

BAB II

LANDASAN TEORI

2.1. Konsep Dasar Sistem

Dalam proses pengembangan sistem informasi, diperlukan adanya pemahaman mengenai konsep-konsep dari sistem tersebut. Pendekatan sistem yang merupakan jaringan kerja dari prosedur lebih menekankan urutan-urutan operasi di dalam sistem.

Menurut (Maniah & Hamidin, 2017) Menjelaskan bahwa “Sistem dapat didefinisikan sebagai kumpulan dari elemen-elemen berupa data, jaringan kerja dari prosedur-prosedur yang saling berhubungan, sumber daya manusia, teknologi baik hardware maupun software yang saling berinteraksi sebagai satu kesatuan untuk mencapai tujuan/sasaran tertentu yang sama.”

Menurut (Tyoso, 2016) Mengatakan bahwa “Sistem merupakan suatu kumpulan dari komponen-komponen yang membentuk satu kesatuan.”

Menurut (Mulyani, 2016) Menjelaskan bahwa “Informasi merupakan data yang sudah diolah yang ditunjukkan untuk seseorang, organisasi ataupun siapa saja yang membutuhkan.”

Menurut (Anggraeni & Irviani, 2017) Menjelaskan bahwa “Sistem Informasi adalah suatu sistem dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian yang mendukung fungsi operasi organisasi yang bersifat manajerial dengan kegiatan strategi dari suatu organisasi untuk dapat menyediakan kepada pihak luar tertentu dengan informasi yang di perlukan untuk pengambilan keputusan.”

Menurut Rangkuti dalam (Haryani, 2017) mengatakan bahwa “Persediaan (*Inventory*) merupakan sebagai suatu aktiva yang meliputi barang-barang milik perusahaan dengan dengan maksud untuk dijual dalam suatu periode usaha tertentu memenuhi permintaan dari konsumen atau pelanggan setiap waktu”.

A. Model Pembelajaran Berbasis *Web*

Pembelajaran berbasis *web* adalah pembelajaran yang berhubungan dengan materi ajar yang disajikan melalui *web browser* seperti Mozilla Firefox, Google Chrome dan lain-lain.

Menurut (Batubara, 2018) Menjelaskan bahwa “Pembelajaran berbasis web merupakan salah satu bagian dari contoh pembelajaran elektronik (*e-learning*) dengan menggunakan teknologi internet sebagai sarana belajar.”

Berikut adalah fungsi pembelajaran berbasis *web* :

1. Suplemen (tambahan)

Fungsi ini menjadikan *e-learning* sebagai sumber belajar tambahan yang dapat memperkaya khasanah pengetahuan pelajar. Dalam hal ini, pengajar tidak mewajibkan pelajar untuk mengakses materi-materi yang terdapat pada *website e-learning*, meskipun dengan mengaksesnya pelajar dapat meningkatkan pemahaman mereka terhadap materi yang dipelajari di kelas.

2. Komplemen (pelengkap)

Fungsi ini mengharuskan situs *web e-learning* memiliki konten yang sesuai dengan kurikulum dan kebutuhan pembelajaran agar konten tersebut dapat dijadikan sebagai materi penguatan, remedial, media latihan, atau alat bantu dalam memberikan penugasan secara *online* bagi pelajar yang mengikuti pembelajaran di kelas.

3. Substitusi (pengganti)

Fungsi ini mengharuskan situs *web e-learning* memiliki konten pembelajaran yang mengacu pada kurikulum, lengkap dengan metode yang terintegrasi dalam materi, dan berbagai fitur pengelolaan kegiatan pembelajaran sehingga sistem pembelajaran berbasis *web* tersebut dapat digunakan untuk menggantikan sebagian dari pembelajaran tatap muka. Model pembelajaran ini memberikan kesempatan yang seluas-luasnya kepada pelajar dalam mengelola kegiatan belajarnya sesuai dengan waktu yang dimilikinya. Dalam hal ini, pembelajaran berbasis *web* berfungsi untuk mengatasi kelemahan sistem pembelajaran tatap muka dalam hal ruang dan waktu pelaksanaan pembelajaran dan penyediaan sumber belajar yang beragam. Model pembelajaran ini biasanya dikenal dengan istilah *blended learning* atau *hybrid learning*.

Menurut (Hamdan,2018) Dari berbagai pengalaman dan hasil penelitian telah diperoleh beberapa keunggulan pembelajaran berbasis *web*, yaitu sebagai berikut:

1. Memungkinkan setiap orang untuk belajar sesuai dengan waktu dan tempat yang disukainya sehingga proses pembelajaran tidak membosankan.
2. Peserta didik dapat belajar sesuai dengan karakteristik dan caranya sendiri karena pembelajaran berbasis *web* membuat kegiatan pembelajaran menjadi sifat individual.
3. Dapat mendorong peserta didik untuk lebih aktif dan mandiri dalam belajar.
4. Materi pelajaran dapat di perbaharui dengan mudah dan murah dibanding dengan penggunaan materi buku cetak.

5. Menyediakan sumber belajar tambahan dalam bentuk tautan yang dapat digunakan untuk mengakses informasi dari berbagai sumber, baik di dalam maupun luar lingkungan belajar.
6. Tersedianya fitur *chatting*, *forum* dan *web conference* yang dapat digunakan untuk berkomunikasi dan berdiskusi melalui jaringan internet dengan tanpa dibatasi oleh jarak, tempat dan waktu.

B. Sistem

Menurut (Anggraeni & Irviani, 2017) Mengatakan bahwa “Sistem adalah sekumpulan orang yang saling bekerja sama dengan ketentuan-ketentuan aturan yang sistematis dan terstruktur untuk membentuk satu kesatuan yang melaksanakan suatu fungsi untuk mencapai suatu tujuan.”

Menurut (Rosa & Shalahuddin, 2016) Menjelaskan bahwa “Pendekatan berorientasi objek merupakan suatu teknik atau cara pendekatan dalam melihat permasalahan dan sistem (sistem perangkat lunak, sistem informasi, atau sistem lainnya).”

C. Website

Website atau situs dapat di artikan sebagai kumpulan halaman yang menampilkan informasi data teks, data gambar diam atau gerak, data animasi, suara, video dan atau gabungan dari semuanya, baik yang bersifat statis maupun dinamis yang membentuk satu rangkaian bangunan yang saling terkait dimana masing-masing dihubungkan dengan jaringan-jaringan halaman (*hyperlink*).

Menurut Yuhefizar dalam (Fandhilah, Prاتمanto, & Fatakhudin, 2017) mengatakan bahwa “*Website* adalah keseluruhan halaman-halaman *web* yang terdapat dalam sebuah domain yang mengandung informasi.”

1. *World Wide Web*

Salah satu elemen yang paling menarik dari internet ialah adanya *World Wide Web* (WWW) yang merupakan teknologi canggih saat ini dalam dunia internet. Menurut (Jumardi, 2019) mengemukakan bahwa “*World Wide Web* (WWW) yang biasa disebut dengan web merupakan aplikasi internet yang paling populer yang berjalan pada HTTP.”

2. *Web Browser*

Menurut (Jumardi, 2019) mengemukakan bahwa “*Web browser* sering disebut juga dengan *browser* adalah perangkat lunak yang digunakan untuk mengakses informasi yang ada di *World Wide Web*.”

3. *Web Server*

Menurut (Supono & Putratama, 2016) menjelaskan bahwa “*Web Server* adalah sebuah perangkat lunak server yang berfungsi untuk menerima permintaan dalam bentuk situs *web* melalui HTTP atau HTTPS dari klien itu, yang dikenal sebagai *browser web* dan mengirimkan kembali (reaksi) hasil dalam bentuk situs yang biasanya merupakan dokumen HTML”.

4. *Web Statis*

Menurut (Jumardi, 2019) menjelaskan bahwa “*Web Statis* merupakan sebuah *web* dengan konsep konten dari *website* tidak terhubung dengan *database* yang dibangun dengan menggunakan pemrograman *web* yang hanya berjalan di sisi *client* (*Client Side Programming*) yaitu HTML, CSS dan JavaScript”.

5. *Web* Dinamis

Menurut (Jumardi, 2019) menjelaskan bahwa “*Web* Dinamis merupakan sebuah *web* dengan konsep konten dari *website* terhubung dengan *database* dimana konten dari *website* berubah mengikuti isi dari *database* yang dibangun dengan menggunakan pemrograman *web* yang berjalan disisi *server* (*Server Side Programming*) seperti php, jsp, python, asp dan sebagainya”.

6. Bahasa Pemrograman

Menurut (Abdulloh, 2016) Mengatakan bahwa “Bahasa pemrograman merupakan bahasa pemrograman yang dapat di pahami oleh komputer. Ada banyak bahasa pemrograman yang memiliki fungsi berbeda-beda, di antaranya bahasa pemrograman untuk membuat aplikasi *desktop*, membuat *game*, membuat aplikasi web, membuat aplikasi *handphone*, dan sebagainya”.

7. PHP

Pada awalnya PHP merupakan kependekan dari *Personal Home Page* (situs personal). PHP pertama kali dibuat oleh Rasmus Lerdorf pada tahun 1995, pada waktu itu PHP masih bernama *Form Interpreted* (FI), yang wujudnya berupa sekumpulan skrip yang digunakan untuk mengolah data formulir dari web. Selanjutnya Rasmus merilis kode sumber tersebut untuk umum dan menamakannya PHP/FI. Dengan perilsan kode sumber ini menjadi sumber terbuka, maka banyak pemrogram yang tertarik untuk ikut mengembangkan PHP.

Menurut (Supono & Putratama, 2016) mengatakan bahwa “PHP (*PHP: Hypertext Preprocessor*) adalah suatu bahasa pemrograman yang digunakan untuk menerjemahkan baris kode program menjadi kode mesin yang dapat dimengerti oleh komputer yang bersifat *server-side* yang dapat ditambahkan kedalam HTML”.

8. HTML

HTML pertama kali diperkenalkan oleh IBM pada tahun 1980, saat itu tercetus ide untuk menempatkan elemen-elemen yang berguna untuk menandai bagian suatu dokumen seperti judul, alamat dan isi dokumen. Pada akhirnya elemen-elemen tersebut dibentuk menjadi suatu program untuk melakukan pemformatan dokumen secara otomatis.

Menurut (Jumardi, 2019) Mengatakan bahwa “*HyperText Markup Language* (HTML) adalah sebuah bahasa yang digunakan untuk membuat sebuah halaman *web*, menampilkan berbagai informasi di dalam sebuah penjelajah *web Internet* dan pemformatan *hipertext* sederhana yang ditulis dalam berkas format ASCII agar dapat menghasilkan tampilan wujud yang terintegrasi.”

9. CSS (*Cascading Style Sheet*)

Menurut (Sulistiono, 2018) Menjelaskan bahwa “CSS (*Cascading Style Sheet*) merupakan aturan untuk mengendalikan beberapa komponen dalam sebuah web sehingga akan lebih terstruktur dan seragam.”

10. JavaScript

Menurut (Sulistiono, 2018) mengatakan bahwa “Javascript adalah bahasa pemrograman tingkat tinggi untuk membuat *website* dinamis.”

11. Framework

Menurut (Rahmawati, 2017) mengatakan bahwa “*Framework* adalah struktur konseptual dasar yang digunakan untuk memecahkan sebuah permasalahan atau isu kompleks.”

12. jQuery

Jquery pertama kali dikembangkan oleh Jhon Resig pada tahun 2005. Pada saat itu ia terinspirasi oleh kode *behavior*. Jhon merasa kode-kode *behavior* tidak

elegan dan bahkan sangat jelek, maka dia mulai memikirkan cara untuk membuat *library* yang handal ringan untuk javascript.

Menurut (Sulistiono, 2018) mengatakan bahwa “JQuery sama dengan Javascript Library, yaitu kumpulan kode atau fungsi Javascript siap pakai sehingga mempermudah dan mempercepat kita dalam membuat kode Javascript.”

13. Bootstrap

Menurut (Jubille Enterprise, 2016) Mengatakan bahwa “Bootstrap adalah *framework front-end* yang intuitif dan powerful untuk pengembangan aplikasi *web* yang lebih cepat dan mudah.”

14. CodeIgniter

Menurut Raharjo dalam (Mufida, Rahmawati, & Hertiana, 2019) Mengatakan bahwa “Codeigniter adalah *framework web* untuk bahasa pemrograman PHP, yang dibuat oleh Rick Ellis pada tahun 2006, penemu dan pendiri EllisLab yaitu suatu tim kerja yang berdiri pada tahun 2002 yang bergerak dibidang pembuatan *software* dan *tool* untuk para pengembang *web*.”

D. Basis Data

Menurut (Pamungkas, 2017) mengatakan bahwa “Basis data merupakan suatu kumpulan data terhubung yang disimpan secara bersama-sama pada suatu media, yang diorganisasikan berdasarkan sebuah skema atau struktur tertentu, dan dengan *software* untuk melakukan manipulasi untuk kegunaan tertentu”.

Aplikasi basis data yang digunakan yang digunakan dalam pembuatan *web* antara lain :

1. Xampp

Menurut (Abdulloh, 2016) mengatakan bahwa “Xampp adalah salah satu paket *installer* yang berisi Apache yang merupakan *web server* tempat menyimpan *file-file* yang diperlukan *website*, dan Phpmyadmin sebagai aplikasi yang digunakan untuk perancangan database MySQL.”

2. Apache

Menurut (Abdulloh, 2016) mengatakan bahwa “Apache berfungsi sebagai *web server*, yaitu tempat-tempat menyimpan *file-file* php dan lainnya yang di perlukan *website*.”

3. PhpMyAdmin

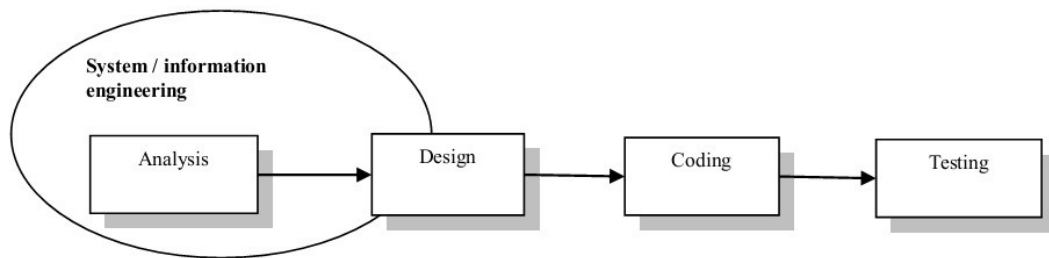
Menurut (Supono & Putratama, 2016) “Phpmyadmin merupakan aplikasi berbasis *web* yang digunakan untuk membuat *database* MySQL sebagai tempat untuk menyimpan data-data *website*.”

4. MySQL

Menurut (Jubilee Enterprise, 2018) mengatakan bahwa “MySQL merupakan server yang melayani *database*. Untuk membuat dan mengolah *database*, kita dapat mempelajari pemrograman khusus yang disebut *query* (perintah) SQL.”

D. Model Pengembangan Perangkat Lunak

Menurut (Rosa & Shalahuddin, 2016) menjelaskan bahwa “Model SDLC air terjun (*waterfall*) sering juga disebut model sequensial linier (*sequential linear*) atau alur hidup klasik (*classic life cycle*). Model air terjun menyediakan pendekatan alur hidup perangkat lunak secara sequensial atau terurut dimulai dari analisis, desain, pengodean, pengujian, dan tahap pendukung”. Berikut ini adalah gambar model air terjun (*waterfall*):



Sumber : (Rosa & Shalahuddin, 2016)

Gambar II.1
Contoh Ilustrasi Model *waterfall*

Berikut ini adalah tahapan model air terjun (*waterfall*) ;

1. Analis Kebutuhan Perangkat Lunak

Proses pengumpulan kebutuhan dilakukan secara intensif untuk menspesifikasikan kebutuhan perangkat lunak agar dapat dipahami perangkat lunak seperti apa yang dibutuhkan oleh *user*.

2. Desain

Proses multistep yang fokus pada desain pembuatan program perangkat lunak termasuk struktur data, arsitektur perangkat lunak, representasi antar muka, dan prosedur pengodean.

3. Pembuatan Kode Program

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

4. Pengujian

Pengujian fokus pada perangkat lunak secara dari segi logik dan fungsional dan memastikan bahwa semua bagian sudah diuji. Hal ini dilakukan untuk

meminimalisir kesalahan (*error*) dan memastikan keluaran yang dihasilkan sesuai dengan apa yang diinginkan.

5. Pendukung (*Support*) dan Pemeliharaan (*Maintenance*)

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.

2.2. Teori Pendukung

A. *Unified Modelling Language*

Menurut (Hanief & Pramana, 2018) mengatakan bahwa “*Unified Modelling Language* (UML) adalah sebuah bahasa yang telah menjadi standar dalam industri untuk visualisasi, merancang dan mendokumentasikan perangkat lunak.”

Sedangkan menurut (Rosa & Shalahuddin, 2016) mengatakan bahwa “UML (*Unified Modelling Language*) adalah salah satu standar mendefinisikan *requirement*, membuat analisi & desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek”.

UML muncul karena adanya kebutuhan pemodelan visual untuk menspesifikasikan, menggambarkan, membangun, dan dokumentasi dari sistem perangkat lunak. UML merupakan bahasa *visual* untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks pendukung.

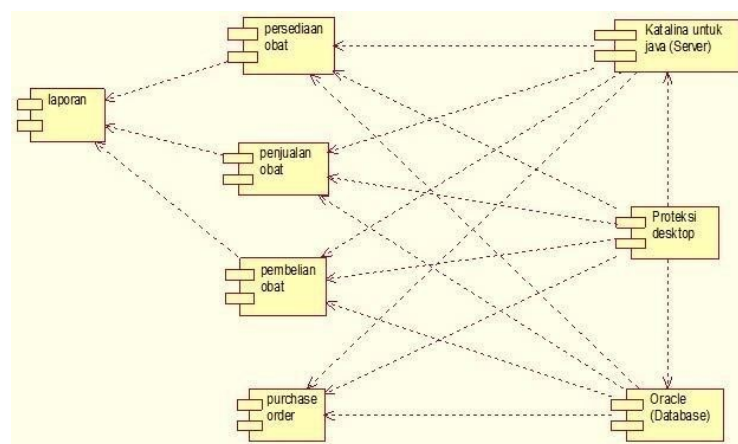
Macam-macam UML menurut (Rosa & Shalahuddin, 2016) adalah :

1. *Component Diagram*

Diagram komponen dibuat untuk menunjukkan organisasi dan ketergantungan diantara kumpulan komponen dalam sebuah sistem. Diagram komponen fokus pada komponen sistem yang dibutuhkan dan ada di dalam

sistem. Diagram komponen juga dapat digunakan untuk memodelkan hal-hal berikut :

- a) *Source code* program perangkat lunak
- b) Komponen *executable* yang dilepas ke *user*
- c) Basis data secara fisik
- d) Sistem yang harus beradaptasi dengan sistem lain
- e) *Framework* Sistem



Sumber : (Rosa & Shalahuddin, 2016)

Gambar II.2

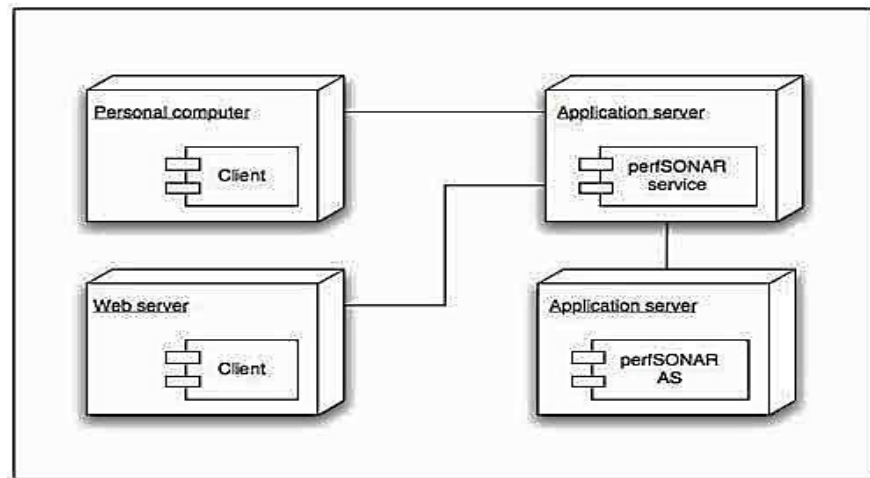
Contoh *Component Diagram*

1. *Deployment Diagram*

Diagram *deployment* atau *deployment diagram* menunjukkan konfigurasi komponen dalam proses eksekusi aplikasi. Diagram *deployment* juga dapat digunakan untuk memodelkan hal-hal berikut :

- a) Sistem tambahan (*embedded system*) yang menggambarkan rancangan *device*, *node*, dan *hardware*.
- b) Sistem *client/server*

- c) Sistem distribusi murni
- d) Rekayasa ulang aplikasi



Sumber : (Rosa & Shalahuddin, 2016)

Gambar II.3

Contoh Deployment Diagram

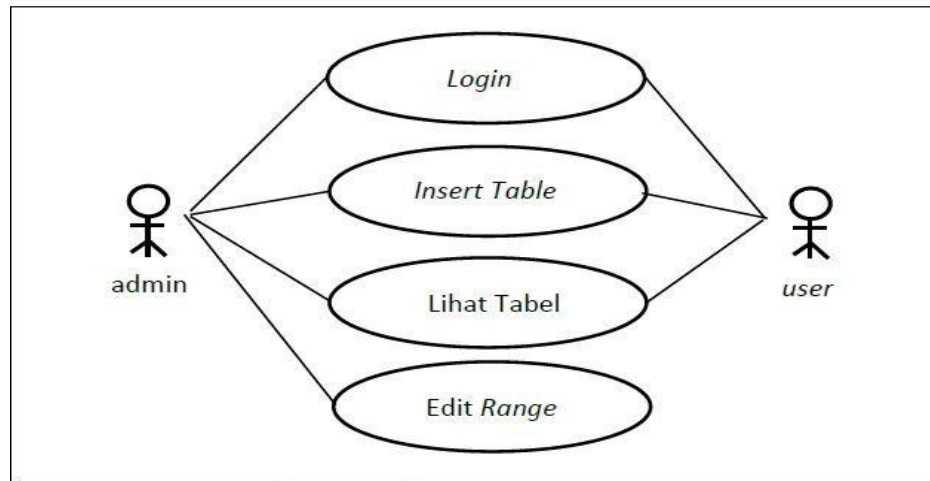
3. Use Case Diagram

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

Ada dua hal utama pada *use case* yaitu pendefinisian apa yang disebut aktor dan *use case*, yaitu :

- a) Aktor merupakan orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar 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.



Sumber : (Rosa & Shalahuddin, 2016)

Gambar II.4

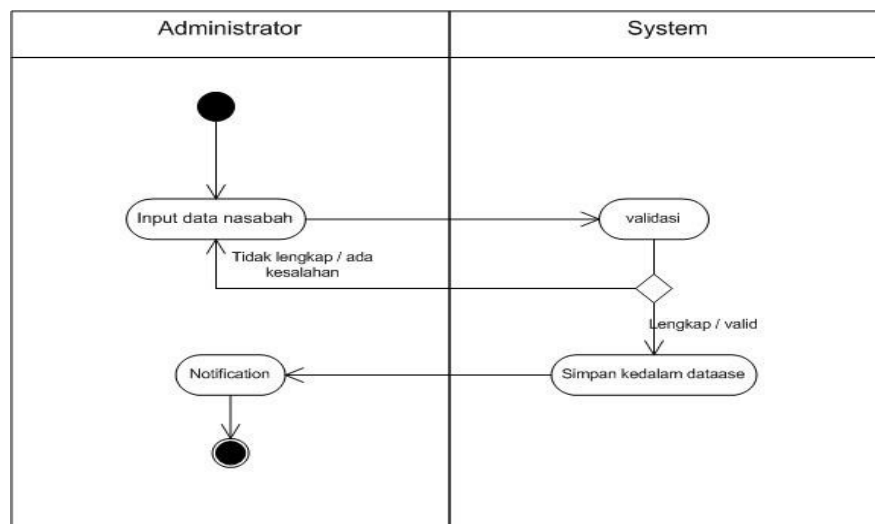
Contoh Use Case Diagram

4. Activity Diagram

Diagram aktivitas atau *activity diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Yang perlu diperhatikan disini adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem. Diagram aktivitas juga banyak digunakan untuk mendefinisikan hal-hal berikut :

- a) Rancangan proses bisnis dimana setiap urutan aktivitas yang digambarkan merupakan proses bisnis sistem yang didefinisikan.
- b) Urutan atau pengelompokan tampilan dari sistem / *user interface* dimana setiap aktivitas dianggap memiliki sebuah rancangan antarmuka tampilan.

- c) Rancangan pengujian dimana setiap aktivitas dianggap memerlukan sebuah pengujian yang perlu didefinisikan kasus ujinya



- d) Rancangan menu yang ditampilkan pada perangkat lunak

Sumber : (Rosa & Shalahuddin, 2016)

Gambar II.5

Contoh Activity Diagram

B. Entity Relationship Diagram (ERD)

Menurut (Rosa & Shalahuddin, 2016) mengatakan bahwa “ERD digunakan untuk pemodelan basis data menggunakan relasional. Sehingga jika penyimpanan basis data menggunakan OODBMS maka perancangan basis data tidak menggunakan ERD”. Berikut ini adalah komponen-komponen yang digunakan dalam pembuatan ERD:

1. Entity atau Entitas

Entitas merupakan data inti yang akan disimpan, bakal tabel pada basis data, 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. Entitas biasanya digambarkan dalam bentuk persegi panjang.

2. Atribut

Atribut adalah karakteristik dari *entity* atau entitas, yang menyediakan penjelasan detail tentang *entity* atau entitas tersebut. Atribut digambarkan dalam bentuk oval.

3. *Relationship* atau Relasi

Relationship atau relasi adalah hubungan alamiah yang terjadi antara satu atau lebih *entity*, misalnya proses pembayaran pegawai. Relasi menentukan kejadian suatu entitas untuk satu kejadian pada entitas yang berhubungan, Misalnya mahasiswa bisa mengambil banyak mata kuliah.

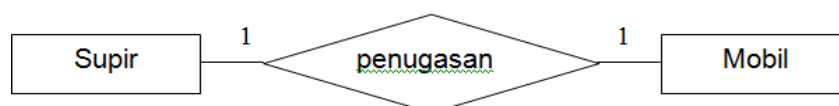
4. Kardinalitas (Derajat relasi)

Kardinalitas relasi menunjukkan jumlah maksimum tupel yang dapat berelasi dengan entitas pada entitas yang lain. Kardinalitas relasi merujuk kepada hubungan maksimum yang terjadi dari entitas yang satu ke entitas yang lain dan juga begitu sebaliknya. Terdapat 3 macam kardinalitas relasi, yaitu:

a. *One to one*

Hubungan satu *entity* dengan satu *entity*

Contoh



Gambar II.6
Relasi *One to one*

b. *One to Many* (1:M) atau *Many to One* (M:1)

Hubungan satu *entity* dengan banyak *entity* atau banyak *entity* dengan satu *entity*.

Contoh:

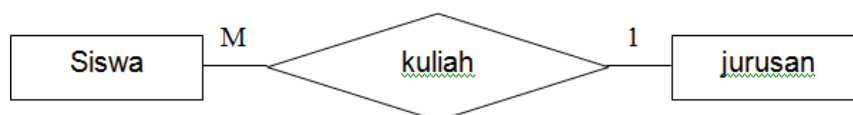


Gambar II.7
Relasi *One to Many* (1:M) atau *Many to One* (M:1)

c. *Many to Many* (M:N)

Hubungan banyak *entity* dengan banyak *entity*

Contoh:



Gambar II.8
Relasi *Many to Many* (M:N)

C. *Logical Record Structure* (LRS)

Menurut Sinarmata dalam (Pakpahan, Fitriani, & Asriani, 2018) mengatakan bahwa “*Logical Record Structure* (LRS) adalah representasi dan struktur *record-record* pada tabel-tabel yang terbentuk dari hasil antara himpunan entitas.

