

BAB II

TINJAUAN PUSTAKA

II.1. Penelitian Terkait

Adapun penelitian terkait yang akan digunakan sebagai sumber acuan yang relevan dan terkini yaitu:

1. Menurut Aleksius Madu, 2016, yang berjudul “ Aplikasi Peramalan Pengadaan Barang Dengan Metode *Trend Projection* Dan Metode *Single Exponential Smoothing* (Studi Kasus Di Toko Pionir Jaya)” . Tujuan dari penelitian ini adalah meramalkan jumlah korban kecelakaan lalu lintas yang meninggal dunia di Kabupaten TTU dengan metode Trend Projection dan metode Backpropagation, serta membandingkan kedua metode tersebut berdasarkan tingkat kesalahannya dan meramalkan jumlah korban kecelakaan lalu lintas di kabupaten TTU untuk tahun mendatang. Penelitian ini dilakukan di Kabupaten TTU dimana data yang digunakan dalam penelitian ini diperoleh dari Unit Laka Lantas Polres Kabupaten TTU. Data tersebut berupa data mengenai jumlah kecelakaan lalu lintas di Kabupaten TTU tahun 2000 – 2013 dengan analisis kuantitatif yang menggunakan metode Trend Projection dan metode Backpropagation.
2. Menurut Maftahatul Hakimah, 2015, yang berjudul “Rancang Bangun Aplikasi Peramalan Persediaan Barang Dengan Metode Trend Projection” Penelitian ini bertujuan untuk membuat suatu aplikasi yang dapat membantu memudahkan dan memaksimalkan kinerja pegawai administrasi

gudang dalam meramalkan jumlah barang yang harus dipesan ke pabrik untuk periode berikutnya. Metode peramalan yang digunakan adalah metode kuantitatif trend projection dengan variasi data musiman. Metode ini membutuhkan informasi data pada tahun-tahun sebelumnya sehingga pada penelitian ini mengambil data 3 tahun sebelumnya. Dengan metode peramalan ini diperoleh hasil peramalan nilai permintaan hampir mendekati dengan data sebenarnya. Dari hasil pengujian sistem yang dikenakan pada 5 data permintaan diperoleh tingkat akurasi sistem sebesar 86%.

3. Menurut Billy Eden William Asrul, 2018, yang berjudul “ Implementasi Metode Trend Projection Untuk Peramalan Persediaan Ikan Hias Air Tawar Studi Kasus: Batara Indo Aquatic” hasil dari penelitian yaitu dengan adanya sistem Pendukung Keputusan Peramalan Persediaan ini dapat membantu pihak Batara Indo Aquatic dalam menentukan jumlah persediaan ikan untuk periode yang akan datang. Hal ini dikarenakan nilai penjualan 2017 yang diramalkan hampir mendekati data penjualan sebenarnya. Sistem ini memberikan kemudahan bagi customer untuk mengetahui jenis ikan, harga ikan yang dijual, serta informasi umum tentang Batara Indo Aquatic serta. Metode peramalan trend projection ini adalah suatu metode peramalan menggunakan analisa regresi runtun waktu yang sesuai dengan garis trend dengan variasi data penjualan musiman terhadap serangkaian data masa lalu yang diproyeksikan ke permasalahan masa depan.

4. Menurut Giandicka, 2020, yang berjudul “Sistem Informasi Pengolahan Data Penjualan Dan Peramalan Pembelian Setiap Pelanggan Dengan Metode Trend Projection Berbasis Website Pada PT. Aldora Sukses Perkasa Tanjungpinang” hasil dari penelitian yaitu Aplikasi yang dibuat dapat mengolah peramalan pembelian setiap pelanggan dengan metode trend projection. Sistem yang dibuat dapat membantu dalam mencetak laporan stok dan laporan penjualan. Pelanggan dapat melakukan retur barang di sistem yang disediakan dengan beberapa kriteria yang harus di cantumkan..

II.2. Uraian Teoritis

II.2.1. Sistem Informasi

Sistem informasi didefinisikan oleh Mulyadi (2001) sebagai salah satu sumber daya bisnis, seperti bahan mentah, modal, dan tenaga kerja, informasi merupakan sumber daya vital bagi kelangsungan hidup organisasi bisnis. (Agus Irawan : 2 : 2016).

Sistem informasi ini terdiri dari kata sistem dan informasi. Sistem berasal dari bahasa Latin (*systema*) dan bahasa Yunani (*sustema*) adalah suatu kesatuan yang terdiri komponen atau elemen yang saling bekerja sama atau yang dihubungkan dengan cara-cara tertentu sehingga membentuk satu kesatuan untuk melaksanakan suatu fungsi guna mencapai suatu tujuan. Sedangkan kata informasi berasal dari kata Perancis kuno *informacion* (tahun 1387) yang diambil dari bahasa Latin *informationem* yang berarti “garis besar, konsep, ide”. Informasi merupakan data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti

bagi yang menerimanya Berdasarkan definisi di atas, dapat disimpulkan bahwa sistem informasi adalah sekumpulan prosedur organisasi yang dilaksanakan untuk mencapai suatu tujuan yaitu memberikan informasi bagi pengambil keputusan dan untuk mengendalikan organisasi (Zaenal Sekty Wijaya: 2 : 2015).

Sistem informasi berfungsi menghasilkan informasi untuk memenuhi kebutuhan aktivitas organisasi. Informasi yang bernilai tinggi dihasilkan oleh suatu sistem informasi yang efektif dan efisien. Untuk itu, sistem informasi yang efektif dan efisien menghendaki intervensi manajemen secara tepat Tomi Loveri : 5 : 2018

II.2.1.1 Komponen Sistem Informasi

Sistem informasi terdiri dari komponen-komponen sebagai berikut :

1. Blok Masukan

Input mewakili data yang masuk ke dalam sistem informasi. Input disini termasuk metode-metode dan media untuk menangkap data yang akan dimasukkan, yang dapat berupa dokumen-dokumen dasar.

2. Blok Model

Blok model ini terdiri dari kombinasi prosedur, logika dan model matematik yang akan memanipulasi data input dan data yang tersimpan di basis data dengan cara yang sudah tertentu untuk menghasilkan keluaran yang diinginkan.

3. Blok keluaran (Output Block)

Produk dari sistem informasi adalah keluaran yang merupakan informasi yang berkualitas dan dokumentasi yang berguna untuk semua tingkatan manajemen serta semua pemakai sistem.

4. Blok Teknologi

Teknologi digunakan untuk menerima input, menjalankan model, menyimpan dan mengakses data, menghasilkan dan mengirimkan keluaran dan membantu pengendalian dari sistem secara keseluruhan. Teknologi terdiri dari 3 bagian utama, yaitu teknisi (*humanware* atau *brainware*), perangkat lunak (*software*), dan perangkat keras (*hardware*).

5. Blok Basis Data

Basis data (*database*) merupakan kumpulan dari data yang saling berhubungan satu dengan yang lainnya, tersimpan di perangkat keras komputer dan digunakan perangkat lunak untuk memanipulasinya. Basis data diakses atau dimanipulasi dengan menggunakan perangkat lunak paket yang disebut dengan DBMS (*Database Management System*).

6. Blok kendali

Banyak hal yang dapat merusak sistem informasi, seperti misalnya bencana alam, api, temperature, air, debu, kecurangan-kecurangan, kegagalan-kegagalan sistem itu sendiri, kesalahan-kesalahan, ketidak efisienan, sabotase dan lain sebagainya. Beberapa pengendalian perlu dirancang dan diterapkan untuk menyakinkan bahwa hal-hal yang dapat merusak sistem dapat dicegah ataupun

bila terlanjur terjadi kesalahan-kesalahan dapat langsung cepat diatasi. (Tomi Loveri : 2 : 2018)

2.2.1.2 Tujuan Perancangan Sistem Informasi

Pada umumnya tujuan melakukan suatu perancangan sistem informasi adalah (Eka Iswandi : 3 : 2015) :

1. Untuk memenuhi kebutuhan pemakai sistem informasi.
2. Untuk memberikan gambaran yang jelas dan rancang bangun lengkap kepada pemrogram komputer dan ahli teknik lain yang terlibat.
3. Untuk mendukung pengolahan pelaporan manajemen dan mendukung perusahaan.

2.2.1.3. Alat Bantu Dalam Perancangan Sistem Informasi

Dalam perancangan sistem informasi memiliki alat bantu perancangan sistem yaitu (Eka Iswandi : 3 : 2015) :

1. Aliran Sistem Informasi (ASI)

Merupakan bagan alir yang menunjukkan arus dari laporan dan formulir termasuk tembusan-tembusannya. Untuk itu dibutuhkan pedoman– pedoman untuk membuat Aliran Sistem Informasi (ASI).

2. Contex Diagram (CD)

Context diagram merupakan data flow diagram yang menggambarkan garis besar operasional sistem. Konteks diagram menggambarkan hubungan sistem dengan entitas-entitas di luar sistem. CD memperlihatkan sistem sebuah

proses. Tujuannya adalah memberikan pandangan umum sistem. CD memperlihatkan sebuah proses yang berinteraksi dengan lingkungan luarnya. Ada pihak luar yang memberikan masukan dan pihak yang menerima keluaran sistem.

3. Data Flow Diagram (DFD)

Diagram aliran data sistem disebut juga dengan *Data Flow Diagram* (DFD). DFD sering digunakan untuk menggambarkan suatu sistem yang telah ada atau sistem baru yang akan dikembangkan secara logika tanpa mempertimbangkan lingkungan fisik dimana data tersebut mengalir atau lingkungan fisik dimana data tersebut disimpan. DFD menggambarkan arus data didalam sistem dengan terstruktur dan jelas.

4. Entity Relationship Diagram (ERD)

Entity Relationship Diagram (ERD) memiliki dua komponen utama yaitu Entitas (*Entity*) dan Relasi (*Relation*). Kedua komponen ini ,masing-masing dilengkapi dengan sejumlah atribut yang mempresentasikan seluruh fakta yang ada di dunia nyata.

5. Flowchart

Flowchart merupakan urutan-urutan langkah kerja suatu proses yang digambarkan dengan menggunakan simbol-simbol yang disusun secara sistematis.

II.2.2. Metode Peramalan

Metode peramalan (forecasting) merupakan suatu teknik untuk memperkirakan atau memprediksikan suatu nilai pada masa yang akan datang dengan memperhatikan data atau informasi masa lalu atau saat ini baik

secara matematik atau statistik. Sedangkan ramalan adalah suatu situasi atau kondisi yang diperkirakan akan terjadi pada masa yang akan datang. Metode ini sangat membantu dalam mengambil keputusan yang tepat. Baik tidaknya suatu peramalan yang disusun, disamping ditentukan oleh metode yang digunakan, juga ditentukan baik tidaknya informasi yang digunakan. Selama informasi yang digunakan tidak dapat menyakinkan, maka hasil peramalan juga akan sulit untuk dipercaya ketepatannya. Berdasarkan teknik yang digunakan, metode peramalan dapat digolongkan ke dalam dua kategori, yaitu :

1. Metode peramalan kualitatif Peramalan kualitatif adalah peramalan yang berdasarkan atas data kualitatif pada masa lalu. Hasil peramalan yang dibuat sangat bergantung kepada orang yang menyusun.
2. Metode peramalan kuantitatif Peramalan kuantitatif adalah peramalan yang berdasarkan atas data kuantitatif pada masa lalu. Hasil peramalan yang dibuat sangat bergantung pada metode yang dipergunakan. Dengan metode yang berbeda akan diperoleh hasil peramalan yang berbeda. Semakin kecil penyimpangan antara hasil ramalan dan kondisi sebenarnya berarti metode yang dipergunakan semakin baik. (Maftahatul Hakimah : 2: 2015)

II.2.3. Metode Trend Projection

Metode peramalan dengan proyeksi trend ini adalah mencocokkan garis trend ke rangkaian titik data historis kemudian memproyeksikan garis tersebut ke masa depan dengan horizon waktu menengah dan panjang. Metode

proyeksi pada penelitian ini melihat trend pada garis lurus (linier). Pendekatan yang tepat untuk trend linier adalah metode kuadrat terkecil (least square). Metode kuadrat terkecil adalah langkah untuk menentukan garis lurus yang paling sesuai untuk pola data yang muncul. Tujuan utama metode ini adalah meminimalkan jumlah kuadrat kesalahan atau selisih dari persamaan regresi linier dengan masing-masing data sebenarnya (Maftahatul Hakimah : 2 : 2015)

Metode Trend Projection adalah suatu metode peramalan serangkaian waktu yang sesuai dengan garis trend terhadap serangkaian titik-titik data masa lalu, kemudian diproyeksikan ke dalam peramalan masa depan. Metode ini merupakan garis trend untuk persamaan matematis. Pada kenyataannya, anggapan bahwa trend dapat diwakili oleh beberapa fungsi sederhana seperti garis lurus sepanjang periode untuk waktu berkala yang diamati jarang ditemukan. Ada tiga trend yang digunakan untuk meramalkan pergerakan keadaan pada masa yang akan datang, yaitu :

1. Trend Linier, yaitu suatu trend yang kenaikan atau penurunan nilai yang akan diramalkan naik atau turun secara linier. Dengan persamaan tren linear $Y = a + bX$
2. Trend Parabolik (kuadrat) disebut juga trend kuadratis, merupakan “salah satu bentuk trend non linear yaitu trend yang variabel X-nya berpangkat paling tinggi 2 .”. Hubungan variabel dependen dan independen adalah kuadrat, dengan persamaan untuk trend kuadrat adalah $Y = a + bX + cX^2$.
3. Trend Eksponensial, yaitu sebuah trend yang nilai variabel tak bebasnya naik secara berlipat ganda atau tidak linier. Analisis trend yang digunakan secara

umum untuk model trend pertumbuhan eksponensial dengan persamaan trend eksponensial adalah $Y^t = a(1+b)^x$. (Aleksius Madu : 2 : 2016)

Metode trend projection bisa disebut juga metode tren garis lurus. Adapun persamaan *trend linier* dapat ditulis sebagai berikut :

$$Y^t = a + bX \dots \dots \dots (1)$$

Dimana :

Y^t = data berkala (time series data),

a dan b = konstanta

X = waktu (hari, minggu, bulan, tahun)

Pada umumnya yang diberi nilai 0 adalah variable waktu yang letaknya ditengah Untuk mencari nilai variable waktu adalah sebagai berikut :

1. Untuk nilai variable waktu (= X) adalah nol (0)

a. Untuk nilai n ganjil adalah :

$$n = 2k + 1 \quad k = \frac{n-1}{2} \dots \dots \dots (2)$$

$$\frac{Xk+1}{5} = 0 \dots \dots \dots (3)$$

b. Untuk nilai n genap adalah :

$$n = 2k \quad k = \frac{n}{2} \dots \dots \dots (4)$$

$$\frac{Xk+1}{5} (k+1) = 0 \dots \dots \dots (5)$$

(Maftahatul Hakimah : 2 : 2015).

II.2.4. Visual Studio 2010

Microsoft Visual Studio merupakan sebuah perangkat lunak lengkap (*suite*) yang dapat digunakan untuk melakukan pengembangan aplikasi, baik itu aplikasi bisnis, aplikasi personal, ataupun komponen aplikasinya, dalam bentuk aplikasi

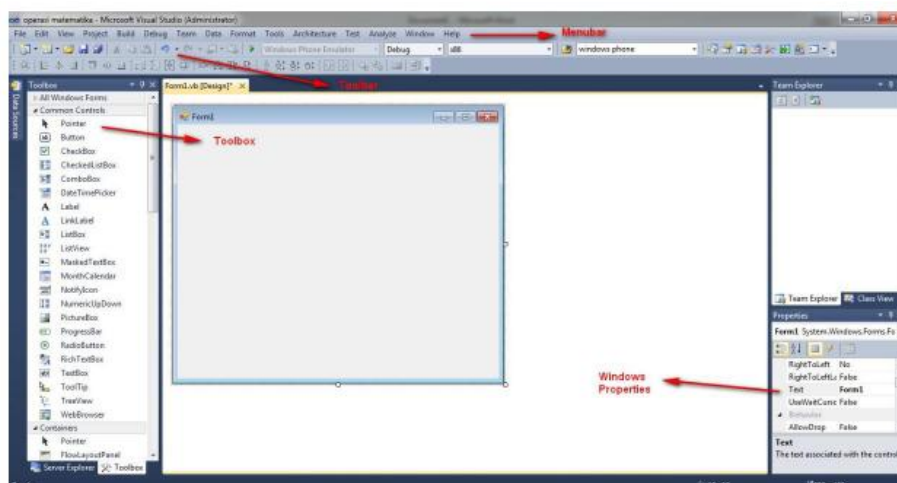
console, aplikasi Windows, ataupun aplikasi Web. Visual Studio mencakup kompiler, *Software Development Kit* (SDK), *Integrated Development Environment* (IDE), dan dokumentasi (umumnya berupa MSDN Library). Kompiler yang dimasukkan ke dalam paket Visual Studio antara lain Visual C++, Visual C#, Visual Basic, Visual Basic .NET, Visual InterDev, Visual J++, Visual J#, Visual FoxPro, dan Visual SourceSafe.

Microsoft Visual Studio dapat digunakan untuk mengembangkan aplikasi dalam *native code* (dalam bentuk bahasa mesin yang berjalan di atas Windows) ataupun *managed code* (dalam bentuk Microsoft Intermediate Language di atas .NET Framework). Selain itu, Visual Studio juga dapat digunakan untuk mengembangkan aplikasi *Silverlight*, aplikasi *Windows Mobile* (yang berjalan di atas *.NET Compact Framework*).

Visual Studio sebelumnya versi Visual Studio 9.0.21022.08, atau dikenal dengan sebutan Microsoft Visual Studio 2008 yang diluncurkan pada 19 November 2007, yang ditujukan untuk *platform* Microsoft *.NET Framework 3.5*. Versi sebelumnya, Visual Studio 2005 ditujukan untuk platform *.NET Framework 2.0* dan *3.0*. Visual Studio 2003. ditujukan untuk *.NET Framework 1.1*, dan Visual Studio 2002 ditujukan untuk *.NET Framework 1.0*. Versi-versi tersebut di atas kini dikenal dengan sebutan Visual Studio .NET, karena memang membutuhkan Microsoft *.NET Framework*. Sementara itu, sebelum muncul Visual Studio .NET, terdapat Microsoft Visual Studio 6.0 (VS1998). (Herpendi :2 : 2016).

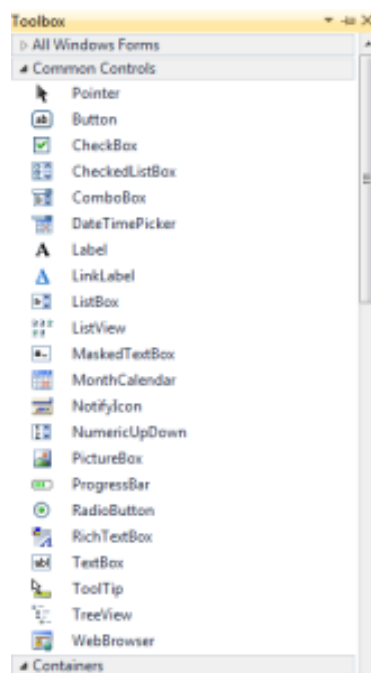
II.2.4.1 Komponen dalam VB 2010

Jendela Visual Basic atau sering juga disebut lingkungan kerja Visual Basic mempunyai tampilan yang hampir sama dengan tampilan jendela program aplikasi Windows yang sudah kita kenal, seperti Word, Excel, dan Power Point. Di aplikasi visual basic terdapat menu-menu dan toolbar yang memuat icon-icon dan tombol-tombol untuk menjalankan perintah-perintah. Perbedaannya, Visual Basic mempunyai beberapa tambahan komponen, yaitu **Toolbox**, **Windows Project**, dan **Windows Properties**.



Gambar II.1. Komponen Visual Studio 2010
(Sumber : Wahyu Trisno Atmojo : 6 : 2018)

Toolbox ditempatkan disebelah kiri jendela kerja. Window Project dan Window Properties berada disebelah kanan jendela kerja. Jika kita akan membuat sebuah project baru, maka pada bagian tengah jendela akan terdapat sebuah form. Form tersebut akan digunakan sebagai tampilan (jendela) untuk program yang dibuat.



Gambar II.2. Toolbox Visual Studio 2010
 (Sumber : Wahyu Trisno Atmojo : 9 : 2018)

II.2.5. SQL Server

SQL Server merupakan suatu *Relational Database Management Systems* (RDBMS) yang digunakan untuk menyimpan data. Data yang disimpan pada *database* bisa dalam skala kecil maupun besar. Selain itu, penyajiannya merupakan penyajian pada level fisik karena kita akan menyimpan langsung data pada *database* dengan kondisi yang sebenarnya, yaitu disimpan pada tabel apa, kolom mana, dan menggunakan tipe data saat penyimpanan.

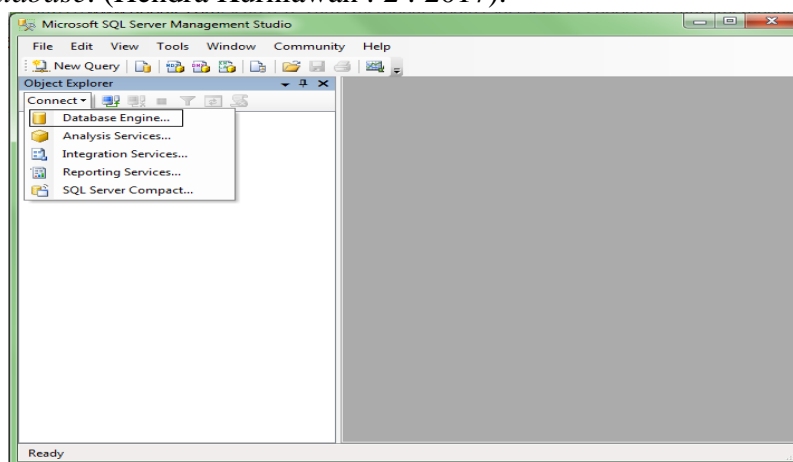
Database merupakan suatu tempat untuk menyimpan data. Pada sebuah *database* bisa terdapat satu atau lebih tabel dan *query*. Operasi yang biasanya dilakukan *database* berhubungan erat dengan pengaksesan tabel atau *query*. Pada *SQL Server 2008* terdapat fitur-fitur yang dapat mengembangkan performa dari *database* tersebut. Beberapa fitur tersebut, yaitu :

- 1) *Date Data Type*: Digunakan untuk menyimpan data tanggal saja sehingga akan menghemat *space* pada *server*.
- 2) *Data Compression*: Digunakan untuk melakukan compress data sehingga ukuran data yang disimpan dalam hal *space hardisk* akan lebih kecil.
- 3) *Sparse Column*: Digunakan untuk menyimpan data yang memiliki lebih banyak data NULL dengan lebih efisien.
- 4) *Row Constructor*: Digunakan untuk melakukan insert beberapa data sekaligus dengan satu perintah INSERT.
- 5) *Table-Valued Parameter*: Digunakan untuk melakukan parsing array pada bahasa pemrograman, dimana satu variable diberikan data-data yang akan diproses setelahnya. (Benardo : 4 : 2015).

Microsoft SQL server 2000 adalah Sistem manajemen basis data yang memakai perintah *Transact-SQL* untuk mengirim perintah dari Komputer client ke Komputer server. Microsoft SQL Server 2000 dapat dijalankan pada NT 4.0 Server atau *Microsoft Windows 2000 Server*, selain itu dapat pula di install pada personal desktop di Windows 2000 Profesional, *Windows 98 Milenium*. Sistem akan dipergunakan oleh beberapa komputer (*client-server*) maka database yang dibangun adalah merupakan database yang berfungsi untuk menunjang hal tersebut untuk itu dalam penulisan ini dipakai database *Microsoft SQL Server 2000*, selain itu database ini juga compatible dengan bahasa pemrograman yang digunakan. Secara teoritis, program *SQL* yang sudah terinstall pada komputer dapat menampung *32.767 database* dan terdapat lebih dari *2 billion objek*. *SQL Server* adalah sistem manajemen database relasional (RDBMS) yang dirancang

untuk aplikasi dengan arsitektur *client/server*. Istilah *client*, *server*, dan *client/server* dapat digunakan untuk merujuk kepada konsep yang sangat umum atau hal yang spesifik dari perangkat keras atau perangkat lunak. Pada level yang sangat umum, sebuah *client* adalah setiap komponen dari sebuah sistem yang meminta layanan atau sumber daya (*resource*) dari komponen sistem lainnya. Sedangkan sebuah *server* adalah setiap komponen sistem yang menyediakan layanan atau sumber daya ke komponen sistem lainnya.

SQL Server menggunakan tipe dari *database* yang disebut database relasional. Database relasional adalah database yang digunakan sebuah data untuk mengatur atau mengorganisasikan kedalam tabel. Tabel-tabel adalah alat bantu untuk mengatur atau mengelompokan data mengenai subyek yang sama dan mengandung informasi dan kolom dan baris. Tabel-tabel saling berhubungan dengan mesin *database* ketika dibutuhkan. *SQL Server* mendukung beberapa tipe data yang berbeda, termasuk untuk karakter, angka, tanggal (*datetime*) dan uang (*money*), *SQL Server* digunakan untuk menggambarkan model dan implementasi pada *database*. (Hendra Kurniawan : 2 : 2017).



Gambar II.3. Tampilan SQL Server
(Sumber : Hendra Kurniawan : 3 : 2017)

II.2.6. Database

Konsep Database Pangkalan data atau basis data (bahasa Inggris: database), atau sering pula dieja basisdata, 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). Sistem basis data dipelajari dalam ilmu informasi.

Database merupakan suatu tempat untuk menyimpan data. Pada sebuah *database* bisa terdapat satu atau lebih tabel dan *query*. Operasi yang biasanya dilakukan *database* berhubungan erat dengan pengaksesan tabel atau *query*. Selain itu juga dalam mengambil data dari server lain akan mengalami penurunan performa. Tetap dengan menggunakan terdistribusi, bisa dengan cepat melakukan akses untuk data pada database server yang didistribusikan. Sedangkan untuk tersentralisasi, karena databasenya hanya satu dan terpusat (misalnya di *head office*) maka seluruh client dari manapun akan mengambil data tersebut dari satu database. Dengan demikian data yang diambil tidak akan bermasalah dalam hal konsistensi karena berada dalam satu sumber, tetapi akan membutuhkan hardware yang jauh lebih besar dan bandwidth yang lebih tinggi. Hal ini dikarenakan server tersebut berfungsi untuk menampung penggunaan connection yang sangat banyak. (Benardo, 2015).

Database menurut Bambang Hariyanto dalam (Minarni & Susanti, 2014) adalah :”kumpulan data (elementer) yang secara logic berkaitan dalam

mempresentasikan fenomena/fakta secara terstruktur dalam domain tertentu untuk mendukung aplikasi dalam system tertentu”. (Ade Putra, 2016).

Basis data atau Database merupakan kumpulan file - file yang saling berkaitan dan berinteraksi, relasi tersebut bila ditunjukan dengan kunci dari tiap - tiap file yang ada. Satu database menunjukkan suatu kumpulan data yang dipakai dalam suatu lingkup perusahaan, instansi. Pengolahan database merupakan suatu cara yang dilakukan terhadap file-file yang berada di suatu instansi yang mana file tersebut dapat disusun, diurut, diambil sewaktu-waktu serta dapat ditampilkan dalam bentuk suatu laporan sehingga dapat mengolah file-file yang berisikan informasi tersebut secara rapi. (Rini Asmara : 4 : 2016)

II.2.6.1 Komponen DBMS

Komponen-komponen DBMS menurut Howe yang dikutip oleh Dzacko (2007) terdiri dari:

- a. *Interface*, yang didalamnya terdapat bahasa manipulasi data (*data manipulation language*).
- b. Bahasa definisi data (*data definition language*) untuk *skema eksternal*, *skema konseptual* dan *skema internal*.
- c. Sistem kontrol basis data (*Database Control System*) yang mengakses basis data karena adanya perintah dari bahasa manipulasi data.

Contoh bahasa menggunakan komponen-komponen tersebut adalah SQL (*Structured Query Language*). SQL merupakan bahasa standar yang digunakan oleh kebanyakan aplikasiaplikasi DBMS (Rini Asmara : 4 : 2016).

II.2.7 UML (*Unified Modelling Language*)

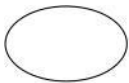

Unified Modeling Language (UML) adalah bahasa spesifikasi standar yang dipergunakan untuk mendokumentasikan, menspesifikasikan dan membangun perangkat lunak. UML merupakan metodologi dalam mengembangkan sistem berorientasi objek dan juga merupakan alat untuk mendukung pengembangan sistem.



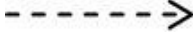

Alat bantu yang digunakan dalam perancangan berorientasi objek berbasis UML adalah sebagai berikut:

II.2.7.1. *Use Case Diagram*

Use case diagram merupakan pemodelan untuk melakukan (behavior) sistem informasi yang akan dibuat. *Use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut. Simbol-simbol yang digunakan dalam *Use Case Diagram* yaitu:

Tabel II.1. Simbol *Use Case Diagram*

Gambar	Keterangan
	<p><i>Use Case</i> menggambarkan fungsionalitas yang disediakan sistem sebagai unit-unit yang bertukaran pesan antara unit dengan aktor, biasanya dinyatakan dengan menggunakan kata kerja diawal nama <i>Use Case</i> .</p>
	<p>Aktor adalah <i>abstraction</i> dari orang atau sistem yang lain yang mengaktifkan fungsi dari target sistem. Untuk mendefinisikan aktor, harus ditentukan pembagian tenaga kerja dan tugas-tugas yang berkaitan dengan peran pada konteks target sistem. Orang atau sistem biasa muncul dalam beberapa peran. Perlu dicatat bahwa aktor berinteraksi</p>

	dengan <i>use case</i> tetapi tidak memiliki <i>control</i> terhadap <i>use case</i> .
	Asosiasi antara aktor dan <i>use case</i> , digambarkan dengan garis tanpa panah yang mengindikasikan siapa atau apa yang meminta interaksi secara langsung dan bukannya mengindikasikan aliran data.
	Asosiasi antara aktor dan <i>use case</i> yang menggunakan panah terbuka untuk mengidentifikasi bila aktor berinteraksi secara pasif dengan sistem
	<i>Include</i> , merupakan didalam <i>use case</i> lain (<i>required</i>) atau pemanggilan <i>use case</i> oleh <i>use case</i> lain, contohnya adalah pemanggilan sebuah fungsi program.
	<i>Extend</i> , merupakan perluasan dari <i>use case</i> lain jika kondisi atau syarat dipenuhi.

(Sumber : Ade Hendini, 2016)

II.2.7.2. Class Diagram

Class Diagram merupakan hubungan antar kelas dan penjelasan detail tiap-tiap kelas dalam model desain dari suatu sistem juga memperlihatkan aturan-aturan dan tanggung jawab entitas yang menentukan perilaku sistem.

Tabel II.2. Simbol Class Diagram




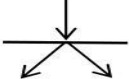
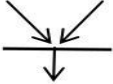
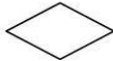

Multiplicity	Penjelasan
1	Satu dan hanya satu
0...*	Boleh tidak ada atau 1 atau lebih
1...*	1 atau lebih
0..1	Boleh tidak ada maksimal 1
n..n	Batasan antara Contoh 2..4 mempunyai arti minimal 2 maksimum 4

(Sumber : Ade Hendini, 2016)

II.2.7.3. Activity Diagram

Activity Diagram menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis. Simbol-simbol yang digunakan dalam *activity diagram* yaitu :

Tabel II.3. Activity Diagram




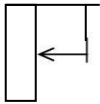


Gambar	Keterangan
	<i>Start Point</i> , diletakkan pada pojok kiri atas dan merupakan awal aktifitas.
	<i>End Point</i> , akhir aktifitas
	<i>Activities</i> menggambarkan suatu proses atau kegiatan bisnis.
	<i>Fork</i> (Percabangan) digunakan untuk menunjukkan kegiatan yang dilakukan secara parallel atau untuk menggabungkan dua kegiatan parallel menjadi satu.
	<i>Join</i> (Penggabungan) atau rake, digunakan untuk menunjukkan adanya dekomposisi.
	<i>Decision points</i> menggambarkan pilihan untuk pengambilan keputusan, true, false
	<i>Swimlane</i> pembagian <i>activity diagram</i> untuk menunjukkan siapa melakukan apa

(Sumber : Ade Hendini, 2016)

II.2.7.4. Sequence Diagram

Sequence Diagram menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan pesan yang dikirimkan dan diterima antar objek. Simbol-simbol yang digunakan dalam *sequence diagram* yaitu :

Tabel II.4. Simbol *Sequence Diagram*

Gambar	Keterangan
	<i>Entity Class</i> , merupakan bagian dari system yang berisi kumpulan kelas berupa entitas-entitas yang membentuk gambaran awal system dan menjadi landasan untuk menyusun basis data
	<i>Boundary Class</i> berisi kumpulan kelas yang menjadi <i>interface</i> atau interaksi antara satu atau lebih aktor dengan sistem. Seperti tampilan <i>formentry</i> dan <i>form</i> cetak.
	<i>Control Class</i> , suatu objek yang berisi logika aplikasi yang tidak memiliki tanggung jawab kepada entitas, contohnya adalah kalkulasi dan aturan bisnis yang melibatkan berbagai objek.
	<i>Message</i> , simbol mengirim pesan antar kelas.
	<i>Recursive</i> , menggambarkan pengiriman pesan yang dikirim untuk dirinya sendiri.
	<i>Activation</i> mewakili sebuah eksekusi operasi dari objek, panjang kotak ini berbanding lurus dengan durasi aktivitas sebuah operasi.

(Sumber : Ade Hendini, 2016)