

BAB II

LANDASAN TEORI

2.1. Konsep Dasar Sistem

2.1.1 Pengertian Sistem

Menurut Pratama (2014:7) dalam jurnal (Lestari et al., 2018) mengemukakan bahwa, “Sistem didefinisikan sebagai sekumpulan prosedur yang saling berkaitan dan saling terhubung untuk melakukan suatu tugas bersama-sama”.

2.1.2. Pengertian Informasi

Pengertian Informasi Menurut Pratama (2014:9) dalam jurnal (Lestari et al., 2018) mengemukakan bahwa, “Informasi merupakan hasil pengolahan data dari satu atau berbagai sumber, yang kemudian diolah, sehingga memberikan nilai, arti dan manfaat”. Menurut Darmawan dan Fauzi (2013:26),

2.1.3. Pengertian Sistem Informasi

Pengertian Sistem Informasi Menurut Hutahaean (2015:13) dalam jurnal (Lestari et al., 2018) mengatakan bahwa, “sistem informasi adalah suatu sistem didalam suatu organisasi yang mempertemukan kebutuhan pengelolaan

2.1.4. Karakteristik Sistem

Suatu sistem mempunyai karakteristik atau sifat-sifat tertentu yang menjadi ciri pada suatu sistem, adapun karakteristik yang dimaksud adalah sebagai berikut :

1. Komponen sistem (*Components*) Sistem terdiri dari jumlah komponen yang saling berinteraksi membentuk suatu kesatuan, komponen system dapat berupa suatu subsistem.

2. Batasan Sistem (*Boundary*)

Merupakan daerah yang membatasi antara suatu sistem dengan sistem yang lainnya atau dengan lingkungan luarnya, batasan ini memungkinkan suatu sistem dipandang sebagai kesatuan yang tidak dapat dipisah-pisah.

3. Lingkungan Luar Sistem (*Environment*)

Merupakan segala bentuk apapun yang berada diluar ruang lingkup atau batasan sistem yang mempengaruhi operasi sistem.

4. Penghubung Sistem (*Interface*)

Merupakan media penghubung antara subsistem dengan subsistem lainnya, melalui penghubung ini, memungkinkan sumber-sumber daya mengalir dari suatu subsistem dengan subsistem lainnya.

5. Masukan Sistem (*Input*)

Merupakan suatu energi yang dimasukkan kedalam sistem yang berupa pemeliharaan (*Maintenance Input*) dan sinyal (*Signal Input*)

6. Keluaran Sistem (*Output*)

Merupakan hasil dari energi yang diolah dan diklasifikasikan menjadi keluaran yang berupa hasil dan sisa pembuangan.

7. Pengolah Sistem (*Process*)

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

8. Sasaran Sistem (*Objective*)

Suatu sistem mempunyai sasaran (*objective*) dan tujuan (*goal*) yang pasti, yang menentukan masukan dan keluaran sistem.

A. Model Pembelajaran Web

Dalam pembelajaran yang menyertai web sebagai salah satu cara pengembangan dalam proses pembelajaran ini sangat membantu kita dalam melakukan berbagai kegiatan secara efektif, karena kita dapat mencari dan mendapatkan ilmu seluas mungkin disini, serta mengetahui banyak hal yang belum kita pahami sebelumnya. Dan yang terpenting ialah menggunakan web ini untuk kebutuhan yang positif tidak untuk kebutuhan yang negatif.

Bila dirancang dengan baik dan tepat pembelajaran berbasis web bisa menjadi pembelajaran yang menyenangkan, dikarenakan sifatnya maya/virtual, pembelajaran berbasis web dianggap telah memberikan fleksibilitas terhadap kegiatan pengaksesan materi pembelajaran. Selain itu, memungkinkan setiap orang dimanapun, kapanpun, untuk mempelajari apapun berbasis web.

2.1.5. Klasifikasi Sistem

Sistem dapat diklasifikasikan dari beberapa sudut pandang di antaranya sebagai berikut :

1. Sistem abstrak (*Abstract System*) dan sistem fisik (*Physical System*)
2. Sistem alamiah (*Nature System*) dan sistem buatan (*Human Mode System*)

2.1.6 Website

Menurut Yuhefizar dalam (Agus Prayitno, 2015) pengertian *website* adalah “keseluruhan halaman-halaman *web* yang terdapat dari sebuah domain yang mengandung informasi”.

1. Internet

Menurut Oneto dan Sugiarto dalam (Agus Prayitno, 2015:2) “internet adalah jaringan komputer”. Ibarat jalan raya, internet dapat dilalui berbagai sarana transportasi, seperti bus, mobil dan motor yang memiliki kegunaan masing-masing.

2. *Web Server*

Web server dapat memberikan layanan data yang berfungsi menerima permintaan HTTP atau HTTPS dari klien yang dikenal dengan *browser web* dan mengirimkan kembali hasilnya dalam bentuk halaman-halaman *web* yang umumnya berbentuk dokumen HTML.

Menurut Sadeli Dalam (Supriyanta dan Nisa, 2015:36) “*Web Server* adalah sebuah perangkat lunak server yang berfungsi menerima permintaan HTTP dan HTTPS dari klien yang dikenal dengan *web browser* yang umumnya berbentuk dokumen HTML”.

3. *Web Browser*

Menurut Abdulloh (2016:5)(Handayani et al., 2018) menggambarkan bahwa Web browser Digunakan untuk menampilkan hasil website yang telah dibuat. Web browser yang paling sering digunakan, di antaranya Mozilla Firefox, Google Chrome, Internet Explorer, Opera, dan Safari.

2.1.7 Basis Data

Menurut Stephens dan Plew dalam (Isty dan Afifah, 2018:2) mengemukakan bahwa “Basis data adalah mekanisme yang digunakan untuk menyimpan informasi atau data”.

1. XAMPP

Menurut Wardana (2015:4) dalam jurnal (Handayani et al., 2018) dalam bukunya menerangkan bahwa XAMPP adalah paket software yang di dalamnya sudah terkandung Web Server Apache, database MySQL, dan PHP interpreter.

2. *Apache*

Menurut Sadeli dalam (Supriyanta dan Nisa, 2015:2) “*Apache (Server HTTP Apache atau ServerWeb/WWW Apache)* adalah *web server* yang dapat dijalankan di banyak

sistem operasi (*Unix, BSD, Linux, Microsoft 14 Windows dan Novell Netware serta platform lainnya*) yang berguna untuk melayani dan memfungsikan situs *web*”.

3. *MySQL*

Menurut Faizal dan Irnawati (2015:4) dalam jurnal (Handayani et al., 2018) mendefinisikan MySQL merupakan perangkat lunak sistem manajemen berbasis data SQL (bahasa Inggris : database management system) atau DBMS yang multithread dan multi- user, dengan sekitar 6 (enam) juta instalasi di seluruh dunia.

4. *Php MyAdmin*

Menurut Abdulloh (2016:4) dalam jurnal (Handayani et al., 2018) PhpMyAdmin merupakan aplikasi berbasis web yang digunakan untuk membuat database MySQL sebagai tempat untuk menyimpan data-data website.

2.1.8 Bahasa Pemrograman

Menurut Sunarti dalam (Afrizal Subhan, 2017:3) “ Bahasa pemrograman adalah perangkat lunak yang digunakan untuk menerjemahkan atau menuliskan algoritma dalam bentuk teks perintah yang dapat dimengerti oleh komputer untuk menyelesaikan masalah”.

1. HTML(*Hypertext Markup Language*)

HTML (Hyper Text Markup Language) Menurut Hidayatullah dan Kawistara (2014:13), dalam jurnal (Eka Wida Fridayanthie danTias Mahdiati, 2016) Hyper Text Markup Language atau HTML adalah bahasa standar yang digunakan untuk menampilkan halaman web. Yang bisa dilakukan dengan HTML yaitu: mengatur tampilan dari halaman web dan isinya, membuat table dalam halaman web, mempublikasikan halaman web secara online, membuat form yang bisa digunakan untuk menangani registrasi dan transaksi via web, menambahkan objek-objek seperti

citra, audio, video, animasi, java applet dalam halaman web, serta menampilkan area gambar (canvas) di browser.

Semua tag-tag HTML bersifat dinamis, artinya kode HTML tidak dapat dijadikan sebagai file executable program. Hal ini disebabkan HTML hanyalah sebuah bahasa scripting yang dapat berjalan apabila dijalankan didalam browser (pengakses web), browser-browser yang mendukung HTML antara lain adalah Internet Explorer, *Netscape Navigator*, *Opera*, *Mozilla* dan lain-lain. Jadi pada saat ingin membuka halaman yang berasal dari HTML anda dapat melihat bentuk pengkodeannya dengan cara mengklik menu *view-source*, maka disana akan ditampilkan semua tag beserta isi dari halaman web tersebut. Karena HTML merupakan sebuah kode scripting dan bukan merupakan program compiler maka untuk menulis kode program harus menggunakan editor, adapun editor yang dapat digunakan adalah *Macromedia Dreamweaver*, *Front Page*, *Home Site* atau *Note pad* sebagai editor standar windows

2. PHP (*Perl Hypertext Preprocessor*)

Menurut Hidayatullah dan Kawistara (2014:231), dalam jurnal (Eka wida ridyanthie dan Tias Mahdiat, 2016), 2016) PHP singkatan dari Perl Hypertext Preprocessor yaitu bahasa pemrograman web server-side yang bersifat open source. PHP merupakan script yang berintergrasi dengan HTML dan berada pada server (server side HTML embedded scripting). PHP adalah script yang digunakan untuk membuat halaman web dinamis. Dinamis berarti halaman yang akan ditampilkan dibuat saat halaman itu diminta oleh client. Mekanisme ini menyebabkan informasi yang diterima client selalu yang terbaru/up to date. Semua script PHP dieksekusi pada server dimana script tersebut dijalankan.

3. JavaScript

Menurut Abdulloh (2016:3) dalam jurnal (Handayani et al., 2018) memaparkan tentang Peran javascript dalam membuat website adalah memberikan efek animasi yang menarik dan interaktif dalam penanganan event yang dilakukan oleh pengguna website.

4. J-Query

Menurut Abdulloh (2016:147) dalam jurnal (Handayani et al., 2018) menjelaskan jika jQuery merupakan salah satu java script library, yaitu kumpulan fungsi java script siap pakai, sehingga mempermudah dan mempercepat kita dalam membuat kode java script

5. CSS (Cascading Style Sheet)

Menurut Abdulloh (2016:2) dalam jurnal (Handayani et al., 2018) menjelaskan tentang bahwa CSS singkatan dari Cascading Style Sheets, yaitu skrip yang digunakan untuk mengatur desain website.

6. Bootstrap

Menurut Abdulloh (2016:157) dalam jurnal (Handayani et al., 2018) Bootstrap merupakan salah satu framework CSS yang sangat populer di kalangan pecinta pemrograman website. Dengan menggunakan bootstrap, proses desain website tidak dibuat dari nol, sehingga proses desain website lebih cepat dan mudah.

7. Sublime Text

Menurut Yosef murya (2017:3) dalam jurnal (Setiawan, 2018) yang menjelaskan bahwa, “Sublime text adalah sebuah syntax editor yang menggunakan python API. Sublime text mempunyai fitur plugin tambahan yang mempermudah bagi penggunanya, tidak hanya memiliki fitur yang menarik, sublime text juga menampilkan desain yang simple dan memiliki cirri khas tersendiri sehingga menjadikan sublime text terkesan elegan untuk sebuah syntax editor”.

2.1.9 Model Pengembangan Perangkat Lunak

Menurut Sukamto dan Shalahuddin (2014:28) dalam jurnal (Haryadi dan Sopandi, Robi sopandi, 2019), “model waterfall adalah model SDLC (Software Development Life Cycle) yang paling sederhana. Model ini hanya cocok untuk pengembangan perangkat lunak dengan spesifikasi yang tidak berubah-ubah”. Model SDLC air terjun (waterfall) sering juga disebut dengan model sekuensial linier (sequential linear) atau alur hidup klasik (classic life cycle). Model air terjun menyediakan pendekatan alur hidup perangkat lunak secara sekuensial atau terurut dimulai dari analisis, desain, pengkodean, pengujian dan tahap pendukung (support).



Sumber : Sukamto, 2017:12

Gambar II.1 Ilustrasi model *waterfall*

Berikut adalah model air terjun (waterfall):

1. Analisis 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. Spesifikasi kebutuhan perangkat lunak pada tahap ini perlu untuk didokumentasikan.
2. Desain Desain perangkat lunak adalah proses multi langkah yang fokus pada desain pembuatan program perangkat lunak termasuk struktur data, arsitektur perangkat lunak, representasi antar muka, dan prosedur pengkodean. Tahap ini mentranslasi kebutuhan perangkat lunak dari tahap analisis kebutuhan ke representasi

desain agar dapat di implementasikan menjadi program pada tahap selanjutnya. Desain perangkat lunak yang dihasilkan pada tahap ini juga perlu di dokumentasikan.

3. Pembuatan Kode Program Desain harus ditranslasikan ke dalam program perangkat lunak. Hasi 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 lojik dan fungsional dan memastikan bahwa semua bagian sudah diuji. Hal ini dilakukan untuk meminimalisir kesalahan (error) dan memastikan keluaran yang dihasilkan sesuai dengan yang di inginkan.

2.2 Teori Pendukung

Adapun peralatan pendukung yang digunakan untuk merancang suatu sistem baru yaitu :

2.2.1 *Entity Relationship Diagram (ERD)*

1. Pengertian ERD

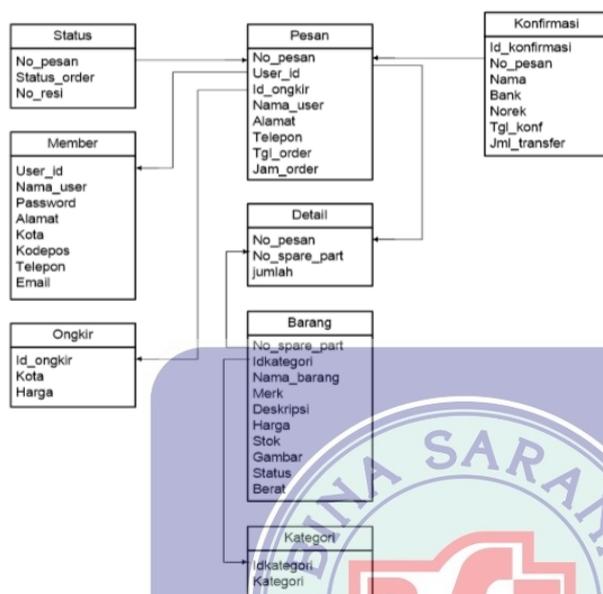
Menurut (Sukamto & Shalahuddin, 2015) dalam jurnal (Baginda Oloan Lubis, 2018:250) “ERD adalah bentuk paling awal dalam melakukan perancangan basis data relasional”. ERD merupakan singkatan dari *Entity Relationship Diagram*, dikembangkan berdasarkan teori himpunan dalam bidang matematika. ERD digunakan untuk pemodelan basis data relasional. Sehingga jika penyimpanan basis data menggunakan OODBMS (*Object Oriented Database Management System*) maka perancangan basis data tidak perlu menggunakan ERD. Model ERD terdiri dari beberapa komponen dasar yaitu Entitas, Atribut, Relasi dan Tingkat Hubungan.

Sedangkan menurut (Handayani dan Septiana, 2017) , “*Entity Relationship Diagram* (ERD) adalah alat pemodelan data utama dan akan membantu mengorganisasi data dalam suatu proyek ke dalam entitas-entitas dan menentukan hubungan antar entitas”.

Proses memungkinkan analisis menghasilkan struktur basis data dapat disimpan dan diambil secara efisien. Simbol-simbol dalam ERD (*Entity Relationship Diagram*) adalah sebagai berikut:

- a. Entitas: suatu yang nyata atau abstrak yang mempunyai karakteristik dimana kita akan menyimpan data.
 - b. Atribut: ciri umum semua atau sebagian besar instansi pada entitas tertentu.
 - c. Relasi: hubungan alamiah yang terjadi antara satu atau lebih entitas.
 - d. Link: garis penghubung atribut dengan kumpulan entitas dan kumpulan entitas dengan relasi.
2. Kardinalitas Relasi
- a. Satu ke satu (*One to One*) Setiap elemen dari Entitas A berhubungan paling banyak dengan elemen pada Entitas B. Demikian juga sebaliknya setiap elemen B berhubungan paling banyak satu elemen pada Entitas A.
 - b. Satu ke banyak (*One to Many*) Setiap elemen dari Entitas A berhubungan dengan maksimal banyak elemen pada Entitas B. Dan sebaliknya setiap elemen dari Entitas B berhubungan dengan paling banyak satu elemen di Entitas A.
 - c. Banyak ke satu (*Many to One*) Setiap elemen dari Entitas A berhubungan paling banyak dengan satu elemen pada Entitas B. Dan sebaliknya setiap elemen dari Entitas B berhubungan dengan maksimal banyak elemen di entitas A.

- b. Jika relasinya satu-ke-banyak, maka *forigen key* diletakan pada entitas Many.
- c. Jika relasinya banyak-ke-banyak, maka di buat”*file konektor*” yang berisi dua *forigen key* yang berasal dari kedua entitas.



Sumber : (Wardani & Sari, 2017)

Gambar II.3 Contoh Gambar LRS (Logical Record Structur)

2.2.3 Unified Modelling Language

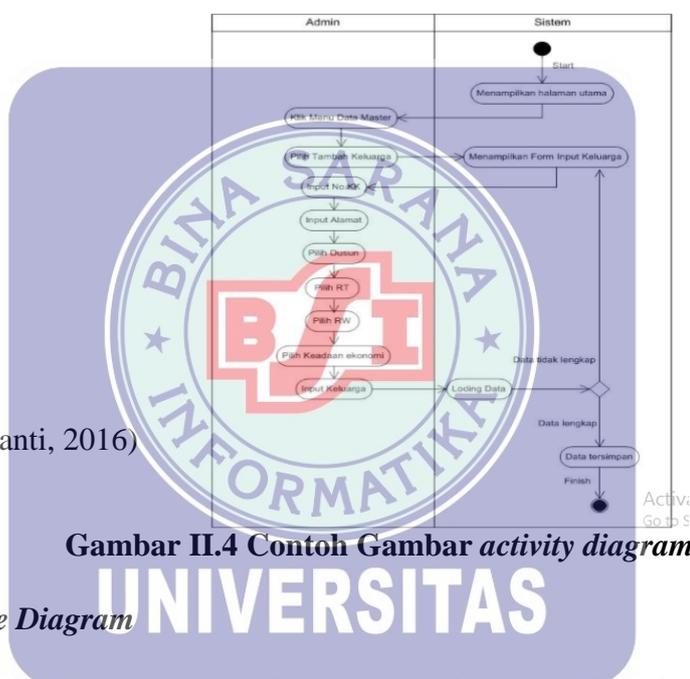
1. Definisi UML

Menurut(Sukamto dan Shalahuddin, 2015) dalam jurnal (Baginda Oloan Lubis, 2018:250)“*Unified Modelling Language (UML)* merupakan sebuah standarisasi bahasa pemodelan untuk pembangunan perangkat lunak yang dibangun dengan menggunakan teknik pemrograman berorientasi objek”. *UML* muncul karena adanya kebutuhan pemodelan visual untuk menspesifikasikan, menggambarkan, membangun, dan dokumentasi dari sistem perangkat lunak. Pada UML terdapat macam-macam diagram yang terbagi didalam 3 kategori, yaitu kategori *structure diagrams*, *behavior diagrams*, dan *interaction diagrams* .Adapun macam- macam diagram yang

didefinisikan menurut (Sukamto dan Shalahuddin, 2015) yaitu *Use Case Diagram*, *Activity Diagram*, *Component Diagram*, dan *Deployment Diagram*

2. Activity Diagram

Menurut Sukamto dan Shalahuddin (2015:161) dalam jurnal (Bayu dan Yanto, 2018)“Activity Diagram menggambarkan workflow (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis. Yang perlu diperhatikan disini adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor”.

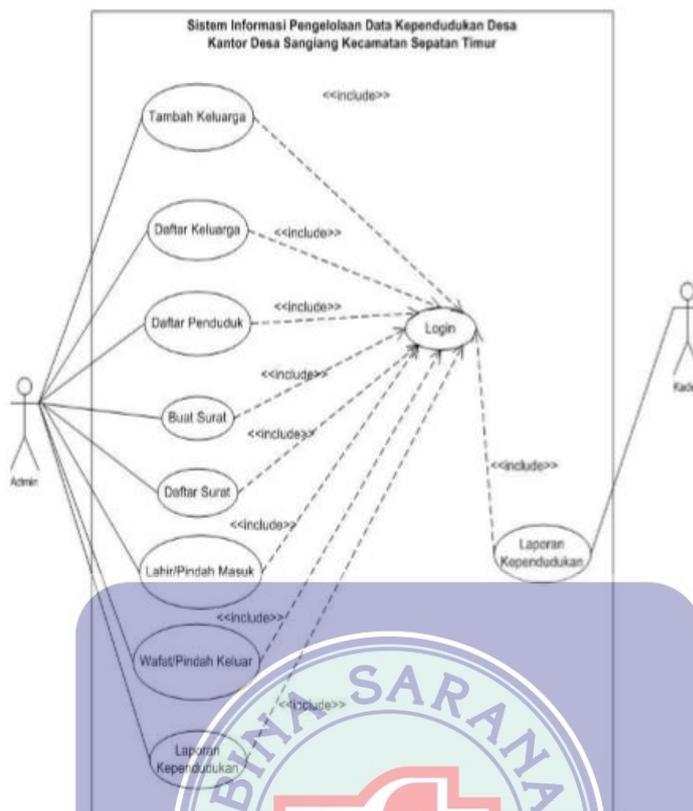


Sumber : (Shanti, 2016)

Gambar II.4 Contoh Gambar activity diagram

3. Use Case Diagram

Usecase Diagram Rosa dan Shalahudin (2014), dala jurnal (Susila dan Taufiq, 2018:504) *Diagram usecase* 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 fungsifungsi itu.



Sumber: (Shanti, 2016)

Gambar 11.5 Contoh gambar *use case diagram*

4. Class Diagram

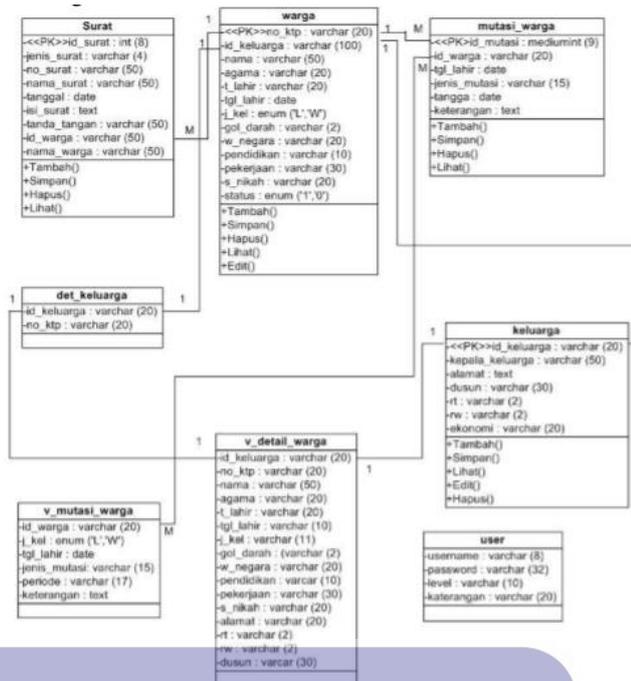
Menggambarkan struktur sistem dari segi pendefinisian kelas- kelas yang akan di buat untuk membangun sistem. Kelas memiliki apa yang di sebut atribut dan metode atau operasi. Berikut adalah pembagian kelas yang disebut atribut dan metode atau operasi Menurut Sukamto dan Shalahuddin dalam (Sunarti, 2016).

- a. Atribut merupakan variabel-variabel yang di miliki oleh suatu kelas.
- b. Operasi atau metode adalah fungsi yang dimiliki oleh suatu kelas Diagram kelas dibuat agar pembuat program atau programmer membuat kelas-kelas sesuai rancangan di dalam diagram kelas agar antara dokumentasi perancangan dan perangkat lunak sinkron. Banyak berbagai kasus, perancangan kelas yang di buat

tidak sesuai dengan kelas-kelas yang dibuat pada perangkat lunak, sehingga tidaklah ada gunanya sebuah perancangan karena yang di rancang dan hasil jadinya tidak sesuai. Kelaskelas yang ada pada struktur sistem harus dapat melakukan fungsi-fungsi sesuai dengan kebutuhan sistem sehingga pembuat perangkat lunak atau programmer dapat membuat kelas-kelas di dalam program perangkat lunak sesuai dengan perancangan diagram kelas.

Susunan struktur kelas yang baik pada diagram kelas sebaiknya memiliki jenis jenis kelas berikut:

- a. Kelas main Kelas yang memiliki fungsi awal di eksekusi ketika sistem dijalankan.
- b. Kelas yang menangani tampilan sistem (*view*) Kelas yang definisikan dan mengatur tampilan ke pemakai
- c. Kelas yang diambil dari pendefinisian *use case* (*controller*) Kelas yang menangani fungsi- fungsi yang harus ada diambil dari pendefinisian use case, kelas ini biasanya disebut dengan kelas proses yang menangani proses bisnis pada perangkat lunak.
- d. Kelas yang diambil dari pendefinisian data (*model*) Kelas yang digunakan untuk memegang atau membungkus data menjadi sebuah kesatuan yang diambil maupun akan disimpan kelas.

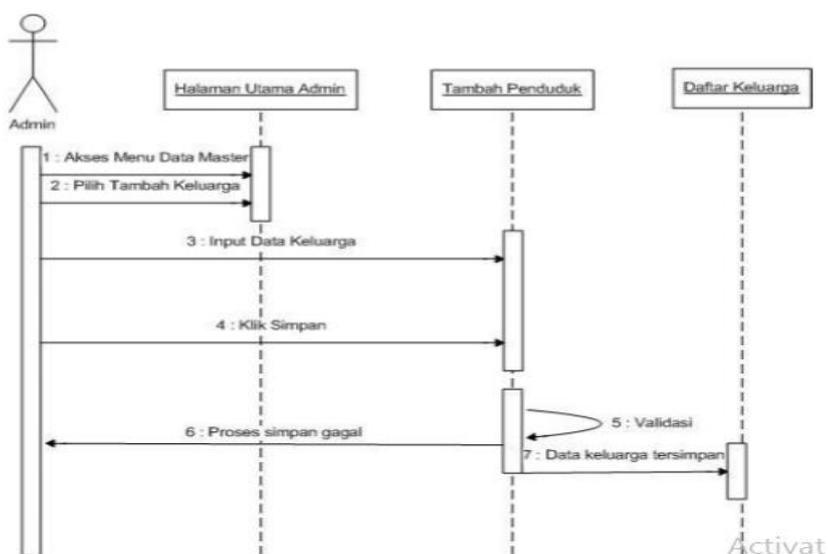


Sumber: (Shinta,2016)

Gambar II.6 Contoh gambar *class diagram*

5. *Sequence Diagram*

Menurut Sukamto dan Shalahuddin, dalam jurnal (Muzarli & Irawan, 2018:168) *Sequence diagram* atau bisa disebut juga dengan *diagram sequence*, yaitu menggambarkan sebuah kelakuan dari objek *use case* dengan mendeskripsikan sebagai waktu hidup objek dan message yang dikirim dan akan diterima antar objek–objek tersebut”. Dari sekian banyaknya diagram sekuen yang harus digambar adalah minimal sebanyak pendefinisian *use case* yang mempunyai proses tersendiri atau yang penting semua *use case* yang telah didefinisikan interaksi dengan jalannya pesan dan dicakup pada diagram sekuen sehingga banyak *use case* yang didefinisikan maka diagram sekuen yang harus dibuat dan lebih banyak sesuai apa yang dicakup



Sumber: (Shinta,2016)

Gambar II.7 Contoh gambar *sequence diagram*

