

## PENERAPAN REST API DAN INTEGRASI MIDTRANS SEBAGAI *PAYMENT GATEWAY* PADA PLATFORM PELATIHAN *ONLINE* (Studi Kasus: *Bisabacakitab*)

Muhammad Al Biruni, Ahmad Faisol, Nurlaily Vendyansyah  
Program Studi Teknik Informatika S1, Fakultas Teknologi Industri  
Institut Teknologi Nasional Malang, Jalan Raya Karanglo km 2 Malang, Indonesia  
1918079@scholar.itn.ac.id

### ABSTRAK

Pelatihan *online* merupakan metode pembelajaran yang dilakukan secara daring yang sangat populer dewasa ini. Adapun proses pembelajaran kitab gundul saat ini masih sedikit yang menerapkan proses pembelajaran secara digital, mayoritas masih harus datang ke pesantren dan melakukan proses belajar mengajar di pesantren tersebut. Salah satu platform *online* baca kitab gundul saat ini adalah *Bisabacakitab*. *Bisabacakitab* adalah platform pelatihan *online* yang memberikan fasilitas belajar baca kitab gundul. Adapun permasalahan sistem *Bisabacakitab* saat ini masih menggunakan sistem pembayaran manual dengan Whatsapp, hal ini menyebabkan transaksi dalam *Bisabacakitab* harus dilakukan secara teliti oleh admin, jika ada yang terlewat akan sangat mempengaruhi proses bisnis. Untuk mengatasi permasalahan tersebut, penulis mengimplementasikan API dari Midtrans sebagai *Payment Gateway*. Sehingga semua transaksi akan dilakukan dengan beberapa pilihan metode pembayaran yang disediakan midtrans seperti QRIS, VA, dan GoPay. Sistem yang dibangun menerapkan REST API dengan *framework* laravel agar integrasi dengan midtrans dapat dilakukan dengan mudah. Berdasarkan pengujian sistem menggunakan *Blackbox Testing* mendapatkan hasil bahwa sistem dapat menangani proses transaksi pembayaran menggunakan *payment gateway*, memonitoring status pesanan, dan memberikan akses kepada pengguna setelah pengguna berhasil membayar dengan tingkat keberhasilan 100%. Dengan adanya hal ini permasalahan sistem *Bisabacakitab* sudah teratasi dengan bantuan API dari Midtrans. Penulis mengambil kesimpulan bahwa pengembangan aplikasi *Bisabacakitab* dapat dikatakan berhasil menyelesaikan masalah-masalah sebelumnya dan berhasil memenuhi kebutuhan customer-nya yang dibuktikan dengan nilai kuisioner sebesar 89% dengan total 10 pertanyaan dari total 55 responden yang berpartisipasi, beberapa dari total responden adalah pengguna lama *Bisabacakitab*.

**Kata kunci :** *Application Programming Interface, e-learning, online course, payment gateway*

## 1. PENDAHULUAN

### 1.1 Latar Belakang

Pelatihan *online* atau e-learning adalah suatu metode pembelajaran yang dilakukan secara daring melalui internet atau platform *e-learning*. Proses pelatihan ini menggunakan berbagai macam teknologi informasi dan komunikasi seperti *video conferencing*, webinar, forum diskusi daring, dan platform pembelajaran berbasis web [1].

Hal ini menjadikan para pebisnis untuk mengembangkan platform pelatihan *online* karena biaya yang relatif murah dan cepat. Salah satu platform pelatihan *online* tersebut adalah *Bisabacakitab*. *Bisabacakitab* adalah sebuah platform pelatihan *online* yang menyediakan pembelajaran baca kitab gundul online.

Adapun permasalahan sistem *Bisabacakitab* saat ini adalah masih belum mengakomodir proses transaksi pembelian kursus yang menyebabkan pengguna ketika ingin membeli kursus baca kitab pengguna harus menghubungi admin lewat *Whatsapp* untuk melakukan transaksi dan proses ini memakan waktu yang sangat lama dan tidak efisien karena harus menunggu respon dari admin *Bisabacakitab*. Hal ini menimbulkan masalah yang lain yaitu tidak adanya sistem yang menjembatani

transaksi sehingga semua rekapan transaksi dilakukan secara manual dan tidak efektif yang menyebabkan tidak terstruktur nya data laporan transaksi pada *Bisabacakitab*.

Untuk dapat mengatasi permasalahan yang terjadi, maka diusulkan sebuah rancangan arsitektur baru untuk platform *Bisabacakitab* berupa Web Service dan integrasi *payment gateway* dengan Midtrans menggunakan REST API. Dengan adanya hal tersebut pengguna yang ingin membeli kursus baca kitab tidak perlu menghubungi admin karena proses pembayaran menggunakan *payment gateway* dan proses bisnis yang terotomasi melalui sistem, dan admin juga tidak perlu merekap transaksi secara manual karena semua data transaksi terekap melalui sistem.

### 1.2 Rumusan Masalah

Berdasarkan permasalahan yang teridentifikasi pada Latar Belakang, maka dapat dirumuskan beberapa rumusan masalah sebagai berikut:

1. Bagaimana cara merancang dan mengembangkan sebuah aplikasi untuk memudahkan pengguna *Bisabacakitab* dalam mengakses dan melakukan pembelian kursus?

2. Bagaimana membangun arsitektur *Web Service* yang terintegrasi dengan *Payment Gateway Midtrans*?
3. Bagaimana mendesain dan mengembangkan sistem yang merekap semua data transaksi *Bisabacakitab* secara otomatis?

### 1.3 Batasan Masalah

Berdasarkan pembuatan aplikasi ini terdapat beberapa batasan dalam pembuatan yaitu sebagai berikut:

1. Teknologi *Web Service* yang dikembangkan hanya menggunakan REST API berbasis framework Laravel.
2. Metode Pembayaran yang digunakan berdasarkan metode yang tersedia pada platform Midtrans untuk akun personal.
3. Penulis tidak menjelaskan bagaimana sistem payment gateway dibangun dari 0.

### 1.4 Tujuan

Adapun tujuan yang ingin dicapai yaitu:

1. Mendesain dan mengembangkan sebuah aplikasi untuk memudahkan pengguna *Bisabacakitab* dalam mengakses dan melakukan pembelian kursus.
2. Menerapkan teknologi *Web Service* dengan REST API dan melakukan integrasi dengan *Payment Gateway Midtrans* untuk mengembangkan proses bisnis secara otomatis.
3. Mempermudah admin dalam hal perekapan transaksi yang dilakukan oleh pengguna.

### 1.5 Manfaat

Terdapat beberapa manfaat dari pembuatan aplikasi ini sebagai berikut:

1. Membantu pengguna untuk mengakses dan membeli materi pembelajaran baca kitab.
2. Mengaplikasikan ilmu yang didapat selama perkuliahan berlangsung khususnya dalam bidang Rekayasa perangkat lunak.
3. Memudahkan admin dalam merekap data transaksi *Bisabacakitab*.

## 2. TINJAUAN PUSTAKA

Khamdun, Eron Rikardo Nainggolan, dan Jordy Lasmana Putra dalam jurnalnya yang berjudul "Perancangan Sistem Informasi Manajemen Pelatihan Kursus Berbasis Web Pada CV Nixtrain Infotama" menjabarkan bahwa konsep sistem informasi yang dimiliki oleh CV Nixtrain Infotama masih belum tersistematis dan masih manual, karena belum tertata atau tersusun rapi yang membuat proses kegiatan pelatihan menjadi kurang efektif dan efisien, semua proses bisnis masih dilakukan secara luring. dengan adanya penelitian ini penulis mengembangkan sebuah aplikasi sistem informasi manajemen berbasis web yang memungkinkan peserta kursus untuk mengakses materi kursus secara *online* dengan mudah dan cepat. Adapun

manfaat dari sisi admin adalah dapat mengolah data kursus, penjualan, peserta dalam satu portal [2].

Ibu Mira Orisa dan Bapak Ahmad Faisol dalam jurnalnya yang berjudul "Pemanfaatan Teknologi Web Service Untuk Mendukung Efisiensi Transaksi Pemesanan Dan Penjualan Pada Dyra Collection" menjelaskan bahwa toko baju DYRA COLLECTION masih menggunakan sistem pemasaran melalui mulut ke mulut sehingga hal ini dapat mengurangi kinerja DYRA COLLECTION dalam melayani konsumen. Masalah lainnya adalah proses bisnis layanan dan pembelian masih menggunakan sistem pencatatan secara manual menggunakan buku arsip. Dalam aplikasi yang dikembangkan, memungkinkan bagi para pengguna untuk melihat katalog baju dan melakukan pembelian baju secara *online* [3].

Yenni Indri Darmayanti dalam jurnalnya yang berjudul "Pengembangan Sistem Informasi Akademik Berbasis Android Studi Kasus Informatika ITN Malang" menjelaskan bahwa teknik informatika ITN belum mengadopsi aplikasi berbasis android untuk sistem informasi akademik. Hal ini menyebabkan mahasiswa harus menggunakan browser untuk mengakses SIAKAD. Maka dari itu, dalam penelitian ini dirancang sebuah aplikasi sistem informasi akademik berbasis android yang dapat digunakan untuk melihat nilai, tagihan kampus dan profil siswa dengan mudah [4].

### 2.2 Web Service

*Web Service* adalah sistem perangkat lunak yang digunakan untuk berkomunikasi antara perangkat lunak yang berbeda melalui jaringan internet. *Web Service* menggunakan protokol jaringan yang sering digunakan seperti HTTP, XML, dan SOAP untuk melakukan *request* dan *response* antara aplikasi yang berbeda. Dalam penggunaannya, sebuah aplikasi mampu menyajikan fungsionalitasnya dengan cara yang bisa dimengerti oleh aplikasi lain, dengan menggunakan protokol yang telah distandarisasi seperti XML (*eXtensible Markup Language*) atau JSON (*JavaScript Object Notation*). Dengan memanfaatkan *Web Service*, aplikasi-aplikasi dapat berinteraksi dan berbagi data dengan efisien tanpa terikat pada platform, bahasa pemrograman, atau arsitektur yang berbeda [5].

### 2.3 REST API

REST API adalah sebuah standar arsitektur perangkat lunak yang digunakan untuk membangun aplikasi web. REST API memungkinkan aplikasi untuk berkomunikasi dengan aplikasi lain melalui protokol HTTP (*Hypertext Transfer Protocol*) dengan memanfaatkan metode HTTP seperti GET, POST, PUT, dan DELETE. REST API juga menggunakan format data umum seperti XML atau JSON untuk pertukaran data antar aplikasi [6]. Secara khusus, istilah "REST" merujuk pada

seperangkat prinsip dalam arsitektur jaringan yang mengatur bagaimana sekumpulan sumber daya didefinisikan dan diakses. Istilah ini digunakan secara luas untuk menggambarkan segala jenis antarmuka sederhana yang memfasilitasi pertukaran data dalam bidang tertentu melalui protokol HTTP [4].

### 2.4 Payment Gateway

*Payment Gateway* atau gerbang pembayaran adalah sistem pembayaran elektronik yang memungkinkan transaksi keuangan antara pelanggan dan pedagang dilakukan secara *online* dan aman melalui *internet*. *Payment Gateway* menyediakan berbagai pilihan metode pembayaran seperti kartu kredit, debit, dan transfer bank. [8] Adapun *Payment Gateway* yang digunakan dalam penelitian ini adalah Midtrans. Dengan mengintegrasikan *Payment Gateway*, diharapkan pelanggan dapat mengalami kemudahan dan kecepatan dalam bertransaksi. Awalnya, Midtrans, yang dulunya dikenal sebagai Veritrans, menjadi salah satu sistem pembayaran online yang populer di Indonesia dan internasional.

### 2.5 Laravel

Laravel merupakan sebuah kerangka kerja untuk aplikasi *web* yang menggunakan bahasa pemrograman PHP dan bersifat *open-source* serta gratis. *Framework* ini dirancang oleh Taylor Otwell dengan tujuan utama untuk memfasilitasi pembuatan aplikasi *web* berdasarkan pola Model-View-Controller (MVC) [6]. Laravel merupakan suatu *framework* yang memiliki popularitas yang signifikan di Indonesia. *Framework* ini diperkenalkan pada tahun 2011 dan hingga sekarang telah banyak digunakan oleh para pengembang untuk membangun situs *web* yang *powerful* dan canggih. Seperti halnya dengan *framework* lainnya, pengembangan menggunakan Laravel bertujuan untuk menyederhanakan proses pembuatan aplikasi *web*. Namun, tidak hanya itu, Laravel juga memiliki keunggulan dalam membuat proses pengembangan aplikasi *web* menjadi lebih rapi, menyenangkan, dan penuh ekspresi.

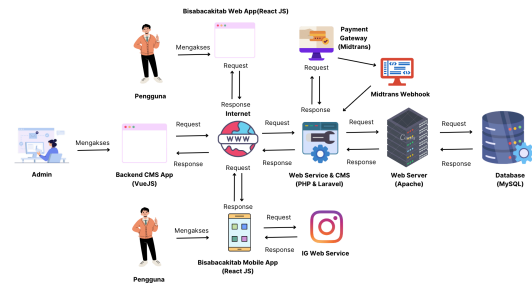
### 2.6 React.JS

React JS adalah sebuah pustaka *JavaScript open-source* yang digunakan untuk membangun antarmuka pengguna (*user interface*) pada aplikasi *web*. ReactJS menggunakan pendekatan berbasis komponen yang memungkinkan pengembang untuk memecah antarmuka pengguna menjadi komponen-komponen kecil yang dapat digunakan kembali. ReactJS juga memungkinkan pengembang untuk membuat aplikasi *web* yang cepat, interaktif, dan mudah dikembangkan [10].

## 3. METODE PENELITIAN

### 3.1 Blok Diagram

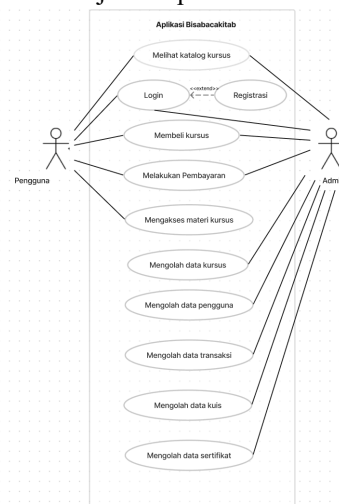
Rancangan arsitektur integrasi *payment gateway* pada aplikasi *Bisabacakitab* menggunakan REST API ditunjukkan pada Gambar 1.



Gambar 1. Blok diagram arsitektur *Bisabacakitab*

### 3.2 Use Case Diagram

Rancangan *Use Case Diagram* dari sistem yang dibuat ditunjukkan pada Gambar 2.



Gambar 2. *Use case diagram Bisabacakitab*

### 3.3 Entity Relationship Diagram (ERD)

Rancangan basis data pada sistem yang dikembangkan ditunjukkan pada Gambar 3 [11].

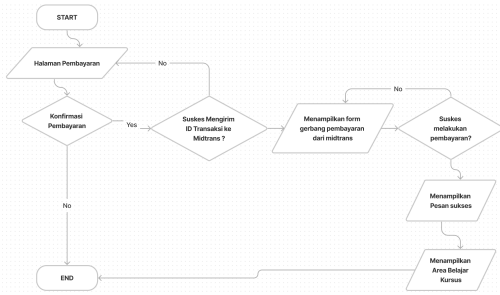


Gambar 3. ERD *Bisabacakitab*

### 3.4 Flowchart

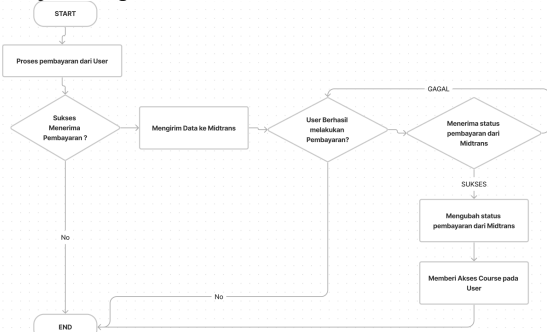
#### a. Flowchart Sistem Pembayaran sisi user

*Flowchart* Sistem Pembayaran sisi *user* ditunjukkan pada Gambar 4.



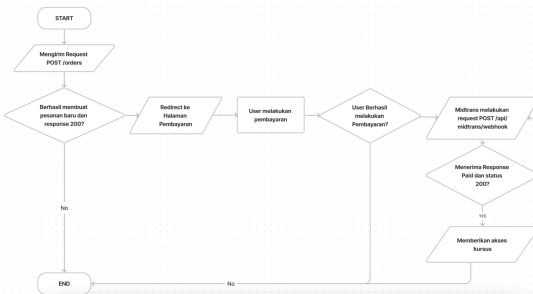
Gambar 4. Flowchart sistem pembayaran sisi user

**b. Flowchart Sistem Pembayaran sisi sistem**  
*Flowchart Sistem Pembayaran sisi sistem* ditunjukkan pada Gambar 5.



Gambar 5. Flowchart pembayaran sisi sistem

**c. Flowchart REST API Sistem Pembayaran**  
*Flowchart REST API Sistem Pembayaran* ditunjukkan pada Gambar 6.



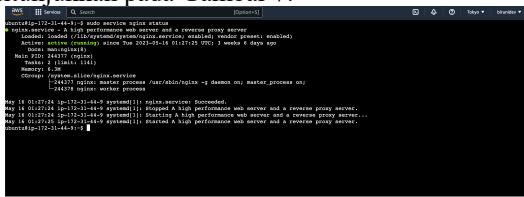
Gambar 6. Flowchart REST API Pembayaran

**4. HASIL DAN PEMBAHASAN**

**4.1 Hasil Implementasi REST API menggunakan Laravel**

**a. Tampilan Web Server Bisabacakitab**

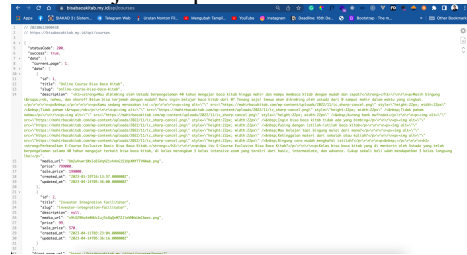
Tampilan *Web Server Bisabacakitab* menunjukkan REST API sedang aktif disisi *server* ditunjukkan pada Gambar 7.



Gambar 7. Tampilan Web Server Bisabacakitab

**b. Tampilan Hasil REST API di browser**

Tampilan Hasil REST API *Bisabacakitab* di browser ditunjukkan pada Gambar 8.

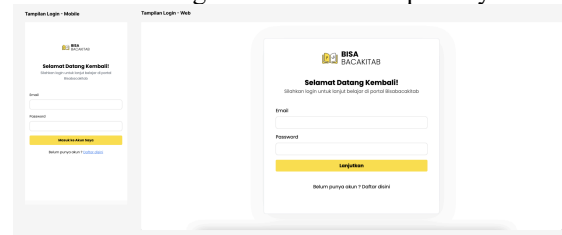


Gambar 8 Tampilan Hasil REST API di Browser

**4.2 Hasil Integrasi REST API dengan Aplikasi Bisabacakitab**

**a. Tampilan Halaman Login**

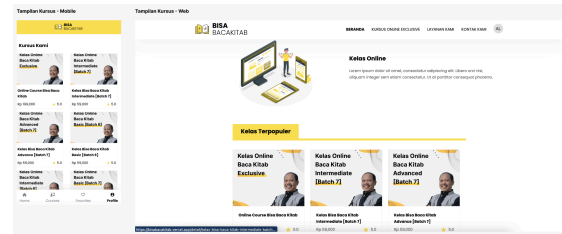
Sebelum mengakses sistem *user* harus *login* terlebih dahulu agar bisa melakukan pembayaran.



Gambar 9 Tampilan halaman login

**b. Tampilan Halaman Kursus**

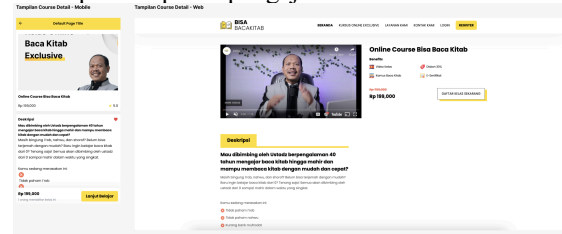
Setelah login, pengguna dapat memilih kursus yang diinginkan seperti yang ditampilkan pada Gambar 10



Gambar 10 Tampilan halaman kursus

**c. Tampilan Halaman Kursus Detail**

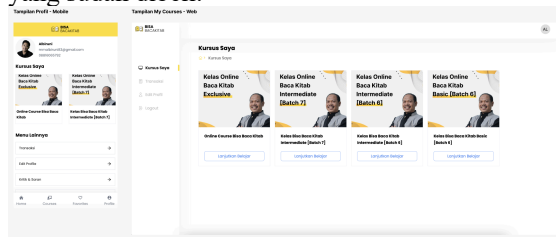
Setelah memilih kursus pengguna dapat melihat detail kursus seperti *preview* materi kursus, deskripsi dan profil pengajar.



Gambar 11. Tampilan halaman kursus detail

**d. Tampilan halaman member area**

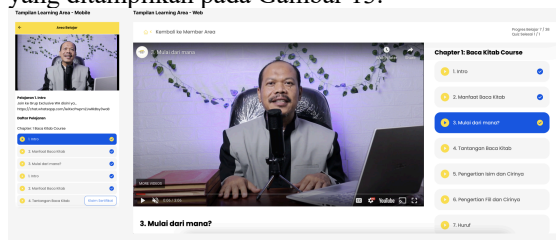
Halaman ini dapat diakses ketika pengguna sudah *login*, halaman ini akan menampilkan kursus yang sudah dibeli.



Gambar 12. Tampilan halaman *member area*

**f. Tampilan Halaman *Learning Area***

Halaman *Learning Area* hanya dapat diakses oleh pengguna yang memiliki akses kursus seperti yang ditampilkan pada Gambar 13.

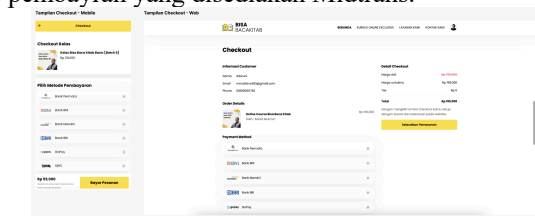


Gambar 13 Tampilan Halaman *Learning Area*

**4.3 Hasil Integrasi Payment Gateway Midtrans pada Aplikasi Bisabacakitab**

**a. Tampilan Halaman *Checkout***

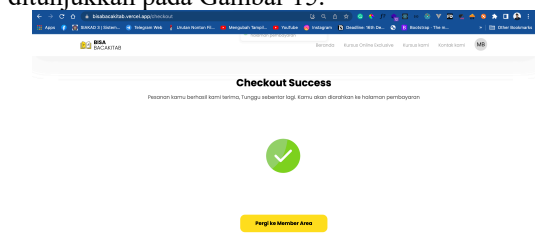
Pengguna yang telah memilih kursus yang dibeli akan langsung diarahkan ke halaman *checkout*. Pada halaman ini pengguna memilih metode pembayaran yang disediakan Midtrans.



Gambar 14 Tampilan hasil halaman *checkout*

**b. Tampilan sukses membuat pesanan**

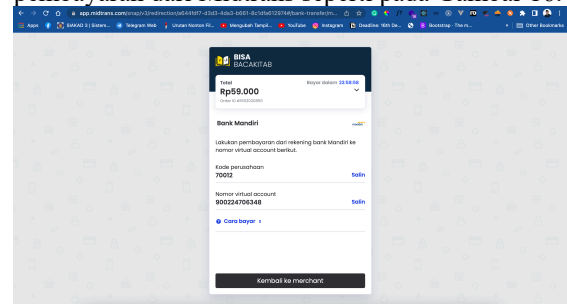
Pengguna yang telah memilih metode pembayaran akan diarahkan ke halaman sukses yang ditunjukkan pada Gambar 15.



Gambar 15. Tampilan sukses membuat pesanan

**c. Tampilan Midtrans menggunakan metode pembayaran Bank Mandiri**

Setelah beberapa detik dari halaman sukses, pengguna akan diarahkan ke halaman pembayaran dari Midtrans seperti pada Gambar 16.



Gambar 16. Tampilan Midtrans metode Bank mandiri

**4.4 Pengujian Sistem**

Pengujian sistem dilakukan untuk mengetahui apakah hasil pengembangan dan integrasi sudah sesuai dengan rancangan yang diharapkan.

**4.4.1 Pengujian Fungsional (*Blackbox Testing*)**

Pengujian *blackbox* dilakukan dengan cara menguji fitur-fitur yang ada pada aplikasi secara langsung. Hasil pengujian dapat dilihat pada Tabel 1

Tabel 1 Hasil Pengujian Sistem Bisabacakitab

Skenario pengujian	Uji Kasus	Hasil yang diharapkan	Hasil
Menampilkan data kursus dengan judul dan harga yang benar	Klik Lihat Selengkapnya pada tab Kelas Online, section Layanan Kami	Menampilkan data kursus yang sesuai dengan database	Berhasil
Melakukan pembelian kursus (sudah login)	Klik Daftar Kelas Sekarang, memilih metode pembayaran dan Klik Selesaikan Pemesanan	Sistem mengarahkan ke gerbang pembayaran midtrans	Berhasil
Melakukan pembelian kursus tanpa memilih metode pembayaran	Klik Daftar Kelas Sekarang, mengosongkan metode pembayaran dan Klik Selesaikan Pemesanan	Muncul Popup error Pilih Metode Pembayaran !	Berhasil
Menyelesaikan pembayaran kursus	Klik Back to merchant setelah pembayaran	Menampilkan kursus yang sudah dibeli	Berhasil

Skenario pengujian	Uji Kasus	Hasil yang diharapkan	Hasil
Tidak Menyelesaikan pembayaran	Abaikan pembayaran setelah klik Selesaikan Pembayaran	Status transaksi tetap BELUM DIBAYAR	Berhasil
Melakukan Request POST pada /api/orders	header: Content-Type: Application/json Authorization: Bearer `\${Token}`  Body: user_id: 1, ordered_product: { id: 2, type: 'course' } gross_amount: 29304 order_code: #4223	Menampilkan link midtrans, data order, kode status dan pesan berupa json	Berhasil
Melakukan Request POST pada /api/orders dengan menginputkan data kursus yang tidak ada dalam database	header: Content-Type: Application/json Authorization: Bearer `\${Token}`  Body: user_id: 1, ordered_product: { id: 100, type: 'course' } gross_amount: 29304 order_code: #4223	Mengembalikan response 404 dan pesan error Kursus tidak ditemukan	Berhasil
Melakukan Request POST pada /api/orders dengan menginputkan data user yang tidak ada dalam database	header: Content-Type: Application/json Authorization: Bearer `\${Token}`  Body: user_id: 100, ordered_product: { id: 1, type: 'course' } gross_amount: 29304	Mengembalikan response 404 dan pesan error User tidak ditemukan	Berhasil

Skenario pengujian	Uji Kasus	Hasil yang diharapkan	Hasil
	order_code: #4223		

#### 4.4.1 Pengujian UAT (User Acceptance Test)

Pada sub bab ini dipaparkan metode pengujian menggunakan UAT berdasarkan data kuisisioner yang dilakukan kepada pengguna lama atau baru aplikasi Bisabacakitab. Pengguna aplikasi mengisi kuisisioner berdasarkan pengalamannya pada saat melakukan uji coba aplikasi. Tiap-tiap pertanyaan dari kuisisioner tersebut menunjukkan tingkat *usability* menurut penerimaan user. Tingkat *usability* akan dinilai dalam skala 5. Form Kuisisioner disebar melalui Google Form ke 55 orang baik dari kalangan mahasiswa atau pelajar. Adapun untuk hasil pengujian kuisisioner dapat dilihat pada Tabel 2.

No	Pertanyaan	Jumlah	Jumlah/Respon	Persentase
1	Apakah tampilan pertama muncul pada aplikasi Bisabacakitab ini menarik?	238	4.33	87%
2	Apakah menu dalam aplikasi mudah dipahami?	248	4.51	90%
3	Apakah tampilan pada halaman detail kursus menarik?	247	4.49	90%
4	Apakah tampilan pada halaman checkout mudah dipahami?	243	4.42	88%
5	Apakah proses pembayaran kursus mudah digunakan?	238	4.33	87%
6	Apakah tampilan area belajar mudah dipelajari?	247	4.49	90%
7	Apakah menu dalam member area mudah digunakan?	238	4.33	87%
8	Apakah fitur pada area belajar sesuai	246	4.47	89%

No	Pertanyaan	Jumlah	Jumlah/Respon den	Persen tase
	dengan kebutuhan?			
9	Apakah fitur klaim sertifikat mudah digunakan?	243	4.42	88%

Dari hasil pengujian fitur aplikasi secara keseluruhan fitur-fitur dapat berjalan dengan baik dari mulai melakukan login hingga melakukan pemesanan dan pembelian kursus, akses kursus, dan. Berdasarkan dari hasil tabel kuesioner pada Tabel 4.11 dengan total responden 55 dengan masing-masing 10 pertanyaan bernilai rata-rata 89% yang masuk dalam kategori sangat baik.

## 5. KESIMPULAN

Berdasarkan pengembangan aplikasi dapat disimpulkan sebagai berikut, Penerapan *Web Service* menggunakan Laravel memberikan fleksibilitas dalam pengembangan arsitektur sistem pada sebuah aplikasi. Berdasarkan hasil pengujian blackbox pada bagian REST API menunjukkan bahwa *Bisabacakitab* dapat berjalan di multiplatform (*Web* dan *Android*) dengan memanfaatkan satu *Web Service* atau REST API yang sama artinya aplikasi *mobile* dan *website* akan selalu terintegrasi satu sama lain. Berdasarkan hasil pengujian UAT, Peng-integrasian *payment gateway* pada sebuah aplikasi memudahkan pengguna karena pengguna memiliki pilihan yang beragam terhadap metode pembayaran. Di sisi bisnis, *payment gateway* memudahkan pebisnis untuk melakukan rekap transaksi dengan melihat *dashboard* akun pada platform *payment gateway*. Hasil penelitian ini menunjukkan bahwa pengembangan aplikasi *Bisabacakitab* dapat dikatakan berhasil menyelesaikan masalah-masalah sebelumnya dan berhasil memenuhi kebutuhan customer-nya dibuktikan dengan nilai kuisisioner sebesar 89% dengan total 10 pertanyaan dari total 55 responden yang berpartisipasi, beberapa dari total responden adalah pengguna lama *Bisabacakitab*. Secara garis besar aplikasi yang dikembangkan sudah siap untuk dirilis ke pasar untuk menjangkau lebih banyak pengguna.

Adapun saran-saran setelah melakukan penerapan *web service* dan integrasi *payment gateway* menggunakan REST API pada platform pelatihan *online Bisabacakitab* ini adalah sebagai berikut: 1) Bagi peneliti, sebaiknya dapat menggunakan metode *payment gateway* yang lain seperti Doku, Google Pay, Xendit atau metode metode yang lain. Hal ini dapat membantu *software developer* untuk menentukan *payment gateway* mana yang terbaik bagi platform mereka; 2) Bagi peneliti, sebaiknya dapat menggunakan metode *Web Service* yang lain selain menggunakan REST API ada beberapa metode yang bisa diterapkan antara lain GraphQL, gRPC, dan SOAP.

## DAFTAR PUSTAKA

- [1] S. L. Smith and M. D. Abell, "Online Training: A Review of the Literature," *Journal of Educational Technology Systems*, pp. 135–153, 2019.
- [2] Khamdun, Rikardo Nainggolan, and Jordy Lasmana Putra, "Perancangan Sistem Informasi Manajemen Pelatihan Kursus Berbasis Web Pada CV Nixtrain Infotama," 2021.
- [3] M. Orisa and A. Faisol, "PEMANFAATAN TEKNOLOGI WEB SERVICE UNTUK Mendukung Efisiensi Transaksi Pemesanan dan Penjualan Pada DYRA Collection," *Industri Inovatif: Jurnal Teknik Industri*, vol. 7, no. 1, pp. 15–19, Feb. 2017, [Online]. Available: <https://ejournal.itn.ac.id/index.php/industri/article/view/866>
- [4] Y. I. Darmayanti, "PENGEMBANGAN SISTEM INFORMASI AKADEMIK BERBASIS ANDROID STUDI KASUS TEKNIK INFORMATIKA ITN MALANG," 2017.
- [5] D. Widya Sari, S. Kosasi, and I. Dewa Ayu Eka Yuliani, "PEMANFAATAN RESTFUL WEB SERVICES PADA PERANGKAT LUNAK PENYEWAAN LAPANGAN BADMINTON UTILIZATION OF RESTFUL WEB SERVICES IN BADMINTON FIELD RENTAL SOFTWARE," 2022.
- [6] A. Kurniawan and R. Budiarto, "Implementasi REST API pada Aplikasi Perpustakaan Digital Menggunakan Framework Laravel," *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, pp. 293–298, 2019.
- [7] H. Wakkang and B. Irianto, "IMPLEMENTASI WEB SERVICE DENGAN METODE REST API UNTUK INTEGRASI DATA COVID 19 DI SULAWESI SELATAN," vol. 2, no. 1, pp. 12–22, 2022, doi: 10.31850/jsilog.v2i1.
- [8] Y. Andriansyah, "Pengembangan Sistem Payment Gateway untuk Menunjang Bisnis Online di Indonesia," *Jurnal Teknik ITS*, vol. 5, no. 2, pp. A1–A6, 2016.
- [9] D. Purnama Sari, R. Wijanarko, and J. X. Menoreh Tengah, "Implementasi Framework Laravel pada Sistem Informasi Penyewaan Kamera (Studi Kasus Di Rumah Kamera Semarang)," vol. 2, no. 1, pp. 32–36, 2019.
- [10] I. Puspitaningrum and H. Sulisty, "Penerapan ReactJS dalam Pengembangan Aplikasi E-Commerce pada Sisi Klien,"

- Jurnal Informatika Upgris*, vol. 5, no. 1, pp. 11–20, 2019.
- [11] H. Nalatissifa, N. Maulidah, A. Fauzi, R. Supriyadi, and S. Diantika, “RANCANG BANGUN SISTEM INFORMASI PERPUSTAKAAN BERBASIS WEBSITE PADA SMK NEGERI 1 BUMIJAWA.”