

**APLIKASI *PAYMENT GATEWAY AGGREGATOR***

**Oleh :**

**MOHAMAD FEBRIAN MOSII**

**4519215002**

**SKRIPSI**

**PROGRAM STUDI TEKNIK INFORMATIKA**

**FAKULTAS TEKNIK UNIVERSITAS PANCASILA**

**JAKARTA**

**2023**



**APLIKASI *PAYMENT GATEWAY AGGREGATOR***

**Oleh :**

**MOHAMAD FEBRIAN MOSII**

**4519215002**

**ESSAY**

**INFORMATICS ENGINEERING STUDY PROGRAM**

**ENGINEERING FACULTY OF PANCASILA UNIVERSITY**

**JAKARTA**

**2023**

# LEMBAR PERSETUJUAN

Skripsi ini dibuat untuk

memenuhi syarat guna mencapai gelar

**SARJANA TEKNIK**

pada Program Studi Teknik Informatika Fakultas Teknik

Universitas Pancasila Jakarta dengan judul:

**APLIKASI *PAYMENT GATEWAY AGGREGATOR***

dinyatakan telah disetujui dan diterima:

Menyetujui:

Gregorius Hendita Artha Kusuma,S.Si.,M.Cs

**Pembimbing**

Mengetahui:

Dr. Ionia Veritawati, S.Si., MT

**Ketua Program Studi Teknik Informatika**

# LEMBAR KEASLIAN

Dengan ini saya yang bertanda tangan di bawah ini :

Nama : Mohamad Febrian Mosii

No. Pokok : 4519215002

Fakultas : Teknik

Program Studi : Teknik Informatika

Peminatan : *Software Enginering*

Menyatakan :

Bahwa skripsi ini dibuat dan diselesaikan secara mandiri dan bukan hasil saduran karya orang lain serta hanya menggunakan acuan dari literatur yang ada.

Jika terbukti tidak sesuai dengan yang tersebut di atas, maka saya bersedia menerima sanksi yang ada.

Jakarta, 2 Juni 2023

**Mohamad Febrian Mosii**

# KATA PENGANTAR

*Assalamu’alaikum Wr.Wb.*

Segala puji dan syukur saya panjatkan kehadirat Tuhan Yang Maha Esa karena atas limpahan rahmat dan karunia-Nya penulis dapat menyelesaikan skripsi dengan baik.

Skripsi ini merupakan salah satu persyaratan akademis yang harus ditempuh oleh setiap mahasiswa Program Studi Teknik Informatika Universitas Pancasila guna memperoleh gelar sarjana. Skripsi ini disusun berdasarkan hasil penelitian dan observasi yang penulis lakukan di PT. ABC, khususnya di bagian divisi Human Resource dan juga jurnal-jurnal ilmiah yang berkaitan dengan judul skripsi.

Dengan selesainya laporan skripsi ini tidak terlepas dari bantuan banyak pihak yang telah memberikan masukan-masukan kepada penulis. Untuk itu penulis mengucapkan banyak terima kasih kepada:

1. Ibu Dr. Ionia Veritawati S.Si., MT., selaku Ketua Program Studi Teknik Informatika Fakultas Teknik Universitas Pancasila.
2. Bapak Amir Murtako, S.Kom., M.Kom, selaku Koordinator Skripsi Program Studi Teknik Informatika Fakultas Teknik Universitas Pancasila.
3. Bapak Dr. Iman Paryudi, S.Si, M.Kom, selaku pembimbing skripsi.
4. Tim skripsi Program Studi Teknik Informatika Universitas Pancasila tahun 2023 yang tidak bisa disebutkan satu persatu.
5. Bapak Marhaena Heronika, selaku Chief Executive Officer PT. ABC.
6. Ibu Putri Calista Wati dari Departemen *Human Resource Management* PT. ABC, sebagai narasumber di lapangan yang telah membantu dalam mengumpulkan data*.*
7. Jajaran manajemen PT. ABC, yang telah memberikan kelancaran dan pengertian ketika harus izin selama menyusun skripsi ini.
8. Rekan pengembang PT. ABC yang telah memberikan masukan dan bantuan teknis selama proses penelitian.
9. Orangtua yang saya cintai dan rindukan, yang membuat saya selalu termotivasi untuk menyelesaikan studi.
10. Febrian dan Icha yang selalu mendengarkan keluh kesah dan memotivasi saya selama menyusun skripsi ini.
11. Seluruh pihak yang telah terlibat dalam kelancaran skripsi ini dan belum disebutkan di atas, penulis ucapkan terima kasih.

Penulis menyadari bahwa masih banyak kekurangan dari laporan ini, baik dari materi maupun teknik penyajiannya, mengingat kurangnya pengetahuan dan pengalaman penulis. Oleh karena itu, kritik dan saran yang membangun sangat penulis harapkan.

*Wassalammu’alaikum, Wr.Wb.*

Jakarta, 2 Juni 2023

Mohamad Febrian Mosii

# DAFTAR ISI

[LEMBAR PERSETUJUAN i](#_Toc136636632)

[LEMBAR KEASLIAN ii](#_Toc136636633)

[KATA PENGANTAR iii](#_Toc136636634)

[DAFTAR ISI v](#_Toc136636635)

[DAFTAR TABEL vii](#_Toc136636636)

[DAFTAR GAMBAR viii](#_Toc136636637)

[BAB I PENDAHULUAN 1](#_Toc136636638)

[1.1 Latar Belakang 1](#_Toc136636639)

[1.2 Rumusan Masalah 5](#_Toc136636640)

[1.3 Tujuan dan Manfaat Penelitian 6](#_Toc136636641)

[1.3.1 Tujuan Penelitian 6](#_Toc136636642)

[1.3.2 Manfaat Penelitian 7](#_Toc136636643)

[1.4 Batasan Masalah 9](#_Toc136636644)

[1.5 Metodologi Penelitian 9](#_Toc136636645)

[1.5.1 Tahapan Pengumpulan Data 10](#_Toc136636646)

[1.5.2 Tahapan Analisis Data 10](#_Toc136636647)

[1.5.3 Tahapan Penyajian Data 10](#_Toc136636648)

[1.6 Sistematika Penulisan 12](#_Toc136636649)

[1.7 Kerangka Pemikiran 14](#_Toc136636650)

[BAB II LANDASAN TEORI 15](#_Toc136636651)

[BAB III ANALISA SISTEM SEDANG BERJALAN 15](#_Toc136636652)

[BAB IV PERANCANGAN SISTEM 16](#_Toc136636653)

[BAB V IMPLEMENTASI 17](#_Toc136636654)

[BAB VI KESIMPULAN DAN SARAN 18](#_Toc136636655)

[DAFTAR PUSTAKA 19](#_Toc136636656)

# DAFTAR TABEL

# DAFTAR GAMBAR

**ABSTRAK**

**ABSTRACT**

# BAB I PENDAHULUAN

## Latar Belakang

(Keadaan Saat Ini)

CV Himalaya Teknologi, yang beralama di Jl. Persada Alam Cibinong Blk. B1 No.18, Pd. Rajeg, Kec. Cibinong, Kabupaten Bogor, Jawa Barat 16914 memiliki sebuah produk bernama Gasmaj, sebuah aplikasi pengelolaan masjid berbayar yang menargetkan puluhan hingga ratusan masjid yang terintegrasi. Ada pun layanan Gasmaj salah satunya adalah menyediakan sebuah fitur pembayaran zakat atau infaq secara online serta memasarkan produk perlengkapan ibadah yang dapat dijual baik secara nasional mau pun internasional.

Saat ini aplikasi masih dalam tahapan pengembangan dan pemilik usaha masih belum terpikirkan untk menggunakan sistem payment gateway untuk menerima dan mengelola pembayaran, pemilik usaha ingin menggunakandengan cara melakukan konfirmasi pembayaran secara manual menggunakan WhatsApp, cara ini tentu sangat kurang efisien jika sudah banyak client yang terintegrasi dengan Gasmaj.

(Tantangan jika tidak menggunakan **Payment Gateway**)

Jika tidak memiliki channel pembayaran menggunakan Payment Gateway dapat menjadi tantangan yang signifikan karena Payment Gateway memainkan peran penting dalam memfasilitasi transaksi online dengan menyediakan metode pembayaran yang aman dan efisien. Tanpa Payment Gateway, Gasmaj akan kehilangan kemampuan untuk menerima pembayaran secara langsung melalui berbagai metode pembayaran populer seperti kartu kredit, kartu debit, transfer bank, atau dompet digital.

Tanpa channel pembayaran yang disediakan oleh Payment Gateway, Gasmaj mungkin terbatas dalam memenuhi preferensi pembayaran pelanggan. Pelanggan modern cenderung mencari kemudahan dan keamanan dalam bertransaksi online. Dengan tidak adanya Payment Gateway yang terintegrasi dengan Gasmaj, Gasmaj mungkin kehilangan peluang untuk menjangkau dan memenuhi kebutuhan pelanggan yang menginginkan opsi pembayaran yang mudah atau beragam.

Selain itu, Gasmaj juga harus menghadapi tantangan dalam mengelola proses pembayaran secara manual. Ini termasuk mengumpulkan informasi pembayaran dari pelanggan, memverifikasi pembayaran, dan melakukan aksi secara manual. Mengelola aspek pembayaran ini secara manual dapat memakan waktu dan sumber daya yang berharga, mengganggu fokus Anda pada pengembangan bisnis dan pengalaman pelanggan secara keseluruhan.

Selain keterbatasan operasional, tidak memiliki channel pembayaran yang terintegrasi juga dapat mempengaruhi kepercayaan pelanggan terhadap Gasmaj. Dalam era digital ini, keamanan pembayaran menjadi perhatian utama bagi pelanggan. Dengan tidak adanya sistem pembayaran yang terintegrasi, pelanggan mungkin merasa kurang percaya untuk memasukkan informasi pembayaran mereka secara manual, meningkatkan risiko kehilangan pelanggan.

(Tantangan jika tidak menggunakan **Payment Gateway Aggregator**)

Selain menggunakan sistem terintegrasi *Payment Gateway*, skema bisnis yang akan dijalankan oleh Gasmaj dimana setiap *merchant* atau *client* juga dapat berjualan secara online akan menghadapi beberapa tantangan. Meskipun menggunakan Payment Gateway sudah cukup memungkinkan Gasmaj untuk menerima pembayaran, tetapi terdapat keterbatasan dalam variasi metode pembayaran yang dapat ditawarkan. Setiap *Payment Gateway* biasanya memiliki beberapa opsi pembayaran seperti kartu kredit, transfer bank, atau dompet digital. Namun, jika hanya bergantung pada satu Payment Gateway, tetap ada batasan ketersediaan metode pembayaran, sebagai contoh Payment Gateway Midtrans menyediakan pembayaran dompet digital Gopay, namun tidak memiliki metode pembayaran OVO, dan sebaliknya, untuk Payment Gateway Brankas memiliki metode pembayaran OVO tapi tidak memiliki metode pembayaran Gopay. Hal ini dapat mengurangi fleksibilitas dalam memenuhi preferensi pembayaran pelanggan.

Gasmaj akan menghadapi tantangan dalam mengelola integrasi Payment Gateway secara terpisah untuk setiap metode pembayaran yang disediakan. Setiap Payment Gateway memiliki persyaratan teknis yang berbeda, dan mengelola masing-masing Payment Gateway secara terpisah membutuhkan waktu dan upaya yang tidak sedikit. Selain itu, jika Gasmaj ingin menambahkan atau mengubah metode pembayaran yang ada, Gasmaj akan menghadapi kompleksitas tambahan dalam mengelola integrasi tersebut. Hal ini dapat meningkatkan kompleksitas administratif CV Himalaya Teknologi.

Tanpa Payment Gateway Aggregator, Gasmaj juga akan menghadapi keterbatasan dalam membandingkan tarif dan biaya antara Payment Gateway. Payment Gateway Aggregator adalah sebuah platform atau layanan yang mengintegrasikan multiple Payment Gateway atau metode pembayaran ke dalam satu antarmuka terpusat. Agregator ini bertindak sebagai perantara antara Gasmaj dan berbagai Payment Gateway yang berbeda, sehingga pemilik usaha tidak perlu mengelola integrasi secara terpisah. Dengan menggunakan Payment Gateway Aggregator, Gasmaj dapat menerima berbagai metode pembayaran seperti kartu kredit, transfer bank, dompet digital, atau metode pembayaran lainnya yang ditawarkan oleh berbagai Payment Gateway yang terhubung.

Setiap Payment Gateway biasanya memiliki struktur biaya yang berbeda, termasuk biaya transaksi, biaya langganan, atau biaya tambahan lainnya. Tanpa kemampuan untuk membandingkan tarif dan biaya secara menyeluruh, Gasmaj mungkin tidak dapat mengoptimalkan pengeluaran dan memaksimalkan keuntungan bisnisnya. Pengelolaan beberapa Payment Gateway secara terpisah juga akan meningkatkan tingkat kompleksitas dalam berbisnis, termasuk menangani pembaruan teknis, perubahan kebijakan, dan permasalahan integrasi. Hal ini dapat menghabiskan waktu dan sumber daya yang berharga yang seharusnya dapat diarahkan untuk mengembangkan bisnis secara keseluruhan.

(Solusi yang ditawarkan)

Dalam rangka membangun dan mengembangkan layanan berbayar *online* yang sukses, peneliti menyarankan kepada pihak CV Himalaya Teknologi untuk mempertimbangkan Gasmaj melakukan integrasi dengan Payment Gateway Aggregator.

Secara keseluruhan, Payment Gateway Aggregator memberikan solusi yang efisien dan terintegrasi, memungkinkan pemilik usaha untuk mengelola dan memperluas pilihan metode pembayaran dengan lebih mudah, mengoptimalkan pengeluaran, dan meningkatkan kepuasan pelanggan.

## Rumusan Masalah

Berdasarkan latar belakang masalah yang telah diuraikan diatas, maka rumusan masalah pada penelitian ini adalah:

1. Bagaimana penggunaan payment aggregator dapat meningkatkan efisiensi operasional dalam pengelolaan pembayaran di aplikasi Gasmaj CV Himalaya Teknologi?
2. Bagaimana payment aggregator dapat memperluas pilihan metode pembayaran dan memenuhi preferensi pembayaran pelanggan dalam aplikasi Gasmaj?
3. Bagaimana penggunaan payment aggregator dapat mengurangi kompleksitas dan waktu yang diperlukan dalam mengelola integrasi payment gateway secara terpisah untuk setiap metode pembayaran dalam Gasmaj?
4. Bagaimana penggunaan payment aggregator dapat meningkatkan kepercayaan pelanggan terhadap aplikasi Gasmaj melalui penggunaan sistem pembayaran yang terintegrasi dan aman?
5. Bagaimana penggunaan payment aggregator dapat membantu CV Himalaya Teknologi dalam membandingkan tarif dan biaya antara payment gateway, serta mengoptimalkan pengeluaran dan keuntungan bisnisnya?
6. Bagaimana penggunaan payment aggregator dapat menyederhanakan pengelolaan administratif dan mengatasi permasalahan teknis, kebijakan, dan integrasi yang dihadapi oleh CV Himalaya Teknologi dalam pengelolaan pembayaran aplikasi Gasmaj?

## Tujuan dan Manfaat Penelitian

Berdasarkan pemaparan yang sudah dijelaskan di atas, berikut adalah tujuan dan manfaat dari penelitian ini.

### Tujuan Penelitian

Berikut adalah beberapa tujuan penelitian berdasarkan masalah-masalah yang telah diidentifikasi:

1. Menganalisis dampak penggunaan sistem payment aggregator terhadap efisiensi operasional dalam pengelolaan pembayaran di aplikasi Gasmaj CV Himalaya Teknologi.
2. Menjelajahi cara penggunaan payment aggregator dapat memperluas pilihan metode pembayaran dan memenuhi preferensi pembayaran pelanggan dalam aplikasi Gasmaj.
3. Mengevaluasi bagaimana penggunaan payment aggregator dapat mengurangi kompleksitas dan waktu yang diperlukan dalam mengelola integrasi payment gateway secara terpisah untuk setiap metode pembayaran dalam Gasmaj.
4. Menginvestigasi pengaruh penggunaan payment aggregator terhadap kepercayaan pelanggan terhadap aplikasi Gasmaj melalui penggunaan sistem pembayaran yang terintegrasi dan aman.
5. Menganalisis dampak penggunaan payment aggregator terhadap pengeluaran dan keuntungan bisnis CV Himalaya Teknologi melalui pembandingan tarif dan biaya antara payment gateway.
6. Menjelajahi bagaimana penggunaan payment aggregator dapat menyederhanakan pengelolaan administratif dan mengatasi permasalahan teknis, kebijakan, dan integrasi yang dihadapi oleh CV Himalaya Teknologi dalam pengelolaan pembayaran aplikasi Gasmaj.

### Manfaat Penelitian

Berikut adalah beberapa manfaat penelitian dari masalah yang telah diidentifikasi di atas:

#### Manfaat Akademis

1. Meningkatkan pemahaman tentang implementasi sistem payment aggregator dalam pengelolaan pembayaran aplikasi berbasis bisnis.
2. Menyediakan wawasan tentang tantangan dan konsekuensi yang mungkin dihadapi jika tidak menggunakan payment aggregator dalam konteks aplikasi berbayar.
3. Memberikan perspektif baru dalam memahami peran payment aggregator sebagai solusi efisien dalam pengelolaan pembayaran online.
4. Mengidentifikasi manfaat penggunaan payment aggregator dalam memenuhi preferensi pembayaran pelanggan dan meningkatkan kepuasan pelanggan.
5. Menganalisis dampak penggunaan payment aggregator terhadap efisiensi operasional dan pengeluaran bisnis.
6. Menyediakan panduan praktis untuk implementasi dan penggunaan payment aggregator dalam aplikasi berbayar.
7. Menyediakan pemahaman tentang kepentingan keamanan dan kepercayaan pelanggan dalam sistem pembayaran online.
8. Menyelidiki perbedaan antara menggunakan payment aggregator dan tidak menggunakan payment aggregator dalam hal integrasi payment gateway dan manfaatnya.
9. Menyediakan kerangka konseptual yang dapat digunakan sebagai referensi dalam penelitian lanjutan tentang penggunaan payment aggregator dalam pengelolaan pembayaran.
10. Menawarkan pemahaman yang lebih luas tentang tantangan dan solusi terkait dengan administrasi, kebijakan, dan permasalahan teknis dalam pengelolaan pembayaran aplikasi berbayar.

#### Manfaat Praktis

Penelitian ini juga memiliki kegunaan bagi pihak-pihak yang mengembangkan atau menggunakan hasil dari penelitian ini, yaitu:

##### Bagi Pengembang

1. Hasil dari penelitian dapat digunakan sebagai bahan masukkan sebagai bahan masukan CV Himalaya Teknologi.
2. Meningkatkan wawasan pengembang terkait implementasi sistem payment aggregator dalam pengelolaan pembayaran aplikasi berbasis bisnis.
3. Hasil dari program dan aplikasi yang dibuat akan dapat dikembangkan kembali, sehingga dapat menjadi peluang bisnis bagi pengembang.

##### Bagi Perusahaan

* 1. Dengan menggunakan payment aggregator, CV Himalaya Teknologi dapat mengoptimalkan pengeluaran dan meminimalkan biaya yang terkait dengan integrasi payment gateway.
  2. Dengan adanya pilihan metode pembayaran yang lebih luas melalui payment aggregator, CV Himalaya Teknologi dapat memenuhi preferensi pembayaran pelanggan.
  3. Dengan menggunakan payment aggregator, CV Himalaya Teknologi dapat menawarkan layanan pembayaran yang lebih efisien dan beragam kepada pelanggan.
  4. Dengan kemampuan untuk membandingkan tarif dan biaya antara payment gateway yang terhubung melalui payment aggregator, CV Himalaya Teknologi dapat mengoptimalkan pengeluaran dan memaksimalkan keuntungan bisnisnya.

## Batasan Masalah

Batasan masalah pada penellitian ini adalah:

## Metodologi Penelitian

Dalam upaya penyusunan penulisan ini, diperlukan pengolahan data yang akurat. Oleh karena itu, penulis telah menerapkan beberapa metode pengumpulan data guna memperoleh informasi yang relevan. Beberapa teknik pengumpulan dan pengembangan sistem yang telah digunakan dalam penelitian ini mencakup:

### Tahapan Pengumpulan Data

Tahapan pengumpulan data akan menggunakan teknik pengumpulan dari studi literatur, observasi, dan wawancara. Studi literatur digunakan untuk mendapatkan teori dan informasi yang terkait dengan penelitian ini. Wawancara digunakan untuk mendapatkan informasi yang lebih mendalam dari responden.

### Tahapan Analisis Data

Analisis data yang digunakan dalam penelitian ini adalah analisis statistik deskriptif dan inferensial. Analisis statistik deskriptif digunakan untuk menjelaskan karakteristik dan profil pengguna aplikasi. Analisis statistik inferensial digunakan untuk menguji hipotesis mengenai pengaruh penggunaan aplikasi terhadap kualitas dan efektivitas pembayaran *payment gateway aggregator*.

### Tahapan Penyajian Data

Berikut adalah tahapan-tahapan dalam penyajian data yang akan diimplementasi pada penelitian ini:

#### Analisis Kebutuhan Pengguna

Melakukan wawancara dengan calon pengguna untuk memahami kebutuhan dan harapan mereka terhadap pengelolaan pembayaran *payment gateway aggregator*. Hal ini dapat dilakukan dengan menggunakan pendekatan kualitatif seperti wawancara mendalam.

#### Analisis Platform Payment Gateway yang Tersedia

Melakukan analisis terhadap *platform payment gateway* yang tersedia dan menentukan *platform* yang paling sesuai dengan kebutuhan pengguna. Dalam analisis ini, dapat digunakan metode-metode kuantitatif seperti pemilihan dan penilaian kriteria berdasarkan bobot yang telah ditentukan.

#### Desain Antarmuka Aplikasi

Merancang antarmuka aplikasi yang mudah digunakan dan sesuai dengan kebutuhan pengguna. Pada tahap ini, dapat digunakan metode desain user *experience* (UX) untuk memastikan antarmuka aplikasi yang dihasilkan dapat memberikan pengalaman pengguna yang baik.

#### Implementasi Algoritma Pengelolaan Data Transaksi

Implementasi algoritma yang sesuai untuk meningkatkan efisiensi pengelolaan data transaksi.

#### Uji Coba Aplikasi

Melakukan uji coba aplikasi untuk memastikan kualitas dan efektivitas aplikasi. Uji coba dapat dilakukan dengan menggunakan metode *usability* testing untuk memastikan antarmuka aplikasi dapat digunakan dengan baik oleh pengguna, serta *performance* testing untuk memastikan kecepatan dan stabilitas aplikasi.

#### Evaluasi dan Pengembangan

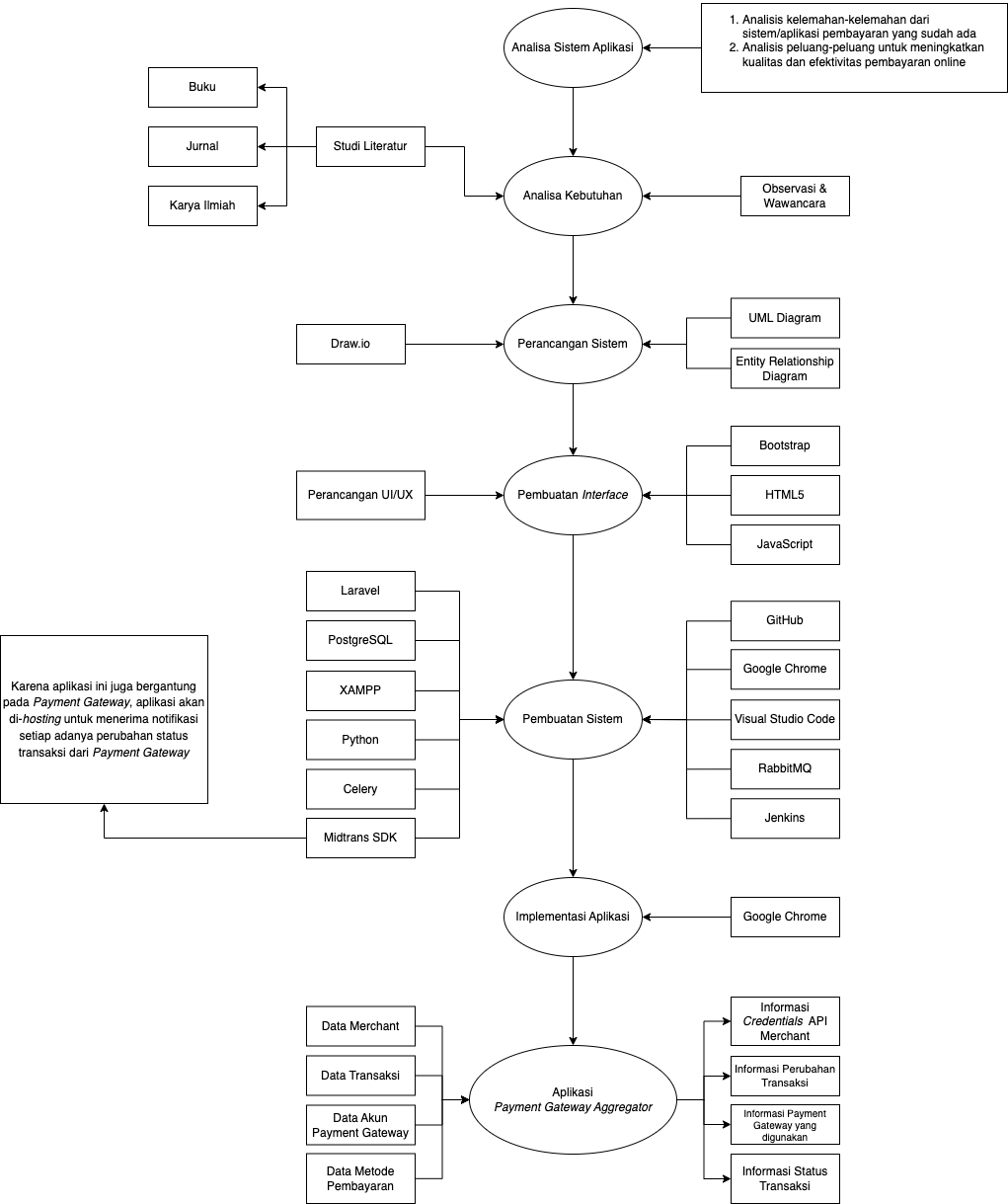
Melakukan evaluasi terhadap hasil implementasi dan pengembangan aplikasi secara terus-menerus untuk memastikan aplikasi selalu dapat memenuhi kebutuhan pengguna dan memberikan pengalaman pengguna yang baik. Evaluasi dapat dilakukan dengan menggunakan metode *feedback* dari pengguna atau *analytics* data dari penggunaan aplikasi.

## Sistematika Penulisan

Penulisan skripsi ini tersusun dari 5 (lima) bab dengan sistematika penulisan sebagai berikut :

|  |  |
| --- | --- |
| **BAB I** | **PENDAHULUAN** |
| Pada bab ini, penulis membahas tentang Latar Belakang Masalah, Rumusan Masalah, Tujuan dan Manfaat Penelitian, Batasan Masalah, Metodologi Penelitian, Sistematika Penulisan, dan Kerangka Pemikiran. |
| **BAB II** | **LANDASAN TEORI** |
|  | Pada bab ini, akan dijelaskan teori-teori yang menjadi dasar dalam implementasi payment aggregator guna meningkatkan efisiensi operasional, memperluas pilihan metode pembayaran, dan mengoptimalkan pengeluaran CV Himalaya Teknologi.. |
| **BAB III** | **ANALISA SISTEM** |
|  | Dalam bab ini, akan dilakukan analisis terperinci mengenai sistem payment aggregator yang dapat membantu CV Himalaya Teknologi dalam mengatasi tantangan operasional, meningkatkan kepuasan pelanggan, dan memaksimalkan efisiensi pengeluaran. |
| **BAB IV** | **PERANCANGAN SISTEM** |
|  | Pada bab ini, akan dijelaskan tentang desain arsitektur sistem, integrasi dengan payment gateway, pengelolaan metode pembayaran, fitur keamanan, dan antarmuka pengguna. |
| **BAB V** | **IMPLEMENTASI** |
|  | Pada bab ini, akan dijelaskan langkah-langkah yang dilakukan dalam mengimplementasikanaplikasi, termasuk konfigurasi teknis, integrasi dengan payment gateway dan uji coba sistem. |
| **BAB VI** | **KESIMPULAN DAN SARAN** |
|  | Dalam bab ini, akan dilakukan analisis terperinci mengenai sistem payment aggregator yang dapat membantu CV Himalaya Teknologi dalam mengatasi tantangan operasional, meningkatkan kepuasan pelanggan, dan memaksimalkan efisiensi pengeluaran. |

## Kerangka Pemikiran

**

# BAB II LANDASAN TEORI

## Payment Gateway

Payment gateway adalah sebuah sistem yang memfasilitasi dan mengelola proses pembayaran secara elektronik antara penjual (merchant) dan pembeli (customer) dalam transaksi online. Payment gateway bertindak sebagai perantara antara berbagai metode pembayaran seperti kartu kredit, kartu debit, transfer bank, dompet digital, dan lain-lain.

Fungsi utama payment gateway adalah memproses transaksi pembayaran dengan aman dan efisien. Ketika pelanggan melakukan pembelian di situs web atau aplikasi, informasi pembayaran mereka akan dikirim melalui payment gateway untuk verifikasi dan otentikasi. Setelah itu, payment gateway akan melakukan pemrosesan transaksi dengan pihak bank atau penyedia layanan pembayaran terkait. Jika transaksi diterima, pembayaran akan berhasil dan penjual akan menerima konfirmasi pembayaran.

Payment gateway juga menyediakan fitur keamanan untuk melindungi data pembayaran pelanggan. Informasi sensitif seperti nomor kartu kredit atau data rekening bank dienkripsi agar tidak dapat diakses oleh pihak yang tidak berwenang. Payment gateway juga dapat mengimplementasikan protokol keamanan seperti Secure Sockets Layer (SSL) untuk menjaga keamanan transmisi data.

Selain itu, payment gateway juga menyediakan laporan dan rekapan transaksi bagi penjual. Hal ini memudahkan penjual dalam melacak dan mengelola transaksi, serta memantau arus kas dan keuntungan bisnis mereka.

Dengan adanya payment gateway, transaksi online menjadi lebih mudah, aman, dan efisien. Payment gateway memungkinkan penjual untuk menerima pembayaran dari berbagai metode pembayaran yang berbeda, sehingga meningkatkan fleksibilitas dan kemudahan bagi pelanggan dalam bertransaksi secara elektronik.

## Payment Gateway Aggregator

Payment gateway aggregator adalah sebuah platform atau layanan yang mengintegrasikan multiple payment gateway atau metode pembayaran ke dalam satu antarmuka terpusat. Agregator ini bertindak sebagai perantara antara penjual (merchant) dan berbagai payment gateway yang berbeda, sehingga penjual tidak perlu mengelola integrasi secara terpisah dengan setiap payment gateway.

Fungsi utama payment gateway aggregator adalah menyediakan solusi yang terintegrasi untuk menerima pembayaran dari berbagai metode pembayaran yang ditawarkan oleh payment gateway yang terhubung. Dengan menggunakan payment gateway aggregator, penjual dapat menerima berbagai jenis pembayaran seperti kartu kredit, kartu debit, transfer bank, dompet digital, dan lain-lain, tanpa perlu melakukan integrasi secara terpisah dengan setiap payment gateway.

fKeuntungan menggunakan payment gateway aggregator adalah meningkatnya fleksibilitas dalam memenuhi preferensi pembayaran pelanggan. Penjual dapat menawarkan beragam opsi pembayaran kepada pelanggan mereka, tanpa harus terbatas pada satu jenis payment gateway atau metode pembayaran tertentu. Hal ini memungkinkan penjual untuk menjangkau lebih banyak pelanggan dan meningkatkan pengalaman berbelanja online.

Selain itu, payment gateway aggregator juga memudahkan penjual dalam mengelola proses pembayaran secara efisien. Penjual hanya perlu mengintegrasikan sistem dengan payment gateway aggregator, sehingga mengurangi kompleksitas administratif dan teknis dalam mengelola integrasi dengan setiap payment gateway secara terpisah. Payment gateway aggregator juga dapat menyediakan laporan transaksi yang terpusat, memudahkan penjual dalam melacak dan mengelola pembayaran.

Dengan adanya payment gateway aggregator, penjual dapat memanfaatkan keuntungan dari berbagai metode pembayaran yang ditawarkan oleh berbagai payment gateway, meningkatkan fleksibilitas, efisiensi, dan kepuasan pelanggan dalam proses pembayaran secara online.

## Midtrans

Midtrans adalah salah satu perusahaan payment gateway yang terkemuka di Indonesia. Mereka menyediakan solusi pembayaran online yang aman dan terpercaya untuk bisnis e-commerce dan layanan digital. Midtrans memungkinkan penjual untuk menerima pembayaran melalui berbagai metode, seperti kartu kredit, kartu debit, transfer bank, dan dompet digital.

Sebagai payment gateway, Midtrans memiliki fitur-fitur yang memudahkan proses pembayaran, seperti integrasi yang mudah dengan platform e-commerce, kemampuan untuk mengelola transaksi, pemantauan real-time, serta sistem keamanan yang canggih untuk melindungi data pembayaran pelanggan. Midtrans juga menyediakan dashboard atau antarmuka yang intuitif bagi penjual untuk melihat dan mengelola riwayat transaksi serta laporan keuangan.

Selain itu, Midtrans juga telah terintegrasi dengan berbagai penyedia layanan finansial, seperti bank-bank lokal di Indonesia dan dompet digital populer, seperti OVO dan Gopay. Hal ini memberikan fleksibilitas kepada penjual dalam menerima pembayaran dari berbagai metode yang digunakan oleh pelanggan.

Dengan menggunakan Midtrans sebagai payment gateway, penjual dapat meningkatkan kemudahan, keamanan, dan efisiensi dalam proses pembayaran online. Midtrans telah membangun reputasi yang kuat dalam industri pembayaran digital di Indonesia dan menjadi pilihan yang populer bagi banyak bisnis dalam mengelola transaksi pembayaran mereka.

## Brankas

Brankas adalah sebuah perusahaan fintech yang menyediakan layanan payment gateway di Indonesia. Sebagai payment gateway, Brankas memungkinkan penjual untuk menerima pembayaran secara online melalui berbagai metode, seperti kartu kredit, kartu debit, transfer bank, dan dompet digital.

Salah satu keunggulan Brankas adalah kemampuannya dalam menyediakan metode pembayaran melalui dompet digital, seperti OVO. Hal ini memungkinkan pelanggan untuk melakukan pembayaran dengan mudah menggunakan saldo dalam dompet digital mereka.

Brankas juga memiliki antarmuka atau dashboard yang intuitif bagi penjual, sehingga mereka dapat dengan mudah mengelola transaksi, memantau pembayaran, dan mengakses laporan keuangan. Selain itu, Brankas menyediakan fitur keamanan yang canggih untuk melindungi data pembayaran pelanggan dan memastikan transaksi berjalan dengan aman.

Dengan menggunakan Brankas sebagai payment gateway, penjual dapat memperluas opsi pembayaran yang mereka tawarkan kepada pelanggan, meningkatkan kemudahan bertransaksi, serta memperkuat kepercayaan dan keamanan dalam proses pembayaran online.

Brankas telah menjadi salah satu pemain utama dalam industri payment gateway di Indonesia dan terus berkembang dengan menyediakan solusi pembayaran yang inovatif dan terpercaya bagi bisnis-bisnis di berbagai sektor.

## Stripe

Stripe adalah sebuah perusahaan fintech yang menyediakan layanan payment gateway yang populer di tingkat global. Mereka memungkinkan penjual untuk menerima pembayaran secara online melalui berbagai metode, termasuk kartu kredit, kartu debit, dan dompet digital.

Salah satu keunggulan utama Stripe adalah kemudahan penggunaannya. Mereka menyediakan integrasi yang mudah dengan platform e-commerce dan API yang lengkap untuk pengembang. Hal ini memungkinkan penjual untuk dengan cepat mengintegrasikan Stripe ke dalam toko online mereka dan mulai menerima pembayaran dengan sedikit kerumitan teknis.

Stripe juga menonjol dalam hal keamanan dan kepatuhan. Mereka memiliki standar keamanan yang tinggi untuk melindungi data pembayaran pelanggan dan menerapkan langkah-langkah untuk memastikan kepatuhan terhadap peraturan keuangan, seperti Standar Keamanan Data Kartu Pembayaran (PCI DSS).

Selain itu, Stripe menawarkan berbagai fitur tambahan yang berguna, seperti kemampuan untuk mengatur penagihan berulang (recurring billing), pembayaran dalam mata uang asing, serta kemampuan untuk melakukan analisis dan pelaporan transaksi secara terperinci.

Dengan menggunakan Stripe sebagai payment gateway, penjual dapat menghadirkan pengalaman pembayaran yang lancar, cepat, dan aman bagi pelanggan mereka. Stripe telah menjadi salah satu pilihan yang populer di kalangan bisnis e-commerce dan startup teknologi, memberikan solusi pembayaran yang handal dan inovatif di pasar global.

## API

API (*Application Programming Interface*) adalah sebuah mekanisme yang memungkinkan aplikasi untuk berinteraksi dengan aplikasi lain atau sistem lainnya. API dapat digunakan untuk mengambil data atau mengirimkan data dari satu aplikasi ke aplikasi lainnya. API dapat digunakan dengan berbagai protokol seperti HTTP, HTTPS, dan WebSocket.

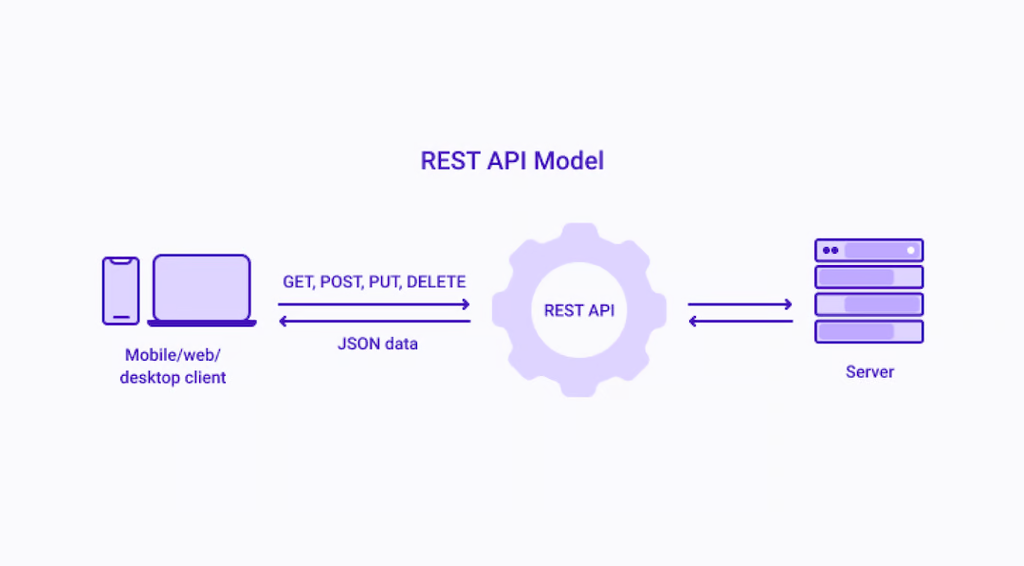
## Rest API

REST (*Representational State Transfer*) API adalah sebuah arsitektur yang digunakan untuk membuat aplikasi web yang terdistribusi. REST API menggunakan protokol HTTP untuk mengirimkan pesan antara klien dan server.

REST API menggunakan metode-metode HTTP seperti GET, POST, PUT, dan DELETE untuk mengambil data atau mengirimkan data ke server. REST API menggunakan *endpoint* (URL) untuk menentukan lokasi data yang akan diambil atau dikirimkan. REST API menggunakan format data seperti JSON atau XML untuk mengirimkan atau menerima data.

REST API mengikuti prinsip-prinsip arsitektur REST, yaitu *stateless*, *cacheable*, *client-server*, dan *layered system*. REST API sangat populer karena mudah digunakan dan dapat digunakan dengan berbagai platform dan bahasa pemrograman.

Pengguna aplikasi web sering menggunakan layanan web REST API untuk berkomunikasi satu sama lain. Misalnya, memperoleh dan meninjau detail akun dalam program media sosial . Peramban REST API dapat dilihat sebagai sintaks web. Pelanggan *online* menggunakan API web untuk menyediakan dan mengelola akses ke operasi digital seiring dengan meningkatnya penggunaan *cloud*.



Gambar 2 - Komunikasi menggunakan REST API

Merancang API web yang memungkinkan pelanggan untuk terhubung, mengelola, dan terlibat dengan layanan web digital secara dinamis dalam konteks yang tersebar adalah keputusan yang masuk akal. Situs web seperti Google, eBay, Facebook, Amazon, dan Twitter menggunakan layanan web RESTful. Aplikasi klien sekarang dapat menggunakan REST untuk mengakses layanan web ini.

## NGINX

NGINX adalah *web server* yang digunakan untuk mengelola *traffic* HTTP dan *reverse proxy* untuk aplikasi web. Ia juga dapat digunakan sebagai *load balancer* dan *content cache*. Nginx populer digunakan karena kinerja dan *skalabilitas* yang baik dibandingkan dengan *web server* lainnya. Berikut penjelasan masing-masing istilah yang digunakan pada NGINX.

1. ***Reverse Proxy***

Sebuah server yang ditempatkan di depan server web lainnya untuk menyediakan layanan tambahan seperti keamanan, pemuatan yang lebih baik, atau akses yang dikendalikan. *Reverse proxy* akan menerima permintaan dari klien dan meneruskannya ke server web yang sesuai. Ia juga dapat menyediakan enkripsi HTTPS, autentikasi pengguna, dan pemblokiran akses yang tidak diinginkan

1. ***Web Server***

Web server adalah perangkat atau aplikasi yang digunakan untuk menangani permintaan dari klien (seperti peramban web) dan menyediakan respon yang sesuai. Web server akan menerima permintaan dari klien berupa permintaan HTTP atau HTTPS dan mengirimkan respon yang sesuai, seperti halaman web, gambar, atau *file* lainnya. Web server juga dapat mengeksekusi skrip seperti PHP atau Python untuk menghasilkan konten dinamis yang disesuaikan dengan permintaan klien. Beberapa contoh web server yang populer adalah Apache, Nginx, dan IIS.

## PHP

Salah satu bahasa pemrograman *web* yang banyak digunakan adalah PHP. PHP merupakan bahasa pemrograman web yang bekerja di sisi server (*server-side*). Dengan menggunakan PHP dapat dibuat *website* yang dinamis, selain itu juga dapat 19 dibuat suatu aplikasi berbasis web. PHP merupakan bahasa pemrograman yang dikembangkan dari bahasa pemrograman Perl dan C++.

PHP pertama kali ditemukan pada 1995 oleh seorang *Software Developer* bernama Rasmus Lerdrof. Ide awal PHP adalah ketika itu Rasmus ingin mengetahui jumlah pengunjung yang membaca resume *online* miliknya. *script* yang dikembangkan baru dapat melakukan dua pekerjaan, yakni merekam informasi *visitor*, dan menampilkan jumlah pengunjung dari suatu *website*. Dan sampai sekarang kedua tugas tersebut masih tetap populer digunakan oleh dunia web saat ini. Kemudian, dari situ banyak orang di milis mendiskusikan *script* buatan Rasmus Lerdrof, hingga akhirnya Rasmus mulai membuat sebuah tool/script, bernama Personal Home Page (PHP).

Kebutuhan PHP sebagai *tool* yang serba guna membuat Lerdorf melanjutkan untuk mengembangkan PHP hingga menjadi suatu bahasa tersendiri yang mungkin dapat melakukan konversi data yang di input melalui Form HTML menjadi suatu *variable*, yang dapat dimanfaatkan oleh sistem lainnya. Untuk merealisasikannya, akhirnya Lerdrof mencoba mengembangkan PHP menggunakan bahasa C ketimbang menggunakan Perl. Tahun 1997, PHP versi 2.0 di rilis, dengan nama Personal Home Page Form Interpreter (PHP-FI). PHP Semakin populer, dan semakin diminati oleh *programmer* *web* dunia.

## October CMS

October CMS adalah sebuah sistem manajemen konten (CMS) yang bersifat open-source, dirancang untuk memudahkan pengembangan dan pengelolaan situs web. CMS ini dibangun dengan menggunakan framework PHP yang kuat dan fleksibel bernama Laravel. October CMS menawarkan antarmuka yang intuitif dan mudah digunakan, serta menyediakan berbagai fitur dan fungsi yang dapat disesuaikan sesuai kebutuhan pengguna.

Salah satu keunggulan October CMS adalah arsitektur yang modular, yang memungkinkan pengguna untuk menginstal dan mengaktifkan hanya modul atau plugin yang diperlukan, tanpa membebani situs dengan fitur yang tidak digunakan. Hal ini memberikan fleksibilitas dan efisiensi dalam mengelola konten dan fungsionalitas situs. October CMS juga memiliki sistem manajemen tema yang kuat, yang memungkinkan pengguna untuk dengan mudah membuat dan mengubah tampilan situs web sesuai dengan keinginan mereka. Pengguna dapat mengatur tata letak, gaya, dan elemen-elemen desain lainnya dengan cepat dan mudah. Selain itu, October CMS memiliki kemampuan untuk mengelola konten dinamis, seperti posting blog, halaman produk, dan konten lainnya. Pengguna dapat dengan mudah membuat dan mengatur konten ini melalui antarmuka pengguna yang intuitif.

Keamanan juga menjadi fokus utama October CMS. CMS ini menyediakan fitur keamanan yang kuat, seperti otentikasi pengguna, pengaturan izin akses, dan perlindungan terhadap serangan keamanan umum. Ini membantu menjaga situs web tetap aman dan melindungi data sensitif. Dalam keseluruhan, October CMS merupakan CMS yang sangat cocok untuk pengembangan situs web yang fleksibel, modular, dan dapat disesuaikan. Dengan antarmuka yang mudah digunakan dan fitur-fitur yang kuat, October CMS menjadi pilihan yang baik bagi pengembang dan pengguna yang ingin memiliki kendali penuh atas tampilan dan fungsionalitas situs web mereka.

## HTML

HTML (Hypertext Markup Language) adalah bahasa *markup* yang digunakan untuk membuat halaman web. HTML menyediakan *tag* yang digunakan untuk menentukan struktur dan format halaman web. HTML juga digunakan untuk menambahkan gambar, video, audio, dan konten lainnya.

## CMS

CMS (Content Management System) adalah perangkat lunak yang digunakan untuk mengelola, mengedit, dan mempublikasikan konten di situs web. CMS memungkinkan pengguna yang tidak memiliki keahlian teknis untuk mengelola konten situs web tanpa harus mengetahui bahasa pemrograman seperti HTML atau CSS. Beberapa contoh CMS yang populer adalah WordPress, Joomla, dan Drupal. CMS juga dapat dikelompokkan menjadi dua kategori yaitu *Open Source* dan *Proprietary*, yang pertama dapat diakses, dimodifikasi, dan digunakan oleh siapa pun, sedangkan yang kedua dikembangkan oleh perusahaan tertentu dan hanya dapat digunakan dengan lisensi yang dibeli.

## JQuery

JQuery adalah sebuah *library* JavaScript yang digunakan untuk membuat pengembangan JavaScript lebih mudah. JQuery menyediakan fungsi yang berguna untuk melakukan manipulasi DOM, menangani *event*, dan melakukan animasi. JQuery juga menyediakan sebuah mekanisme untuk melakukan pemanggilan ajax, yang memungkinkan untuk mengambil dan mengirim data tanpa harus memuat ulang halaman.

JQuery merupakan salah satu dari sekian banyak *framework* yang digunakan dan memiliki pengguna yang paling banyak. JQuery merupakan pustaka JavaScript yang dibangun untuk mempercepat dan memperingkas serta menyederhanakan manipulasi dokumen HTML, penanganan event, animasi, dan interaksi Ajax untuk mempercepat pengembangan web. Dengan JQuery, developer akan dimanjakan dengan suatu pemrograman JavaScript yang sangat sederhana jika dibandingkan dengan native JavaScript. Kemampuan yang dimiliki pustaka JQuery antara lain.

## Database Management System (DBMS)

DBMS adalah sebuah program perangkat lunak yang memungkinkan pengguna untuk menciptakan, mengelola, dan mengakses basis data. DBMS menyediakan antarmuka yang digunakan untuk mengakses data yang disimpan dalam basis data, seperti memasukkan data baru, mengubah data yang sudah ada, dan mengambil data yang diperlukan. Basis data yang dikelola oleh DBMS dapat digunakan oleh berbagai aplikasi dan digunakan oleh banyak pengguna sekaligus.

## Merchant

*Merchant* adalah perusahaan atau individu yang menjual barang atau jasa kepada konsumen. Dalam dunia perbankan dan keuangan, *merchant* adalah perusahaan yang menerima pembayaran dari konsumen melalui kartu kredit atau debit. Merchant dapat beroperasi di berbagai bidang, seperti retail, *e-commerce*, jasa, dll. *Merchant* dapat menjual produk atau jasa secara *online* atau *offline*, dalam bentuk fisik atau digital.

## User

User aplikasi adalah orang yang menggunakan aplikasi atau *software*. User dapat menjadi pengguna akhir yang menggunakan aplikasi untuk tujuan pribadi atau bisnis, atau administrator yang bertanggung jawab untuk mengelola dan mengkonfigurasi aplikasi untuk pengguna lain. User dapat berinteraksi dengan aplikasi melalui antarmuka grafis atau perintah-perintah tertentu. Sebuah aplikasi dapat memiliki satu atau lebih *user*, dan setiap *user* dapat memiliki hak akses yang berbeda tergantung pada peran yang diberikan kepadanya.

# BAB III ANALISA SISTEM SEDANG BERJALAN

# BAB IV PERANCANGAN SISTEM

# BAB V IMPLEMENTASI

# BAB VI KESIMPULAN DAN SARAN

# DAFTAR PUSTAKA