

## **Pemanfaatan *Application Programming Interface* Midtrans dan Raja Ongkir Untuk Membangun Enterprise Application Integration**

**I Ketut Widhi Adnyana<sup>1\*</sup>, I Gede Made Yudi Antara<sup>2</sup>, Dewa Ayu Putri Wulandari<sup>3</sup>**

<sup>1</sup>Sistem Informasi, Institut Teknologi dan Bisnis STIKOM Bali

<sup>2,3</sup>Teknik Informatika, STMIK STIKOM Indonesia

<sup>1</sup> Jl. Raya Puputan No.86, Daging Puri Klod, Bali, Telp (0361)244445

<sup>2,3</sup> Jl. Tukad Pakerisan No.97, Panjer, Kec. Denpasar Sel., Kota Denpasar, Bali 80225,  
 Telp (0361) 256995

*\*Corresponding Author:* ketut.widhi55@gmail.com

### **Abstrak**

*E-Commerce* dapat menghubungkan penjual dan pembeli yang berbeda tempat yang tidak mengharuskan penjual dan pembeli bertemu secara langsung layaknya transaksi tradisional. Masalah yang biasa terjadi pada sisi pemilik yaitu menghitung biaya kirim dan melakukan pengecekan secara manual untuk memastikan barang yang dipesan sudah di bayar. Perlu adanya pengembangan suatu aplikasi *e-commerce* yang terintegrasi dengan sistem penghitungan biaya kirim dan *payment gateway* yang dapat mempermudah konsumen untuk mendapatkan informasi tentang produk yang diinginkan, serta dapat melakukan transaksi dari mana saja secara *online*. Dengan menerapkan *Application Programming Interface* (API) Midtrans sebagai *payment gateway* diharapkan pemilik usaha tidak perlu lagi melakukan pengecekan biaya kirim dan melakukan pengecekan pembayaran secara manual untuk menghitung biaya kirim ke berbagai wilayah di Indonesia. Metode yang digunakan dalam pengembangan aplikasi adalah metode *prototyping*, sebuah metode yang dikembangkan dengan berkomunikasi kepada *client* agar aplikasi yang dihasilkan sesuai dengan keinginan *client*. Hasil uji pencocokan biaya pengiriman dan total biaya yang harus dibayar oleh pelanggan menunjukkan API dari midtrans dapat membantu pelanggan melakukan pembayaran melalui *channel* yang bekerjasama dengan midtrans.

**Kata kunci:** *E-commerce, Application Programming Interface, Midtrans, Raja Ongkir*

### **Abstract**

*E-Commerce* can connect sellers and buyers from different places that do not require sellers and buyers to meet in person like traditional transactions. The problem that usually occurs on the owner's side is calculating shipping costs and checking manually to make sure the goods ordered have been paid for. It is necessary to develop an *e-commerce* application that is integrated with a shipping cost calculation system and a *payment gateway* that can make it easier for consumers to get information about the desired product, and can make transactions from anywhere online. By implementing Midtrans *Application Programming Interface* (API) as a *payment gateway*, it is hoped that business owners will no longer need to check shipping costs and check payments manually to calculate shipping costs to various regions in Indonesia. The method used in application development is the *prototype* method, a method developed by communicating to the *client* so that the resulting application is in accordance with the *client's* wishes. The test results of matching shipping costs and total costs to be paid by customers show that Midtrans API can help customers make payments through channels that work with Midtrans.

**Keywords:** *E-commerce, Application Programming Interface, Midtrans, Raja Ongkir*

## 1. Pendahuluan

Pada saat sekarang konsumen lebih suka membeli sesuatu secara *online* yang dianggap lebih praktis dari pada harus langsung ke toko, karena di toko online konsumen dapat melakukan pembelian dirumah atau di mana pun mereka berada, konsumen bisa melihat dan memilih produk-produk yang ditawarkan pada layar komputer atau *handphone*, mengakses informasinya, memesan dan membayar dengan pilihan yang tersedia. Konsumen dapat menghemat waktu, dan tenaga karena tidak perlu datang langsung ke toko tersebut.

Selama ini sistem penjualan masih menggunakan media promosi melalui *fanspage Facebook* dan *Instagram*, selain itu pemesanan secara online dilakukan dengan menghubungi langsung melalui *WhatsApp* yang masih banyak kekurangan. Data penjualan memperlihatkan rata-rata tingkat penjualan yang bervariasi dan umumnya memperlihatkan angka di bawah rata-rata target yang menjadi harapan dari manajemen toko, oleh karena itu pihak manajemen harus mencari peluang untuk mengantisipasi makin menurunnya omset dan volume penjualan. Kesulitan yang dialami ketika pemesanan online yang dilakukan melalui *WhatsApp* yaitu penentuan ongkos kirim dan konsumen harus menunggu tanggapan dari pihak pemilik toko [1].

*E-Commerce* adalah salah satu metode dalam berbisnis melalui internet. *E-Commerce* dapat menghubungkan penjual dan pembeli yang berbeda tempat yang tidak mengharuskan penjual dan pembeli bertemu secara langsung layaknya transaksi tradisional. Calon pembeli dapat mengetahui info mengenai produk seperti harga, model, warna dan sebagainya. Berdasarkan wawancara yang dilakukan kepada pelaku bisnis *online*, mereka membenarkan bahwa terdapat masalah yang cukup merepotkan yang berkaitan dengan menghitung biaya pengiriman barang yang di pesan dan melakukan pengecekan bahwa barang yang dipesan sudah di bayar atau di transfer harus secara manual.

Penggunaan *website* saat ini tidak hanya sebatas portal mencari informasi, namun juga dapat digunakan untuk penerapan sistem informasi penjualan elektronik yang terintegrasi dengan sistem lain, yang sering disebut *E-Commerce*. Sistem *E-Commerce* sendiri telah meluas digunakan sebagai media promosi dan pemasaran produk, seperti dalam [2],[3], [4] dan [5].

Artikel ini menyajikan sebuah model aplikasi *e-commerce* yang terintegrasi dengan sistem penghitungan biaya kirim dan *payment gateway* yang dapat mempermudah konsumen untuk mendapatkan informasi tentang produk yang diinginkan serta dapat melakukan transaksi dari mana saja secara *online*. Sistem juga dapat mempercepat proses validasi pembayaran yang tidak perlu dilakukan secara manual oleh pemilik toko. Integrasi antar sistem membentuk sebuah EAI (*Enterprise Application Integration*).

## 2. Tinjauan Pustaka

Menurut [6], *E-commerce* adalah penggunaan internet, web, dan aplikasi untuk bertransaksi bisnis. Dengan bahasa lebih formal, transaksi komersial antara organisasi dan individu secara digital. Terdapat beberapa fitur unik yang dimiliki oleh *e-commerce*:

- 1) *Global Reach*: Menjangkau seluruh customer yang dapat dijangkau oleh *e-commerce*.
- 2) *Universal Standards*: Memiliki standard yang diakui oleh seluruh bangsa di Indonesia
- 3) *Richness*: Memberikan informasi yang sangat kompleks dan beragam
- 4) *Interactivity*: Teknologi yang dapat mempermudah komunikasi dua arah antara pedagang dan customer
- 5) *Information Density*: Informasi yang diberikan dapat diakses oleh seluruh pengguna *e-commerce*, konsumen dan pedagang lainnya. Dalam *e-commerce* harga dan biaya lebih transparan
- 6) *Personalization Customization*: Pedagang dapat menentukan sendiri target pasar yang diinginkan
- 7) *Social Technology*: Dengan sosial media yang ada sekarang, pengguna dapat membuat komunitas yang baru atau mengembangkan komunitas yang sudah ada.

Menurut [6] *Enterprise Application Integration* atau disebut EAI, adalah proses mengintegrasikan/menghubungkan aplikasi dalam satu organisasi bersama dalam rangka untuk menyederhanakan dan mengotomasi proses bisnis seluas mungkin, sementara pada saat yang sama menghindari keharusan membuat perubahan besar terhadap aplikasi yang ada. Grup Gartner mengatakan "EAI adalah berbagi data tak terbatas dan menghubungkan aplikasi proses bisnis dalam suatu perusahaan". Salah satu tantangan besar EAI adalah adanya perbedaan kondisi lingkungan dari sistem yang perlu dihubungkan bersama. Perbedaan

tersebut mencakup: sistem operasi, solusi database, dan bahasa pemrograman yang digunakan

Penelitian mengenai E-Commerce telah dilakukan Calive dan Alexander [7] dengan judul Perancangan dan Pembuatan *Website E-Commerce* pada Toko Aksesoris Komputer di Surabaya. penelitian ini menghasilkan sebuah aplikasi berbasis website menjangkau *customer* yang jauh dari lokasi toko aksesoris komputer. Berdasarkan salah satu poin dari pertanyaan kuisisioner, maka dapat ditarik kesimpulan yaitu dalam upaya membantu menjangkau *customer*, website *e-commerce* mendapatkan nilai 70% baik

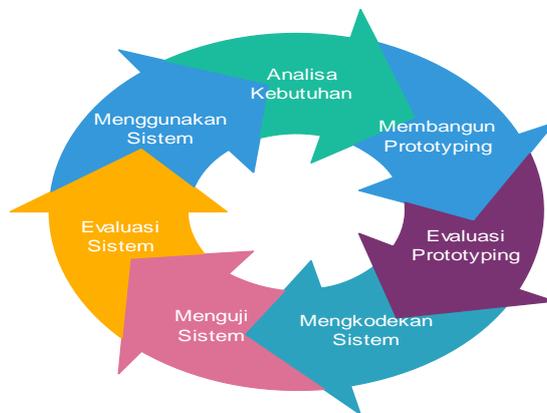
Penelitian Somya dan Esmeralda [8] yang berjudul Pengembangan Sistem Informasi Pelatihan Berbasis Web Menggunakan Teknologi *Web Service* dan *Framework Laravel*. penelitian ini menghasilkan penerapan teknologi web service pada fungsi *viewlistevent* dan *ticketcheckout* pada sistem ini mempermudah dalam proses integrasi data yang dapat dilakukan secara terpusat. Selain itu, penggunaan *framework Laravel* bermanfaat karena membuat sistem memiliki peforma yang cepat dalam memproses data serta ringan untuk dijalankan

Schwarz dan Dirk [9] mengintegrasikan data dalam ekosistem *enterprise* yang dimanfaatkan oleh *microservice*. *Microservice* dapat memanfaatkan *user interface* untuk mengintegrasikan data dan menyajikan hasilnya langsung kepada pengguna. Pendekatan lain yang disarankan adalah menggunakan panggilan layanan untuk mengambil data dan melakukan integrasi data di layanan mikro lain. Mengenai integrasi data melalui *middleware*, hanya teknologi yang menyediakan fungsi *publish-subscribe* yang direkomendasikan.

Penelitian ini menerapkan *Application Programming Interface* (API) Midtrans sebagai *payment gateway* untuk mempercepat proses validasi pembayaran dalam sistem penjualan.

### 3. Metodologi

Metode yang digunakan dalam pengembangan aplikasi adalah metode *prototyping*. *Prototyping* merupakan metode yang dikembangkan dengan berkomunikasi kepada *client* agar aplikasi yang dihasilkan sesuai dengan keinginan *client* [10]. Metode *prototyping* juga mempermudah pengembang perangkat lunak dalam membentuk model dari perangkat lunak yang harus dibuat.



Gambar 1. Fase Dalam Siklus *Prototyping*

#### Fase-Fase Metode Prototyping

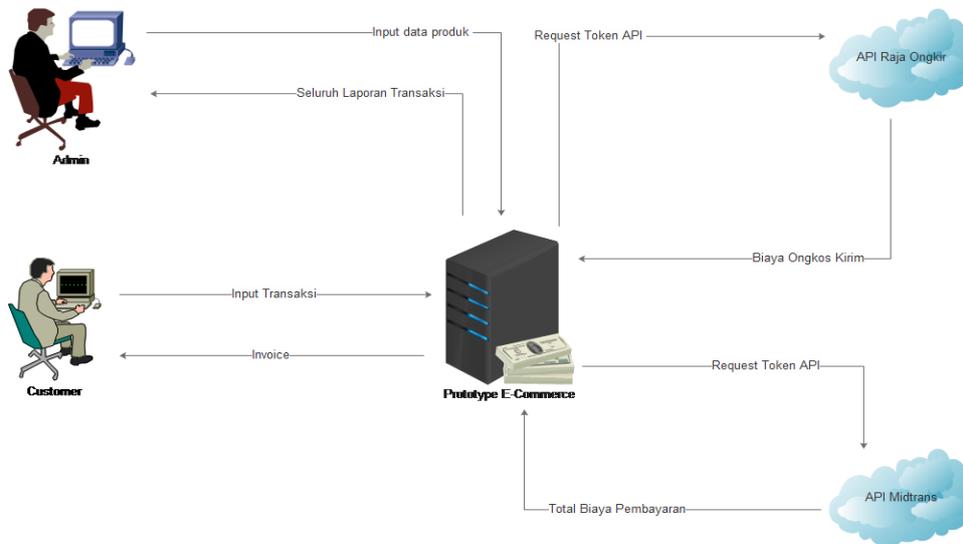
- 1) Analisa Kebutuhan  
Di tahap ini pengembang melakukan identifikasi *software* dan semua kebutuhan sistem yang akan dibuat.
- 2) Membangun Prototyping  
Membangun *prototyping* dengan membuat perancangan sementara yang berfokus pada penyajian kepada pelanggan (misalnya dengan membuat *input* dan format *output*).
- 3) Evaluasi Prototyping  
Evaluasi ini dilakukan untuk mengetahui apakah *prototyping* sudah sesuai dengan harapan pelanggan.
- 4) Mengkodekan Sistem

Pada tahap ini *prototyping* yang sudah disetujui akan diubah ke dalam bahasa pemrograman.

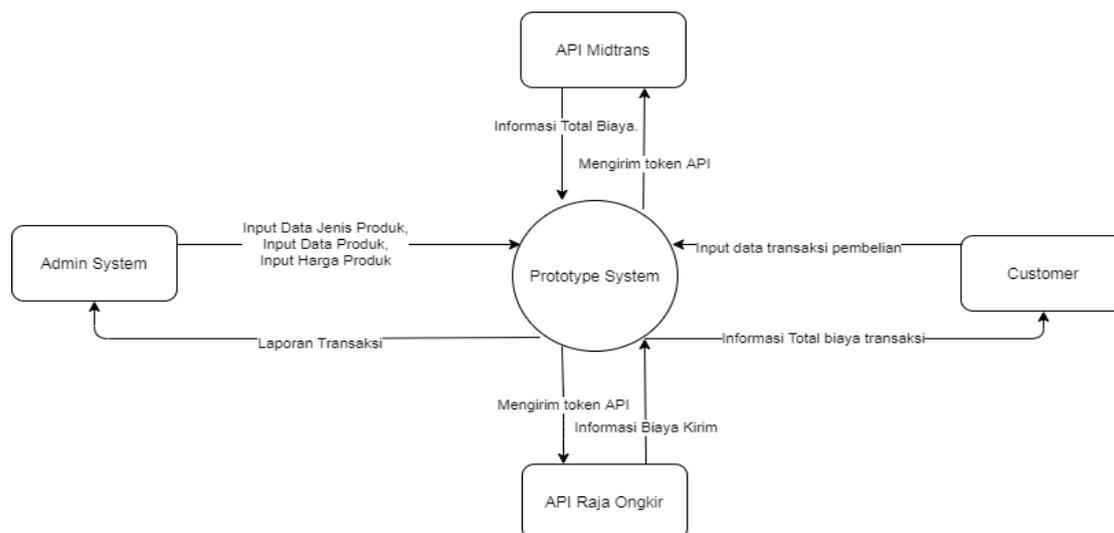
- 5) Menguji Sistem  
Di tahap ini dilakukan untuk menguji sistem perangkat lunak yang sudah dibuat.
- 6) Evaluasi Sistem  
Perangkat lunak yang sudah siap jadi akan dievaluasi oleh pelanggan untuk mengetahui apakah sistem sesuai dengan yang diharapkan.
- 7) Menggunakan Sistem  
Perangkat lunak yang sudah diuji dan disetujui oleh pelanggan siap digunakan.

#### 4. Hasil dan Pembahasan

Pada bagian ini dipaparkan hasil pengujian integrasi antara aplikasi *e-commerce* dengan *payment gateway* midtrans dan penentuan biaya biaya kirim dari raja ongkir dengan memanfaatkan API (*Application Programming Interface*). dengan ilustrasi seperti gambar 2:



Gambar 2. Arsitektur *Prototyping* Aplikasi



Gambar 3. Model Proses Sistem

Gambar 2 menggambarkan alur transaksi dari *prototype e-commerce*. Diawali dengan admin melakukan input data produk yang akan dijual kedalam sistem, tahap berikutnya adalah proses transaksi pembelian akan suatu barang yang di lakukan oleh customer dengan menginputkan data transaksi terhadap barang apa saja yang akan di beli. setelah dirasa barang

yang akan di beli sudah lengkap. maka tahap selanjutnya adalah menentukan alamat pengiriman barang tersebut. secara tidak langsung *prototype e-commerce* melakukan request token ke server raja ongkir untuk mendapatkan informasi biaya pengiriman barang sesuai tujuan yang di inputkan oleh *customer*, tahap selanjutnya adalah secara tidak langsung *prototype e-commerce* melakukan *request* token ke server midtrans untuk mendapatkan total biaya yang harus dibayarkan oleh customer serta pembayaran dapat dilakukan melalui apa saja.

Gambar 3 menggambarkan logika sederhana bagaimana *prototype* aplikasi bekerja sehingga mampu memberikan informasi total biaya transaksi termasuk biaya pengiriman barang kepada *customer*. Terdapat 4 entitas yang akan ber interaksi dengan *prototype* sistem yaitu entitas admin system, customer, api raja ongkir dan api midtrans masing-masing dari entitas tersebut memiliki perannya masing masing

**4.1. Metode Pengujian**

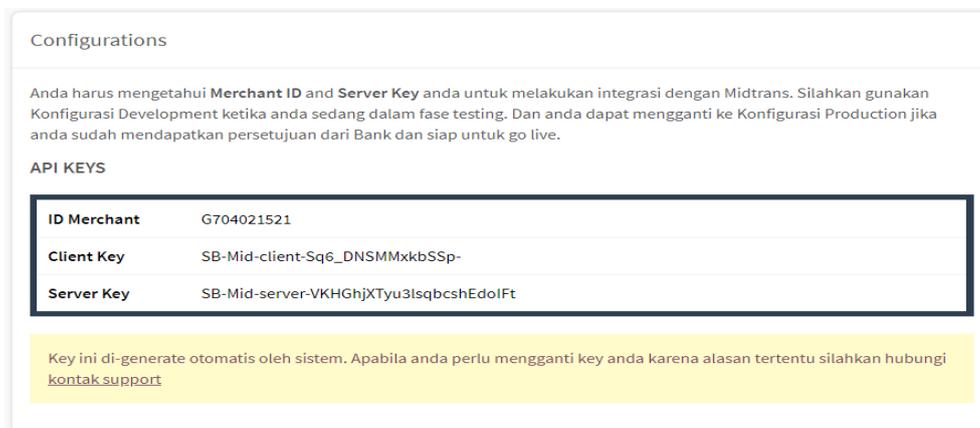
Metode pengujian yang digunakan adalah metode *blackbox*. Hasil pengujian dipaparkan pada tabel 1.

Tabel 1. Hasil Pengujian *Blackbox*

No	Pengujian	Action	Hasil yang diharapkan	Hasil yang didapat	Kesimpulan
1.	Pengujian API Midtrans	Mengirimkan Token API	Berhasil memberikan informasi nominal transfer	Nominal jumlah transfer yang harus di bayarkan berhasil di tampilkan	Berhasil
2.	Pengujian API Raja Ongkir	Mengirimkan Token API	Berhasil memberikan informasi biaya kirim	Nominal biaya pengiriman barang berhasil di tampilkan	Berhasil
3.	Pengujian kecocokan biaya kirim melalui Raja Ongkir dengan situs official cektarif.com	Melakukan perbandingan biaya pengiriman secara langsung	Hasil dari biaya pengiriman sama	Hasil yang didapat adalah biaya pengiriman ke kota cimahi provinsi jawa barat dari kota denpasar dengan nominal sama yaitu Rp.66.000	Berhasil

**4.2. Pengujian API Midtrans**

Pada pengujian transaksi, langkah pertama yang harus dilakukan adalah dengan request token, untuk mendapatkan token yang nantinya akan di gunakan untuk integrasi *prototype ecommerce* dengan aplikasi midtrans, maka pengguna diharuskan untuk registrasi seperti pada gambar 4.



Gambar 4. Token Midtrans

Selanjutnya key yang sudah diberikan oleh pihak midtrans tersebut diterapkan pada bagian *controller* dan *view* dari *prototype e-commerce* seperti pada gambar 5 dan 6.

```
<?php
defined('BASEPATH') OR exit('No direct script access allowed');

class Home extends CI_Controller {
    public function __construct() {
        parent::__construct();
        $this->load->model('Home_model');
        $params = array('server_key' => 'SB-Mid-server-VKHGhJXTyu3lsqbcshEdoIFt',
            'production' => false);
        $this->load->library('midtrans');
        $this->midtrans->config($params);
        $this->load->helper('url');
    }
}
```

Gambar 5. Penerapan Token Server Midtrans pada Controller

```
<script type="text/javascript" src="https://app.sandbox.midtrans.com/snap/
snap.js" data-client-key="SB-Mid-client-Sq6_DNSMMxkbSSp-"></script>
<script src="//ajax.googleapis.com/ajax/libs/jquery/1.11.0/jquery.min.js"></
script>
</head>
<body>
```

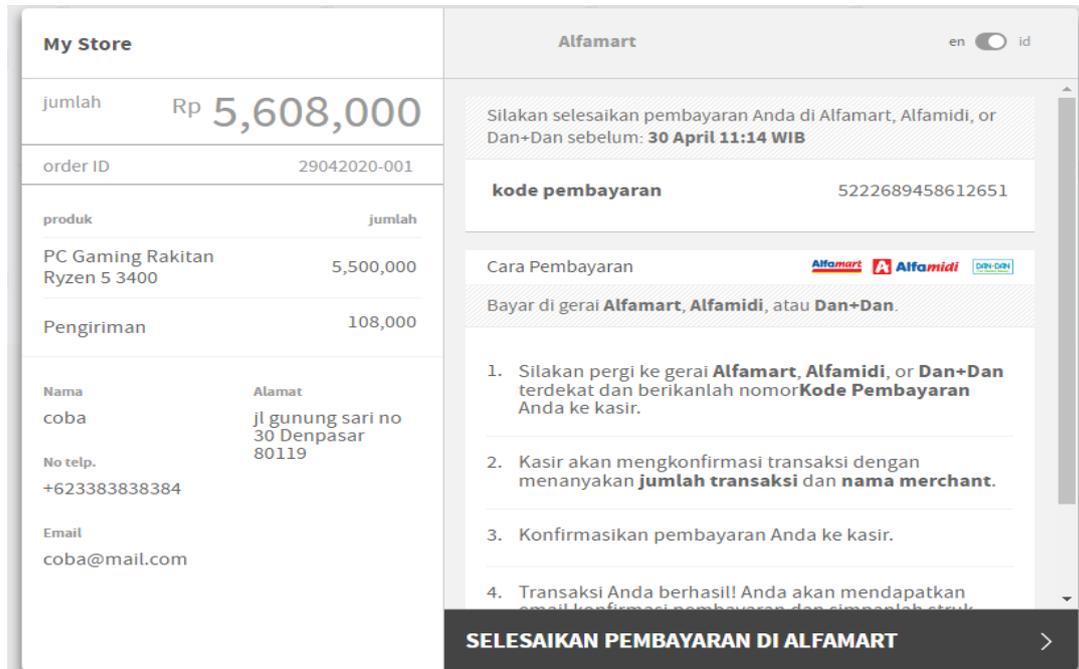
Gambar 6. Penerapan Token Client Midtrans pada View

Setelah konfigurasi token *server key* dan *client key* dilakukan selanjutnya adalah pengujian transaksi. Jika berhasil maka akan muncul tampilan *snap window* seperti pada gambar 7 yang dapat memberikan pilihan cara pembayaran selain transfer bank.

My Store		Pilih Pembayaran	
jumlah	Rp 5,608,000		
order ID	29042020-001		
produk	jumlah		
PC Gaming Rakitan Ryzen 5 3400	5,500,000		
Pengiriman	108,000		
Nama	Alamat		
coba	jl gunung sari no 30 Denpasar 80119		
No telp.			
+623383838384			
Email			
coba@mail.com			
		<div style="border: 1px solid #ccc; padding: 5px;"> <div style="display: flex; justify-content: space-between; align-items: center;"> <span>VISA</span> <span><b>Kartu Kredit</b></span> </div> <p>Bayar dengan Kartu Visa, MasterCard, JCB, atau Amex</p> </div> <div style="border: 1px solid #ccc; padding: 5px;"> <div style="display: flex; justify-content: space-between; align-items: center;"> <span>BCA</span> <span><b>KlikPay</b></span> </div> <p>Bayar dengan BCA KlikPay</p> </div> <div style="border: 1px solid #ccc; padding: 5px;"> <div style="display: flex; justify-content: space-between; align-items: center;"> <span>CIMB</span> <span><b>Clicks</b></span> </div> <p>Bayar dengan akun CIMB Clicks Anda</p> </div> <div style="border: 1px solid #ccc; padding: 5px;"> <div style="display: flex; justify-content: space-between; align-items: center;"> <span>Danamon</span> <span><b>Online Banking</b></span> </div> <p>Bayar dengan akun Danamon Online Banking Anda</p> </div> <div style="border: 1px solid #ccc; padding: 5px;"> <div style="display: flex; justify-content: space-between; align-items: center;"> <span>ATM</span> <span><b>ATM/Bank Transfer</b></span> </div> <p>Bayar dari ATM Bersama, Prima, atau Alto</p> </div> <div style="border: 1px solid #ccc; padding: 5px;"> <div style="display: flex; justify-content: space-between; align-items: center;"> <span>Indomaret</span> <span><b>Indomaret</b></span> </div> <p>Bayar di Indomaret</p> </div> <div style="border: 1px solid #ccc; padding: 5px;"> <div style="display: flex; justify-content: space-between; align-items: center;"> <span>gopay</span> <span><b>Pembayaran QR</b></span> </div> <p>Bayar dengan aplikasi pembayaran pilihan Anda</p> </div>	

Gambar 7. Snap Window Payment Gateway Midtrans

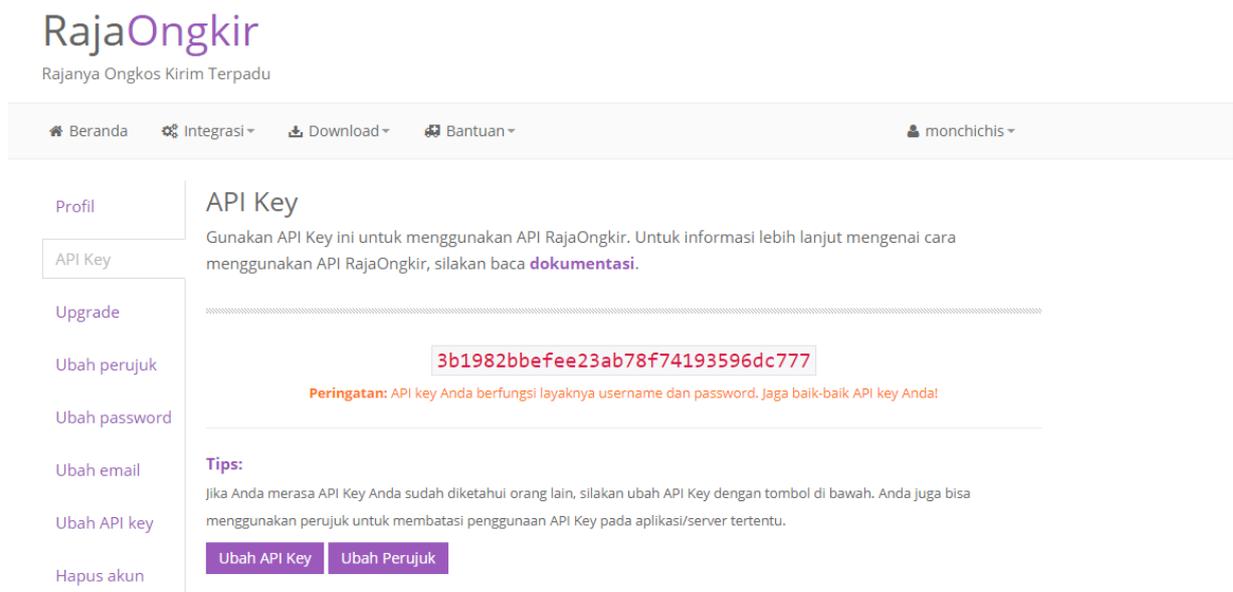
Midtrans sudah memiliki fitur untuk mempermudah melakukan pengujian pembayaran. Dengan memasukkan kode transaksi dan menekan tombol bayar, maka transaksi sudah terbayar. Keberhasilan pengujian pembayaran menggunakan fitur dari midtrans dapat dilihat pada gambar 8.



Gambar 8. Pengujian Pembayaran.

### 4.3. Pengujian API Raja Ongkir

Sama halnya dengan midtrans. untuk pengujian pengecekan biaya kirim dari raja ongkir langkah pertama yang harus dilakukan adalah dengan request token. Untuk mendapatkan token yang nantinya akan di gunakan untuk integrasi *prototype ecommerce* dengan aplikasi raja ongkir, maka pengguna diharuskan untuk registrasi seperti pada gambar 9.



Gambar 9. Token Rajaongkir

Selanjutnya key yang sudah diberikan oleh pihak midtrans tersebut diterapkan pada bagian controller dan view dari *prototype ecommerce* seperti pada gambar 5 dan 6, hasil akhirnya seperti disajikan pada gambar 10 dan gambar 11.

```
public function city() {
    $prov = $this->input->post('prov', TRUE);

    $curl = curl_init();

    curl_setopt_array($curl, array(
        CURLOPT_URL => "http://api.rajaongkir.com/starter/city?province=$prov",
        CURLOPT_RETURNTRANSFER => true,
        CURLOPT_ENCODING => "",
        CURLOPT_MAXREDIRS => 10,
        CURLOPT_TIMEOUT => 30,
        CURLOPT_HTTP_VERSION => CURL_HTTP_VERSION_1_1,
        CURLOPT_CUSTOMREQUEST => "GET",
        CURLOPT_HTTPHEADER => array(
            "key: 3b1982bbefee23ab78f74193596dc777"
        ),
    ));

    $response = curl_exec($curl);
    $err = curl_error($curl);

    curl_close($curl);

    if ($err) {
        echo "cURL Error #:" . $err;
    } else {
        $data = json_decode($response, TRUE);
    }
}
```

Gambar 10. Penerapan Token Rajaongkir pada Controller

```
<?php

$curl = curl_init();

curl_setopt_array($curl, array(
    CURLOPT_URL => "http://api.rajaongkir.com/starter/province",
    CURLOPT_RETURNTRANSFER => true,
    CURLOPT_ENCODING => "",
    CURLOPT_MAXREDIRS => 10,
    CURLOPT_TIMEOUT => 30,
    CURLOPT_HTTP_VERSION => CURL_HTTP_VERSION_1_1,
    CURLOPT_CUSTOMREQUEST => "GET",
    CURLOPT_HTTPHEADER => array(
        "key: 3b1982bbefee23ab78f74193596dc777"
    ),
));

$response = curl_exec($curl);
$err = curl_error($curl);

curl_close($curl);

if ($err) {
    echo "cURL Error #:" . $err;
} else {
    $data = json_decode($response, TRUE);

    for ($i=0; $i < count($data['rajaongkir']['results']); $i++) {
        echo '<option value="'. $data['rajaongkir']['results'][$i]['province_id'].','. $data['rajaongkir']['results'][$i]['province']. '>'. $data['rajaongkir']['results'][$i]['province']. '</option>';
    }
}
```

Gambar 11. Penerapan Token Rajaongkir pada View

Setelah konfigurasi token rajaongkir dilakukan selanjutnya adalah pengujian pengecekan biaya kirim. Jika berhasil maka akan muncul form yang berisi kota tujuan dan ekspedisi yang akan digunakan untuk mengirim seperti gambar 12.

## BILLING DETAILS

Jawa Barat	Cimahi
345	il gunung sari no 30
JNE	
REG	66.000

## YOUR ORDER

PRODUCT	TOTAL
ASUS ZENFONE LIVE L2 × 1	Rp. 999.999
Hp Asus ZenFone Live L2 (ZA550 × 1	Rp. 1.700.000
Vivo Y17 Ram 4Gb Rom 128Gb × 1	Rp. 1.900.000
<b>SUBTOTAL</b>	Rp. 4.599.999
<b>ONGKOS KIRIM</b>	Rp. 66.000
<b>TOTAL PEMBAYARAN</b>	Rp. 4.665.999

Gambar 12. Tampilan Form untuk Menghitung Biaya Kirim

Gambar 12 menampilkan contoh biaya pengiriman sebesar Rp.66.000 ke provinsi Jawa Barat tepatnya ke Kota Cimahi, menunjukkan bahwa pengujian API Raja Ongkir berhasil. Untuk menguji keakuratan tarif biaya kirim menggunakan layanan dari Raja Ongkir, maka penulis melakukan perbandingan tarif dengan situs terpercaya seperti [cektarif.com](http://cektarif.com) dengan tujuan, ekspedisi dan paket yang sama yaitu reguler. Berdasarkan Gambar 12 dan Gambar 13 dapat disimpulkan bahwa tarif biaya kirim menggunakan layanan dari Raja Ongkir terbukti akurat.

[cektarif.com](http://cektarif.com) About Hitung Volume Plus Kodepos

## Expedisi JNE

Customer Service: (021) 2927 8888, Email : [customercare@jne.co.id](mailto:customercare@jne.co.id)

Tarif JNE DENPASAR ke CIMAHI (3 kg)

Layanan	Jenis Kiriman	Tarif	ETD
JTR	JTR	60,000	3-3 Days
JTR250	JTR250	1,600,000	
JTR<150	JTR<150	1,000,000	
JTR>250	JTR>250	2,200,000	
OKE	OKE	53,000	3-3 Days
REG	REG	66,000	2-2 Days

Data di atas adalah tarif terbaru yang dapat kami tampilkan.

Jika harga di atas berbeda dengan kantor agen JNE di lokasi Anda, bisa jadi karena ada perhitungan volumetrik, penambahan asuransi, packing kayu, dll.

Jika hasil di atas tidak sesuai, jangan sungkan untuk menghubungi kami di menu [kontak](#).

Gambar 13. Tarif biaya kirim berdasarkan situs [cektarif.com](http://cektarif.com)

Gambar 13 merupakan hasil perbandingan biaya pengiriman melalui situs official untuk pengecekan biaya pengiriman cektarif.com. dari dan tujuan yang sama yaitu dari kota Denpasar ke provinsi Jawa Barat Kota Cimahi. Hasil yang didapat adalah biaya pengiriman Rp.66.000 sama seperti pengecekan dengan memanfaatkan API dari Raja Ongkir

## 5. Kesimpulan

Setelah dilakukan pengujian dengan pencocokan biaya pengiriman dan total biaya yang harus dibayar. maka dapat disimpulkan bahwa dengan memanfaatkan API dari Midtrans dapat membantu customer untuk melakukan pembayaran melalui channel yang bekerjasama dengan Midtrans. dengan rincian sebagai berikut:

- 1) Dengan memanfaatkan API dari RajaOngkir customer dapat mengetahui secara langsung biaya kirim sebuah produk, dan bagi pengelola tidak memerlukan pengecekan biaya kirim secara konvensional
- 2) Dengan memanfaatkan API dari Midtrans sebagai *payment gateway* dan API dari Raja Ongkir yang digunakan untuk mengidentifikasi biaya kirim sebuah produk. maka integrasi antar sistem sudah dapat dilakukan sehingga terbentuknya sebuah EAI (*Enterprise Application Integration*)

## DAFTAR REFERENSI

- [1] Febrianto, M. *Penerapan Payment Gateway Dan Tracking Barang Pada E-Commerce Toko Dazzle Berbasis Website*, Tugas Akhir, University of Technology Yogyakarta, 2020.
- [2] Susanti, M. Pemanfaatan E-Commerce Untuk Mendukung UMKM Dalam Pemasaran. *Prosiding SISFOTEK*, 2017, 1(1): 201-208.
- [3] Darnis, F., & Azdy, R. A. Pemanfaatan Media Informasi Website Promosi (e-Commerce) sebagai Upaya Peningkatan Pendapatan UMKM Desa Pedado. *SINDIMAS*, 2019, 1(1): 275-278.
- [4] Widani, N. M., Abiyasa, A. P., & Darma, G. S. Menguji Ketajaman Implementasi E-Commerce Dalam Penjualan Kamar Hotel di Bali. *Jurnal Manajemen Bisnis*, 2019, 16(2): 79-98.
- [5] Listianto, K. F., Fauzi, R. I., Irviani, R., Kasmi, K., & Garaika, G. Aplikasi E-Commerce Berbasis Web Mobile Pada Industri Konveksi Seragam Drumband Di Pekon Klaten Gadingrejo Kabupaten Pringsewu. *J. TAM (Technol. Accept. Model)*, 2017, 8(2): 146-152.
- [6] Karya, G. Penerapan enterprise application integration sebagai model integrasi sistem informasi di Universitas Katolik Parahyangan. *Semantik*, 2012, 2(1).
- [7] Tanuwidjaja, C. J., & Setiawan, A. Perancangan dan Pembuatan Website E-Commerce pada Toko Aksesoris Komputer di Surabaya. *Jurnal Infra*, 2017, 5(2): 301-307.
- [8] Somya, R., & Nathanael, T. M. E. Pengembangan Sistem Informasi Pelatihan Berbasis Web Menggunakan Teknologi Web Service Dan Framework Laravel. *Jurnal Techno Nusa Mandiri*, 2019, 16(1): 51-58.
- [9] Schwarz, G. D., & Riehle, D. What Microservices Can Learn from Enterprise Information Integration. In *HICSS January 2020*: 1-10.
- [10] Susanto Anna Dara Andriana, R. Perbandingan Model Waterfall dan Prototyping untuk Pengembangan Sistem Informasi. *Majalah Ilmiah UNIKOM*, 2016, 14(1): 41-46