MICROSERVICE BASED SOFTWARE ARCHITECTURE ON CANTEEN INFORMATION MANAGEMENT SYSTEM

by Mukhammad Dayu Anwar

Submission date: 10-Aug-2023 09:57AM (UTC+0700)
Submission ID: 2143771918
File name: jurnal_dayu.pdf (728.94K)
Word count: 2275
Character count: 14107

MICROSERVICE BASED SOFTWARE ARCHITECTURE ON CANTEEN INFORMATION MANAGEMENT SYSTEM

Mukhammad Dayu Anwar[1], Irwan A. Kautsar, S.Kom., M.Kom., Ph.D[2]

[1]Program Studi Informatika, Universitas Muhammadiyah Sidoarjo, Indonesia
[2]Program Studi Informatika, Universitas Muhammadiyah Sidoarjo, Indonesia
Email : 191080200269@umsida.ac.id[1]

Abstract. *Microservices is one way to maximize the performance of the data service that will be used to create the system. In addition, the service will be easy to develop, because each service is structured and separate for data that will be consumed by other developers. In using microservice, the data that will be displayed is the Application Programming Interface (API), which is currently a very suitable format. The case we use in this research is the E-Canteen Application, where the circulation of data that will be consumed or used by users must run smoothly, because in the system there are transactions, payments and monitoring. If there are many data requests from users and data circulation is not good, it will affect the service or data server. By implementing microservices in this E-Canteen Application, data circulation, the speed of the data generated will be very helpful to maximize each service used. The results of testing the microservices that have been created, show that the circulation of data and services that have been used become very efficient and stable when there are many requests or accesses from users. With these results, the effect of microservices for data circulation used becomes very efficient and stable.*

Keywords *- Microservices, E-Canteen, API*

Abstrak. *Microservices menjadi salah satu cara untuk memaksimalkan performa dari service data yang akan digunakan untuk membuat sistem. Selain itu service akan mudah di kembangkan, karena setiap service nya sudah terstruktur dan terpisah untuk data yang akan dikonsumsi oleh developer lain. Dalam penggunaan microservice data yang akan ditampilkan adalah Application Programming Interface (API), yang dimana format tersebut saat ini sangat cocok digunakan. Kasus yang kita gunakan dalam penelitian ini adalah Aplikasi E-Kantin, yang dimana sirkulasi data yang akan dikonsumsi atau digunakan oleh user harus berjalan dengan lancar, karena dalam sistem tersebut terdapat transaksi, pembayaran dan monitoring. Jika ada banyak request data dari user pemakai dan sirkulasi data kurang baik, maka akan berpengaruh pada service maupun server data tersebut. Dengan diimplementasikan nya microservices pada Aplikasi E-Kantin ini, maka sirkulasi data, kecepatan data yang dihasilkan akan sangat membantu untuk memaksimalkan setiap service yang dipakai. Hasil dari pengujian microservices yang telah dibuat, menunjukkan bahwa sirkulasi data dan service yang telah digunakan menjadi sangat efisien dan stabil pada saat banyak nya request atau akses dari user pemakainya. Dengan hasil tersebut, maka pengaruh dari microservices untuk sirkulasi data yang digunakan menjadi sangat efisien dan stabil.*

Kata Kunci *– Microservices, E-Kantin, API*

I. PENDAHULUAN

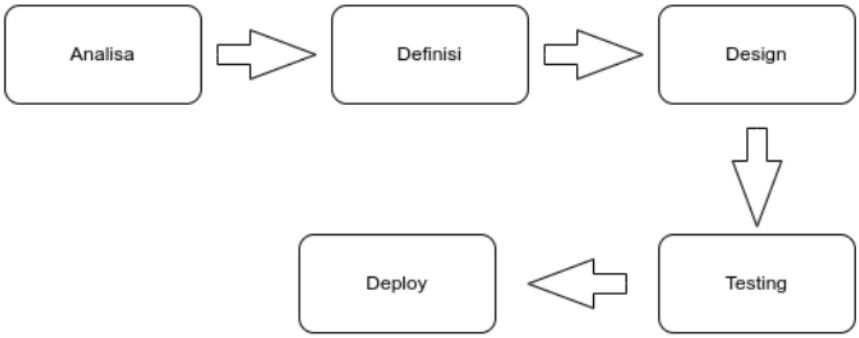
Microservices adalah kumpulan layanan yang saling berkaitan untuk membangun sebuah sistem yang terstruktur, agar sistem tersebut berjalan dengan maksimal [1]. Komunikasi antar setiap service atau layanan nya menggunakan API (Application programming Interface), karena cocok sebagai perantara service yang saling berhubungan. API sendiri adalah sebuah perantara yang bisa digunakan untuk menyatukan banyak service atau menghubungkan antara tampilan dan data di sebuah sistem. Keunggulan dari Microservices adalah leluasa saat ada pengembangan sistem yang rumit, data saat terdistribusi menjadi lebih cepat dan efisien, akses untuk setiap service nya lebih mudah dan terstruktur.

Kantin adalah tempat atau fasilitas yang menyediakan makanan maupun minuman untuk orang orang yang ingin makan, baik itu beli di tempat tersebut atau membawa bekal sendiri. Selain itu kantin harus bersih dan higenis agar orang orang disitu merasa nyaman saat berada di tempat tersebut. Sekarang ini kebanyakan cara pembayaran untuk membeli makanan atau minumannya menggunakan manual atau uang cash. Di situasi yang sedang ada wabah penyakit (covid) saat ini, sangat tidak di anjurkan menggunakan uang cash karena dapat menyebarkan wabah penyakit ke orang lain. Saat sistem kantin tersebut sudah dibuat dan banyak pemakainya, data akan sering diambil, diupdate, ditambah dan dikurangi oleh user pemakainya, Jika architecture datanya itu tidak dibuat sebagus mungkin, maka efeknya performa dari aplikasi tersebut akan menurun dan pada saat pengembangan sistemnya akan sangat rumit karena sudah banyak fitur yang dibuat maupun struktur yang sudah dikerjakan oleh developer.

Untuk mengatasi masalah ini, peneliti membuat sebuah aplikasi yang bisa memesan makanan ataupun minuman di kantin tersebut dan cara pembayaran nya tidak menggunakan manual lagi. Dalam aplikasi tersebut akan diterapkan Architecture Microservices agar sirkulasi data yang masuk dan keluar lebih optimal, efisien dan cepat. Dengan menggunakan Microservice ini diharapkan aplikasi yang dibuat akan semakin maksimal performanya, karena setiap layanan datanya terstruktur dan terdistribusi dengan baik.

Dengan adanya permasalahan di atas maka peneliti membuat solusi dengan judul "PEMANFAATAN MICROSERVICES DI APLIKASI KOMERSIAL STUDY KASUS APLIKASI E-KANTIN". Agar data dari aplikasi tersebut bisa efisien dan cepat maka Architecture Microservices harus diterapkan untuk sirkulasi data layanannya. Untuk bahasa pemrograman yang dipakai dalam membuat Architecture Microservices diatas, peneliti menggunakan bahasa Python dan menggunakan Framework Flask.

II. METODE



Gambar 1. Metode Agile

2.1 Analisa

Pada penelitian ini metode yang digunakan adalah agile. Karena metode ini memudahkan pengembang sistem untuk menyesuaikan kebutuhan yang akan diselesaikan, selain itu metode ini sangat cocok digunakan karena dilakukan secara bertahap dan berulang. Selanjutnya penelitian terdahulu dijadikan sebagai masukan dan referensi untuk membantu penelitian sehingga dapat menambah teori dan wawasan. Jurnal penelitian yang digunakan untuk referensi adalah penelitian [1] menjelaskan proses migrasi sistem alat pendukung rapid prototyping dari monolit ke arsitektur microservice yang akan digunakan sebagai implementasi Project Based Learning. Penelitian yang dilakukan oleh [2] yang berjudul "Arsitektur Microservice untuk Resiliensi Sistem Informasi". Didalam penelitian tersebut dijelaskan bahwa dengan menggunakan Microservices hasil pengujian pada sistem dengan arsitektur yang diusulkan telah menunjukkan peningkatan kualitas pada aspek resiliensi, misalkan ketika beberapa service mengalami gangguan, sistem dapat tetap berjalan sebagaimana mestinya. penelitian yang dilakukan oleh [3] yang berjudul "Implementasi Arsitektur Microservices pada Rancang Bangun Aplikasi Marketplace Berbasis Web". Aplikasi tersebut menggunakan arsitektur microservices telah berhasil dibangun dengan hasil persentase pengujian black box 100% dari 25 aktivitas berhasil dan hasil persentase pengujian endpoint api 100% dari 29 aktivitas berhasil.

2.2 Definisi

Paragraf ini akan menjelaskan tentang definisi apa saja yang digunakan untuk membuat sebuah microservices pada sistem.

2.2.1 Microservices

Microservices adalah kumpulan layanan yang saling berkaitan untuk membangun sebuah sistem yang terstruktur, agar sistem tersebut berjalan dengan maksimal [1]. Komunikasi antar setiap service atau layanan nya menggunakan API (Application programming Interface), karena cocok sebagai perantara service yang saling berhubungan. Microservices berguna untuk memaksimalkan layanan dari setiap data yang digunakan dan mempermudah developer mengembangkan sistem tersebut.

2.2.2 Python

Python merupakan salah satu bahasa pemrograman yang banyak digunakan oleh perusahaaan besar maupun para developer untuk mengembangkan berbagai macam aplikasi berbasis desktop, web dan mobile [4]. Bahasa pemrograman Python menggunakan struktur orientasi objek yang bertujuan untuk memudahkan seorang programmer menulikan kode yang jelas, bagus dan logis untuk project skala kecil mapun besar.

2.2.3 Framework Flask

Framework Flask merupakan micro web framework yang menggunakan bahasa python [5]. Micro framework ini berguna untuk mempercepat pembangunan suatu aplikasi karena sudah ada struktur, library dan komponen lain nya untuk menunjang developer tanpa harus membuat aplikasi dari awal atau nol. Dikatakan micro framework karena masuk dalam salah satu kategori framework micro, Sehingga mengurangi ketergantungan library atau extensi dari luar dan dapat membuat kode core dengan sintaks yang sangat sederhana.

2.2.4 API

API merupakan sebuah sistem penyedia layanan yang berjalan pada sisi server [6][7]. Application Programming Interface (API) juga dapat diartikan sebagai perantara atau penghubung antara service database dengan interface aplikasi. API memudahkan developer untuk membangun aplikasi karena service dari data dipisah dan tidak dijadikan satu dengan aplikasi atau monolith, sehingga developer akan mudah untuk mengembangkan aplikasi tersebut meskipun banyak fitur yang sudah berjalan.

2.2.5 Firebase

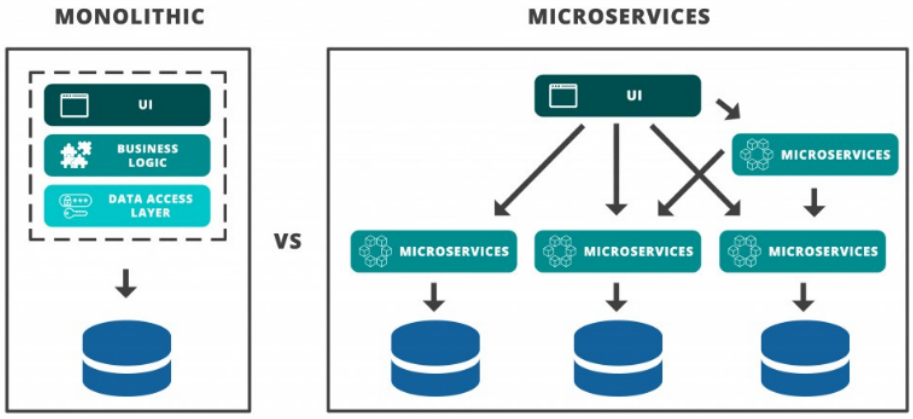
Firebase Database merupakan penyimpanan basis data nonSQL yang memungkinan untuk menyimpan beberapa tipe data [8]. Firebase memudahkan developer untuk menyimpan data karena berbasis BaaS (Backend as a Service), Sehingga dalam mengembangkan aplikasi developer tidak memberikan effort yang besar untuk urusan backend. Backend as a Service adalah kategori layanan cloud yang mengelola dibagian backend dan BaaS akan mengurusi segala hal mengenai backend seperti database, authentication, hosting, API dan lain lain.

2.2.6 Postman

Postman adalah sebuah aplikasi yang digunakan untuk menguji API yang telah dibuat oleh developer backend untuk menampilkan data yang dibutuhkan. Cara penggunaan dari postman sendiri yaitu request ke url yang akan kita tuju dan isi semua parameter yang sudah dibuat oleh developer, agar data yang muncul sesuai dengan yang anda inginkan.

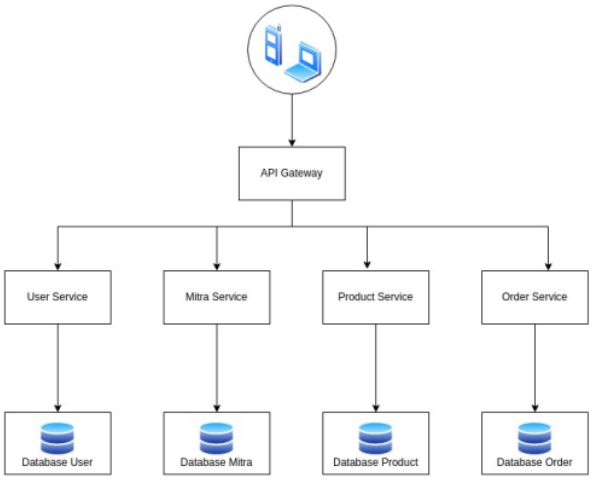
2.3 Design

Setelah definisi dari microservices, tahap selanjutnya adalah desain untuk microservices. Pada gambar dibawah menjelaskan antara monolit dan microservices yang berjalan, monolit merupakan model yang dibangun dalam satu codebase yang sama atau ui dan data nya dibuat pada halaman yang sama. Sedangkan microservices dibangun secara terpisah dari ui atau tampilan nya, dan berbagai services nya juga dibuat terpisah agar memaksimalkan performa dari request data yang akan dipanggil atau dipakai.



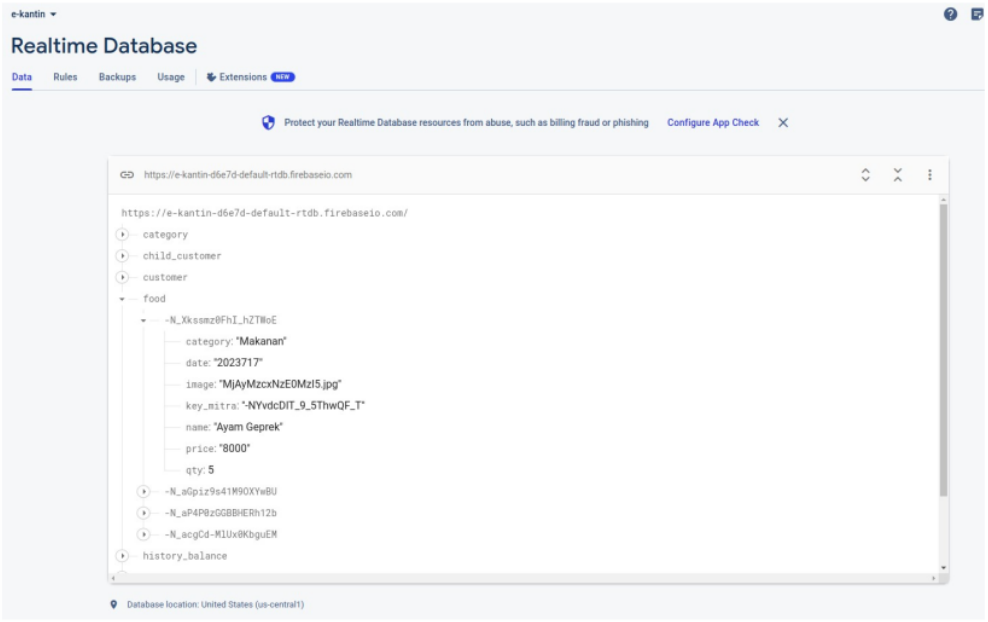
Gambar 2. Monolit dan Microservices

Pada gambar selanjutnya adalah alur dari setiap services nya mengakses atau mengkonsumsi database yang sudah dibuat.



Gambar 3. Alur Service Database

Selanjutnya adalah gambar dari tampilan database firebase yang digunakan. Pada firebase tampilan data yang masuk akan disimpan seperti JSON dan disinkronkan secara realtime ke setiap user penggunanya, yang memudahkan mengelola suatu data yang berskala besar.



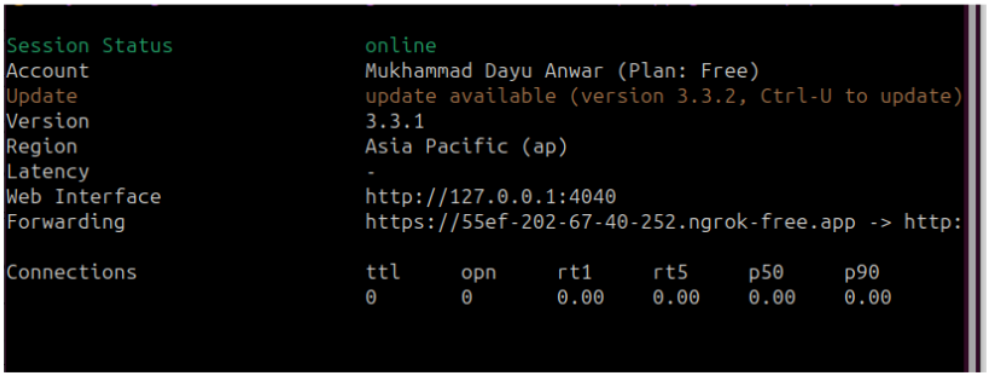
Gambar 4. Firebase

2.4 Testing

Untuk tahap testing API menggunakan aplikasi postman, karena untuk memastikan seluruh services yang dibuat telah berjalan sesuai yang diharapkan. Pada tahap ini sangat penting mengevaluasi dan menyesuaikan jika ada salah satu services yang tidak sesuai dengan apa yang telah disepakati.

2.5 Deploy

Dalam tahap deploy menggunakan aplikasi ngrok. Aplikasi ini cocok untuk membuat sebuah sistem agar bisa diakses secara online dan gratis, karena dapat membuka jaringan private melalui firewall dan menghubungkan localhost ke internet dengan tunel yang aman.



Gambar 5. Run Ngrok

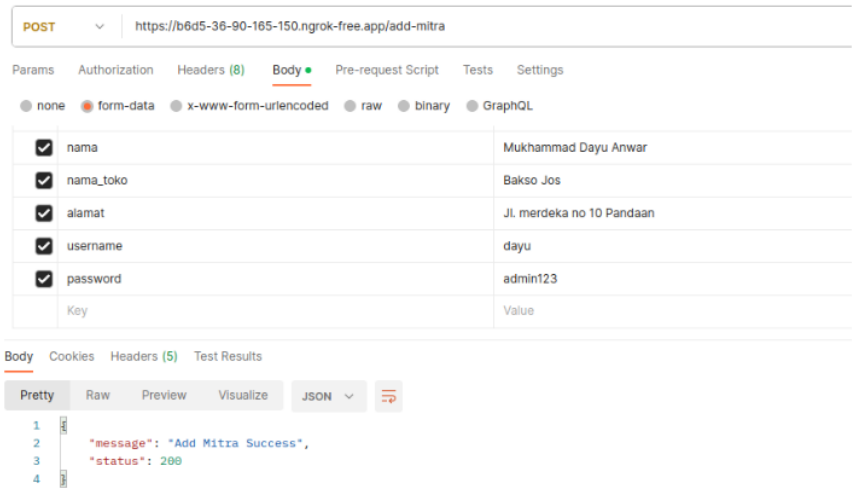
III. HASIL DAN PEMBAHASAN

3.1 Implementasi

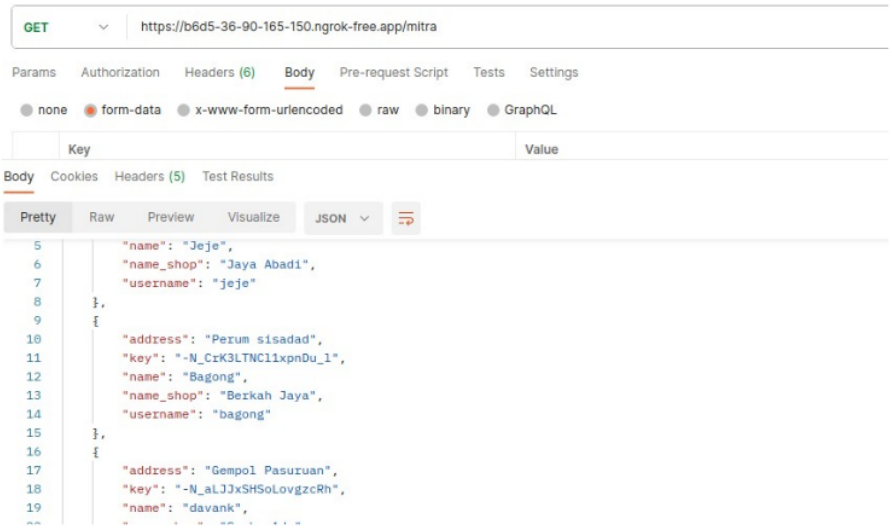
Pada bab ini akan menjelaskan mengenai hasil dan pembahasan dari API yang telah dibuat berdasarkan analisa yang telah dijelaskan diatas. Format data yang akan ditampilkan pada API dibawah adalah json, yang dimana format data ini kompatible dengan banyak bahasa pemrograman, environment dan library. Berikut ini adalah tampilan dari API yang telah dibuat dan diakses menggunakan postman.

3.1.1 API Mitra

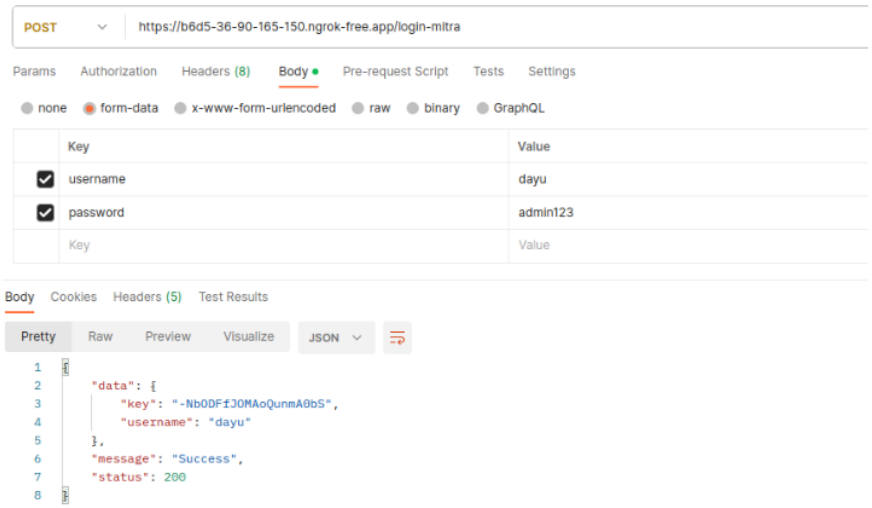
Pada API ini akan menampilkan data mitra, menambah mitra baru dan login untuk mitra.



Gambar 6. API Tambah Mitra



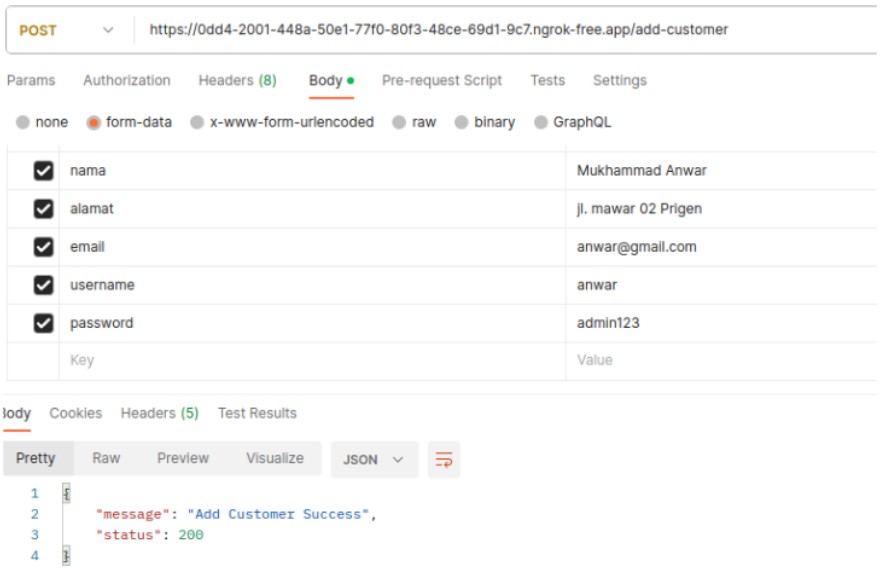
Gambar 7. API List Mitra



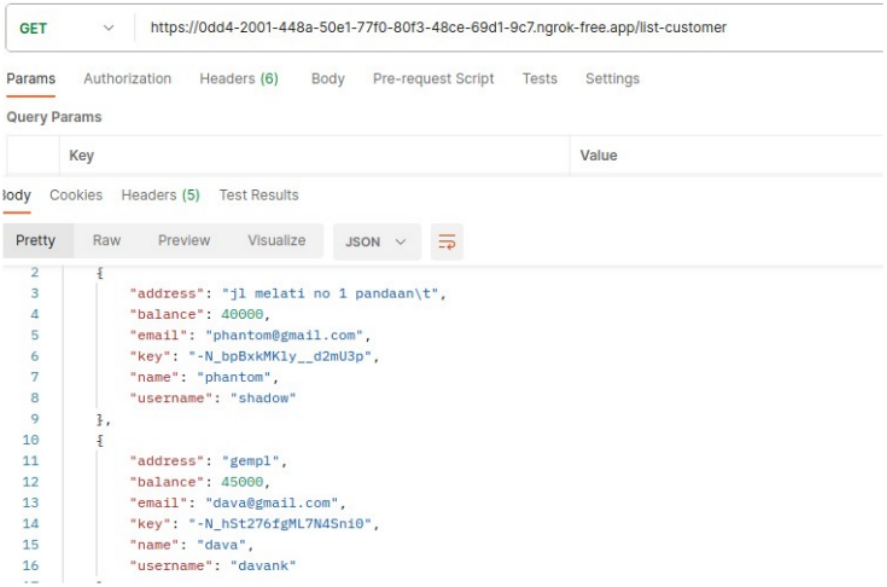
Gambar 8. API Login Mitra

3.1.2 API Customer

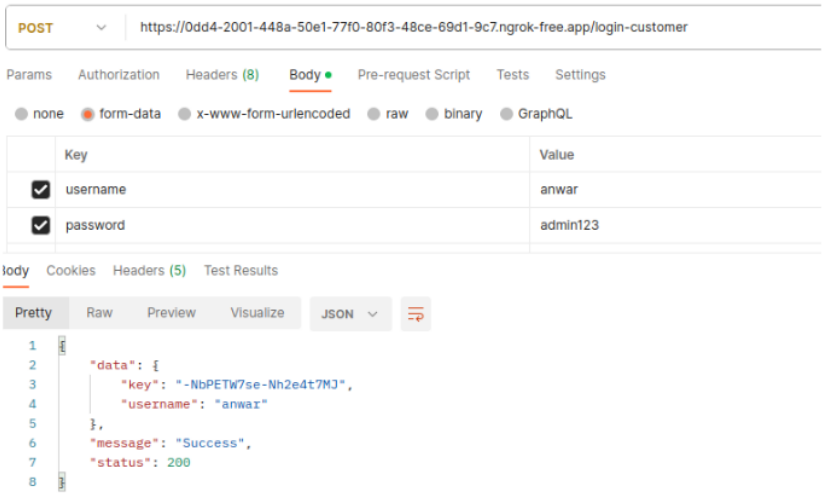
Dalam API ini bisa menambah customer, menampilkan semua customer dan login untuk customer.



Gambar 9. API Tambah Customer



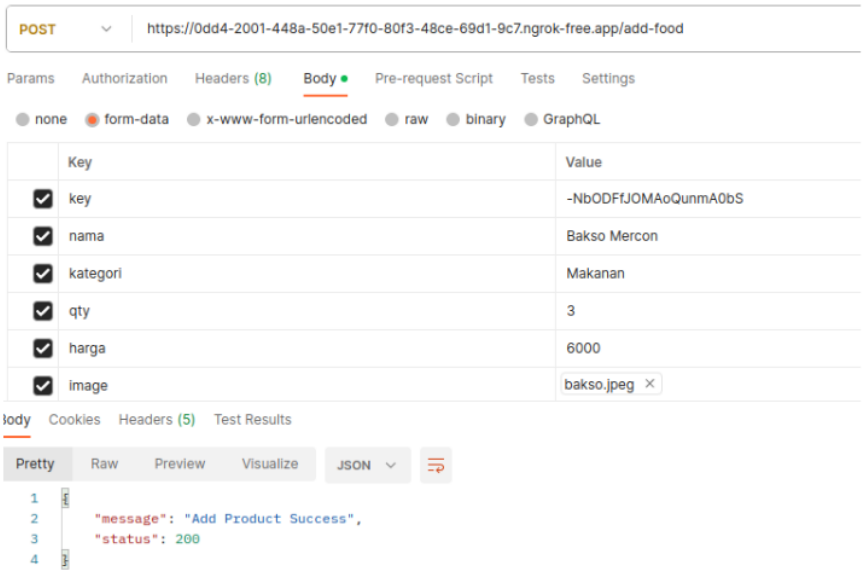
Gambar 10. API List Customer



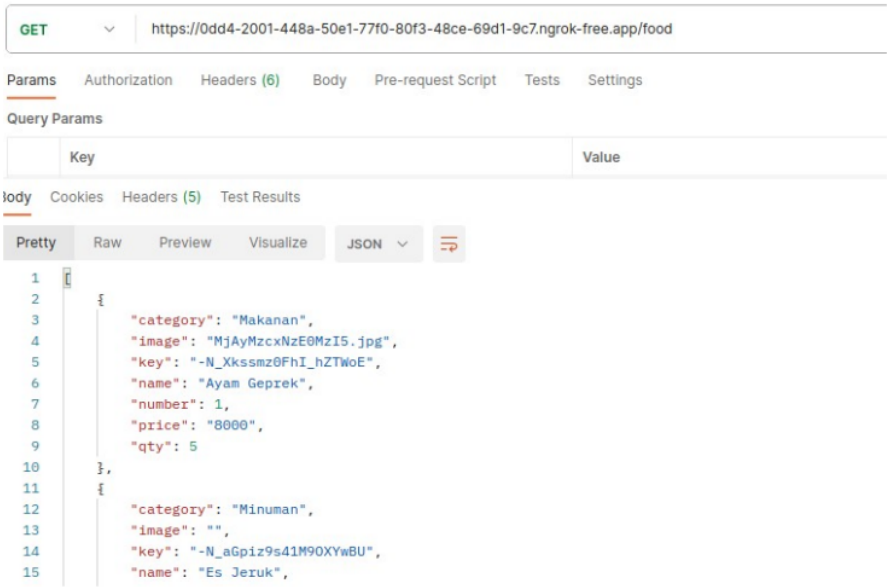
Gambar 11. API Login Customer

3.1.3 API Produk

Pada API ini menampilkan data makanan / minuman dan bisa menambahkan makanan / minuman.



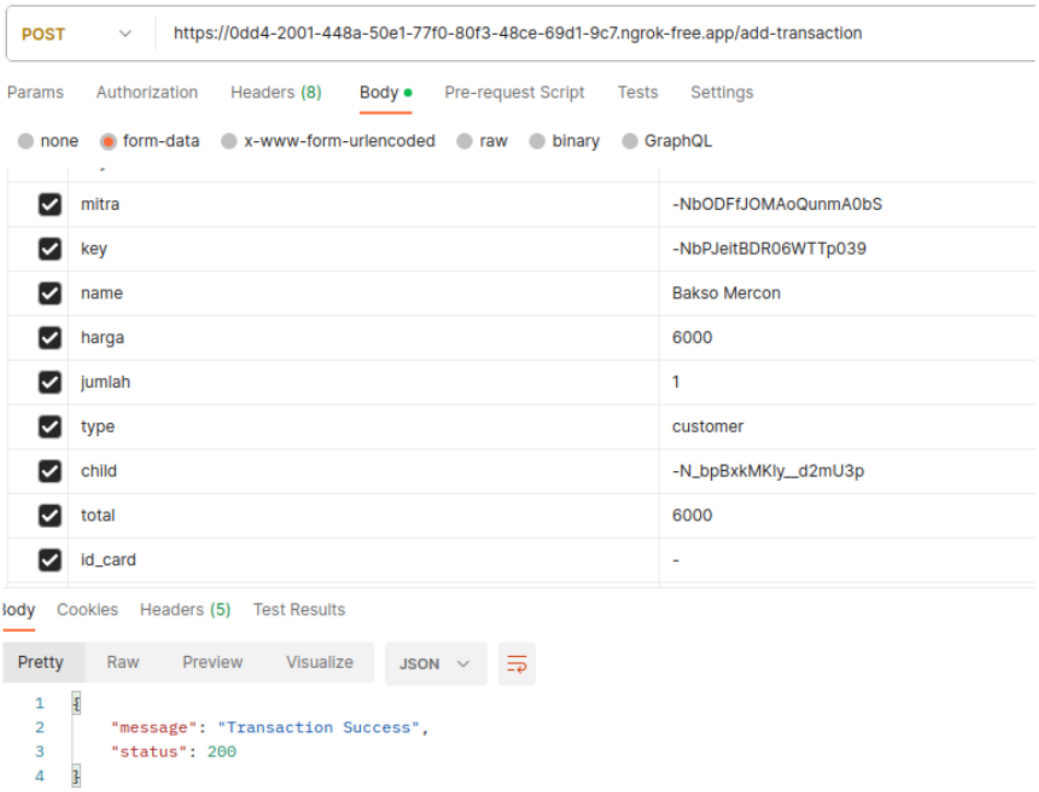
Gambar 12. API Tambah Produk



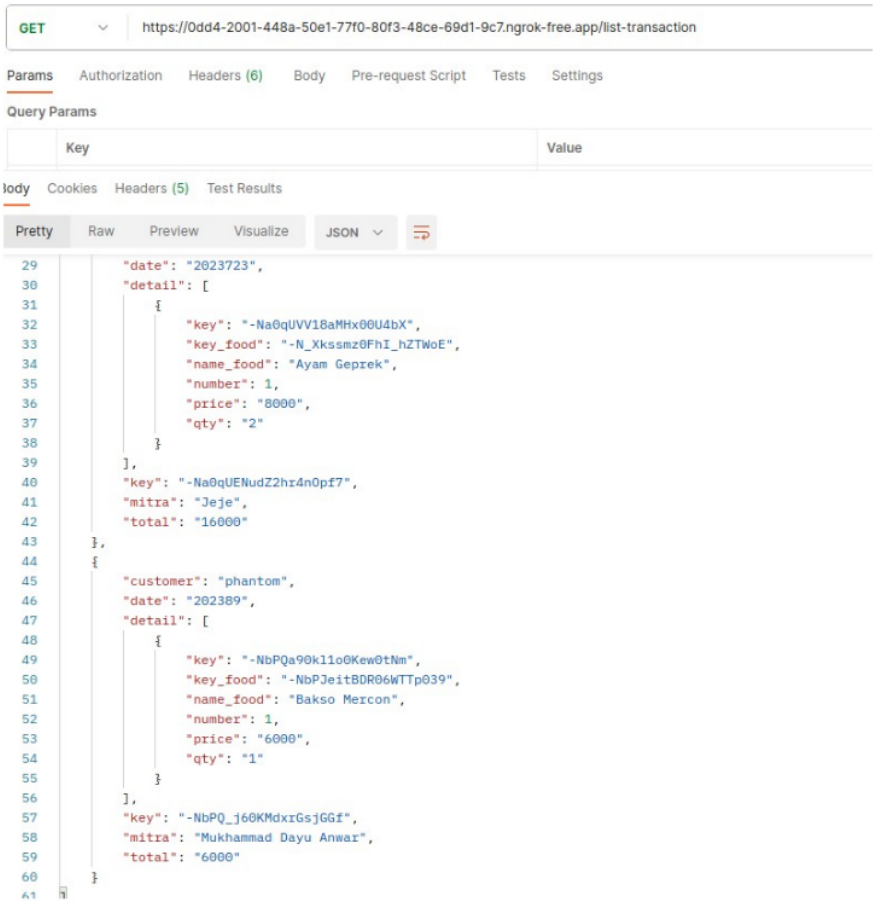
Gambar 13. API List Produk

3.1.4 API Transaksi

Pada API ini bisa menambah transaksi dengan mengisi parameter mitra, customer dan produk agar transaksi berhasil, selain itu juga ada list data transaksi.



Gambar 14. API Transaksi



Gambar 15. List Transaksi Berhasil

3.2 Pengujian Services

Dari pengujian API menggunakan postman diatas, diketahui bahwa setiap services nya berjalan dengan lancar dan sesuai dengan kebutuhan yang sudah direncanakan. Pada tabel dibawah ini akan menjelaskan bahwa pengujian terhadap endpoint yang ada pada setiap services nya telah berhasil dibuat dan dijalankan tanpa ada masalah.

Tabel 1. Pengujian endpoint / services API menggunakan postman

No	Skenario Pengujian	Hasil Yang Diharapkan	Hasil Pengujian
1.	API Tambah Mitra	Menampilkan data json yang berhasil	[√] Berhasil [ ] Tidak Berhasil
2.	API List Mitra	Menampilkan list mitra yang sudah dimasukkan	[√] Berhasil [ ] Tidak Berhasil
3.	API Login Mitra	Menampilkan json berhasil dan data login	[√] Berhasil [ ] Tidak Berhasil
4.	API Tambah Customer	Menampilkan data json yang berhasil	[√] Berhasil [ ] Tidak Berhasil

5.	API List Customer	Menampilkan list customer yang sudah dimasukkan	[√] Berhasil [ ] Tidak Berhasil
6.	API Login Customer	Menampilkan json berhasil dan data login	[√] Berhasil [ ] Tidak Berhasil
7.	API Tambah Produk	Menampilkan data json yang berhasil	[√] Berhasil [ ] Tidak Berhasil
8.	API List Produk	Menampilkan list produk yang sudah dimasukkan	[√] Berhasil [ ] Tidak Berhasil
9.	API Transaksi	Menampilkan data json yang berhasil	[√] Berhasil [ ] Tidak Berhasil
10.	API List Transaksi Berhasil	Menampilkan list transkasi yang berhasil	[√] Berhasil [ ] Tidak Berhasil

IV. Simpulan

Berdasarkan penelitian diatas disimpulkan bahwa microservices yang telah dibuat dapat membantu memaksimalkan performa data yang akan dikonsumsi oleh user pengguna. Selain itu untuk pengembangan dengan mengikuti bahasa pemrograman / framework yang terbaru dan terpopuler akan sangat mudah, karena struktur dan setiap servicesnya mudah untuk diubah dan ditambah. Dari sisi Frontend / Tampilan pun mudah dikonsumsi dan meskipun services nya berubah tidak harus mengubah semua tampilan yang ada, hanya menyesuaikan services yang diubah saja.

DAFTAR PUSTAKA

[1] Irwan Alnarus Kautsar, M. Ruslianor Maika, Agoes Nur Budiman, Arik Bagus Setyawan, Jagad Yudha Awali, “Microservice Based Architecture: The Development of Rapid Prototyping Supportive Tools for Project Based Learning,” 22963954, April 2023, 10.1109/EDUNINE57531.2023.10102884.

[2] Hatma Suryotrisongko, “Arsitektur Microservice untuk Resiliensi Sistem Informasi,” Jurnal Sisfo., vol. 06, no. 02, pp. 235–250, Jan. 2017.

[3] Alessandro Sinambela, Ernawati , Funny Farady Coastera, “Implementasi Arsitektur Microservices pada Rancang Bangun Aplikasi Marketplace Berbasis Web,” Jurnal Rekursif., vol. 9, no. 1, Mar. 2021.

[4] Schuerer, Katja dan Corinne Maufrais (2010). “Introduction to Programming using Python,” Boston: Pearson, hal. 1–242. ISBN: 0132747189.

[5] D. Ghimire, “Comparative study on Python web frameworks: Flask and Django,” no. May, pp. 1–40, 2020, [Online]. Available: http://www.theseus.fi/handle/10024/339796.

[6] Xu, Hao, Hong Wang, and Suo Zhang. "Application programming interface (api) service apparatus and application programming interface (api) service system." U.S. Patent Application No. 16/181,927.

[7] Rulloh, Amin, Dewi Erla Mahmudah, and Herman Kabetta. "Implementasi REST API pada Aplikasi Panduan Kepaskibraan Berbasis Android." Teknikom: Teknologi Informasi, Ilmu Komputer dan Manajemen 1.2 (2017): 8589.

[8] Luffi Aditya Sandy, Rizky Januar ,dan Ridho Rahman Hariadi, “Rancang Bangun Aplikasi Chat pada Platform Android dengan Media Input berupa Canvas dan Shareable Canvas untuk Bekerja Dalam Satu Canvas secara Online,” JURNAL TEKNIK ITS., vol. 6, no. 2, 2337-3520, 2017.

MICROSERVICE BASED SOFTWARE ARCHITECTURE ON CANTEEN INFORMATION MANAGEMENT SYSTEM

ORIGINALITY REPORT

0%	0%	0%	0%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

Exclude quotes	On	Exclude matches	< 10%
Exclude bibliography	On		