

## *Desain dan Implementasi REST-API Pada Website Marketplace Katering “Ketringan” Menggunakan Framework Laravel*

Alvin Lander<sup>1</sup>, Dody Qori Utama<sup>2</sup>, Aji Gautama Putrada<sup>3</sup>

<sup>1,2,3</sup> Universitas Telkom, Bandung

<sup>1</sup>alvinlander@student.telkomuniversity.ac.id, <sup>2</sup>dodyqori@telkomuniversity.ac.id,

<sup>3</sup>ajigps@telkomuniversity.ac.id

### ABSTRAK

Jasa Katering adalah salah satu jenis Usaha Mikro Kecil (UMK) yang dapat memesan makanan dalam jumlah besar untuk kebutuhan konsumsi suatu acara. Saat ini, proses pemesanan masih dilakukan secara manual. Customer harus mencari dan datang ke tempat untuk mendapatkan jasa yang cocok. Hal tersebut dapat memakan waktu dan menguras tenaga. Maka, diperlukan suatu teknologi untuk memangkas hal tersebut. Saat ini, marketplace sangat membantu bagi pengusaha UMK untuk memasarkan produknya. Maka, telah dibuat marketplace untuk jasa katering “Ketringan” berbasis website yang mengimplementasikan REST-API dan menggunakan framework Laravel. Aplikasi ini dapat memangkas aktivitas dari mencari informasi mengenai jasa katering hingga proses transaksi. Telah dilakukan uji fungsionalitas dan non-fungsionalitas sehingga aplikasi ini dapat berjalan sesuai fungsionalitasnya.

**Kata kunci :** Marketplace, Laravel, Katering ,REST-API.

### ABSTRACT

*Catering Services is one type of Micro Small Business (UMK) that can order food in large quantities for the consumption needs of an event. Currently, the ordering process is still done manually. Customers have to find and come to the place to get a suitable service. This can be time consuming and exhausting. So, we need technology to curtail it. Currently, the marketplace is very helpful for Micro, Small, and Medium Enterprises (MSE) entrepreneurs to market their products. Thus, a website-based marketplace for “Ketringan” catering services has been created that implements the REST-API and uses the Laravel framework. This application can curtail activities from finding information about catering services to processing transactions. Functionality and non-functionality tests have been carried out so that this application can run according to its functionality.*

**Keywords:** Marketplace, Laravel, Catering , REST-API.

## 1. Pendahuluan

Indonesia mempunyai potensi basis ekonomi nasional yang kuat karena jumlah Usaha Mikro Kecil (UMK) terutama di bidang Akomodasi dan Penyediaan Makan Minum[1]. Salah satu UMK yang sering ditemui adalah usaha jasa katering. Katering adalah jenis layanan makanan yang dipesan dalam jumlah besar dengan hidangan yang sama, biasanya untuk suatu acara seperti rapat, pesta, atau jamuan[2]. Saat ini pemesanan jasa katering masih dilakukan secara manual. Pembeli harus mencari dan datang ke tempat untuk mendapatkan jasa katering yang cocok. Hal tersebut dapat memakan waktu dan menguras tenaga. Sehingga, diperlukan suatu teknologi untuk membuatnya terotomatisasi.

Saat ini, *marketplace* sangat membantu bagi pengusaha UMK untuk memasarkan produknya [3]Marketplace dapat diakses melalui *website* dan aplikasi *mobile*. Sehingga mempermudah *user* untuk memesan produk yang ingin dibelinya. Dalam hal ini, membuat *marketplace* berbasis *website* cocok untuk digunakan oleh penggiat UMK jasa katering. *User* tidak harus menginstal aplikasi *mobile* untuk memesan jasa katering.

Dalam penelitian ini, telah didesain dan di implementasi REST-API pada *website marketplace* katering “Ketringan” menggunakan *framework* Laravel dan telah diuji. Dilakukan dua pengujian yaitu, Pengujian fungsional dan non-fungsional. Pengujian fungsional berfokus pada kesesuaian fitur yang telah dibuat dan non-fungsional.

## 2. Pembahasan

### 2.1 Perspektif produk

Perangkat lunak ini merupakan API yang menggunakan arsitektur REST. REST adalah salah satu implementasi API yang memanfaatkan protokol HTTP untuk berkomunikasi, yang sering digunakan sebagai web service[4]. REST-API yang dibangun untuk memenuhi proses bisnis marketplace website Ketringan. REST-API ini dibangun dengan tujuan agar dapat digunakan untuk berkomunikasi antar aplikasi. Dengan begitu, cukup membangun sebuah REST-API dapat digunakan oleh berbagai aplikasi sehingga dapat mempermudah *developer* untuk mengontrolnya. REST-API akan menerima *request* dan memberikan *response* berupa format JSON. Javascript Object Notation (JSON) merupakan format yang sangat umum digunakan untuk pertukaran data karena ringan dan mudah dipahami oleh manusia dan mudah diurai oleh komputer[5].

Untuk dapat mengakses REST-API, *client* harus mengirim request terlebih dahulu kepada web *service* agar bisa mendapatkan informasi yang diinginkan. Bentuk *request client* terdiri dari alamat *endpoint*, HTTP *method* (GET, POST, PUT, DELETE), *headers*, dan *body request* sebagai parameter.

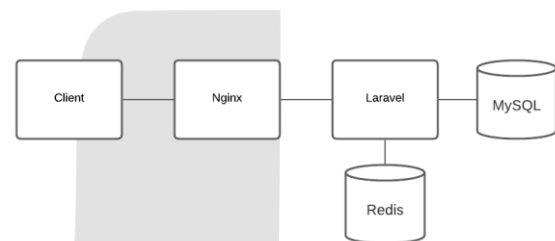
### 2.2 Fitur Sistem

Fitur utama dari *web service* yang dibuat untuk mendukung semua proses bisnis *marketplace website* “Ketringan”. Yang terdiri dari *login user*, registrasi *user*, *forgot password user*, melihat menu, menambahkan menu ke keranjang, melihat diskon, melakukan pemesanan, melihat daftar transaksi, mengunggah bukti pembayaran, dan ubah data user. Beberapa API akan membutuhkan “*Authorization Token*” agar dapat mengaksesnya yang didapat berdasarkan *user* yang *login*.

## 2.3 Perancangan

### 2.3.1 Rancangan Perangkat Lunak

Perangkat lunak dibangun dengan menerapkan arsitektur REST-API yang dibuat menggunakan *framework* Laravel 8. Berikut ilustrasi rancangan arsitektur perangkat lunak yang akan dibangun.



Gambar 1 Arsitektur Aplikasi Web Service

Pada gambar 1, *client* akan mengirim *request* pada web *service* dan jika *request* valid maka web *service* akan merespon dengan data yang di *request* oleh *client* dalam bentuk JSON. Berikut bentuk struktur *response* yang akan diberikan kepada *client*:

```

{
  "meta": {
    "code": 200,
    "status": "success",
    "message": "Register Success with tp 36.72.88.50"
  },
  "data": {
    ...
  }
}

```

Gambar 2 Struktur Response REST-API

Pada gambar 2, merupakan struktur response yang akan diberikan kepada *client*. Ada 4 kata kunci utama yang harus ada dalam *response* seperti *response code*, *status*, *message*, dan *data*.

Pengembangan REST-API ini menggunakan *framework* Laravel 8 yang menggunakan bahasa pemrograman PHP 7.4. Laravel memiliki ekosistem yang sangat lengkap yang dapat mempercepat dalam pengembangan REST-API ini. Untuk teknologi penyimpanan data menggunakan Relational Database Management System (RDBMS) MySQL dikarenakan

memiliki keunggulan dalam segi performa yang baik, murah, mudah diatur, dan dapat diandalkan. Untuk meningkatkan performa REST-API yang lebih baik penulis menambahkan NoSQL Redis digunakan untuk melakukan teknik antrian agar web service tidak perlu menunggu proses pengolahan data hingga selesai dan dapat memberikan respon lebih cepat kepada client. Agar web service dapat berjalan lancar maka memerlukan web service. Nginx dipilih karena memiliki fitur load balancer agar dapat membagi beban traffic.

Berikut tech-stack yang digunakan penulis untuk membangun perangkat lunak ini yang terdapat pada tabel 1.

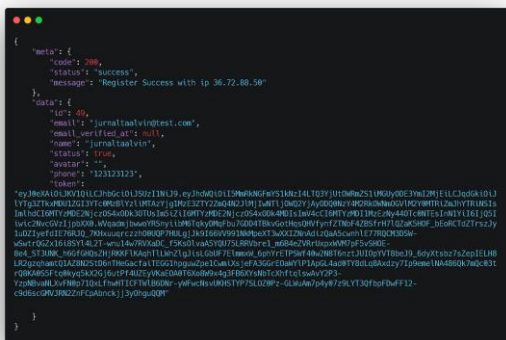
Table 1 Tech-stack yang digunakan untuk web service

Teknologi	Keterangan
PHP 7.4	Bahasa pemrograman yang digunakan.
Laravel 8	Untuk membangun kerangka kerja untuk membangun website.
MySQL	RDBMS yang digunakan untuk menyimpan data
Redis	Untuk melakukan Teknik queueing
Nginx	Web server yang digunakan

## 2.4 Implementasi

### 2.4.1 Implementasi Rancangan Perangkat Lunak

Penelitian ini dilakukan dengan menggunakan metode SCRUM, yang memecah beberapa masalah kompleks menjadi aktivitas backlog dan mengeksekusinya dalam waktu sprint. Aktivitas tersebut kemudian disesuaikan menjadi pull request (PR) di platform Github, setelah itu PR ditinjau oleh anggota lain dan PR digabungkan ke cabang utama sesuai kebutuhan. Proses pengerjaan perangkat lunak ini membutuhkan waktu 7 kali sprint planing. Hasil keluaran dari perangkat lunak ini adalah struktur data menggunakan format JSON. Berikut response yang diberikan web service “Ketringan” yang terdapat pada gambar 3 ketika user melakukan login user



Gambar 3 Tangkapan layar response user login

### 2.4.4 Kesimpulan

Implementasi REST-API ini berjalan dengan baik dan tidak memiliki kendala ketika client melakukan request. Response yang diberikan juga sudah cukup baik berdasarkan beberapa user yang telah mengujinya. Meskipun begitu penulis akan terus mengembangkan, meningkatkan dan menambah fitur baru agar proses bisnis “Ketringan” dapat berjalan lancar.

Untuk peningkatan selanjutnya perlu adanya keamanan yang lebih aman, dan efisiensi kode yang lebih baik. Saat ini karena keterbatasan developer dan waktu oleh karena itu, penulis akan terus mengembangkan.

## 2.5 Pengujian Sistem

### 2.5.1 Pengujian Fungsional

Pengujian Fungsional dilakukan pada setiap endpoint yang ada pada web service untuk melihat apakah layak digunakan atau tidak. Skenario pengujian dengan hipotesa apabila response time rata-rata dari 10 kali request di bawah 2 detik maka REST-API tersebut sudah baik dan layak untuk digunakan. Pengujian ini dilakukan langsung pada server deployment dengan spesifikasi perangkat keras seperti yang tertera pada tabel 2.

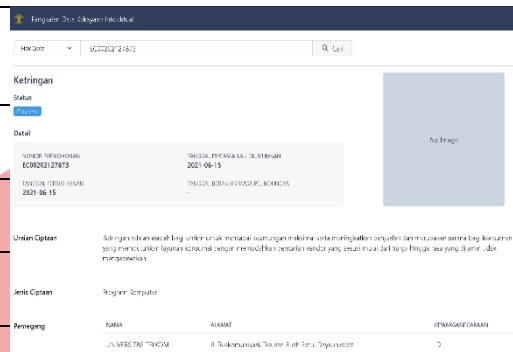
Table 2 Spesifikasi Perangkat Keras Untuk Pengujian

Sistem Operasi	Linux CentOS
RAM	1 GB
CPU	2 Core
SSD Storage	30 GB

Table 3 Hasil Pengujian Fungsional Berdasarkan Response Time

Endpoint	Response Time	Hasil Pengujian	Keterangan
User Login	0.7s	Pass	Lulus Pengujian
User Register	0.6s	Pass	Lulus Pengujian
Ubah Profile User	0.5s	Pass	Lulus Pengujian
Melihat Menu	1.2s	Pass	Lulus Pengujian
Melihat diskon	0.5s	Pass	Lulus Pengujian

Menambah menu ke keranjang	0.5s	Pass	Lulus Pengujian
Checkout pesanan	1.4s	Pass	Lulus Pengujian
Melihat pesanan	0.7s	Pass	Lulus Pengujian
Lupa password	0.5s	Pass	Lulus Pengujian



Gambar 5 Bukti HKI Telah Terdaftar

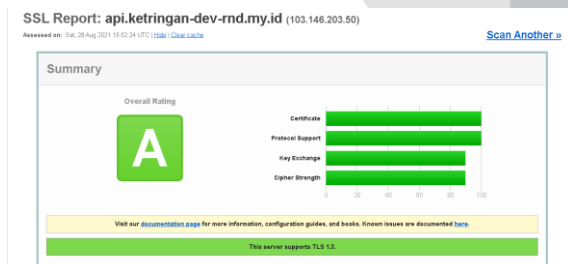
Berdasarkan hasil pada tabel 3, Hasil yang dapat disimpulkan adalah semua REST-API yang telah dibuat telah lulus dalam pengujian dan dapat digunakan oleh client.

**2.5.2 Pengujian Non-Fungsional**

Pada pengujian non-fungsional ini guna untuk menguji keamanan dari endpoint yang telah dibuat menggunakan tool SSL Lab Test untuk protokol HTTPS dan pendaftaran hak cipta.

**a. Protokol HTTPS untuk keamanan sistem**

Protokol HTTPS diuji untuk memastikan bahwa pertukaran data dari aplikasi dienkripsi. Pengujian dilakukan di situs web ssllabs.com dan memberikan hasil yang sangat baik dengan perincian berikut pada gambar 4.



Gambar 4 Hasil pengujian Sertifikat SSL untuk web service Ketringan.

**b. Pendaftaran Hak Paten**

Aplikasi Ketringan telah didaftarkan hak patennya sesuai dengan Pasal 72 Undang-Undang Nomor 28 Tahun 2014 tentang Hak Cipta dengan nomor pendaftaran: EC00202127873, 15 Juni 2020 dan berlaku selama 50 (lima puluh) tahun sejak Ciptaan pertama kali dilakukan pengumuman. Bukti terdaftar HKI dapat dilihat pada gambar 5.

**Referensi**

- [1] D. M. Haryanti, "UMKM Indonesia," 24 07 2018. [Online]. Available: <https://www.ukmindonesia.id/baca-artikel/62>. [Diakses 08 28 2021].
- [2] P. A. Prabowo, "ANALISA PERILAKU KONSUMEN JASA KATERING UNTUK KEPERLUAN PESTA DI SURABAYA," *Jurnal Ekbis UNISLA*, vol. 21, no. 1, 2020.
- [3] T. P. I Putu Artaya, "EFEKTIFITAS MARKETPLACE DALAM MENINGKATKAN KONSENTRASI," 2019.
- [4] M. Masse, *REST API Design Rulebook: Designing Consistent RESTful Web Service Interfaces*, O'Reilly Media, Inc, 2011.
- [5] M. P. R. R. C. I. Nurzhan Nurseitov, "Comparison of JSON and XML Data Interchange Formats: A Case Study," pp. 157-162, 2009.