjadi *Online Shop*. Atau dimana pemilik *online shop* adalah hanya sebagai fasilitator yang memusatkan berbagai macam informasi mengenai produk dan jasa dari berbagai penjual sehingga pembeli bisa membandingkan harga.

E. Docker

Docker adalah sebuah platform terbuka untuk siapapun yang bertujuan menggunakan sebuah platform untuk membangun, mendistribusikan dan menjalankan aplikasi dimanapun seperti laptop, data center, virtual machine ataupun cloud. *Docker* menggunakan arsitektur *client-server*. *Docker client* menghubungi *Docker daemon*, yang melakukan pekerjaan berat, menjalankan, dan mendistribusikan *Docker container* anda. Kedua *Docker client* dan *daemon* dapat berjalan pada sistem yang sama.

F. Node Js

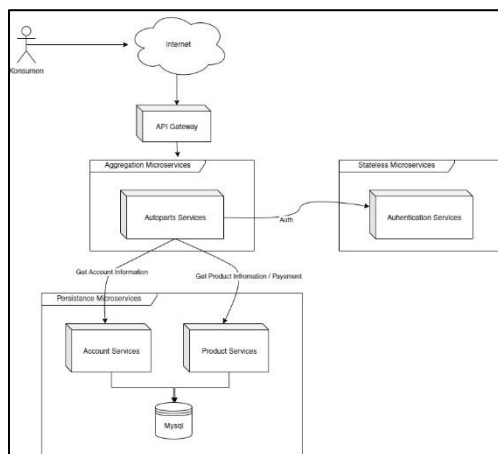
Node.js adalah sistem perangkat lunak yang didesain untuk pengembangan aplikasi web. Aplikasi ini ditulis dalam bahasa *JavaScript*, Tidak seperti kebanyakan bahasa *JavaScript* yang dijalankan pada peramban, *Node.js* dieksekusi sebagai aplikasi server. Aplikasi ini terdiri dari

V8 *JavaScript Engine* buatan Google dan beberapa modul bawaan yang terintegrasi.^[5]

III. ANALISIS DAN PERANCANGAN

A. Analisis

Dalam pembuatan aplikasi *Autoparts (Online Shop)* ini perlu dilakukan analisis *system* atau melakukan penguraian terhadap sebuah *system* yang utuh menjadi bagian-bagian komponennya. Dengan tujuan untuk mengidentifikasi dan mengevaluasi permasalahan, kesempatan, hambatan apa yang akan terjadi, dan bagaimana cara mengatasinya. Jika digambarkan dalam *Arsitektur Microservices* dari website *online shop Autoparts* adalah seperti di bawah ini :



Gambar 1. Desain Arsitektur Microservice

Dalam proses pembuatan toko *online* ini menggunakan arsitektur *microservices*, pada penggunaan

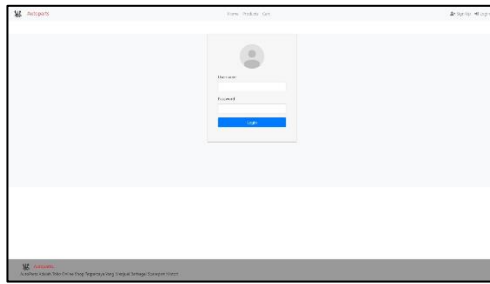
arsitektur ini terdiri dari banyaknya servis-servis. Servis-servis tersebut saling berhubungan, yang berfungsi menghubungkannya adalah *API-Gateway* dan juga sebagai pengaman agar servis yang ada tidak dapat diakses oleh publik atau bersifat *private*.

Sebelum konsumen dapat mengakses Menu *Home*, Menu Produk, Menu Profil dan melakukan *Checkout*. Pengguna atau Konsumen harus terlebih dahulu melakukan registrasi atau pendaftaran pada menu *Signup* dan jika sudah punya akun masuk ke menu *Login*. Untuk melakukan autentikasi. Jika proses autentikasi berhasil, maka *API-Gateway* akan memberikan akses kepada konsumen agar bisa masuk ke menu selanjutnya.

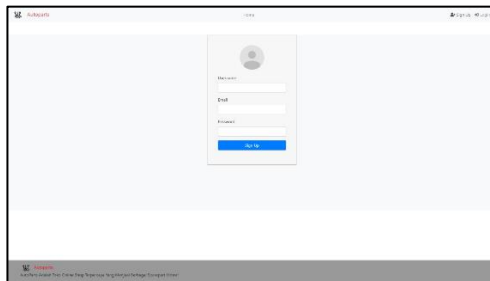
B. Implementasi Antarmuka Autoparts

- Halaman *Login & Sign Up*

Halaman ini merupakan halaman paling utama atau halaman untuk autentikasi akun sebelum konsumen atau pengguna bisa mengakses halaman lainnya.



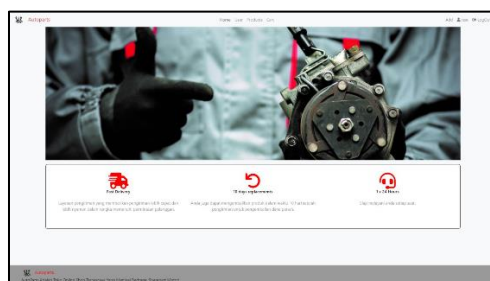
Gambar 2. Halaman *Login*



Gambar 3. Halaman *Signup*

- Halaman *Home*/Utama

Pada halaman ini terdapat sebuah tampilan banner terkait Autoparts dan di halaman ini juga terdapat mengenai informasi ekspedisi toko tersebut dan juga tentunya pada halaman tersebut terdapat juga menu bar dari menu produk, profil, *search*, dan *logout*.



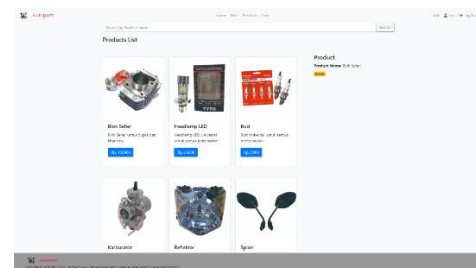
Gambar 4. Halaman *Home*

- Halaman Produk

Pelanggan/konsumen dapat mencari produk yang di inginkan terkait sparepart sepeda motor pada menu

Produk. Pada website Autoparts disediakan juga *search bar*, agar pengguna/konsumen dengan mudah menemukan barang taupun produk yang sedang di cari.

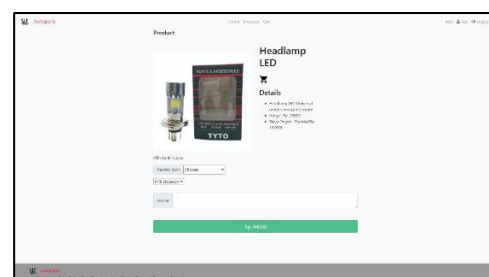
Pada halaman pproduk ini terdapat card yang berisi foto produk, harga produk dan deskripsi produk, Konsumen bisa memesan produk dengan melakukan klik pada tombol “detail/harga produk”.



Gambar 5. Halaman *Product*

- Halaman Detail Produk/Checkout

Di halaman ini terdapat mengenai detail dari produk terkait seperti (harga, ongkos kirim, deskripsi produk, rekening tujuan , ekspedisi, jumlah produk yang ingin dibeli, alamat dan tombol checkout).



Gambar 6. Halaman *Detail/Checkout*

IV. KESIMPULAN DAN SARAN

A. Kesimpulan

Berdasarkan hasil penelitian, pengujian dan implementasi dan pembahasan yang telah dilakukan, maka didapatkan kesimpulan sebagai berikut :

1. Pada penelitian ini menghasilkan sebuah aplikasi *Online Shop* berbasis Website yang menggunakan arsitektur *microservices* yang mana terdiri dari 4 servis meliputi *api-gateway*, *home* servis, produk servis, profil/akun servis.
2. Aplikasi marketplace menggunakan arsitektur *microservices* telah berhasil dibangun dengan hasil persentase pengujian black box 95% dari 10 aktivitas berhasil dan hasil persentase pengujian endpoint api 90% dari 10 aktivitas berhasil.

B. Saran

Berdasarkan hasil penelitian, pengujian, implementasi dan pembahasan yang telah dilakukan, maka penulis menyarankan sebagai berikut :

1. Peneliti selanjutnya dapat mengembangkan pembayaran beberapa produk langsung untuk mempermudah proses pembayaran.

2. Peneliti selanjutnya dapat mengembangkan tampilan *UI* dari website *Autoparts* ini.

V. REFRENSI

- [1] Sinambela A., Ernawati dan Coastera F. F. "Implementasi Arsitektur *Microservices* Pada Rancang Bangun Aplikasi Marketplace Berbasis Web" Universitas Bengkulu, 2021
- [2] Alchuluq M. L. dan Nurzaman F. "Analisis Pada Arsitektur *Microservice* Untuk Layanan Bisnis Toko Online"
- [3] W. Fitri, "Apa Itu RESTful API? Pengertian, Cara Kerja, dan Manfaatnya" Dec -2022.
- [4] K. B. Surarso and W. A. Syafei, "Implementasi Metode Analytic Network Proses Untuk Penentuan Prioritas Penanganan Jalan Berdasarkan Tingkat Pelayanan Jalan," Jurnal Sistem Informasi Bisnis, pp. 105-113, 2016.
- [5] R. and M. Shalahuddin, Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek, Bandung: Informatika Bandung, 2018.
- [6] R. "Analisa Perbandingan Metode Anp Dan Saw Dalam Menentukan Mahasiswa Terbaik," Jurnal Mantik Penusa, pp. 43-50, 2019.