

## BAB IV. ANALISIS DAN PERANCANGAN SISTEM

### 4.1 Analisa Kebutuhan Sistem

Analisis kebutuhan digunakan untuk mengetahui apa saja yang akan digunakan dalam proses perancangan sistem. Berdasarkan hasil analisa terhadap kebutuhan dalam perancangan sistem yang sedang dikembangkan dalam penelitian ini dibutuhkan *hardware* dan *software*. Kebutuhan *hardware* dan kebutuhan *software* dapat dilihat pada Tabel 4.1 sedangkan kebutuhan *hardware* dapat dilihat pada Tabel 4.2 dan kebutuhan *software* dapat dilihat pada Tabel 4.3

Tabel 4.1 Kebutuhan Hardware dan Software

<i>Hardware</i>	<i>Software</i>	Bahasa Pemrograman
Laptop	Visual Studio Code 2019	PHP (CI)
	XAMPP	

#### 4.1.1 Kebutuhan *Hardware*

Untuk dapat melakukan penelitian pada sistem penerapan metode *K-Medoid* pada analisis respon emosi wanita jawa berikut adalah kebutuhan *software* :

Tabel 4.2 Kebutuhan Hardware

<i>Hardware</i>	Keterangan
Laptop	Prosesor : Intel Core i5 10 <sup>TH</sup> GEN CPU : 1.60 GHz 2.11 GHz RAM : 8 GB

### 4.1.2 Kebutuhan *Software*

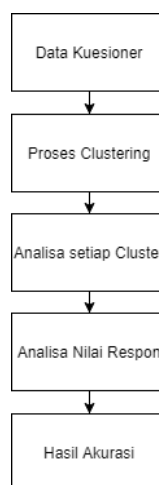
Untuk dapat melakukan penelitian pada sistem sistem penerapan metode *K-Medoid* pada analisis respon emosi wanita jawa dapat dilihat pada Tabel 4.3:

Tabel 4.3 Kebutuhan Software

NO.	<i>Software</i>	Keterangan
1.	Visual Studio Code	Menggunakan Visual Studio Code tahun 2019
2.	PHP	Sebagai Bahasa Pemrograman.
3.	<i>Web Server</i>	Apache
4.	<i>Database</i>	MySQL
5.	Sistem Operasi	Windows 10

### 4.2 Desain Sistem

Desain Sistem digunakan untuk gambaran umum dalam proses pengambilan *input* data sampai *output* data yang dihasilkan oleh sistem. Desain Sistem dapat dilihat pada Gambar 4.1



Gambar 4.1 Desain Sistem Clustering

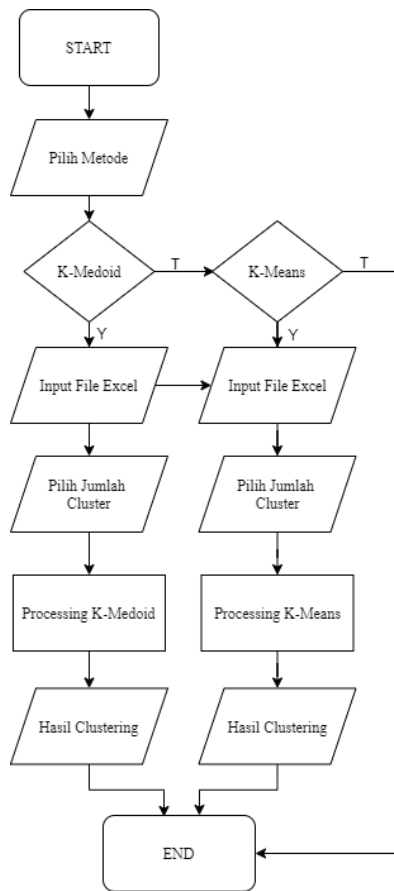
Berdasarkan gambar 4.1 blok diagram sistem, maka langkah-langkah yang dilakukan pada sistem penerapan metode *K-Medoid* pada *clustering* analisis respon marah wanita jawa adalah sebagai berikut.

- a. Data kuesioner akan menjadi data yang akan diproses yaitu dari nilai setiap responden pada jawaban kuesioner respon emosi.
- b. Kemudian setelah mendapatkan data yang sudah didapat, proses *clustering* akan berlangsung dan menghasilkan *cluster* sesuai dengan jumlah *cluster* yang dipilih.
- c. Selanjutnya dilakukan analisa anggota apa saja yang ada dalam *cluster* tersebut.
- d. Pada anggota *cluster* dilakukan analisa jawaban respon emosi pada setiap aspek untuk mengetahui berapa persentase tingkat kemarahannya.
- e. Terakhir, pada hasil *clustering* dilakukan perhitungan akurasi untuk melihat berapa nilai akurasi *clustering* yang didapatkan pada metode *K-Medoid*.

## **4.2 Perancangan Sistem**

### **4.3.1 Flowchart**

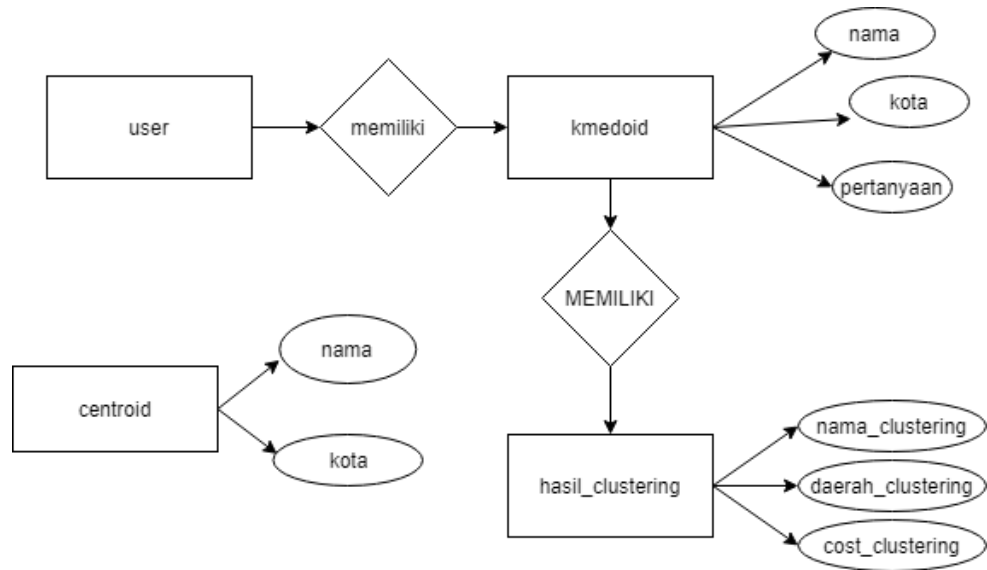
Pada tahap perancangan sistem dilakukan dengan mengupload file excel yang sudah di *export* sebelumnya melalui web kuesioner yang telah diisi sebelumnya. Lalu dilakukan proses perhitungan dengan menggunakan metode yang di inginkan, pada sistem terdapat dua buah metode, antara lain : *K-Medoid* dan *K-Means Clustering*. Hal tersebut akan menghasilkan keluaran berupa hasil *clustering* pada kumpulan data. Secara keseluruhan, proses alur jalannya sistem dapat dilihat pada Gambar 4.2.



Gambar 4.2 Flowchart Sistem Clustering

Pada *flowchart* tersebut merupakan cara penulisan algoritma dengan menggunakan notasi garis. Flowchart merupakan gambar atau bagan yang memperlihatkan urutan atau langkah-langkah dari suatu program dan hubungan antar proses beserta pernyataannya. Gambaran ini dinyatakan dengan simbol. Dengan demikian setiap simbol menggambarkan proses tertentu. Sedangkan antara proses digambarkan dengan garis penghubung. Dengan menggunakan flowchart akan memudahkan kita untuk melakukan pengecekan bagian-bagian yang terlupakan dalam analisis masalah. Disamping itu flowchart juga berguna sebagai fasilitas untuk berkomunikasi antara pemrograman yang bekerja dalam tim suatu proyek. Flowchart menolong analis dan programmer untuk memecahkan masalah ke dalam segmen-segmen yang lebih kecil dan menolong dalam menganalisis alternatif-alternatif lain dalam pengoperasian (Ali & al., 2013).

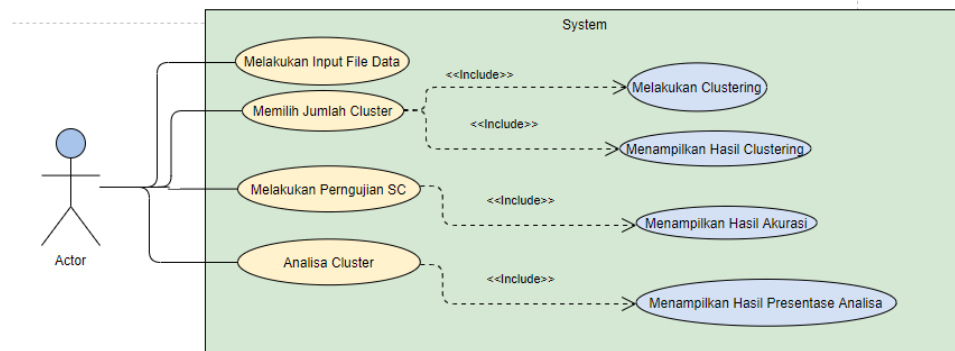
### 1.3.2 ERD



Gambar 4.3 ERD *K-Medoid Clustering*

Jadi, pada gambar 4.3 merupakan ERD atau *Entity Relationship Diagram* merupakan suatu bentuk diagram yang menjelaskan hubungan antar objek-objek data yang mempunyai hubungan. ERD disini digunakan untuk menyusun struktur data dan hubungan antar data. Pada sistem analisis respon marah wanita jawa dengan menggunakan metode *K-Medoid* ini tergambar ERD yaitu user memiliki metode *K-Medoid* yang mana *K-Medoid* memiliki atribut nama, kota dan juga pertanyaan-pertanyaan yang nantinya *K-Medoid* akan memiliki sebuah hasil *clustering* yang terdiri dari atribut nama *clustering*, daerah *clustering* dan *cost clustering*. Pada metode ini juga memiliki *centroid* yang mana *centroid* nya memiliki atribut nama dan kota

### 1.3.3 Use case Diagram



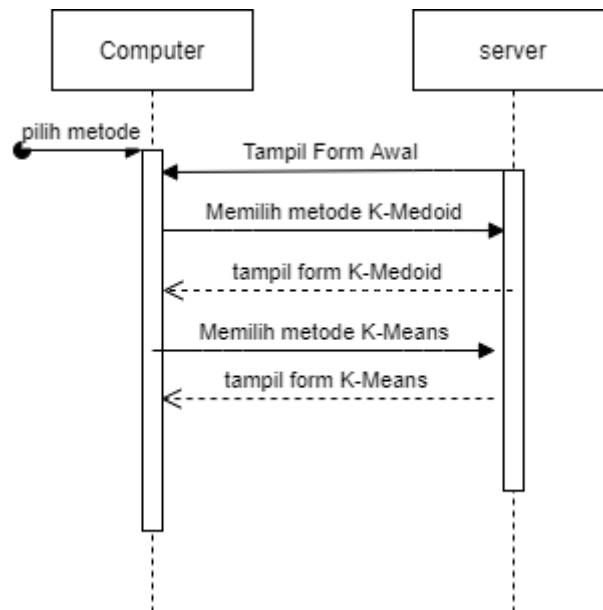
Gambar 4.4 Use case Diagram Clustering

Pada gambar 4.4 menggambarkan sebuah *use case* diagram *clustering* yang mana digunakan untuk memodelkan unit fungsi/layanan yang disediakan oleh sistem. Diagram *use case* tidak menjelaskan secara detail tentang penggunaan *use case*. Namun hanya memberi gambaran singkat hubungan antara *use case*, aktor dan sistem. Pada gambar tersebut pengguna / *user* dapat melakukan *input file* data yang berasal dari kuesioner yang sudah di isikan sebelumnya oleh responden. *User* juga dapat memilih jumlah *cluster*, dimana berdasarkan jumlah *cluster* ini akan dilakukan proses *clustering* dan lalu akan menampilkan hasil *clustering* nya. Setelah itu, *user* juga dapat melakukan pengujian keakuratan sistem dengan menggunakan *silhouette coefficient* yang mana untuk menguji dan nanti akan mengeluarkan hasil akurasinya, bagaimana metode *k-medoid* ini diterapkan dalam sistem akurasi, apakah metode yang digunakan baik atau buruk menurut strukturnya. Selain itu, *user* juga dapat melakukan analisis *cluster* yang mana akan menampilkan hasil persentase analisa berdasarkan tiap-tiap aspek pemicu marah wanita jawa berdasarkan tingkat kemarahannya.

### 1.3.4 Sequence Diagram

*Sequence* diagram *clustering* yang mana merupakan jenis diagram interaksi karena menggambarkan “Bagaimana” dan dalam urutan “Apa” sekelompok objek bekerja sama. Berikut beberapa proses *clustering* yang sudah digambarkan melalui *sequence* diagram.

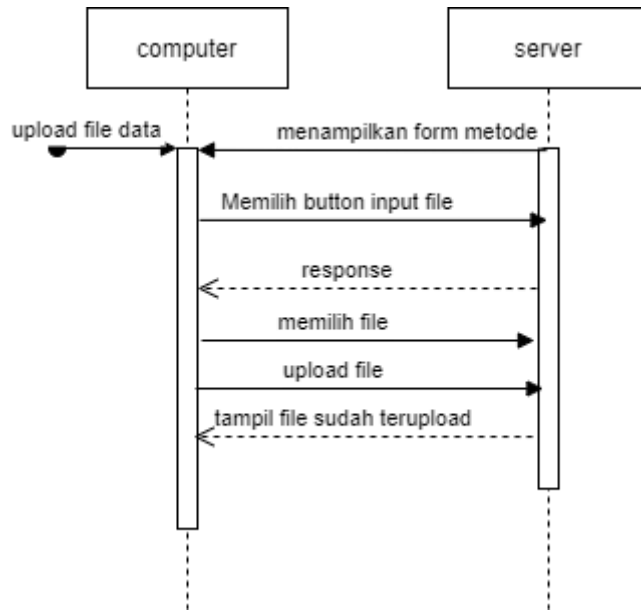
#### 1.3.4.1 Sequence Diagram Pilih Metode



Gambar 4.5 *Sequence* Diagram Pilih Metode

Pada Gambar 4.5 dijelaskan bahwa langkah dalam pemilihan metode, berawal dari komputer yang akan memilih metode disuguhkan terlebih dahulu oleh server tampilan dari form awal yang berupa tampilan ingin memilih metode apa. Lalu komputer melakukan pemilihan metode *K-Medoid* setelah itu server akan merespon dan menampilkan *form K-Medoid*. Lalu komputer bisa secara langsung memilih *form K-Means* lalu server akan merespon dan menampilkan *form K-Means*.

#### 1.3.4.2 Sequence Diagram *Input File Data*

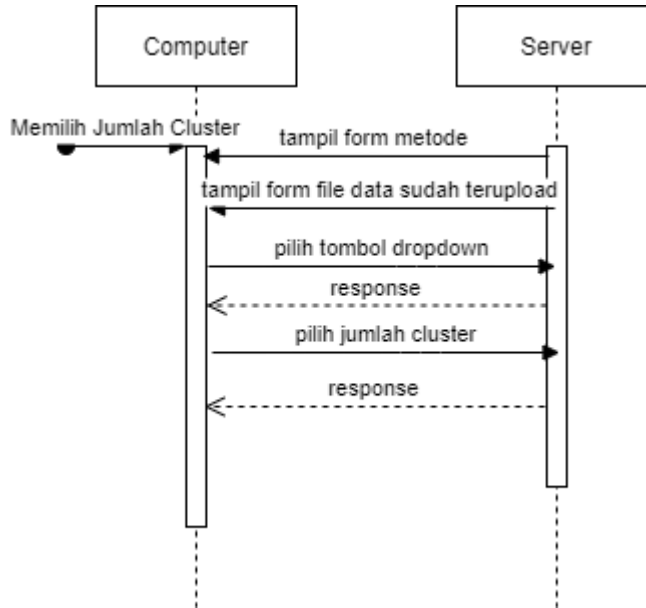


Gambar 4.6 *Sequence Diagram Input File Data*

Pada Gambar 4.6 dijelaskan bahwa langkah dalam *Input File Data*, berawal dari komputer yang akan *Input File Data* disuguhkan terlebih dahulu oleh server tampilan dari form metode yang sudah dipilih sebelumnya. Lalu komputer melakukan pemilihan *button click input file* untuk memasukkan *file* data yang akan diproses. Setelah itu server akan merespon dan komputer akan memilih *file* yang akan diproses dan setelah dipilih langsung komputer meng-*upload file* yang sudah dipilih. Maka server akan merespon dan menampilkan *form* yang mana sudah terdapat tampilan *file* sudah terupload.



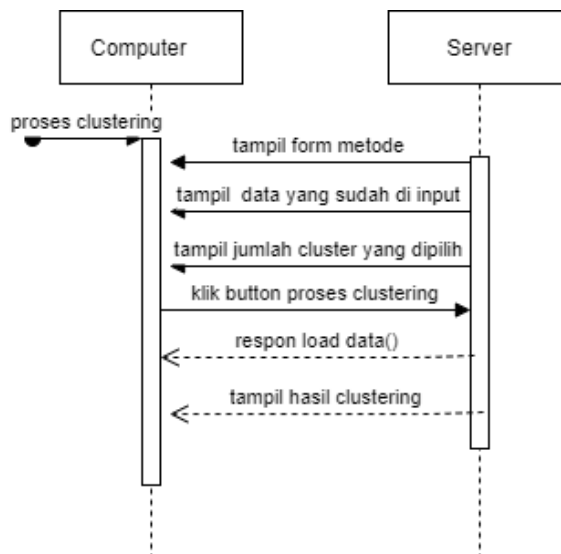
### 1.3.4.3 Sequence Diagram Pilih Cluster



Gambar 4.7 Sequence Diagram Pilih Cluster

Pada Gambar 4.7 dijelaskan bahwa langkah dalam memilih jumlah *cluster*, berawal dari server menampilkan *form* yang sudah dipilih metode nya, lalu *form* yang sudah terdapat tampilan *file* data yang telah dilampirkan yang lalu, komputer melakukan pemilihan pada tombol *dropdown* dan server merespon lalu komputer memilih jumlah *cluster* yang diperlukan. Setelah itu server merespon apa yang sudah dipilih oleh komputer.

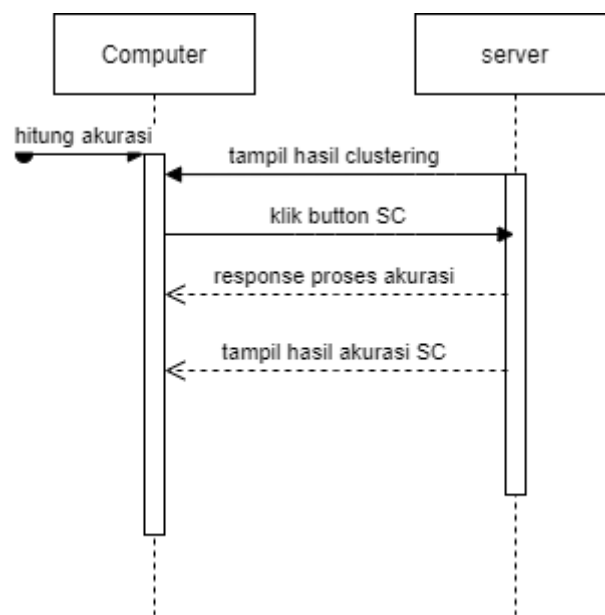
### 1.3.4.4 Sequence Diagram Proses Clustering



Gambar 4.8 Sequence Diagram Proses Clustering

Pada Gambar 4.8 dijelaskan bahwa langkah dalam proses *clustering*, berawal dari server menampilkan *form* yang sudah dipilih metode nya, lalu *form* yang sudah terdapat tampilan *file* data yang telah dilampirkan dan juga jumlah *cluster* yang sudah ditentukan. Lalu, komputer melakukan pemilihan pada tombol proses dan server merespon dan melakukan proses *clustering* data. Setelah proses selesai, server menampilkan hasil *clustering*.

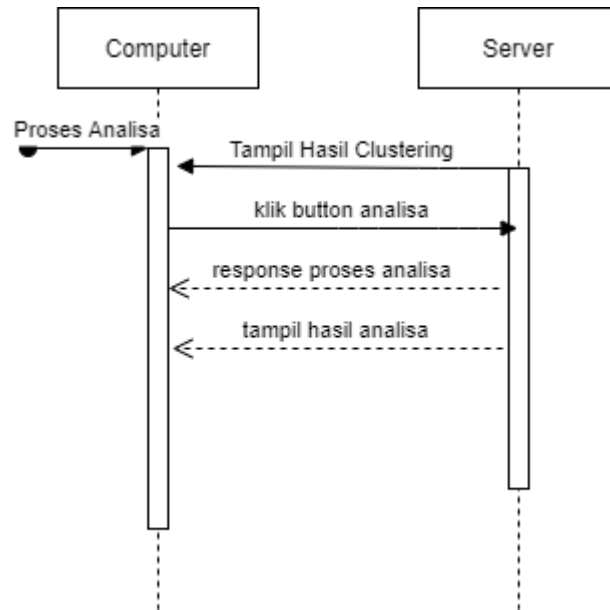
#### 1.3.4.5 Sequence Diagram Akurasi



Gambar 4.9 Sequence Diagram Akurasi

Pada Gambar 4.9 dijelaskan bahwa langkah dalam proses perhitungan akurasi dengan menggunakan *silhouette coefficient* (SC). Berawal dari server menampilkan hasil *clustering* yang lalu terdapat tombol SC pada form hasil *clustering* tersebut. Lalu, setelah di klik tombol SC, maka server akan merespon proses akurasi tersebut kemudian server akan menampilkan hasil dari akurasi SC pada hasil *clustering*.

#### 1.3.4.6 Sequence Diagram Analisa

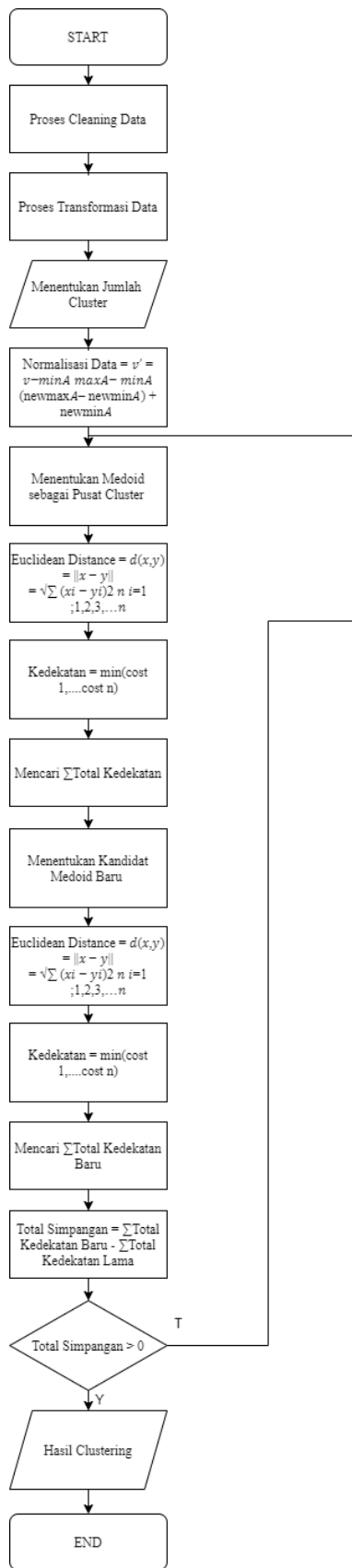


Gambar 4.10 Sequence Diagram Analisa

Pada Gambar 4.10 dijelaskan bahwa langkah dalam proses Analisa. Berawal dari server menampilkan hasil *clustering* yang lalu terdapat tombol analisa pada form hasil *clustering* tersebut. Lalu, setelah di klik tombol analisa, maka server akan merespon proses analisa hasil *clustering* tersebut kemudian server akan menampilkan hasil dari analisa pada hasil *clustering*.

### 4.3 Perancangan Perhitungan

Pada implementasi software ini menggunakan metode K-Medoid *Clustering* dan juga metode K-Means *Clustering*. Berikut ini merupakan contoh perhitungan dengan menggunakan metode K-Medoid *Clustering* digambarkan dengan *flowchart* pada Gambar 4.11.



Gambar 4.11 Flowchart K-Medoid

Untuk contoh perhitungan dicontohkan dengan 12 data beserta 6 buah pertanyaan dari aspek suami. Berikut contoh sample data responden pada Tabel 4.4. Sebelumnya, data ini sudah dilakukan proses *cleaning* data dan juga transformasi data. Sebelum dilakukan proses *cleaning data* dan transformasi data masih merupakan data yang belum relevan atau data tersebut masih belum valid. Ada data yang kosong. Akan tetapi, setelah melalui sistem yang tersedia pada kuesioner *online*. Data kosong tersebut diminta agar tidak kosong atau harus diisi semuanya, sehingga data tersebut lengkap dan siap untuk dilakukan proses klusterisasi. Berikut contoh data sebelum dilakukan *cleaning* data dan transformasi data. Berikut untuk

1. Sebelum mendapatkan data, pada sistem pengisian kuesioner, dilakukan pengecekan yaitu *cleaning* data dan transformasi data guna untuk menanggulangi kelemahan *K-Means* yaitu *noise* sehingga diusahakan data tidak

Nama	Usia	Pendidikan Terakhir	Pekerjaan	Asal Suku	Status Pernikahan	Jumlah Anak	Usia anak tertua	Usia anak terkecil	Tempat Lahir	Nomor HP	Alamat	Masa anak-anak	Masa remaja	Masa dewasa-menikah	Usia Suami	Pendidikan Terakhir Suami	Pekerjaan Suami	Asal Suku Suami	Masa anak-anak Suami	Masa remaja Suami
Titin	43	3	8	1,1	1	3	23	8	Blitar	0	0	Blitar	Blitar	Blitar	46	4	8	1,1	Blitar	Blitar
Laili	43	3	8	1,1	1	2	14	13	Blitar	0	0	Blitar	Blitar	Blitar	46	4	2	1,1	Blitar	Blitar
Emi	41	4	8	1,1	1	3	24	19	Blitar	0	0	Blitar	Blitar	Blitar	44	4	1	1,1	Blitar	Blitar
Rida	46	4	2	1,1	1	1	20	0	Sidoarjo	0	0	Sidoarjo	Sidoarjo	Sidoarjo	46	3	5	1,1	Malang	Malang
Chudaya	45	4	8	1,1	1	2	23	18	Sidoarjo	0	0	Sidoarjo	Sidoarjo	Sidoarjo	49	4	5	1,1	Sidoarjo	Sidoarjo
Isbandiya	51	4	8	1,1	1	2	22	18	Sidoarjo	0	0	Sidoarjo	Sidoarjo	Sidoarjo	53	3	2	1,1	Sidoarjo	Sidoarjo
Dwi	46	4	8	1,1	1	2	23	12	Surabaya	0	0	Driyorejo	Driyorejo	Sidoarjo	49	4	5	1,1	Wonokoro	Wonokoro
Sumiati	43	3	7	1,1	1	1	21	0	Madun	0	0	Pasuruan			3	8	1,1	Nganjuk	Surabaya	
Ratna	49	7	5	1,1	1	3	17	13	Trenggalek	0	0	Trenggalek	Trenggalek	Sidoarjo	53	6	1	1,1	Surabaya	Surabaya
Siti	47	4	8	1,1	1	2	22	18	Jombang	0	0	Jombang	Surabaya	Sidoarjo	51	7	2	1,1	Surabaya	Surabaya
Maryunik	44	7	3	1,1	3	2	25	17	Malang	0	0	Blitar	Malang	Malang						
Umi Masli	51	7	2	1,1	1	2	27	15	Blitar	0	0	Blitar	Surabaya	Blitar	57	2	2	1,1	Blitar	Blitar

Gambar 4.12 Sebelum dilakukan *Cleaning* Data dan Transformasi Data

Nama	Usia	Pendidikan Terakhir	Pekerjaan	Asal Suku	Status Pernikahan	Jumlah Anak	Usia anak tertua	Usia anak terkecil	Tempat Lahir	Nomor HP	Alamat	Masa anak-anak	Masa remaja	Masa dewasa-menikah	Usia Suami	Pendidikan Terakhir Suami	Pekerjaan Suami	Asal Suku Suami	Masa anak-anak Suami	Masa remaja Suami
Titin	43	3	8	1,1	1	3	23	8	Blitar	0	0	Blitar	Blitar	Blitar	46	4	8	1,1	Blitar	Blitar
Laili	43	3	8	1,1	1	2	14	13	Blitar	0	0	Blitar	Blitar	Blitar	46	4	2	1,1	Blitar	Blitar
Emi	41	4	8	1,1	1	3	24	19	Blitar	0	0	Blitar	Blitar	Blitar	44	4	1	1,1	Blitar	Blitar
Rida	46	4	2	1,1	1	1	20	0	Sidoarjo	0	0	Sidoarjo	Sidoarjo	Sidoarjo	46	3	5	1,1	Malang	Malang
Chudaya	45	4	8	1,1	1	2	23	18	Sidoarjo	0	0	Sidoarjo	Sidoarjo	Sidoarjo	49	4	5	1,1	Sidoarjo	Sidoarjo
Isbandiya	51	4	8	1,1	1	2	22	18	Sidoarjo	0	0	Sidoarjo	Sidoarjo	Sidoarjo	53	3	2	1,1	Sidoarjo	Sidoarjo
Dwi	46	4	8	1,1	1	2	23	12	Surabaya	0	0	Driyorejo	Driyorejo	Sidoarjo	49	4	5	1,1	Wonokoro	Wonokoro
Ratna	49	7	5	1,1	1	3	17	13	Trenggalek	0	0	Trenggalek	Trenggalek	Sidoarjo	53	6	1	1,1	Surabaya	Surabaya
Siti	47	4	8	1,1	1	2	22	18	Jombang	0	0	Jombang	Surabaya	Sidoarjo	51	7	2	1,1	Surabaya	Surabaya
Umi Masli	51	7	2	1,1	1	2	27	15	Blitar	0	0	Blitar	Surabaya	Blitar	57	2	2	1,1	Blitar	Blitar
Rofah	35	7	3	1,1	1	2	17	9	Blitar	0	0	Blitar	Blitar	Blitar	40	4	8	1,1	Blitar	Blitar
Juanti	40	3	8	1,1	1	2	22	12	Blitar	0	0	Blitar	Blitar	Blitar	42	2	2	1,1	Blitar	Blitar

Gambar 4.13 Sesudah dilakukan *Cleaning* Data dan Transformasi Data

## 2. Data hasil data responden

Tabel 4.4 Sampel Data Responden

SAMPLE DATA RESPONDEN						
Nama	1_p1	1_p2	1_p3	1_p4	1_p5	1_p6
siska	3	2	3	2	4	2
linda	3	2	3	3	3	2
Mia maulida	3	4	2	4	2	4
Larasaty	2	4	2	3	2	1
Sindy Oktiana	4	1	3	1	2	1
Yayik Elise	4	3	2	4	5	1
Grinas Prabaniska	5	4	1	2	3	1
Anjani Dwi	4	2	1	3	4	1
Gayatri	2	2	1	2	3	1
Risma	4	3	4	2	3	2
Vina Perwitasari	4	3	1	2	4	1
Habibatul	3	2	3	1	3	1

Contoh data pada Tabel 4.4 merupakan contoh data responden yang nantinya digunakan sebagai bahan untuk data uji pada penelitian ini. Contoh data pada Tabel 4.4 hanya sebagian pertanyaan pada aspek suami, kolom 1\_p1 itu berarti pertanyaan pertama pada aspek 1 yaitu aspek suami. Nantinya, data sebenarnya menggunakan 48 pertanyaan yaitu aspek suami 8 pertanyaan, aspek anak 10 pertanyaan, aspek orang tua 10 pertanyaan, aspek saudara 10 pertanyaan dan aspek pekerjaan 10 pertanyaan. Angka 1, 2, 3, 4 dan 5 merupakan konversi data menjadi data angka. Untuk angka 1 berarti jawaban responden tidak marah, angka 2 berarti sedikit marah, angka 3 berarti agak marah, angka 4 berarti marah dan angka 5 berarti sangat marah.

3. Proses selanjutnya yaitu dilakukan proses normalisasi. Untuk menghitung proses normalisasi pada Tabel 4.5 dan 4.6, penulis menggunakan bentuk perhitungan manual sebagai berikut :

$$\text{mindata} = \min(\text{keseluruhan data}) \quad (1)$$

$$\text{maxdata} = \max(\text{keseluruhan data}) \quad (2)$$

Contoh untuk mencari nilai Minimum dan Maximum adalah sebagai berikut :

Nilai minimum dari data (3,3,3,2,4,4,5,4,2,4,4,3) adalah 2, jadi nilai minimum pada kolom pertama (p1\_1) adalah 2.

Nilai maksimum dari data (3,3,3,2,4,4,5,4,2,4,4,3) adalah 5, jadi nilai minimum pada kolom pertama (p1\_1) adalah 5.

Tabel 4.5 Normalisasi (1)

NILAI MINIMUM DAN MAXIMUM DATA						
MIN	2	1	1	1	2	1
MAX	5	4	4	4	5	4

Tabel 4.5 merupakan salah satu langkah untuk metode *K-Medoids* setelah dilakukan proses *cleaning* data dan juga transformasi data.

Setelah dilakukan pencarian *Min-Max* data. Selanjutnya dilakukanlah proses normalisasi sesuai dengan alur perhitungan *K-Medoids*. Hasil normalisasi contohnya dapat dilihat pada Tabel 4.6. Cara mendapatkan nilai normalisasi pada kolom pertama yaitu : data pertama pada kolom sample data responden 1\_p1 = 3, lalu nilai minimum pada kolom 1\_p1 = 2, kemudian nilai maksimum pada kolom 1\_p1 = 5. Jadi nilai normalisasi pada kolom pertama yaitu :

$$normalisasi = \frac{(data\ awal - mindata)}{(maxdata - mindata)} \quad (3)$$

$$normalisasi = \frac{(3-2)}{(5-2)} = \frac{1}{3} = 0,333 \text{ (tulisan berwarna biru). Begitu seterusnya}$$

hingga seluruh data sudah dilakukan normalisasi.

Tabel 4.6 Normalisasi (2)

NORMALISASI						
Nama	1_p1	1_p2	1_p3	1_p4	1_p5	1_p6
siska	0,33333 3	0,33333 3	0,66666 7	0,33333 3	0,66666 7	0,33333 3
linda	0,33333 3	0,33333 3	0,66666 7	0,66666 7	0,33333 3	0,33333 3
Mia maulida	0,33333 3		0,33333 3		0	1
Larasaty	0	1	0,33333 3	0,66666 7	0	0
Sindy Oktiana	0,66666 7		0,66666 7		0	0
Yayik Elise	0,66666 7	0,66666 7	0,33333 3		1	0
Grinas Prabaniska	1	1	0	0,33333 3	0,33333 3	0
Anjani Dwi	0,66666 7	0,33333 3		0,66666 7	0,66666 7	0

Gayatri	0	0,33333 3	0	0,33333 3	0,33333 3	0
Risma	0,66666 7	0,66666 7	1	0,33333 3	0,33333 3	0,33333 3
Vina Perwitasari	0,66666 7	0,66666 7	0	0,33333 3	0,66666 7	0
Habibatul	0,33333 3	0,33333 3	0,66666 7	0	0,33333 3	0

4. Langkah selanjutnya setelah normalisasi adalah mengambil pusat *cluster* secara *random* sesuai dengan hasil normalisasi. Disini, penulis mencontohkan mengambil pusat *cluster* sebanyak 4 dengan objek data 1, 2, 3 dan 4. Berikut contoh data pusat *cluster* yang telah dipilih, terlihat pada Tabel 4.7.

Tabel 4.7 Pusat Cluster

PUSAT CLUSTER						
Nama	1_p1	1_p2	1_p3	1_p4	1_p5	1_p6
siska	0,33333 3	0,33333 3	0,66666 7	0,33333 3	0,66666 7	0,33333 3
linda	0,33333 3	0,33333 3	0,66666 7	0,66666 7	0,33333 3	0,33333 3
Mia maulida	0,33333 3	1	0,33333 3	1	0	1
Larasaty	0	1	0,33333 3	0,66666 7	0	0

Tabel 4.7 merupakan hasil dari pengambilan pusat *cluster* atau medoid secara acak berdasarkan data yang telah disuguhkan.

5. Setelah dilakukan pengambilan pusat *cluster* secara acak, langkah selanjutnya yaitu menghitung jarak dengan menggunakan rumus *Euclidean Distance*. Berikut cara perhitungannya.

$$d(x, y) = \sqrt{\sum (x_i - y_i)^2} \quad i = 1; 1, 2, 3, \dots, n \quad (4)$$

Keterangan :

d = *distance* (jarak)

x = data normalisasi

y = data pusat *cluster*

i = banyaknya cost



Hasil dari data *cost* merupakan hasil yang akan digunakan untuk mencari nilai kedekatan yang nantinya akan membentuk *cluster-cluster*. Berikut contoh untuk mendapatkan nilai *cost* :

$$\begin{aligned} \text{jarak} &= \sqrt{(0 - 0,333)^2 + (0,333 - 0,333)^2 + (0,666667 - 0,666667)^2} \\ &\quad + (0,3333 - 0,333)^2 + (0,666667 - 0,666667)^2 \\ &\quad + (0,333 - 0,333)^2 = 0 \end{aligned}$$

Hal tersebut dilakukan dengan cara yang sama hingga banyaknya *cost*. Banyaknya *cost* tergantung dengan jumlah *cluster* yang dipilih.

Berikut contoh tabel hasil data yang terlihat pada Tabel 4.8 setelah dilakukan perhitungan manual.

Tabel 4.8 Hasil Data *Cost*

DATA JARAK			
COST 1	COST 2	COST 3	COST 4
0	3,789606	3,556098	3,099059
3,789606	0	3,602661	3,606514
3,556098	3,602661	0	2,948634
3,099059	3,606514	2,948634	0
3,210227	4,171664	3,868498	3,110243
3,478106	3,765708	4,185789	3,667614
3,198307	3,764786	3,5	3,088959
3,714648	4,007805	3,970376	3,91578
3,243583	4,004338	3,488075	3,104656
3,085585	3,731436	3,333333	3,22318
3,251068	3,813354	3,430865	3,371655
3,36547	3,951969	3,263859	3,004626

6. Langkah berikutnya, menghitung kedekatan masing-masing data. Cara menghitung kedekatan masing-masing data dengan menggunakan cara berikut :

$$\text{kedekatan} = \min(\text{cost1}, \dots, \text{costn}) \quad (5)$$

Contoh :

$$\text{kedekatan} = \min(0, 3,789606, 3,556098, 3,099059) = 0$$

Begitu untuk seterusnya hingga 12 data terpenuhi.

Berikut contoh tabel hasil kedekatan setelah dihitung terlihat pada Tabel 4.9

Tabel 4.9 Hasil Kedekatan

HASIL KEDEKATAN				
COST 1	COST 2	COST 3	COST 4	KEDEKATAN
0	3,789606	3,556098	3,099059	0
3,789606	0	3,602661	3,606514	0
3,556098	3,602661	0	2,948634	0
3,099059	3,606514	2,948634	0	0
3,210227	4,171664	3,868498	3,110243	3,11024293
3,478106	3,765708	4,185789	3,667614	3,47810613
3,198307	3,764786	3,5	3,088959	3,08895883
3,714648	4,007805	3,970376	3,91578	3,71464818
3,243583	4,004338	3,488075	3,104656	3,104656
3,085585	3,731436	3,333333	3,22318	3,08558476
3,251068	3,813354	3,430865	3,371655	3,2510682
3,36547	3,951969	3,263859	3,004626	3,00462606

Setelah mendapatkan nilai kedekatan sesuai dengan yang ada pada Tabel 4.9, maka selanjutnya mengelompokkan data-data nya sesuai dengan letak *cost* nya, Misal kedekatan pada data pertama masuk pada *cost* 1 maka data pertama merupakan salah satu anggota dari *cluster* 1, begitupun selanjutnya juga begitu.

7. Langkah berikutnya, menghitung total kedekatan awal, berikut cara menghitung kedekatan awal dengan menggunakan perhitungan manual:

$$\sum \text{total kedekatan} \quad (6)$$

Contoh :

*Total Kedekatan*

$$\begin{aligned}
 &= (0 + 0 + 0 + 0 + 3,11024293 + 3,47810613 \\
 &+ 3,08895883 + 3,71464818 + 3,104656 + 3,08558476 \\
 &+ 3,2510682 + 3,00462606 = \mathbf{21,60212902}
 \end{aligned}$$

Berikut contoh tabel hasil jumlah total kedekatan. Terlihat pada tabel 4.10

Tabel 4.10 Hasil Total Kedekatan

<b>JUMLAH KEDEKATAN</b>	<b>21,60212902</b>
-------------------------	--------------------

8. Setelah mendapatkan hasil jumlah kedekatan awal, selanjutnya menentukan kandidat medoid baru dengan cara mengambil secara *random* yang berbeda dengan pusat *cluster* awal. Disini, penulis memberikan contoh pengambilan medoid baru pada objek 5, 6, 7 dan 8. Berikut terlihat pada Tabel 4.11

Tabel 4.11 Pusat Cluster Baru

PUSAT CLUSTER BARU						
Nama	1_p1	1_p2	1_p3	1_p4	1_p5	1_p6
Sindy Oktiana	0,666667	0	0,666667	0	0	0
Yayik Elise	0,666667	0,666667	0,333333	1	1	0
Grinas Prabaniska	1	1	0	0,333333	0,333333	0
Anjani Dwi	0,666667	0,333333	0	0,666667	0,666667	0

Pusat *cluster* baru digunakan sebagai titik acuan untuk bisa mendapatkan hasil iterasi yang nantinya hasil iterasi tersebut dapat diketahui total kedekatannya dan lalu dicari selisihnya. Nantinya selisih nanti sebagai penentu apakah perhitungan terus dilanjutkan atau berhenti. Penentuan *cluster* baru ini merupakan salah satu langkah *K-Medoid* yang dapat menutupi kekurangan *K-Means* yaitu *outlier*.

9. Langkah selanjutnya mengulang langkah 4 hingga langkah 6.
10. Setelah mendapatkan total jumlah kedekatan baru, dilakukan proses perhitungan untuk mendapatkan nilai total simpangan. Caranya, dengan menggunakan perhitungan manual yaitu sebagai berikut :

$$S = \sum \text{total kedekatan baru} - \sum \text{total kedekatan lama} \quad (7)$$

Contoh :

$$S = 22,78669 - 25,83789 = -3,0512$$

Berikut contoh tabel hasil total simpangan setelah dilakukan proses perhitungan dengan menggunakan rumus perhitungan manual. Terlihat pada Tabel 4.12

Tabel 4.12 Hasil Total Simpangan

<b>JUMLAH KEDEKATAN BARU</b>	22,78669
<b>JUMLAH KEDEKATAN LAMA</b>	25,83789
<b>SELISIH KEDEKATAN</b>	-3,0512

Menurut hasil yang didapatkan, nilai selisih kedekatan  $\leq 0$ , maka hendak dilakukan perhitungan ulang. Langkah perhitungan ulang dimulai dari langkah 3 hingga langkah 9. Jika hasil selisih kedekatan dikatakan  $\geq 0$ , maka perhitungan berhenti dan mengambil perhitungan awal sebagai hasil *clustering*. Berikut contoh selisih kedekatan jika  $\geq 0$  terlihat pada Tabel 4.13

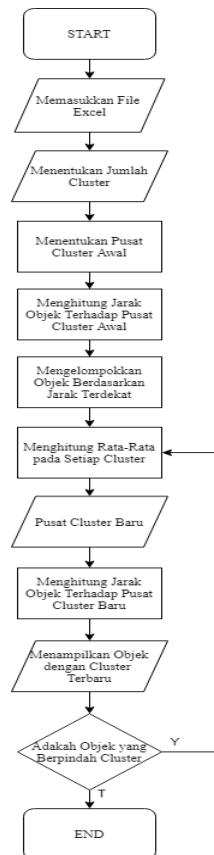
Contoh :

$$S = 22,21657442 - 21,60212902 = 0,614445407$$

Tabel 4.13 Hasil Selisih Kedekatan

<b>JUMLAH KEDEKATAN BARU</b>	22,21657442
<b>JUMLAH KEDEKATAN LAMA</b>	21,60212902
<b>SELISIH KEDEKATAN</b>	0,614445407

Untuk perencanaan perhitungan K-Means *Clustering* dapat dilihat melalui Gambar 4.7.



Gambar 4.14 Flowchart K-Means

Berdasarkan *flowchart* pada Gambar 4.9 menjelaskan bahwa :

1. Input yang digunakan dalam metode ini merupakan data responden yang menjawab pertanyaan yang tersedia dengan rentang nilai antara 1 hingga 5
2. Menentukan jumlah  $k$  atau jumlah *cluster*
3. Menentukan pusat *cluster* awal sesuai dengan jumlah *cluster*
4. Hitung jarak objek ke pusat *cluster* awal atau *centroid* awal menggunakan rumus *euclidean distance* sama seperti K-Medoid sehingga membentuk tabel seperti Tabel 4.1

Tabel 4.14 Hasil Data Jarak

DATA JARAK		
C1	C2	C3
0	10,81665	9,110434
10,81665	0	10,19804
9,219544	10,29563	0
8,42615	10,19804	7,745967
8,124038	12,04159	11,09054
9,539392	11,6619	12
8,42615	11,04536	9,797959
10,29563	11,7047	12,12436
8,888194	11,40175	10,29563
7,81025	10,77033	9,69536
8,485281	11,26943	10,34408
8,774964	11,74734	10

5. Menghitung jarak minimum sesuai *cluster* sehingga membentuk seperti pada Tabel 4.15

Tabel 4.15 Hasil Kedekatan

HASIL KEDEKATAN			
C1	C2	C3	Kedekatan
0	10,81665	9,110434	0
10,81665	0	10,19804	0
9,219544	10,29563	0	0
8,42615	10,19804	7,745967	7,74596669
8,124038	12,04159	11,09054	8,1240384

9,539392	11,6619	12	9,53939201
8,42615	11,04536	9,797959	8,42614977
10,29563	11,7047	12,12436	10,2956301
8,888194	11,40175	10,29563	8,88819442
7,81025	10,77033	9,69536	7,81024968
8,485281	11,26943	10,34408	8,48528137
8,774964	11,74734	10	8,77496439

6. Mengelompokkan *cluster* berdasarkan nilai kedekatan, sehingga dapat terbentuk seperti pada Tabel 4.16

Tabel 4.16 Hasil Clustering K-Means

HASIL CLUSTERING				
C1	C2	C3	Kedekatan	Hasil
0	10,81665	9,110434	0	1
10,81665	0	10,19804	0	2
9,219544	10,29563	0	0	3
8,42615	10,19804	7,745967	7,74596669	3
8,124038	12,04159	11,09054	8,1240384	1
9,539392	11,6619	12	9,53939201	1
8,42615	11,04536	9,797959	8,42614977	1
10,29563	11,7047	12,12436	10,2956301	1
8,888194	11,40175	10,29563	8,88819442	1
7,81025	10,77033	9,69536	7,81024968	1
8,485281	11,26943	10,34408	8,48528137	1
8,774964	11,74734	10	8,77496439	1

7. Hitung rata-rata setiap *cluster* untuk mendapatkan pusat *cluster* baru. Terlihat pada Tabel 4.17

Tabel 4.17 Pusat Cluster Baru

PUSAT CLUSTER BARU	
c1	3,873838
c2	0
c3	3,872983

8. Lalu, dilakukan kembali langkah 4 hingga langkah 7, jika pada hasil objek tidak berubah tempat, maka perhitungan selesai dan akan menampilkan hasil proses *clustering*.

Perhitungan akurasi pada penelitian kali ini juga dibutuhkan, karena adanya perbandingan antara metode K-Medoid *clustering* dengan metode K-Means *clustering*. Untuk perhitungan akurasi itu sendiri, disini peneliti menggunakan *silhouette coefficient*. Rumus untuk *silhouette coefficient* itu sendiri yaitu dengan persamaan berikut :

$$a(i) = \frac{1}{[A]-1} \sum_{j \in A, j \neq i} d(i, j) \quad (1)$$

$$d(i, C) = \frac{1}{[A]} \sum_{j \in C} d(i, j) \quad (2)$$

$$s(i) = \frac{b(i)-a(i)}{\max(a(i), b(i))} \quad (3)$$

#### 4.4 Desain Wireframe

Diperlukan desain *wireframe* atau *mockup* untuk memudahkan penyusunan antarmuka yang disediakan bagi pengguna aplikasi. Berikut *mockup* sistem aplikasi *clustering* Respon Emosi Marah Wanita Jawa.

##### 4.5.1 Rancangan Tampilan Kuesioner untuk Responden

No	Nama	Usia	Pendidikan	Pekerjaan	Action
1	Anggia	25	S1/D4	PNS	Exp Del
2	Wahyu	25	D3	Lainnya	Exp Del
3			SMA	Lainnya	Exp Del
4	Isna	22	SMA	PNS	Exp Del
5	Rudhi	36	SD/MI	Lainnya	Exp Del
6	Parjiyem	48	D3	Lainnya	Exp Del
7	Icha	24	S1/D4	Lainnya	Exp Del
8	Fitria	33	S1/D4	PNS	Exp Del

Gambar 4.15 Tampilan Responden (1)

A Web Page  
<http://www.kuisieronresponemosi.epizy.com/Kuisieron>

Kuisieron  
 Kontak

**Data Responden**

Nama

Usia

Pendidikan Terakhir

**Pertanyaan Responden**

	Tidak Marah	Sedikit Marah	Agak Marah	Marah	Sangat Marah
1 Saat suami saya mengobrol dengan seorang gadis, dan gadis itu terlihat bukan perempuan baik-baik.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
2 Saat kedua anak laki-laki (remaja) saya berkelahi karena masalah perempuan yang sama-sama mereka suka.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Gambar 4.16 Tampilan Responden (2)

Pada Gambar 4.15 dan Gambar 4.16 merupakan Gambar tampilan kuesioner yang dibagikan kepada responden yang mana nantinya kuesioner ini akan diisi oleh responden dan tidak boleh ada yang terlewatkan. Datanya ini nanti akan dijadikan bahan untuk penelitian kali ini.

#### 4.5.2 Rancangan Sistem Aplikasi *Clustering* K-Medoid

A Web Page  
[http://localhost/skripsi\\_wahid/](http://localhost/skripsi_wahid/)

K-medoid K-means

K-medoid

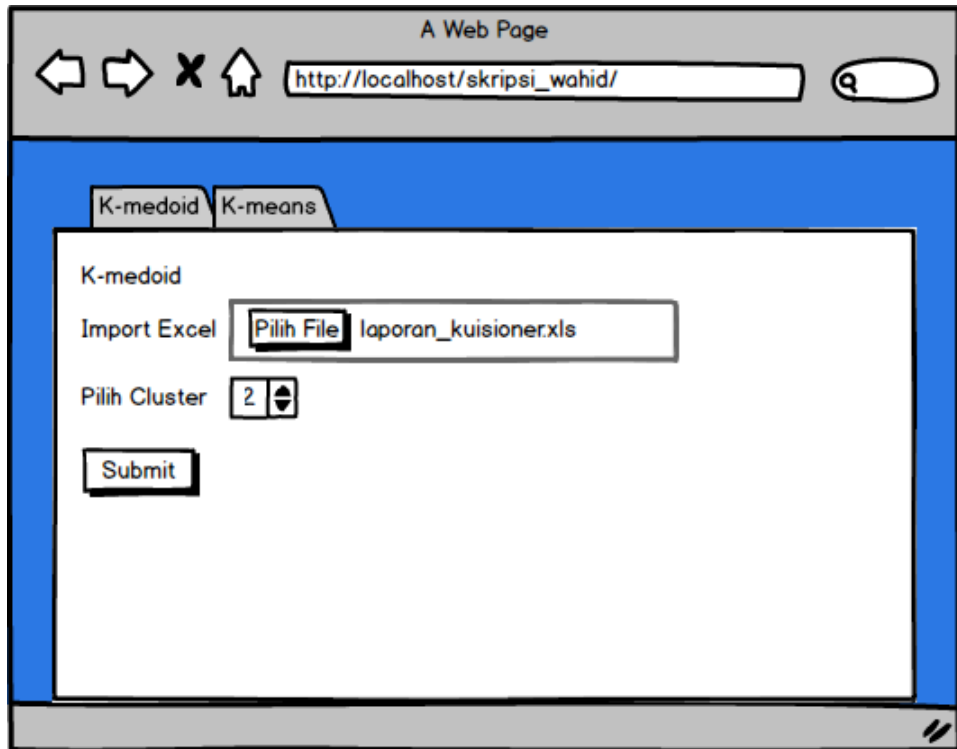
Import Excel

Pilih Cluster

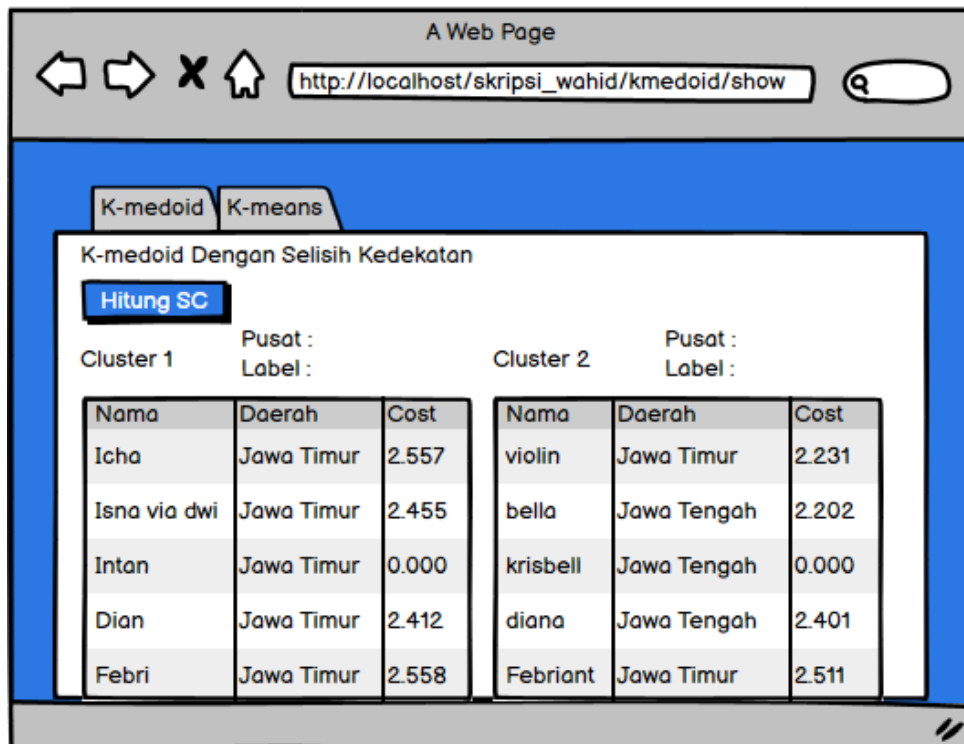
Gambar 4.17 Tampilan Awal Sistem Clustering K-Medoid



Gambar 4.17 merupakan Gambar tampilan awal yang berada pada sistem. Pada Gambar tersebut terdapat kotak dan tombol untuk dapat menginputkan *file data excel* yang merupakan *file data kuesioner* yang telah diisi pada situs kuesioner. Selain itu juga terdapat tombol fitur *dropdown* guna untuk memilih jumlah *cluster* yang akan dibentuk. Lalu juga terdapat tombol *submit* yang mana fungsinya untuk klik proses *clustering* nya.



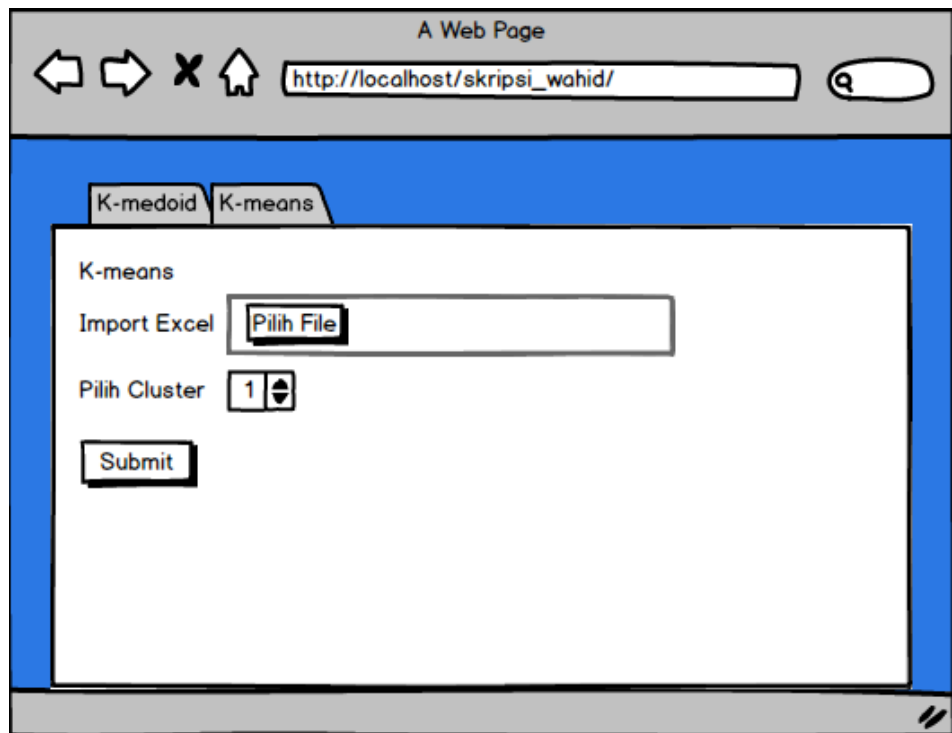
Gambar 4.18 Tampilan Awal Sistem Setelah Input File



Gambar 4.19 Tampilan Hasil Clustering K-Medoid

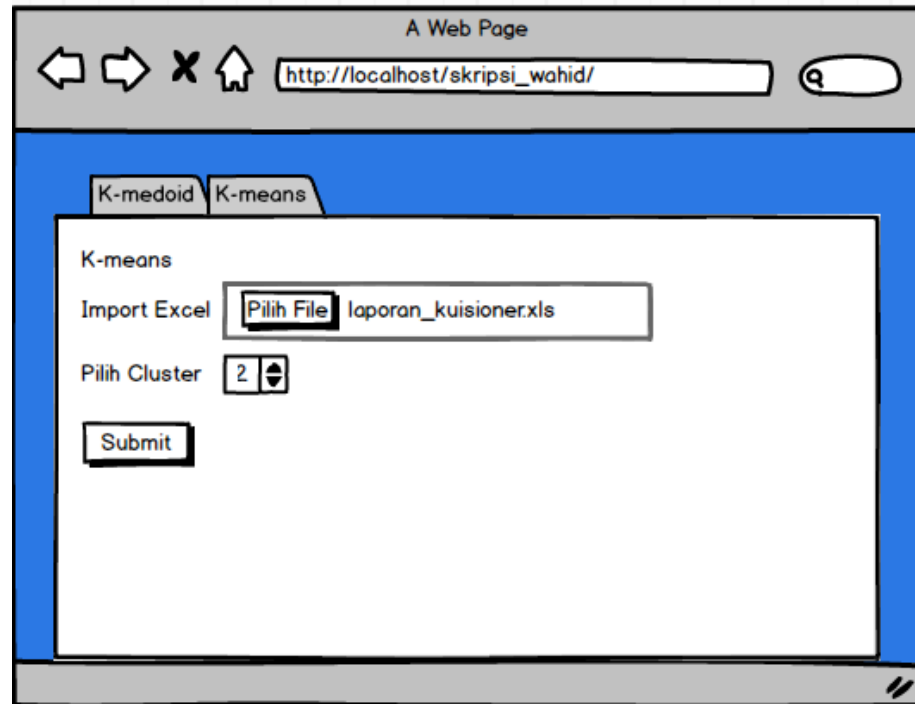
Setelah dilakukan input file data excel lalu dipilih jumlah cluster yang diinginkan maka ketika di klik proses maka akan menampilkan beberapa cluster sesuai yang dipilih dan menampilkan pusat, label dan juga anggota-anggota clusternya. Dapat dilihat pada Gambar 4.18 dan 4.19 dan juga terdapat tombol SC yang gunanya untuk menghitung hasil akurasi pada sistem *clustering K-Medoids*.

#### 4.5.3 Rancangan Sistem Aplikasi *Clustering* K-Means

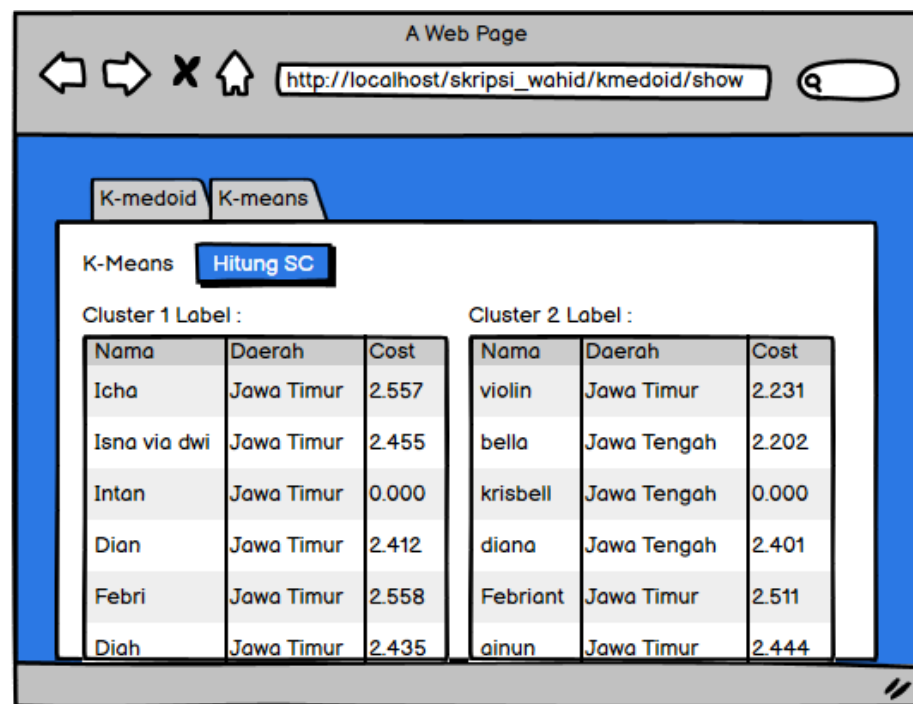


Gambar 4.20 Tampilan Awal Sistem Clustering K-Means

Pada Gambar 4.20 merupakan gambar tampilan awal sistem *clustering* untuk bagian *K-Means*. Secara umum, tampilan awal *clustering* memang sama persis dengan bagian awal sistem *K-Medoids*. Terdapat kotak dan juga tombol pilih *file* untuk melampirkan *file* data *excel*. Terdapat 1 buah kotak dan tombol *dropdown* untuk dapat memilih berapa jumlah *cluster* yang akan dipilih. Terdapat juga tombol *submit* untuk memproses data *clustering*.



Gambar 4.21 Tampilan Setelah Input File

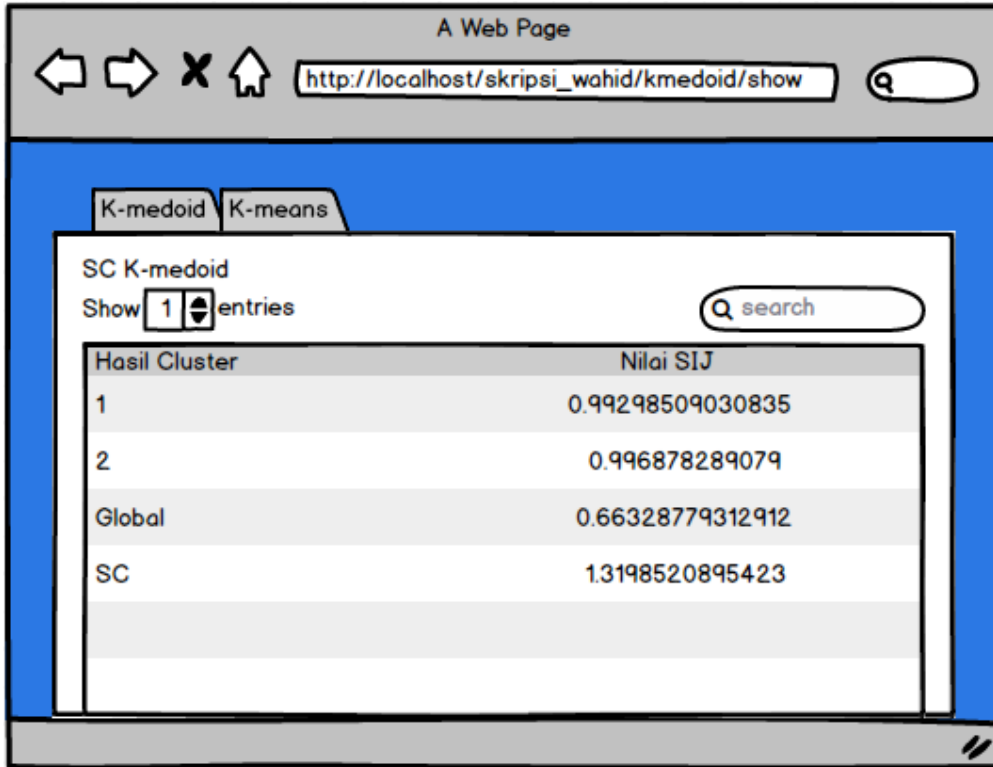


Gambar 4.22 Tampilan Hasil Clustering K-Means

Pada Gambar 4.20 merupakan tampilan dimana setelah melakukan *input file* data excel yang telah dimasukkan lalu dilakukan proses peng-*clusteran* maka akan tampil seperti pada Gambar 4.21. Pada Gambar 4.22 juga terdapat

tombol SC yang gunanya untuk menghitung hasil akurasi pada sistem *clustering K-Means*.

#### 4.5.4 Rancangan Sistem Perhitungan Akurasi *Silhouette Coefficient*



Hasil Cluster	Nilai SIJ
1	0.99298509030835
2	0.996878289079
Global	0.66328779312912
SC	1.3198520895423

Gambar 4.23 Tampilan Silhouette Coefficient

Pada Gambar 4.23 merupakan tampilan akurasi nantinya yang akan digunakan pada tiap-tiap metode *clustering*. Disana akan ditampilkan berapa nilai hasil akurasi *clustering* nya.

