

BAB II

TINJAUAN PUSTAKA

II.1. Penelitian Terdahulu

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

Berdasarkan penelitian yang dilakukan oleh Adi Wahyu Pribadi (2016) dengan judul “Perancangan Sistem Informasi Administrasi Karyawan Radio” Dalam penelitian ini penulis menggunakan metode waterfall, observasi dan wawancara. Pada implementasi ditampilkan proses login, penginputan : data karyawan, absensi, pengajuan cuti, pengunduran diri, klarifikasi surat peringatan dan PHK, perubahan gaji, pengembangan skill, penilaian kinerja. Dapat disimpulkan bahwa sistem ini menangani semua pencatatan master karyawan, perubahan gaji berdasarkan penilaian kinerja karyawan dan UMR, pengunduran diri karyawan dan pencatatan dan pengklasifikasian SP dan PHK yang akan diberikan kepada karyawan, sehingga HRD, R&D, Finance & Administration dan Manager Perusahaan dapat mendapatkan informasi dan laporan dengan mudah.

Berdasarkan penelitian yang dilakukan Akbar A Poipessy (2018) dengan judul “Pembuatan Aplikasi Jadwal Kerja Karyawan Berbasis Web Pada Stasiun Pengisian Bahan Bakar Umum (Spbu) Kalumata Ternate” Pertamina SPBU Kalumata Kota Ternate dalam mengelola data karyawan, absen dan jadwal karyawan masih dikelola menggunakan Aplikasi Microsoft Excel. Dimana data-data tersebut diinput dan dikelola pada setiap sheet, kekurangannya disaat admin

ingin mencari data karyawan dan data jadwal atau ingin membuat laporan harus membuka dan memproses satu persatu pada sheet dan tidak secara otomatis dikelola pada satu sheet atau satu halaman saja, disisi lain data-data tersebut juga tidak tersimpan pada database sehingga suatu saat data tersebut bisa hilang dan terhapus.

Berdasarkan penelitian yang dilakukan Setiawansyah (2021) dengan judul “Perancangan Sistem Informasi Akuntansi Upah Lembur Karyawan Menggunakan Extreme Programming”, Bagian staff dalam menghitung upah lembur karyawan harus melihat surat perintah kerja lembur (SPKL) karyawan secara satu persatu sehingga membutuhkan waktu yang lama, lalu menginputkan jam lembur karyawan, gaji pokok, menghitung jam kerja lembur secara manual, serta menghitung pembayaran upah lembur dan total upah lembur, hal ini membutuhkan ketelitian sehingga jika terjadi kesalahan akan merugikan karyawan dan perhitungan upah lembur akan dihitung ulang.

Berdasarkan penelitian dari Kapung Rizki Siregar (2019) dengan judul “Perancangan Sistem Reporting Kerja Karyawan Berbasis Web Pada Ruangan Network Operation Center Bagian Front Office (Studi Kasus: Network Operation Center Xl Axiata)” Sistem reporting karyawan yang dirancang nantinya akan digunakan mengubah sistem reporting yang ada di NOC yang semula melalui pengiriman email menjadi berbasis web. Dimana website ini akan memiliki fitur untuk mengarsipkan seluruh data pekerjaan karyawan berupa report pekerjaan yang sudah selesai dan juga report pekerjaan yang masih dalam proses. Dalam perancangan sistem reporting karyawan berbasis web ini metode pengumpulan

data menggunakan teknik wawancara dan observasi. Metode yang digunakan adalah dengan model pengembangan Waterfall melalui tahapan analysis dan design.

Berdasarkan penelitian dari Fenny Oktavia (2019) dengan judul “Perancangan Sistem Informasi Penggajian Karyawan Pada PT. Sawmill Jambi” penelitian ini bertujuan memberikan solusi untuk permasalahan yang terjadi dengan menawarkan sistem informasi penggajian karyawan menggunakan bahasa pemrograman PHP dan DBMS MySQL dimana penulis melakukan pengembangan sistem dengan metode waterfall dan menggunakan pendekatan model sistem unified model language menggunakan usecase diagram, activity diagram, class diagram dan flowchart. Sistem baru menghasilkan output yang menampilkan data karyawan, data tunjangan, data bonus, data absensi yang secara terstruktur dan menampilkan laporan-laporan yang diperlukan untuk memudahkan PT. Sawmill Jambi dalam pengolahan data.

II.2. Landasan Teori

II.2.1. Perancangan Sistem

Perancangan sistem dapat didefinisikan yaitu merumuskan bagaimana cara membentuk suatu struktur. Perancangan sistem juga dapat diartikan sebagai berikut :

Tahap setelah analisis dari siklus pengembangan sistem.

1. Pendefinisian dari kebutuhan-kebutuhan fungsional.
2. Persiapan untuk rancang bangun implementasi.

3. Menggambarkan bagaimana suatu sistem dibentuk.
4. Yang dapat berupa penggambaran, perencanaan dan pembuatan sketsa atau pengaturan dari beberapa elemen yang terpisah ke dalam suatu kesalasan yang utuh dan berfungsi.

Termasuk menyangkut mengkonfigurasi dari komponen-komponen perangkat lunak dan perangkat keras dari suatu sistem. (Andy Victor, 2018).

II.2.2. Sistem

Sistem dapat terdiri dari sistem-sistem bagian (*subsystem*). Sebagai misal, sistem komputer dapat terdiri dari subsistem perangkat keras dan subsistem perangkat lunak. Masing-masing subsistem dapat terdiri dari subsistem-subsistem yang lebih kecil atau terdiri dari komponen-komponen. Subsistem perangkat keras (*hardware*) dapat terdiri dari alat masukan, alat pemroses, alat keluaran dan simpanan luar, dan kemudian subsistem-subsistem tersebut akan berinteraksi sedemikian rupa sehingga dapat mencapai satu kesatuan yang terpadu. Dalam buku Analisa dan Design Sistem Informasi Pendekatan Terstruktur Teori dan Praktek Aplikasi Bisnis. Sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk menyelesaikan suatu sasaran yang tertentu (M. Syariat, 2020)

II.2.2.1. Karakteristik Sistem

Sistem mempunyai karakteristik atau sifat-sifat tertentu, diantaranya adalah sebagai berikut :

1. Komponen Sistem (*Components*)

Sistem terdiri dari sejumlah komponen yang saling berinteraksi, yang artinya saling bekerja sama membentuk suatu kesatuan. Komponen-komponen sistem dapat berupa suatu subsistem atau bagian-bagian dari sistem.

2. Batas Sistem (*Boundary*)

Batas sistem merupakan daerah yang membatasi antara sistem dengan sistem yang lainnya atau dengan lingkungan luarnya. Batas sistem ini memungkinkan suatu sistem dipandang sebagai suatu kesatuan. Batas sistem menunjukkan ruang lingkup (*scope*) dari sistem tersebut.

3. Lingkungan Luar Sistem (*Environments*)

Lingkungan luar dari suatu sistem adalah apapun diluar batas dari sistem yang mempengaruhi operasi sistem. Lingkungan luar sistem dapat bersifat menguntungkan dan dapat juga merugikan suatu sistem.

4. Penghubung Sistem (*Interface*)

Penghubung merupakan media penghubung antara suatu subsistem dengan subsistem yang lainnya. Melalui penghubung ini memungkinkan sumber-sumber daya mengalir dari suatu subsistem ke subsistem lainnya.

5. Masukan Sistem (*Input*)

Masukan adalah energi yang dimasukkan ke dalam sistem. Masukan dapat berupa masukan perawatan (*maintenance input*) dan masukan sinyal (*signal input*).

6. Keluaran sistem (*Output*)

Keluaran adalah hasil energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna. Keluaran dapat berupa masukan untuk subsistem yang lain.

7. Pengolah Sistem (*Process*)

Sistem dapat mempunyai suatu bagian pengolah atau sistem itu sendiri sebagai pengolah. Pengolah yang akan merubah masukan menjadi keluaran.

8. Sasaran Sistem (*Objectives*)

Sistem mempunyai tujuan atau sasaran yang akan menentukan masukan yang dibutuhkan sistem dan keluaran yang akan dihasilkan sistem. (M. Syariat, 2020)

II.2.3. Informasi

Informasi adalah data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi yang menerimanya.

- (1) Akurat (*Accurate*) Berarti informasi harus bebas dari kesalahan-kesalahan dan tidak bias atau menyesatkan. Akurat juga berarti informasi harus jelas mencerminkan maksudnya.
- (2) Tepat waktu (*Timelines*) Berarti informasi yang datang pada si penerima tidak boleh terlambat. Informasi yang sudah usang tidak akan mempunyai nilai logika karena informasi merupakan landasan dalam pengambilan keputusan.
- (3) Relevan (*Relevance*) Berarti informasi tersebut bermanfaat bagi pemakainya. (Fajar Ramadhan, 2020).

II.2.4. Sistem Informasi

Sistem informasi adalah sekumpulan komponen-komponen sistem yang berada didalam suatu ruang lingkup organisasi, saling berinteraksi untuk menghasilkan sebuah informasi yang bertujuan untuk pihak manajemen tertentu dan untuk mencapai tujuan tertentu. Sistem informasi adalah menggambarkan campuran dari empat faktor pokok keempat komponen utama tersebut mencakup perangkat keras (*hardware*), perangkat lunak (*software*) prasarana dan sumber daya manusia (SDM) yang berkompeten. Sementara implementasi sistem informasi bisa berbentuk sebuah mainframe, sebuah server dari kebanyakan komputer maupun hosting dalam internet di suatu komputer *server*. (Fajar Ramadhan, 2020).

Faktor-faktor yang menentukan kehandalan dari suatu sistem informasi atau informasi dapat dikatakan baik jika memenuhi kriteria-kriteria sebagai berikut :

1. Keunggulan (*Usefulness*)

Yaitu suatu sistem yang harus dapat menghasilkan informasi yang tepat dan relevan untuk mengambil keputusan manajemen dan personil operasi dalam organisasi.

2. Ekonomis

Kemampuan sistem yang mempengaruhi sistem harus bernilai manfaat minimal, sebesar biayanya.

3. Keandalan (*Reliability*)

Keluaran dari sistem harus mempunyai tingkat ketelitian tinggi dan sistem tersebut harus beroperasi secara efektif.

4. Pelayanan (*Customer Service*)

Yakni suatu sistem memberikan pelayanan yang baik dan efisien kepada para pengguna sistem pada saat berhubungan dengan organisasi.

5. Kapasitas (*Capacity*)

Setiap sistem harus mempunyai kapasitas yang memadai untuk menangani setiap periode sesuai yang dibutuhkan.

6. Sederhana dalam kemudahan (*Simplicity*)

Sistem tersebut lebih sederhana (umum) sehingga struktur dan operasinya dapat dengan mudah dimengerti dan prosedur mudah diikuti.

7. Fleksibel (*Fleksibility*)

Sistem informasi ini harus dapat digunakan dalam kondisi yang bagaimana yang diinginkan oleh organisasi tersebut atau pengguna tertentu.

8. Komponen Sistem Informasi

Istilah dalam komponen sistem informasi adalah blok bangunan (building block) yang dapat di bagi menjadi enam blok yaitu :

a. Blok masukan (*Input block*)

Blok *input* merupakan data-data yang masuk ke dalam sistem informasi, yang dapat berupa *document-document* dasar yang dapat diolah menjadi suatu informasi tertentu.

b. Blok model (*Model block*)

Blok ini terdiri dari kombinasi prosedur, logika dan model matematik yang akan mengolah data *input* untuk menghasilkan suatu informasi yang dibutuhkan.

c. Blok keluaran (*Output block*)

Merupakan informasi yang menghasilkan sekumpulan data yang nantinya akan disimpan berupa data cetak laporan.

d. Blok teknologi (*Technologi block*)

Blok teknologi merupakan penunjang utama dalam berlangsungnya sistem informasi. Yang memiliki beberapa komponen yaitu diantaranya alat memasukkan data (*input device*), alat untuk menyimpan dan mengakses data (*storege device*), alat untuk menghasilkan dan mengirimkan keluaran (*output divice*) dan alat untuk membentuk pengendalian sistem secara keseluruhan (*control device*). Teknologi informasi terdiri dari 3 (tiga) bagian utama, yaitu teknisi (*humanware* atau *braiware*), perangkat lunak (*software*), dan perangkat keras (*hardware*).

e. Blok basis data (*Database block*)

Basis data merupakan kumpulan dari data yang saling berhubungan satu dengan yang lainnya, tersimpan di perangkat keras komputer dan digunakan perangkat lunak untuk memanipulasinya. Data perlu di simpan dan perlu di organisasi sedemikian rupa, supaya informasi yang dihasilkan berkualitas.

f. Blok kendali (*Control block*)

Beberapa pengendalian perlu dirancang dan diterapkan untuk meyakinkan bahwa hal-hal yang dapat merusak sistem dapat di cegah bila terlanjur terjadi. (Fajar Ramadhan, 2020).

II.2.5. Karyawan

Karyawan dalam perusahaan atau lembaga merupakan aset yang sangat di butuhkan dan penting, oleh karena itu dalam perekrutan karyawan harus lebih selektif agar mendapatkan karyawan yang berkualitas dan dapat menunjang kinerja di dalam perusahaan. Tanpa adanya karyawan maka perusahaan tidak dapat melakukan proses produksi atau prose yang lainnya. Karyawan bisa disebut juga sebagai seseorang yang menjual jasa (pikiran dan tenaga) supaya mendapatkan kompensasi atau gaji yang besarnya telah ditetapkan terlebih dahulu oleh perusahaan (Rahmat Agusli, dkk : 2020).

II.2.6. Buruh

Buruh berupa orang-orang yang bekerja pada suatu tempat, pekerja tersebut harus tunduk kepada perintah dan peraturan kerja yang diadakan oleh pengusaha (majikan) yang bertanggung jawab atas lingkungan perusahaannya yang kemudian atas pekerjaannya pekerja tersebut akan memperoleh upah dan atau jaminan hidup lainnya yang layak. Hal ini didasarkan kerena adanya hubungan kerja antara pekerja/buruh dengan pengusaha (majikan). Suatu hubungan antara seorang buruh dengan seorang majikan, hubungan kerja hendak menunjukkan

kedudukan kedua pihak itu yang pada dasarnya menggambarkan hak-hak dan kewajiban-kewajiban buruh terhadap majikan serta hak-hak dan kewajiban-kewajiban majikan terhadap buruh. (Anju Mastaurat Natanael: 2022).

II.2.7. Overshift

Shift kerja mempunyai berbagai definisi tetapi biasanya shift kerja disamakan dengan pekerjaan yang dibentuk di luar jam kerja biasa (08.00-17.00). Ciri khas tersebut adalah kontinuitas, pergantian dan jadwal kerja khusus. Secara umum yang dimaksud dengan shift kerja adalah semua pengaturan jam kerja, sebagai pengganti atau tambahan kerja siang hari maupun malam hari sebagaimana yang biasa dilakukan. Namun demikian adapula definisi yang lebih operasional dengan menyebutkan jenis shift kerja tersebut. Sistem shift merupakan suatu sistem pengaturan kerja yang memberi peluang untuk memanfaatkan keseluruhan waktu yang tersedia untuk mengoperasikan pekerjaan. shift kerja merupakan pola waktu kerja yang diberikan pada tenaga kerja untuk mengerjakan sesuatu oleh perusahaan dan biasanya dibagi atas kerja pagi, sore dan malam. Proporsi pekerja shift semakin meningkat dari tahun ke tahun, ini disebabkan oleh investasi yang dikeluarkan untuk pembelian mesin-mesin yang mengharuskan penggunaannya secara terus menerus siang dan malam untuk memperoleh hasil yang lebih baik. Sebagai akibatnya pekerja juga harus bekerja siang dan malam. Hal ini menimbulkan banyak masalah terutama bagi tenaga kerja yang tidak atau kurang dapat menyesuaikan diri dengan jam kerja yang lazim. (Poniah Juliawati : 2020)

II.2.8 Website

Website adalah halaman informasi yang disediakan melalui jalur internet dan memiliki fasilitas hiperteks untuk menampilkan data berupa teks, gambar, suara, animasi, dan data multimedia lain nya sehingga bisa diakses di seluruh dunia selama terkoneksi dengan jaringan internet. *Website* dapat diartikan sekumpulan halaman situs yang saling berhubungan menampilkan berbagai macam informasi maupun data berupa teks, gambar diam ataupun bergerak, data animasi, suara, video maupun gabungan dari semuanya yang bisa diakses dimana saja kapan saja diseluruh dunia. Dimana *website* dikategorikan menjadi dua, yaitu:

1. *Web Statis*

Adalah web yang berisi/menampilkan informasi-informasi yang sifatnya *statis* (tetap). Disebut statis karena pengguna tidak dapat berinteraksi dengan *web* tersebut. Pengguna hanya dapat melihat isi dokumen pada halaman web dan apabila diklik akan berpindah kehalaman web yang lain. Interaksi pengguna terbatas hanya melihat informasi yang ditampilkan, tetapi tidak bisa mengolah informasi yang dihasilkan. Biasanya merupakan *HTML* yang ditulis pada editor teks dan disimpan dalam bentuk *.html* atau *.htm*.

2. *Web Dinamis*

Adalah *web* yang menampilkan informasi serta dapat berinteraksi dengan pengguna. *Web* yang dinamis memungkinkan pengguna untuk berinteraksi dengan *form* sehingga dapat mengolah informasi yang ditampilkan.

Bersifat teraktif, tidak kaku, dan terlihat lebih indah. (Andi Ridho Rachman, 2018).

II.2.5 Basis Data (*Database*)

Basis data atau *database* merupakan koleksi dari data-data yang terorganisir dengan rapi sehingga data dapat dengan mudah disimpan dan dimanipulasi. Salah satu tujuan dari *database* adalah memberikan pengguna suatu pandangan abstrak dari data, yaitu sistem menyembunyikan rincian bagaimana data disimpan dan dipelihara. Sistem *database* harus dibuat semudah mungkin untuk dimengerti karena kebanyakan pengguna sistem *database* adalah orang-orang yang kurang terlatih di bidang teknologi. (Ni Kadek Ceryna : 2018 : 102)

II.2.11. XAMPP

XAMPP adalah perangkat lunak bebas, yang mendukung banyak sistem operasi, merupakan kompilasi dari beberapa program. Fungsinya adalah sebagai server yang berdiri sendiri (*localhost*), yang terdiri atas program *Apache HTTP Server*, *MySQL database*, dan penerjemah bahasa yang ditulis dengan bahasa pemrograman *PHP* dan *Perl*. Nama *XAMPP* merupakan singkatan dari X (empat sistem operasi apapun), *Apache*, *MySQL*, *PHP* dan *Perl*. Program ini tersedia dalam *GNU General Public License* dan bebas, merupakan *web server* yang mudah digunakan yang dapat melayani tampilan halaman *web* yang dinamis. Untuk mendapatkannya dapat men-*download* langsung dari *web* resminya. (Randi Palit, 2018).

II.2.12. MySQL

MySQL (*My Structure Query Language*) merupakan sebuah program pembuat database yang bersifat *Open Source*, artinya semua orang dapat menggunakannya dan dapat dijalankan pada semua *platform* baik *Windows* maupun *linux*. MySQL juga merupakan sebuah perangkat lunak sistem manajemen basis data *SQL* yang bersifat jaringan sehingga dapat digunakan untuk aplikasi multi *user*. MySQL juga sering dikenal dengan nama sistem manajemen *database* relasional. Suatu *database* relasional menyimpan data dalam *table* yang terpisah. *Tabel -table* tersebut terhubung oleh suatu relasi terdefinisi yang memungkinkan memperoleh kombinasi data dari beberapa *table* dalam suatu permintaan. Untuk administrasi *database*, seperti pembuatan *database*, pembuatan *table*, dan sebagainya dapat digunakan aplikasi berbasis web seperti *PHP MyAdmin* dengan aplikasi *XAMPP*. (Muhammad Faisal Widad : 2018)

II.2.13.PHP

PHP atau kependekan dari *Hypertext Preprocessor* adalah salah satu bahasa pemrograman *open source* yang sangat cocok atau dikhususkan untuk pengembangan *web* dan dapat ditanamkan pada sebuah skripsi *HTML*. Bahasa PHP dapat dikatakan menggambarkan beberapa bahasa pemrograman seperti *C*, *Java*, dan *Perl* serta mudah untuk dipelajari. PHP merupakan bahasa *scripting server side*, dimana pemrosesan datanya dilakukan pada sisi *server*. Sederhananya, *server* yang akan menterjemahkan skrip program, baru kemudian hasilnya akan dikirim kepada *client* yang melakukan permintaan. Adapun

pengertian lain PHP adalah *akronim* dari *Hypertext Preprocessor*, yaitu suatu bahasa pemrograman berbasis kode – kode (*script*) yang digunakan untuk mengolah suatu data dan mengirimkannya kembali ke *web browser* menjadi kode HTML”. PHP (singkatan rekursif: *PHP Hypertext Preprocessor*) merupakan script yang dapat disisipkan ke dalam HTML. PHP dipakai untuk membuat sistem berbasis web yang dinamis. PHP bahasa pemrograman yang *server-side* sehingga program tersebut akan dijalankan atau diproses oleh server . Hingga saat ini PHP sudah rilis versi 7.4.1 pada 18 desember 2019. (Roberto kaban : 2019)

II.2.14. UML (*Unified Modelling Language*)

Unified Modelling Language (UML) merupakan satu kumpulan konvensi pemodelan yang digunakan untuk menentukan atau menggambarkan sebuah sistem *software* yang terkait dengan objek. UML merupakan salah satu alat bantu yang sangat handal dalam bidang pengembangan sistem berorientasi objek karena UML menyediakan Bahasa pemodelan *visual* yang memungkinkan pengembang sistem membuat *blue print* atas visinya dalam bentuk yang baku. UML berfungsi sebagai jembatan dalam mengkomunikasikan beberapa aspek dalam sistem melalui jumlah elemen grafis yang bisa dikombinasikan menjadi.

Unified Modeling Language (UML) biasa digunakan untuk :

- a. Menggambarkan batasan sitem dan fungsi - fungsi sistem secara umum, dibuat dengan *use case* dan *actor*.
- b. Menggambarkan kegiatan atau proses bisnis yang dilaksanakan secara umum, dibuat dengan *interaction diagrams*.

- c. Menggambarkan representasi struktur *static* sebuah sistem dalam bentuk *class diagrams*.
- d. Membuat model behavior “yang menggambarkan kebiasaan atau sifat sebuah sistem” dengan *state transition diagrams*.
- e. Menyatakan arsitektur implementasi fisik menggunakan *component and development*.
- f. Menyampaikan atau memperluas *fungsi* dengan *stereotypes*. (Omni Alfina dan Fitriana Harahap : 2019)

UML merupakan singkatan dari *Unified Modeling Language* yang merupakan sekumpulan alat yang digunakan untuk melakukan abstraksi terhadap sebuah sistem atau perangkat lunak berbasis objek. UML juga menjadi salah satu cara untuk mempermudah pengembangan aplikasi yang berkelanjutan. Aplikasi atau sistem yang tidak terdokumentasi biasanya dapat menghambat pengembangan karena *developer* harus melakukan penelusuran dan mempelajari kode program. UML juga dapat menjadi alat bantu untuk transfer ilmu tentang sistem atau aplikasi yang akan dikembangkan dari satu *developer* ke *developer* lainnya. (Janiver W. Janis : 2020)

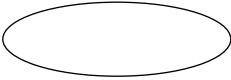
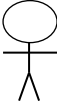
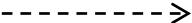
1. Use case Diagram

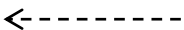
Use case diagram merupakan pemodelan untuk melakukan (*behavior*) sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Dapat dikatakan *use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut.

Simbol-simbol yang digunakan dalam *use case* diagram dapat dilihat pada tabel

II.1 dibawah ini:

Tabel II.1. Simbol Use Case

Gambar	Keterangan
	<i>Use case</i> menggambarkan fungsionalitas yang disediakan sistem sebagai unit-unit yang bertukar pesan antar unit dengan aktor, biasanya dinyatakan dengan menggunakan kata kerja di awal nama <i>use case</i> .
	Aktor adalah <i>abstraction</i> dari orang atau sistem yang lain yang mengaktifkan fungsi dari target sistem. Untuk mengidentifikasi aktor, harus ditentukan pembagian tenaga kerja dan tugas-tugas yang berkaitan dengan peran pada konteks target sistem. Orang atau sistem bisa muncul dalam beberapa peran. Perlu dicatat bahwa aktor berinteraksi dengan <i>use case</i> , tetapi tidak memiliki control 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 mengidikasikan aliran data.
	Asosiasi antara aktor dan <i>use case</i> yang menggunakan panah terbuka untuk mengidinkasikan bila aktor berinteraksi secara pasif dengan sistem.
	<i>Include</i> , merupakan di dalam <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.

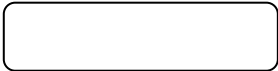
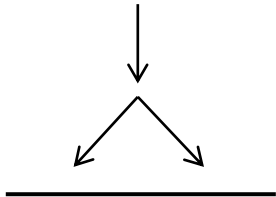
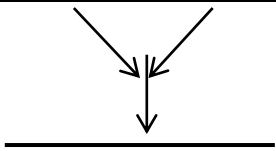
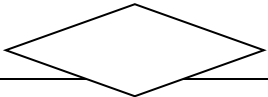
	<i>Extend</i> , merupakan perluasan dari <i>use case</i> lain jika kondisi atau syarat terpenuhi.
---	---

(Sumber : Janiver W. Janis : 2020)

2. Diagram Aktivitas (*Activity Diagram*)

Activity Diagram menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis. Simbol-simbol yang digunakan dalam *activity diagram* dapat dilihat pada tabel II.2 dibawah ini:

Tabel II.2. Simbol *Activity Diagram*

Gambar	Keterangan
	<i>Start point</i> , diletakkan pada pojok kiri atas dan merupakan awal aktifitas.
	<i>End point</i> , akhir aktifitas.
	<i>Activites</i> , menggambarkan suatu proses/kegiatan bisnis.
	<i>Fork</i> (Percabangan), digunakan untuk menunjukkan kegiatan yang dilakukan secara parallel atau untuk menggabungkan dua kegiatan pararel menjadi satu.
	<i>Join</i> (penggabungan) atau rake, digunakan untuk menunjukkan adanya dekomposisi.
	<i>Decision Points</i> , menggambarkan pilihan untuk

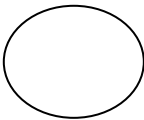
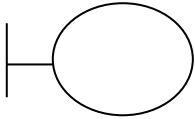
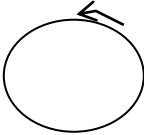
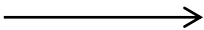
	pengambilan keputusan, <i>true</i> , <i>false</i> .
New Swimlane	<i>Swimlane</i> , pembagian <i>activity</i> diagram untuk menunjukkan siapa melakukan apa.

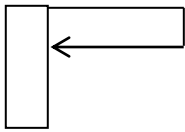
(Sumber : Janiver W. Janis : 2020)

3. Diagram Urutan (*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* dapat dilihat pada tabel II.3 dibawah ini :

Tabel II.3. Simbol *Sequence Diagram*

Gambar	Keterangan
	<i>Entity Class</i> , merupakan bagian dari sistem yang berisi kumpulan kelas berupa entitas-entitas yang membentuk gambaran awal sistem 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 formentry 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 <i>class</i> .

	<i>Recursive</i> , menggambarkan pengiriman pesan yang dikirim untuk dirinya sendiri.
	<i>Activation</i> , <i>activation</i> mewakili sebuah eksekusi operasi dari objek, panjang kotak ini berbanding lurus dengan durasi aktivitas sebuah operasi.
	<i>Lifeline</i> , garis titik-titik yang terhubung dengan objek, sepanjang <i>lifeline</i> terdapat <i>activation</i> .

(Sumber : Janiver W. Janis : 2020)

4. Class Diagram (Diagram Kelas)

Merupakan hubungan antar kelas dan penjelasan detail tiap-tiap kelas di dalam model desain dari suatu sistem, juga memperlihatkan aturan-aturan dan tanggung jawab entitas yang menentukan perilaku sistem. *Class diagram* juga menunjukkan atribut-atribut dan operasi-operasi dari sebuah kelas dan *constraint* yang berhubungan dengan objek yang dikoneksikan. *Class diagram* secara khas meliputi: Kelas (*Class*), Relasi, *Associations*, *Generalization* dan *Aggregation*, Atribut (*Attributes*), Operasi (*Operations/ Method*), *Visibility*, tingkat akses objek eksternal kepada suatu operasi atau atribut. Hubungan antar kelas mempunyai keterangan yang disebut dengan *multiplicity* atau kardinaliti yang dapat dilihat pada tabel II.4 dibawah ini :

Tabel II.4. Multiplicity 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 : Janiver W. Janis : 2020)