

BAB II

LANDASAN TEORI

2.1 Sistem Informasi

Sistem informasi adalah cara-cara yang diorganisasi untuk mengumpulkan, memasukkan, dan mengolah serta menyimpan data, dan cara-cara yang diorganisasi untuk menyimpan, mengelola, mengendalikan, dan melaporkan informasi sedemikian rupa sehingga sebuah organisasi dapat mencapai tujuan yang telah ditetapkan (Krismiaji, 2016).

2.1.1 Manfaat Sistem Informasi

Organisasi menggunakan sistem informasi untuk mengolah transaksi-transaksi, mengurangi biaya dan menghasilkan pendapatan sebagai salah satu produk atau pelayanan mereka. Perusahaan menggunakan sistem informasi untuk mempertahankan persediaan pada tingkat paling rendah agar konsisten dengan jenis barang yang tersedia (Krismiaji, 2016).

2.1.2 Komponen Sistem Informasi

Sistem informasi terdiri dari komponen-komponen yang disebut dengan istilah blok bangunan (building block). Adapun penjelasan dari blok bangunan (Hutahean, 2015) yaitu

1. Blok masukan (input block)

Blok masukan merupakan blok yang bertugas dalam input data agar masuk ke dalam sistem informasi. Blok masukan bertugas dalam merekam data yang akan dimasukkan, biasanya berupa dokumen-dokumen dasar.

2. Blok model (model block)

Blok model terbentuk dari kombinasi prosedur, logika dan model matematik yang memproses data input dan data yang tersimpan di basis data dengan cara yang sudah ditentukan untuk menghasilkan keluaran yang diinginkan.

3. Blok keluaran (output block)

Sistem informasi menghasilkan keluaran (output) yaitu informasi yang berkualitas dan berguna untuk semua tingkatan manajemen serta semua pemakai sistem.

4. Blok teknologi (technology block)

Teknologi digunakan merupakan kotak alat dalam sistem informasi. Teknologi digunakan untuk menerima input, menjalankan model, menyimpan dan mengakses data, menghasilkan dan mengirimkan keluaran berupa informasi dan membantu pengendalian dari sistem secara menyeluruh. Bloteknologi perangkat lunak (software) dan perangkat keras (hardware) yang dioperasikan oleh teknisi (brainware).

5. Blok basis data (database block)

Basis data (database) merupakan media untuk menyimpan data yang saling berhubungan satu sama lainnya, tersimpan di perangkat keras komputer dan dapat dipergunakan kembali, diperlukan perangkat lunak untuk memanipulasinya.

6. Blok kendali (control block)

Sistem informasi memiliki kontrol kendali untuk menanggulangi gangguan-gangguan terhadap sistem apabila terlanjur terjadi kesalahan maka dapat langsung diantisipasi atau diatasi

2.2 Persediaan (*Inventory*)

Persediaan adalah salah satu aset yang sangat penting bagi suatu entitas baik bagi perusahaan ritel, manufaktur, jasa, maupun entitas lainnya. (Martani dkk, 2016).

Persediaan adalah suatu aktiva lancar yang ada dalam suatu perusahaan, apabila perusahaan tersebut perusahaan dagang maka persediaan diartikan sebagai barang dagangan yang disimpan untuk dijual dalam operasi normal perusahaan. Sedangkan apabila perusahaan merupakan perusahaan manufaktur maka persediaan diartikan sebagai baan baku yang terdapat dalam proses produksi/ yang disimpan untuk tujuan tersebut (Heizer & Render, 2015).

2.2.1 Jenis-Jenis Persediaan

Persediaan dapat diklasifikasikan sebagai berikut (Martani dkk, 2016):

1. Persediaan barang baku

Barang berwujud yang dibeli atau diperoleh dengan cara lain (misalnya dengan menabung) dan disimpan untuk penggunaan langsung dalam membuat barang untuk dijual kembali.

2. Persediaan barang dalam proses

Barang yang terdiri dari baha-bahan yang telah diproses namun masih membutuhkan pekerjaan lebih lanjut sebelum dijual. Persediaan bahan dalam proses, pada umumnya dinilai jumlah harga pokok bahan baku, biaya tenaga kerja langsung, dan biaya overhead pabrik yang telah dikeluarkan/terjadi sampai dengan tanggal tertentu

3. Barang jadi

Adalah barang yang sudah selesai diproduksi dan siap untuk dipasarkan. Persediaan produk jadi, meliputi semua barang yang telah diselesaikan dari proses produksi dan siap untuk dijual. Produk jadi pada umumnya dinilai sebesar jumlah harga pokok bahan baku, biaya tenaga kerja langsung dan biaya overhead pabrik yang diperlukan untuk menghasilkan prosuk tersebut.

4. Persediaan barang penolong

Meliputi semua barang yang dimiliki untuk keperluan produksi, tetapi tidak merupakan bahan baku yang membentuk produk jadi

2.2.2 Fungsi Persediaan

Fungsi utama persediaan yaitu sebagai penyangga, penghubung antar proses produksi dan distribusi untuk memperoleh efisiensi. Fungsi lain persediaan yaitu sebagai stabilisator harga terhadap fluktuasi permintaan. Fungsi persediaan, yaitu (Martani dkk, 2016):

1. Fungsi *Decuoping*, untuk membantu persediaan agar bisa memenuhi permintaan langganan tanpa tergantung pada supplier.
2. Fungsi *Econmic lot sizing*, persediaan ini perlu mempertimbangkan penghematan-penghematan (potongan pembelian, biaya pengangkutan per unit lebih murah dan sebagainya) karena perusahaan melakukan pembelian dalam kuantitas yang besar. Dibandingkan dengan biayabiaya yang timbul Karena besarnya persediaan (biaya sewa gudang, investasi, risiko, dan sebagainya).
3. Fungsi antisipasi, untuk mengantisipasi dan mengadakan permintaan musiman (seasonal inventories), menghadapi ketidak pastian jangka waktu pengiriman dan untuk menyediakan persediaan pengamanan (*safety stock*).

Sedangkan menurut (Heizer & Render, 2015) fungsi persediaan adalah:

1. Untuk memisahkan beragam bagian produksi, sebagai contoh jika pasokan sebuah perusahaan berfluktuasi, maka mungkin diperlukan persediaan-persediaan tambahan men-decouple proses produksi dari pemasok.
2. Untuk men-decouple perusahaan dari fluktuasi permintaan dan menyediakan persediaan barang-barang yang akan memberikan pilihan bagi pelanggan. Persediaan semacam ini umumnya terjadi pada pedagang eceran.
3. Untuk mengambil keuntungan diskon kuantitas, sebab pembelian dalam jumlah lebih besar dapat mengurangi biaya produksi atau pengiriman barang.
4. Untuk menjaga pengaruh inflasi dan naiknya harga.

2.3 *First Input First Output (FIFO)*

Metode *First In First Out* (FIFO) adalah metode yang menganggap barang yang dibeli lebih dulu maka akan dijual lebih dulu, sehingga harga perolehan barang yang dibeli pertama kali akan dibebankan lebih dahulu sebagai harga pokok penjualan. Metode ini konsisten dengan arus biaya aktual, dimana persediaan lama dijual pertama kali. Metode *First In First Out* (FIFO) seringkali tidak terlihat secara langsung pada aliran fisik dari barang tersebut karena pengambilan barang dari gudang lebih didasarkan pada pengaturan barangnya, sehingga metode *First In First Out* (FIFO) lebih terlihat pada perhitungan harga pokok barang. Dalam metode *First In First Out* (FIFO), biaya yang digunakan untuk membeli barang pertama kali akan dikenali sebagai harga pokok penjualan dan untuk perhitungan harga akan menggunakan harga dari stok barang dari transaksi yang terdahulu (Riswan dan Fasa, 2016).

Dalam metode FIFO menghasilkan jumlah persediaan akhir tinggi yang berpengaruh pada jumlah aset pada laporan posisi keuangan mengalami kenaikan, dan harga pokok penjualan menjadi rendah sehingga laba yang dihasilkan perusahaan menjadi tinggi, sehingga metode ini lebih cocok digunakan dalam kondisi ekonomi saat ini dibanding menggunakan metode rata-rata (Mahardika dan Nuraina, 2015).

Metode *First In First Out* (FIFO) dapat dihitung dengan sistem periodik atau sistem perpetual. Dalam sistem periodik, persediaan akhir ditentukan dengan cara mengkalikan nilai persediaan

yang ada dengan harga pokok per unit barang yang terakhir kali masuk, jika nilai persediaan ternyata lebih besar dari jumlah barang yang terakhir masuk maka sisanya menggunakan harga pokok per unit yang masuk sebelumnya. Sedangkan pada sistem perpetual, persediaan akan dicatat terus menerus dalam kartu persediaan, harga pokok penjualan akan dicatat berdasarkan harga pokok barang pertama masuk dan jumlah yang masih tersisa merupakan nilai persediaan akhir (Sari dan Suzan, 2015).

2.3.1 Kelebihan FIFO

Kelebihan dari metode FIFO :

1. Barang lebih terjaga kualitasnya

Penjelasan kali ini akan mengulas lebih banyak mengenai penyimpanan barang di gudang dengan metode atau sistem penyimpanan barang FIFO (First in First Out). Metode yang mengandalkan prinsip barang yang pertama masuk adalah yang pertama keluar ini memiliki beberapa kelebihan.

Contohnya, bila gudang penyimpanan beras, beras tersebut disimpan terlalu lama dalam gudang, maka dikhawatirkan beras akan rusak dan mengundang kutu. Jika ini terjadi, maka nantinya juga dapat berpengaruh terhadap kualitas beras lain yang mungkin juga akan terserang kutu.

2. Pengendalian harga lebih terjamin

Selain dapat menjaga kualitas barang, kelebihan sistem FIFO lainnya adalah adanya kestabilan harga dari barang-barang yang disimpan. Perlu diketahui bahwa tidak selamanya harga dapat selalu sama. Ada kalanya harga naik, dan ada pula kalanya harga tersebut turun.

Dengan sistem FIFO, diharapkan barang yang pertama kali masuk dengan harga tertentu dapat terjual sama harganya (sesuai harapan) pada saat dikeluarkan nanti. Selain itu, terkait juga dengan kelebihan sebelumnya, yakni kualitas barang yang tetap terjamin, maka nantinya kekhawatiran bahwa harga barang akan anjlok pun dapat dihindari.

Sederhananya, harga pembelian nantinya tidak akan jadi lebih tinggi daripada harga penjualan. Jadi, tetap akan diperoleh keuntungan maksimal. Contohnya, di suatu gudang beras, terdapat beras pertama yang dibeli dengan harga beli Rp 7500/ kg. Lalu, masuk lagi barang kedua

yang dibeli dengan harga beli Rp 7900/ kg. Lalu, ada permintaan pasar dan saat itu harga barang sedang mengalami kenaikan. Maka, harga beras yang di pasaran mengalami kenaikan tidak akan terjadi pada beras. Dengan kata lain, perusahaan akan mampu mengendalikan harga pasar

3. Pencatatan yang lebih sistematis

Keuntungan sistem FIFO lainnya adalah pada saat pencatatan barang di gudang, yakni akan membuat petugas pencatatan barang masuk dan barang keluar jadi lebih mudah dalam melakukan kontrol. Hal dimungkinkan karena keluarnya barang dilakukan secara berurutan atau sesuai kronologis.

4. Jadi petugas pencatatan biasanya tidak perlu melakukan pengecekan terhadap semua barang. Petugas biasanya hanya perlu mengecek jumlah barang yang keluar pada saat itu, apakah jumlahnya sudah sesuai dengan jumlah barang pada saat barang tersebut masuk.

2.4 *Unified Modelling Language (UML)*

UML yang merupakan singkatan dari Unified Modelling Language adalah sekumpulan pemodelan konvensi yang digunakan untuk menentukan atau menggambarkan sebuah sistem perangkat lunak dalam kaitannya dengan objek. (Rosa dan Shalahudin, 2014).

UML dapat juga diartikan sebuah bahasa grafik standar yang digunakan untuk memodelkan perangkat lunak berbasis objek. UML pertama kali dikembangkan pada pertengahan tahun 1990an dengan kerjasama antara James Rumbaugh, Grady Booch dan Ivar Jacobson, yang masing-masing telah mengembangkan notasi mereka sendiri di awal tahun 1990an. (Rosa dan Shalahudin, 2014).

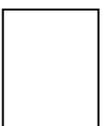
2.3.1 Komponen-komponen UML

UML mendefinisikan diagram-diagram berikut ini: (Rosa dan Shalahudin, 2014).

- a. 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.

Tabel 2.1 Simbol *Use Case*

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Actor</i>	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i> .
2		<i>Dependency</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>).
3		<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>).
4		<i>Include</i>	Menspesifikasikan bahwa <i>use case</i> sumber secara <i>eksplisit</i> .
5		<i>Extend</i>	Menspesifikasikan bahwa <i>use case</i> target memperluas perilaku dari <i>use case</i> sumber pada suatu titik yang diberikan.
6		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.
7		<i>System</i>	Menspesifikasikan paket yang menampilkan sistem secara terbatas.

8		<i>Use Case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor
9		<i>Collaboration</i>	Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemen-elemennya (sinergi).
10		<i>Note</i>	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi

b. Class Diagram

- 1) Class adalah sebuah spesifikasi yang jika diinstansiasi akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek. Class menggambarkan keadaan (atribut/properti) suatu sistem, sekaligus menawarkan layanan untuk memanipulasi keadaan tersebut (metoda/fungsi).
- 2) Class diagram menggambarkan struktur dan deskripsi class, package dan objek beserta hubungan satu sama lain seperti containment, pewarisan, asosiasi, dan lain-lain.

Tabel 2.2. Simbol Class Diagram

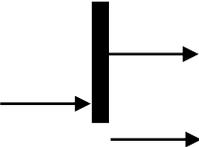
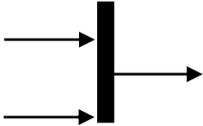
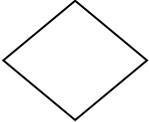
NO	GAMBAR	NAMA	KETERANGAN
1		<i>Generalization</i>	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).
2		<i>Nary Association</i>	Upaya untuk menghindari asosiasi dengan lebih dari 2 objek.

3		<i>Class</i>	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.
4		<i>Collaboration</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu actor
5		<i>Realization</i>	Operasi yang benar-benar dilakukan oleh suatu objek.
6		<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan mempegaruhi elemen yang bergantung padanya elemen yang tidak mandiri
7		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya

c. Activity Diagram

- 1) Activity diagram menggambarkan berbagai aliran aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing aliran berawal, keputusan yang mungkin terjadi, dan bagaimana mereka berakhir. Activity diagram juga dapat menggambarkan proses paralel yang mungkin terjadi pada beberapa eksekusi.
- 2) Activity diagram merupakan state diagram khusus, yang sebagian besar state adalah aksi dan sebagian besar transisi di-trigger oleh selesainya state sebelumnya (internal processing). Oleh karena itu, activity diagram tidak menggambarkan perilaku internal sebuah sistem dan interaksi antar subsistem, tetapi lebih menggambarkan proses-proses dan jalur-jalur aktivitas dari level atas secara umum.

Tabel 2.3 Simbol *Activity Diagram*

No.	Simbol	Keterangan
1	<i>Start State</i> 	<i>Start state</i> adalah sebuah kondisi awal sebuah <i>object</i> sebelum ada perubahan keadaan. <i>Start state</i> digambarkan dengan sebuah lingkaran solid.
2.	<i>End State</i> 	<i>End state</i> adalah menggambarkan ketika objek berhenti memberi respon terhadap sebuah event. <i>End state</i> digambarkan dengan lingkaran solid di dalam sebuah lingkaran kosong.
3.	<i>State/Activities</i> 	<i>State</i> atau <i>activities</i> menggambarkan kondisi sebuah entitas, dan digambarkan dengan segiempat yang pinggirnya.
4.	<i>Fork (Percabangan)</i> 	<i>Fork</i> atau percabangan merupakan pemisalah beberapa aliran konkuren dari suatu aliran tunggal.
5.	<i>Join (Penggabungan)</i> 	<i>Join</i> atau penggabungan merupakan penggabungan beberapa aliran konkuren dalam aliran tunggal.
6.	<i>Decision</i> 	<i>Decision</i> merupakan suatu logika aliran konkuren yang mempunyai dua cabang aliran konkuren.

--	--	--

d. Sequence Diagram

Sequence diagram secara grafis menggambarkan bagaimana objek berinteraksi antara satu sama lain melalui pesan pada sebuah use case atau operasi.

Tabel 2.4 Simbol Sequence Diagram

NO	GAMBAR	NAMA	KETERANGAN
1		<i>State</i>	Nilai atribut dan nilai link pada suatu waktu tertentu, yang dimiliki oleh suatu objek.
2		<i>Initial Pseudo State</i>	Bagaimana objek dibentuk atau diawali
3		<i>Final State</i>	Bagaimana objek dibentuk dan dihancurkan
4		<i>Transition</i>	Sebuah kejadian yang memicu sebuah state objek dengan cara memperbaharui satu atau lebih nilai atributnya
5		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.
6		<i>Node</i>	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi.

2.4 Basis Data

Database adalah kumpulan file-file yang mempunyai kaitan antara satu file dengan file yang lain sehingga membentuk satu bangunan data untuk menginformasikan satu perusahaan, instansi dalam batasan tertentu. (Pamungkas, 2017)

Istilah-istilah yang digunakan dalam basis data:

- 1) *File* : merupakan kumpulan dari atribut *record-record* sejenis yang mempunyai panjang elemen yang sama, atribut yang sama namun berbeda-beda dalam data *value*-nya.
- 2) *Record* : merupakan kumpulan dari elemen-elemen yang saling berhubungan atau berkaitan menginformasikan tentang *entry* secara lengkap.
- 3) *Field* : merupakan sekumpulan tanda-tanda yang berbentuk kesatuan tersendiri, merupakan bagian terkecil dari *record* dan bentuknya unik dijadikan *field* kunci yang dapat mewakili *record*-nya.
- 4) *Entity* : merupakan tempat kejadian atau konsep yang informasikan direkam.

2.5 HTML (Hyper Text Markup Language)

HTML merupakan kepanjangan dari *Hyper Text Markup Language* adalah suatu bahasa yang digunakan untuk membuat halaman-halaman hypertext (hypertext page) pada internet. Dengan konsep hypertext ini, untuk membaca suatu dokumen anda tidak harus melakukannya secara urut, baris demi baris, atau halaman demi halaman. Tetapi anda tidak dapat dengan mudah melompat dari satu topik ke topik lainnya yang anda sukai, seperti halnya jika anda melakukan pada online Help dari suatu aplikasi Windows. HTML dirancang untuk digunakan tanpa tergantung pada suatu platform tertentu (platform independent) (Enterprise, 2017).

2.6 PHP (Hypertext Preprocessor)

PHP diciptakan oleh Rasmus Lerdorf, seorang pemrogram C yang handal. Semula PHP hanya digunakan untuk mencatat jumlah pengunjung pada homepagenya. Rasmus adalah seorang

pendukung open source. Karena itulah ia mengeluarkan Personal Home Page Tools versi 1.0 secara gratis. Setelah mempelajari YACC dan GNU Bison, Rasmus menambah kemampuan PHP 1.0 dan menerbitkan PHP 2.0. PHP mudah dibuat dan cepat dijalankan, PHP dapat berjalan dalam web server yang berbeda dan dalam sistem operasi yang berbeda pula. PHP dapat berjalan di sistem operasi UNIX, Windows 98, Windows XP, Windows NT, dan Macintosh (Enterprise, 2017).

2.6.1 Sejarah

PHP pertama kali dibuat oleh Rasmus Lerdorf pada tahun 1995. Pada waktu itu PHP bernama FI (Form Interpreted). Pada saat tersebut PHP adalah sekumpulan script yang digunakan untuk mengolah data form dari web. Perkembangan selanjutnya adalah Rasmus melepaskan kode sumber tersebut dan menamakannya PHP/FI, pada saat tersebut kepanjangan dari PHP/FI adalah Personal Home Page/Form Interpreter. Dengan pelepasan kode sumber ini menjadi open source, maka banyak programmer yang tertarik untuk ikut mengembangkan PHP. Pada November 1997, dirilis PHP/FI 2.0. Pada rilis ini interpreter sudah diimplementasikan dalam C. Dalam rilis ini disertakan juga modul-modul ekstensi yang meningkatkan kemampuan PHP/FI secara signifikan. Pada tahun 1997, sebuah perusahaan bernama Zend, menulis ulang interpreter PHP menjadi lebih bersih, lebih baik dan lebih cepat. Kemudian pada Juni 1998 perusahaan tersebut merilis interpreter baru untuk PHP dan meresmikan nama rilis tersebut menjadi PHP 3.0. Pada pertengahan tahun 1999, Zend merilis interpreter PHP baru dan rilis tersebut dikenal dengan PHP 4.0. PHP 4.0 adalah versi PHP yang paling banyak dipakai. Versi ini banyak dipakai sebab versi ini mampu dipakai untuk membangun aplikasi web kompleks tetapi tetap memiliki kecepatan proses dan stabilitas yang tinggi. Pada Juni 2004, Zend merilis PHP 5.0. Versi ini adalah versi mutakhir dari PHP. Dalam versi ini, inti dari interpreter PHP mengalami perubahan besar. Dalam versi ini juga dikenalkan model pemrograman berorientasi objek baru untuk menjawab perkembangan bahasa pemrograman ke arah pemrograman berorientasi objek.

2.6.3 Kelebihan

1. PHP adalah sebuah bahasa script yang tidak melakukan sebuah kompilasi dalam penggunaannya.
2. Web Server yang mendukung PHP dapat ditemukan dimana - mana dari mulai IIS sampai

dengan apache, dengan konfigurasi yang relatif mudah.

3. Dalam sisi pengembangan lebih mudah, karena banyaknya milis - milis dan developer yang siap membantu dalam pengembangan.
4. Dalam sisi pemahaman, PHP adalah bahasa scripting yang paling mudah karena referensi yang banyak. PHP adalah bahasa open source yang dapat digunakan di berbagai mesin (linux, unix, windows) dan dapat dijalankan secara runtime melalui console serta juga dapat menjalankan perintah-perintah system.

2.7 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 instalasi di seluruh dunia. MySQL AB membuat MySQL tersedia sebagai perangkat lunak gratis dibawah lisensi GNU General Public License (GPL), tetapi mereka juga menjual dibawah lisensi komersial untuk kasus-kasus dimana penggunaannya tidak cocok dengan penggunaan GPL. MySQL adalah Relational Database Management System (RDBMS) yang didistribusikan secara gratis dibawah lisensi GPL (General Public License). Dimana setiap orang bebas untuk menggunakan MySQL, namun tidak boleh dijadikan produk turunan yang bersifat closed source atau komersial. MySQL sebenarnya merupakan turunan salah satu konsep utama dalam database sejak lama, yaitu SQL (Structured Query Language). SQL adalah sebuah konsep pengoperasian database, terutama untuk pemilihan atau seleksi dan pemasukan data, yang memungkinkan pengoperasian data dikerjakan dengan mudah secara otomatis (Pamungkas, 2017).

2.8 Metode Pengembangan Perangkat Lunak Menggunakan Metode *Prototype*

Metode penelitian yang digunakan penulis didasarkan pada metoda pengembangan perangkat lunak prototype yang memiliki lima tahap (Pressman, 2015). Alasan penulis mengambil metode pengembangan sistem prototype karena pengguna dapat dengan mudah beradaptasi dengan sistem yang dikembangkan sesuai dengan kebutuhan pemakai. Selain itu pada saat tahap evaluasi sistem, jika terdapat revisi pada sistem yang dirancang, programmer tidak harus mengulang tahapan dari awal.



Gambar 2.1 Prototyping Model
(Sumber : Pressman, 2015)

Berikut penjelasan tahapan metoda prototype yang digunakan dalam penelitian ini.

1. *Communication.*

Penulis menganalisis sistem dengan mewawancarai pihak-pihak pengguna sebagai pemakai untuk mendapatkan gagasan dari apa yang diinginkan pemakai terhadap sistem yang dibuat.

2. *Quick Plan*

Analisis sistem bekerja sama dengan spesialis informasi lain pihak, menggunakan satu atau lebih peralatan prototyping untuk mengembangkan sebuah prototype. (misalnya dengan menambahkan tampilan yang diinginkan pemakai dalam sistem baru seperti menu, interface, fitur, dan database).

3. *Modelling Quick Design*

Analisis membuat perancangan sistem untuk mengembangkan prototipe dengan menggunakan UML, sebagai dasar perancangan aplikasi. Dalam hal ini programmer menggunakan PHP sebagai bahasa pemrograman dan MySQL sebagai database.

4. *Construction Of Prototype*

Tahapan ini adalah tahapan yang dilakukan setelah kegiatan analisi dan perancangan. Pada bagian ini akan dijelaskan kegiatan-kegiatan yang dilakukan pada tahap pengkodean (coding) sistem operasional, implementasi pembuatan program (programming) dan pengujian (testing).

5. *Deployment delivery & feedback*

Penulis yang menguji sistem baru tersebut dan melakukan uji coba terhadap beberapa calon pengguna sehingga penulis dapat menentukan apakah sistem baru dapat diterima. Pada tahap ini pemakai memberi masukan kepada analis apakah sistem dapat diterima. Jika ya sistem baru yang telah diuji dan terima oleh pengguna aplikasi, jika tidak, langkah 4 dan 5 diulangi.

2.8 Penelitian Terkait

Berikut ini penelitian terkait dengan penelitian ini :

1. Oleh Uus Rusmawan tahun 2018 dari Teknik Informatika; STMIK Bina Insani. Dengan judul “Sistem Informasi Persediaan Barang Menggunakan Metode First In First Out”. Penelitian ini membahas penggunaan metode FIFO untuk menyesuaikan biaya pembuatan produk dengan harga pokok penjualan dan margin keuntungan yang ingin dicapai oleh sebuah perusahaan.

Penelitian ini membuat aplikasi Sistem Informasi Persediaan Barang Menggunakan Metode First In First Out berbasis desktop.

Sumber : BINA INSANI ICT JOURNAL, Vol.5, No. 1, Juni 2018, 71 – 80 ISSN: 2355-3421 (Print) ISSN: 2527-9777 (Online).

2. Oleh Julianto Simatupang tahun 2017 dari Prodi Manajemen Informatika – AMIK Mahaputra Riau. Dengan judul “Perancangan Sistem Informasi Inventori Barang Pada Toko Nicholas Jaya Menggunakan Metode FIFO”. Penelitian ini membahas pembuatan laporan

persediaan barang admin harus membuka lembar perlembar buku persediaan kemudian memindahkan ke dalam dokumen laporan sehingga membutuhkan waktu yang sangat lama. Penelitian ini membuat aplikasi Sistem Informasi Inventori Barang Menggunakan Metode First In First Out berbasis desktop.

Sumber : Jurnal Intra-Tech, Volume 1, No.1 April 2017 ISSN. 2549-0222.

3. Oleh Desnita Nur Fazli dan Yuwan Jumaryadi tahun 2019 dari Fakultas Ilmu Komputer, Universitas Mercu Buana. Dengan judul “Perancangan Sistem Informasi Inventori Barang Menggunakan Metode FIFO (First In First Out) Pada CV Jaya Mas Elektronik”. Penelitian ini membahas proses pencatatan persediaan barang pada toko ini masih menggunakan cara konvensional, dimana setiap transaksi masih dilakukan dengan mencatat ke dalam nota. Maka perlu adanya sebuah sistem terkomputerisasi dan terintegrasi untuk pengelolaan barang sehingga persediaan barang dapat dikendalikan dengan baik serta informasi ketersediaan barang dapat diketahui secara real time.

Penelitian ini tidak membuat aplikasi, hanya sampai pada perancangan aplikasi menggunakan UML

Sumber : Ensiklopedia of Journal, Vol. 1 No.2 Edisi 1 Januari 2019 P-ISSN 2622-9110 E-ISSN 2654-8399.