

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **2.1 Tinjauan Perusahaan**

Tahap ini merupakan tahap peninjauan secara langsung terhadap tempat penelitian yaitu DM Sunthree yang membahas profil, sejarah singkat, dan logo.

##### **2.1.1 Profil**

DM Sunthree adalah bisnis yang berfokus pada top up digital yang mencakup berbagai layanan seperti di berbagai game dan voucher. Selain itu, toko ini juga membuka peluang kepada *reseller* untuk melakukan top up dengan harga lebih murah dari umumnya. DM Sunthree didirikan pada pertengahan tahun 2020 di Jl. Cijawura Hilir RT 04 RW 12, Cijaura, Kecamatan Buahbatu, Kota Bandung, Jawa Barat.

##### **2.1.2 Sejarah singkat**

DM Sunthree merupakan bisnis penjualan Top Up di bidang game. DM Sunthree ini pada awalnya didirikan pada pertengahan tahun 2020 dengan nama Kang Sun3 untuk memenuhi kebutuhan remaja setempat sebagai penyedia layanan Top Up game, karena pada tahun tersebut peningkatan remaja setempat untuk bermain game cukup meningkat dan belum adanya tempat untuk melakukan top up. Seiring berjalannya waktu, pada tahun 2022 nama bisnis Kang Sun3 diubah menjadi DM Sunthree dan menerima pendapat dari konsumen untuk menyediakan top up di luar game seperti top up voucher game seperti Steam, Garena, dan Razer. Oleh karena itu, pada tahun tersebut DM Sunthree mengalami peningkatan layanan top up tidak hanya pada game saja, akan tetapi menerima penjualan top up voucher untuk digunakan sebagai bahan top up.

##### **2.1.3 Logo**

Logo merupakan sebuah gambar atau sketsa dengan arti tertentu dan mewakili suatu arti dari perusahaan yang dapat mudah dikenali oleh para pelanggan. Adapun logo dari DM Sunthree dapat dilihat pada gambar 2.1.



Gambar 2. 1 Logo DM Sunthree

## **2.2 Landasan Teori**

### **2.2.1 Website**

Website merupakan kumpulan halaman di dalam suatu domain yang berisi beragam informasi yang dapat diakses dan dilihat oleh pengguna internet melalui mesin pencari. Informasi yang umumnya terdapat di dalam halaman web mencakup gambar, ilustrasi, video, dan teks untuk berbagai tujuan yang di dalamnya menggunakan protokol HTTP (Hypertext Transfer Protokol) untuk mengakses dengan menggunakan perangkat lunak yang disebut browser[11]. Pada umumnya, seseorang dapat mengakses tampilan awal suatu situs web melalui halaman utama dengan menggunakan browser dan mengetikkan URL yang tepat. Di dalam halaman beranda tersebut, juga terdapat tautan menuju beberapa situs turunan yang terkait.

Halaman utama atau beranda sering kali menjadi titik awal yang menyediakan navigasi untuk menjelajahi isi lebih lanjut di dalam situs tersebut. Website dapat memiliki berbagai tujuan seperti memberikan informasi, menjual produk atau layanan, sebagai platform komunikasi atau menyediakan hiburan. Seiring dengan perkembangan teknologi, website juga dapat dirancang responsif untuk beradaptasi dengan berbagai perangkat seperti komputer, tablet, dan handphone.

### **2.2.2 Top Up (Isi Ulang)**

Pengisian ulang saldo atau yang sering dikenal sebagai “Top Up” merupakan rutinitas umum di kalangan masyarakat untuk meningkatkan kenyamanan dalam bertransaksi. Sebagaimana yang dijelaskan oleh *Islamic Economic Analysis*

*Reviews* Top Up tidak hanya merupakan pengisian ulang saldo saja, akan tetapi sebuah fasilitas yang diberikan oleh lembaga keuangan kepada nasabah sebagai bentuk pinjaman yang bertujuan untuk memperluas kredit peminjam[12]. Top up adalah hal yang umumnya dilakukan untuk melakukan berbagai jenis transaksi, baik itu e-wallet maupun kebutuhan games.

### **2.2.3 Digital Marketing**

Digital Marketing atau pemasaran digital adalah segala bentuk aktivitas produk maupun jasa yang dilakukan secara virtual dengan menggunakan media digital atau internet. Tidak seperti pemasaran tradisional yang kegiatan beriklannya menggunakan benda-benda fisik seperti baliho, brosur, spanduk, dan sebagainya, maka *Digital Marketing* ini menggunakan berbagai media digital sebagai sarana pemasarannya. Perlu diketahui bahwa *Digital Marketing* dalam ranah cara pemasarannya yang bisa diterapkan untuk berbagai jenis bisnis, baik itu berupa bisnis online maupun offline.

Tujuan dari melakukan *Digital Marketing* adalah untuk mendatangkan *Traffic* yang tertarget, *Traffic* yang berkualitas ke dalam bisnis yaitu pembeli berpotensi ke dalam bisnis melalui promosi-promosi yang akan dijalankan nantinya di berbagai media digital. *Traffic* itu sendiri bisa diartikan sebagai lalu lintas kunjungan atau lebih mudahnya adalah pengunjung yang datang ke dalam bisnis atau mengunjungi penawaran yang telah dibuat. Jadi, segala upaya promosi yang akan dibuat nanti, baik itu lewat website, sosial media, e-mail, ataupun berbagai media digital lainnya[13].

### **2.2.4 Hypertext Preprocessor (PHP)**

PHP adalah singkatan dari *Hypertext Preprocessor* yang digunakan sebagai bahasa script server-side dalam pengembangan web dan disisipkan dalam dokumen HTML. Pemanfaatan PHP memungkinkan pengembang situs web menjadi dinamis yang memberikan kemudahan dalam pemeliharaan situs tersebut. Oleh karena itu terbuka bahasa scripting sumber dengan banyak keuntungan, serta PHP merupakan bahasa yang banyak digunakan dan memiliki banyak dokumentasi di Web.

PHP dapat dijalankan pada semua operasi sistem, mulai dari *Unix Variants* (termasuk Linux, FreeBSD, Ubuntu, Debian, dan Solaris) hingga Windows dan macOS. PHP dapat digunakan dengan berbagai server web terkemuka seperti Apache, Nginx, dan OpenBSD. Selain itu *Cloud Environments* seperti Azure dan Amazon juga dapat mendukung PHP. Selanjutnya, PHP menyediakan *library* untuk menjalankan tugas-tugas umumnya[14].

### **2.2.5 Hyper Text Markup Language (HTML)**

*Hyper Text Markup Language* (HTML) adalah bahasa pemformatan yang digunakan untuk membuat halaman website. Dalam dunia pemrograman berbasis website, HTML menjadi dasar utama dalam pembuatan halaman website. Sebuah file HTML disimpan dengan ekstensi .html dan dapat diakses menggunakan web browser. Namun untuk membuat website yang dinamis, penggunaan HTML saja tidaklah cukup dan membutuhkan bantuan CSS, JavaScript, dan PHP[15].

Istilah *Hypertext* mengacu pada hyperlink yang mungkin terdapat pada halaman HTML dan istilah *Markup Language* mengacu pada cara tag yang digunakan untuk susunan halaman dan elemen di dalamnya. Bahasa pemrograman HTML terdiri dari serangkaian code yang digunakan untuk membangun suatu website. Selain itu, HTML juga mampu menampung elemen-elemen seperti gambar, suara, dan bahkan video. Struktur dasar HTML terdiri dari tiga jenis, yaitu <head>, <body>, Tag dan Atributnya.

### **2.2.6 JavaScript**

JavaScript adalah bahasa script yang diinterpretasikan, di mana *Host Environment* menyediakan akses ke semua objek yang diperlukan untuk mengeksekusi kode. Contoh dari penggunaan JavaScript yaitu kemampuannya untuk menambahkan interaktivitas pada situs web dengan keunggulan utamanya terletak pada penerjemah Javascript yang sudah tertanam di dalam browser web, sehingga tidak diperlukan penambahan perangkat lunak. Dengan menggunakan editor teks dan browser, JavaScript menjadi bahasa yang mudah diakses. Selain itu, JavaScript juga memungkinkan penggunaan API yang sudah terintegrasi dalam

browser ataupun dapat dieksekusi di server dengan menggunakan *Environment* seperti Node.js[16].

### **2.2.7 Cascading Style Sheets (CSS)**

Dalam istilah umum, Cascading Style Sheets (CSS) dapat diartikan sebagai kumpulan kode yang digunakan untuk merancang tata letak bahasa markup yang bertujuan untuk mengatur atau mempercantik tampilan pada halaman HTML. Dengan menggunakan CSS, kita dapat mengubah desain teks, warna, gambar, dan latar belakang kode markup HTML apapun. Pentingnya CSS selalu terkait erat dengan HTML, karena keduanya saling melengkapi. HTML dirancang untuk membentuk struktur atau konten halaman web, sementara CSS digunakan untuk menampilkan halaman web tersebut.

### **2.2.8 MySQL**

MySQL merupakan sebuah sistem manajemen basis data relasional yang bersifat open source, artinya kode sumbernya dapat diakses dan dimodifikasi oleh pengguna sesuai kebutuhan. MySQL menyimpan data dalam tabel yang terstruktur dan menggunakan bahasa SQL (Structured Query Language) untuk mengelola dan mengakses data. Salah satu fitur utama MySQL adalah kemampuannya untuk mendukung transaksi yaitu operasi-operasi yang harus dijalankan sebagai satuan unit konsisten.

MySQL menjadi basis data *open source* yang banyak diminati (Oracle yang menempati peringkat pertama, akan tetapi tidak open source), permintaan akan proses instalasi yang cepat semakin meningkat. *Source code* adalah resepnya, namun dengan *source code* yang tersedia, membuat perangkat lunak tidak selalu mudah diikuti. Proses kompilasi memerlukan waktu dan biasanya perlu menginstal perpustakaan pengembangan tambahan yang dapat mengekspos lingkungan produksi terhadap risiko[17].

### **2.2.9 Application Programming Interface (API)**

*Application Programming Interface* (API) adalah sekumpulan perintah, fungsi, protokol, dan objek yang dapat digunakan untuk membuat perangkat lunak atau

berinteraksi dengan sistem eksternal. API dapat ditemukan dalam *library* yang kita gunakan dari *language package managers* dan pada dasarnya dalam kode yang kita tulis sendiri, meskipun mungkin tidak dimaksudkan untuk digunakan oleh orang lain. Namun ada satu jenis API khusus yang dirancang untuk diakses melalui jaringan dan digunakan secara remote oleh berbagai pengguna dan itulah jenis API yang sering disebut sebagai “Web APIs”[18].

Web API dapat disederhanakan sebagai kumpulan elemen XML dengan beberapa perintah dasar untuk mengambil informasi. Sebuah web API mirip dengan alamat web (end point) yang diciptakan untuk menangani tugas tertentu sesuai dengan permintaan yang diterima. Terkadang, web API juga memiliki parameter sebagai data yang diperlukan untuk menghasilkan hasil yang diinginkan. Beberapa kasus memerlukan kode otentikasi yang disetujui untuk mengakses API dan melihat data yang diinginkan. Dapat dijelaskan bahwa semua web service menggunakan API, tetapi tidak semua API diterapkan dalam web service. API berperan sebagai penghubung antara dua aplikasi yang berbeda dengan memungkinkan komunikasi antar keduanya.

Penggunaan API bertujuan untuk mempercepat pengembangan perangkat lunak dengan menyediakan fungsi secara terpisah. Ini mengurangi kebutuhan pengembang untuk membuat fitur yang serupa, karena API memberikan akses ke sumber data yang dapat digunakan oleh berbagai sistem atau aplikasi tertentu dan API memungkinkan untuk dapat bertukar informasi data dari satu aplikasi ke aplikasi lainnya. Gambaran umum cara kerja API dapat dilihat pada gambar 2.2 berikut.



Gambar 2. 2 Cara Kerja API

Sumber : <https://course-net.com/blog/rest-api-adalah-pengertian-fungsi-dan-cara-kerja/>

### 2.2.10 Digiflazz

Digiflazz adalah sebuah platform layanan distribusi dan pembayaran yang menyediakan berbagai layanan seperti Top Up pulsa, pembelian token listrik, pembayaran tagihan, pembelian voucher game, dan layanan keuangan lainnya. Digiflazz dapat digunakan para pembisnis untuk melakukan transaksi secara mudah dan cepat melalui platform website mereka. Platform ini yang menghubungkan antara pembeli dengan penjual dalam melakukan transaksi produk digital supaya transaksi yang dilakukan lebih aman dan nyaman. Digiflazz telah bekerjasama dengan perusahaan switching dan aggregator pulsa dan produk digital terbaik di Indonesia. Dengan bergabung dan membeli di Digiflazz, para pembeli tidak perlu merasa kesusahan saat mencari supplier pulsa dan produk digital yang lengkap. Adapun pembagian user dari Digiflazz dapat dilihat pada gambar 2.3 berikut.

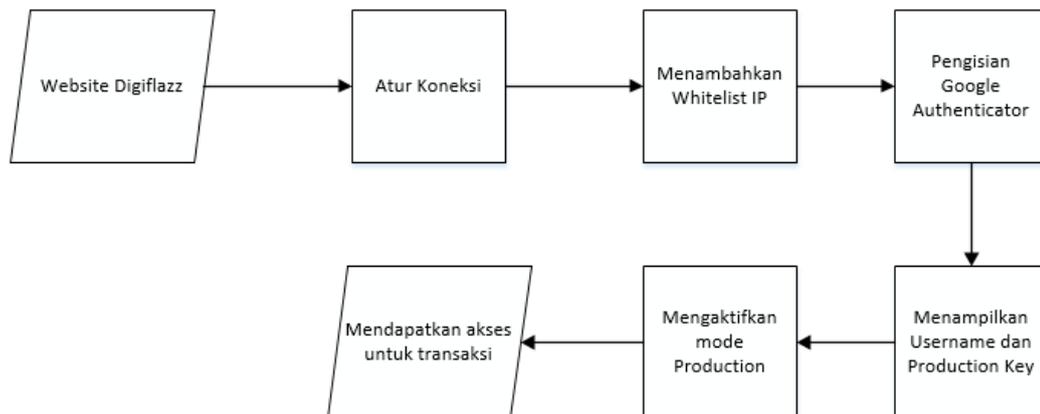


Gambar 2. 3 Sistem Kerja Digiflazz

Sumber : <https://digiflazz.com/>

*Buyer* dari Digiflazz merupakan server pulsa atau web yang sedang mencari supplier atau stok produk H2H (Host To Host) dengan melakukan deposit 1x deposit dan mengkoneksikannya dengan server buyer, Digiflazz dapat membuat transaksi menjadi lebih aman. Sedangkan *Seller* merupakan server pulsa atau web

yang memiliki banyak stok dan ingin menjual stok tersebut ke server H2H lainnya dengan memanfaatkan web Digiflazz untuk membuka peluang usaha yang sangat bagus dan tidak perlu susah-susah mencari *Client*. Gambar 2.4 berikut merupakan tahapan untuk mendapatkan username dan production key dari Digiflazz.



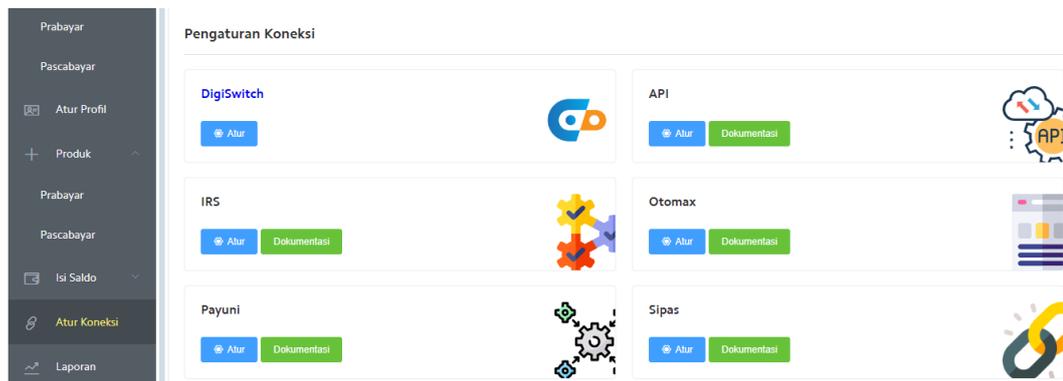
Gambar 2. 4 Tahapan mendapatkan username dan Production Key Digiflazz

Dalam menghubungkan sebuah aplikasi ke API digiflazz, dibutuhkannya username dan production key untuk otorisasi. Username berfungsi sebagai identifikasi unik untuk setiap pengguna atau pelanggan dalam sistem Digiflazz. Sedangkan untuk production key berfungsi sebagai kunci API atau token otorisasinya. Key ini digunakan untuk mengamankan setiap permintaan API yang dikirimkan ke server Digiflazz. Fungsi lainnya yaitu memastikan bahwa pengguna berhak dapat melakukan transaksi melalui API. Berikut merupakan proses bagaimana menggunakan API Digiflazz sehingga dapat melakukan Top Up secara otomatis dengan menggunakan sistem Host To Host (H2H).

#### 1. Mendapatkan Username dan Production Key Digiflazz

Berikut merupakan langkah-langkah yang dilakukan untuk mendapatkan Username dan Production Key dari Digiflazz yaitu :

- a. Masuk ke website Digiflazz atau pada alamat utamanya menggunakan link berikut <https://digiflazz.com>
- b. Pilih menu Atur koneksi dan pilih bagian API



Gambar 2. 5 Atur Koneksi Digiflazz

- c. Di menu Atur Koneksi, sebelum mendapatkan Username dan Production Key, isi Whitelist IP yang telah disediakan di kolom Atur Koneksi API

### Whitelist IP

**Lebih dari 1 IP Statis?**  
Pisahkan setiap IP dengan koma (,) Contoh: 192.1.1.1,192.1.1.2

Development IP

Production IP

Gambar 2. 6 Whitelist IP Digiflazz

- d. Lakukan pengisian Google Authenticator sebagai salah satu syarat untuk bisa menyimpan data.

### Verifikasi OTP

**!** Untuk menyimpan perubahan lakukan, silahkan isi **Verifikasi OTP**, lalu klik **Simpan**

Google Authenticator

Gambar 2. 7 Google Authenticator Digiflazz

- e. Setelah mengisi Whitelist IP dan pengisian Google Authenticator, maka sistem Digiflazz dapat menampilkan Username dan Production Keynya.

**Informasi Pengguna**

Username

Development Key

Production Key

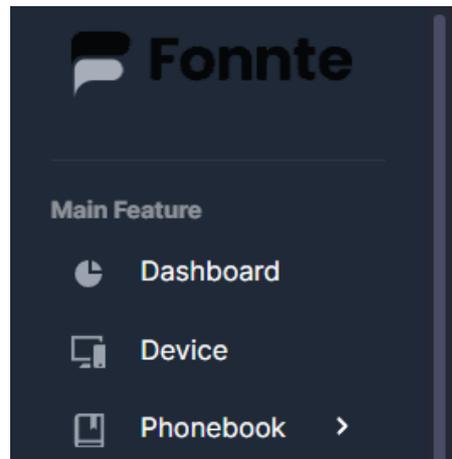
Gambar 2. 8 Production Key Digiflazz

Untuk dapat menampilkan Production Key, User harus klik button “Tampilkan Production Key” dan mengisi ulang verifikasi Google Authenticator.

### 2.2.11 Fonnte

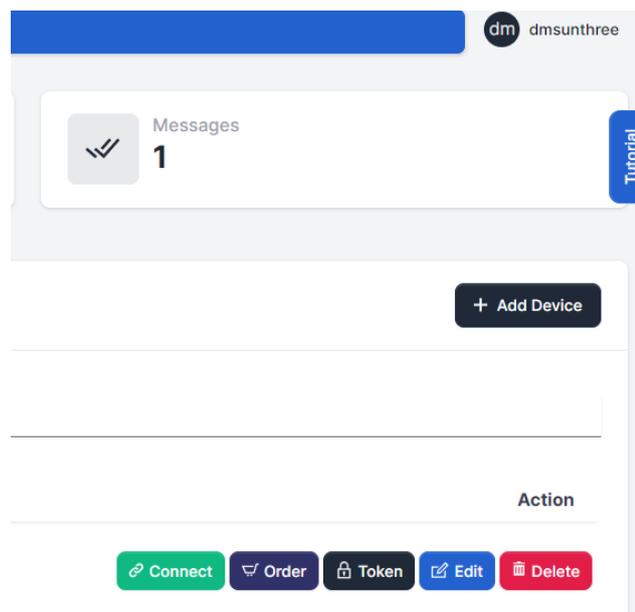
Fonnte adalah salah satu platform komunikasi cloud yang memberikan solusi untuk pengembangan dan pengelolaan aplikasi komunikasi dengan berfokus pada komunikasi melalui pesan atau teks, panggilan suara, dan layanan multimedia. Fonnte memungkinkan perusahaan untuk menyediakan pengalaman komunikasi dan terintegrasi dalam aplikasi mereka. Keunggulan Fonnte terletak pada fleksibilitasnya dalam mendukung berbagai bentuk interaksi pengguna mulai dari notifikasi otomatis dan pemesanan melalui pesan teks. Melalui antarmuka API yang ramah pengembang, Fonnte memudahkan integrasi layanannya ke dalam aplikasi dengan cepat. Adapun tahapan untuk mendapatkan API Fonnte yaitu sebagai berikut :

1. Login ke website Fonnte atau ke halaman berikut <https://fonnte.com>
2. Setelah login, user dapat memilih menu Device yang tersedia di halaman utama Fonnte



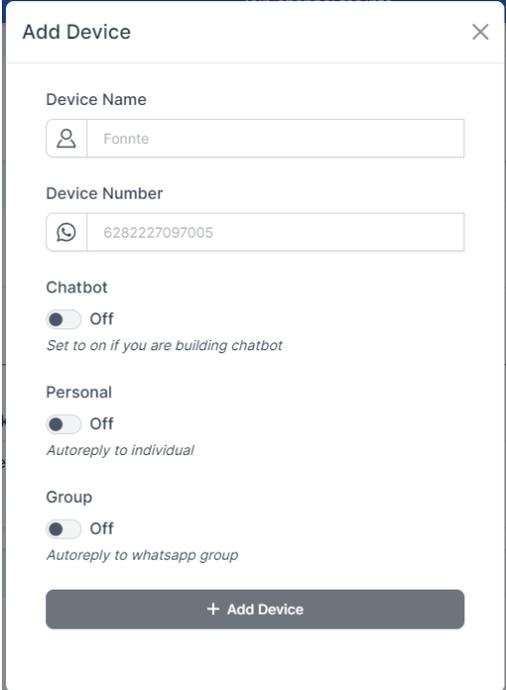
Gambar 2. 9 Menu Device Fonnte

3. Memilih button “+ Add Device” untuk melakukan proses penambahan device



Gambar 2. 10 Add Device Fonnte

4. Setelah user menekan button *Add Device*, maka user memasukan data-data yang diperlukan oleh penyedia layanan seperti nama device, nomor, dan fungsi dari device.



**Add Device** [X]

Device Name  
[User Icon] Fonnte

Device Number  
[WhatsApp Icon] 6282227097005

Chatbot  
 Off  
*Set to on if you are building chatbot*

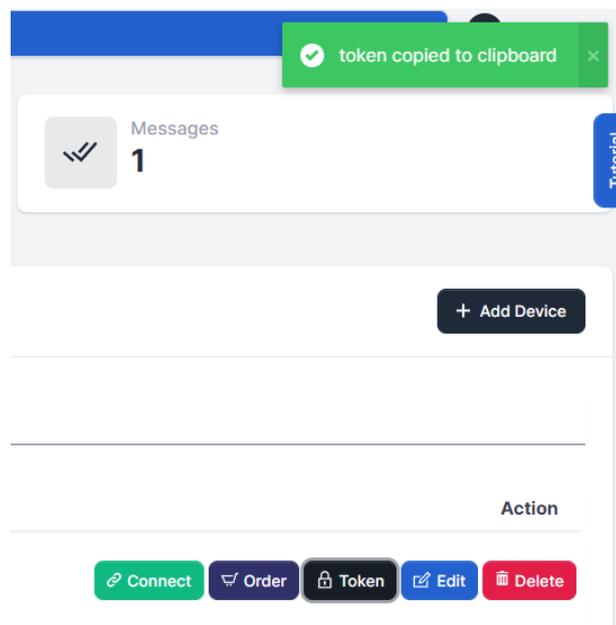
Personal  
 Off  
*Autoreply to individual*

Group  
 Off  
*Autoreply to whatsapp group*

[+ Add Device]

Gambar 2. 11 Pengisian Device di Fonnte

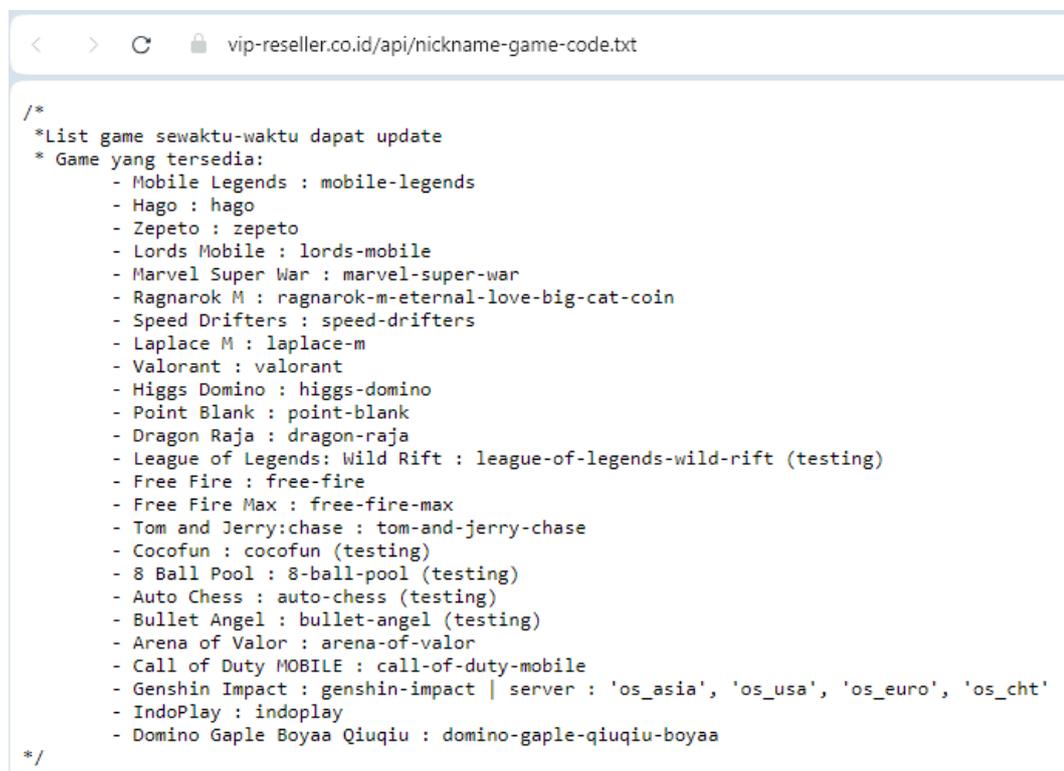
5. Langkah terakhir yaitu menekan tombol “Token” dan token akan menyalin secara otomatis.



Gambar 2. 12 Token Fonnte

### 2.2.12 VIPayment

VIPayment merupakan salah satu platform yang dapat memberikan solusi untuk pengembangan aplikasi yang berkaitan dengan game. Platform ini sangat berguna karena menyediakan layanan khusus yang memungkinkan pengguna untuk melakukan pengecekan terhadap nickname game dari identitas akun secara akurat. Dengan adanya layanan ini, pengguna dapat memastikan bahwa identitas akun yang digunakan dalam permainan game benar-benar sesuai dan valid. Fungsi utamanya untuk mencegah terjadinya kesalahan dalam pengiriman baik itu data, item, atau pesan ke akun lain yang tidak dimaksudkan. Hal ini sangat penting dalam dunia game, di mana kesalahan pengiriman bisa berakibat fatal. Selain itu, VIPayment juga membantu meningkatkan kepercayaan pengguna terhadap aplikasi game yang digunakan. Adapun yang code dan list game yang diberikan oleh VIPayment dalam melakukan pengambilan data yaitu dilihatkan pada gambar 2.13 mengenai *list* game.



```

/*
 *List game sewaktu-waktu dapat update
 * Game yang tersedia:
   - Mobile Legends : mobile-legends
   - Hago : hago
   - Zepeto : zepeto
   - Lords Mobile : lords-mobile
   - Marvel Super War : marvel-super-war
   - Ragnarok M : ragnarok-m-eternal-love-big-cat-coin
   - Speed Drifters : speed-drifters
   - Laplace M : laplace-m
   - Valorant : valorant
   - Higgs Domino : higgs-domino
   - Point Blank : point-blank
   - Dragon Raja : dragon-raja
   - League of Legends: Wild Rift : league-of-legends-wild-rift (testing)
   - Free Fire : free-fire
   - Free Fire Max : free-fire-max
   - Tom and Jerry:chase : tom-and-jerry-chase
   - Cocofun : cocofun (testing)
   - 8 Ball Pool : 8-ball-pool (testing)
   - Auto Chess : auto-chess (testing)
   - Bullet Angel : bullet-angel (testing)
   - Arena of Valor : arena-of-valor
   - Call of Duty MOBILE : call-of-duty-mobile
   - Genshin Impact : genshin-impact | server : 'os_asia', 'os_usa', 'os_euro', 'os_cht'
   - IndoPlay : indoplay
   - Domino Gable Boyaa Qiuqiu : domino-gable-qiuqiu-boyaa
 */

```

Gambar 2. 13 *List* game

Pada gambar 2.13 mengenai *list* game terdapat beberapa kebutuhan pergame yang berbeda. Hal ini karena setiap game memiliki format ID yang berbeda seperti Free Fire format ID gamenya yaitu cukup dengan ID saja, akan tetapi untuk game Mobile Legends dibutuhkan ID game beserta Server atau Zone yang terkait dengan akunnya. Adapun contoh respon yang diberikan oleh VIPayment pada game Mobile Legends yaitu sebagai berikut.

```
{ LF
  .... "result": true, LF
  .... "data": "Itacimo", LF
  .... "message": "Success." LF
}
```

Gambar 2. 14 Respon VIPayment

Potongan respon di atas merupakan hasil *get* data nickname pada game Mobile Legends yang memiliki nickname Itacimo. Dengan memanfaatkan teknologi layanan API VIPayment, mampu meminimalisir kesalahan yang mungkin terjadi akibat salah *input* data.

### 2.2.13 Payment Gateway

*Payment Gateway* adalah sebuah sistem perangkat lunak atau layanan yang menghubungkan situs web atau aplikasi dengan lembaga keuangan yang memproses transaksi pembayaran.. *Payment gateway* membantu dan memfasilitasi proses pembayaran secara elektronik, serta mengautentikasi dan merutekan detail pembayaran dalam lingkungan yang sangat aman dengan berbagai pihak dan bank terkait. *Payment gateway* pada dasarnya berfungsi sebagai saluran “terenkripsi” dengan aman yang meneruskan rincian transaksi dari *Personal Computer* (PC) pembeli ke bank otorisasi. Setelah mendapatkan persetujuan, *Payment gateway* mengirimkan kembali informasi tersebut ke pedagang. Dengan demikian menyelesaikan pesanan dan memberikan verifikasi. Adapun gambaran cara kerja dari *Payment gateway* dapat dilihat pada gambar 2.15 berikut.



Gambar 2. 15 Cara Kerja *Payment Gateway*

Sumber : <https://www.xendit.co/id/blog/cara-kerja-payment-gateway-dan-penerapannya-dalam-bisnis-e-commerce-anda/>

*Payment gateway* sangat bermanfaat karena berbagai manfaat yang ditawarkannya termasuk[19] :

1. Kenyamanan 24x7x365 yang jelas
2. Otorisasi kartu kredit/debit secara real-time
3. Pemrosesan transaksi yang cepat dan efisien
4. Dapat memilih berbagai pilihan pembayaran
5. Mengamankan alur transaksi antara pembeli, penjual, dan lembaga keuangan
6. Pembuatan laporan real-time yang fleksibel dan canggih
7. Penyelesaian multi-mata uang jika diperlukan
8. Pengembalian dana pelanggan
9. Pedagang dapat mengurangi database yang besar, pemrosesan yang ekstensif, dan perangkat lunak yang kompleks
10. CA (Otoritas Sertifikasi) mengautentikasi server aman
11. Pengumpulan data massal dengan cara yang hemat biaya dengan memanfaatkan tambahan berupa pemeriksaan validitas kartu
12. Akses ke “*Hot-List*” kartu untuk menyaring transaksi palsu

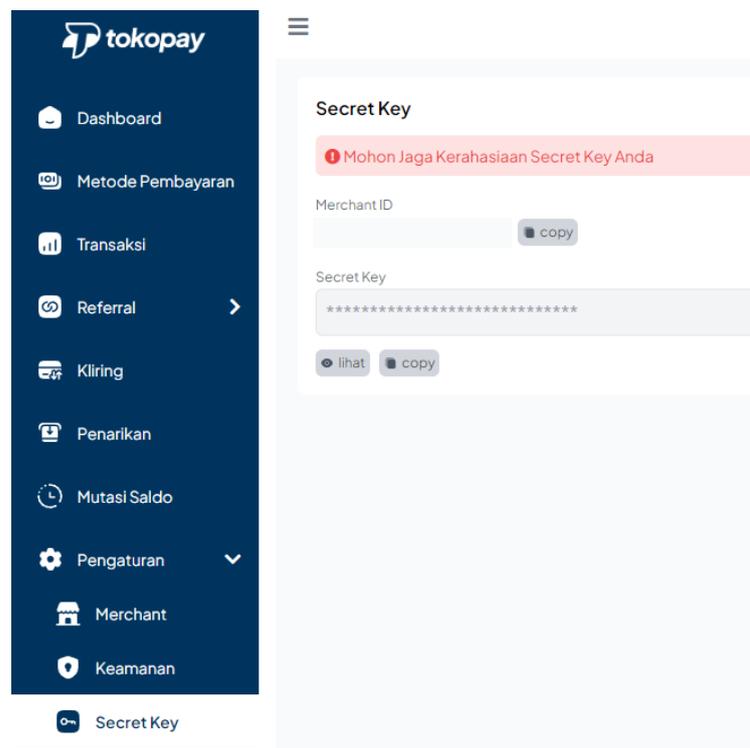
13. Kemampuan untuk memberikan layanan bernilai tambah kepada pedagang, bank pengakuisisi, dan penerbit
14. Penyedia beberapa antarmuka host
15. Kontrol administratif yang komprehensif dan sederhana
16. Langkah-langkah keamanan yang ketat untuk mendapatkan kepercayaan pelanggan dan pedagang.

#### **2.2.14 Tokopay**

Tokopay adalah sebuah platform pembayaran yang menyediakan berbagai layanan transaksi secara digital. Platform ini memungkinkan pengguna untuk melakukan berbagai jenis pembayaran seperti pembelian pulsa, pembayaran tagihan, pembelian token listrik, pembelian voucher game, dan transfer uang antar pengguna. Tokopay dirancang untuk mempermudah transaksi keuangan secara elektronik antara pelanggan dan penjual dengan tujuan utama yaitu menyediakan solusi pembayaran yang efisien dan dapat diandalkan bagi semua pihak yang terlibat dalam transaksi online.

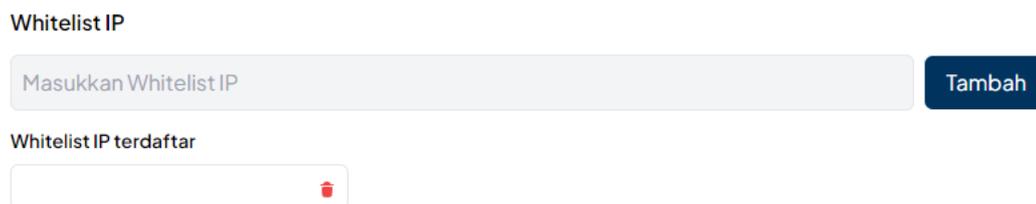
Tokopay ini berfungsi sebagai jembatan antara penjual yang ingin menerima pembayaran elektronik dan pelanggan yang ingin melakukan pembelian secara online. Dengan menggunakan layanan ini, penjual dapat menerima pembayaran dari berbagai metode seperti kartu kredit, transfer bank, dompet digital, dan metode pembayaran online lainnya. Penjual dapat meningkatkan konversi penjualan mereka dengan menyediakan metode pembayaran yang beragam dan aman bagi pelanggan mereka, sedangkan pelanggan dapat merasakan kenyamanan dan keamanan dalam melakukan pembayaran online. Adapun tahapan untuk mendapatkan API Tokopay yaitu sebagai berikut :

1. Mendapatkan Merchant ID dan Secret Key
  1. Login ke website Tokopay atau ke halaman berikut <https://tokopay.id>
  2. Setelah login, user dapat memilih menu Pengaturan dan Secret Key untuk menampilkan Merchant ID dan Secret Key Tokopay di akun yang akan digunakan.



Gambar 2. 16 Merchant ID dan Secret Key Tokopay

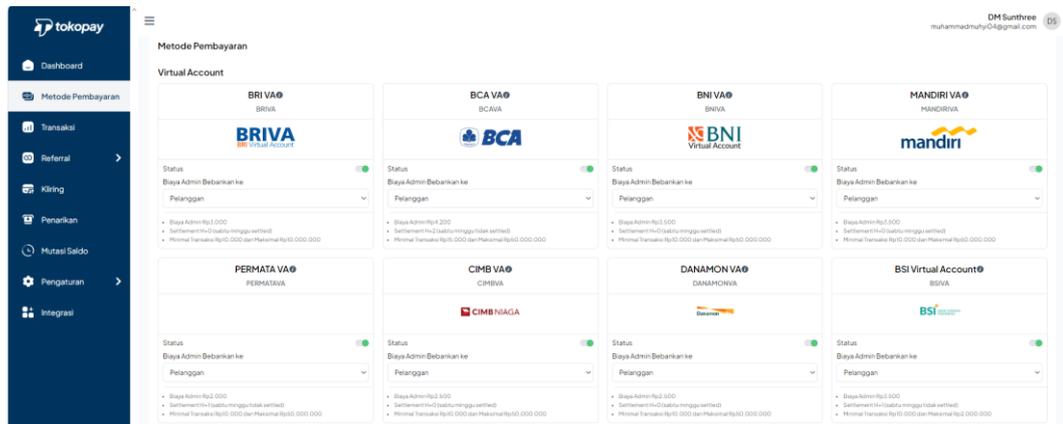
- Setelah mengambil Merchant ID dan Secret Key, maka user dapat menambahkan Whitelist IP untuk melakukan validasi konfigurasi sebagai admin.



Gambar 2. 17 Whitelist IP Tokopay

- Mengaktifkan metode pembayaran di Tokopay

Untuk mengaktifkan metode pembayaran, pilihlah Menu Pembayaran lalu mengaktifkan semua status Metode Pembayaran beserta mengatur siapa yang menanggung Biaya Adminnya. Berikut merupakan penerapan yang dilakukan peneliti dalam menambahkan metode pembayaran otomatis pada gambar 2.18.



Gambar 2. 18 Metode Pembayaran Tokopay

Adapun fungsi dari pengaktifan metode pembayaran yaitu untuk menambahkan metode pembayaran pada web Top Up yang mendukung pembayaran secara otomatis. Selain itu, terdapat biaya admin yang dibebankan kepada 2 pilihan yaitu Pelanggan (konsumen) atau Merchant (Owner web). Jika biaya admin dibebankan kepada pelanggan, maka pada proses Checkout terdapat biaya tambahan dengan ketentuan biaya admin tergantung metode pembayarannya. Jika biaya adminnya dibebankan kepada Merchant, maka sistem Tokopay akan mengurangi saldo yang masuk ke Merchant berdasarkan dengan ketentuan biaya adminnya.

### 2.2.15 Unified Modelling Language (UML)

Basaha pemodelan terdiri dari *pseudo-code*, *actual code*, gambar, diagram, atau deksripsi yang panjang dan unsur-unsur yang membentuk bahasa pemodelan disebut dengan notasi. Bahasa pemodelan dapat berupa apa saja yang mengandung notasi (cara mengungkapkan model) dan deskripsi arti notasi tersebut. Setiap pendekatan pemodelan memiliki kelebihan dan kekurangan yang berbeda-beda. Terdapat 6 keunggulan utama dari *Unified Modelling Language (UML)* yaitu[20] :

1. Setiap elemen bahasa mempunyai ari yang sangat jelas, sehingga dapat meyakinkan bahwa ketika memodelkan aspek tertentu dari sistem dapat dipahami

2. Singkat hanya notasi yang sederhana
3. Menggambarkan semua aspek penting dari suatu sistem
4. Bahasa ini merupakan bahasa yang cukup formal untuk menangani proyek pemodelan sistem yang besar, namun juga dapat diperkecil menjadi proyek kecil sehingga tidak memerlukan upaya yang berlebihan.
5. UML dibangun berdasarkan pembelajaran dari praktik terbaik dalam komunitas berorientasi objek selama 15 tahun terakhir
6. Standar dari UML dikontrol oleh kelompok terbuka dengan kontribusi aktif dari vendor dan akademisi di seluruh dunia.

Versi pertama dari UML dapat memungkinkan orang untuk mengkomunikasikan desain dengan jelas, menyampaikan esensi desain, dan bahkan menangkap dan memetakan persyaratan fungsional solusi perangkat lunak mereka. Namun, dunia berubah secara lebih mendasar dengan adanya pengakuan bahwa pemodelan sistem bukan hanya semata pemodelan perangkat lunak, tapi juga bisa manfaat dari bahasa terpadu seperti UML. Faktor pendorong pengembangan perangkat lunak berorientasi yaitu berbasis model arsitektur, UML yang dapat dieksekusi, dan kebutuhan untuk berbagi model antar alat berbeda yang menempatkan tuntutan pada UML yang awalnya tidak dirancang untuk dipenuhi. Selain itu, UML dengan versi 1.X dan semua revisi sebelumnya dirancang sebagai bahasa terpadu untuk manusia khususnya antara alat *Computer Aided Systems Engineering (CASE)*. Aturan notasi yang mendasari UML 1.X dan model metanya tidak didefinisikan secara formal untuk memungkinkan berbagi mesin ke mesin.

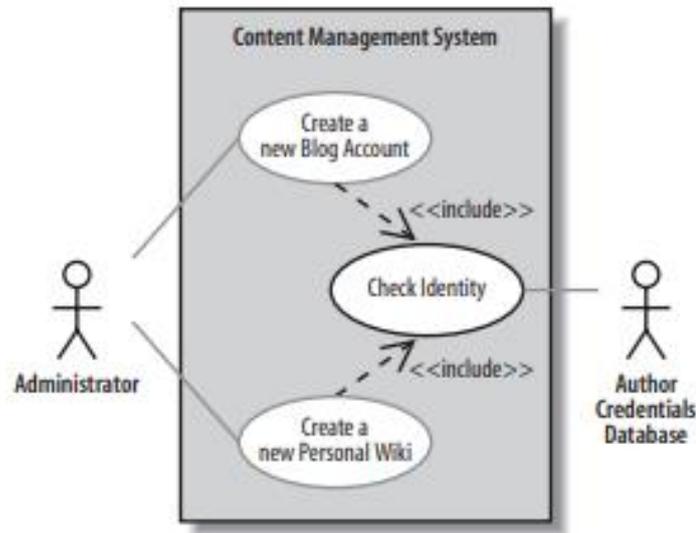
Dengan adanya berbagai kekurangan dari versi sebelumnya, perancangan UML 2.0 sangat berhati-hati untuk memastikan bahwa UML 2.0 tidak akan menjadi terlalu asing bagi orang yang sudah menggunakan UML 1.X.

#### **2.2.15.1 Use Case**

*Use Case* merupakan kasus (situasi) di mana sistem digunakan untuk memenuhi satu atau lebih hal kebutuhan pengguna. Sebuah *use case* menangkap sebagian fungsionalitas sistem yang tersedia. *Use Case* merupakan inti dari UML, oleh karena itu dapat mempengaruhi dan memandu semua elemen lain dalam desain

sistem. *Use Case* adalah titik awal yang baik untuk setiap aspek berorientasi objek pengembangan sistem, desain, pengujian, dan dokumentasi. Dalam hal ini, *Use Case* dapat membantu mengelola proyek beban kerja dan dapat ditugaskan ke tim atau individu untuk diimplementasikan karena mewakili nilai pengguna yang dapat melacak kemajuannya proyek berdasarkan kasus penggunaan. Jika suatu proyek mengalami masalah jadwal, *Use Case* dapat dibuang atau ditunda untuk memberikan nilai tertinggi secepatnya. Dan terakhir, kasus penggunaan juga membantu menyusun pengujian untuk sistem.

Seorang aktor digambarkan dalam notasi UML dengan menggunakan “*stick man*” atau kotak stereotip dan diberi label dengan nama yang sesuai. Aktor tidak harus orang sungguhan, meskipun seorang aktor mungkin adalah seseorang, namun bisa juga menjadi sistem pihak ketiga seperti dalam aplikasi *Business to Business* (B2B)[20]. Selain dari aktor, komponen terdapat komponen lainnya seperti sebuah sistem yang digambarkan dalam bentuk persegi dalam sebuah diagram dengan tujuannya untuk menetapkan batas-batas interaksi dari luar sistem, sistem biasanya diberi label yang sesuai dan komponen *Use Case* yang merupakan komponen untuk menggambarkan fungsi-fungsi dalam suatu sistem, hal ini memungkinkan konsumen dan pembuat sistem untuk saling memahami dan mengetahui alur sistem yang akan dibuat. Adapun contoh dari *Use Case* diagram yaitu seperti pada gambar 2.19 mengenai *Use Case* diagram.

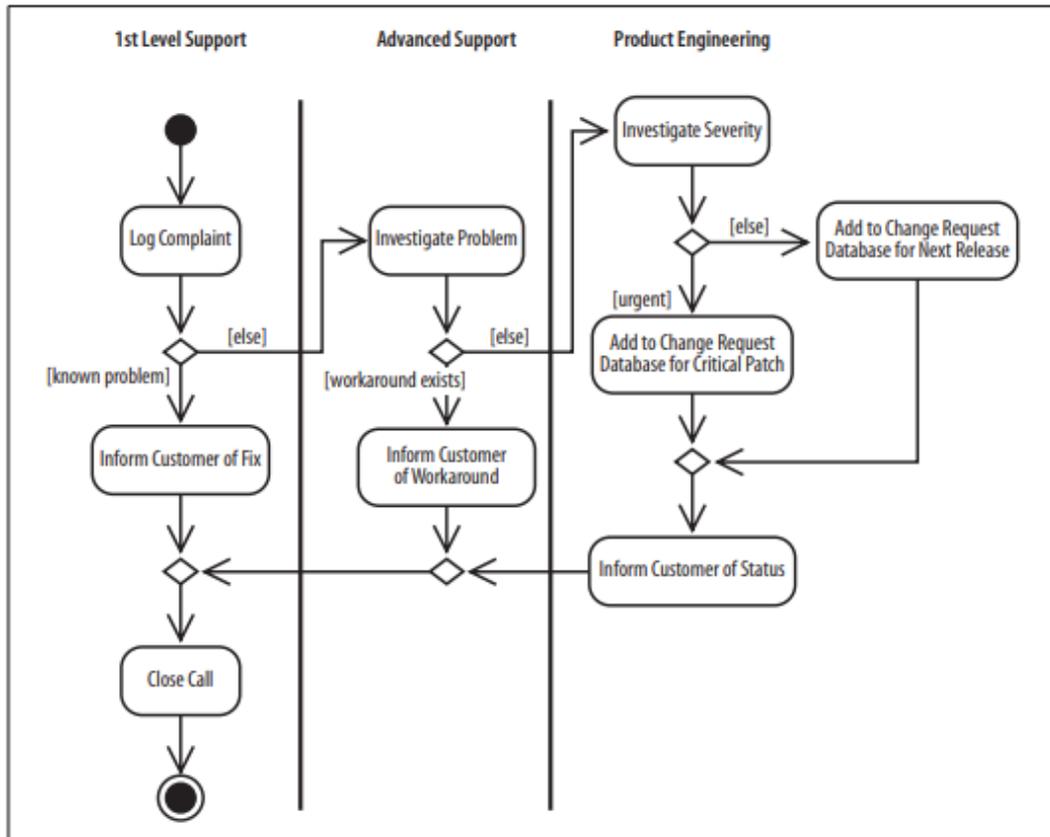


Gambar 2. 19 Contoh Penggunaan Use Case

sumber : Buku UML Russmiles : Learning UML 2.0[20]

### 2.2.15.2 Activity Diagram

*Activity Diagram* menggambarkan sebuah alur aktivitas dalam sistem yang sedang dibangun, termasuk setiap alur dimulai dan keputusan yang mungkin diambil. Diagram ini juga dapat menunjukkan proses paralel yang mungkin terjadi dalam beberapa eksekusi. Sebuah *Activity Diagram* dapat membantu untuk memvisualisasikan *Use Case* dengan lebih baik langkah-langkahnya dibandingkan dengan notasi tabel dalam deskripsi *Use Case*, khususnya pada bagian percabangan yang bergantung pada penulisan. Pada gambar 2.20 merupakan gambaran fungsional dari *Activity Diagram*.



Gambar 2. 20 Fungsional Activity Diagram

Sumber : Buku UML Russmiles : Learning UML 2.0[20]

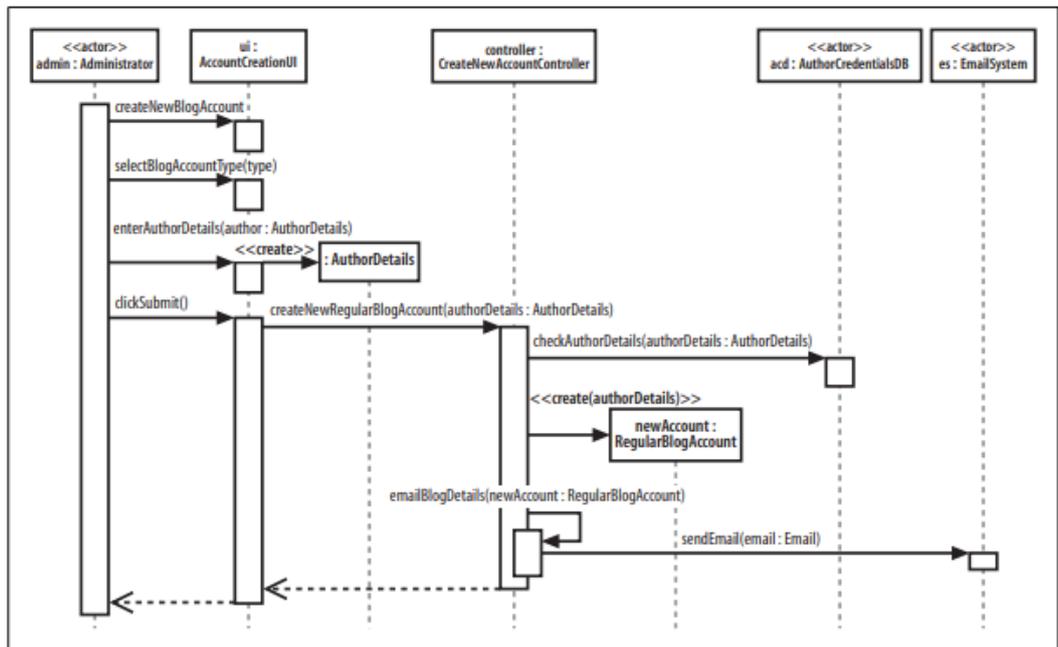
Pada gambar 2.6 mengenai fungsional *Activity Diagram* terdapat aktivitas pertama yang diluncurkan yaitu node awal yang digambarkan sebagai node terisi lingkaran. Node awal hanya menandai dimulainya suatu aktivitas, sedangkan simpul akhir aktivitas digambarkan sebagai dua lingkaran konsentris dengan bagian dalam terisi lingkaran untuk menandai berakhirnya kegiatan. Di antara node awal dan node akhir aktivitas, terdapat tindakan-tindakan yang digambarkan sebagai persegi panjang bulat. Tindakannya merupakan langkah penting yang terjadi secara keseluruhan aktivitas seperti memilih jenis akun, memasukkan detail penulis, dan seterusnya. Alur aktivitas ditunjukkan dengan menggunakan garis panah yang disebut tepi atau jalur. Panah tersebut di tepi aktivitas menunjukkan arah aliran dari satu tindakan ke tindakan lainnya dan garis yang masuk ke dalam suatu simpul

disebut dengan tepi masuk, sedangkan garis yang keluar dari suatu simpul disebut dengan tepi keluar[20].

Secara umum, *Activity Diagram* ini adalah bentuk khusus dari state diagram di mana sebagian besar state adalah tindakan dan transisi utamanya dipicu oleh penyelesaian state sebelumnya. *Activity Diagram* digunakan untuk menggambarkan langkah-langkah atau aktivitas pada suatu sistem. Setiap *Use Case* dalam sistem umumnya memiliki setidaknya satu diagram aktivitas yang menggambarkan urutan aktivitas yang terjadi dalam prosesnya.

### 2.2.15.3 Sequence Diagram

*Sequence Diagram* adalah anggota penting dari bagian yang dikenal sebagai *Interaction Diagrams* atau diagram interaksi dan gambar yang digunakan untuk menjelaskan dan menampilkan interaksi antara objek-objek dalam suatu sistem dengan rincian yang teliti. Diagram ini juga menampilkan pesan atau perintah yang dikirim beserta waktu eksekusinya dan objek-objek yang terlibat dalam proses operasi biasanya diatur dari kiri ke kanan. Diagram interaksi ini memodelkan interaksi runtime yang penting antar bagian yang membentuk sistem dan merupakan bagian dari tampilan logis model. *Sequence Diagram* terdiri dari kumpulan partisipan yang berinteraksi satu sama lain selama pengurutan. Di mana partisipan ditempatkan baik itu secara vertikal maupun horizontal dengan tidak ada 2 partisipan yang saling tumpang tindih. Setiap partisipan memiliki *Lifeline* yang sesuai di bagian bawah halaman. *Lifeline* peserta hanya menyatakan bahwa bagian tersebut ada pada titik dalam urutan dan hanya benar-benar menarik ketika suatu bagian dibuat atau dihapus selama suatu rangkaian. Adapun contoh dari bentuk *Sequence Diagram* terdapat pada gambar 2.21 mengenai pola dasar *Sequence Diagram*.



Gambar 2. 21 Pola Dasar *Sequence Diagram*

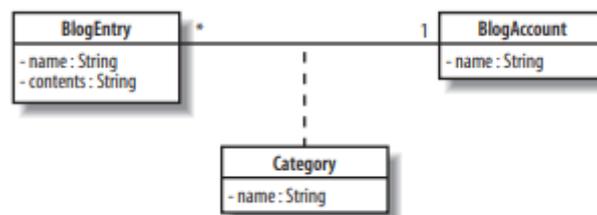
Sumber : Buku UML Russmiles : Learning UML 2.0[20]

Pesan pada *Sequence Diagram* ditentukan menggunakan panah dari partisipan yang ingin menyampaikan pesan, pemanggil pesan, dan kepada partisipan yang dituju menerima pesan, penerima pesan. Pesan dapat mengalir ke arah mana pun sesuai dengan alur dari sistem yang akan dibangun. Ketika sebuah pesan diteruskan ke partisipan lainnya, pesan tersebut memicu atau memanggil partisipan lainnya untuk melakukan sesuatu. Untuk menunjukkan bahwa seorang partisipan aktif yaitu dengan cara melakukan sesuatu yang dapat menggunakan aktivasi. Sebelumnya harus ditentukan jenis interaksi apa yang akan terjadi ketika suatu *Use Case* dijalankan dan memerlukan deskripsi yang lebih rinci tentang apa yang dilakukan *Use Case*.

#### 2.2.15.4 Class Diagram

*Class Diagram* merupakan salah satu jenis diagram dalam *Unified Modelling Language (UML)* yang digunakan untuk memodelkan struktur statis dari sebuah sistem perangkat lunak. Diagram ini menggambarkan kelas-kelas dalam sistem, atribut-atribut, serta hubungan antar kelas. Hubungan antar kelas bekerja sama

menggunakan berbagai jenis hubungannya seperti sebuah kelas yang bisa menjadi tipe dari kelas lain melalui generalisasi atau bisa berisi objek dari kelas lain dalam berbagai cara, tergantung dengan seberapa kuat hubungan antar kedua kelas tersebut. Berikut merupakan gambar 2.22 yang memberikan contoh mengenai *class diagram*.



Gambar 2. 22 Contoh *Class Diagram*

Sumber : Buku UML Russmiles : Learning UML 2.0[20]

Contoh pada gambar 2.22 di atas menggunakan asosiasi yang berguna ketika ingin menunjukkan bahwa suatu kelas terkait dengan dua kelas karena mempunyai hubungan satu sama lainnya. Namun, bergantung pada kategori yang terdapat dalam akun tersebut, entri blog juga dikaitkan dengan sejumlah kategori. Dalam arti lain hubungan asosiasi antar sebuah blog akun dengan entri blog dapat menghasilkan hubungan asosiasi dengan sekumpulan kategori[20].

### 2.2.16 *Blackbox Testing*

Pada tahun 1958, Wolpe menganggap algoritma sebagai sebuah kotak hitam (black box). Kemudian, pada tahun 1962, Sauder menyebutkan bahwa pengujian black box adalah sebuah metode pengujian sintaksis untuk kompiler. Dalam buku "The Art of Testing" yang ditulis oleh Glenford J. Myers pada tahun 2004, disebutkan bahwa konsep black box testing pertama kali muncul pada tahun 1979. Buku tersebut menjelaskan bahwa black box testing tidak fokus pada perilaku internal atau struktur program, melainkan merupakan salah satu strategi penting dalam pengujian perangkat lunak yang bertujuan untuk menemukan kondisi di mana program tidak berfungsi sesuai dengan spesifikasinya. Selanjutnya, pada

tahun 1988, Kaner menyatakan bahwa black box testing digunakan untuk pengujian eksplorasi[21]. Berikut adalah metode pengujian black box :

1. Equivalence Partitioning

Metode ini membagi input data menjadi beberapa kelompok atau partisi yang dianggap setara. Uji kasus dipilih dari setiap partisi, dengan asumsi bahwa jika satu nilai dalam partisi bekerja dengan benar, maka nilai lainnya dalam partisi tersebut juga akan bekerja dengan benar.

2. Boundary Value Analysis

Metode ini fokus pada pengujian nilai batas dari partisi input. Nilai yang berada di batas-batas partisi sering kali lebih rentan terhadap kesalahan, sehingga diuji secara khusus untuk memastikan program berfungsi dengan benar di titik-titik batas tersebut.

3. Syntax Testing

Metode ini membuat kasus uji dari input yang didefinisikan secara formal, seperti BNF, dengan memetakan hubungan hierarki di Abstract Syntax Tree (AST). Tes valid menutupi cabang-cabang AST, sementara tes tidak valid menambahkan kesalahan ke node AST.

4. Random Testing:

Metode ini memilih data uji secara acak dari domain input. Random Testing adalah bagian dari metodologi "Cleanroom" IBM, bertujuan untuk crash-proofing sistem. Pendekatan ini mencakup generasi acak sepenuhnya, pengujian dalam kelas kesetaraan, kombinasi acak, dan pengujian statistik acak.

5. Non-prescriptive Approaches

a. Error Guessing: Penguji menebak situasi rawan kesalahan dan membuat uji kasus untuk mendeteksinya.

b. Exploratory Testing: Penguji merancang uji kasus berdasarkan pengetahuan yang diperoleh selama pengujian berlangsung.

6. Test Catalogues, Categories, and Matrices

Metode ini mendokumentasikan aturan desain uji untuk digunakan kembali, membantu merencanakan dan melacak pengujian.

7. Combinatorial Test Methods

Metode ini memungkinkan otomatisasi pengujian black box menggunakan algoritma kombinatorial untuk mengurangi jumlah kasus uji.

- a. All Combinations: Menghasilkan semua kombinasi nilai data uji.
- b. Each Choice: Memastikan semua nilai data uji dicakup.
- c. Base Choice: Digunakan dalam Category Partition Method.
- d. Orthogonal Array Testing: Menggunakan array dua dimensi untuk mencakup semua kombinasi pasangan nilai data.
- e. Specification-Based Mutation Testing: Memeriksa kemampuan set uji untuk mendeteksi kesalahan tertentu dengan memperkenalkan mutasi.

#### 8. Category Partition Method

Digunakan untuk memformalkan dokumentasi uji dalam bahasa Test Specification Language (TSL).

#### 9. Classification Trees

Menyusun kategori dan pilihan dalam bentuk pohon, dengan uji kasus dicatat sebagai garis horizontal yang bersinggungan dengan pilihan yang relevan.

Contoh dari penggunaan metode *blackbox boundary analysis* pada studi kasus testing Yoopies yaitu sebagai berikut[21] :

Tabel 2. 1 Contoh Penggunaan Metode *Blackbox*

Test Step	Test Data	Expected Result	Actual Result	Status
Memasukkan karakter alfabetic yang berjumlah 5	Firstname : Ridho	Data firstname akan diterima oleh sistem	Data firstname diterima oleh sistem	Berhasil

#### 2.2.17 Pengolahan Data

Pengolahan data adalah proses mengumpulkan, mengorganisir, menganalisis, dan menafsirkan data untuk menghasilkan informasi yang berguna dan mendukung pengambilan keputusan. Pengolahan data salah satunya dengan menggunakan metode kuesioner melibatkan beberapa langkah penting untuk memastikan data

yang dikumpulkan dapat dianalisis dengan efektif. Pertama, desain kuesioner harus sesuai dengan tujuan penelitian, dengan memilih jenis pertanyaan yang tepat seperti pertanyaan terbuka atau tertutup.. Berikut merupakan kategori skor skala Likert menurut sugiyono yang diperoleh jawaban kuesioner dari responden[22] :

Tabel 2. 2 Kategori Skor Skala Likert

No	Skor	Keterangan
1	5	Sangat baik/sangat setuju
2	4	Baik/setuju
3	3	Cukup baik
4	2	Tidak baik/Tidak setuju
5	1	Sangat Tidak baik

Sedangkan perhitungannya menurut arifin untuk mengetahui hasil dari penilaiannya yaitu sebagai berikut[22] :

Tabel 2. 3 Perhitungan Penilaian Data

$$P = \frac{\text{Total nilai responden}}{\text{maximum skor}} \times 100\%$$

Data yang terkumpul dengan menggunakan skala likert, selanjutnya dilakukan penyimpulan penilaian dari setiap aspek dengan kriteria sebagai berikut[23] :

Tabel 2. 4 Penilaian Persentase

Persentase	Kategori
0% - 20%	Sangat Tidak Setuju
21% - 40%	Tidak Setuju
41% - 60%	Ragu-ragu
61% - 80%	Setuju
81% - 100%	Sangat Setuju