

## BAB II LANDASAN TEORI

### 2.1 Studi Literatur

Pada penelitian terkait ini, penulis mengkaji penelitian terdahulu mengenai sistem aplikasi yang dilakukan oleh penelitian-penelitian sebelumnya yang dapat menjadi dasar dari penelitian yang dilakukan. Beberapa penelitian mengenai perangkat lunak automasi pemasaran pada CRM diantaranya adalah sebagai berikut:

Carrisa (2013) melakukan penelitian dengan judul “Penerapan *Customer Relationship Management (CRM)* Sebagai Upaya Untuk Meningkatkan Loyalitas Pelanggan (Studi Kasus Pada Bandung Sport Distro Malang)” dalam penelitian tersebut aplikasi dibangun dengan tujuan untuk memberikan informasi tentang penerapan *Customer Relationship Management (CRM)* pada toko distro Bandung Sport melalui serangkaian proses yaitu mengumpulkan data pelanggan, analisis data pelanggan dan identifikasi pelanggan sasaran, pengembangan program *Customer Relationship Management (CRM)* dan implementasi CRM. Tiga tataran atau tingkatan CRM yang dilakukan adalah CRM Strategis, CRM Operasional dan CRM Analitis. Penerapan *Customer Relationship Management (CRM)* yang dilakukan Bandung Sport sebagai upaya untuk meningkatkan loyalitas pelanggan peningkatan pembelian mencapai 25% dari total penjualan setiap bulan.

Amin (2016) melakukan penelitian dengan judul “Implementasi *CRM (Customer Relationship Management)* Pada Sistem Informasi Reservasi Fotografi Berbasis Web Di Toko Aini Photo Kuningan” dalam penelitian tersebut aplikasi dibangun dengan tujuan untuk memberikan informasi tentang sistem reservasi fotografi dengan menerapkan CRM operasional untuk membantu pelanggan dalam pemesanan jasa fotografi dan transaksi pembelian alat fotografi. Pengembangan sistem menggunakan metodologi pengembangan *waterfall*. Penerapan CRM operasional pada Aini Photo mampu meningkatkan kualitas layanan yang berupa paket foto, cetak foto, pembelian alat fotografi, halaman order dan konfirmasi pembayaran.

Kurniaty (2017) melakukan penelitian dengan judul “Sistem Pemesanan Jasa Fotografi Foosyshoot Studio Dengan Penerapan Model *Customer Relationship Management (CRM)* Berbasis *Website*”. Dalam perancangannya, sistem ini dibangun dengan menggunakan metode *Waterfall* untuk memudahkan perancangan sistem. Berdasarkan hasil pengujian aplikasi maka diketahui bahwa aplikasi ini dibuat dengan memberikan kemudahan kepada pelanggan untuk melakukan pemesanan jasa fotografer secara *online*. Dengan digunakannya model CRM, pelanggan dapat mendapatkan *feedback*

berupa diskon berdasarkan transaksi yang dibuat, serta dapat melihat rekam jejak atau rating para fotografer.

Magdalena (2020) melakukan penelitian dengan judul “Aplikasi *E-CRM (Customer Relationship Management)* Berbasis *Web* Pada Belle Studio Cirebon” dalam penelitian tersebut aplikasi dibangun dengan tujuan untuk memberikan informasi tentang penerapan CRM pada sistem reservasi foto dan pemberian *rating* serta *feedback* yang diberikan oleh pelanggan untuk Belle Studio Cirebon. Penerapan konsep CRM dapat menjaga pelanggan yang sudah ada dan meningkatkan loyalitas pelanggan terhadap Belle Studio dengan memberikan penghargaan serta diskon yang menarik kepada pelanggan. Untuk meningkatkan loyalitas pelanggan, Belle Studio membuat sistem keanggotaan dengan tingkat levelisasi yang terdiri atas lima level yaitu *Member Basic*, *Member Bronze*, *Member Silver*, *Member Gold*, dan *Member Legend*.

Berikut ini adalah penjelasan hal-hal yang membedakan setiap kajian terkait, seperti terlihat pada Tabel 2.1 berikut ini.

**Tabel 2.1** Perbandingan Kajian Terkait

No	Penulis	Judul	Keterangan
1	Anatasha Onna Carissa (2013), Universitas Brawijaya Malang	Penerapan <i>Customer Relationship Management (CRM)</i> Sebagai Upaya Untuk Meningkatkan Loyalitas Pelanggan (Studi Kasus Pada Bandung Sport Distro Malang)	<ol style="list-style-type: none"> <li>1. Menganalisis aplikasi Bandung Sport Distro Malang yang berbasis <i>website</i>.</li> <li>2. Menerapkan CRM untuk meningkatkan loyalitas pelanggan.</li> <li>3. Menganalisis penerapan CRM Strategis, CRM Operasional dan CRM Analitis.</li> </ol>
2	Akhmad Izzudin Amin (2016), Universitas Kuningan	Implementasi <i>CRM (Customer Relationship Management)</i> Pada Sistem Informasi Reservasi	<ol style="list-style-type: none"> <li>1. Aplikasi berbasis <i>website</i>.</li> <li>2. Menerapkan CRM operasional.</li> <li>3. Menampilkan daftar member.</li> <li>4. Menampilkan detail paket foto.</li> <li>5. Menampilkan detail alur proses <i>booking</i> foto.</li> <li>6. Menampilkan cetak foto <i>online</i>.</li> </ol>

No	Penulis	Judul	Keterangan
		Fotografi Berbasis Web Di Toko Aini Photo Kuningan	7. Menampilkan <i>list</i> alat-alat fotografi yang dijual.
3	Nurlina Kurniaty (2017), STIMIK Dipanegara	Sistem Pemesanan Jasa Fotografi Foosyshoot Studio Dengan Penerapan Model <i>Customer Relationship Management (CRM)</i> Berbasis <i>Website</i>	<ol style="list-style-type: none"> <li>1. Aplikasi berbasis <i>website</i>.</li> <li>2. Menampilkan data fotografer.</li> <li>3. Memanfaatkan strategi CRM.</li> <li>4. Menampilkan paket foto.</li> <li>5. Menampilkan <i>booking</i> foto.</li> <li>6. Menampilkan <i>rating</i> pelayanan studio.</li> <li>7. Menampilkan <i>rating</i> fotografer.</li> </ol>
4	Lena Magdalena (2020), Universitas CIC Cirebon	Aplikasi <i>E-CRM (Customer Relationship Management)</i> Berbasis <i>Web</i> Pada Belle Studio Cirebon	<ol style="list-style-type: none"> <li>1. Aplikasi berbasis <i>website</i></li> <li>2. Menampilkan <i>booking</i> foto.</li> <li>3. Menerapkan strategi CRM pada studio foto.</li> <li>4. Menampilan pembayaran foto.</li> <li>5. Menampilkan paket foto.</li> <li>6. Menampilkan <i>member</i> pelanggan studio.</li> </ol>

Berikut adalah penelitian yang dilakukan oleh penulis, seperti terlihat pada tabel 2.2 berikut ini.

**Tabel 2.2** Penelitian Yang Dilakukan Penulis

No	Penulis	Judul	Keterangan
1	Andreas Nababan (2022), Universitas Tanjungpura Pontianak	Perangkat Lunak Automasi Pemasaran Pada Bisnis Jasa Fotografi (Studi Kasus: Sinema Untan)	<ol style="list-style-type: none"> <li>1. Aplikasi berbasis <i>website</i>.</li> <li>2. Menampilkan paket foto.</li> <li>3. Menampilkan alur <i>booking</i> atau reservasi foto.</li> <li>4. Menampilkan levelisasi <i>member</i> pelanggan.</li> <li>5. Menampilkan testimoni dan pemberian <i>rating</i> oleh pelanggan.</li> <li>6. Memanfaatkan strategi CRM sebagai konsep membangun loyalitas dengan pelanggan dan strategi pemasaran.</li> <li>7. Menerapkan CRM Strategis, CRM Operasional dan CRM Analitis.</li> </ol>

## 2.2 Perangkat Lunak (*Software*)

Perangkat lunak merupakan sebuah perangkat yang tidak berbentuk secara fisik, namun dapat dioperasikan oleh pengguna. Perangkat lunak (*software*) adalah program komputer yang terasosiasi dengan dokumen perangkat lunak seperti dokumentasi kebutuhan, model desain dan cara penggunaan (*user manual*) (Sukanto & Shalahuddin, 2016).

Perangkat lunak sebagai bagian sistem komputer yang sifatnya non riil, merupakan sebuah program sebagai sederetan instruksi yang segera dibuat atau dibangun untuk mengendalikan komputer sehingga komputer dapat melakukan tindakan sesuai yang dikehendaki pembuatnya. Komputer hanyalah sekedar mesin yang tidak akan dapat melakukan tugas yang dikehendaki pemakai sekiranya tidak didukung oleh perangkat lunak.

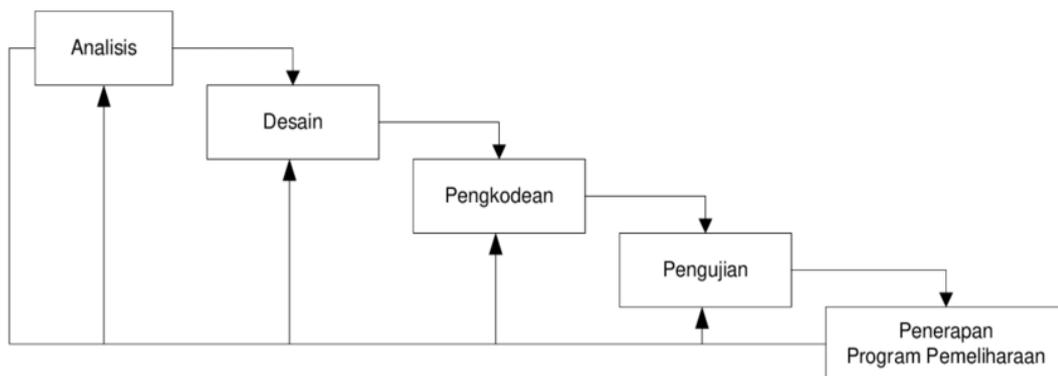
Menurut Kadir (2014), yang menyatakan bahwa perangkat lunak adalah instruksi-instruksi yang ditunjukkan kepada komputer agar dapat melaksanakan tugas sesuai kehendak pengguna. Senada dengan itu, Yurindra (2017) mengemukakan perangkat lunak merupakan transformasi informasi yang memproduksi, mengatur, memperoleh,

memodifikasi, menampilkan atau memancarkan informasi sehingga pekerjaan dapat menjadi lebih sederhana.

Dari beberapa pendapat yang sudah dijelaskan, maka dapat disimpulkan bahwa perangkat lunak adalah instruksi-instruksi atau data yang diformat secara digital yang bisa dibaca dan ditulis oleh komputer sesuai kehendak pengguna.

### 2.3 Model *Waterfall*

Terdapat beberapa metodologi *Systems Development Life Cycle* (SDLC) yang biasa digunakan dalam membangun sebuah sistem, salah satunya adalah model *waterfall*. *Waterfall* merupakan model yang bersifat sistematis dan termasuk dalam model klasik, nama lainnya adalah *Linear Sequential Model* (Pressman, 2015). Tahapan-tahapan model *waterfall* dapat dilihat pada gambar 2.1.



Sumber: Pressman (2015)

**Gambar 2.1** Model *Waterfall*

Penjelasan tahapan-tahapan *waterfall* tersebut yaitu:

1. Analisis

Fase ini merupakan proses analisis terhadap sistem yang sedang berjalan dengan tujuan untuk mendapatkan jawaban mengenai pengguna sistem, cara kerja sistem dan waktu penggunaan sistem, sehingga kebutuhan yang diperlukan untuk sistem baru akan didapatkan.

2. Desain

Desain merupakan proses penentuan cara kerja sistem dalam hal perancangan antarmuka, *database*, dan perancangan alur program. Perancangan diperlukan untuk menggambarkan sistem baru untuk memenuhi kebutuhan *user*.

3. Pengkodean

Tahapan pengkodean yaitu tahap rancangan sistem yang dibentuk menjadi suatu kode program untuk pembuatan sistem.

#### 4. Pengujian

Pengujian dilakukan untuk mengetahui kesesuaian sistem berjalan sesuai prosedur atau tidak dan memastikan sistem terhindar dari *error* yang terjadi. Pengujian juga dilakukan untuk memastikan kevalidan dalam proses *input* sehingga dapat menghasilkan output yang sesuai.

#### 5. Penerapan Program Pemeliharaan

Fase ini yaitu penerapan program dan pemeliharaan sistem yang berguna untuk melihat kemampuannya, mengecek jika masih ada ditemukan *error* atau ada penambahan fitur-fitur yang belum ada pada sistem tersebut. Pemeliharaan diperlukan ketika adanya perubahan dari pengguna seperti ketika ada pergantian sistem operasi, atau perangkat lainnya.

### 2.4 *Customer Relationship Management (CRM)*

*Customer Relationship Management (CRM)* adalah suatu konsep *customer oriented* dan *IT-based management* dengan tujuan membangun relasi jangka panjang serta menguntungkan. Dengan mengatur, menyimpan dan mengolah pengetahuan yang ada mengenai pemakai, memungkinkan untuk membangun dan menjaga kepuasan juga keuntungan bersama. Melalui loyalitas pemakai yang tinggi, perusahaan bisa meraih keuntungan secara kompetitif (Wilde, 2011).

CRM adalah strategi inti dalam bisnis yang mengintegrasikan proses-proses dan fungsi-fungsi internal dengan semua jaringan eksternal untuk menciptakan serta mewujudkan nilai bagi para konsumen sasaran secara profitabel (Buttle, 2007).

CRM merupakan proses mengelola informasi rinci tentang masing-masing pelanggan dan secara cermat mengelola semua “titik sentuhan” pelanggan demi memaksimalkan kesetiaan pelanggan (Kotler & Keller, 2011).

Menurut Utami (2010), CRM adalah suatu proses interaktif yang mengubah data-data pelanggan kedalam kesetiaan pelanggan melalui beberapa kegiatan, yaitu mengumpulkan data pelanggan, menganalisis data pelanggan tersebut dan mengidentifikasi target pelanggan, mengembangkan program CRM, dan menerapkan program CRM.

CRM berhubungan dengan mekanisme membangun sebuah hubungan, termasuk perolehan data, analisis dan penyerdehanaan proses bisnis. Dengan menggunakan strategi ini, maka perusahaan memiliki kemampuan untuk memahami perilaku pembelian pelanggan dengan lebih baik, atau paling tidak memahami berbagai hal lain yang dapat diperoleh dengan konsep ini, dan untuk menentukan tipe komunikasi yang harus dijalin perusahaan dengan pelanggan. Selain itu CRM merupakan strategi komprehensif dari perusahaan agar setiap proses dari daur hidup pelanggan itu dimanfaatkan secara optimal.

Dari beberapa pendapat yang sudah dijelaskan maka dapat disimpulkan bahwa CRM adalah suatu strategi bisnis yang mengintegrasikan proses dan fungsi internal dengan eksternal untuk untuk menciptakan nilai dan memanjakan pelanggan serta menciptakan keadaan saling menguntungkan melalui serangkaian kegiatan mengelola informasi yang rinci tentang masing-masing pelanggan dengan tujuan untuk menciptakan kesetiaan pelanggan dan agar tidak berpaling kepada pesaing. Pada intinya perusahaan bermaksud membangun ikatan yang lebih kuat dengan para pelanggan yang bertujuan untuk memaksimalkan loyalitas pelanggan.



Sumber: Utami (2010)

**Gambar 2.2** *Customer Relationship Management (CRM)*

#### **2.4.1 Tujuan *Customer Relationship Management (CRM)***

*Customer Relationship Management (CRM)* adalah sebuah strategi bisnis yang terdiri dari *software* dan layanan yang didesain untuk meningkatkan keuntungan (*profit*), pendapatan (*revenue*) dan kepuasan pelanggan (*customer satisfaction*) dengan cara membantu berbagai bentuk perusahaan untuk mengidentifikasi pelanggannya dengan tepat, memperoleh lebih banyak pelanggan dengan lebih cepat dan mempertahankan kesetiaan pelanggannya.

Pada dasarnya tujuan CRM menurut Gaffar (2007) yaitu mengenali konsumen yang terbaik dan memberikan kepercayaan terhadap konsumen, memenuhi harapan mereka dan membuat hidup mereka berubah, maka konsumen suatu perusahaan tidak boleh diperlakukan secara sama. Secara umum dapat dikatakan bahwa tujuan setiap strategi CRM adalah untuk mengembangkan hubungan yang menguntungkan dengan pelanggan (Buttle, 2007).

Dari beberapa pendapat yang sudah dijelaskan maka dapat disimpulkan bahwa tujuan dari CRM adalah mengembangkan hubungan yang menguntungkan dengan pelanggan dengan cara mengenali konsumen yang terbaik dan memberikan kepercayaan

terhadap konsumen, memenuhi harapan mereka dan membuat hidup mereka berubah, maka konsumen suatu perusahaan tidak boleh diperlakukan secara sama.

#### **2.4.2 Tataran *Customer Relationship Management* (CRM)**

Menurut Buttle (2007) konsep *Customer Relationship Management* (CRM) dapat dipahami dalam tiga tataran atau tingkatan yaitu:

##### **1. CRM Strategis**

CRM strategis terfokus pada upaya untuk mengembangkan kultur usaha yang berorientasi pada pelanggan atau *customer-centric*. Kultur ini ditujukan untuk merebut hati konsumen dan menjaga loyalitas mereka dengan menciptakan serta memberikan nilai bagi pelanggan yang mengungguli para pesaing. Kultur ini tercermin dari pelaku puncak pimpinan perusahaan, desain sistem formal di dalam perusahaan dan berbagai mitos serta cerita yang beredar di dalam perusahaan. Pandangan *top-down* tentang CRM sebagai strategi bisnis paling penting yang mengutamakan konsumen dan bertujuan memikat dan mempertahankan konsumen yang menguntungkan.

##### **2. CRM Operasional**

CRM operasional lebih terfokus pada automasi cara-cara perusahaan dalam berhubungan dengan pelanggan. Berbagai aplikasi perangkat lunak CRM memungkinkan fungsi-fungsi pemasaran dan penjualan dapat berjalan secara otomatis. Salah satu contoh dari CRM operasional adalah automasi pemasaran. Automasi pemasaran atau disebut juga *Marketing Automation* adalah teknologi yang mengelola proses pemasaran dan kampanye multifungsi di berbagai saluran secara otomatis. Berikut adalah fitur-fitur penting automasi pemasaran, yaitu:

###### **a. Segmentasi Pasar**

Segmentasi pasar adalah proses memilah kelompok pelanggan yang mempunyai kebutuhan yang serupa, membeli produk yang serupa dan mempunyai tanggapan serupa tentang produk yang ditawarkan. Contoh segmentasi pasar pengelompokan berdasarkan umur pelanggan, jumlah pembelian, profil pelanggan dan produk yang laku terjual.

###### **b. Manajemen Kampanye Komunikasi**

Melakukan kampanye pemasaran kepada pelanggan dengan memanfaatkan *platform* perangkat lunak dan teknologi yang dirancang untuk memasarkan secara lebih efektif pada beberapa saluran daring (contohnya seperti email, media sosial dan situs web).

###### **c. *Event-based marketing***

Pemasaran berbasis peristiwa adalah strategi pemasaran yang memanfaatkan peristiwa atau kejadian yang sedang viral saat itu. Informasi yang dikumpulkan akan diolah staf pemasaran dan dikonversikan menjadi kampanye pemasaran yang relevan dengan produk untuk ditawarkan kepada pelanggan.

### 3. CRM Analitis

CRM analitis digunakan untuk mengeksploitasi data konsumen demi meningkatkan nilai mereka dan nilai perusahaan. Sistem ini dikembangkan berdasarkan informasi mengenai konsumen. Data pelanggan dapat diperoleh dari pusat-pusat informasi atau bank data yang dimiliki setiap perusahaan yang relevan, yakni data penjualan (riwayat pembelian barang atau jasa oleh pelanggan), data finansial (riwayat pembayaran atau skor kredit), data pemasaran (respon konsumen terhadap kampanye iklan, data skala loyalitas produk) dan data layanan.

#### 2.4.3 Manfaat *Customer Relationship Management* (CRM)

Penerapan *Customer Relationship Management* (CRM) memberikan berbagai manfaat bagi perusahaan yang menjalankannya. Jika menerapkan strategi ini dengan baik, perusahaan akan mendapatkan keuntungan untuk ke depannya yaitu keuntungan atau profit jangka panjang dan hubungan yang baik dengan pelanggannya. *Customer Relationship Management* (CRM) memberikan manfaat bagi perusahaan, yaitu:

1. Mengidentifikasi berbagai penjualan dari *database* pelanggan lama atau pelanggan potensial.
2. Membantu seluruh aspek penjualan, seperti memberikan akses *online* pada status pesanan dan penelaahan status pelanggan ketika proses penjualan telah lengkap.
3. Segmentasi pasar dengan memilah kelompok pelanggan yang mempunyai kebutuhan yang serupa, membeli produk yang serupa dan mempunyai tanggapan serupa tentang produk yang ditawarkan. Manajemen aktivitas untuk mengatur kegiatan pemasaran.
4. Dapat diintegrasikan dengan *database* yang relevan dan aplikasi *supply-chain management* untuk membantu mengalokasikan berbagai sumber daya.
5. Memantau pola penggunaan pelanggan sehingga dapat mengidentifikasi pola-pola tidak normal atau bila pelanggan mengurangi menggunakan produk yang ditawarkan.

Dengan menerapkan CRM, perusahaan dapat membantu seluruh aspek penjualan seperti mengidentifikasi berbagai penjualan dari basis data (*database*) pelanggan serta dapat menjual lebih banyak produk dan jasa kepada pelanggan, bermanfaat untuk

mengumpulkan informasi tentang pelanggan lebih dalam yang kemudian dimasukkan dalam basis data perusahaan sehingga dapat digunakan sebagai referensi untuk melakukan pendekatan kepada pelanggan, dengan menerapkan CRM, anggaran untuk iklan dan promosi penggunaannya dapat lebih efektif, selain itu tingkat akuisisi pelanggan lebih hemat.

Penerapan CRM memungkinkan perusahaan untuk lebih mengenal dan memahami pelanggannya dengan lebih baik, sehingga perusahaan dapat mengetahui kebutuhan dari pelanggan tersebut. Implementasi CRM jika dikombinasikan dengan teknologi dapat memberikan manfaat yang maksimal untuk perusahaan. salah satunya adalah penggunaan basis data pelanggan. Basis data pelanggan dapat digunakan sebagai *track record* bagi perusahaan yang berisikan pengalaman pembelian pelanggan terhadap perusahaan tersebut.

#### **2.4.4 Cara Kerja *Customer Relationship Management* (CRM)**

Menurut Temporal & Troot (2002), *Customer Relationship Management* (CRM) bekerja dengan:

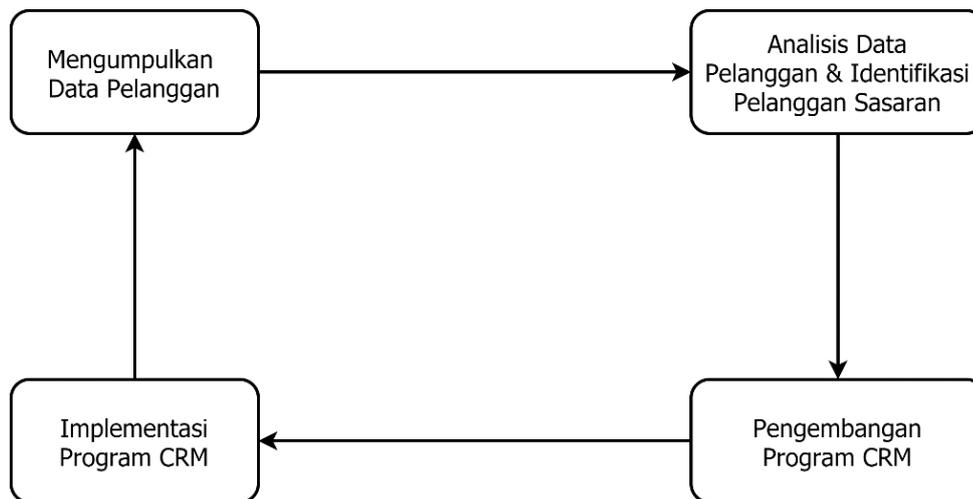
1. Menciptakan lingkaran komunikasi secara terus-menerus antara *merk* dan konsumen. Hal yang paling penting adalah membuka saluran komunikasi dan memudahkan para konsumen untuk berinteraksi.
2. Mengetahui konsumen dengan menggunakan saluran komunikasi yang baru untuk mengetahui konsumen, tidak hanya nama dan alamat mereka tetapi juga siapa mereka, siapa saja yang ada dalam keluarga mereka, apa pekerjaan mereka, apa ambisi-ambisi mereka, apa yang mereka sukai dan tidak sukai.
3. Bertanya kepada konsumen tentang hal yang mereka inginkan dari produk perusahaan.
4. Mengembangkan potensi yang belum terbuka.
5. Menggunakan data konsumen yang ada. Perusahaan perlu melihat informasi yang seharusnya telah dimiliki mengenai konsumen. Termasuk di dalamnya adalah informasi seperti : seberapa sering mereka membeli dari anda, berapa uang yang mereka belanjakan ketika mereka membeli dari anda, kapan terakhir kali mereka membeli dari anda.

Menjadi dekat dengan pelanggan, maka perusahaan berada pada posisi yang paling baik untuk menjual sesuatu kepada mereka pada saat mereka membutuhkannya. Hubungan yang terjalin menjadi lebih kuat, sehingga jika pelanggan ingin memenuhi kebutuhannya, maka mereka akan mencari perusahaan. Dengan menerapkan cara kerja tersebut, perusahaan dapat lebih maksimal untuk menerapkan CRM sehingga lebih tepat

sasaran dalam pemenuhan kebutuhan pelanggan yang pada akhirnya pelanggan tersebut menjadi loyal kepada perusahaan dan enggan untuk pindah ke pesaing.

#### 2.4.5 Proses *Customer Relationship Management* (CRM)

*Customer Relationship Management* (CRM) memungkinkan perusahaan untuk memberikan pelayanan kepada pelanggan secara langsung dan secara maksimal yaitu dengan cara mengembangkan hubungan dengan setiap pelanggan yang berharga melalui penggunaan informasi atau basis data pelanggan yang dimiliki perusahaan. Dalam hal ini perusahaan menggunakan strategi CRM dengan tujuan agar perusahaan tetap fokus terhadap pelanggan berdasarkan informasi atau basis data pelanggan yang dimiliki. Proses dari CRM dapat dilihat pada Gambar 2.3:



Sumber: Utami (2010)

**Gambar 2.3** Proses CRM

Berdasarkan gambar proses CRM Menurut Utami (2010) menjelaskan proses dari CRM adalah sebagai berikut:

##### 1. Mengumpul Data Pelanggan

Tahap pertama dari proses CRM adalah menciptakan *database* pelanggan. *Database* berisikan data yang dikumpulkan perusahaan tentang pelanggan dan menjadi dasar kegiatan CRM. Basis data pelanggan berisikan informasi-informasi mulai dari transaksi, kontak pelanggan, pilihan pelanggan, tanggapan dari kegiatan pasar

##### 2. Menganalisis Data Pelanggan dan Identitas Target Pelanggan

Tahap kedua dari proses CRM adalah menganalisis gudang data pelanggan dan mengubah data tersebut menjadi informasi yang dapat membantu perusahaan dalam mengembangkan program untuk membangun kesetiaan pelanggan.

a. Mengidentifikasi segmen pasar

Analisis data pelanggan difokuskan pada identifikasi segmen pasar kelompok pelanggan yang mempunyai kebutuhan yang serupa, membeli barang dagangan yang serupa, dan mempunyai tanggapan serupa tentang beberapa cara dari kegiatan pasar.

b. Mengidentifikasi pelanggan terbaik

Mengidentifikasi pelanggan terbaik dapat dilakukan dengan cara nilai waktu hidup yang digunakan untuk mengukur nilai setiap pelanggan, piramida pelanggan dan analisis (*Recency, Frequency, Monetary*).

3. Mengembangkan Program CRM

Tahap ketiga dari proses CRM adalah pengembangan program CRM yang akan dibuat harus memiliki tujuan yang pasti yang pada akhirnya akan dievaluasi setiap akhir bulan. Program tersebut juga bertujuan untuk menjalin hubungan dengan pelanggan.

4. Mengimplementasikan Program CRM

Tahap terakhir dari proses CRM adalah mengimplementasikan program CRM yang sudah dibuat. Dalam mengimplementasikan program pelanggan, tujuan-tujuan yang sudah dibuat harus dilaksanakan serta dibutuhkan fokus terhadap program setiap bulannya. Dibutuhkan kerjasama dari semua lini departemen guna mencapai sasaran yang ditetapkan, yaitu kerjasama antar manajemen pusat, staf pusat, dan *front liner*. Ketiga lapisan tersebut merupakan fondasi dari aplikasi CRM agar program pelanggan dapat berjalan secara efektif sehingga tercipta loyalitas.

#### 2.4.6 Ukuran Keberhasilan *Customer Relationship Management* (CRM)

Informasi dan pemahaman mengenai pelanggan merupakan kunci keberhasilan dari *Customer Relationship Management* (CRM). Meskipun demikian, jika informasi dan pemahaman tersebut tidak disebar dan digunakan di seluruh lapisan organisasi maka upaya pemenuhan kebutuhan pelanggan yang setiap saat selalu mengalami perubahan tidak akan tercapai. Menurut Cook (2004), konsep CRM yang berhasil memiliki ciri-ciri sebagai berikut :

1. Ada komitmen yang kuat dari manajemen puncak.
2. Ada definisi yang jelas dan tujuan yang terukur.
3. Ada keterlibatan pelanggan.
4. Lebih berfokus pada bisnis ketimbang pendekatan teknologi.
5. Memiliki tim yang tepat untuk merancang dan mengimplementasikan proyek.
6. Melakukan pendekatan yang semakin meningkat ketimbang perubahan bertahap.

Keenam faktor tersebut merupakan kunci program CRM yang berhasil. CRM

memungkinkan perusahaan melakukan sesuatu yang luar biasa, memberikan manfaat bagi pelanggan dan perusahaan itu sendiri pada saat bersamaan, memungkinkan perusahaan mengerti siapa pelanggan mereka sebenarnya, apa yang mereka beli, dan yang lebih penting lagi apa yang sebenarnya kebutuhan mereka dan memungkinkan perusahaan menambah nilai atas kehidupan mereka sehingga mendorong terciptanya loyalitas pelanggan.

## 2.5 Automasi Pemasaran (*Marketing Automation*)

Sebelum membahas pengertian automasi pemasaran, alangkah baiknya memahami makna dari pemasaran. Pemasaran sebagai suatu fungsi organisasi dan serangkaian proses untuk menciptakan, mengkomunikasikan dan memberikan nilai kepada pelanggan dan untuk mengelola hubungan pelanggan dengan cara yang menguntungkan organisasi dan pemangku kepentingannya (Kotler & Keller, 2011). Pemasaran merupakan penyampaian kebutuhan-kebutuhan hidup individu. Hal ini berkaitan dengan mengidentifikasi apa yang diinginkan oleh pelanggan, kemudian merencanakan dan mengembangkan produk maupun jasa untuk memuaskan kebutuhan serta bagaimana penetapan harga, promosi, saluran distribusi yang tepat. Tujuan sebuah bisnis adalah untuk menciptakan pelanggan yang menekankan pada dua aspek yaitu mengidentifikasi kebutuhan konsumen dan mengelola bisnis untuk memenuhi kebutuhan konsumen. Maka dapat disimpulkan bahwa inti dari pemasaran adalah mengidentifikasi dan memenuhi kebutuhan manusia dan sosial dengan cara yang menguntungkan.

Aktivitas pemasaran melibatkan banyak tugas berulang seperti mengirim email secara massal, mengelola kampanye, *social media marketing*, analisis situs *web*, dan yang lainnya. Apabila tugas-tugas ini dikelola secara manual, maka tidak hanya membosankan dan memakan waktu tetapi juga dapat menciptakan ruang lingkup untuk kesalahan manual. Automasi pemasaran dapat membantu perusahaan untuk mengelola tugas-tugas ini menjadi lebih mudah dan teratur. Ini adalah bagian dari *Customer Relationship Management* atau CRM.

Automasi pemasaran atau disebut juga otomatisasi pemasaran atau *Marketing Automation* adalah teknologi yang mengelola proses pemasaran dan kampanye multifungsi di berbagai saluran secara otomatis (Mulyawan, 2019). Dengan automasi pemasaran, perusahaan dapat menargetkan pelanggan dengan pesan otomatis melalui email, *website*, media sosial, dan teks. Pesan dikirim secara otomatis, sesuai dengan set instruksi yang disebut alur kerja. Alur kerja dapat ditentukan oleh templat, dibuat khusus dari awal, atau dimodifikasi pada pertengahan kampanye untuk mencapai hasil yang lebih baik.

Automasi pemasaran, dalam bentuknya yang paling mendasar adalah seperangkat alat yang dirancang untuk merampingkan dan menyederhanakan beberapa tanggung jawab

yang paling memakan waktu dari peran pemasaran dan penjualan *modern*. Dari mengotomatiskan proses kualifikasi utama hingga menciptakan hubungan untuk pembuatan kampanye digital, automasi adalah tentang menyederhanakan dunia bisnis yang tumbuh terlalu rumit, terlalu cepat. Automasi pemasaran mengacu pada *platform* perangkat lunak dan teknologi yang dirancang untuk memasarkan secara lebih efektif pada beberapa saluran daring (seperti email, media sosial dan situs web) dan mengotomatiskan tugas yang berulang (Mulyawan, 2019).

Automasi pemasaran merupakan *platform* yang digunakan pemasar untuk merencanakan, mengoordinasikan, mengelola, dan mengukur semua kampanye pemasaran mereka, baik secara daring maupun luring. Ini sering digunakan bersama dengan strategi pemasaran siklus hidup untuk mengelola dan memelihara secara dekat prospek yang dihasilkan, yang bertujuan untuk mengubah prospek menjadi pelanggan. Automasi pemasaran memungkinkan perusahaan menerapkan strategi pemasaran digital secara otomatis pada setiap email, pesan, kampanye, atau postingan yang dibuat. Alat Automasi yang baik membantu mengidentifikasi audiens dalam hal ini pelanggan, merancang konten yang tepat, dan secara otomatis memicu tindakan berdasarkan jadwal dan perilaku pelanggan. Setelah kampanye berjalan, perusahaan dapat fokus pada tugas-tugas lain, kemudian menganalisis dan mengubah rencana pemasaran saat hasilnya mulai masuk. Strategi pemasaran otomatis dapat menghemat waktu dan sumber daya, mendorong pendapatan sambil fokus pada pengembangan bisnis.

### **2.5.1 Keunggulan Menggunakan Automasi Pemasaran**

Berikut 4 (empat) keunggulan yang didapat dengan menggunakan Automasi pemasaran, yaitu:

#### **1. Hemat Waktu dan Uang**

Manfaat terbesar dari Automasi pemasaran adalah kemampuannya untuk menghemat waktu. Dalam pemasaran, menghemat waktu akan memberdayakan pemasaran untuk melakukan lebih banyak dengan staf lebih sedikit dan anggaran terbatas. Dengan Automasi pemasaran akan memungkinkan pemasaran untuk memfokuskan waktu dan perhatian pada hal-hal lain yang berdampak besar pada perusahaan atau bisnis.

#### **2. Kemudahan Berbagi Konten Personal**

Salah satu cara paling efektif agar Automasi pemasaran dapat menghemat waktu dan uang adalah dengan menyajikan konten pemasaran yang personal untuk calon pelanggan pada titik-titik perjalanan pembelian mereka. Membuat konten untuk pelanggan berdasarkan tingkat keterlibatan jauh lebih baik digunakan bila dibandingkan dengan memberikan tawaran yang tidak relevan dengan target audiens.

### 3. Melakukan *Close Loop-Report*

Sebagian besar *platform* Automasi pemasaran menawarkan integrasi *Customer Relationship Management (CRM)* yang memungkinkan pengguna untuk membuat sistem *close-loop report*. *Close-loop report* adalah laporan yang berisikan tentang apa saja yang terjadi pada calon pelanggan, dan membantu untuk memahami sumber calon pelanggan terbaik dan terburuk. Dengan *close-loop report*, marketer dapat lebih strategis untuk merencanakan masa depan dengan berfokus pada sumber calon pelanggan terbaik dengan tingkat konversi terbaik.

### 4. Mencapai Keselarasan Antara Pemasaran Dan Penjualan

Automasi pemasaran dapat membantu menjembatani dan menyediakan transparansi dari *close-loop report*. Kolaborasi adalah bagian penting dalam mendorong pertumbuhan jangka panjang.

## 2.5.2 Contoh Perangkat Lunak Automasi Pemasaran

### 1. Marketo

Marketo *automation software* untuk pemasaran direkomendasikan untuk digunakan dalam mengatur bisnis skala kecil hingga menengah tetapi dapat digunakan juga di perusahaan besar. Perusahaan telah memperbarui fitur produk secara teratur untuk meningkatkan efektivitas dari sebuah *software*.

### 2. HubSpot

*Software* ini hadir dilengkapi dengan *advanced marketing tools* untuk berbagai proses dan tugas pemasaran *online*, seperti *web analytics*, *social media marketing*, pembuatan *landing page* yang menarik, *content management*, *search engine optimization*, dan masih banyak lagi.

### 3. Campaigner

Campaigner dapat memilih alamat pelanggan tertentu dan mengelola komunikasi email secara efisien. *Software* ini juga dapat mengintegrasikan pesan email dengan situs media sosial untuk meningkatkan peluang keberhasilan email *campaign*. Campaigner juga dapat melacak aktivitas seperti pembelian khusus untuk membuat data belanja dan membuat email *campaign* yang efektif untuk dikirim ke pengguna.

### 4. Infusionsoft

Infusionsoft adalah perusahaan email pemasaran yang menawarkan *software* otomatisasi untuk usaha kecil dan menengah, difokuskan untuk membantu perusahaan mencapai sasaran pertumbuhan mereka melalui penggunaan teknologi yang powerful dan *platform* pemasaran otomatis.

### 5. Netsuite

Netsuite adalah software Automasi pemasaran tingkat lanjut yang mampu mengotomatisasi berbagai proses, *flows*, dan tugas khusus pemasaran. Pelaksanaan beberapa kampanye pemasaran yang didorong oleh email, media sosial, *event management*, dan lainnya dapat disederhanakan dengan mudah menggunakan Netsuite.

## 2.6 Jasa

Jasa adalah setiap tindakan atau unjuk kerja yang ditawarkan oleh salah satu pihak kepada pihak lain yang secara prinsip *intangible* dan tidak menyebabkan perpindahan kepemilikan apa pun (Kotler & Keller, 2011).

Menurut Lupiyoadi (2013), jasa adalah semua aktivitas ekonomi yang hasilnya tidak merupakan produk dalam bentuk fisik atau konstruksi yang biasanya di konsumsi pada saat yang sama dengan waktu yang dihasilkan dan memberikan nilai tambah (seperti misalnya kenyamanan, hiburan, kesenangan dan kesehatan) atau pemecahan akan masalah yang dihadapi konsumen.

Jasa adalah produk yang tidak dapat dilihat yang kita beli dan gunakan tetapi tidak pernah memiliki. Kualitas jasa sering didefinisikan sebagai usaha pemenuhan dari keinginan pelanggan serta ketepatan penyampaian jasa dalam rangka memenuhi harapan pelanggan. Kualitas jasa merupakan tingkat keunggulan (*excellence*) yang diharapkan dalam pengendalian atas keunggulan tersebut untuk memenuhi keinginan pelanggan. Dengan kata lain, terdapat dua faktor utama yang mempengaruhi kualitas jasa yakni, jasa yang diharapkan (*expected service*) dan jasa yang dipersepsikan (*perceived service*).

Terdapat dua faktor utama yang mempengaruhi kualitas jasa yaitu jasa yang diharapkan (*exspected service*) dan jasa yang dipersepsikan (*perceived service*). Bila jasa yang diterima atau dirasakan (*perceived service*) sesuai dengan yang diharapkan (*exspected service*), maka kualitas jasa dipersepsikan baik dan memuaskan. Jika jasa yang diterima melebihi harapan pelanggan, maka kualitas jasa dipersepsikan sebagai kualitas jasa yang ideal. Akan tetapi bila jasa yang diterima lebih rendah daripada yang diharapkan, maka kualitas jasa sangat bergantung pada kemampuan penyedia jasa menyediakan jasa kepada konsumen secara berkelanjutan dan konsisten.

Dari beberapa pendapat yang sudah dijelaskan maka dapat disimpulkan bahwa jasa pada dasarnya merupakan proses aktivitas yang diberikan kepada konsumen yang tidak berwujud dan memiliki nilai bagi konsumen karena dapat memenuhi kebutuhannya.

### 2.6.1 Karakteristik Jasa

Jasa memiliki karakteristik yang sangat mempengaruhi perencanaan program pemasarannya tersendiri. Jasa merupakan suatu yang khusus karena merupakan sesuatu

yang tidak nyata dan tentu saja berbeda dengan barang (produk fisik). Karena itu ada 5 (lima) karakteristik produk jasa menurut yang membedakannya dengan barang. Keempat karakteristik tersebut meliputi (Tjiptono, 2006):

1. Tidak Berwujud (*Intangible*)

Jasa berbeda dengan barang. Bila barang merupakan suatu objek, alat, atau benda, maka jasa adalah suatu perbuatan, tindakan, pengalaman, proses, kinerja (*performance*), atau usaha. Oleh sebab itu, jasa tidak dapat dilihat, dirasa, dicium, didengar, atau diraba sebelum dibeli dan dikonsumsi.

2. Tidak Dapat Dipisahkan (*Inseparability*)

Barang biasanya diproduksi, kemudian dijual, lalu dikonsumsi. Sedangkan jasa pada umumnya dijual terlebih dahulu, baru kemudian diproduksi dan dikonsumsi pada waktu dan tempat yang sama.

3. Bervariasi (*Variability*)

Jasa bersifat sangat variabel karena merupakan *non-standardized output*, artinya banyak variasi bentuk, kualitas, dan jenis, tergantung kepada siapa, kapan, dan dimana jasa tersebut diproduksi.

4. Daya Tahan (*Perishability*)

*Perishability* berarti jasa tidak tahan lama dan tidak dapat disimpan. Bila permintaan bersifat konstan, kondisi ini tidak menjadi masalah, karena staf dan penyedia jasa bias direncanakan untuk memenuhi permintaan.

5. *Lack of Ownership*

*Lack of ownership* merupakan perbedaan dasar antara jasa dan barang. Pada pembelian barang, konsumen memiliki hak penuh atas penggunaan dan manfaat produk yang dibelinya. Mereka bias mengkonsumsi, menyimpan atau menjualnya. Di lain pihak, pada pembelian jasa, pelanggan mungkin hanya memiliki akses personal atas suatu jasa untuk jangka waktu yang terbatas (misalnya kamar hotel, bioskop, jasa penerbangan, dan pendidikan).

## 2.7 Fotografi

Fotografi adalah suatu seni melukis dengan cahaya, jadi faktor cahaya merupakan unsur terpenting dalam seni fotografi, untuk melakukan suatu pemotretan diperlukannya cahaya. Baik cahaya yang tampak seperti pemotretan biasa atau dengan sinar merah atau sinar x untuk *rontgen*. Alat yang paling populer untuk menangkap cahaya gambar adalah kamera.

Prinsip fotografi adalah memfokuskan cahaya dengan bantuan pembiasan sehingga mampu membakar medium penangkapan cahaya. Secara filosofis, fotografi juga

mempunyai banyak definisi maupun pengertian, entah dipandang secara objektif maupun subjektif (Aditiawan & Bianca, 2011).

Pada dasarnya tujuan dan hakekat fotografi adalah komunikasi. Suatu karya fotografi dapat disebut memiliki nilai komunikasi ketika dalam penampilan subjeknya digunakan sebagai medium penyampaian pesan atau merupakan ide yang terekspresikan kepada pemirsanya sehingga terjalin suatu kontak pemahaman makna. Dalam hal ini karya foto tersebut juga dapat dikatakan sebagai medium yang memiliki nilai guna fungsional dan sekaligus sebagai instrumen karena dijadikan alat dalam proses komunikasi penyampaian pesan atau ide si pencipta karya foto (Soedjono, 2007).

### **2.7.1 Pendekatan Fotografi**

Fotografi memiliki maksud dan tujuan tertentu serta merupakan medium berekspresi yang kuat dan melakukannya tentu perlu pendekatan yang tepat dalam pengaplikasiannya. Pendekatan-pendekatan yang dimaksud disini adalah pendekatan fotografer terhadap objek atau subjek yang ingin ditangkapnya dalam gambar, terutama dalam fotografi jenis potrait yang dominannya memotret ekspresi dan pose manusia.

Menurut Freinenger (2003), ada dua pilihan pendekatan yang pada dasarnya sangat berbeda kepada subjeknya. Pendekatan ini lebih kepada pendekatan subjek yang tergantung oleh sifatnya. Meskipun pendekatan ini berbeda, masing-masing pendekatan ini memiliki hasil yang bernilai artistik yang berbeda-beda juga.

#### **1. Pendekatan Objektif**

Fotografer menjadikan gambar dengan sadar dan sesuai dengan kenyataan yang ada tanpa memerdulikan unsur interpretasi dari sang fotografer itu sendiri

#### **2. Pendekatan Subjektif**

Fotografer menggunakan interpretasi pengalamannya terhadap gambar yang dilihatnya, lalu disajikan dalam karya fotografi.

### **2.7.2 Jenis-Jenis Fotografi**

Di dunia fotografi juga memiliki beragam jenis foto yang biasa digunakan oleh fotografer untuk berbagai kepentingan (Aditiawan & Bianca, 2011), yaitu:

1. *Photo Journalism*, bentuk khusus dari jurnalisme (mengumpulkan, mengedit, dan menyajikan bahan berita untuk diterbitkan atau disiarkan) yang menciptakan gambar agar dapat menceritakan sebuah kisah berita.
2. *Photo Still Life*, merekam gambar benda mati sehari secara artistik dengan menggunakan cahaya pembantu termasuk makro (benda kecil).

3. *Portrait Photograph*, potret fotografi atau potret adalah penangkapan dengan cara fotografi serupa dengan seseorang atau sekelompok kecil orang (potret kelompok), di mana ekspresi wajah dan dominan. Tujuannya adalah untuk menampilkan rupa, kepribadian, dan bahkan *mood* subjek.
4. Foto *Comercial Advertising*, foto diambil untuk keperluan promosi, biasanya di bikin menarik dengan bantuan *editing* dan *computer graphics*.
5. Foto Abstrak, aliran abstrak dalam fotografi sebenarnya bisa disebut sebagai aliran para pemuja komposisi. Dengan demikian, seorang fotografer yang akan membuat foto abstrak akan mengisi kanvasnya dengan sebuah komposisi yang dilihatnya di alam. Dari sebuah realitas tiga dimensi yang ada, bisa tercipta jumlah tak terhingga komposisi foto abstrak ini.
6. *Wedding Photography*, tipe ini merupakan salah satu yang paling populer karena setiap orang pasti ingin memiliki foto yang bagus pada momen penting mereka. Tipe ini membutuhkan fotografer yang berpengalaman karena dibutuhkan keahlian untuk menangkap momen-momen penting. Biasanya dibutuhkan lebih dari ratusan foto, baik berupa foto warna, BW (*black and white*) dan sepia.
7. *Fashion Photography*, adalah genre fotografi yang ditujukan untuk menampilkan pakaian dan barang-barang *fashion* lainnya. Fotografi *fashion* yang paling sering dilakukan untuk iklan atau majalah *fashion* seperti *Vogue*, *Vanity Fair*, atau *Allure*. Seiring waktu, fotografi *fashion* telah mengembangkan estetika sendiri di mana pakaian dan mode diperkuat dengan adanya lokasi eksotis atau aksesoris.
8. *Food Photography*, biasanya digunakan untuk membuat kemasan suatu produk atau iklan. Hanya saja dibutuhkan keterampilan dan peralatan yang berkualitas baik untuk menangkap esensi dari makanan yang dijadikan sebagai objek foto.
9. *Fine Art Photography*, fotografi tipe ini bertujuan untuk menangkap visi dari suatu karya seni. Biasanya tipe ini banyak ditemukan pada pameran dan museum.
10. *Landscape Photography*, tipe ini merupakan kumpulan foto dari berbagai tempat yang biasanya digunakan pada kalender, kartu pos dan memorabilia.
11. *Wildlife Photography*, jenis fotografi ini bertujuan untuk mengambil foto dari beberapa hewan yang menarik ketika mereka sedang melakukan aktifitas seperti makan, terbang atau berkelahi. Biasanya foto diambil dengan menggunakan lensa telephoto yang panjang dari kejauhan.
12. *Street Photography* atau fotografi jalanan adalah aliran fotografi yang menarik. Sedikit berbeda dengan foto jurnalistik yang fokusnya mengabadikan momen puncak atau klimaks.

## 2.8 Business Process Modeling Notation (BPMN)

*Business Process Modeling Notation* atau Model dan Notasi Proses Bisnis (BPMN) adalah pemodelan proses bisnis merupakan suatu konsep yang digambarkan dalam bentuk diagram yang mewakili urutan kegiatan yang berfokus pada proses dan tindakan sehingga melalui proses yang dilaksanakan maka hasil yang dicapai bisa selaras dengan target yang ditetapkan (Priyono, 2018).

BPMN merupakan suatu standar untuk memodelkan proses bisnis yang dapat diterima diseluruh dunia dengan tujuan untuk mendukung manajemen proses bisnis baik untuk pengguna teknis maupun untuk pengguna bisnis. Pemodelan proses bisnis dengan BPMN juga dapat memberikan gambaran sangat jelas mengenai proses bisnis yang terjadi dan hubungan antar tiap unit atau bagian bisa digambarkan dengan lebih spesifik (Wagiu, 2018).

Dalam metode pemodelan proses bisnis menggunakan BPMN terdapat empat element yang dapat digunakan beserta dengan beberapa notasi yang menjadi bagian dari masing-masing elemen. Elemen dan notasi yang digunakan dapat dilihat pada tabel 2.3.

**Tabel 2.3** Elemen BPMN

Elemen	Notasi
<i>Swimlane</i>	<i>Pool</i>
	<i>Lane</i>
<i>Connecting Object</i>	<i>Sequence Flow</i>
	<i>Message Flow</i>
	<i>Association</i>
<i>Artifact</i>	<i>Annotation</i>
	<i>Group</i>
	<i>Data Object</i>
	<i>Data Store</i>
<i>Flow Object</i>	<i>Event</i>
	<i>Activity</i>
	<i>Gateway</i>

## 2.9 Unified Modelling Language (UML)

*Unified Modeling Language* (UML) adalah salah satu standar bahasa yang banyak digunakan di dunia industri untuk mendefinisikan *requirement*, membuat analisis dan desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek. UML merupakan bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks pendukung.

UML muncul karena adanya kebutuhan pemodelan visual untuk menspesifikasikan, menggambarkan, membangun, dan dokumentasi dari sistem perangkat lunak.

Menurut Sukamto & Shalahuddin (2016) UML adalah salah satu standar bahasa visual yang banyak digunakan di dunia industri untuk mengidentifikasi *requirement*, membuat analisis & desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek. UML muncul karena adanya kebutuhan pemodelan visual untuk menspesifikasikan, menggambarkan, membangun, dan dokumentasi dari sistem perangkat lunak. UML hanya berfungsi untuk melakukan pemodelan, jadi penggunaan UML tidak terbatas pada metodologi tertentu, meskipun pada kenyataannya UML paling banyak digunakan pada metodologi berorientasi objek.

UML hanya berfungsi untuk melakukan pemodelan. Jadi penggunaan UML tidak terbatas pada metodologi tertentu, meskipun pada kenyataannya UML paling banyak digunakan pada metodologi berorientasi objek (Sukamto & Shalahuddin, 2016).

UML adalah bahasa pemodelan untuk sistem atau perangkat lunak. Pemodelan (*modeling*) sesungguhnya digunakan untuk penyederhanaan permasalahan-permasalahan yang kompleks sedemikian rupa sehingga lebih mudah dipelajari dan dipahami (Nugroho, 2010).

UML disebut sebagai bahasa pemodelan bukan metode. Kebanyakan metode terdiri paling sedikit prinsip, bahasa pemodelan dan proses. Bahasa pemodelan (sebagian besar grafik) merupakan notasi dari metode yang digunakan untuk mendesain secara cepat.

Di dalam UML ada 13 buah diagram yang dikelompokkan ke dalam tiga kategori yaitu:

1. *Structure Diagrams*, yaitu kumpulan diagram-diagram yang menggambarkan struktur statis dari sistem yang dimodelkan.
2. *Behavior Diagrams*, yaitu kumpulan diagram-diagram yang menggambarkan kelakuan sistem atau rangkaian perubahan yang terjadi di dalam sistem.
3. *Interaction Diagram*, yaitu kumpulan diagram-diagram yang menggambarkan interaksi sistem dengan sistem lain ataupun interaksi antar subsistem dalam sebuah sistem.

Secara garis besar, beberapa diagram utama sudah dapat menggambarkan keseluruhan sistem. Diagram tersebut antara lain *use case diagram*, *class diagram*, dan *activity diagram*.

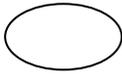
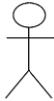
### **2.9.1 Use Case Diagram**

*Use Case Diagram* menurut Widodo (2011) Diagram *use case* bersifat statis, yang memperlihatkan himpunan *Use Case* dan aktor-aktor (suatu jenis khusus dari kelas) dan menggambarkan apa saja aktivitas yang dilakukan oleh suatu sistem dari sudut pandang pengamatan luar. Diagram ini terutama sangat penting untuk mengorganisasi dan

memodelkan perilaku suatu sistem yang dibutuhkan serta diharapkan pengguna yang menjadi persoalan itu apa yang dilakukan bukan bagaimana melakukannya. *Use Case Diagram* menggambarkan fungsionalitas yang diharapkan dari sistem.

Deskripsi simbol-simbol yang digunakan pada *Use Case Diagram* (Nugroho, 2010) dapat dilihat pada tabel 2.4.

**Tabel 2.4** Deskripsi Notasi pada *Use Case Diagram*

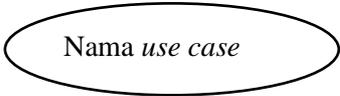
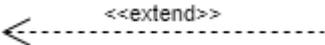
No	Notasi	Nama	Deskripsi
1		<i>Use Case</i>	Menggambarkan fungsionalitas yang dimiliki sistem.
2		<i>Actor</i>	Menggambarkan semua objek di luar sistem (bukan hanya pengguna sistem/perangkat lunak) yang berinteraksi dengan sistem yang dikembangkan.
3		<i>Association</i>	Lintasan komunikasi antara <i>actor</i> dengan <i>use case</i> .
4	<<extend>> 	<i>Extended</i> (Ekstensi)	Penambahan perilaku ke suatu <i>use case</i> dasar.
5	<<include>> 	<i>Include</i> (Menggunakan)	Penambahan perilaku ke suatu <i>use case</i> dasar yang secara <i>explicit</i> mendeskripsikan penambahan tersebut.
6		<i>Generalization</i> (Generalisasi)	Hubungan generalisasi dan spesialisasi (umum - khusus) antara dua buah <i>use case</i> dimana fungsi yang satu merupakan fungsi yang lebih umum dari lainnya.

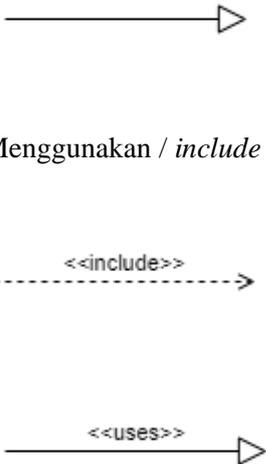
### 2.9.2 Activity Diagram

*Activity Diagram* menggambarkan *workflow* (aliran kerja) proses bisnis dan urutan aktivitas dalam sebuah proses. *Activity diagram* sangat mirip dengan *flowchart* karena memodelkan *workflow* dari satu aktivitas ke aktivitas lainnya atau dari aktivitas ke status. Membuat *activity diagram* pada awal pemodelan proses cukup menguntungkan untuk membantu memahami keseluruhan proses. *Activity diagram* juga bermanfaat untuk menggambarkan *parallel behavior* atau menggambarkan interaksi antara beberapa *use case*.

Berikut merupakan simbol notasi *Activity Diagram* (Sukanto & Shalahuddin, 2016) pada tabel 2.5.

**Tabel 2.5** Deskripsi Notasi pada *Activity Diagram*

No.	Simbol	Deskripsi
1.	<p><i>Use case</i></p> 	Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor, biasanya dinyatakan dengan menggunakan kata kerja diawal frase nama <i>use case</i> .
2.	<p>Aktor/<i>Actor</i></p> 	Aktor adalah orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang, biasanya dinyatakan menggunakan kata benda di awal <i>frase</i> nama aktor.
3.	<p>Assosiasi/<i>association</i></p> 	Komunikasi antara aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan aktor.
4.	<p>Exstensi/<i>extend</i></p> 	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan itu, mirip dengan prinsip

No.	Simbol	Deskripsi
		<i>inheritance</i> pada pemrograman berorientasi objek, biasanya <i>use case</i> tambahan memiliki nama depan yang sama dengan <i>use case</i> yang ditambahkan.
5.	<p data-bbox="400 501 727 533">Generalisasi/<i>generalization</i></p>  <p data-bbox="400 714 756 745">Menggunakan / <i>include</i> / <i>uses</i></p>	<p data-bbox="863 501 1353 936">Hubungan generalisasi dan spesialisasi (umum-khusus) antara dua <i>buah use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan memerlukan <i>use case</i> ini untuk menjalankan fungsinya atau sebagai syarat dijalankan <i>use case</i> ini</p> <p data-bbox="863 954 1353 1032">Ada dua sudut pandang mengenai <i>include</i> di <i>use case</i> :</p> <ul data-bbox="863 1055 1353 1435" style="list-style-type: none"> <li>- <i>Include</i> berarti <i>use case</i> yang ditambahkan akan selalu di panggil saat <i>use case</i> tambahan dijalankan.</li> <li>- <i>Include</i> berarti <i>use case</i> yang tambahan akan selalu melakukan pengecekan apakah <i>use case</i> yang di tambahkan telah dijalankan sebelum <i>use case</i> tambahan di jalankan.</li> </ul> <p data-bbox="863 1458 1353 1637">Kedua interpretasi di atas dapat dianut salah satu atau keduanya tergantung pada pertimbangan dan interpretasi yang dibutuhkan.</p>

### 2.9.3 Class Diagram

Diagram kelas atau *class diagram* menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem (Sukamto & Shalahuddin, 2016). Sedangkan menurut Fowler (2005) yang berpendapat bahwa *Class diagram* mendeskripsikan jenis-jenis objek dalam sistem dan berbagai macam hubungan statis yang terdapat diantara mereka.

*Class diagram* digunakan untuk menggambarkan hubungan kelas-kelas antara satu dengan yang lain serta memiliki atribut dan operasi yang terdapat dalam sistem yang akan dibuat. Atribut merupakan variabel-variabel yang dimiliki oleh suatu kelas. Operasi atau metode adalah fungsi – fungsi yang dimiliki oleh suatu kelas. Atribut dan metode dapat memiliki salah satu sifat sebagai berikut:

1. *Private* (-), hanya dapat digunakan oleh *class* yang memilikinya
2. *Public* (+), dapat digunakan oleh *class* lain.
3. *Protected* (#), hanya dapat dipanggil oleh *class* yang bersangkutan jumlah suatu anak yang mewarisinya.

Nilai kardinalitas atau *multiplicity* sebuah *class* menunjukkan jumlah suatu objek yang dapat berhubungan dengan objek lain. Berikut nilai kardinalitas atau *multiplicity* (Sukanto & Shalahuddin, 2016) pada Tabel 2.5 dan notasi *class diagram* pada tabel 2.6.

**Tabel 2.6** Jenis-jenis *Multiplicity*

No	Indikator	Keterangan
1	0 .. 1	Nol atau satu
2	1	Hanya satu
3	0 .. *	Nol atau lebih
4	1 .. *	Satu atau lebih

**Tabel 2.7** Deskripsi Notasi pada Class Diagram

No	Notasi	Nama	Deskripsi			
1	<table border="1" style="width: 100px; height: 100px;"> <tr> <td style="text-align: center;">Kelas</td> </tr> <tr> <td style="text-align: center;">+atribut</td> </tr> <tr> <td style="text-align: center;">+metode()</td> </tr> </table>	Kelas	+atribut	+metode()	<i>Class</i>	Menggambarkan konsep dasar pemodelan sistem.
Kelas						
+atribut						
+metode()						
2		Asosiasi ( <i>Association</i> )	Sebuah garis solid antara dua <i>class</i> , ditarik dari <i>class</i> sumber ke <i>class</i> target lebih spesifik, digunakan dalam struktur pewarisan.			

No	Notasi	Nama	Deskripsi
3	----->	Ketergantungan ( <i>Dependency</i> )	Relasi antara dua elemen jika perubahan definisi sebuah elemen ( <i>supplier</i> atau sumber) dapat menyebabkan perubahan pada elemen lainnya ( <i>Client</i> atau target).

## 2.10 Teknologi Pendukung

### 2.10.1 Web

Menurut Kadir (2014), *World Wide Web* (WWW) atau biasa disebut dengan web merupakan salah satu sumber daya Internet yang berkembang pesat. Pertama kali aplikasi web dibangun hanya dengan menggunakan bahasa yang disebut HTML (*HyperText Markup Language*) dan protokol yang digunakan dinamakan HTTP (*HyperText Transfer Protocol*). Pada perkembangan berikutnya, sejumlah skrip dan objek dikembangkan untuk memperluas kemampuan HTML yang sekarang ini terdapat banyak skrip seperti: PHP dan ASP, sedangkan contoh yang berupa objek antara lain adalah applet (*java*) (Kadir, 2014). Jadi aplikasi web atau aplikasi berbasis web (*Web-based application*) adalah aplikasi untuk menyampaikan informasi kepada pengguna yang menggunakan layanan Internet berbasis web.

Dalam aplikasi tersebut, terjadi pertukaran antara *klien* (komputer yang meminta informasi) dengan *server* (komputer yang memasok atau menanggapi informasi). *Web* memberikan informasi secara *online* melalui internet langsung. *Klien* melakukan permintaan informasi dengan menggunakan *browser* (contoh *browser*: *Internet Explorer*, *Opera*, *Mozilla*, dan sebagainya). *Server* menerima informasi dan melayani permintaan dari *client*. Hal ini biasa disebut dengan web *server* (contoh *web server*: *Apache*, *IIS*, *Xitami*, dan sebagainya). Setelah itu, web *server* akan berkomunikasi dengan *middleware* (contoh *middleware*: *ASP*, *JSP*, *PHP*, dan sebagainya) untuk bisa berhubungan dengan basis data atau *database* (contoh *database*: *access*, *oracle*, *sql*, dan sebagainya). Setelah berinteraksi dengan *database*, *server* yang telah mendapatkan informasi akan memberikan tanggapan terhadap *klien* yang meminta informasi tadi.

### 2.10.2 *HyperText Markup Language (HTML5)*

Menurut Solihin (2016), HTML merupakan singkatan dari *Hyper Text Markup Language*. HTML dikembangkan pertama kali oleh Tim Berners-Lee bersamaan dengan protokol HTTP (*Hypertext Transfer Protocol*) pada tahun 1989. Tujuan utama pengembangan HTML adalah untuk menghubungkan suatu halaman web dengan halaman web lainnya.

Menurut Kuswayatno (2006), HTML merupakan halaman yang berada pada suatu situs internet atau web. HTML merupakan metode yang menautkan (*link*) satu dokumen ke dokumen lain melalui teks. Menurut Deris Setiawan, HTML merupakan framework internet, hampir semua situs web yang ada menggunakan HTML untuk menampilkan teks, grafik, suara, dan animasinya.

HTML adalah sebuah bahasa markah yang digunakan untuk membuat sebuah halaman web, menampilkan berbagai informasi di dalam sebuah penjelajah web Internet dan pemformatan hiperteks sederhana yang ditulis dalam berkas format ASCII agar dapat menghasilkan tampilan wujud yang terintegrasi. Dengan kata lain, berkas yang dibuat dalam perangkat lunak pengolah kata dan disimpan dalam format ASCII normal sehingga menjadi halaman web dengan perintah-perintah HTML.

Bermula dari sebuah bahasa yang sebelumnya banyak digunakan di dunia penerbitan dan percetakan yang disebut dengan SGML (*Standard Generalized Markup Language*), HTML adalah sebuah standar yang digunakan secara luas untuk menampilkan halaman web. HTML saat ini merupakan standar Internet yang didefinisikan dan dikendalikan penggunaannya oleh *World Wide Web Consortium (W3C)*. HTML dibuat oleh kolaborasi Caillau TIM dengan Berners-lee Robert ketika mereka bekerja di CERN pada tahun 1989 (CERN adalah lembaga penelitian fisika energi tinggi di Jenewa).

### 2.10.3 *Hypertext Preprocessing (PHP) Versi 7.4.19*

Menurut Saputra (2011) yang mengemukakan bahwa PHP atau yang memiliki kepanjangan PHP *Hypertext Preprocessor* merupakan suatu bahasa pemrograman yang difungsikan untuk membangun suatu *website* dinamis. PHP menyatu dengan kode HTML, maksudnya adalah beda kondisi. HTML digunakan sebagai pembangun atau pondasi dari kerangka *layout web*, sedangkan PHP difungsikan sebagai prosesnya sehingga dengan adanya PHP tersebut, *web* akan sangat mudah di *maintenance*. PHP berjalan pada sisi *server* sehingga PHP disebut juga sebagai bahasa *Server Side Scripting*. Artinya bahwa dalam setiap menjalankan PHP, wajib adanya *web server*.

PHP ini bersifat *open source* sehingga dapat dipakai secara cuma-cuma dan mampu lintas *platform*, yaitu dapat berjalan pada sistem operasi *Windows* maupun *Linux*.

PHP juga dibangun sebagai modul pada *web server apache* dan sebagai *binary* yang dapat berjalan sebagai CGI.

#### 2.10.4 JavaScript Versi 1.7

JavaScript menurut Sunyoto (2007) adalah bahasa *scripting* yang populer di internet dan dapat bekerja di sebagian besar *browser* populer seperti *Internet Explorer (IE)*, *Mozilla Firefox*, *Netscape* dan *Opera*. Kode JavaScript dapat disisipkan dalam halaman web menggunakan *tag SCRIPT*.

Menurut Utomo (2007), JavaScript merupakan bahasa *scripting* yang pertama kali dikembangkan oleh Netscape pada tahun 1995. Penulisan JavaScript berada di dalam dokumen HTML dan pemanggilan program tersebut tergantung pada *browser* (navigator) yang digunakan dalam memanggil halaman yang terdapat pada script tersebut. JavaScript juga tidak memerlukan kompilator atau penerjemah khusus untuk menjalankannya.

Selanjutnya menurut Bride (2007), JavaScript adalah bahasa pemrograman berbasis *browser*. Kode-kodenya ditulis langsung ke dalam HTML dari halaman-halaman *web* dan diterjemahkan serta dieksekusi sebagai respon terhadap aktivitas-aktivitas pada halaman *web*.

Dengan adanya JavaScript, maka teknik penulisan HTML dapat dilaksanakan dan dijalankan dengan dua cara, yakni dengan membuat program Javascript untuk menghasilkan dokumen HTML atau dengan membuat dokumen HTML layaknya seperti biasa, sesudah itu jika tersedia program JavaScript, maka menambahkan program JavaScript tersebut sebagai sisipan saja. Beberapa hal tentang JavaScript:

1. Javascript didesain untuk menambah interaktif suatu web.
2. Javascript merupakan sebuah bahasa *scripting*.
3. Bahasa *scripting* merupakan bahasa pemrograman yang ringan.
4. Javascript berisi baris kode yang dijalankan di komputer (*web browser*).
5. Javascript biasanya disisipkan (*embedded*) dalam halaman HTML.
6. Javascript adalah bahasa *interpreter* (yang berarti skrip dieksekusi tanpa proses kompilasi).

#### 2.10.5 Bootstrap Versi 5.0

Bootstrap merupakan sebuah *framework CSS* yang memudahkan pengembang untuk membangun website yang menarik dan responsif. Bootstrap adalah *CSS* tetapi dibentuk dengan *LESS*, sebuah *pre-processor* yang memberi fleksibilitas dari penggunaan *CSS* biasa. Bootstrap dapat dikembangkan dengan tambahan lainnya karena ini cukup fleksibel terhadap pekerjaan *web* yang mengutamakan desain (Otto, 2011).

Bootstrap adalah sebuah *library framework* CSS yang di buat khusus untuk bagian pengembangan *front-end website*. Bootstrap merupakan salah satu *framework* HTML, CSS dan JavaScript yang paling populer di kalangan *web developer*. Pada saat ini hampir semua *web developer* telah menggunakan Bootstrap untuk membuat tampilan *front-end* menjadi lebih mudah dan sangat cepat. karena anda hanya perlu menambahkan *class-class* tertentu untuk misalnya membuat tombol, *grid*, navigasi dan lainnya.

Bootstrap telah menyediakan kumpulan komponen *class interface* dasar yang telah di rancang sedemikian rupa untuk menciptakan tampilan yang menarik, bersih dan ringan. Selain komponen *class interface*, Bootstrap juga memiliki fitur *grid* yang berfungsi untuk mengatur *layout* pada halaman *website* yang bisa digunakan dengan sangat mudah dan cepat. dengan menggunakan Bootstrap kita juga di beri keleluasaan dalam mengembangkan tampilan *website* yang menggunakan Bootstrap yaitu dengan cara mengubah tampilan Bootstrap dengan menambahkan *class* dan CSS sendiri.

#### **2.10.6 Laragon Versi 5.0.0**

Laragon adalah perangkat lunak bebas yang di dalamnya terdapat banyak sistem operasi sebagai *localhost* atau *server* mandiri. Laragon menyediakan banyak layanan, peralatan, dan fitur yang terdiri dari Apache, PHP Server, PHPMyAdmin, MySQL, Memcached, Redis, Composer, Xdebug, Cmdr dan Laravel (Putra et al., 2019).

Laragon merupakan aplikasi yang dapat merubah komputer menjadi sebuah server maupun lokal atau dapat disebut sistem *web stack* untuk pengembangan *web*. Laragon berada di lingkungan yang terikat dengan sistem operasi *Windows* dan menawarkan program-program pendukung untuk pengembangan *web modern* seperti Ruby on Rails, Laravel, Django, Flask, MEAN, dan Spring Boot serta penggunaan MySQL, PostgreSQL, MongoDB, Memcached, Redis, PHP, Ruby, Python, Node.js, dan Java yang mudah. Laragon dapat bekerja dengan baik untuk mengembangkan sebuah *web* dan memiliki kecepatan yang luar biasa. Laragon memiliki beberapa fitur unggulan seperti mendukung SSL, memungkinkan pengembang mengubah versi program pendukung dengan mudah, mengatur database, membuat proyek aplikasi dengan cepat, dan tampilan yang *user friendly*.

Laragon digunakan sebagai *Universal Development Environment* untuk *PHP*, Node.js, Python, Java, Go, Ruby, yang portable, terisolasi, cepat, ringan, dan mudah dipakai. Perangkat lunak sama halnya dengan XAMPP dan WAMP, yang digunakan untuk membangun *local development envirotment* pada sistem operasi *windows*. Namun selain menyertakan PHP, Apache Web Server, dan database MySQL dalam paket instalasinya, Laragon juga menyediakan segudang fitur menarik yang sangat relevan dengan kebutuhan

pengembangan *web*. Laragon adalah perangkat lunak bebas yang mendukung banyak sistem operasi, berfungsi sebagai *server* diri sendiri atau *localhost*.

### 2.10.7 Laravel Versi 5.0.0

Laravel adalah sebuah *framework* PHP yang dirilis dibawah lisensi MIT dan dikembangkan pertama kali oleh Taylor Otwell, dibangun dengan konsep MVC (*Model, View, Controller*). Laravel merupakan pengembangan *website* berbasis MVC yang ditulis dalam PHP yang dirancang untuk meningkatkan kualitas perangkat lunak dengan mengurangi biaya pengembangan awal dan biaya pemeliharaan, dan untuk meningkatkan (Awaludin, 2016).

MVC (*Model, View, Controller*) adalah sebuah pendekatan perangkat lunak yang memisahkan aplikasi logika dari presentasi. MVC memisahkan aplikasi berdasarkan komponen- komponen aplikasi seperti: manipulasi data, *controller*, dan *user interface*. Berikut detail dari pembagian fungsi dari MVC antara lain:

1. *Model* mewakili struktur data, biasanya model berisi fungsi-fungsi yang membantu seseorang dalam pengelolaan basis data seperti memasukkan data ke basis data, pembaruan data dan lain-lain.
2. *View* adalah bagian yang mengatur tampilan ke pengguna, bisa dikatakan berupa halaman web.
3. *Controller* merupakan bagian yang menjembatani model dan *view*.

Beberapa fitur keunggulan yang terdapat di Laravel:

1. *Bundles* yaitu sebuah fitur dengan sistem pengemasan modular dan tersedia beragam di aplikasi.
2. *Eloquent ORM* merupakan penerapan *PHP* lanjutan menyediakan metode internal dari pola "*active record*" yang mengatasi masalah pada hubungan objek *database*.
3. *Application Logic* merupakan bagian dari aplikasi, menggunakan *controller* atau bagian *Route*.
4. *Reverse Routing* mendefinisikan relasi atau hubungan antara *Link* dan *Route*.
5. *Restful controllers* memisahkan logika dalam melayani *HTTP, GET* dan *POST*.
6. *Class Auto Loading* menyediakan loading otomatis untuk *class PHP*.
7. *View Composer* adalah kode unit logikal yang dapat dieksekusi ketika *view* sedang *loading*.
8. *IoC Container* memungkinkan objek baru dihasilkan dengan pembalikan *controller*.
9. *Migration* menyediakan sistem kontrol untuk skema *database*.
10. *Unit Testing* banyak tes untuk mendeteksi dan mencegah regresi.

11. *Automatic Pagination* menyederhanakan tugas dari penerapan halaman.

### 2.10.8 MySQL Versi 5.7.33

Menurut Winarno et al. (2014) yang mengemukakan bahwa MySQL adalah sebuah *software database*. MySQL merupakan tipe data relasional yang artinya MySQL menyimpan datanya dalam bentuk table-tabel yang saling berhubungan. Keuntungan menyimpan data di *database* adalah kemudahannya dalam penyimpanan dan menampilkan data karena dalam bentuk tabel. Sedangkan Menurut Kustiyarningsih (2011), MySQL adalah sebuah basis data yang mengandung satu atau jumlah tabel. Tabel terdiri atas sejumlah baris dan setiap baris mengandung satu atau sejumlah tabel. MySQL merupakan sistem manajemen hubungan antara basis data yang sangat cepat dan sempurna. MySQL berupa alat bantu untuk memanipulasi basis data, sehingga bisa data dapat dengan mudah diisi, diambil, disusun dan diubah datanya. *Server MySQL* pun dapat mengatur kontrol akses dari data sehingga beberapa pengguna dapat sekaligus bekerja pada waktu yang bersamaan.

MySQL adalah *database server* yang sangat terkenal dan banyak digunakan untuk membangun aplikasi yang menggunakan *database* sebagai sumber dan pengolahan datanya, kepopuleran MySQL antara lain karena MySQL menggunakan SQL sebagai bahasa dasar untuk mengakses *database*-nya sehingga mudah untuk digunakan, kinerja *query* sangat cepat dan mencukupi untuk kebutuhan *database* perusahaan-perusahaan yang berskala kecil sampai menengah, MySQL juga bersifat tidak berbayar (Arief, 2011). Berikut beberapa kelebihan MySQL sebagai berikut yaitu:

1. Kemampuan yang tinggi.
2. Tidak dibutuhkan biaya untuk mendapatkan MySQL
3. Mudah untuk konfigurasi dan dipelajari
4. Dapat dijalankan pada beberapa sistem operasi seperti sistem *Unix* dan *Microsoft Windows*.

## 2.11 Pengujian Perangkat Lunak

Pengujian perangkat lunak merupakan suatu teknik yang digunakan menguji apakah sebuah perangkat lunak yang dihasilkan telah sesuai dengan yang diharapkan atau belum. Menurut (Pressman, 2015), pengujian adalah proses eksekusi suatu program untuk menemukan kesalahan sebelum digunakan oleh pengguna akhir (*end-user*).

### 2.11.1 Pengujian Scenario Test

Dalam penelitian ini menggunakan teknik *test case* berdasarkan *skenario test*. *Test case* adalah pengujian yang dilakukan berdasarkan beberapa masukan seperti kondisi

dan hasil yang telah ditentukan sebelumnya (Romeo, 2003)

Sebuah *Scenario Test* biasanya memiliki isi *pre-conditions*, *expected results* dan *post-conditions*. *Test case* bertindak sebagai titik awal untuk pelaksanaan tes dan setelah mengaplikasikan sekumpulan nilai input, aplikasi memiliki hasil yang definitif dan meninggalkan sistem di beberapa titik akhir atau juga dikenal sebagai *post-condition* eksekusi.

*Test case* yang baik adalah *test case* yang mempunyai kemungkinan tinggi dalam menemukan kesalahan dan cacat yang sebelumnya belum ditemukan, bukan yang dapat memperlihatkan bahwa program telah bekerja dengan benar (Romeo, 2003). *Test case* merupakan kunci untuk proses karena mengidentifikasi dan mengkomunikasikan kondisi yang akan diimplementasikan pada *test* dan yang diperlukan untuk memverifikasi keberhasilan.

**Tabel 2.8** Keterangan Kode

Kode	Keterangan Kode
V	<i>Valid</i> : Menunjukkan bahwa komponen yang membentuk skenario memiliki nilai yang benar atau <i>valid</i>
I	<i>Invalid</i> : Menunjukkan bahwa komponen yang membentuk skenario memiliki nilai yang salah atau <i>invalid</i>
NA	<i>Not Access</i> : Menunjukkan bahwa komponen yang membentuk skenario tersebut tidak memiliki peranan atau tidak bisa diakses pada saat tertentu

**Tabel 2.9** Pengujian *Scenario Test*

Kode	Fungsi	<i>Pre-Condition</i>	<i>Test Case</i>	Langkah Tes	Hasil yang diharapkan	Hasil
STA-AP1						
STA-AP2						

### 2.11.2 Pengujian *User Acceptance Test* (UAT)

*User Acceptance Test* (UAT) atau uji penerimaan pengguna adalah suatu proses pengujian oleh pengguna yang dimaksudkan untuk menghasilkan dokumen yang dijadikan

bukti bahwa perangkat lunak yang telah dikembangkan telah dapat diterima oleh pengguna, apabila hasil pengujian sudah bisa dianggap memenuhi kebutuhan dari pengguna. Pengujian UAT dilakukan oleh pengguna target dari aplikasi yang dibangun untuk mengetahui aplikasi berjalan sesuai dengan proses yang telah ditentukan bertujuan untuk mencatat dan mengoreksi *bug* yang ditemukan sebelum aplikasi di rilis. UAT biasanya dinilai oleh pengguna aplikasi tersebut, dengan menggunakan kusioner sesuai *test case* yang diinginkan.

Proses pengujian UAT diawali dengan menyediakan dokumentasi persyaratan bisnis, kemudian dilanjutkan dengan proses bisnis (alur kerja) atau skenario dan yang terakhir yaitu pengujian menggunakan data (Branch, 2008).

Pengujian UAT didasarkan pada dokumen *requirement* yang disepakati bersama. Dokumen *requirement* adalah dokumen yang berisi lingkup pekerjaan perangkat lunak yang harus dikembangkan, dengan demikian maka dokumen ini semestinya menjadi acuan untuk pengujian.

Proses dalam UAT meliputi pemeriksaan dan pengujian terhadap hasil pekerjaan. Setelah diperiksa selanjutnya memastikan item-item yang ada dalam dokumen *requirement* sudah ada dalam perangkat lunak yang diuji atau tidak. Selanjutnya menguji semua item yang telah ada telah dapat memenuhi kebutuhan penggunaannya. Hasil dari UAT adalah dokumen yang menunjukkan bukti pengujian, berdasarkan bukti pengujian inilah dapat diambil kesimpulan, apakah *software* yang diuji telah dapat diterima atau tidak. Dokumen dapat berupa kuesioner uji *test case*.

**Tabel 2.10** Pilihan Jawaban UAT

A	Sangat : Mudah/Bagus/Sesuai/Jelas
B	Mudah/Bagus/Sesuai/Jelas
C	Netral
D	Cukup : Sulit/ Bagus/Sesuai/Jelas
E	Sangat : Sulit/ Bagus/Sesuai/Jelas

**Tabel 2.11** Bobot Nilai Jawaban

Jawaban	Bobot
A. Sangat : Mudah/Bagus/Sesuai/Jelas	5
B. Mudah/Bagus/Sesuai/Jelas	4
C. Netral	3
D. Cukup : Sulit/ Bagus/Sesuai/Jelas	2

E. Sangat : Sulit/ Bagus/Sesuai/Jelas	1
---------------------------------------	---

**Tabel 2.12** Kriteria Interpretasi Skor

Jawaban	Bobot
81% - 100%	Sangat Baik
61% - 80%	Baik
41% - 60%	Cukup Baik
21% - 40%	Kurang Baik
0% - 20%	Sangat Kurang Baik

**Tabel 2.13** Pertanyaan Kuesioner

No	Pertanyaan	A	B	C	D	E
1						
2						
3						