

BAB II

LANDASAN TEORI

2.1 Kajian Terhadap Penelitian Terkait Sebelumnya

2.1.1. Metode K-Means

K-Means merupakan algoritma *clustering* yang berulang-ulang. Algoritma K-Means dimulai dengan pemilihan secara acak N , N disini merupakan banyaknya *cluster* yang ingin dibentuk. Kemudian tetapkan nilai-nilai N secara random, untuk sementara nilai tersebut menjadi pusat dari cluster atau biasa disebut dengan centroid, mean atau “means”. Hitung jarak setiap data yang ada terhadap masingmasing centroid menggunakan rumus Euclidian hingga ditemukan jarak yang paling dekat dari setiap data dengan centroid. Lakukan langkah tersebut hingga nilai centroid tidak berubah, “Teknik Data Meaning non hirarki yang mengelompokan data ke dalam K cluster, dengan K adalah jumlah cluster yang ditentukan sebelumnya, melalui penentuan titik pusat cluster (centroid) yang bertujuan untuk meminimalkan kesalahan kuadrat (SSE) dan meminimalkan variasi data di dalam setiap cluster” Santosa, 2019).

Dari berberapa teknik *clustering* yang paling sederhana dan umum dikenal adalah *clustering* K-Means. Dalam teknik ini kita ingin mengelompokan obyek kedalam N kelompok atau *cluster*. Untuk melakukan *clustering*, nilai N harus ditentukan terlebih dahulu. Biasanya user atau pemakai sudah mempunyai informasi awal tentang obyek yang sedang dipelajari, termasuk berberapa jumlah *cluster* yang paling tepat. Secara detail kita bias menggunakan ukuran ketidakmiripan untuk mengelompokkan obyek kita. Ketidakmiripan bisa diterjemahkan dalam konsep jarak. Jika jarak dua obyek atau dua titik cukup dekat maka dua obyek itu mirip. Semakin dekat berarti semakin tinggi kemiripannya. Semakin tinggi jarak semakin tinggi ketidakmiripannya.

Dalam penelitiannya mengungkapkan langkah-langkah pengerjaan algoritma K-Means (Santosa, 2007) yaitu :

1. Penentuan pusat *cluster* awal

Dalam penentuan nilai N buah pusat *cluster* awal dilakukan pembangkitan bilangan *random* yang mempresentasikan urutan data input pusat awal *cluster* didapatkan dari data sendiri bukan dengan menentukan titik baru, yaitu dengan merandom pusat awal dari data.

2. Perhitungan jarak dengan pusat *cluster*

Untuk mengukur jarak antara data dengan pusat *cluster* digunakan rumus *Euclidian distance*. Algoritma perhitungan jarak data dengan pusat *cluster*

- a. Ambil nilai data dan nilai pusat *cluster*.
- b. Hitung *Euclidian distance* data dengan tiap pusat *cluster*.

Rumus Euclidian distance

$$d(x, y) = [x - y] = \sqrt{\sum_i^n 1(x_i - y_i)^2}$$

Keterangan:

d = jarak antara x dan y

x = data pusat klaster

y = data pada atribut

i = setiap data

n = jumlah data

x_i = data pada pusat klaster ke i

y_i = data pada setiap data ke i

1. Pengelompokan data

Jarak hasil perhitungan akan dilakukan perbandingan dan dipilih jarak terdekat antara data dengan pusat *cluster*, jarak ini akan menunjukkan data tersebut berada dalam suatu kelompok dengan pusat *cluster* terdekat. Algoritma pengelompokan data :

- a. Ambil nilai jarak tiap pusat *cluster* dengan data.
- b. Cari nilai jarak terkecil.
- c. Kelompokkan data dengan pusat *cluster* yang memiliki jarak terkecil.

Rumus Cluster :

Hitung rata-rata jarak dari dokumen 1 di cluster di ambil nilai terkecilnya

$$d(i, C) = \frac{1}{|A|} \sum_{j \in C} d(i, j) [6]$$

dengan $d(i, C)$ = Jarak rata - rata dokumen i dengan semua objek pada cluster lain dimana $A \neq C$, $b(i) = \min_{C \neq A} d(i, C)$

2. Penentuan pusat *cluster* baru

Untuk mendapatkan pusat *cluster* baru bisa dihitung dari rata-rata nilai anggota *cluster* dan pusat *cluster*. Pusat *cluster* yang baru digunakan untuk melakukan iterasi selanjutnya, jika hasil yang didapatkan belum konvergen. Proses itersi ini akan berhenti jika telah memenuhi maksimum iterasi yang dimasukkan oleh user atau hasil dicapai sudah konvergen (pusat *cluster* baru sama dengan pusat *cluster* lama).

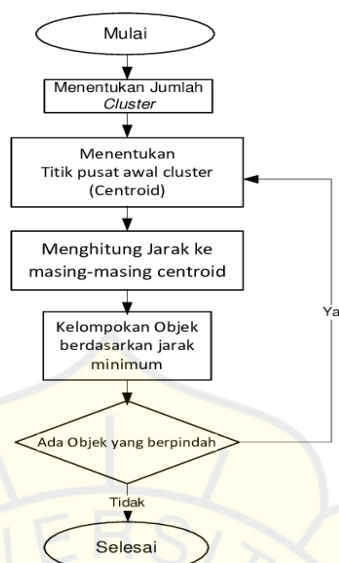
Algoritma penentuan pusat *cluster* baru yaitu:

- a. Cari jumlah anggota tiap *cluster*.
- b. Hitung pusat baru dengan rumus.

Pusat *cluster* baru $= \frac{x_1 + x_2 + x_3 + \dots + x_n}{\text{jumlah} + 1}$

Dimana :

$x_1, x_2, x_3, \dots, x_n$ adalah anggota *cluster*.



Gambar 2. 1 Flowchart Algoritma K-Means

(Artikel: <https://www.researchgate.net/>)

2.2 Website

2.2.1. HTML

Hypertext Markup Language (HTML) merupakan bahasa dasar *web* yang berfungsi untuk menampilkan berbagai komponen *web*. HTML dikembangkan pertama kali oleh Tim Berners. Menurut Jurnal yang di tulis oleh Achmad Solichin (2019) tujuan utama pengembangan HTML adalah untuk menghubungkan satu halaman *web* dengan halam *web* lainya. Dengan kata lain HTML adalah fondasi *web*. HTML disusun dengan bahasa yang sederhana, sehingga sangat mudah diimplementasikan.

Kode HTML yang dibuat nantinya akan diterjemahkan *web browser* supaya bisa tampil seperti apa yang sudah dirancang. Sebenarnya, semua *web browser* bisa menampilkan kode HTML dengan baik, akan tetapi jika berbicara tentang desain halaman, maka setiap *browser* tentu memiliki beberapa perbedaan.

HTML memang dirancang serta diatur badan standarisasi dunia khusus yang menangani *web* yakni *World Wide Web Consortium* [W3C]. Ini disebabkan karena masing masing program *web browser* akan menerjemahkan kode HTML dengan berbeda sehingga dibutuhkan standar yang sama untuk semua *browser*.

2.2.2. CSS

Cascading Style Sheets (CSS) merupakan salah satu kode pemrograman yang bertujuan untuk menghias dan mengatur gaya tampilan halaman *web* supaya lebih elegan dan menarik. CSS adalah teknologi internet yang direkomendasikan *world wide web consortium* (W3C) pada tahun 1996. CSS juga digunakan oleh *web programmer* dan juga *web designer* untuk menentukan warna, tata letak font, dan semua aspek lain dari presentasi dokumen disitus mereka menurut Jurnal yang di tulis oleh (Didik Setiawan, 2018). 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.

2.2.3. PHP

Supono dan Putratama mengemukakan 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 berbasis server-side yang dapat ditambahkan ke dalam HTML menurut jurnal yang di tulis oleh (Saputra, Agus. 2018).

2.2.4. JavaScript

Menurut Jurnal yang di tulis oleh (R.H. Sianipar, 2018) 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 :

- Menambahkan interaktivitas ke halaman HTML.
- Merupakan bahasa pemrograman scripting.
- Bahasa Scripting merupakan bahasa yang ringan.
- Javascript merupakan bahasa terinterpretasi.

2.2.5. Bootstrap

Bootstrap adalah *framework front-end* yang intuitif dan *powerful* untuk pengembangan aplikasi *web* yang lebih cepat dan mudah. *Bootstrap* menggunakan HTML, CSS, dan *Javascript*. *Bootstrap* memiliki fitur-fitur komponen *interface* yang bagus seperti *Typografi, Forms, Buttons, Navigations, Dropdowns, Alerts, Modals, Tabs, Accordion, Carousel*, dan lain sebagainya. Dengan demikian dalam membuat *website* kita bisa menghemat waktu, fitur yang *responsive*, dan memiliki *design* yang konsisten menurut Jurnal yang di tulis oleh (Gregorius Agung, 2018). *Bootstrap* telah menyediakan kumpulan aturan dan komponen *class interface* dasar sebagai modal dalam pembuatan *web* yang telah dirancang sangat baik untuk memberikan tampilan yang sangat menarik, bersih, ringan dan memudahkan bagi penggunanya. Dan penggunaan *bootstrap* ini kita juga diberikan keleluasan selama pengembangan *website*, anda bisa merubah dan menambah *class* sesuai dengan keinginan. *Bootstrap* memberikan kemudahan bagi anda, dengan menggunakannya dapat memangkas waktu, tenaga dalam proses pengerjaan suatu *website*. Kita selalu dituntut melakukan

pekerjaan apapun dengan efisien dan efektif, dengan demikian penggunaan *framework twitter bootstrap* ini bisa anda pilih ketika membuat suatu *website* bagi anda maupun klien anda.

2.2.6. SQL

Menurut Jurnal yang di tulis oleh (R.H. Sianipar, 2018) 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)*. Pada umumnya, perintah yang paling sering digunakan dalam MySQL adalah *SELECT* (mengambil), *INSERT* (menambah), *UPDATE* (mengubah), dan *DELETE* (menghapus). Selain itu, SQL juga menyediakan perintah untuk membuat *database, field*, ataupun *index* untuk menambah atau menghapus data jurnal (Sianipar, R.H. 2015).

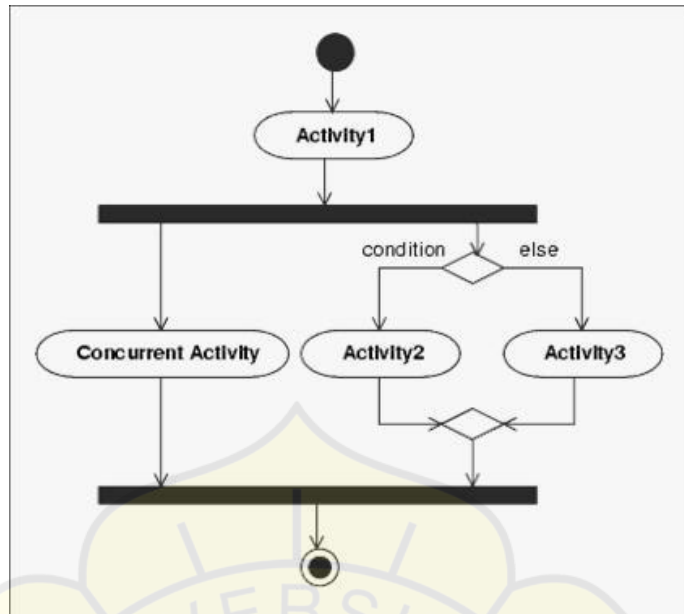
2.3 Pemodelan Sistem

Unified Modelling Language (UML) adalah sebuah bahasa yang telah menjadi standar dalam industri untuk evaluasi, merancang, dan mendokumentasikan sistem peranti lunak. UML menawarkan sebuah standar untuk merancang model sebuah sistem. UML juga dapat digunakan untuk aplikasi modeling procedural seperti VB atau C. jurnal (Yuni Sugiarti, 2018).

2.3.1. UseCase Diagram

UseCase diagram merupakan pemodelan untuk menggambarkan behavior dan mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem yang akan dibuat. Jurnal (Yuni Sugiarti,2018).

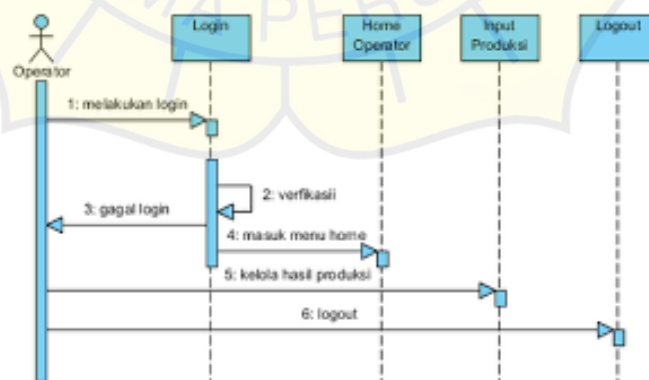
2.3.2. Activity Diagram



Gambar 2. 2 Activity Diagram Sumber : Jurnal (Yuni Sugiarti,2018)

Dari gambar 2.2 Activity diagram menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis. Hal yang perlu diperhatikan di sini adalah bahwa diagram aktivitas menggambarkan kegiatan sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem. (Yuni Sugiarti,2018).

2.3.3. Sequence Diagram



Gambar 2.3 Sequence Diagram Sumber Jurnal (Yuni Sugiarti,2018)

Dari gambar 2.3 Diagram sekuens (*sequence*) menggambarkan behavior objek pada *Usecase* dengan mendeskripsikan waktu hidup dan message yang dikirimkan dan

diterima antarobjek. Banyaknya diagram sekuens yang harus digambar adalah sebanyak pendefinisian *Usecase* yang memiliki proses sendiri atau yang penting semua *Usecase* telah didefinisikan interaksi jalannya pesan sudah dicakup pada diagram sekuens. (Yuni Sugiarti,2018)

