

BAB II

TINJAUAN PUSTAKA

II.1. Penelitian Terdahulu

Berdasarkan penelitian yang dilakukan Suriyanto (2016) dengan judul Pengelolaan Kotak Review/Ulasan Pada Sistem Penjualan Pada Toko *Online*, Suriyanto menyimpulkan toko *online* menjadi sebuah terobosan yang dihasilkan dari perkembangan teknologi informasi dan perkembangan teknologi komputer. Agar sebuah toko *online* dapat bertahan dari persaingan yang sangat ketat baik dengan toko *online* lainnya atau dengan model penjualan barang yang lain.

Berdasarkan penelitian yang dilakukan Lenti (2017) dengan judul Rekaya Proses Bisnis Pada *E-Commerce* B2B-B2C Menggunakan Sistem Afiliasi, Lenti menyimpulkan penelitian ini menghasilkan suatu sistem *E-Commerce* B2B-B2C berbasis *online* untuk antar muka konsumen akhir bertransaksi dan berbasis *mobile* untuk antar muka antara *affiliator* dan *supplier* / pemilik barang. Dari Hasil Pengujian sistem dapat mengelola data *affiliator*, data barang, transaksi penjualan dan komisi untuk *affiliator*, produk-produk baru ter-*update* secara *real time* data para *affiliator* teridentifikasi dengan baik.

Berdasarkan penelitian yang dilakukan Setyawan (2018) dengan judul Model Estimasi Relationship Marketing Dalam *Business to Business*: Pendekatan Dyadic VS Non-Dyadic, Setyawan menyimpulkan hasil penelitian ini konsisten dengan asumsi dalam aliran pemikiran IMP (*Industrial or International Marketing and Purchasing*) Group yang menunjukkan bahwa sifat dasar konstruk

atau variabel dalam penelitian *relationship marketing* adalah *dyadic* dan tentang kuatnya dominasi *Business to Business* dalam penelitian *relationship marketing*. Saran bagi peneliti yang akan meneliti tentang *Business to Business* adalah menggunakan pendekatan *dyadic*.

Berdasarkan penelitian yang dilakukan Utami dan Hasanah (2017) dengan judul Model Pemasaran *Business-To-Business* Dan Jaringan Nilai Produk Agro Industri Olahan Tebu Molasses, Utami dan Hasanah menyimpulkan Pemasaran relasional berdasarkan dimensi budaya yang dilakukan antar perusahaan B2B melalui dimensi jarak kekuasaan, orientasi social dan penghindaran ketidakpastian menunjukkan adanya perbedaan orientasi pemasaran yang dilakukan oleh perusahaan yang melakukan pemasaran secara relasional dengan perusahaan yang melakukan pemasaran trasaksional, terutama perbedaan dalam faktor kepastian harga, penyampaian informasi, tingkat jalinan informasi serta pentingnya peraturan tertulis yang mengikat masing- masing pihak yang terlibat dalam aktivitas pemasaran tersebut termasuk di dalamnya mengenai hak dan kewajiban serta sanksi yang akan diterima oleh pihak yang melakukan pelanggaran perjanjian pemasaran.

Berdasarkan penelitian yang dilakukan Yunita dan Suryani (2018) dengan judul Perancangan *E-Commerce* Batik Pada Batik Banten, Yunita dan Suryani menyimpulkan bahwa dengan adanya sistem informasi ini Batik Banten ingin memberikan pelayanan yang lebih baik untuk konsumennya yaitu dengan cara membangun perancangan *e-commerce* pada Batik Banten. Adapun pengembangan sistem yaitu dengan metode *waterfall*. Dengan adanya perancangan sistem

informasi ini akan sangat membantu konsumen dalam pembelian batik. Konsumen bisa memesan batik sesuai dengan pilihan tanpa harus pergi ke toko.

II.2. Landasan Teori

Berikut ini adalah landasan teori yang berkaitan dengan penelitian yang di ambil dari beberapa jurnal dan buku:

II.2.1. *Business to Business (B2B)*

Business to Business (B2B) adalah model *e-commerce* dimana pelaku bisnisnya adalah perusahaan, proses transaksi yang terjadi melibatkan antara satu perusahaan dengan perusahaan lainnya. Contoh model *e-commerce* ini adalah beberapa situs *e-banking* yang melayani transaksi antar perusahaan. (Suriyanto, 2016: 45).

Business to Business (B2B) adalah bentuk interaksi *e-commerce* secara *online* yang terjadi antara produsen (perusahaan, industri, penyedia Bahan baku dan jasa) dengan distributor (*Supplier*) dan pengecer. Distributor atau pengecer ini kemudian menyalurkan produk tersebut ke konsumen masing masing. Didalam proses B2B terjadi kegiatan yang mencakup *Supply Chain*, pertukaran informasi, manajemen operasional dan lain lain. Ada tiga tahap pada *e-commerce B2B*, yaitu: *Physical Distribution*, *Information Exchange*, dan *Business Process Integration*. Pada tahap *Physical Distribution* terjadi distribusi produk secara fisik, dimana didalamnya melibatkan *Manufacturing Supply Chain*. *Manufacturing Supply Chain* terdiri atas *Supplier (S)*, *Manufacturer (M)* dan *Customer(C)*. *Manufacturing Supply Chain* terdiri atas dua bagian yaitu:

1. *Upstream (level atas)* yang melibatkan *Supplier (S)* dengan *manufacturer (M)*. *Manufacturer* menyalurkan produk ke *supplier*, bukan ke konsumen akhir.
2. *Downstream (level bawah)* yang melibatkan *Manufacturer (M)* dengan *Customer (C)*. *Manufacturer* menyalurkan produk langsung ke konsumen akhir. (Lenti, 2017: 42).

II.2.2. Karakteristik *Business to Business (B2B)*

Business to Business memiliki karakteristik:

1. *Trading partners* yang sudah saling mengetahui dan antara mereka sudah terjalin hubungan yang berlangsung cukup lama. Informasi yang dimiliki hanya ditukar dengan partner tersebut.
2. Pertukaran data dilakukan secara berulang-ulang dan berkala dengan format data yang telah disepakati bersama.
3. Salah satu pelaku tidak harus menunggu rekan mereka lainnya untuk mengirimkan data.
4. Model yang umum digunakan adalah *peer to peer* dimana *processing intelligence* dapat di distribusikan di kedua pelaku bisnis. (Yunita dan Suryani, 2018: 68).

II.2.3. Penjualan

Penjualan merupakan kegiatan yang dilakukan oleh penjual dalam menjual Bahan baku atau jasa dengan harapan akan memperoleh laba dari adanya transaksi-transaksi tersebut dan penjualan dapat diartikan sebagai pengalihan atau

pemindahan hak kepemilikan atas Bahan baku atau jasa dari pihak penjual ke pembeli. (Arfianto dan Nugrahanti, 2018: 175).

Penjualan merupakan kegiatan pemasaran yang langsung berhubungan dengan konsumen pengguna atau pemakai langsung. (Mawarni, dkk, 2020: 1).

II.2.4. Bahan Baku

Salah satu fungsi pokok perusahaan manufaktur adalah fungsi produksi. Sebagai fungsi produksi, perusahaan bertugas mengolah bahan baku menjadi produk jadi. Bahan baku adalah Bahan baku yang dibeli perusahaan untuk digunakan dalam proses produksi. Bahan baku adalah bahan yang menjadi bagian produk jadi dan dapat diidentifikasi ke produk jadi. Bahan baku adalah persediaan yang dibeli oleh perusahaan untuk diproses menjadi Bahan baku setengah jadi dan akhirnya Bahan baku jadi atau produk akhir dari perusahaan. Berdasarkan pendapat tersebut, dapat disimpulkan bahwa bahan baku merupakan bahan yang dibuat menjadi Bahan baku jadi. Bahan Baku adalah Bahan baku-Bahan baku yang dibeli untuk digunakan dalam proses produksi. Sebagian bahan baku diambil langsung dari sumber aslinya. Namun yang lebih sering terjadi, bahan baku dibeli dari perusahaan lain yang merupakan Bahan baku jadi dari sisi pemasok. (Maryanto, 2020: 9).

II.2.5. Java

Java adalah sebuah bahasa pemrograman yang populer kalangan para akademisi dan praktisi komputer. *Java* pertama kali dikembangkan untuk memenuhi kebutuhan akan sebuah bahasa komputer yang ditulis satu kali dan dapat dijalankan di banyak sistem komputer berbeda tanpa perubahan kode berarti.

Pada umumnya, para pakar pemrograman berpendapat bahwa bahasa *java* memiliki konsep yang konsisten dengan teori pemrograman objek dan aman untuk digunakan. (Wardhani dan Yaqin, 2015: 474).

II.2.6. Web

Web adalah salah satu aplikasi yang berisikan dokumen-dokumen multimedia (teks, gambar, suara, animasi, video) di dalamnya yang menggunakan *protocol HTTP (hypertext transfer protocol)* dan untuk mengaksesnya menggunakan perangkat lunak yang disebut *browser*. (Arfianto dan Nugrahanti, 2018: 175).

II.2.7. Android

Android adalah salah satu *platform* sistem operasi yang digemari masyarakat karena sifatnya yang *open source* sehingga memungkinkan pengguna untuk melakukan pengembangan. *Android* merupakan generasi baru *platform mobile* berbasis *linux* yang mencakup sistem operasi, *middleware*, dan aplikasi. Arsitektur *Android* terdiri dari bagian-bagian seperti berikut:

1. *Applications* dan *Widgets: layer* (lapisan) dimana pengguna hanya berhubungan dengan aplikasi saja.
2. *Applications Framework*: lapisan dimana para pengembang melakukan pembuatan aplikasi yang akan dijalankan di sistem operasi *Android* dengan komponen-komponennya meliputi *views*, *contents provider*, *resource manager*, *notification manager*, *activity manager*.

3. *Libraries*: lapisan dimana fitur-fitur *Android* berada yang berada diatas kernel meliputi *library C/C++* inti seperti *Libc* dan *SSL*.
4. *Android Run Time*: lapisan yang membuat aplikasi *Android* dapat dijalankan dimana dalam prosesnya menggunakan implementasi *Linux* yang terbagi menjadi dua bagian yaitu *Core Libraries* dan *Dalvik virtual Machine*.
5. *Linux Kernel: Layer* yang berisi *file-file system* untuk mengatur *processing*, *memory*, *resource*, *driver*, dan sistem operasi *Android* lainnya.

Sistem operasi yang mendasari *Android* dilisensikan dibawah GNU, GPLv2 (*General Public License Versi 2*) yang sering dikenal dengan istilah *copyleft*. Pendistribusian *Android* dibawah lisensi dari *Apache Software* yang memungkinkan untuk distribusi kedua dan seterusnya. (Andilala dan Gunawan, 2018: 15).

II.2.8. Hypertext Preprocessor (PHP)

Hypertext Preprocessor atau yang sering disebut dengan PHP merupakan bahasa pemrograman yang bersifat *Open Source*. Bahasa pemrograman ini bersifat *server side*, maksudnya adalah tanpa adanya *server* yang berjalan beriringan dengan program atau aplikasi *script* PHP tidak dapat berjalan. PHP merupakan *script* yang disisipkan dalam *code* HTML untuk membuat halaman pada *Website* bekerja secara otomatis dan dapat berfungsi sebagai pengolah data pada *server* dimana *script* tersebut dijalankan. PHP memiliki kemampuan untuk memisahkan diri dari kode HTML saat sebuah *Website* dilakukan “*View Page Source*” oleh *user*.(Septyasari, dkk, 2018: 215).

PHP adalah singkatan dari *PHP Hypertext Preprocessor*. PHP adalah

bahasa *server-side scripting* yang menyatu dengan HTML (*Hypertext Markup Language*) untuk membuat halaman *Web* yang dinamis. Maksud dari *server-side scripting* adalah sinaks dan perintah-perintah yang diberikan akan sepenuhnya dijalankan di *server* tetapi disertakan pada dokumen HTML sebagai pembangun halaman *Web*. (Kosasih, dkk, 2017: 115).

II.2.9. Javascript

Javascript merupakan bahasa pemograman *Website* yang bersifat *Client Side Programming Language*. Yang artinya bahsa pemograman yang dalam pemrosesannya dilakukan pada sisi *client*. Bahasa pemograman ini berbeda dengan bahsa pemograman PHP yang dalam pemrosesannya dilakukan pada sisi *server*. *Javascript* sendiri memiliki fitur yang berupa: *high-level programming language, client-side, loosely typed* dan berorientasi objek.

Javascript memiliki fungsi untuk membuat interaksi antar *user* dan *Website* menjadi lebih cepat tanpa harus menunggu proses yang dilakukan pada *Web server*. *JavaScript* dapat digunakan untuk melakukan validasi *form*, animasi, fitur *chatting, games*, dan lainnya yang semuanya dapat dibuat menggunakan *JavaScript*. (Septyasari, dkk, 2017: 215).

II.2.10. XAMPP

XAMPP adalah program aplikasi pengembang yang berguna untuk pengembangan *Website* berbasis PHP dan MySQL. Perangkat lunak komputer ini memiliki kelebihan untuk bisa berperan sebagai *server Web Apache* untuk simulasi pengembangan *Website*. *Tool* pengembangan *Web* ini mendukung

teknologi *Web* populer seperti PHP dan MySQL. Melalui program ini, *programmer Web* dapat menguji aplikasi *Web* yang dikembangkan dan mempresentasikannya ke pihak lain secara langsung dari komputer, tanpa perlu terkoneksi ke internet. XAMPP juga dilengkapi fitur manajemen *database* PHP MyAdmin seperti pada *server hosting* sungguhan, sehingga pengembang *Web* dapat mengembangkan *Web* berbasis *database* dengan mudah. (Kosasih, dkk, 2017: 115).

II.2.11. *Unified Modeling Language (UML)*

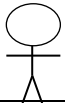
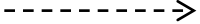
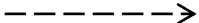
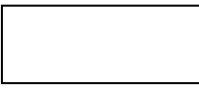
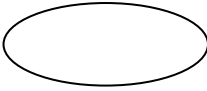
Unified Modeling Language (UML) yaitu satu kumpulan konvensi permodelan yang digunakan untuk menentukan atau menggambarkan sebuah sistem perangkat lunak yang terkait dengan objek. UML merupakan suatu kumpulan teknik terbaik yang telah terbukti sukses dalam memodelkan sistem yang besar dan kompleks. UML tidak hanya digunakan dalam proses pemodelan perangkat lunak, namun hampir dalam semua bidang yang membutuhkan pemodelan. (Andikos, 2019: 39).

Alat bantu yang digunakan dalam perancangan berorientasi objek berbasis UML adalah sebagai berikut:

1. *Use Case Diagram*

Use case diagram menggambarkan fungsionalitas yang diharapkan dari sebuah sistem. Yang ditekankan adalah “apa” yang diperbuat sistem, dan bukan “bagaimana”. Sebuah *use case* merepresentasikan sebuah interaksi antara aktor dengan sistem. *Use case diagram* dapat digambarkan dengan sumber-sumber pada Tabel II.1.

Tabel II.1. Simbol Use Case

Gambar	Nama	Keterangan
	<i>Actor</i>	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i> .
	<i>Depedency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri (<i>independent</i>).
	<i>Generalization</i>	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).
	<i>Include</i>	Menspesifikasikan bahwa <i>use case</i> sumber secara eksplisit.
	<i>Extend</i>	Menspesifikasikan bahwa <i>use case</i> target memperluas perilaku dari <i>use case</i> sumber pada suatu titik yang diberikan.
	<i>Association</i>	Apa yang mnghubungkan antara objek satu dengan objek lainnya.
	<i>System</i>	Menspesifikasikan paket yang menampilkan sistem secara terbatas.
	<i>Use Case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu <i>actor</i> .
	<i>Collaboration</i>	Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan prilaku yang lebih besar dari jumlah dan elemen-elemennya (<i>sinergi</i>).
	<i>Note</i>	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi

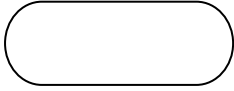
(Sumber: Andikos, 2019: 39)

2. Diagram Aktivitas (*Activity Diagram*)

Activity diagram menggambarkan berbagai alir aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing alir berawal, *decision* yang mungkin terjadi, dan bagaimana mereka berakhir. *Activity diagram* juga dapat

menggambarkan proses paralel yang mungkin terjadi pada beberapa eksekusi. *Activity diagram* dapat digambarkan dengan simbol-simbol seperti pada tabel II.2.

Tabel II.2. Simbol *Activity Diagram*

Gambar	Nama	Keterangan
	<i>Activity</i>	Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain.
	<i>Initial Node</i>	Bagaimana objek dibentuk atau diawali.
	<i>Activity Final</i>	Bagaimana objek dibentuk dan dihancurkan
	<i>Fork Node</i>	Satu aliran yang pada tahap tertentu berubah menjadi beberapa aliran

(Sumber: Andikos, 2019: 39)

3. Diagram Urutan (*Sequence Diagram*)

Sequence diagram menggambarkan interaksi antar objek di dalam dan di sekitar sistem (termasuk pengguna, *display*, dan sebagainya) berupa *message* yang digambarkan terhadap waktu. *Sequence Diagram* dapat digambarkan dengan simbol-simbol seperti pada Tabel II.3.

Tabel II.3. Simbol *Sequence Diagram*

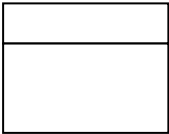
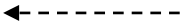
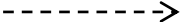
Gambar	Nama	Keterangan
	<i>Lifeline</i>	Objek entity, antarmuka yang saling berinteraksi.
	<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi.
	<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi.

(Sumber: Andikos, 2019: 39)

4. *Class Diagram* (Diagram Kelas)

Class adalah sebuah spesifikasi yang jika diinstansiasi akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek. *Class diagram* menggambarkan struktur dan deskripsi *class*, *package* dan objek beserta hubungan satu sama lain seperti *containment*, pewarisan, asosiasi, dan lain-lain. *Class diagram* dapat digambarkan dengan simbol-simbol seperti pada Tabel II.4.

Tabel II.4. Class Diagram

Gambar	Nama	Keterangan
	<i>Generalization</i>	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).
	<i>Nary Association</i>	Upaya untuk menghindari asosiasi dengan lebih dari 2 objek.
	<i>Class</i>	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.
	<i>Collaboration</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor.
	<i>Realization</i>	Operasi yang benar-benar dilakukan oleh suatu objek.
	<i>Depedency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan mempegaruhi elemen yang bergantung padanya elemen yang tidak mandiri.
	<i>Assocation</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya

(Sumber: Andikos, 2019: 39)

II.2.12. Aplikasi

Aplikasi adalah program siap pakai yang dapat digunakan untuk menjalankan perintah- perintah dari pengguna aplikasi tersebut dengan tujuan mendapatkan hasil yang lebih akurat sesuai dengan tujuan pembuatan aplikasi tersebut, aplikasi mempunyai arti yaitu pemecahan masalah yang menggunakan salah satu tehnik pemrosesan data aplikasi yang biasanya berpacu pada sebuah

komputansi yang diinginkan atau diharapkan maupun pemrosesan data yang diharapkan. (Limbong, 2017: 81).

Aplikasi didefinisikan sebagai suatu kelompok *file* yang bertujuan untuk melakukan aktivitas tertentu yang saling terkait. Aplikasi merupakan perangkat lunak komputer yang dimanfaatkan pengguna untuk menyelesaikan atau mempermudah pengguna dalam menyelesaikan tugas. (Mesran, 2016: 81).

Berdasarkan jenisnya aplikasi terbagi kedalam beberapa kategori:

1. Aplikasi jaringan.
2. Aplikasi grafik.
3. Aplikasi animasi.
4. Aplikasi *Web*.
5. Aplikasi pendidikan.
6. Aplikasi perkantoran. (Mesran, 2016: 81).

II.2.13. MySQL

MySQL adalah sebuah perangkat lunak sistem manajemen basis data SQL (bahasa Inggris *database management system*) atau DBMS yang *multithread*, *multi-user*, dengan sekitar 6 juta instansi di seluruh dunia. MySQL AB membuat MySQL tersedia sebagai perangkat lunak gratis di bawah lisensi GNU *General Public License* (GPL), tetapi mereka juga menjual dibawah lisensi komersial untuk kasus-kasus dimana penggunaannya tidak cocok dengan penggunaan GPL. (Kosasih, dkk, 2017: 115).

MySQL adalah *database open source* terpopuler di dunia. Dengan kinerja, kehandalan dan kemudahan penggunaan yang terbukti, MySQL telah menjadi pilihan *database* terdepan untuk aplikasi berbasis *Web*, yang digunakan oleh properti *Web profil* tinggi termasuk Facebook, Twitter, YouTube, Yahoo! dan banyak lagi. Kata “SQL” dari "MySQL" adalah singkatan dari "*Structured Query Language*". SQL adalah bahasa standar yang paling umum digunakan untuk mengakses *database*. Bergantung pada lingkungan pemrograman yang kita pakai, kita bisa memasukkan SQL secara langsung (misalnya, untuk men-*generate* laporan), memasukkan pernyataan SQL ke dalam kode yang ditulis dalam bahasa lain, atau menggunakan API khusus yang dapat menyembunyikan sintaks SQL. (Arfianto dan Nugrahanti, 2018: 175).