

BAB II

TINJAUAN PUSTAKA

II.1. Penelitian Terkait

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

Menurut Felix Alfa Yudhistira, 2020, yang berjudul “ Analisis Penerapan Bunga Pinjaman Dengan Metode *Sliding Rate* Dan *Flat Rate* Pada Koperasi Kredit Gentiaras Pringsewu”. Tujuan dari penelitian ini yaitu Memahami perhitungan dan penerapan metode bunga pinjaman yang lebih menguntungkan bagi anggota dan koperasi bila menggunakan metode *sliding rate* di Kopdit Gentiaras Pringsewu serta Memahami perhitungan dan penerapan metode bunga pinjaman yang lebih menguntungkan bagi anggota dan koperasi bila menggunakan metode *flat rate* di Kopdit Gentiaras Pringsewu.

Menurut Muhtajuddin Danny, 2015, yang berjudul “ Sistem Informasi Pinjaman Dana Tunai Berbasis Web Dengan Menggunakan Database MySQL”. Tujuan dari penelitian ini yaitu menghasilkan Dengan menggunakan system informasi pinjaman dana tunai yang penulis usulkan di harapkan dapat menghemat waktu dan biaya serta dapat mempermudah proses operational serta perkembangan teknologi informasi dapat membantu dalam berbagai bidang, salah satunya di bidang finance.

Menurut Angga Aditya Permana, 2018, yang berjudul “ Rancangan Sistem Informasi Simpan Pinjam Pada Koperasi Guru Dan Pegawai Smp Negeri

45 Jakarta”. Tujuan dari penelitian ini yaitu dapat memberikan pengetahuan mengenai data-data yang dibutuhkan dengan akurat sehingga membantu dalam melakukan pengontrolan, mengurangi kesalahan pencatatan dan perhitungan yang sering dilakukan oleh manusia (*human error*), proses simpan pinjam dapat dilakukan dengan cepat dan tidak perlu lagi dilakukan secara manual, keamanan data akan lebih terjamin dan juga mempermudah dalam pencarian data serta mempercepat pengolahan data dan pembuatan laporan serta informasi yang dihasilkan lebih akurat, cepat dan lengkap.

II.2. Uraian Teoritis

II.2.1. Sistem

Ada dua kelompok pendekatan dalam mendefinisikan sistem. Ada yang menekankan pada prosedurnya dan ada yang menekankan pada komponen atau elemennya, diantaranya : Pendapat pertama menekankan sistem pada komponennya. “Sistem adalah kumpulan dari elemen-elemen yang berinteraksi untuk mencapai suatu tujuan tertentu”. Pendapat kedua menekankan sistem pada prosedurnya. “Sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk menyelesaikan suatu sasaran tertentu”.” (Fanny Andalia : 2015).

Sistem adalah kumpulan dari elemen – elemen yang berinteraksi untuk mencapai suatu tujuan tertentu (Jogiyanto,2005: 2), sedangkan menurut Kadir (2003: 54), Sistem adalah sekumpulan elemen yang saling terkait atau terpadu yang dimaksudkan untuk mencapai suatu tujuan. Berdasarkan pendapat para ahli

tersebut maka dapat disimpulkan sistem adalah kumpulan elemen-elemen yang saling bekerja sama dan berinteraksi untuk memproses masukan kemudian saling berhubungan untuk mencapai suatu sasaran tertentu. (Rudi Hermawan : 2016)

Sebuah sistem terdiri dari berbagai unsur yang saling melengkapi dalam mencapai tujuan dan sasaran. Unsur-unsur yang terdapat dalam sistem itulah yang disebut dengan subsistem. Subsistem-subsistem tersebut harus saling berhubungan dan berinteraksi melalui komunikasi yang relevan sehingga sistem dapat bekerja secara efektif dan efisien. (Eka Iswandy : 2015).

Suatu sistem mempunyai karakteristik atau sifat yang tertentu, yaitu mempunyai komponen-komponen (*component*), batas sistem (*boundary*), lingkungan luar sistem (*environment*), penghubung (*interface*), masukan (*input*), keluaran (*output*), pengolah (*proses*), sasaran (*objectives*), ataupun tujuan (*goal*). Adapun Karakteristik dari sistem adalah (Tomi Loveri : 2018) :

a. Komponen Sistem (*Components*)

Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, yang bekerja sama membentuk satu kesatuan. Komponen-komponen sistem tersebut dapat berupa suatu bentuk subsistem.

b. Batasan Sistem (*Boundary*)

Ruang lingkup sistem merupakan daerah yang membatasi antara sistem dengan sistem lainnya atau sistem dengan lingkungan luarnya. Batasan sistem ini memungkinkan suatu sistem dipandang sebagai satu kesatuan yang tidak dapat dipisah-pisahkan.

c. Lingkungan Luar Sistem (*Environment*)

Lingkungan luar sistem adalah bentuk apapun yang ada di luar ruang lingkup atau batasan sistem yang mempengaruhi operasi sistem tersebut.

d. Penghubung Sistem (*Interface*)

Penghubung sistem atau *interface* adalah media yang menghubungkan sistem dengan subsistem yang lain.

e. Masukan Sistem (*Input*)

Energi yang dimasukkan ke dalam sistem disebut masukan sistem, yang dapat berupa pemeliharaan (*maintenance input*) dan sinyal (*signal input*).

f. Keluaran Sistem (*Output*)

Keluaran ini merupakan masukan bagi subsistem yang lain. Seperti contoh sistem informasi, keluaran yang dihasilkan adalah informasi, di mana informasi ini dapat digunakan sebagai masukan untuk pengambilan keputusan atau hal-hal lain yang merupakan *input* bagi subsistem lain.

g. Pengolah Sistem (*Process*)

Suatu sistem dapat mempunyai suatu proses yang akan mengubah masukan menjadi keluaran.

h. Sasaran Sistem (*Objective*)

Suatu sistem memiliki tujuan dan sasaran yang pasti dan bersifat deterministik.

II.2.2. Informasi

Di dalam pengolahan sistem pada akhirnya menghasilkan suatu informasi, untuk itu pendefinisian informasi diperlukan untuk menunjang berhasilnya pengembangan sistem yang akan dirancang. Informasi adalah data yang dapat diolah yang lebih berguna dan berarti bagi yang menerimanya. (Rini Asmara : 2016).

Informasi adalah data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi penerimanya. Sumber informasi adalah data. Data kenyataannya yang menggambarkan suatu kejadian – kejadian dan kesatuan nyata. Kejadian – kejadian (*event*) adalah kejadian yang terjadi pada saat tertentu. Menurut Gordon B Davis (2015:8) Informasi adalah data yang telah diolah menjadi suatu bentuk yang penting bagi si penerima dan mempunyai nilai nyata yang dapat dirasakan dalam keputusan – keputusan yang sekarang atau keputusan – keputusan yang akan datang. (Rini Asmara : 2016)

II.2.3. Sistem Informasi

Sistem informasi merupakan sistem yang disusun secara sistematis dan teratur dari jaringan-jaringan aliran informasi yang menghubungkan setiap bagian dari suatu sistem, sehingga memungkinkan diadakannya komunikasi antar bagian atau satuan fungsional. Berikut pengertian sistem informasi menurut para ahli adalah :

James O'Brien sistem informasi adalah suatu kombinasi teratur apapun dari *people* (orang), *hardware* (perangkat keras), *software* (perangkat lunak),

computer networks and data communications (jaringan komunikasi), dan database (basis data) yang mengumpulkan, mengubah dan menyebarkan informasi didalam suatu bentuk organisasi.”

Loudon Sistem informasi adalah komponen-komponen yang saling berhubungan dan bekerjasama untuk mengumpulkan, memproses, menyimpan dan mendistribusikan informasi untuk mendukung pengambilan keputusan, koordinasi, kontrol, analisis dan visualisasi dalam suatu organisasi.”. (Agus Irawan : 2016).

Sistem informasi merupakan sistem yang disusun secara sistematis dan teratur dari jaringan-jaringan aliran informasi yang menghubungkan setiap bagian dari suatu sistem, sehingga memungkinkan diadakannya komunikasi antar bagian atau satuan fungsional. (Agus Irawan : 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 (Zaenal Sekty Wijaya:2015).

II.2.3.1 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, temperatur, 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 : 2018)

II.2.4. Pinjaman

Pinjaman (kredit) dana tunai adalah pinjaman uang dengan pembayaran pengembalian secara mengangsur. (Menurut Kamus Besar Bahasa Indonesia). Secara yuridis bahkan Undang-undang Perbankan no 7 tahun 1992 mendefinisikan secara lugas bahwa Kredit adalah penyediaan uang atau tagihan yang dapat dipersamakan dengan itu, berdasarkan persetujuan dan kesepakatan pinjam meminjam antara suatu perusahaan dengan pihak lain yang mewajibkan pihak peminjam untuk melunasi hutangnya setelah jangka waktu tertentu dengan

jumlah uang, imbalan atau pembagian hasil keuntungan.. (Muhtajuddin Danny : 2015)

II.2.5. Metode *Flat Rate*

Bunga *flat rate* adalah metode pembebanan di mana bunga dan besarnya pokok pinjaman yang dibayar tetap sama pada setiap bulan sampai kredit tersebut lunas. Berikut rumusnya :. (Lutfiyah Dwi Setia : 2018)

Berikut adalah cara perhitungannya :

$$\text{Angsuran Pokok} = \text{Jumlah Pinjaman} / \text{Jangka Waktu} \dots \dots \dots (1)$$

Menghitung bunga pinjaman per bulan, sebagai berikut :

$$BG = \frac{BUNGA \times \text{NOMINAL PINJAMAN}}{12 \text{ BULAN}} \times 1 \dots \dots \dots (2)$$

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 (Benardo, 2015).

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).

Menurut Sutabri (2016) dalam buku Sistem Informasi Manajemen: ‘*Database* adalah suatu kumpulan data terhubung (*interrelated data*) yang disimpan secara bersama-sama pada suatu media, tanpa mengatap satu sama lain atau tidak perlu suatu kerangkapan data (*controlled redundancy*’. (Fitri Ayu : 2018)

Beberapa model *Database* diantaranya :

1. *Object based* data model merupakan himpunan data dan prosedur atau relasi yang menjelaskan hubungan logis antar data dalam suatu *database* berdasarkan objek datanya.
2. *Record Based* data model. Model ini berdasarkan pada *record* untuk menjelaskan kepada *user* tentang hubungan logis antardata dalam *database*.

II.2.7. 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. (Herpendi, 2016).

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*). (Herpendi, 2016).

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, 2016).

II.2.8. 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. (Benardo, 2015).

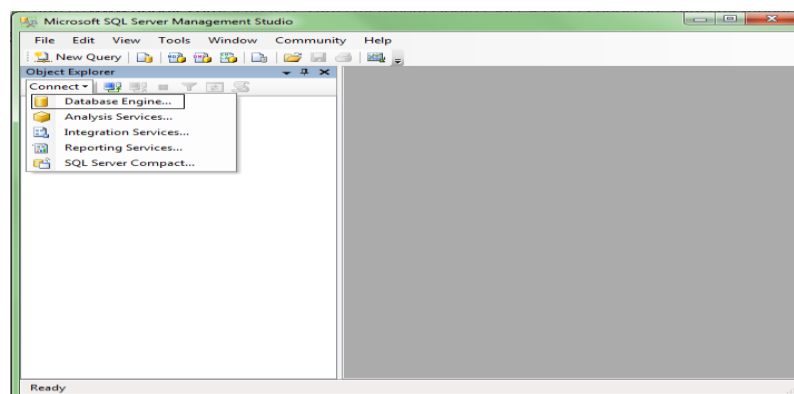
Database merupakan suatu tempat untuk menyimpan data. Pada sebuah *database* bisa terdapat satu atau lebih tabel dan *query*. Operasi yang biasanya dilakukan *database* berhubungan erat dengan pengaksesan tabel atau *query* Pada *SQL Server 2008* terdapat fitur-fitur yang dapat mengembangkan performa dari *database* tersebut. Beberapa fitur tersebut, yaitu (Benardo, 2015) :

- 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.

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 : 2017).



Gambar II.1. Tampilan SQL Server
(Sumber : Hendra Kurniawan : 2017)

II.2.9. 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.2.9.1. *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
	<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> .
	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

	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.2.9.2. 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



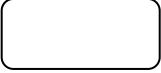
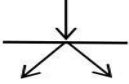
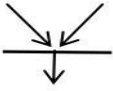
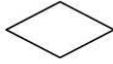

Multiplicity	Penjelasan
1	Satu dan hanya satu
0..*	Boleh tidak ada atau 1 atau lebih
1..*	1 atau lebih
0..1	Boleh tidak ada maksimal 1
n..n	Batasan antara Contoh 2..4 mempunyai arti minimal 2 maksimum 4

(Sumber : Ade Hendini, 2016)

II.2.9.3. 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




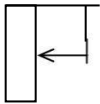


Gambar	Keterangan
	<i>Start Point</i> , diletakkan pada pojok kiri atas dan merupakan awal aktifitas.
	<i>End Point</i> , akhir aktifitas
	<i>Activities</i> menggambarkan suatu proses atau kegiatan bisnis.
	<i>Fork</i> (Percabangan) digunakan untuk menunjukkan kegiatan yang dilakukan secara parallel 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.2.9.4. 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 cetak</i> .
	<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)