BAB II

LANDASAN TEORI

2.1 Konsep Dasar Sistem

Pada umumnya setiap organisasi mempunyai sistem informasi dalam mengumpulkan, menyimpan, melihat, dan menyalurkan informasi dalam membuat rancangan sistem informasi.

Menurut (Anggraeni & Rita, 2017) suatu sistem pada dasarnya adalah sekelompok unsur yang erat hubungannya satu dengan yang lain, yang berfungsi bersama-sama untuk mencapai tujuan tertentu. Secara sederhana, suatu sistem dapat diartikan sebagai suatu kumpulan atau himpunan dari unsur, komponen, atau variable yang teroganisasi, saling berinteraksi, saling tergantung satu sama lain, dan terpadu.

A. Model Pembelajaran Berbasis *Web*

1. Definisi Pembelajaran Berbis *Web*

Menurut (Hamdan, 2018)Pembelajaran berbasis *web* merupakan salah satu bagian dari contoh pembelajaran elektronik (*e-learning*) dengan menggunakan teknologi internet sebagai sarana belajar. Pembelajaran berbasis *web* yang dikenal dengan sebutan *web-based traning* (WBT) atau *web-education* (WBE) dapat didefinisikan sebagai aplikasi teknologi *web* dalam dunia pembalajaran untuk sebuah proses pendidikan. Dengan demikian, pembelajaran berbasis *web* adalah pembelajaran yang pelaksanaannya didukung oleh bantuan teknologi internet.

Gambar II.1
Pembelajaran Berbasis Web

2. Kelebihan Model Pembelajaran Berbasis *Web*

Keuntungan penggunaan *web* ini antara lain :

1. Kemungkinan siswa dimanapun, kapanpun, untuk mempelajari apapun,
2. Pebelajar dapat belajar sesuai dengan karakteristik dan langkahnya dirinya sendiri karena pembelajaran berbasis *web* membuat pembelajaran menjadi bersifat individual,
3. Kemampuan untuk membuat tautan *(link)*, sehingga pebelajar dapat mengakses informasi dari berbagai sumber, baik didalam maupun luar lingkungan belajar,
4. Sangat potensial sebagai sumber belajar bagi pebelajar yang tidak memiliki cukup waktu untuk belajar,

Dapat mendorong pebelajar untuk lebih aktif dan mandiri didalam belajar,

5. Menyediakan sumber belajar tambahan yang dapat digunakan untuk memperkaya materi pembelajaran,
6. Menyediakan sumber belajar tambahan yang dapat digunakan untuk mencari informasi yang mereka butuhkan,
7. Isi dan materi pelajaran dapat di *update* dengan mudah.

3. Fungsi Pembelajaran Berbasis *Web*

Menurut Sidik dalam (Arizona, 2017) mengatakan bahwa,"Situs *Web (Website)* awalnya merupakan suatu layanan sajian informasi yang menggunakan konsep hyperlink yang memudahkan surfer (sebutan bagi pemakai komputer yang melakukan penyelusuran informasi di *internet*)untuk mendapatkan informasi dengan cukup mengklik suatu link berupa teks atau gambar maka informasi dari teks atau gambar akan ditampilkan secara lebih terperinci (detail)".

B. Sistem

Gambar II.2 Sistem

1. Definisi Sistem

Definisi sistem menurut (Mulyadi, 2016), Sistem adalah “suatu jaringan prosedur yang dibuat menurut pola yang terpadu untuk melaksanakan kegiatan pokok perusahaan”.

Menurut (Fahrisal, pohan,& Nasution 2018) mengemukakan bahwa, “Sistem adalah kumpulan atau himpunan dari unsure atau variabel-variabel yang saling berkait, saling berinteraksi, dan saling tergantung satu sama lain untuk mencapai tujuan. Dan juga sebagai sekumpulan objek-objek yang saling berelasi dan berinteraksi”.

2. Karakteristik Sistem

Menurut Jogiyanto dalam(Asahan, Ahmad,& Kisaran 2017) menyatakan bahwa: Suatu sistem yang mempunyai karakteristik atau sifat-sifat yang tertentu, yaitu yang mempunyai komponen (components), batasan sistem (boundary), lingkungan luar sistem (environments), penghubung (interface), masukan (input), keluaran (output), pengolah (process), dan sasaran (objectives) atau tujuan (goal).

3. Klasifikasi Sistem

Sistem dapat diklasifikasikan dari beberapa sudut pandangan menurut (Rahman et al, 2019), diantaranya adalah :

a. Sistem diklasifikasikan sebagai sistem abstrak *(abstract system)*dan sistem fisik *(physical system).* Sistem abstrak adalah sistem yang berupa pemikiran atau ide-ide yang tidak nampak secara fisik. Sistem fisik merupakan sistem yang ada secara fisik.

b. Sistem diklasifikasikan sebagai sistem alamiah *(natural system)* dan sistem buatan manusia *(human made system).* Sistem alamiah adalah sistem yang terjadi melalui proses alam, tidak dibuat manusia. Sistem buatan manusia adalah sistem yang dirancang oleh manusia. Sistem buatan manusia yang melibatkan iteraksi manusia dengan mesin disebut dengan *human machine system.*

c. Sistem diklasifikasikan sebagai sistem tertentu *(deterministic system)*dan sistem tak tentu *(probalistic system).* Sistem tertentu beroperasi dengan tingkah laku yang sudah dapat diprediksi. Sistem tak tentu adalah sistem yang kondisi masa depannya tidak dapat di prediksi karena mengandung unsure probabilitas.

d. Sistem diklasifikasikan sebagai sistem terutup *(closed system)*dan sistem terbuka *(open system).* Sistem tertutup merupakan sistem yang tidak berhubungan dan tidak terpengaruh lingkungan luarnya. Sistem terbuka adalah sistem yang berhubungan dan terpengaruh dengan lingkungan luarnya.

4. Pengertian Informasi

Menurut Kusrini dalam (Effendy & Hasan, 2016) menjelaskan bahwa "Informasi dapat didefinisikan sebagai hasil dari olahan data menjadi sebuah bentuk yang bermakna yang digunakan untuk pengambilan keputusan".

5. Pengertian Sistem Informasi

Menurut Pratama dalam (Maulana & Purwaningtias, 2016)mendefinisikan bahwa "Sistem Informasi adalah gabungan dari software, hardware, infastruktur dan sumber daya manusia (SDM) yang saling berkaitan untuk menciptakan sebuah sistem yang dapat mengolah data menjadi informasi yang bermanfaat".

Menurut (Krismaji, 2015) “Sistem informasi adalah cara-cara yang diorganisasi untuk mengumpulkan, memasukkan, dan mengolah serta menyimpan data, dan cara-cara yang diorganisasi untuk menyimpan, mengelola mengendalikan, dan melaporkan informasi sedemikian rupa sehingga sebuah organisasi dapat mencapai tujuan yang telah ditetapkan.

C. Website

Gambar II.3
Website

Website adalah suatu halaman *web* yang saling berhubungan yang umumnya berisikan kumpulan informasi berupa data teks, gambar, animasi, audio, video maupun gabungan dari semuanya yang biasanya dibuat untuk personal, organisasi dan perusahaan.

Menurut Kustiyaningsih dan Devie Rosa dalam(Fridayantie & Mahdiati, 2016) berpendapat bahwa “Jenis *website* dapat dikategorikan menjadi dua yaitu *web* statis dan dinamis. *Web* statis adalah *web* yang menampilkan informasi-informasi yang sifatnya statis (tetap).Disebut statis karena pengguna tidak dapat

berinteraksi dengan *web* tersebut. Sedangkan *web* dinamis adalah *web* yang menampilkan informasi serta dapat berinteraksi dengan pengguna. *Web* dinamis memungkinkan pengguna untuk berinteraksi menggunakan *form* sehingga dapat mengolah informasi yang ditampilkan.

Menurut Puspitosari dalam(Kesuma & Rahmawati, 2017) menjelaskan bahwa "*Website* adalah halaman informasi yang disediakan melalui jalur internetsehingga bisa diakses diseluruh dunia, selama terkoneksi dengan jaringan internet"

1. **Internet**

Gambar II.4
Internet

> Menurut Siregar dalam (Sutopo et al., 2016) mendefinisikan:
> *Internet* atau *Interconnected Networking* merupakan dua 13omputer atau lebih yang saling berhubungan membentuk jaringan 13omputer hingga meliputi jutaan 13omputer di dunia, yang saling berinteraksi dan bertukar informasi, terhadap masalah sosial misalnya dengan menggunakan alat-alat bantu *online* untuk mencapai bisnis elektronik (*e-commerce*), kepemilikan informasi dan interaksi dengan masyarakat.

Menurut Sibero Internet adalah jaringan komputer yang menghubungkan antar jaringan secara global. Berdasarkan penjelasan para

ahli, dapat disimpulkan bahwa internet adalah jaringan komputer yang terkoneksi dengan jaringan lain yang mempunyai cangkupan luas untuk mendapatkan informasi dari jaringan tersebut.(Arizona, 2017).

2. Web Browser

Gambar II.5
Web Browser

Web browser adalah program untuk menampilkan halaman yang berbentuk kode HTML. Semua halaman *web* ditulis dengan Bahasa HTML (*Hypertext Mark Up Language*). Walaupun beberapa *file* mempunyai ekstensi yang berbeda (contoh: .html, .php, .php3), output *file-file* tersebut tetap HTML. HTML adalah medium yang selalu dikirimkan ke *web browser* baik halaman itu berupa halaman statis, sebuah *script* (seperti PHP), ataupun yang dibuat oleh program CGI (*Common Gateway Interface*). Kasiman dalam (Susanti, 2016)

3. Web Server

Gambar II.6
Web Server

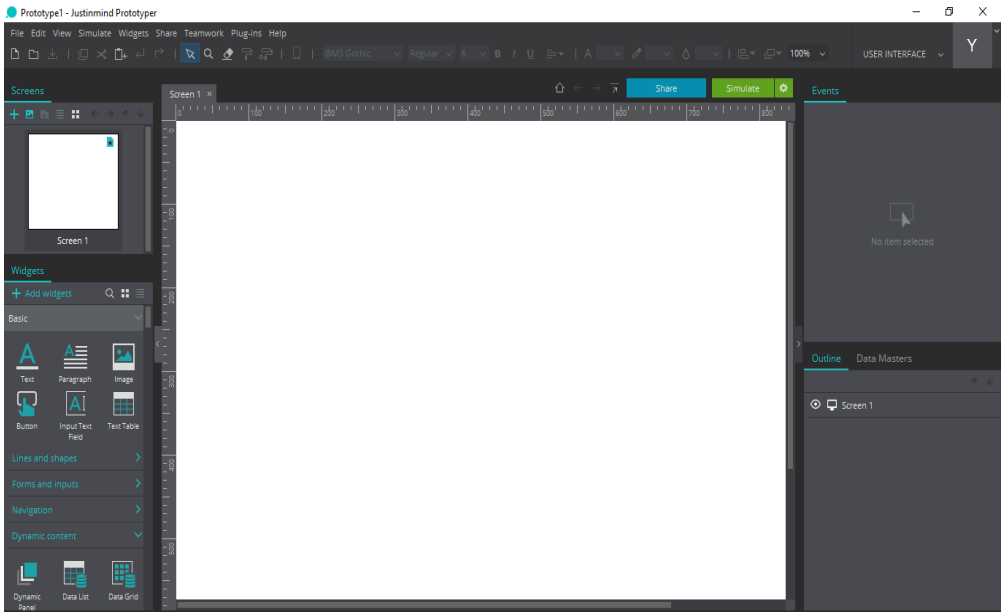
Web server adalah sebuah komputer yang terdiri dari perangkat keras dan perangkat lunak. Secara bentuk fisik dan cara kerjanya, perangkat *web* server tidak berbeda dengan komputer rumah atau PC, yang membedakan adalah kapasitas dan kapabilitas. (Fridayanthie & Mahdiati, 2016).

D. Program

Gambar II.7
Program

Menurut(Kawengian, Mingid, Pantow 2017) Program merupakan himpunan atau kumpulan instruksi tertulis. Orang yang membuat program sering disebut sebagai pemograman atau programmer.

JIM (Justimind) Prototyper adalah suatu tool yang mengadopsi konsep cepat dan intan dalam membuat prototipe, serta memudahkan anda untuk membuat sebuah simulasi secara interaktif dan akurat dari aplikasi yang anda kembangkan sebelumnya (Fiqih, 2016).



Gambar II.8
Tampilan *Justinmind Prototyper*

Melalui JIM *Prototyper*, dapat dengan mudah membuat aplikasi apapun untuk perancangan prototype yang dibuat. Selain itu, dengan Jim *Prototyper* juga dapat mengekspor prototipe tersebut ke dalam bentuk format HTML untuk menunjukkan bahwa prototipe tersebut dapat diakses secara *online.* Serta otomatis, prototipe dapat menghasilkan semua file dokumentasi dalam bentuk *Open Office* atau dalam bentuk *Microsoft Word*. Semua itu dapat dilakukan tanpa membutuhkan pengetahuan coding pemograman sedikitpun.

Adapun beberapa karakteristik JIM Prototyper yang dirujuk oleh sebuah buku kupas tuntas istilah No Progamming sebagai berikut: (Ismawan, 2018).

a. No Progamming, dimana justinmind prototipe merupakan sebuah tool yang intuitif. Kebutuhan pada tool ini, semua dipermudah dan disajikan

dalam kemasan yang praktis. Ketika ingin memuat prototipe pada tool ini, dapat lakukan hanya dengan menyeret dan meletakkan komponen-komponen yang dibutuhkan atau disebut teknik Drag & Drop. Kemudian dapat melakukan sebuah interaksi dari palet-palet yang ada ke area kerja yang diinginkan. Semua itu dapat diimplementasikan dengan istilah NO PROGAMMING.

b. Simulasi secara instan. Hanya dengan satu klik pada tombol *"simulate",* maka sudah dapat melihat hasil prototipe aplikasi yang sudah dibuat dalam satu waktu.

c. Terdokumentasi secara otomatis. Jika ingin prototype yang dibuat dilengkapi dengan semua *file* dokumen yang dinutuhkan, maka *JIM prototype* dapat menghasilkan file dokumentasi *prototype.*

d. Ekspor dalam format HTML, file prototipe dapat di ekspor kedalam bentuk HTML. Tujuan eskpor ini agar para pengguna yang menggunakan prototipe aplikasi dapat memberikan *feedback* (umpan balik) ketika menggunakan protoitpe aplikasi tersebut, serta mereka dapat memberikan pendapat dan pandangan positif yang sifatnya membangun.

E. Basis Data

Menurut Sauptra dalam(Kesuma & Rahmawati, 2017) mengatakan bahwa, "Basis data merupakan data yang saling terhubung dan berkaitan dengan subjek tertentu pada tujuan tertentu pula. Hubungan antar data ini dapat dilihat oleh adanya field ataupun kolom".

a. MySQL *(my Structured Language)*

Gambar II.9
Logo MySQL

Menurut Kurniawan dalam(Wasiyanti & Talaohu, 2016) mengatakan bawha," MySQL adalah "satu jenis database server yang sangat terkenal. MySQL termasuk jenis RDMS *(Relational Database Management System)*.MySQL ini mendukung bahasa pemograman PHP".

b. phpMyAdmin

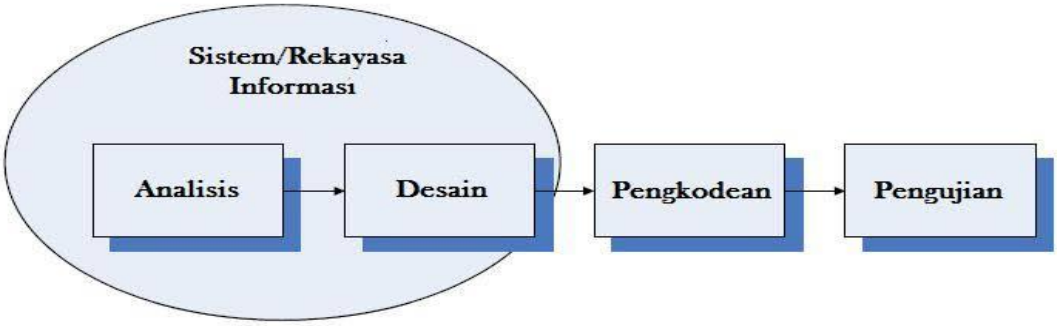
Gambar II.10
Logo phpMyAdmin

Menurut (Fuad, Sutarman, Yayah 2018) menjelaskan bahwa "PhpMyAdmin adalah aplikasi yang dapat digunakan untuk membuat database, pengguna *(user)*, memodifikasi *table* maupun mengirim *database* secara tepat dan mudah tanpa harus menggunakan perintah *(command)* SQL".

F. Model Pengembang Perangkat Lunak

Menurut (A.S & Shalahuddin, 2018) menyatakan bahwa “Model SDLC air terjun *(waterfall)*sering juga disebut model sekuensial linier *(sequential linier)* atau alur hidup klasik *(class life cycle).* Model air terjun menyediakan pendekatan-pendekatan alur hidup perangkat lunak secara sekuensial atau terurut dimulai dar analisis, desain, pengodean, dan tahap pendukung *(support)*”.

Berikut adalah gambar model air terjun (waterfall).



Gambar II.11
Ilustrasi Model *Waterfall*

Penjelasan tahapan-tahapan dari model air terjun *(waterfall)* ini adalah sebagai berikut :

a. Analisi Kebutuhan Perangkat Lunak

   Proses pengumpulan kebutuhan dilakukan secara intensif untuk mespesifikasikan kebutuhan perangkat lunak agar dapat dipahami perangkat lunak seperti apa yang dibutuhkan oleh user. Spesifikasi kebutuhan perangkat lunak pada tahap ini perlu untuk didokumnetasikan.

b. Desain

   Desain perangkat lunak adalah proses multi langkah yang fokus pada desain pembuatan program perangkat lunak termasuk struktur data, arsitektur perangkat lunak, representasi antarmuka, dan prosedur

pengodean. Tahap ini mentranslasi kebutuhan perangkat lunak dari tahapan analisis kebutuhan kerepresentasi desain agar dapat diimplementasikan menjadi program pada tahap selanjutnya.Desain perangkat lunk yang dihasilkan pada tahap ini juga perlu didokumentasikan.

c. Pembuatan Kode Program

   Desain harus ditranslasikan kedalam program perangkat lunak.Hasil dari tahap ini adalah program komputer sesuai dengan desain yang telah dibuat pada tahap desain.

d. Pengujian

   Pengujian fokus pada perangkat lunak secara dari segi lojik dan fungsional dan memastikan bahwa semua bagian sudah diuji.Hal ini dilakukan untuk meminimalisir kesalahan (eror) dan memastikan keluaran yang dihasilkan sesuai dengan yang diinginkan.

e. Pendukung (support) atau pemeliharaan (maintenance)

   Tidak menutup kemungkinan sebuah perangkat lunak mengalami perubahan ketika sudah dikirimkan ke user.Perubahan bisa terjadi karrena adanya kesalahan yang muncul dan tidak terdeteksi saat pengujian atau perangkat lunak harus beradaptasi dengan lingkungan baru. Tahap pendukung atau pemeliharaan dapat mengulangi proses pengembangan mulai dari analisis spesifikasi untuk perubahan perangkat lunak yang sudah ada, tapi tidak untuk membuat perangkat lunak baru.

Model air terjun *(waterfall)* sangat cocok digunakan.Kebutuhan pelanggan sudah sangat dipahami dan kemungkinan terjadinya perubahan kebutuhan selama pengembang perangkat lunak kecil.Hal positif dari model

air terjun *(waterfall)* adalah struktur tahap pengembang sistem jelas, dokumentasi dihasilkan di setiap tahap pengembang, dan sebuah tahap dijalankan setelah tahap selesai dijalankan (tidak ada tumpang tindih dalam pelaksanaan tahap).(A.S & Shalahuddin, 2018).

2.2 Teori Pendukung

Teori yang digunakan untuk menggambarkan bentuk logika model dari suatu sistem dengan menggunakan simbol-simbol, lambing-lambang, diagram-diagram yang menunjukan secara tepat dan fungsinya. Adapun teori pendukung yang dijelaskan sebagai model sistem yang dirancang adalah sebagai berikut :

A. Entity Relationship Diagram (ERD)

1. Definisi ERD

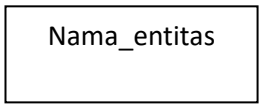
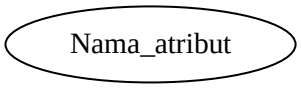
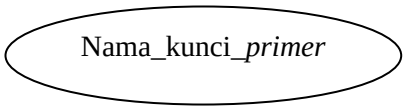
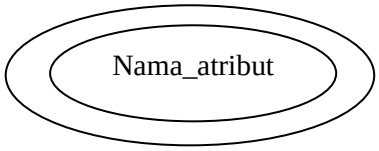
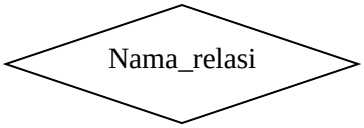
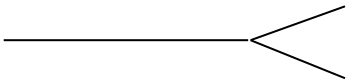
Menurut Brady dan Loonam dalam (Mubarok & Hadianti, 2016) mengatakan bahwa," Entity Relationship Diagram (ERD) merupakan teknik yang digunakan untuk memodelkan kebutuhan data dari suatu organisasi, biasanya oleh System Analys dalam tahap analisis persayatan proyek pengembangan system".

Menurut Sutanta dalam(Anwar & Irawan, 2017) mengatakan bahwa," ERD (Entity Relationship Diagram) merupakan suatu model data yang dikembangkan berdasarkan objek". Sumber lain menyatakan ERD adalah sebuah model diagram yang mendefinisikan keadaan maya-nya dari suatu data yang nyata saling berhubungan yang memiliki beberapa komponen-komponen seperti entitas, atribut, relasi dan kardinalitas yang diwakili dengan gambar atau symbol.

2. Komponen ERD

Berikut adalah simbol-simbol yang digunakan pada ERD dengan notasi Chen:

Tabel II.1.
Simbol notasi Chen

Simbol	Deskripsi
Entitas / Entity 	Entitas merupakan data inti yang akan disimpan; benda yang memiliki data dan harus disimpan datanya agar dapat diakses oleh aplikasi komputer; penamaan entitas biasanya lebih ke kata benda dan belum merupakan nama tabel
Atribut 	*Field* atau kolom data yang butuh disimpan dalam suatu entitas.
Atribut kunci *primer* 	*Field* atau kolom data yang butuh disimpan dalam suatu entitas dan digunakan sebagai kunci akses *record* yang diinginkan; biasanya berupa id; kunci *primer* dapat lebih dari satu kolom, asalkan kombinasi dari beberapa kolom tersebut dapat bersifat unik (berbeda tanpa ada yang sama).
Atribut multi nilai / multivalue 	Field atau kolom data yang butuh disimpan dalam suatu entitas yang dapat memiliki nilai lebih dari satu.
Relasi 	Relasi yang menghubungkan antar entitas: biasanya diawali dengan kata kerja.
Asosiasi / *association* 	Penghubung antara relasi dan entitas dimana dikedua ujungnya memiliki *multiplicity* kemungkinan jumlah pemakaian kemungkinan jumlah maksimun keterhubungan antara entitas yang lain disebut dengan kerdinalitas. Misalkan ada kardinalitas 1 ke N atau sering disebut *one to many*.

3. Pengertian Logical Record Structured (LRS)

Menurut Simarmata dan paryudi dalam (Fridayanthie & Mahdiati, 2016) mengatakan bahwa," Logical Record Structured (LRS) adalah representasi dari stuktur record-record pada table-tabel yang terbentuk dari hasil relasi antar himpunan entitas". Menentukan kardinalitas, jumlah table, dan *ForeignKey* (FK)".

B. *Unified Modeling Language* (UML)

1. Definisi UML

Menurut Martin Fowler dalam (Aprianti Winda dan Umi Maliha, 2016) "*Unifed Modelling Language* (UML) adalah keluarga notasi grafis yang didukung oleh mata-model tunggal, yang membantu pendeskripsian dan sistem perangkat lunak, khusunya sistem yang dibangun menggunakan pemograman berorientasi objek".

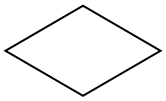
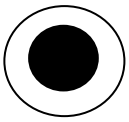
2. Activity Diagram

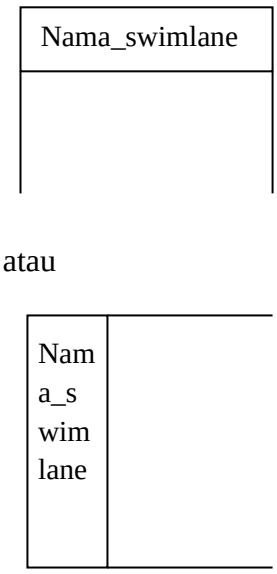
"*Acivity Diagram* (diagram aktivitas) adalah teknik untuk menggambarkan *logica procedural*, proses bisnis, dan jalur dalam beberapa hal, diagram ini memainkan peran mirip sebuah diagram alir, tetapi perbedaan prinsip antara diagram ini dan notasi diagram alur adalah diagram ini mendukung behavior pararel (No, Harahap, Bei, Wirjaamadja,& Bank 2019)".

Definisi Activity diagram adalah sebuah diagram yang menjelaskan sebuah alur kerja atau kegaiatan didalam program yang sedang dirancang. Alur atau aktivitas ini dapat berupa menu-menu atau proses bisnis yang terdapat didalam sistem tersebut.

Berikut adalah simbol-simbol yang ada pada diagram aktivitas :

Tabel II.2
Simbol-simbol Activity Diagram

Simbol	Deskripsi
Status awal	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal.
Aktivitas	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
Percabangan / *decision* 	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu.
Penggabungan / *join*	Asosiasi penggabungna dimana lebih dari satu aktivitas digabungkan menjadi satu.
Status akhir 	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir.

Swimlane 	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi.

3. Use case Diagram

Use case Diagram adalah pemodelan untuk menggambarkan behavior / kelakuan sistem yang dibuat. Use case diagram menggambarkan sebuah interaksi antara satu atau lebih actor dengan sistem yang akan dibuat. Secara sederhana, diagram use case digunakan untuk memahami fungsi apa saja yang ada didalam sebuah sistem dan siapa saja yang dapat mengunakan fungsi-fungsi tersebut.

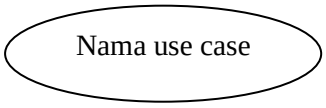
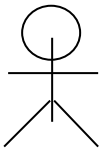
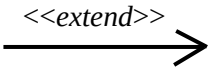
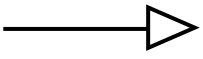
Syarat penamaan pada use case adalah nama didefinisikan sesimpel mungkin dan dapat dipahami. Ada dua hal utama pada *use case*yaitu pendefisinian apa yang disebut actor dan *use case* yaitu : (A.S & Shalahuddin, 2018)

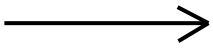
a. Aktor merupakan orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi yang akan dibuat itu sendiri, jadi walaupun symbol dari aktor adalah gambaran orang, tapi aktor belum tentu merupakan orang.

b. *Use case* merupakan fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor.

Berikut adalah simbol-simbol yang ada pada diagram *use case* :

Tabel II.3
Simbol-simbol *Use case diagram*

Simbol	Deskripsi
Use case 	Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor; biasanya dinyatakan dengan menggunakan kata terjadi awal frase nama use case.
Aktor / *actor*  nama aktor	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi yang akan dibuat itu sendiri, jadi walaupun symbol dari aktor adalah gambaran orang, biasanya dinyatakan menggunakan kata benda di awal frase nama aktor.
Asosiasi / *association*	Komunikasi antar aktor dan *use case* yang berpatisipasi antar *use case* atau *use case* berinteraksi dengan aktor.
Ekstensi / *extend* 	Relasi *use case* tambahan ke sebuah *use case* dimana *use case* yang ditambahkan dapat berdiri sendiri walau tanpa *use case* tambahan itu; mirip dengan prinsip *inheritance* pada pemograman berorientasi objek; biasanya *use case* tambahan memiliki nama depan yang sama dengan *use case* yang ditambahkan.
Generalisasi / *generalization* 	Hubungan generalisasi dan spesialisasi (umum – khusus) antara dua buah use case dimana fungsi yang satu adalah fungsi yang lebih umum dari yang lainnya.

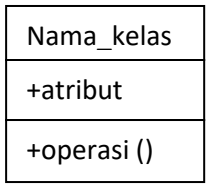
Menggunakan / *include* / *uses* <<*include*>> 	Relasi *use case* tambahan ke sebuah *use case* dimana *use case* ditambahkan memerlukan *use case* ini untuk menjalankan fungsinya atau sebagai syarat di jalankan *use case* ini. Ada dua sudut pandang yang cukup besar mengenai *include* di *use case* : • *Include* berarti *use case* yang ditambahkan akan selalu dipangggil saat *use case* tambahan dijalankan. • *Include* berarti *use case* yang tambahan akan selalu melakukan pengecekan apakah *use case* yang ditambahkan telah dijalankan sebelum *use case* tambahan dijalankan.

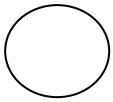
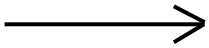
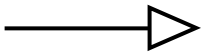
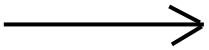
4. Class Diagram

Class diagram adalah diagram yang digunakan untuk menampilkan beberapa kelas serta paket-paket yang ada didalam sistem / perangkat lunak yang sedang kita gunakan.Class diagram memberi kita gambaran (diagram statis) tentang sistem / perangkat lunak dan relasi-relasi yang ada didalamnya.

Berikut adalah simbol-simbol yang ada pada diagram kelas :

Tabel II.4
Simbol-simbol *Class Diagram*

Simbol	Deskripsi
Kelas 	Kelas pada struktur sistem.

Antarmuka / *interface*  Nama_interface	Sama dengan konsep ingterface dalam pemograman berorientasi objek.
Asosiasi / *association* 	Relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan multiplicity.
Asosiasi berarah / directed association 	Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya disertai dengan multiplicity.
Generalisasi 	Relasi antar kelas dengan makna generlasisasi-spesialisasi (umum khusus)
Kebergantungan / *dependency* 	Relasi antar kelas dengan makna kebergantungan antar kelas.
Agregasi / *aggregation*	Relasi antar kelas dengan makna semua-bagian *(whole-part)*

5. Sequence Diagram

Diagram sekuen menggambarkan kelakuan objek pada *use case* dengan mendreskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek. Oleh karena itu untuk menggambar diagram sekuen maka harus diketahui objek-objek yang terlibat dalam sebuah *use case* beserta metode metode yang dimiliki kelas yang diinstansiasi menjadi objek

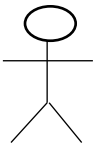
itu. Membuat diagram sekuen juga dibutuhkan untuk melihat skenario yang ada pada *use case*.

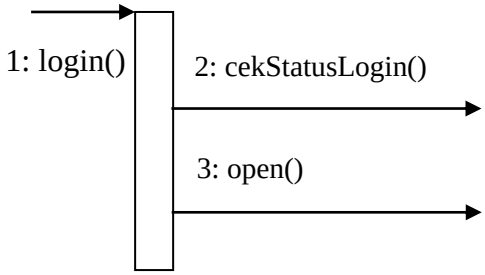
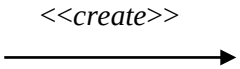
Sequence diagram merupakan sebuah diagram antar kelas dan penjelasan detail tiap-tiap kelas di alam model *desain* dari suatu sistem, juga memperlihatkan aturan-aturan dan tanggung jawab entitas yang menentukan perilaku sistem(Fauzi Siregar & Melani, 2018).

Berikut adalah symbol-simbol yang ada pada diagram sekuen :

Tabel II.

Simbol-simbol Sequence Diagram

Simbol	Deskripsi
Aktor  Nama aktor atau namaaktor Tanpa waktu aktif	Orang, proses, atau sistem lain yang berinterkasi dengan sistem informasi yang akan dibuat diluar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari actor adalah gambar orang, tapi actor belum tentu merupakan orang; biasanya dinyatakan menggunakan kata benda di awal frase nama aktor.
Garis hidup / *lifeline*	Menyatakan kehidupan suatu objek.
Objek Namaobjek :namakelas	Menyatakan objek yang berinteraksi pesan.

Waktu aktif	Menyatakn objek dalam keadaan aktif dan berinteraksi, semua yang terhubung dengan waktu aktif ini adalah sebuah tahapan yang dilakukan didalamnya, sinyalmya  Maka cek Status Login() dan open () dilakukan didalam metode login () Aktor tidak memliki waktu aktif.
Pesan tipe *create* 	Menyatakan suatu objek membuat objek yang lain, arah panah mengarah pada objek yang dibuat.