

BAB II

LANDASAN TEORI

2.1 Studi Literatur

Dalam penelitian ini penulis memaparkan tiga penelitian terdahulu yang relevan dengan permasalahan yang akan diteliti.

1. Aliyazid(2016), Berdasarkan pengamatan menggunakan model *Saw* dan *Topsis* dengan melakukan pergantian bobot maka kami melihat bahwa kedua model tersebut memiliki kesamaan dalam proses pemecahan masalah.
2. Mardiana, Tati (2017), Metode *Topsis* yang diterapkan untuk mendukung keputusan membuat penilaian layak atau tidaknya menerima pinjaman diambil berdasarkan penilaian yang objektif sehingga hasil yang didapat bisa lebih tepat dan terukur.
3. Setiadi, Ahmad(2018), Untuk menentukan siswa terbaik tidak cukup hanya dengan penilaian akademiknya saja tetapi juga memperhitungkan penilaian nonakademiknya.

2.2 Prestasi

Menurut Zaenal Arifin (2012:3), pada buku evaluasi pembelajaran. Prestasi adalah hasil dari kemampuan, keterampilan, dan sikap seseorang dalam menyelesaikan beberapa hal.

2.3 Pesantren

Dhofier, Zamakhsyari (2011), Pesantren bukanlah untuk mengejar kepentingan kekuasaan, uang dan keagungan duniawi, tetapi ditanamkan kepada mereka bahwa belajar adalah semata-mata kewajiban dan pengabdian kepada Tuhan. Oleh karena itu, sebagai salah satu lembaga pendidikan, pesantren juga mempunyai tanggung jawab dalam membentuk karakter para santri.

2.4 Sistem

Menurut Rosa dan Shalahuddin dalam Eva Meilinda (2016) mendefinisikan "Sistem adalah kegiatan untuk melihat sistem yang sudah berjalan, melihat bagaimana yang bagus dan tidak bagus, dan kemudian mendokumentasikan kebutuhan yang akan dipenuhi dalam sistem yang baru.

2.5 Aplikasi *Mobile*

Menurut Pressman dan Bruce (2014:9), aplikasi *mobile* adalah aplikasi yang telah dirancang khusus untuk *platform mobile* (misalnya iOS, android, atau windows *mobile*). Dalam banyak kasus, aplikasi *mobile* memiliki *user interface* dengan mekanisme interaksi unik yang disediakan oleh *platform mobile*, interoperabilitas dengan sumber daya berbasis web yang menyediakan akses ke beragam informasi yang relevan dengan aplikasi, dan kemampuan pemrosesan lokal untuk pengumpulan, analisis, dan format informasi dengan cara yang paling

cocok untuk *platform mobile*. Selain itu aplikasi *mobile* menyediakan kemampuan penyimpanan persisten dalam *platform*.

2.6 Perangkat lunak yang digunakan dalam membuat aplikasi

2.6.1 HTML

Menurut Solihin (2016:10), HTML merupakan singkatan dari Hyper Text Markup Language. HTML dikembangkan pertama kali oleh Tim Berners-Lee bersamaan dengan protokol HTTP (Hypertext Transfer Protocol) pada tahun 1989. Tujuan utama pengembangan HTML adalah untuk menghubungkan suatu halaman web dengan halaman web lainnya. Tentunya pada awal pengembangannya.

Dokumen yang berisi script HTML merupakan dokumen yang disajikan dalam bentuk *website*. Dokumen HTML disebut *markup language* karena mengandung tanda-tanda tertentu yang digunakan untuk menentukan tampilan suatu teks dan tingkat kepentingan dari teks tersebut dalam suatu dokumen. Kita dapat menentukan baris-baris mana yang merupakan judul menentukan gambar yang harus tampil, pengaturan format teks dan lain sebagainya.

2.6.2 CSS

Menurut Hidayatullah dan Kawistara (2017:83), “CSS berfungsi untuk membuat tampilan lebih rapih dan menarik. ”*Cascading Style Sheet (CSS)*

merupakan salah satu bahasa pemrograman web untuk mengendalikan beberapa komponen dalam sebuah web sehingga akan lebih terstruktur dan seragam. Sama halnya styles dalam aplikasi pengolahan kata seperti Microsoft Word yang dapat mengatur beberapa style, misalnya heading, subbab, bodytext, footer, images, dan style lainnya untuk dapat digunakan bersama-sama dalam beberapa berkas (file). Pada umumnya CSS dipakai untuk memformat tampilan web yang dibuat dengan bahasa HTML dan XHTML.

CSS dapat mengendalikan ukuran gambar, warna bagian tubuh pada teks, warna tabel, ukuran border, warna border, warna hyperlink, warna mouse over, spasi antar paragraf, spasi antar teks, margin kiri, kanan, atas, bawah, dan parameter lainnya. CSS adalah bahasa style sheet yang digunakan untuk mengatur tampilan dokumen. Dengan adanya CSS memungkinkan kita untuk menampilkan halaman yang sama dengan format yang berbeda.

Ada dua sifat CSS yaitu internal dan eksternal. Jika internal yang dipilih, maka skrip itu dimasukkan secara langsung ke halaman website yang akan didesain. Kalau halaman web yang lain akan didesain dengan model yang sama, maka skrip CSS itu harus dimasukkan lagi ke dalam halaman web yang lain itu. Sifat yang kedua adalah eksternal dimana skrip CSS dipisahkan dan diletakkan dalam berkas khusus. Nanti cukup gunakan semacam tautan menuju berkas CSS itu jika halaman web yang didesain akan dibuat seperti model yang ada di skrip tersebut.

2.6.3 PHP

Menurut Solihin (2016), PHP merupakan salah satu Bahasa pemrograman berbasis web yang ditulis oleh dan pengembang web. Ketika dipanggil dari web browser, program yang ditulis dengan PHP akan di-parsing didalam web server oleh interpreter PHP dan diterjemahkan dalam dokumen HTML, yang selanjutnya akan ditampilkan kembali ke web browser. Karena pemrosesan program PHP dilakukan dilingkungan web server, PHP dikatakan sebagai bahasa sisi server (server-side) oleh sebab itu, seperti yang telah dikemukakan sebelumnya, kode PHP tidak akan terlihat pada saat user memilih perintah “view source” pada web browser yang mereka gunakan. Selain menggunakan PHP, aplikasi web juga dapat dibangun dengan Java (JSP- JavaServer Pages dan Servlet), Perl, maupun ASP (Active Server Pages).

Syntax Program PHP adalah bahasa yang dirancang untuk mudah diletakkan dalam kode HTML. Banyak dijumpai kode PHP yang menyatu dengan kode HTML. Kode PHP diawali dengan tag `<?php` dan diakhiri dengan tag `?>`. apabila kita melakukan konfigurasi terhadap file PHP ini untuk mengizinkan pengguna tag pendek (short tag) dengan mengubah nilai `short_open_tag` menjadi on, maka tag tersebut dapat digantikan dengan `<? dan ?>`. dalam PHP, nilai default dari `short_open_tag` adalah off. Selain itu, PHP kita juga dapat menggunakan tag gaya ASP `<% dan %>`, dengan mengubah nilai `asp_tags` dalam file PHP ini menjadi on. Berikut ini contoh kode PHP :

```
<? Php  
  
echo "Hello World";  
  
?>
```

Perintah echo di dalam PHP berguna untuk mencetak nilai, baik teks maupun numerik ke layar web browser.

2.6.4 JavaScript

Menurut (R.H. Sianipar, 2015) javascript adalah bahasa scripting yang populer di internet dan dapat bekerja di sebagian besar browser populer seperti *Internet Explorer, Mozilla Firefox, Netscape dan Opera Mini*. Kode javascript dapat disisipkan dalam halaman web menggunakan tag script.

Berikut ini beberapa sifat dari javascript:

1. Menambahkan interaktivitas ke halaman HTML.
2. Merupakan bahasa pemrograman scripting.
3. Bahasa Scripting merupakan bahasa yang ringan.
4. Javascript merupakan bahasa terinterpretasi.

2.6.5 Bootstrap

Menurut (Husein Alatas, 2013) bootstrap merupakan *Framework* ataupun *Tools* untuk membuat aplikasi web ataupun situs web responsive secara cepat, mudah dan gratis. Bootstrap terdiri dari CSS dan HTML untuk menghasilkan *Grid, Layout, Typography, Table, Form, Navigation*, dan lain-lain. Di dalam Bootstrap juga sudah terdapat JQuery plugins untuk menghasilkan komponen UI yang cantik seperti *Transitions, Modal, Dropdown, Scrollspy, Tooltip, Tab, Popover, Alert, Button, Carousel* dan lain-lain.

2.6.6 JQuery

Menurut Becti (2015:59), JQuery merupakan suatu library JavaScript yang memungkinkan anda untuk membuat program web pada suatu halaman web tanpa harus menambahkan event atau pun property pada halaman web tersebut. Hal ini sesuai dengan slogannya “*Write less, do more*”, cukup tulis sedikit tapi bisa melakukan banyak hal.

2.6.7 MySQL

Menurut (R.H. Sianipar, 2015) dalam Buku “*Membangun Web dengan PHP & MYSQL untuk Pemula & Programmer*”. MySQL bukan termasuk bahasa pemrograman. MySQL merupakan salah satu database populer dan mendunia. MySQL bekerja menggunakan *SQL Language (Structure Query Language)*.

2.7 Observasi

Tersiana (2018: 12) mendefinisikan observasi yaitu proses pengamatan menyeluruh dan mencermati perilaku pada suatu kondisi tertentu. Pada dasarnya, observasi bertujuan untuk mendeskripsikan aktivitas, individu, serta makna kejadian berdasarkan perspektif individu.

2.8 Wawancara

Menurut Tersiana (2018: 12), wawancara merupakan suatu cara pengumpulan data penelitian dengan cara tanya-jawab secara langsung dengan subjek yang berkontribusi langsung dengan objek yang diteliti. Dalam pelaksanaan wawancara, agar pokok-pokok permasalahan terarah, penulis menggunakan pedoman wawancara semi terstruktur.

2.9 Waterfall

Disiplin ilmu yang penulis pilih sebagai dasar untuk menentukan pembuatan aplikasi berdasarkan model pengembangan perangkat lunak yaitu model air terjun (waterfall). Menurut Sukamto dan Shalahuddin (2015:28), “Model air terjun menyediakan pendekatan alur hidup perangkat lunak secara 16 sekuensial atau terurut dimulai dari analisis, desain pengodean, pengujian dan tahap pendukung (support)”.

2.10 Sistem Pendukung Keputusan

Konsep Sistem Pendukung Keputusan (SPK) pertama kali diungkapkan pada awal tahun 1970-an oleh Michael S. Scott Morton dengan istilah Management Decision System. Sistem tersebut adalah suatu sistem yang berbasis komputer yang ditunjukkan untuk membantu pengambil keputusan dengan memanfaatkan data dan model tertentu untuk memecahkan berbagai persoalan yang tidak terstruktur (Dewanto, 2015).

2.11 TOPSIS

TOPSIS (Technique For Order Preference by Similarity to Ideal Solution) diperkenalkan pertama kali oleh Yoon dan Hwang pada tahun 1981 untuk digunakan sebagai salah satu metode dalam memecahkan masalah multikriteria. *TOPSIS* mengasumsikan bahwa setiap kriteria akan dimaksimalkan ataupun diminimalkan. Maka dari itu nilai solusi ideal positif dan solusi ideal negatif dari setiap kriteria ditentukan, dan setiap alternatif dipertimbangkan dari informasi tersebut. Solusi ideal positif didefinisikan sebagai jumlah dari seluruh nilai terbaik yang dapat dicapai untuk setiap atribut, sedangkan solusi ideal negatif terdiri dari seluruh nilai terburuk yang dicapai untuk setiap atribut. Namun, solusi ideal positif jarang dicapai ketika menyelesaikan masalah dalam kehidupan nyata. Maka asumsi dasar dari *TOPSIS* adalah ketika solusi ideal positif tidak dapat dicapai, pembuat keputusan akan mencari solusi yang sedekat mungkin dengan

solusi ideal positif. *TOPSIS* memberikan solusi ideal positif yang relatif dan bukan solusi ideal positif yang absolut. Dalam metode *TOPSIS* klasik, nilai bobot dari setiap kriteria telah diketahui dengan jelas. Setiap bobot kriteria ditentukan berdasarkan tingkat kepentingannya menurut pengambil keputusan. Berikut adalah langkah-langkah dari *TOPSIS*:

1. Rangkaian Tiap Alternatif *TOPSIS* membutuhkan rating kinerja setiap alternatif A_i pada setiap kriteria C_j yang ternormalisasi, yaitu:

$$r_{ij} = \frac{x_{ij}}{\sqrt{\sum_{i=1}^m x_{ij}^2}}$$

dengan $i=1,2,\dots,m$; dan $j=1,2,\dots,n$;

2. Matriks keputusan ternormalisasi terbobot

$$y_{ij} = w_j r_{ij}$$

dengan $i=1,2,\dots,m$; dan $j=1,2,\dots,n$;

3. Solusi Ideal Positif Dan Negatif

Solusi ideal positif A^+ dan solusi ideal negatif A^- dapat ditentukan berdasarkan ranking bobot ternormalisasi (y_{ij}) sebagai berikut :

$$A^+ = (y_1^+, y_2^+, \dots, y_n^+)$$

$$A^- = (y_1^-, y_2^-, \dots, y_n^-)$$

Dengan

$$y_j^+ = \begin{cases} \max_i y_{ij}; & \text{Jika } j \text{ adalah atribut keuntungan;} \\ \min_i y_{ij}; & \text{Jika } j \text{ adalah atribut biaya;} \end{cases}$$

$$y_j^- = \begin{cases} \min_i y_{ij}; & \text{Jika } j \text{ adalah atribut keuntungan;} \\ \max_i y_{ij}; & \text{Jika } j \text{ adalah atribut biaya;} \end{cases}$$

4. Jarak Dengan Solusi Ideal

Jarak antara alternatif A_i dengan solusi ideal positif dirumuskan sebagai:

$$D_i^+ = \sqrt{\sum_{j=1}^n (y_i^+ - y_{ij})^2}; \quad \text{dengan } i=1,2,\dots,m;$$

Jarak antara alternatif A_i dengan solusi ideal negatif dirumuskan sebagai:

$$D_i^- = \sqrt{\sum_{j=1}^n (y_{ij} - y_i^-)^2}; \quad \text{dengan } i=1,2,\dots,m;$$

5. Nilai Preferensi Untuk Setiap Alternatif

Nilai preferensi untuk setiap alternatif (V_i) diberikan sebagai:

$$V_i = \frac{D_i^-}{D_i^- + D_i^+}; \quad \text{dengan } i=1,2,\dots,m;$$

Nilai V_i yang lebih besar menunjukkan bahwa alternatif A_i lebih dipilih.

2.12 SAW

SAW (Metode Simple Additive Weighting) sering juga dikenal istilah metode penjumlahan terbobot. Konsep dasar metode SAW adalah mencari penjumlahan terbobot dari rating kinerja pada setiap alternatif pada semua atribut (Fishburn, 1967) (MacCrimmon, 1968). Metode SAW membutuhkan proses normalisasi matriks keputusan (X) ke suatu skala yang dapat diperbandingkan dengan semua rating alternatif yang ada. Metode ini merupakan metode yang paling terkenal dan paling banyak digunakan dalam menghadapi situasi Multiple Attribute Decision Making (MADM).

MADM itu sendiri merupakan suatu metode yang digunakan untuk mencari alternatif optimal dari sejumlah alternatif dengan kriteria tertentu. Metode SAW ini mengharuskan pembuat keputusan menentukan bobot bagi setiap atribut. Skor total untuk alternatif diperoleh dengan menjumlahkan seluruh hasil perkalian antara rating (yang dapat dibandingkan lintas atribut) dan bobot tiap atribut. Rating tiap atribut haruslah bebas dimensi dalam arti telah melewati proses normalisasi matriks sebelumnya.

Langkah – langkah penyelesaian menggunakan metode SAW:

1. Menentukan kriteria – kriteria yang akan dijadikan acuan dalam mengambil keputusan yaitu C_i .
2. Menentukan rating kecocokan setiap alternatif pada setiap kriteria.
3. Membuat matrik keputusan berdasarkan kriteria (C_i), kemudian melakukan normalisasi matrik berdasarkan persamaan yang disesuaikan dengan jenis atribut

(atribut keuntungan ataupun atribut biaya) sehingga diperoleh matrik ternormalisasi R.

Formula untuk melakukan normalisasi tersebut adalah :

$$R_{ij} = \begin{cases} \frac{x_{ij}}{\text{Max}_i x_{ij}} & \text{Jika } j \text{ adalah atribut keuntungan;} \\ \frac{\text{Min}_i x_{ij}}{x_{ij}} & \text{Jika } j \text{ adalah atribut biaya;} \end{cases}$$

Keterangan :

- a. R_{ij} = rating kinerja ternormalisasi.
- b. Max_i = nilai terbesar dari setiap kriteria.
- c. Min_i = nilai terkecil dari setiap kriteria.
- d. X_{ij} = nilai atribut yang dimiliki.
- e. Benefit = jika nilai terbesar adalah terbaik.
- f. Cost = jika nilai terkecil adalah terbaik.

Nilai preferensi untuk setiap alternative (V_i) diberikan sebagai :

$$V_i = \sum_{j=1}^n W_j R_{ij}$$

Keterangan :

- a. V_i = ranking untuk setiap alternatif.
- b. W_j = nilai bobot dari setiap kriteria.

2.13 UML (*Unified Modelling Language*)

Sukamto dan Shalahuddin (2014:137) menjelaskan bahwa UML (Unified Modeling Language) adalah standarisasi bahasa pemodelan untuk pembangunan perangkat lunak yang dibangun menggunakan teknik pemograman 15 berorientasi objek. UML (Unified Modeling Language) muncul karena adanya kebutuhan pemodelan visual untuk menspesifikasi, menggambarkan, membangun, dan dokumentasi dari sistem perangkat lunak

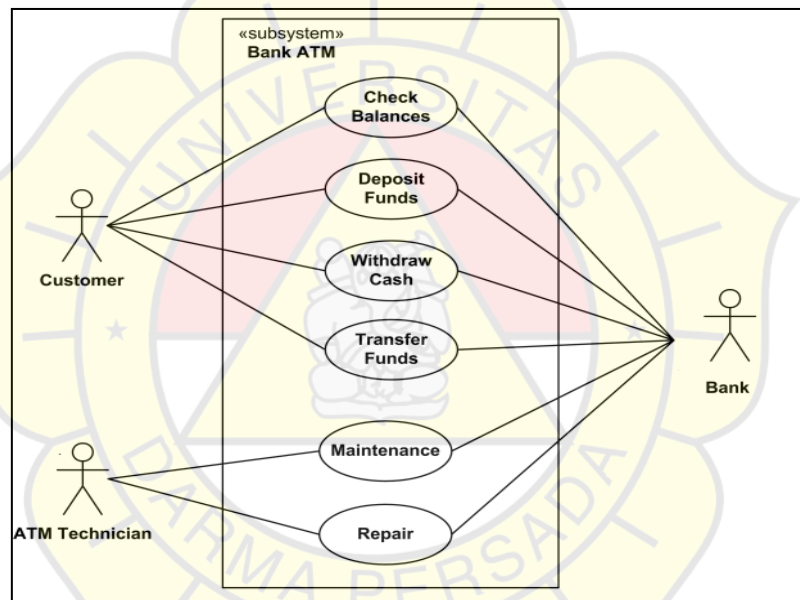
Berdasarkan pendapat yang dikemukakan di atas dapat ditarik kesimpulan bahwa UML adalah sebuah bahasa yang berdasarkan grafik atau gambar untuk memvisualisasikan, menspesifikasikan, membangun dan pendokumentasian dari sebuah sistem pengembangan perangkat lunak berbasis Objek (*Object Oriented programming*).

2.13.1 Use Case Diagram

Menurut Sukamto dan Shalahuddin (2014:155) berpendapat use case diagram merupakan pemodelan untuk kelakuan (behavior) sistem informasi yang akan dibuat, use case diagram digunakan untuk mengetahui fungsi apa saja yang ada di dalam sistem dan yang berhak menggunakan fungsi-fungsi tersebut.

Yang ditekankan dalam *use case* diagram adalah “apa” yang diperbuat sistem, dan bukan “bagaimana”. Secara umum, *use case* diagram terdiri dari tiga bagian antara lain :

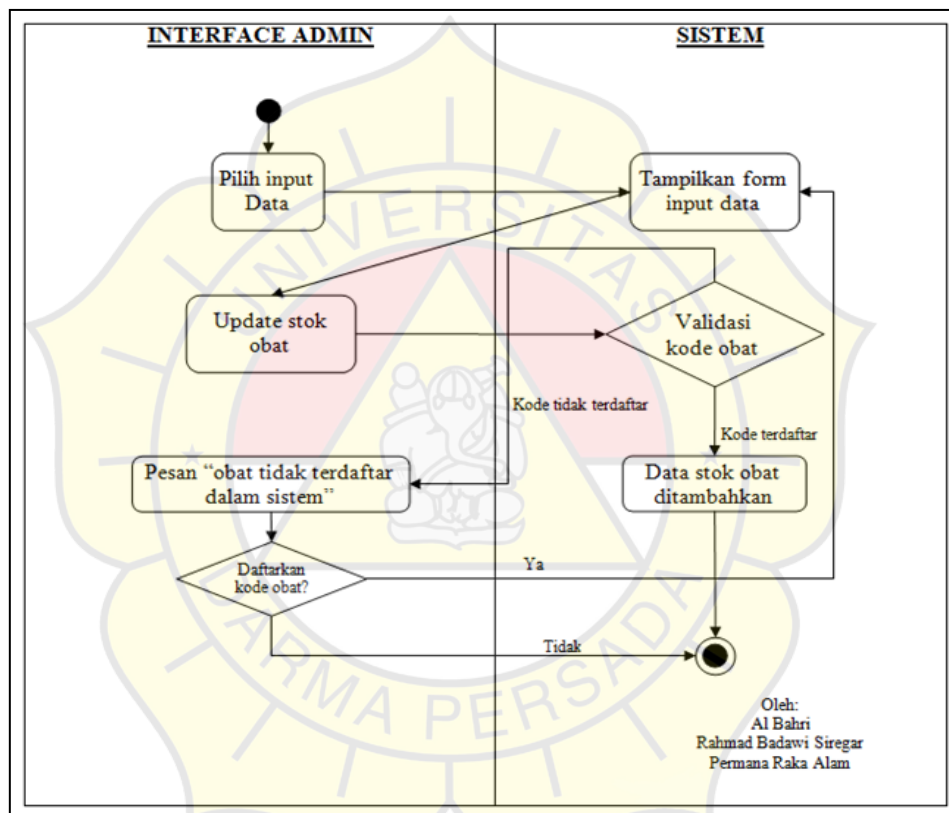
1. Aktor, adalah sebuah peran dari *user*, subsystem, atau piranti yang berinteraksi dengan sistem. Aktor digambarkan dengan figur 'stick'.
2. *Use case* menyediakan aktor rangkaian aksi yang bisa dilakukan. *Use case* direpresentasikan dengan bentuk oval yang digambarkan secara horizontal.
3. *Association* adalah hubungan dua atau lebih *classifier*, yaitu hubungan koneksi yang terdapat diantara instansi-instansi yang ada dan digambarkan dengan menggunakan garis.



Gambar 2.1 *Use Case Diagram* (Prabowo Pudjo Widodo, 2011)

2.13.2 Activity Diagram

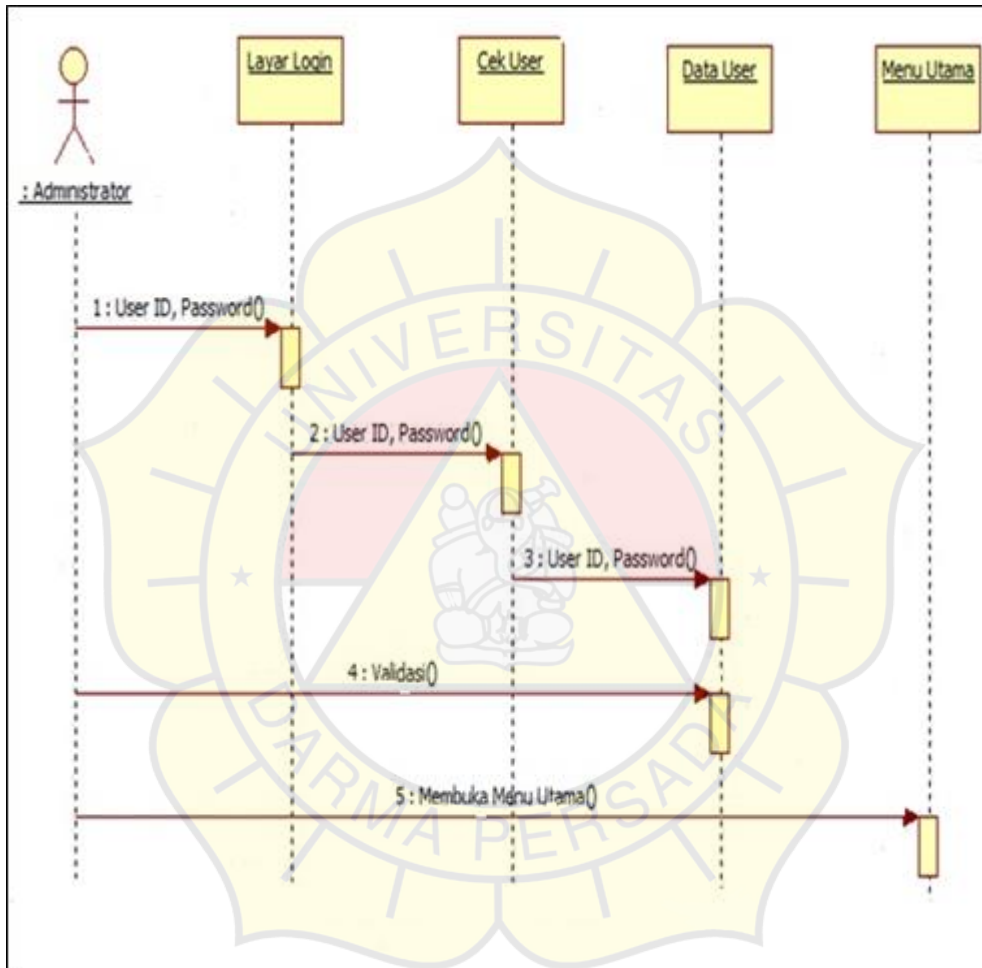
Menurut Sukamto dan Shalahuddin (2014:161) diagram aktivitas atau activity diagram adalah menggambarkan aliran kerja atau aktifitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Diagram aktifitas menggambarkan aktifitas sistem bukan apa yang dilakukan oleh aktor.



Gambar 2.2 Activity Diagram (Prabowo Pudjo Widodo, 2011)

2.13.3 Sequence Diagram

Sukamto dan Shalahuddin (2014:165) berpendapat sequence diagram menggambarkan kelakuan objek pada use case dengan mendeskripsikan waktu hidup objek dan pesan yang dikirimkan dan diterima antar objek



Gambar 2.3 Sequence Diagram (Prabowo Pudjo Widodo, 2011)