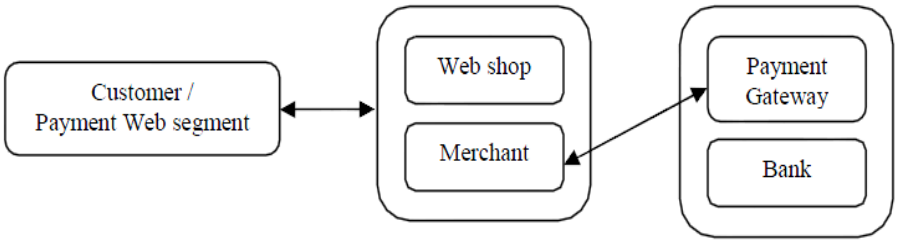
BAB II

LANDASAN TEORI

2.1 *Payment Gateway Service*

Payment Gateway Service merupakan sebuah layanan 3rd *party service* yang menghubungkan antara *merchant* dengan bank. Dengan tersedianya layanan tersebut maka *merchant* dapat menyediakan layanan *online payment* pada *website online shopping*-nya, dengan menghubungkan *website* mereka pada *payment gateway service* menggunakan *service* dari *Application Program Interface (API)*. *Payment gateway service* sangat dibutuhkan, dikarenakan tingginya *initial cost* dan *maintenance cost* untuk melakukan koneksi dengan bank, selain dari pada itu dibutuhkan juga sebuah sistem yang dapat menghubungkan *account* bank dari *customer* dan *merchant* (Gulati et al., 2007) (Đurić et al., 2007) (lihat gambar 2.1).



Gambar 2.1 Relasi pada *internet payment system*

Berikut merupakan keunggulan dari adanya *payment gateway service* :

- Selalu uptime 24 x 7 x 365.
- *Authorisation credit card* secara *real time.*
- Memproses transaksi secara cepat dan effisien.
- Memungkinkan untuk melakukan berbagai jenis pembayaran.
- Informasi data transaksi diproses secara aman.
- Fleksibel dan memungkinkan untuk mengenerelasi report dari history transaksi.
- Pembayaran dengan menggunakan *multi-currency.*
- Mempermudah *merchant* untuk mengatasi masalah pembayaran, sehingga *merchant* dapat berfokus pada website *online*nya saja.
- Menggunakan *certifying authority* dengan *secure server*.
- Filterasi awal sebelum mengirim informasi pembayaran kepada pihak bank (mempermudah sisi bank).
- Dengan sekuritas yang baik akan memberikan positif *user experience* untuk *merchant* maupun *consumer.*

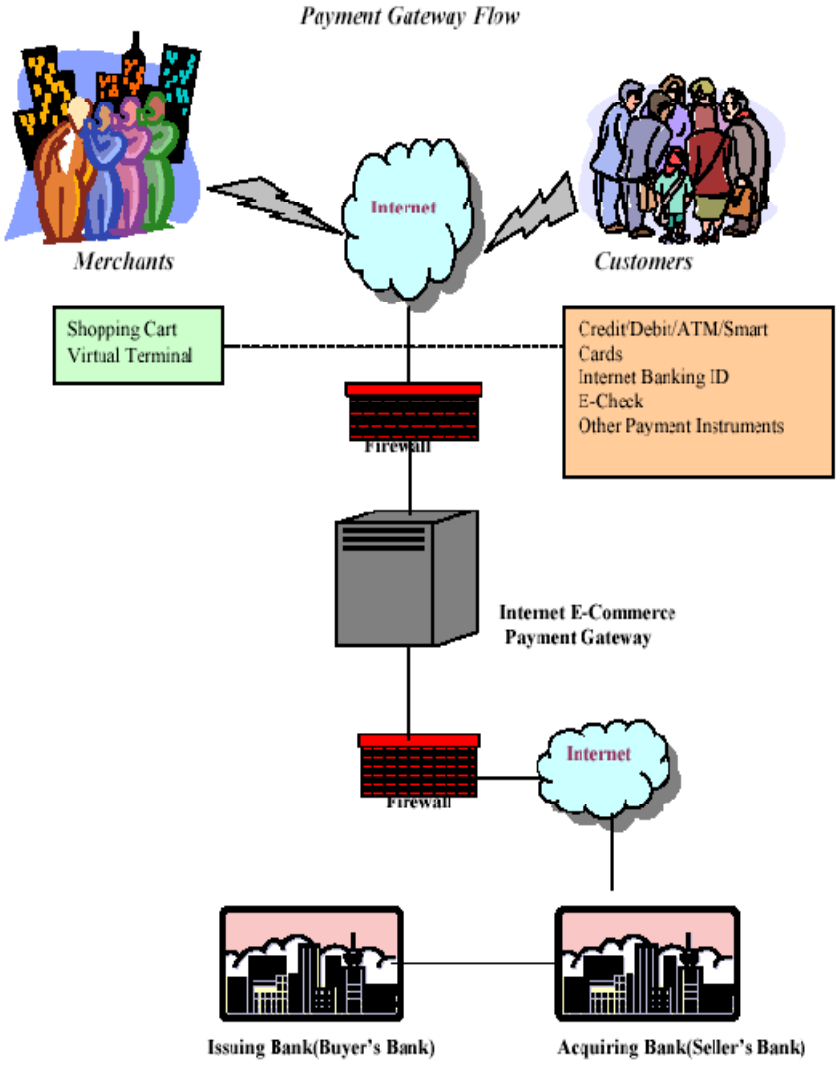
2.1.1 *Payment Gateway Transaction*

Berikut merupakan tahapan – tahapan yang terjadi dari awal hingga akhir ketika melakukan pembelian barang / servis secara *online* (Gulati et al., 2007) (lihat gambar 2.2)

1. *Customer* mengunjungi sebuah *online website shop* untuk membeli barang / servis, setelah *customer* pemilihan barang / servis dan menekan tombol "*buy*". Data dari komputer tempat *customer* itu berada akan dikirim kepada *online website shop* tersebut.
2. *Server online website shop* tersebut akan menerima data yang dikirim pada langkah pertama dan menambahkan *digital certificate* untuk mengenali *website shop* tersebut. Setelah ditambahkan dengan informasi *IP customer* dan informasi transaksi pembelian, *message* tersebut biasa disebut dengan "*Digital Order*". *Digital Order* akan dikirim menuju *payment gateway* melalui *secure network* (dimana data tersebut di enkripsi).
3. Berdasarkan *digital certificate*, *payment gateway* akan mengautentikasi *website shop*.
4. *Payment gateway* akan memberikan jenis – jenis pembayaran *online* yang tersedia untuk dipilih oleh *customer*.
5. *Customer* memilih jenis pembayaran yang ingin dilakukan.
6. *Payment gateway* akan mengirim detail pembayaran ke *acquiring bank* (dimana jenis pembayaran dilakukan dengan menggunakan *credit card*).
7. *Acquiring bank* akan mengirim informasi kepada *issuing bank* dimana *credit card customer* itu terdaftar.
8. Berdasarkan limit dari *credit card* maupun validasi informasi pembayaran yang digunakan, *issuing bank* akan menerima atau

menolak transaksi yang dikirim. Informasi hasil tersebut akan dikirim kepada *Payment Gateway* melalui *acquiring bank*.

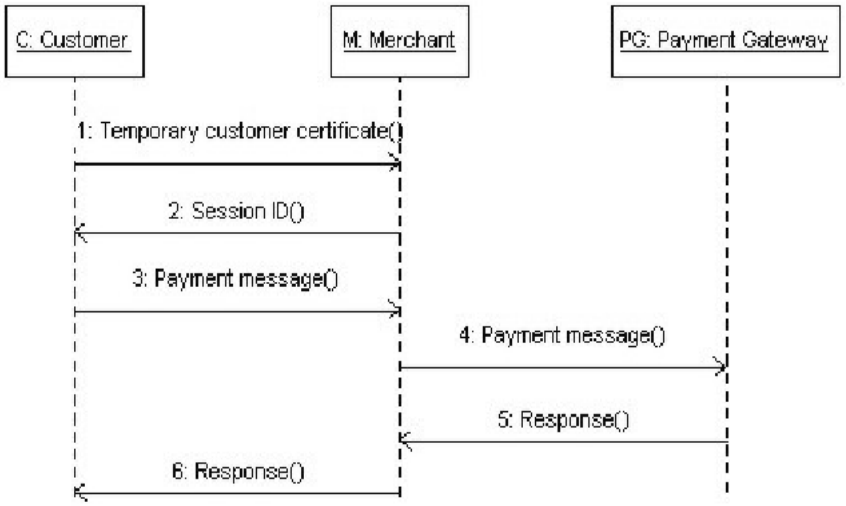
9. *Payment gateway* akan mengirim bukti tanda pembayaran kepada *merchant* maupun *customer*.
10. *Website online* mengirim barang / servis kepada *customer*.



Gambar 2.2 Relasi pada transaksi *online*

Proses pada *payment gateway* terjadi setelah *payment message* dari *website merchant* telah diterima dan sebelum memberikan *response* balik kepada *website merchant* (lihat gambar 2.3). Langkah - langkah yang dilakukan *payment gateway service* selama proses tersebut berlangsung yaitu :

- *Authorising*, verifikasi *credit card*
- *Clearing*, mentransfer hasil transaksi ke *account* bank *merchant*
- *Reporting*, menyimpan data transaksi



Gambar 2.3 *Flow chart* transaksi *online*

2.1.2 *Application Program Interface (API)*

API merupakan sebuah *service* yang bertujuan agar applikasi yang berbeda dapat saling berhubungan. Penggunaan *API* pada *web development* biasanya menggunakan *Hypertext Transfer Protocol (HTTP)*

request messages, yang mempunyai sebagian besar bentuk formatnya adalah *Extensive Markup Language (XML)* atau *JavaScript Object Notation (JSON)*. Berikut merupakan contoh *API* dengan menggunakan *JSON* (lihat gambar 2.4).

```
{
     "firstName": "John",
     "lastName" : "Smith",
     "age"      : 25,
     "address"  :
     {
         "streetAddress": "21 2nd Street",
         "city"         : "New York",
         "state"        : "NY",
         "postalCode"   : "10021"
     },
     "phoneNumber":
     [
         {
           "type"   : "home",
           "number": "212 555-1234"
         },
         {
           "type"   : "fax",
           "number": "646 555-4567"
         }
     ]
 }
```

Gambar 2.4 Contoh arsitektur *JSON*

Penggunaan *API* pada kasus penelitian ini terdapat pada 2 tempat yaitu pada sisi *payment gateway service* dan pada sisi bank. *Payment gateway service* memberikan layanan *API* yang dapat digunakan oleh *merchant* untuk menggunakan *service* yang ditawarkan yaitu cara pengiriman informasi data pembayaran dan *return message* dari proses yang telah dilakukan oleh *payment gateway service*. Untuk *API* pada pihak bank berupa *service* cara pengiriman data informasi pembayaran *credit card* dan *return message* hasil proses *auto debit* dari bank.

2.2 *Fraud Transaction*

Fraud Transaction adalah transaksi yang bertujuan untuk mendapatkan suatu barang / servis dengan menggunakan *credit card* ataupun bank *account* yang bukan merupakan milik dari pelaku transaksi pembelian. Pada kasus yang terjadi pada *payment gateway service, fraud transaction* memiliki kecenderungan terjadi pada pembayaran dengan menggunakan *credit card*. Jenis pencurian data *credit card* dimana *credit card* dinyatakan hilang maupun dicuri adalah jenis yang paling sering muncul untuk *fraud transaction* pada *E-Commerce* (Kou et al., 2004).

Berikut merupakan jenis – jenis teknik yang digunakan untuk mendapatkan informasi *credit card* secara ilegal :

- *Stolen / Lost*

  Terjadi ketika fisik *credit card* dicuri atau hilang.

- *Identity Theft*

  Dikategorikan menjadi 2 bagian yaitu *application fraud* dan *account takeover. Application fraud* terjadi ketika pelaku kriminal menggunakan dokumen palsu maupun dokumen yang dicuri untuk membuka *account* dari orang lain. *Account takeover* terjadi ketika pelaku kriminal berpura – pura sebagai pemilik *credit card* dan melaporkan kehilangan *credit card* kepada *card issuer* untuk dikirimkan kepada si pelaku.

- *Skimming*

  Merupakan teknik pencurian informasi *credit card* ketika *credit card* tersebut digunakan pada transaksi lain, data yang dicuri berupa *credit card number* dan *credit card PIN number.*

- *Carding*

  Merupakan proses untuk me-*verify* validasi dari *credit card*. Setelah *credit card* tersebut dapat diketahui masih *valid* maka pelaku kriminal akan mencoba untuk melakukan *skimming* atau *phishing* untuk mendapatkan informasi lebih lanjut mengenai data si pemilik *credit card*.

- *BIN attack*

  Credit card dibuat berdasarkan *range BIN* tertentu. Ketika *credit card* yang dibuat dengan me-*random generate BIN number*, terdapat kemungkinan didapatnya nomor *credit card* yang valid.

- *Tele Phishing*

  Pelaku kriminal menelpon target korban dengan berpura – pura sebagai agent dari organisasi tertentu untuk mendapatkan informasi *credit card* dari target.

- *Balance Transfer Check*

  Pelaku kriminal berusaha untuk memancing target korban untuk mengaktifkan *transfer balance* yang didalamnya terdapat data informasi *credit card* untuk dikirimkan kepada pelaku.

2.3 *Anti Fraud pada E-Commerce*

Pendeteksian *fraud*, mulai berkembang dengan berbasiskan pada teknologi *Artificial Intelligence*. *System Artificial Intelligence* merupakan sebuah cabang ilmu komputer yang bertujuan untuk membuat program komputer yang mampu melakukan aksi seperti layaknya manusia. "*increasingly, techniques such as neural nets, genetic algorithms and fuzzy logic are being applied in business paradigms for a wide range of forecasting, analysis, optimization and data base tasks. It is not surprising therefore, that these applications are increasingly being seen in the development of combating fraud*" (Giarratano and Riley, 1994). Maka tidaklah mengherankan bila applikasi untuk mencegah *fraud transaction* banyak menggunakan applikasi yang berbasiskan dengan system *AI*.

Hal – hal yang menyebabkan *AI* dianggap cukup efektif untuk menyelesaikan permasalah tersebut (Khin, 2009)

- *AI* sangat fleksibel dan mudah diimplementasikan untuk solusi yang dibangun. Sebagai contoh, teknik *AI* mampu belajar dari pengalaman, sehingga *AI* dapat merespon dengan baik perubahan kondisi pada bisnis pada lingkungan dimana *AI* tersebut di applikasikan.
- Applikasi *AI* tidak memerlukan spesifik design untuk semua kondisi dikarenakan mereka dapat belajar dari training yang mereka lakukan.

- Applikasi *AI* dapat menciptakan kreasi baru, dimana system *AI* yang dibuat mampu berkontribusi untuk mencari *pattern* baru pada data *fraud*.

Akan tetapi pada kondisi *payment gateway service (start up)* di Indonesia, dimana untuk melakukan pengklasifikasian *fraud / non fraud* dengan menggunakan proses pembelajaran tidaklah memungkinkan, hal tersebut disebabkan karena data *training* yang dibutuhkan untuk dapat mengklasifikasikan *fraud / non fraud* belum tersedia. Oleh dikarenakan hal tersebut maka dibutuhkannya sebuah sistem *AI* yang dapat mengindetifikasi *fraud / non fraud* dengan menggunakan metode *expert system*. Berikut merupakan metode – metode yang telah dikembangkan untuk mengklasifikasikan *fraud / non fraud* dengan menggunakan metode *expert system* :

1. Metode Address Verification System (AVS)
2. Metode yang telah dikembangkan oleh Anwer et al. (2009)

yang akan dijelaskan secara ringkas dibawah ini.

2.3.1 Metode *Address Verification System (AVS)*

Address Verification System (AVS) merupakan suatu metode untuk memverifikasi *credit card* berdasarkan data *address* yang terdaftar pada *credit card* (yang digunakan untuk pembayaran online tersebut). Metode ini banyak digunakan pada sistem *anti fraud website online* di negara – negara maju seperti Amerika maupun Ingris. Pada transaksi *online* dimana fisik dari *credit card* tidak dapat dikonfirmasi, sistem *AVS*

membandingkan bagian data *numeric address* yang diinput oleh *customer* dengan data *billing address* yang terdaftar pada *credit card* (Saleh et al., 2002).

Berikut merupakan algoritma yang digunakan pada sistem *AVS* :

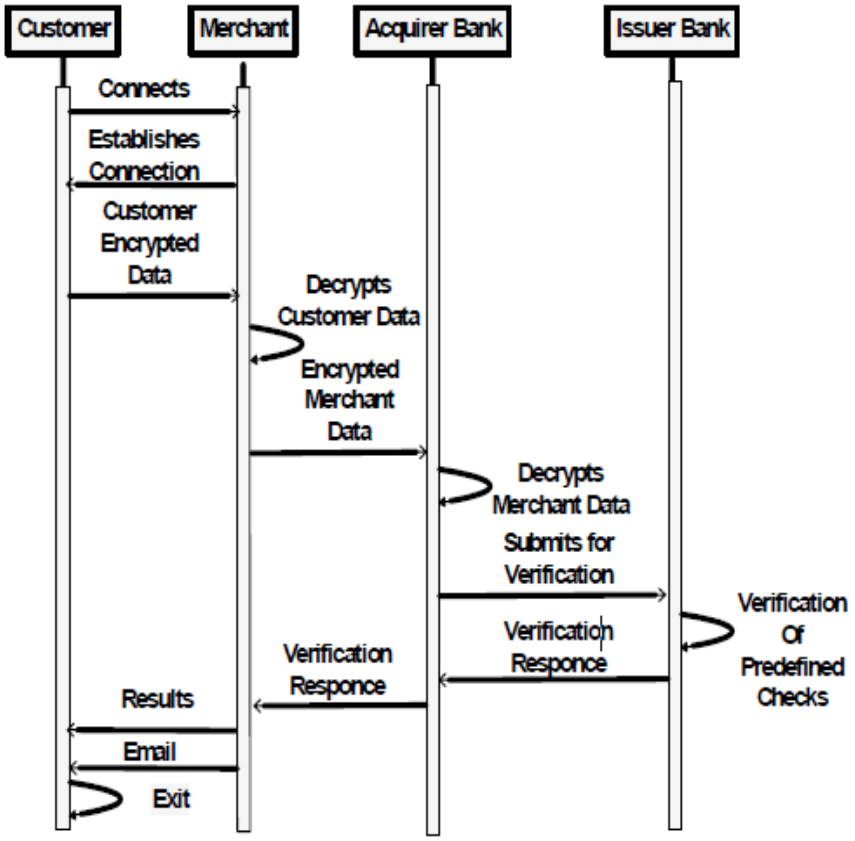
1. Sistem membaca input *credit card numbers*.
2. Sistem membandingkan input *address* dengan data *billing address* pada *credit card* yang digunakan.
3. *IF true THEN* transaksi online diperbolehkan.
4. *ELSE* transaksi tersebut merupakan *fraud.*

Keunggulan dari metode ini adalah memberikan sistem sekuritas terhadap teknik pencurian *account*, terutama terhadap *card generators (carding)* dan *credit card skimmers*. Akan tetapi metode ini juga mempunyai kelemahan sebagai berikut :

- *Input* dari *address* hanya bisa berupa data *billing address* pada *credit card*.
- *AVS* tidak dapat digunakan di Indonesia dikarenakan tidak uniknya variabel tersebut di Indonesia (memungkinkan adanya address yang sama di 2 atau lebih tempat yang berbeda).
- *AVS* tidak dapat berguna untuk mengecek apakah barang yang dibeli mengalami kerusakan atau tidak. Hal ini terjadi pada kasus dimana *address* yang diinput bukan merupakan *address* dari si pembeli.

2.3.2 Metode Anwer et al. (2009)

Anwer et al. (2009) juga telah mengembangkan sebuah metode *anti fraud detection* dengan menggunakan metode *expert system*. Metode yang dilakukan pada penelitian ini berada pada fungsi *verification of predefined checks* yang berlangsung ketika *issuer bank* melakukan proses pengklasifikasian *fraud / non fraud* (lihat gambar 2.5).



Gambar 2.5 *Sequence diagram* pada metode Anwer et al. (2009)

Berikut merupakan langkah – langkah algoritma yang dilakukan pada penelitian tersebut untuk mengklasifikasikan *fraud / non fraud* :

1. *Check Lost / Stolen*

   Melakukan pengecekan variabel *credit card number* yang digunakan pada pembayaran *online* tersebut apakah terdaftar dalam database *credit card* yang dilaporkan hilang maupun dicuri. Jika tidak maka akan melanjutkan ke tahap 2.

2. *Credit Card Validation*

   Melakukan pengecekan variabel *credit card number* yang digunakan pada pembayaran *online* merupakan *credit card* yang valid (*credit card* tersebut memang telah terdaftar pada database). Jika benar maka akan melanjutkan ke tahap 3.

3. *Security Code Check*

   Melakukan pengecekan variabel *credit card PIN number*. Jika benar akan melanjutkan ke tahap 4.

4. *Expiry*

   Melakukan pengecekan variabel *credit card number* yang digunakan pada pembayaran *online* masih belum melewati masa *expire date* nya. Jika benar maka akan melanjutkan ke tahap 5.

5. *Multiple IP*

   Melakukan pengecekan terhadap variabel *credit card number, IP, date and time* yang digunakan dengan adanya kemungkinan terjadinya *multiple IP* dalam waktu yang bersamaan. Jika tidak maka akan melanjutkan ke tahap 6.

6. *Repeated IP*

   Melakukan pengecekan terhadap variabel *credit card number, IP, date and time* dimana hanya diperbolehkan melakukan 1 transaksi *online* dengan *credit card number* yang sama dalam setiap 30 menitnya (transaksi berikut dengan menggunakan *credit card number* yang sama hanya boleh dilakukan 30 menit setelah melakukan transaksi *online* terakhir). Jika tidak maka transaksi tersebut diperbolehkan (*non fraud*).

Keunggulan dari metode Anwer et al. (2009) ini adalah penambahan feature – feature baru untuk melakukan pengklasifikasian *fraud / non fraud transaction*. Dengan adanya penambahan *feature – feature* tersebut maka sekuritas yang ditawarkan tentu menjadi lebih baik dari pada metode *AVS*, akan tetapi metode ini mempunyai kelemahan yaitu sulit diimplementasikannya pada *payment gateway service* di Indonesia. Hal ini dikarenakan pada *feature security code check* yang diusulkan oleh Anwer dan kawan - kawannya menggunakan variabel *credit card PIN number* untuk menverifikasikannya. Pada *payment gateway service* di Indonesia dimana servis yang ditawarkan berupa penghubung antara *merchant* dan bank, *customer* tidak dapat melakukan penginputan *credit card PIN number* pada sisi *payment gateway service*.