



## BAB II

### TINJAUAN PUSTAKA

#### 2.1 Teori Judul

##### 2.1.1 Pengertian Aplikasi

Menurut Sommerville “Aplikasi (*application*) adalah kumpulan aktifitas yang menuju sebuah produksi perangkat lunak”

Menurut Elisa (2016), “Aplikasi sering juga disebut sebagai perangkat lunak, merupakan program komputer yang isi instruksinya dapat diubah dengan mudah.”

Sedangkan menurut Chan (2017:4), “Aplikasi adalah koleksi window dan objek-objek yang menyediakan fungsi untuk aktivitas user, seperti pemasukan data, proses dan pelaporan.”

Dari pengetian-pengertian diatas, penulis menyimpulkan bahwa aplikasi merupakan suatu perangkat lunak, software yang bertugas untuk melakukan sekumpulan perintah atau tugas-tugas tertentu yang dijalankan oleh user yang saling berkaitan.

##### 2.1.2 Pengertian Penjualan

Menurut Moekijat (2019), “Penjualan (*sales*) adalah aktivitas atau bisnis dalam menjual produk atau jasa. Aktivitas penjualan adalah hal yang sangat penting bagi perusahaan, terutama untuk meraih keuntungan. Pengertian penjualan secara umum adalah kegiatan jual beli dijalankan oleh dua belah pihak atau lebih dengan alat pembayaran yang sah.

Penjualan ini dapat dilakukan dengan berbagai metode, seperti penjualan langsung serta agen penjualan. Tujuan utama penjualan adalah mendatangkan keuntungan dari produk atau barang yang dijual. Dalam pelaksanaannya, penjualan tidak dapat dilakukan tanpa adanya kontribusi dari pelaku yang bekerja, seperti pedagang, agen, dan tenaga pemasaran.

### 2.1.3 Pengertian Material

Pengertian material mencakup setiap zat yang dipentingkan keberadaannya, penempatannya dalam ruang, dan sifat-sifat mekanikanya. Misalnya bahan bangunan, bahan untuk membuat mesin dan peralatan, bahan perabot, bahan pengemas, bahan pengisi dalam berbagai bidang (makanan, obat, kosmetik).

### 2.1.4 Pengertian Baru

Menurut KBBI, Baru adalah belum pernah ada (dilihat) sebelumnya.

### 2.1.5 Pengertian Metode *Cros Selling*

*Cross selling*, adalah teknik menjual produk (barang atau jasa) dengan cara menawarkan produk-produk yang lain, pada saat kita sedang membeli sebuah produk.

*Cross-selling* adalah suatu gagasan yang menjadi populer di akhir abad 20. Ekonom menggambarkan “gagasan sinergis bahwa pembeli dari salah satu layanan perusahaan akan menjadi pelanggan untuk yang lain”.

Sebagai contoh, kita sering sekali saat selesai berbelanja, begitu kita mau membayar, kasir tanpa sungkan akan menawarkan produk yang lain, seperti pulsa, atau beberapa produk / barang yang sengaja dipajang di dekat kasir.

Saat kita hendak membeli handphone, anggap kita sudah budget kan 2 juta, begitu sampai di counter HP, handphone tersebut memang berharga 2 juta, tapi begitu kita membeli, kita ditawari anti gores, misal seharga 100 rb, itung-itung dari pada nanti kacanya tergores, kan jadi jelek, begitu kira-kira kata sang penjaga counter, yang kita aminin juga.

### 2.1.6 Pengertian Website

Menurut Juniarto (2018), “Website adalah kumpulan informasi/page yang biasa di akses lewat jalur internet yang bisa diakses orang diberbagai tempat, dan segala waktu selama terhubung secara online”



### **2.1.7 Pengertian PT Rimbang Jaya**

PT. Rimbang jaya Kayuagung adalah perusahaan yang bergerak dalam bidang pendistribusian semen. Yang berdiri pada bulan maret 2017. Yang beralamat dijalan Letnan Mukthar Saleh No.17 Kelurahan Cinta Raja Kecamatan Kayuagung, Kabupaten Ogan Komerling Ilir, Sumatera Selatan 30618 .

### **2.1.7 Aplikasi Penjualan Bahan Material Bangunan pada PT Rimbang Jaya dengan Metode Cross Selling Berbasis Website**

Aplikasi Penerimaan Karyawan Baru Menggunakan Metode *Cross Selling* pada PT Rimbang Jaya merupakan aplikasi yang di gunakan perusahaan untuk membantu perusahaan dalam penentuan penjualan Material

## **2.2 Teori Khusus**

### **2.2.1 Object Oriented Program (OOP)**

Menurut Abdullah (2017) OOP (*Object Oriented Program*) merupakan teknik pemograman dengan menggunakan konsep objek. Tujuan dari OOP adalah untuk memudahkan programmer dalam pembuatan program dengan menggunakan konsep objek yang ada dalam kehidupan sehari-hari. Jadi setiap bagian permasalahan adalah objek, dan objek itu sendiri merupakan gabungan dari beberapa objek yang lebih kecil.

Sebuah objek pada OOP memiliki data atau disebut property yang menjelaskan tentang sifat-sifat objek tersebut. Seperti sebuah handpone dapat memiliki data warna, merk, ukuran layar, dan sebagainya. Begitu juga dengan objek-objek yang ada didalamnya seperti layar memiliki data berupa lebar, tinggi dan sebagainya.

Selain memiliki property, sebuah objek dalam OOP memiliki method berupa fungsi yang dapat dipanggil untuk melakukan tindakan atau merubah nilai dari property yang ada di dalamnya. Seperti handphone dapat melakukan tindakan merekam, restrart, memanggil, mengirim pesan dan sebagainya. Handphone juga dapat diganti *casing* untuk mengubah warnanya (mengubah nilai property).

### 2.2.2 Unified Modelling Language (UML)

Menurut Salahuddin (2014) Pada perkembangan teknologi perangkat lunak, diperlukan adanya bahasa yang digunakan untuk memodelkan perangkat lunak yang akan dibuat dan perlu adanya standarisasi agar orang di berbagai negara dapat mengerti pemodelan perangkat lunak.

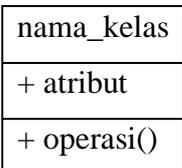
Pada perkembangan perkembangan teknik pemograman berorientasi objek, muncullah sebuah standarisasi bahasa pemodelan untuk pembangunan perangkat lunak yang dibangun dengan menggunakan teknik pemodelan berorientasi objek, yaitu *Unified Modeling Language* (UML). UML muncul karena adanya kebutuhan pemodelan visual untuk menspesifikasikan, menggambarkan, membangun, dan dokumentasi dari sistem perangkat lunak. UML merupakan bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks pendukung.

### 2.2.3 Class Diagram

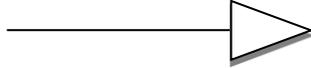
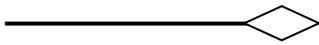
Menurut Salahuddin (2014) Diagram kelas atau *class diagram* menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi.

Diagram kelas dibuat agar pembuat program atau *programmer* membuat kelas-kelas sesuai rancangan di dalam diagram kelas agar antara dokumentasi perancangan dan perangkat lunak sinkron.

**Tabel 2.1** Simbol Class Diagram

No	Simbol	Deskripsi
1		Kelas pada struktur sistem

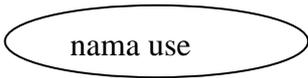
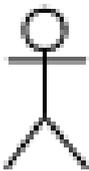
Lanjutan Tabel 2.1 Simbol Class Diagram

No	Simbol	Deskripsi
2	Antarmuka / <i>interface</i>  name_interface	sama dengan konsep <i>interface</i> dalam pemograman berorientasi objek.
3	Asosiasi / <i>association</i> 	Relasi antarkelas dengan makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i> .
4	Asosiasi berarah / <i>directed association</i> 	Relasi antarkelas dengan makna kelas yang saat digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan <i>multiplicity</i> .
5	Generalisasi 	Relasi antarkelas dengan makna generalisasi-spesialisasi (umum khusus).
6	Kebergantungan / <i>dependency</i> 	Relasi antarkelas dengan makna kebergantungan antar kelas.
7	Agregasi / <i>aggregation</i> 	Relasi antarkelas dengan makna semua-bagian ( <i>whole-part</i> )

#### 2.2.4 Use Case Diagram

Menurut Salahuddin (2014) *Use case* atau diagram *use case* merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Secara kasar, *use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu.

Tabel 2.2 Simbol Use Case Diagram

No	Simbol	Deskripsi
1	<i>Use case</i> 	Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antara unit atau aktor; biasanya dinyatakan dengan menggunakan kata kerja di awal fase nama <i>use case</i>
2	Aktor / <i>actor</i> 	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 fase nama aktor
3	Asosiasi / <i>association</i> 	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	Ekstensi / <i>extend</i> << extend >> 	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 <i>inheritance</i> pada pemrograman berorientasi objek; biasanya <i>use case</i> tambahan memiliki nama depan yang sama dengan <i>use case</i>

Lanjutan **Tabel 2.2** Simbol Use Case Diagram

No	Simbol	Deskripsi
5	Generalisasi / <i>generalization</i> 	Hubungan generalisasi dan spesialisasi (umum-khusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya.
6	Menggunakan / <i>include</i> / <i>uses</i>  << include >> 	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> di mana <i>use case</i> yang ditambahkan memerlukan <i>use case</i> ini untuk menjalankan fungsinya atau sebagai syarat dijalankan <i>use case</i> ini

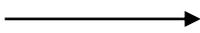
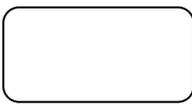
### 2.2.5 Activity Diagram

Menurut Salahuddin (2014) Diagram aktivitas atau *activity diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Yang perlu diperhatikan disini adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem.

**Tabel 2.3** Simbol Activity Diagram

No	Simbol	Deskripsi
1	Start / status awal ( <i>Intial State</i> ) 	Start atau intial state adalah state atau keadaan awal pada saat sistem mulai hidup

Lanjutan Tabel 2.5 Simbol Activity Diagram

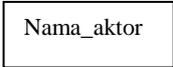
No	Simbol	Deskripsi
2	End / status akhir ( <i>final state</i> ) 	End atau final state adalah state keadaan akhir dari daur hidup suatu sistem.
3	Event 	Event adalah kegiatan yang menyebabkan berubahnya status mesin.
4	State 	Sistem pada waktu tertentu. State dapat berubah jika ada event tertentu yang memicu perubahan tersebut.

### 2.2.6 Sequence Diagram

Menurut Salahuddin (2014) Diagram sekuen menggambarkan kelakuan objek pada use case dengan mendeskripsikan waktu hidup objek dan message yang dikirimkan dan diterima antar objek. Oleh karena itu untuk menggambarkan diagram sekuen maka harus diketahui objek-objek yang terlibat dalam sebuah use case beserta metode-metode yang dimiliki kelas yang diinstantiasi menjadi objek itu. Membuat diagram sekuen juga dibutuhkan untuk melihat skenario yang ada pada use case.

Banyaknya diagram sekuen yang harus digambar adalah minimal sebanyak pendefinisian use case yang memiliki proses sendiri atau yang penting semua use case yang telah didefinisikan interaksinya pesan sudah dicakup pada diagram sekuen sehingga semakin banyak use case yang didefinisikan maka diagram sekuen yang harus dibuat juga semakin banyak.

**Tabel 2.4** Simbol Sequence Diagram

No	Simbol	Deskripsi
	Aktor  atau 	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 frase nama aktor.

## 2.3 Teori Program

### 2.3.1 Pengertian PHP

Menurut Supono dan Virdiandry (2018:3) PHP adalah suatu bahasa pemrograman yang di gunakan untuk menerjemahkan baris kode program menjadi kode mesin yang dapat dimengerti oleh computer yang bersifat *server-side* yang dapat ditambahkan ke HTML. Sedangkan menurut Sidik (2017:4) PHP merupakan secara umum dikenal sebagai bahasa pemrograman script-script yang membuat dokumen HTML secara *on the fly* yang dieksekusi di server web, dokumen HTML yang dihasilkan dari suatu aplikasi bukan dokumen HTML yang dibuat dengan menggunakan editor teks atau editor HTML.

### 2.3.2 Pengertian MySQL



Menurut Sidik (2017:301) MySQL merupakan database yang termasuk paling populer di Linux, kepopuleran ini karena ditunjang karena performansi query dari databasenya yang saat itu bias di katakana paling cepat, dan jarang bermasalah. MySQL telah tersedia juga di lingkungan windows. Sedangkan Anhar (2010:21) berpendapat bahwa MySQL adalah sebuah perangkat lunak system manajemen basis data SQL (Database Management System) atau DBMS dari sekian banyak DBMS, seperti Oracle, MS SQL, Postagre SQL dan lain-lain.