

BAB 2

TINJAUAN PUSTAKA

2.1 Profil Perusahaan

Male Store merupakan salah satu toko baju yang bergerak dalam bidang jasa khususnya *fashion*, *Male Store* menjual beberapa macam katalog pakaian jadi pria seperti baju, kemeja, celana, dan jaket yang dipesan melalui *supplier*.

2.1.1 Sejarah Perusahaan

Toko baju yang telah berdiri sejak 2015 ini beralamat di Jl. Ahmad Yani Km. 5.5 Komp. Banjar Indah Raya No.7G Banjarmasin, Kalimantan Selatan. Toko Baju ini awalnya didirikan oleh Faridz Zulkarnaen dan Muhammad Mirza, tidak lama kemudian bergabung lagi dua orang yaitu M. Aldy Syahputra dan Muhammad Reffier Yudio.

2.1.2 Logo Perusahaan

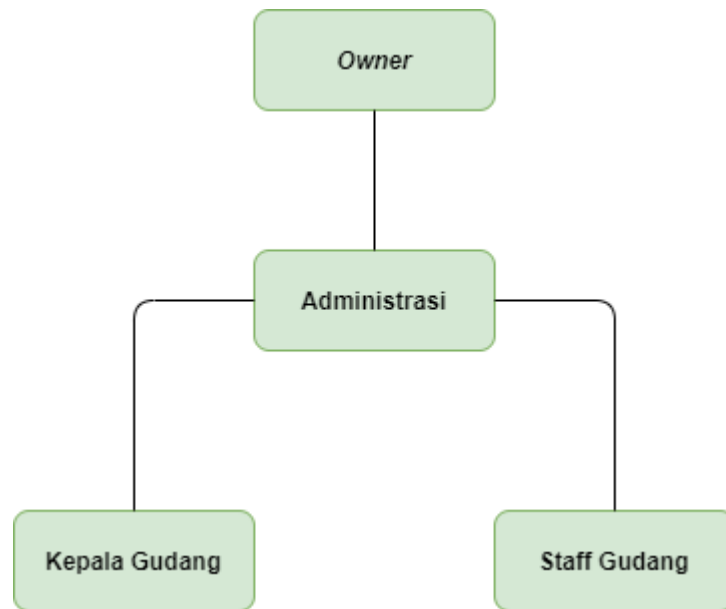
Logo dari *Male Store* dapat dilihat pada gambar 2.1



Gambar 2. 1 Logo Perusahaan

2.1.3 Struktur Organisasi

Untuk menjelaskan struktur organisasi yang menjadi sasaran kegiatan penelitian, maka struktur organisasi *Male Store* dapat dilihat pada gambar 2.2.



Gambar 2.2 Struktur Organisasi

2.1.4 Job Description

Berikut ini adalah job description dari tiap-tiap bagian :

A. *Owner*

Tugas dari *Owner* dalam *Male Store* adalah sebagai berikut :

1. Membuat peraturan-peraturan yang ada di toko
2. Bertanggung jawab dalam memimpin dan menjalankan perusahaan

B. Adiministrasi

Tugas dari bagian administrasi dalam *Male Store* adalah sebagai berikut:

Mengelola aplikasi yang digunakan, seperti mengurus akun-akun dari tiap jabatan.

C. Kepala Gudang

Tugas dari kepala gudang dalam perusahaan adalah sebagai berikut :

1. Melakukan pengadaan barang kepada *supplier*.
2. Menerima laporan barang masuk berdasarkan pencatatan staff gudang.

D. Staff Gudang

Tugas dari staff gudang dalam perusahaan adalah sebagai berikut :

1. Mencatat setiap barang masuk yang datang dari supplier.
2. Memberikan catatan laporan barang masuk ke kepala gudang.

2.2 Landasan Teori

Landasan teori yang akan dibahas dalam menyusun laporan tugas akhir dan pembangunan Sistem Informasi Manajemen Pengadaan Barang di *Male Store*. Teori yang akan dibahas dalam landasan ini adalah sebagai berikut :

2.2.1 Sistem Informasi

Pengertian Sistem Informasi adalah sebagai berikut : Sistem Informasi adalah suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung operasi, bersifat manajerial dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang diperlukan [1].

Dalam menyediakan informasi kerap kali terjadi ketidaksesuaian antara kebutuhan informasi dan informasi yang disediakan, maka dari itu sistem informasi memiliki keterkaitan dengan kebutuhan informasi. Menurut Bambang Hartono dalam buku Sistem Informasi Manajemen Berbasis Komputer tahun 2003, menyatakan bahwa kebutuhan informasi adalah pernyataan yang dikemukakan oleh seorang (misalnya seorang manager) tentang informasi yang diperlukan dan akan digunakannya, agar ia dapat melakukan suatu (misal pengambilan keputusan) dengan baik. Ketidaksesuaian penyampaian informasi ini terjadi akibat ketidaksesuaian tafsir (konsensus) tentang kebutuhan informasi antara pemakai informasi dan pengelola/penyedia informasi.

Sistem Informasi (SI) adalah kombinasi dari teknologi informasi dan aktivitas orang yang menggunakan teknologi itu untuk mendukung operasi dan manajemen. Dalam arti yang sangat luas, istilah sistem informasi yang sering digunakan merujuk kepada interaksi antara orang, proses algoritmik, data, dan teknologi. Dalam pengertian ini, istilah ini digunakan untuk merujuk tidak hanya pada penggunaan organisasi teknologi informasi dan komunikasi (TIK), tetapi juga untuk cara di mana orang berinteraksi dengan teknologi ini dalam mendukung proses bisnis.

2.2.2 Sistem Informasi Manajemen

Sistem informasi manajemen adalah sistem informasi yang digunakan untuk menyajikan informasi yang digunakan untuk mendukung operasi, manajemen, dan pengambilan keputusan dalam sebuah organisasi. Umumnya sistem informasi manajemen mengambil data dari sistem pemrosesan transaksi. Sistem Informasi Manajemen seringkali disebut juga sebagai system peringatan manajemen (*management alerting system*) karena sistem ini memberikan peringatan kepada pemakai (umumnya manajemen) terhadap masalah maupun peluang, istilah yang lain dari SIM adalah sistem pelaporan manajemen atau *management reporting system*. Sistem informasi manajemen adalah sebuah sistem informasi yang selain melakukan semua pengolahan transaksi yang perlu untuk sebuah organisasi juga memberi dukungan informasi dan pengolahan untuk fungsi manajemen dan pengambilan keputusannya [2].

2.2.3 Pengadaan

Pengadaan (*procurement*) adalah proses bisnis memilih sumber, pemesanan, dan memperoleh barang/jasa. Barang/jasa tersebut bisa diperoleh secara internal bila barang dihasilkan oleh entitas lain dalam perusahaan. Pembelian adalah sinonim untuk *procurement* [18]. Dari pengertian tersebut, pengadaan merupakan suatu proses yang digunakan perusahaan dalam memperoleh kebutuhan akan barang/jasa dari luar perusahaan. langkah-langkah umum dalam proses pengadaan adalah [18]:

- a. Penentuan persyaratan

Pada tahap ini perusahaan membuat permohonan pembelian dengan menyertakan persyaratan spesifikasi barang/jasa yang dibutuhkan.

b. Pemilihan sumber

Langkah berikutnya adalah menentukan sumber penyedia atau pemasok.

c. Permintaan penawaran

Permintaan untuk penawaran dibuat untuk item atau jasa yang sangat mahal atau jasa yang penawarannya diperlukan sebagai kebijakan perusahaan.

d. Pemilihan pemasok

Berdasarkan dokumen penawaran, maka perusahaan dengan menggunakan seorang ahli yang mengerti mengenai barang atau jasa yang diminta bisa mengevaluasi dan memberikan penilaian terhadap pemasok.

e. Membuat pesanan pembelian

Dokumen pesanan pembelian mengidentifikasi pemasok dan mengkonfirmasi barang yang dipesan, jumlah, harga, tanggal pengiriman, jangka waktu pengiriman, dan jangka waktu pembayaran.

f. Penerimaan barang

Penerimaan barang dilakukan oleh fungsi penerimaan.

g. Verifikasi faktur

Faktur yang diterima harus diperiksa dan dicocokkan dengan dokumen penerimaan barang dan pesanan pembelian.

h. Pembayaran kepada pemasok

Jika barang yang diterima sudah sesuai dengan pesanan, maka proses selanjutnya adalah pembayaran kepada pemasok.

2.2.4 Economic Order Quantity (EOQ)

Model EOQ adalah salah satu model perhitungan untuk mendapatkan nilai kuantitas pesanan optimal suatu perusahaan. Dengan asumsi nilai permintaan untuk suatu produk (rate of demand), biaya pemesanan (*ordering cost*), harga pembelian per unit (*purchasing unit price*) adalah bernilai konstan [4].

Metode EOQ berusaha mencapai tingkat persediaan semimumimum mungkin, biaya rendah dan mutu yang lebih baik. Perencanaan persediaan yang menggunakan metode EOQ dalam suatu perusahaan akan mampu meminimalisasi terjadinya out of stock sehingga tidak mengganggu proses produksi dalam perusahaan dan mampu menghemat biaya persediaan bahan baku dalam perusahaan. Dengan adanya penerapan metode EOQ pada perusahaan diharapkan akan mampu mengurangi biaya penyimpanan, penghematan ruang, baik gudang maupun ruang kerja, menyelesaikan masalah-masalah yang timbul dari banyaknya persediaan yang menumpuk sehingga mengurangi resiko yang dapat ditimbulkan karena persediaan yang berlebihan didalam ruang penyimpanan atau gudang.

Apabila EOQ model menjawab pertanyaan berapa banyak pemesanan yang optimal, maka reorder point (ROP) menjawab pertanyaan kapan mulai mengadakan pesanan. ROP terjadi apabila jumlah persediaan yang terdapat di dalam stok berkurang terus dalam artian proses produksi terus berjalan, dengan demikian kita harus menentukan berapa banyak batas minimal tingkat persediaan yang harus dipertimbangkan sehingga tidak terjadi kekurangan persediaan. Persamaan untuk ROP mengasumsikan bahwa permintaan selama lead time dan lead time itu sendiri konstan. Dan bila tidak seperti itu maka diperlukan persediaan tambahan yang disebut persediaan pengaman (*e stock*).

2.2.5 Kebijakan-Kebijakan EOQ

Bahan baku yang tersedia dalam menjamin kelancaran proses produksi dan biaya yang dikeluarkan oleh perusahaan sehubungan dengan perusahaan tersebut semimal mungkin, maka tindakan yang perlu dilakukan adalah menentukan *Economic Order Quantity* (EOQ), *Safety Stock*, *Reorder Point* (ROP). Syarat data yang menggunakan metode EOQ (*economic Order Quantity*) sebagai berikut :

1. Tingkat permintaan diketahui.
2. *Lead time* diketahui.
3. Barang yang dipesan diasumsikan dapat segera tersedia.

4. Tidak ada pesanan ulang (*back order*) karena kehabisan persediaan (*Storage*).
5. Setiap pemesanan diterima dalam sekali pengiriman dan langsung dapat digunakan.
6. Harga pembelian tidak berubah.
7. Tidak ada potongan harga (*quality discount*).

Variable biaya hanya biaya pesan (*ordering cost*) dan biaya simpan (*Holding cost*).

2.2.7 Perhitungan EOQ

Beberapa rumusan dalam perhitungan *Economic Order Quantity* (EOQ).

1. Rumus untuk mengetahui EOQ

EOQ banyak dipergunakan sampai saat ini karena mudah dalam penggunaannya, meskipun dalam penerapannya harus memperhatikan asumsi yang dipakai. Asumsi dasar untuk menggunakan metode EOQ adalah sebagai berikut :

- a) Permintaan dapat ditentukan secara pasti.
- b) Item yang dipesan independent dengan item yang lain.
- c) Harga item yang konstan.
- d) Pemesanan diterima dengan segera dan pasti.

Adapun rumus yang biasa digunakan untuk menghitung EOQ adalah sebagai berikut :

$$EOQ = \sqrt{\frac{2 \times R \times S}{P}} \dots\dots\dots (2.1)$$

Dimana :

EOQ = Total biaya pemesanan Optimal ekonomis

R = Jumlah Pesanan Selama *Lead Time*

S = Biaya setiap melakukan pemesanan

P = Harga barang per unit

$$EOQ = \sqrt{\frac{2 \times D \times S}{C}} \dots\dots\dots (2.2)$$

Dimana :

EOQ = Total biaya pemesanan optimal ekonomis

D = jumlah permintaan 1 tahun atau 1 periode

S = biaya setiap melakukan pemesanan

C = biaya penyimpanan

2. Rumus untuk mengetahui biaya pemesanan

Adapun rumus yang biasa digunakan untuk menghitung Biaya Pemesanan yaitu :

$$(TOC) = (D / EOQ) S \dots\dots\dots (2.3)$$

Dimana :

TOC = Biaya pemesanan

D = jumlah permintaan 1 tahun atau 1 periode

S = biaya setiap melakukan pemesanan

3. Rumus untuk mengetahui Frekuensi Pemesanan

Adapun rumus yang biasa digunakan untuk menghitung Frekuensi Pemesanan yaitu :

$$(F) = D / EOQ \dots\dots\dots (2.4)$$

Dimana :

F = frekuensi pemesanan

D = jumlah permintaan 1 tahun atau 1 periode

EOQ = Total biaya pemesanan Optimal ekonomis

4. Rumus untuk mengetahui jumlah permintaan per hari

Adapun rumus yang biasa digunakan untuk menghitung biaya pemesanan yaitu:

$$d = D / \text{Jumlah hari kerja} \dots\dots\dots (2.5)$$

Dimana :

d = Jumlah permintaan perhari

D = jumlah permintaan 1 tahun atau 1 periode

5. Rumus untuk mengetahui jumlah pesanan selama lead time

Adapun rumus yang biasa digunakan untuk menghitung jumlah pesanan selama lead time yaitu:

$$R = d \times L \dots\dots\dots (2.6)$$

Dimana :

R = Jumlah Pesanan Selama Lead Time

d = Jumlah permintaan perhari

L = Lead Time

6. Rumus untuk mengetahui Safety Stock

Persediaan pengamanan adalah persediaan tambahan yang diadakan untuk menjaga kemungkinan kekurangan barang (*stock out*).

Adapun rumus yang biasa digunakan untuk menghitung *Safety Stock* yaitu :

$$S_s = (\text{Pemakaian Max} - \text{Pemakaian Rata-Rata}) \text{ Lead Time} \dots\dots\dots (2.7)$$

$$S_s = Z \times \sqrt{\left(\frac{PC}{T}\right) \times \sigma D} \dots\dots\dots (2.8)$$

Dimana :

Z = Safety Factor

PC = Performance

S_s = Safety Stock

σD = Standara Deviasi dari demand

T = Siklus Periode

$$S_s = Z \times \sqrt{(PC/T \times \sigma D)^2 + (\sigma_{LTLT} \times D \text{ rata2})^2} \dots\dots\dots (2.9)$$

Dimana :

S_s = Safety Stock

Z = Safety Factor

PC = Performance cycle

σD = Standar Deviasi dari demand

σ_{LTLT} = Standar deviasi lead

D rata2 = demand/kebutuhan rata2

2.2.8 Pemrograman Berbasisi Objek

Pemrograman Berorientasi Objek (*Object Oriented Programming* atau disingkat OOP) adalah konsep pemrograman yang difokuskan pada penciptaan kelas yang merupakan abstraksi/*blueprint/prototype* dari suatu objek. Sedangkan sistem berorientasi objek merupakan sebuah sistem yang dibangun berdasarkan metode berorientasi objek. Pemrograman berorientasi objek (PBO) dapat menggunakan bahasa pemrograman berorientasi objek seperti Java, C#, C++, Smalltalk, Eiffel dan PHP [19].

1. Kelas (Class)

Kelas merupakan penggambaran dari sebuah objek atau benda, sifat objek, dan juga apa yang bisa dilakukan oleh objek tersebut. 22

2. Objek (Obyek)

Objek merupakan *instans* (perwujudan) dari suatu kelas. Objek merupakan suatu *entitas* yang mampu menyimpan informasi dan mempunyai operasi yang dapat diterapkan atau dapat berpengaruh pada status objeknya.

3. Metode (Method)

Metode adalah prosedur atau fungsi yang dimiliki oleh sebuah objek. Sebuah kelas boleh memiliki lebih dari satu metode yang berfungsi untuk memanipulasi objek itu sendiri. Metode juga merupakan cara objek untuk berkomunikasi antar objek.

4. Atribut (Attribute)

Atribut adalah sifat karakteristik atau kondisi yang dimiliki oleh suatu objek. Atribut dapat berupa nilai atau elemen – elemen data yang dimiliki oleh objek dalam kelas objek.

5. Enkapsulasi (Encapsulation)

Enkapsulasi ialah lapisan pelindung yang mencegah kode dan data yang secara acak diakses oleh kode lain yang didefinisikan di luar kelas.

6. Pewarisan (Inheritance)

Mekanisme yang memungkinkan satu objek mewarisi sebagian atau seluruh definisi dan objek lain sebagai bagian dari dirinya. Konsep pemrograman dimana sebuah class dapat menurunkan *property* dan metode yang dimilikinya kepada class lain.

7. Antarmuka (Interface)

Antarmuka atau *interface* sangat mirip dengan kelas, tapi tanpa atribut kelas dan memiliki metode yang dideklarasikan tanpa isi. Deklarasi metode pada sebuah *interface* dapat diimplementasikan oleh kelas lain.

8. Generalisasi dan Spesialisasi

Generalisasi dan Spesialisasi menunjukkan hubungan antara kelas dan objek yang umum dengan kelas dan objek yang khusus.

9. Polimorfisme (Polymorfisme)

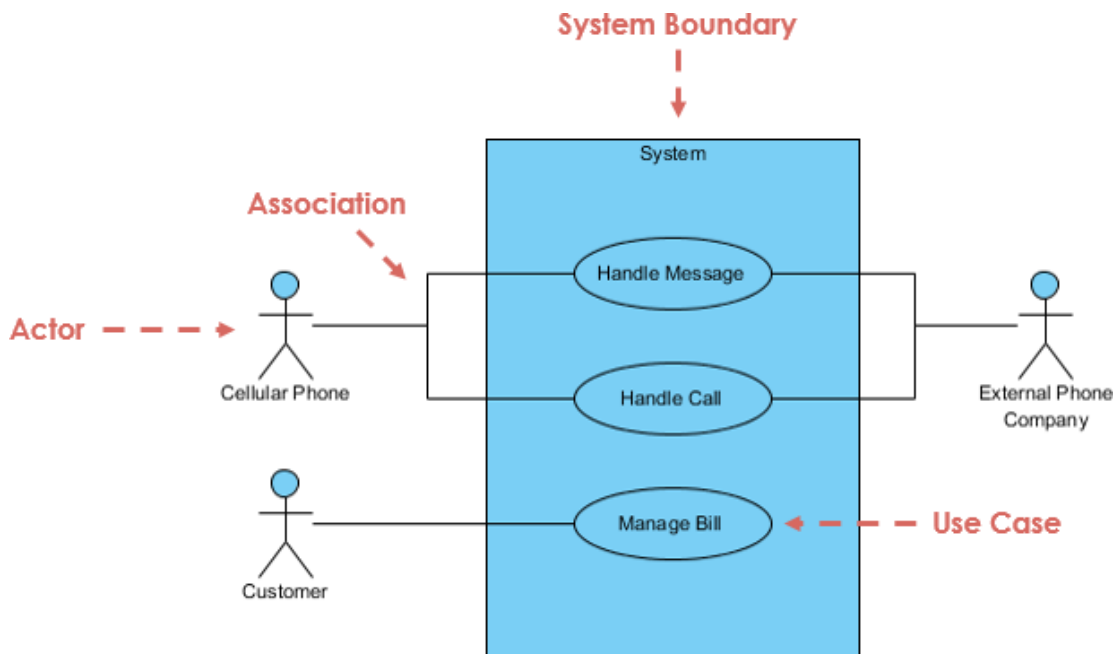
Polimorfisme yaitu suatu objek bisa bertindak lain terhadap *message/method* yang sama.

2.2.9 Unified Modelling Language (UML)

Unified Modeling Language (UML) merupakan bahasa penulisan spesifikasi standar cetak biru perangkat lunak. UML digunakan untuk menspesifikasikan, mendokumentasikan, dan membangun sistem perangkat lunak. Pengembangan UML dimulai dari kerja sama Grady Booch dan James Rumbaugh pada 1994 untuk menggabungkan dua metodologi terkenal yaitu Booch dan OMT. Kemudian Ivan Jacobson, pencipta metode OOSE (*Object Oriented Software Engineering*) bergabung. Usulan UML diberikan kepada OMG (*Object Management Group*) sebagai konsorsium standarisasi teknologi objek agar UML dijadikan bahasa dan notasi pemodelan dilakukan pada tahun 1997. OMG menerima UML, UML telah menjadi standar de-facto karena pencipta-penciptanya sangat populer. Banyak pengembang perangkat lunak yang mengadopsi UML [20].

2.2.9.1 Use Case Diagram

Use Case Diagram menjabarkan manfaat sistem jika dilihat menurut pandangan orang yang berada diluar sistem (Aktor). Diagram ini menunjukkan fungsionalitas suatu sistem yang berinteraksi dengan dunia luar. *Use Case Diagram* dapat digunakan selama proses analisis untuk menangkap requirement sistem dan untuk memahami bagaimana sistem bekerja.

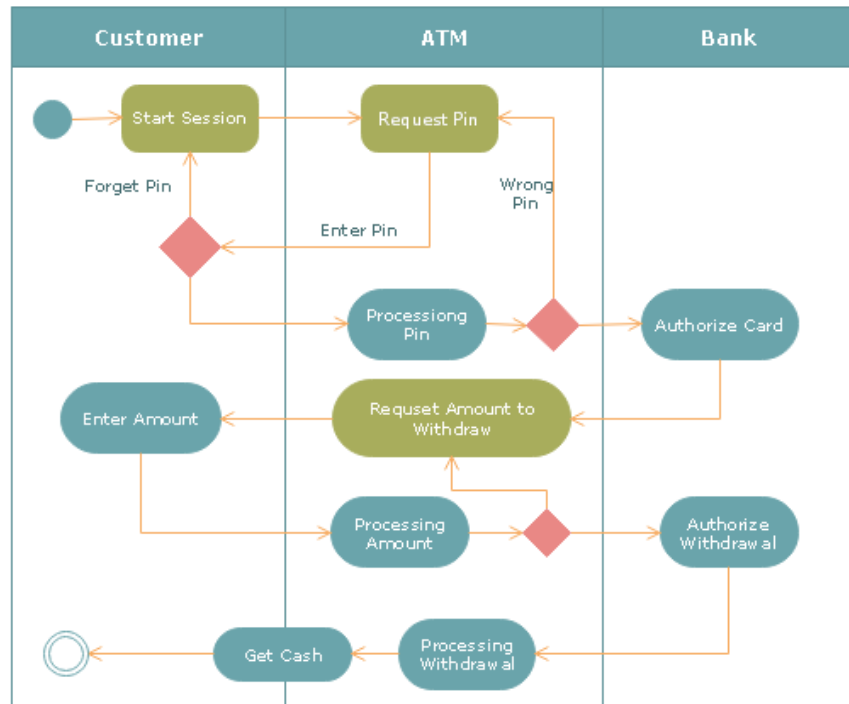


Gambar 2. 2 Contoh Use Case Diagram

2.2.9.2 Activity Diagram

Activity diagram UML menggambarkan perilaku dinamis dari suatu sistem atau suatu komponen sistem melalui aliran control diantara aksi-aksi yang dilakukan system. Ini mirip dengan diagram alir kecuali bahwa diagram aktivitas dapat menunjukkan aliran-aliran konkuren. Komponen utama suatu diagram aktivitas adalah node aksi (*action node*), direpresentasikan dengan suatu persegi panjang dengan sudut bulat, yang berhubungan dengan suatu tugas yang dilakukan oleh sistem perangkat lunak.

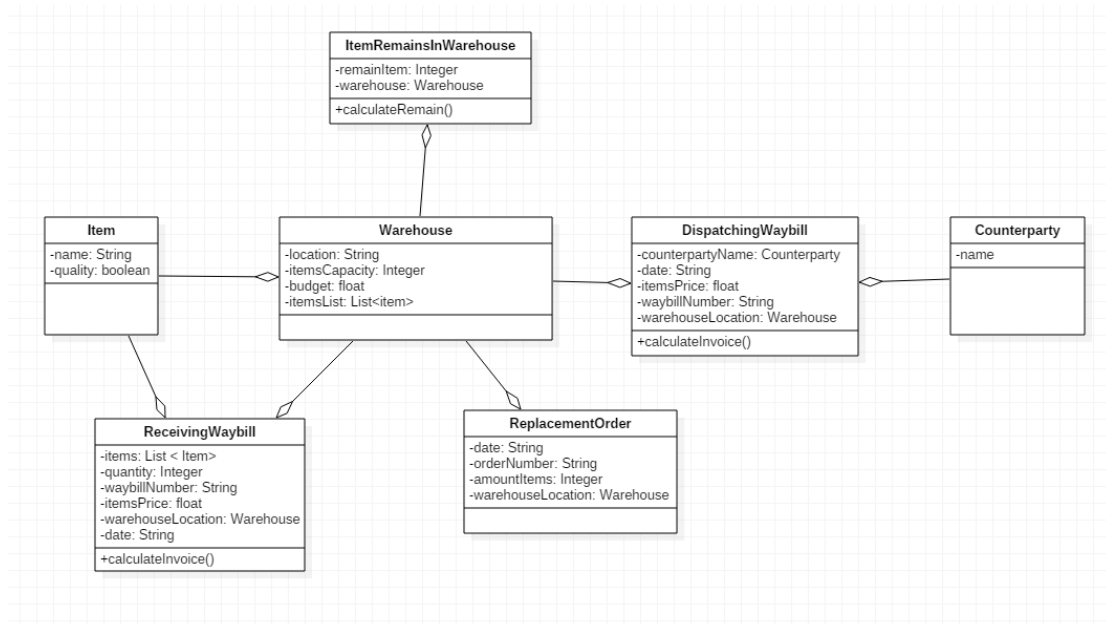
ATM Withdrawal Activity Diagram



Gambar 2. 3 Contoh Activity Diagram

2.2.9.3 Class Diagram

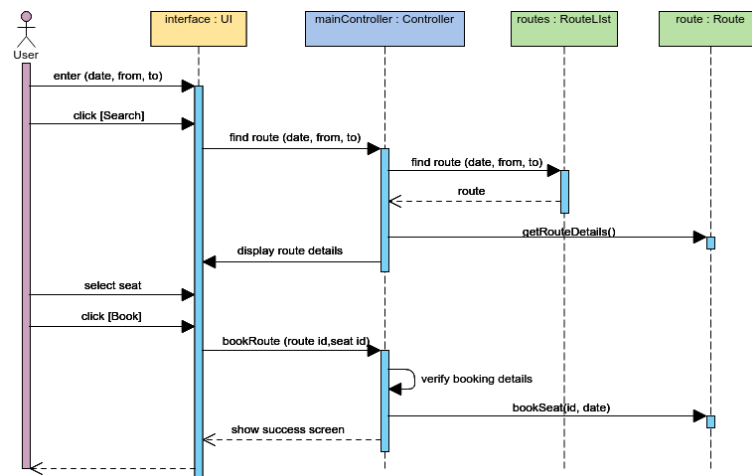
Class Diagram menjelaskan dalam visualisasi struktur kelas-kelas dari suatu sistem dan merupakan tipe diagram yang paling banyak dipakai. *Class* Diagram memperlihatkan hubungan antar kelas dan penjelasan detail tiap-tiap kelas dalam model desain dari suatu sistem. Selama proses analisis, *class* diagram memperlihatkan aturan-aturan dan tanggung jawab *entitas* yang menentukan 25 perilaku sistem. Selama tahap desain, *class* diagram berperan dalam menangkap struktur dari semua kelas yang membentuk arsitektur sistem yang dibuat.



Gambar 2. 4 Contoh class diagram

2.2.9.4 Sequence Diagram

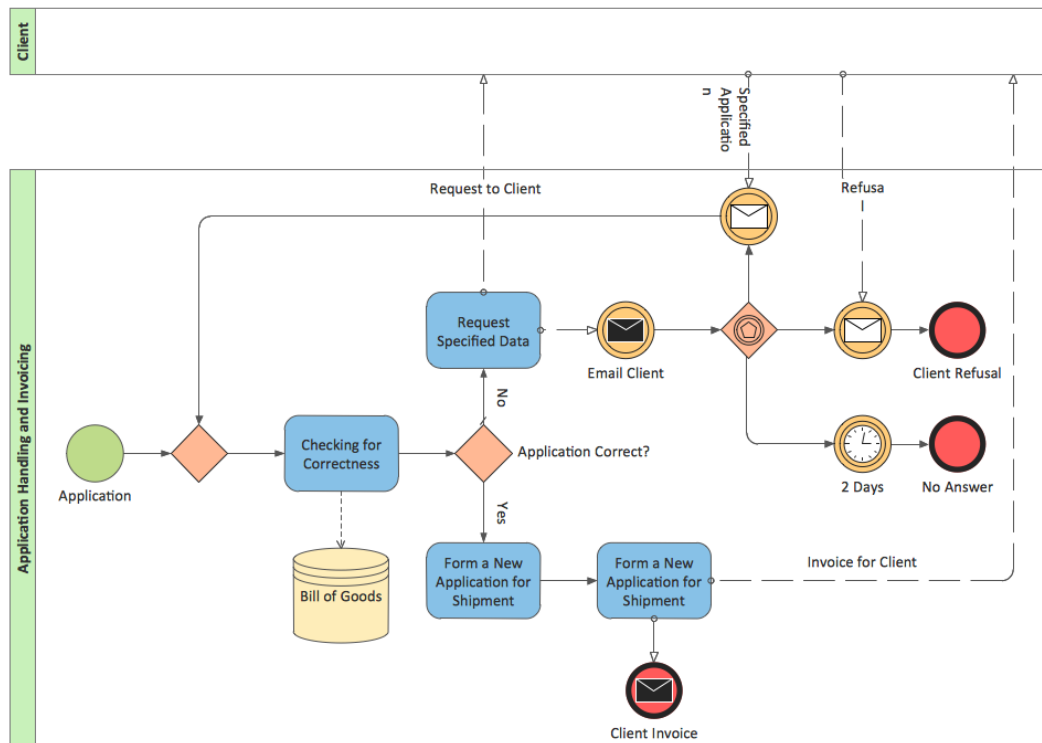
Sequence diagram menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan pesan yang diterima antar objek. Oleh karena itu untuk menggambarkan diagram sekuen maka harus diketahui objek-objek yang terlibat dalam use case beserta metode-metode yang dimiliki kelas yang diinstansiasi menjadi objek itu.



Gambar 2. 5 Contoh sequence diagram

2.2.9.5 Business Process Modelling and Notation

Business Process Modelling Notation (BPMN) adalah suatu metode penyelarasan secara efisien suatu organisasi dengan keinginan dan kebutuhan organisasi. BPMN merupakan suatu pendekatan manajemen holistik untuk meningkatkan efektivitas dan efisiensi bisnis seiring upaya untuk mencapai inovasi, fleksibilitas dan integrasi dengan teknologi [21].



Gambar 2. 6 Contoh BPNM

2.2.9.6 Basis Data

Basis Data adalah kumpulan informasi yang disimpan di dalam komputer secara sistematis sehingga dapat diperiksa menggunakan suatu program komputer untuk memperoleh informasi dari basis data tersebut. Perangkat lunak yang digunakan

untuk mengelola dan memanggil kueri (query) basis data disebut sistem manajemen basis data (*database management system*, DBMS) [7].

Pengolahan basis data secara fisik tidak dilakukan oleh pemakai secara langsung, tetapi ditangani oleh sebuah perangkat lunak atau sistem yang khusus. *DataBase Management System (DBMS)* yang akan menentukan bagaimana data diorganisasi, disimpan, diubah dan diambil kembali. Ia juga menerapkan mekanisme pengamanan data, pemakaian data secara bersama, pemaksaan keakuratan/konsistensi data dan sebagainya.

Maka dari itu bisa disimpulkan bahwa *database* merupakan kumpulan informasi yang disimpan di dalam komputer secara sistematis sehingga dapat diperiksa menggunakan suatu program komputer untuk memperoleh informasi dari basis data tersebut. Database berfungsi untuk menampung atau menyimpan data – data, dimana masing – masing data yang ada pada tabel atau *file* tersebut saling berhubungan dengan satu sama lainnya. Basis data telah digunakan pada hampir seluruh area dimana komputer digunakan, termasuk bisnis, teknik, kesehatan, hukum, pendidikan dan sebagainya. Tujuan basis data pada suatu perusahaan pada dasarnya adalah kemudahan dan kecepatan dalam pengambilan kembali data.

Database mempunyai 6 operasi dasar diantaranya adalah *Create database*, *Drop database*, *create table*, *Drop table*, *Insert*, *Read*, *Update* dan *Delete*, adapun penjelasannya sebagai berikut :

1. Pembuatan basis data baru (*create database*), yang identik dengan pembuatan lemari arsip yang baru.
2. Penghapus basis data (*drop database*), yang identik dengan perusakan arsip yang baru.
3. Pembuatan tabel baru kesuatu basis data (*create table*) dengan perusakan maaf arsip lama yang ada disebuah lemari arsip.
4. Pengambilan data dari sebuah tabel (*query*) dengan pencarian lembaran arsip dari sebuah map arsip.

5. Pengubahan data dari sebuah tabel (*update*) dengan perbaikan isi lembaran arsip dari sebuah map arsip.
6. Penghapusan data dari sebuah tabel (*delete*) dengan penghapusan sebuah lembaran arsip yang ada di sebuah map arsip

2.2.9.7 Sistem Basis Data

Gabungan antara basis data dan perangkat lunak SMBD (Sistem Manajemen Basis Data) termasuk di dalamnya program aplikasi yang dibuat dan bekerja dalam satu sistem disebut dengan Sistem Basis Data. Sistem basis data dapat dianggap sebagai tempat untuk sekumpulan berkas data yang terkomputerisasi dengan tujuan untuk memelihara informasi dan membuat informasi tersebut tersedia saat dibutuhkan [9].

Merancang *database* merupakan suatu hal yang sangat penting. Perancangan model konseptual perlu dilakukan disamping perancangan model fisik. Unsur-unsur konsep pembangun *database* sebagai berikut :

1. Atribut (*Field*)

Atribut atau *Field* adalah identitas yang mewakili satu jenis data.

2. *File*

File adalah kumpulan *record-record* sejenis yang mempunyai panjang elemen yang sama, atribut yang sama namun berbeda data valuenya.

3. *Record*

Record adalah kumpulan elemen yang saling terkait yang menginformasikan tentang suatu *entity* secara lengkap. Suatu *record* mewakili satu data atau Informasi tentang seseorang.

4. Tabel

Tabel adalah sebuah file yang menampung data-data dalam kelompok tertentu.

2.2.9.8 DBMS

DBMS adalah sebuah piranti lunak yang dapat memudahkan sebuah organisasi yang memusatkan data, mengelola data secara efisien dan efektif, dan menyediakan akses data untuk aplikasi sistem informasi organisasi atau perusahaan. Sebagai contoh adalah Oracle database, DB2, Microsoft SQL Server, merupakan DBMS untuk komputer server & mainframe [10].

DBMS ialah perantara untuk user dengan basis data, untuk dapat berinteraksi dengan DBMS dapat memakai bahasa basis data yang sudah ditentukan oleh perusahaan DBMS. Bahasa basis data umumnya terdiri dari berbagai macam instruksi yang diformulasikan sehingga instruksi tersebut dapat di proses oleh DBMS. Perintah atau instruksi tersebut umumnya ditentukan oleh user, adapun bahasa yang digunakan dibagi kedalam 2 (dua) macam diantaranya sebagaimana di bawah ini:

1. DDL (*Data Definition Language*)

Yang pertama adalah bahasa DDL atau kepanjangannya *Data Definition Language*, yaitu dipakai untuk menggambarkan desain dari basis data secara menyeluruh. DDL (*Data Definition Language*) dapat dipakai untuk membuat tabel baru, memuat indeks, maupun mengubah tabel. Hasil dari kompilasi DDL akan disimpan di kamus data. Itulah definisi dari DDL.

2. DML (*Data Manipulation Language*)

Dan yang kedua adalah DML atau kepanjangannya *Data Manipulation Language*, yaitu dipakai untuk memanipulasi dan pengambilan data pada suatu basis data, misalnya seperti penambahan data yang baru ke dalam suatu basis data, menghapus data pada suatu basis data dan mengubah data pada suatu basis data. Itulah definisi dari DML.

2.2.9.9 HTML

HTML (*Hyper Text Markup Language*) adalah sebuah bahasa markup yang digunakan untuk membuat sebuah halaman web dan menampilkan berbagai informasi didalam sebuah browser internet. Bermula dari sebuah bahasa yang sebelumnya banyak digunakan di dunia penerbitan dan percetakan yang disebut dengan SGML (*Standard Generalized Markup Language*), Pengertian 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)* [11].

2.2.10 CSS

CSS (*Cascading Style Sheet*) adalah salah satu bahasa desain web (*style sheet language*) yang mengontrol format tampilan sebuah halaman web yang ditulis dengan menggunakan penanda(markup language). Biasanya CSS digunakan untuk mendesain sebuah halaman HTML dan XHTML, tetapi sekarang CSS bisa diaplikasikan untuk segala dokumen XML, termasuk SVG dan XUL bahkan ANDROID.

CSS dibuat untuk memisahkan konten utama dengan tampilan dokumen yang meliputi layout, warna da font. Pemisahan ini dapat meningkatkann daya akses konten pada web, menyediakan lebih banyak fleksibilitas dan kontrol dalam spesifikasi darisebuah karakteristik dari sebuah tampilan, memungkinkan untuk membagi halaman untuk sebuah formatting dan mengurangi kerumitan dalam penulisan kode dan struktur dari konten, contohnya teknik tableless pada desain web.

Tujuan utama CSS diciptakan untuk membedakan konten dari dokumen dan dari tampilan dokumen, dengan itu, pembuatan ataupun pemrograman ulang web akan lebih mudah dilakukan. Hal yang termasuk dalam desain web diantaranya adalah warna, ukura dan formatting. Dengan adanya CSS, konten dan desain web akan mudah dibedakan, jadi memungkinkan untuk melakukan pengulangan pada tampilan-tampilan tertentu dalam suatu web, sehingga akan memudahkan dalam membuat

halaman web yang banyak, yang pada akhirnya dapat memangkas waktu pembuatan web.

Fungsi utama css adalah merancang, merubah, mendisain, membentuk halaman website(blog juga website). Dan isi dari halaman website adalah tag-tag html, logikanya css itu dapat merubah tag-tag html(yang sederhana) sehingga menjadi lebih fungsional dan menarik.

2.2.11 JavaScript

JavaScript adalah bahasa pemrograman web yang bersifat *Client Side Programming Language*. *Client Side Programming Language* adalah tipe bahasa pemrograman yang pemrosesannya dilakukan oleh *client*. Aplikasi *client* yang dimaksud merujuk kepada web *browser* seperti Google Chrome dan Mozilla Firefox. Bahasa pemrograman Client Side berbeda dengan bahasa pemrograman Server Side seperti PHP, dimana untuk server side seluruh kode program dijalankan di sisi server. JavaScript pada awal perkembangannya berfungsi untuk membuat interaksi antara user dengan situs web menjadi lebih cepat tanpa harus menunggu pemrosesan di web server. Sebelum javascript, setiap interaksi dari user harus diproses oleh web server. Dalam perkembangan selanjutnya, JavaScript tidak hanya berguna untuk validasi form, namun untuk berbagai keperluan yang lebih modern. Berbagai animasi untuk mempercantik halaman web, fitur chatting, efek-efek modern, games, semuanya bisa dibuat menggunakan JavaScript.

2.2.12 PHP

Hypertext Preprocessor atau di singkat PHP adalah suatu bahasa pemrograman yang digunakan untuk membuat web dinamis, walau bisa juga digunakan untuk membuat program lain. Tentunya bahasa pemrograman PHP berbeda dengan HTML, pada PHP Script/kode yang di buat tidak dapat di tampilkan pada halaman/muka website begitu saja, tapi harus diproses terlebih dahulu oleh web server lalu di tampilkan dalam bentuk halaman website di web browser, Script PHP juga dapat di sisipkan pada HTML dan script PHP selalu diawali dengan `<?php` dan di akhiri dengan `?>`. Manajemen database yang biasanya digunakan untuk pemrograman PHP misalnya seperti MySQL, tapi ada juga yang menggunakan Oracle, Microsoft Access, dan lain-lain. PHP disebut juga sebagai bahasa pemrograman script server side, karena PHP di proses pada komputer server.

Beberapa kelebihan PHP yang tidak dimiliki oleh bahasa-bahasa sejenisnya, yaitu [17]:

1. Bahasa pemrograman 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.
5. 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 sistem.

2.3 Perangkat Lunak Pendukung

Perangkat lunak pendukung untuk pembangunan system informasi ini adalah sebagai berikut :

2.3.1 XAMPP

Xampp adalah sebuah aplikasi *open source* untuk pengembangan sebuah web di Windows. Xampp memungkinkan untuk mengembangkan web dengan Apache2, PHP dan database MySQL. Dengan kata lain, wampserver digunakan untuk membuat web server lokal pada komputer guna memudahkan perancangan dan pembuatan web sebelum dipublikasikan ke internet atau jaringan lokal (LAN).

2.3.2 Sublime Text

Sublime Text Editor adalah editor teks untuk berbagai bahasa pemrograman termasuk pemrograman PHP. Sublime Text Editor merupakan editor text lintas-platform dengan Python application programming interface (API). Sublime Text Editor juga mendukung banyak bahasa pemrograman dan bahasa markup, dan fungsinya dapat ditambah dengan plugin, dan Sublime Text Editor tanpa lisensi perangkat lunak. Sublime Text mendukung berbagai bahasa pemrograman dan mampu menyajikan fitur *syntax highlight* hamper di semua bahasa pemrograman yang didukung ataupun dikembangkan oleh komunitas seperti; C, C++, C#, CSS, D, Dylan, Erlang, HTML, Groovy, Haskell, Java, JavaScript, LaTeX, Lisp, Markdown, MATLAB, OCaml, Perl, PHP, Python, R, Ruby, SQL, TCL, Textile and XML. Biasanya bagi bahasa pemrograman yang didukung belum terdukung secara *default* dapat lebih dimaksimalkan atau didukung dengan menggunakan *add-ons* yang bisa didownload sesuai kebutuhan *user*

2.3.3 Google Chrome

Google Chrome adalah sebuah peramban web sumber terbuka yang dikembangkan oleh Google dengan menggunakan mesin rendering WebKit. Proyek sumber terbukanya sendiri dinamakan Chromium.

Adapun kelebihan dan kekurangan pada google chrome, diantaranya :

1. Kelebihan Google Chrome
 - a. Tampilan : Pada Interface dari Chrome terlihat bahawa google ingin para penggunanya lebih fokus pada web dan melupakan browser yang digunakan.

Ini artinya Google Chrome memiliki tampilan yang tidak mengusik dan nyaman ketika sedang digunakan.

b. Aplikasi Web : Google memberikan opsi “Make Application Shortcut” . Dengan underline ini sebuah aplikasi web seperti GMAIL atau Google Teader dapat dijalankan lewat shrtcut pada Desktop atau Start Menu. Sehingga kelihatan seperti sebuah aplikasi local

c. Pencarian : Chrome memiliki fungsi pencarian yang sangat baik. Contohnya, chrome daat mendeteksi ketika pengguna pernah melakukan pencarian di suatu website dan memasukkan website tersebut dalam daftar penyedia pencarian.

d. Pengelolaan Memori : Pada setiap TAB yang dibuka di Chrome memiliki proses yang terpisah, sehingga ketika eror/crash pada salah satu ta tidak akan meyebabkan seluruh browser eror. (ini kekurangan yang terdapat pada Firefox dan browser lainnya).

No	Peneliti	Judul	Hasil Penelitian	Alasan Menjadi Tujuan Penelitian
1	Sony Taufan Hambali	Penerapan Metode <i>Economic Order Quantity (EOQ)</i> Pada Sistem Informasi Manajemen Persediaan Barang Di CV. Muda Mandiri	Persediaan bahan baku yang dihitung dengan menggunakan metode EOQ dapat menentukan pengadaan barang yang sesuai di setiap periodenya, sehingga barang yang diadakan tidak melebihi kapasitas gudang yang dimiliki CV. Muda Mandiri.	Jurnal ini memiliki kesamaan tujuan yaitu mempunyai kasus yang serupa dengan kasus penelitian saya tetapi dengan yang membedakan dari kasus penelitian saya dengan jurnal ini yaitu hanya dari segi barang yang diadakan.
2	Haryanto Tanuwijaya	Penerapan Metode <i>Winter's Exponential Smoothing</i> Dan <i>Single Moving Average</i> Dalam Sistem Informasi Pengadaan Obat Rumah Sakit	Kesimpulan yang dapat diambil dari pembuatan SI Pengadaan Obat dengan Metode <i>Winter's Exponential Smoothing</i> dan <i>Single Moving Average</i> adalah sebagai berikut: 1. Sistem Informasi Pengadaan Obat dapat menghasilkan prediksi kebutuhan pemakaian obat periode bulan mendatang untuk mendukung pengambilan keputusan	Alasan jurnal ini menjadi tujuan penelitian saya karena memiliki tujuan yang sama yaitu menghasilkan kebutuhan yang diperlukan untuk tiap periodenya, dan perbedaannya hanya dari segi barang yang diadakan.

			<p>pembelian obat yang tepat oleh Kepala kamar obat RS.</p> <p>2. Dari hasil menunjukkan bahwa 61,9% obat RS memiliki pola data bersifat stationer sehingga lebih tepat menggunakan metode SMA. Sedangkan 38,1% obat sisanya memiliki pola data bersifat musiman sehingga lebih tepat menggunakan metode WES.</p> <p>3. Setiap jenis obat memiliki karakteristik data <i>time series</i> yang berbeda.</p> <p>4. Panjang periode pada metode SMA menentukan keakurasian hasil peramalan..[3]</p>	
3	Agung wahana, Asep Ririh Riswaya	Sistem Informasi Pengadaan Barang ATK di PT. Mekar Cipta Indah Menggunakan PHP dan MYSQL	<p>Kesimpulan adalah sebagai berikut :</p> <p>1.Sistem ini mampu menghasilkan data stock barang yang akurat, dari hasil penjumlahan stock awal dan transaksi masuk-transaksi keluar.</p> <p>2. Sistem ini dapat menampilkan daftar stock barang yang kurang dari minimal stock, sehingga bagian gudang dapat melakukan pengadaan barang secara</p>	Alasan jurnal ini menjadi tujuan penelitian saya adalah karena jurnal ini memiliki tujuan yang sama dengan kasus yang sama ambil.

			teratur Laporan yang dihasilkan oleh sistem ini cukup rapi, jelas dan akurat.[4]	
4	Ni Ketut Dewi Ari Jayanti	Perencanaan Sistem Informasi Persediaan Barang Menggunakan Model Reorder Point	Menghasilkan sebuah dokumen perancangan yang baik untuk sistem informasi persediaan barang. Sedangkan untuk menentukan jumlah persediaan barang yang tepat menggunakan model reorder point. Metodologi rekayasa perangkat lunak yang digunakan dalam perancangan ini adalah berorientasi objek dan tools yang digunakan adalah UML diagram grafis. Output atau luaran dari penelitian ini berupa dokumen perancangan yang dapat dijadikan sebagai panduan dalam pembangunan / implementasi sistem informasi persediaan barang. [2]	Alasan saya menjadikan jurnal ini sebagai tujuan penelitian saya adalah karena jurnal ini memiliki tujuan yang sama dengan kasus yang saya ambil.
5	Helmi Kurniawan	Perancangan Aplikasi Sistem Pengendalian Stock Barang Pada UD.Makmur Jaya	Berdasarkan hasil pengamatan yang dilakukan pada UD. Makmur Jaya dapat diketahui bahwa sistem yang digunakan masih menggunakan sistem semi komputerisasi	Alasan saya menjadikan jurnal ini sebagai tujuan penelitian saya adalah karena jurnal ini memiliki tujuan yang sama dengan kasus yang saya ambil,

			<p>sehingga dalam pengolahan data masih cukup lambat dan informasi yang dihasilkan sering tidak akurat. Sedangkan sistem baru yang akan dibangun merupakan sebuah sistem informasi yang dirancang menggunakan Visual Basic.Net dan database menggunakan SQL Server. Sistem informasi pengendalian stock makanan ringan yang dibangun ini bisa digunakan oleh karyawan perusahaan untuk membuat laporan pengendalian stock makanan ringan yang ada pada perusahaan ini secara cepat dan akurat.[5]</p>	<p>yang membedakan antara jurnal ini dan jurnal saya adalah jurnal ini menggunakan Visual Basic Net.</p>
--	--	--	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	----------------------------------------------------------------------------------------------------------