

## **BAB IV. ANALISIS DAN PERANCANGAN SISTEM**

### **4.1. Deskripsi Sistem**

Sistem *Clustering* Jawaban Uraian Mahasiswa Menggunakan *TF-IDF* dan *Kmeans* ini merupakan sistem berbasis *website* yang berguna untuk memudahkan proses penilaian jawaban uraian mahasiswa dengan cara mengelompokkan jawaban-jawaban dalam bentuk klaster. Sistem ini berfungsi seperti sistem tanya jawab soal pada umumnya dimana akan ada kategori soal, soal dan jawaban yang perlu dimasukkan sebagai *input*.

Data *input* itu sendiri berasal dari data kuis atau tugas dari dosen yang telah peneliti kumpulkan. Pada sistem ini mahasiswa akan menjawab soal uraian yang diberikan oleh dosen dan jawaban dari setiap soal tersebut yang akan di lakukan *clustering* berdasarkan soalnya yang akan memunculkan *output* data jawaban mahasiswa dalam bentuk klaster, sehingga diharapkan dengan adanya jawaban yang sudah di klaster dapat mempermudah dosen dalam menentukan nilai dari setiap mahasiswa.

### **4.2. Analisis Permasalahan**

Dari latar belakang yang sudah dipaparkan diatas maka peneliti membuat analisis permasalahan akan dijadikan sebagai bahan penelitian sebagai berikut:

1. Kurang efektifnya metode yang dilakukan dosen dalam menilai jawaban soal uraian
2. Karena dituntut harus tetap netral dan transparan proses penilaian yang dilakukan dosen dapat menjadi sangat melelahkan dan memakan banyak waktu
3. Semakin banyak mahasiswa membuat semakin banyaknya proses penilaian yang harus dilakukan dosen
4. Proses tanya jawab soal sudah banyak yang dilakukan menggunakan bantuan sistem namun untuk proses penilaian masih dilakukan dengan manual yaitu membaca satu persatu jawaban.

Penelitian ini bertujuan untuk mengembangkan metode-metode yang diharapkan dapat efektif membantu dosen dalam melakukan penilaian jawaban

uraian mahasiswa dengan cara mengelompokkan jawaban menggunakan *TF-IDF* dan *Kmeans*

### 4.3. Analisis Kebutuhan Sistem

Pada bagian ini akan dibahas hasil dari proses identifikasi dari kebutuhan apa saja yang diperlukan dalam membangun Sistem *Clustering* Jawaban Uraian Mahasiswa dengan Menggunakan *TF-IDF* dan *Kmeans*

#### 4.3.1. Analisis Pengguna

Pada bagian analisis pengguna peneliti mendapati pengguna yang akan menggunakan sistem ini adalah pengajar khususnya dosen. Namun disini peneliti memutuskan pengguna dari sistem ini tidak hanya dosen saja, tetapi dibagi menjadi tiga karena tujuan umum dari sistem ini adalah untuk mempermudah dosen, sehingga untuk target penggunanya ada tiga yaitu admin, dosen, dan mahasiswa.

Admin adalah pengguna yang dapat mengelola *user* dan kategori soal. Dosen adalah pengguna yang dapat mengelola soal berdasarkan kategorinya, melihat jawaban mahasiswa, dan melihat hasil dari proses klaster, sedangkan mahasiswa adalah pengguna yang hanya dapat menjawab soal saja

#### 4.3.2. Analisis Kebutuhan Fungsional

Pada analisis kebutuhan fungsional ini peneliti akan mengidentifikasi proses apa saja yang dapat dilakukan oleh sistem, proses-proses yang dapat dilakukan oleh sistem dapat dilihat pada tabel 4.1.

Tabel 4. 1. Kebutuhan Fungsional Sistem

Pengguna	Proses
Sistem	Dapat membedakan pengguna berdasarkan role nya (admin, dosen, mahasiswa)
Admin	Dapat login menggunakan username, password dan logout dari sistem
	Dapat melakukan proses CRUD (create, read, update, delete) terhadap data pengguna
	Dapat melakukan proses CRUD (create, read, update, delete) terhadap data kategori
Dosen	Dapat login menggunakan username, password dan logout dari sistem

	Dapat melakukan proses CRUD (create, read, update, delete) terhadap data soal
	Dapat melihat data jawaban berdasarkan soal
	Dapat melihat data klaster terbaik berdasarkan soal yang dipilih
Mahasiswa	Dapat login menggunakan username, password dan logout dari sistem
	Dapat memasukkan data jawaban terhadap soal yang dipilih

### 4.3.3. Analisis Kebutuhan Non Fungsional

Kebutuhan non-fungsional adalah kebutuhan perangkat lunak atau *software* dan kebutuhan perangkat keras atau *hardware* yang digunakan peneliti untuk membuat sistem. Rincian kebutuhan non-fungsional sistem berupa perangkat lunak dapat dilihat pada tabel 4.2. Sedangkan rincian kebutuhan perangkat keras dapat dilihat pada tabel 4.3.

Tabel 4. 2. Kebutuhan Perangkat Lunak

Perangkat Lunak	Spesifikasi
Sistem Operasi	Windows 11 Home
Browser	Chrome
Compiler	Google Colab, Visual Studio Code
Framework	Flask, React
Bahasa Pemrograman	Python 3.11.4, HTML, CSS, Javascript
Database	MySQL

Tabel 4. 3. Kebutuhan Perangkat Keras

Perangkat Keras	Spesifikasi
PC/Laptop	HP Pavilion Gaming 15
Processor	AMD Ryzen 5 5600H @ 3.30GHz
Memory	16 GB
Storage	512 GB

#### 4.4. Perancangan Sistem

Tahap perancangan sistem merupakan tahapan yang melibatkan perancangan sistem secara keseluruhan mulai dari perancangan data, perancangan algoritma, perancangan kebutuhan fungsional, perancangan antarmuka, dan perancangan pengujian.

##### 4.4.1. Perancangan Data

Perancangan data merupakan tahap pertama, pada tahap ini peneliti akan menunjukkan bagaimana merancang data sehingga data tersebut siap diteruskan ke proses selanjutnya. Dalam tahap ini data yang masuk akan disebut data kotor dimana data ini akan dilakukan *preprocessing* yang akan menghasilkan data bersih. Contoh dari data kotor dapat dilihat pada tabel 4.4.

Tabel 4. 4. Contoh Data Kotor

Soal	Jawaban (Data Kotor)
Menurut anda, bagaimana prospek pekerjaan pada bidang BI	Menurut saya diera sekarang skil pengolahan data seperti pada bidang BI sangat dibutuhkan di perusahaan-perusahaan. Karena dengan teknik pengolahan dan analisis data dapat membantu perusahaan untuk meningkatkan keuntungan dan omset perusahaan

*Preprocessing* disini menggunakan bantuan dari *library* Sastrawi pada proses *stemming*, dan *stopword removal*. *Preprocessing* sendiri secara berurutan terdiri dari proses *case folding*, *stemming*, dan *stopword removal* dimana satu proses dilakukan setelah proses sebelumnya selesai. Berikut detail dari setiap prosesnya.

##### 1. Case Folding

*Case Folding* merupakan metode yang digunakan untuk mengubah atau menghilangkan semua huruf kapital yang ada pada sebuah dokumen menjadi huruf kecil. Contoh hasil proses *case folding* dapat dilihat pada tabel 4.5.

Tabel 4. 5. Contoh Hasil *Case Folding*

Data Kotor	Hasil <i>Case Folding</i>
------------	---------------------------

Menurut saya di era sekarang skill pengolahan data seperti pada bidang BI sangat dibutuhkan di perusahaan-perusahaan. Karena dengan teknik pengolahan dan analisis data dapat membantu perusahaan untuk meningkatkan keuntungan dan omset perusahaan	menurut saya di era sekarang skill pengolahan data seperti pada bidang <b>bi</b> sangat dibutuhkan di perusahaan-perusahaan. <b>karena</b> dengan teknik pengolahan dan analisis data dapat membantu perusahaan untuk meningkatkan keuntungan dan omset perusahaan.
--	---

## 2. Tokenization

*Tokenization* merupakan tahapan penguraian *string* teks menjadi *term* atau kata. Tujuan dari *Tokenization* adalah memisahkan kata-kata dalam sebuah paragraf, kalimat, atau halaman ke dalam kata tunggal. Contoh hasil proses *tokenization* dapat dilihat pada tabel 4.6.

Tabel 4. 6. Contoh Hasil *Tokenization*

Hasil <i>Case Folding</i>	Hasil <i>Tokenization</i>
menurut saya di era sekarang skill pengolahan data seperti pada bidang bi sangat dibutuhkan di perusahaan-perusahaan. karena dengan teknik pengolahan dan analisis data dapat membantu perusahaan untuk meningkatkan keuntungan dan omset perusahaan.	['menurut', 'saya', 'diera', 'sekarang', 'skil', 'pengolahan', 'data', 'seperti', 'pada', 'bidang', 'bi', 'sangat', 'dibutuhkan', 'di', 'perusahaan-perusahaan.', 'karena', 'dengan', 'teknik', 'pengolahan', 'dan', 'analisis', 'data', 'dapat', 'membantu', 'perusahaan', 'untuk', 'meningkatkan', 'keuntungan', 'dan', 'omset', 'perusahaan.']

## 3. Stemming

*Stemming* merupakan tahapan pengubahan suatu kata menjadi kata dasarnya. Contoh hasil proses *stemming* dapat dilihat pada tabel 4.7.

Tabel 4. 7. Contoh Hasil *Stemming*

Hasil <i>Tokenization</i>	Hasil <i>Stemming</i>
['menurut', 'saya', 'diera', 'sekarang', 'skil', 'pengolahan', 'data', 'seperti', 'pada', 'bidang', 'bi', 'sangat', 'dibutuhkan', 'di', 'perusahaan-perusahaan.', 'karena', 'dengan', 'teknik', 'pengolahan', 'dan', 'analisis', 'data', 'dapat', 'membantu', 'perusahaan', 'untuk', 'meningkatkan',	['turut', 'saya', 'era', 'sekarang', 'skil', 'olah', 'data', 'seperti', 'pada', 'bidang', 'bi', 'sangat', 'butuh', 'di', 'usaha', 'karena', 'dengan', 'teknik', 'olah', 'dan', 'analisis', 'data', 'dapat', 'bantu', 'usaha', 'untuk', 'tingkat', 'untung', 'dan', 'omset', 'usaha']

'keuntungan', 'dan', 'omset', 'perusahaan.']	
--	--

#### 4. *Stopword Removal*

*Stopword removal*, merupakan tahapan penghapusan kata-kata yang tidak relevan dalam penentuan topik sebuah dokumen dan kata yang sering muncul pada dokumen, misalnya “dan”, “atau”, “sebuah”, “adalah”, pada dokumen berbahasa Indonesia. Contoh hasil proses *stopword removal* dapat dilihat pada tabel 4.8.

Tabel 4. 8. Contoh Hasil *Stopword Removal*

Hasil <i>Stemming</i>	Hasil <i>Stopword Removal</i>
['turut', 'saya', 'era', 'sekarang', 'skil', 'olah', 'data', 'seperti', 'pada', 'bidang', 'bi', 'sangat', 'butuh', 'di', 'usaha', 'karena', 'dengan', 'teknik', 'olah', 'dan', 'analisis', 'data', 'dapat', 'bantu', 'usaha', 'untuk', 'tingkat', 'untung', 'dan', 'omset', 'usaha']	turut era sekarang skil olah data pada bidang bi sangat butuh usaha dengan teknik olah analisis data bantu usaha tingkat untung omset usaha

Setelah preprocessing selesai data yang diproses akan menjadi data bersih. Contoh perbandingan data kotor dan data bersih dapat dilihat pada tabel 4.9.

Tabel 4. 9. Contoh Perbandingan Data Kotor dan Data Bersih

Data Kotor	Data Bersih
Menurut saya di era sekarang skil pengolahan data seperti pada bidang BI sangat dibutuhkan di perusahaan-perusahaan. Karena dengan teknik pengolahan dan analisis data dapat membantu perusahaan untuk meningkatkan keuntungan dan omset perusahaan	turut era sekarang skil olah data pada bidang bi sangat butuh usaha dengan teknik olah analisis data bantu usaha tingkat untung omset usaha

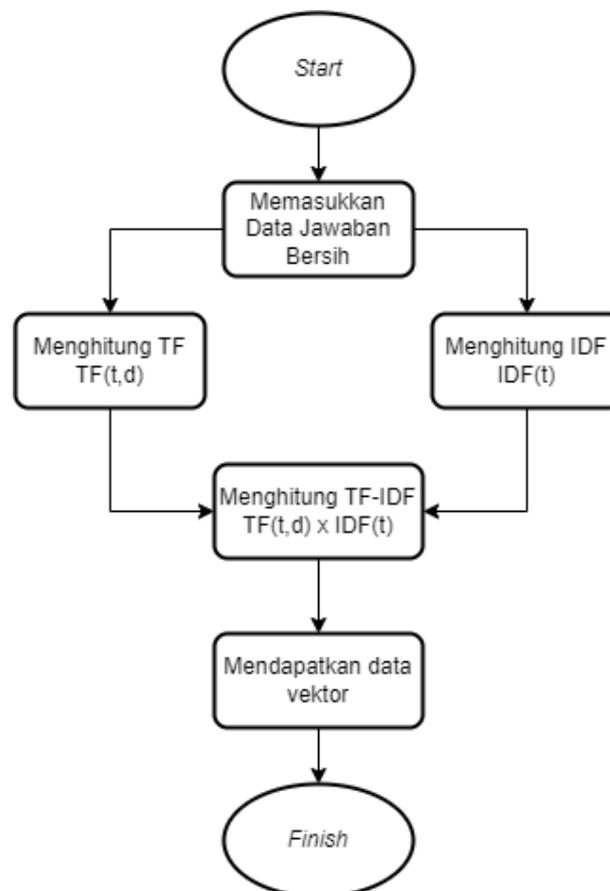
#### 4.4.2. Perancangan Algoritma

Pada perancangan algoritma ini penulis akan memaparkan rancangan algoritma apa saja yang akan diimplementasikan pada data, proses ini bertujuan untuk menghasilkan informasi yang dicari dari data yang dimiliki. Ada tiga algoritma yang akan digunakan dalam proses ini yaitu *TF-IDF* sebagai *vectorizer*,

*Kmeans* untuk proses *clustering* dan *silhouette coefficient* sebagai *tester* hasil *clustering*.

### 1. *TF-IDF*

Data yang bersih akan masuk ke tahap pembobotan menggunakan metode *TF-IDF*. Proses ini bertujuan untuk mengubah data teks menjadi data vektor karena untuk melakukan proses selanjutnya yaitu *clustering* dibutuhkan data vektor. Contoh algoritma *TF-IDF* dapat dilihat pada gambar 4.1.



Gambar 4. 1. Alur Proses *TF-IDF*

*TF-IDF* adalah metode ekstraksi fitur yang mengukur nilai sebuah kata dengan cara membandingkan frekuensi kata tersebut muncul dalam sebuah kalimat atau dalam kasus ini dalam sebuah jawaban dengan jumlah jawaban dimana kata tersebut muncul. Prosesnya kurang lebih sebagai berikut:

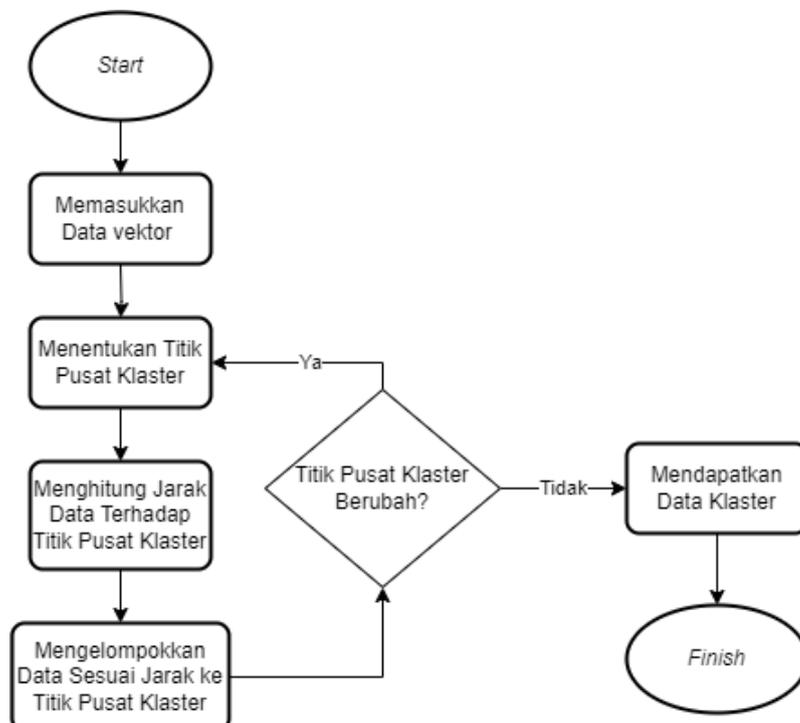
- A. Setelah data bersih dimasukkan maka akan dilakukan dua proses secara bersamaan yaitu menghitung  $TF(t,d)$  dan  $IDF(t)$ .  $TF$

sendiri adalah frekuensi sebuah kata muncul dalam satu jawaban contoh jawaban “*TF* singkatan dari *Term Frequency*” didapatkan nilai *TF* (singkatan) adalah berapa kali kata “singkatan” muncul pada jawaban dibagi ada berapa kata dalam jawaban tersebut. Didapat nilai *TF* (singkatan) adalah  $1/5$ , sedangkan *IDF* adalah invers frekuensi dokumen, *IDF* ini berfungsi untuk mencari seberapa sering sebuah kata muncul dalam semua jawaban, dimana *IDF* (singkatan) =  $\log(\text{jumlah jawaban}/\text{jumlah jawaban yang mempunyai kata "singkatan"})$ .

- B. Setelah didapat nilai  $TF(t,d)$  dan  $IDF(t)$  akan didapat nilai  $TF \cdot IDF$  dengan cara mengalikan  $TF$  dengan  $IDF$  dimana data hasil perkalian ini akan menjadi data terekstrak atau data vektor.

## 2. *Kmeans*

Setelah mendapatkan data vektor maka proses selanjutnya adalah *clustering* menggunakan algoritma *Kmeans*. Proses ini bertujuan untuk mengelompokkan data berdasarkan ciri atau hasil pembobotan pada proses sebelumnya. Untuk alur proses *Kmeans clustering* ditunjukkan pada gambar 4.2.



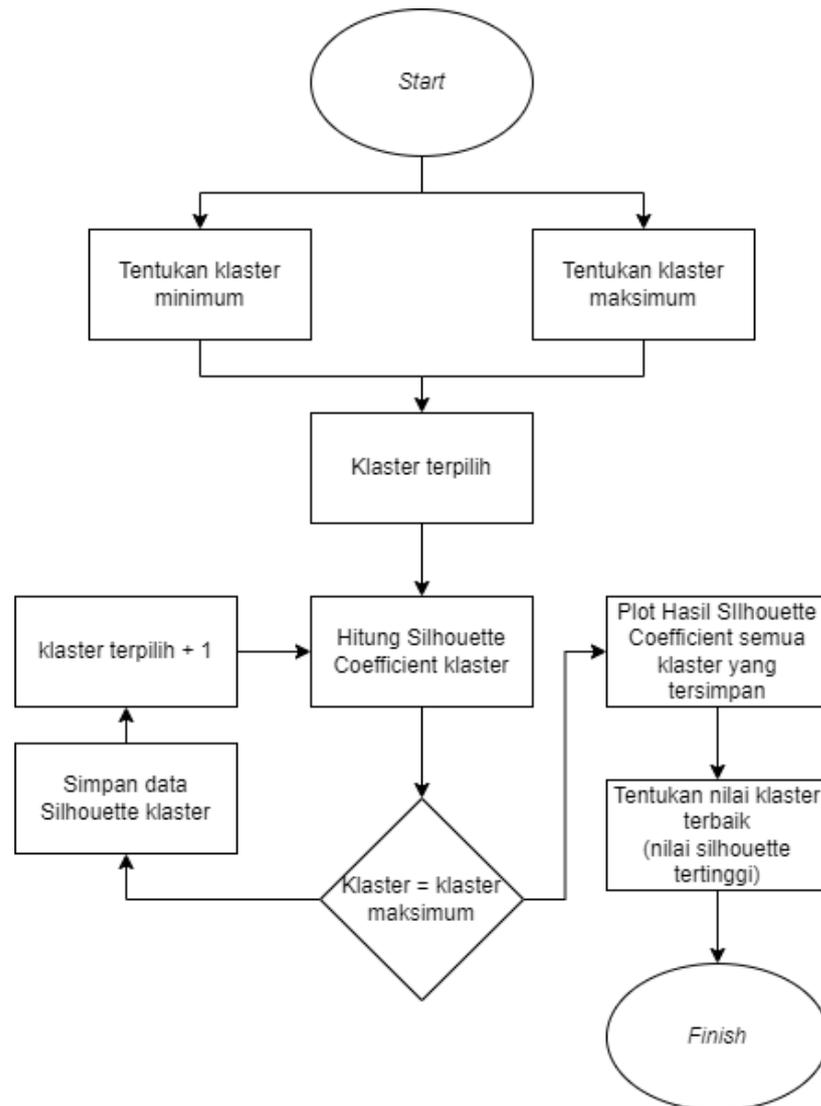
Gambar 4. 2. Alur Proses *Kmeans*

*Kmeans* adalah salah satu algoritma *clustering* yang cukup populer karena kecepatan dan skalabilitasnya. *Kmeans* menggunakan proses iterasi pusat dari klaster atau disebut *centroid* ke posisi rata-rata setiap data dan mengelompokkan data tersebut ke klaster dengan *centroid* terdekat menggunakan *euclidean distance*. Proses dari *kmeans clustering* kurang lebih sebagai berikut:

- A. Data vektor dimasukkan sebagai *input* dan sistem akan otomatis melakukan *looping* atau pengulangan terhadap setiap data jumlah klaster yang memungkinkan
- B. Selanjutnya menentukan titik pusat klaster. Titik ini berfungsi sebagai patokan jarak yang digunakan untuk mengelompokkan data.
- C. Selanjutnya menghitung jarak data terhadap titik pusat klaster.
- D. Selanjutnya data yang sudah dihitung jaraknya akan dikelompokkan sesuai jarak terdekat dengan klaster.
- E. Selanjutnya ditentukan titik pusat baru dimana titik ini didapat dari nilai rata-rata data yang masuk ke klaster tersebut.
- F. Setelah itu kembali dilakukan kembali langkah ke tiga sampai lima.
- G. Jika titik pusat terbaru nilainya sama dengan titik pusat pada iterasi sebelumnya, maka penentuan titik pusat baru dapat dihentikan dan telah diperoleh *output* berupa hasil klaster.

### 3. *Silhouette Coefficient*

Hasil dari *clustering* ini selanjutnya akan masuk ke proses *testing* menggunakan *silhouette coefficient*. Hasil *testing* ini dilakukan untuk mencari tau jumlah klaster terbaik yang akan direkomendasikan sistem untuk ditampilkan pada *user* sehingga *user* tidak perlu menentukan jumlah klaster. Alur algoritma *silhouette coefficient* dapat dilihat pada gambar 4.3.



Gambar 4. 3. Alur Proses *Silhouette Coefficient*

Pada gambar diatas dilakukan *testing* menggunakan *silhouette coefficient* yang akan menghitung jarak rata-rata data antar klaster dibagi dengan jarak rata rata terdekat data dengan klaster. Proses dari algoritma *silhouette coefficient* kurang lebih sebagai berikut.

- A. Menentukan jumlah klaster minimum dan maksimum, hal ini berguna untuk proses pengulangan dimana pada proses *testing* data semua jumlah klaster akan diproses.
- B. Setelah didapat jumlah klaster terpilih, sistem akan menghitung nilai *silhouette*
- C. Setelah itu akan ada kondisi yang memeriksa apakah jumlah klaster terpilih sama dengan klaster maksimum? Jika tidak maka

data akan disimpan dan data jumlah klaster terpilih akan ditambah 1, lalu dilakukan perhitungan *silhouette* kembali menggunakan data jumlah klaster yang baru. Jika kondisi terpenuhi maka dilakukan *plotting* atau pembuatan grafik dari semua data yang tersimpan

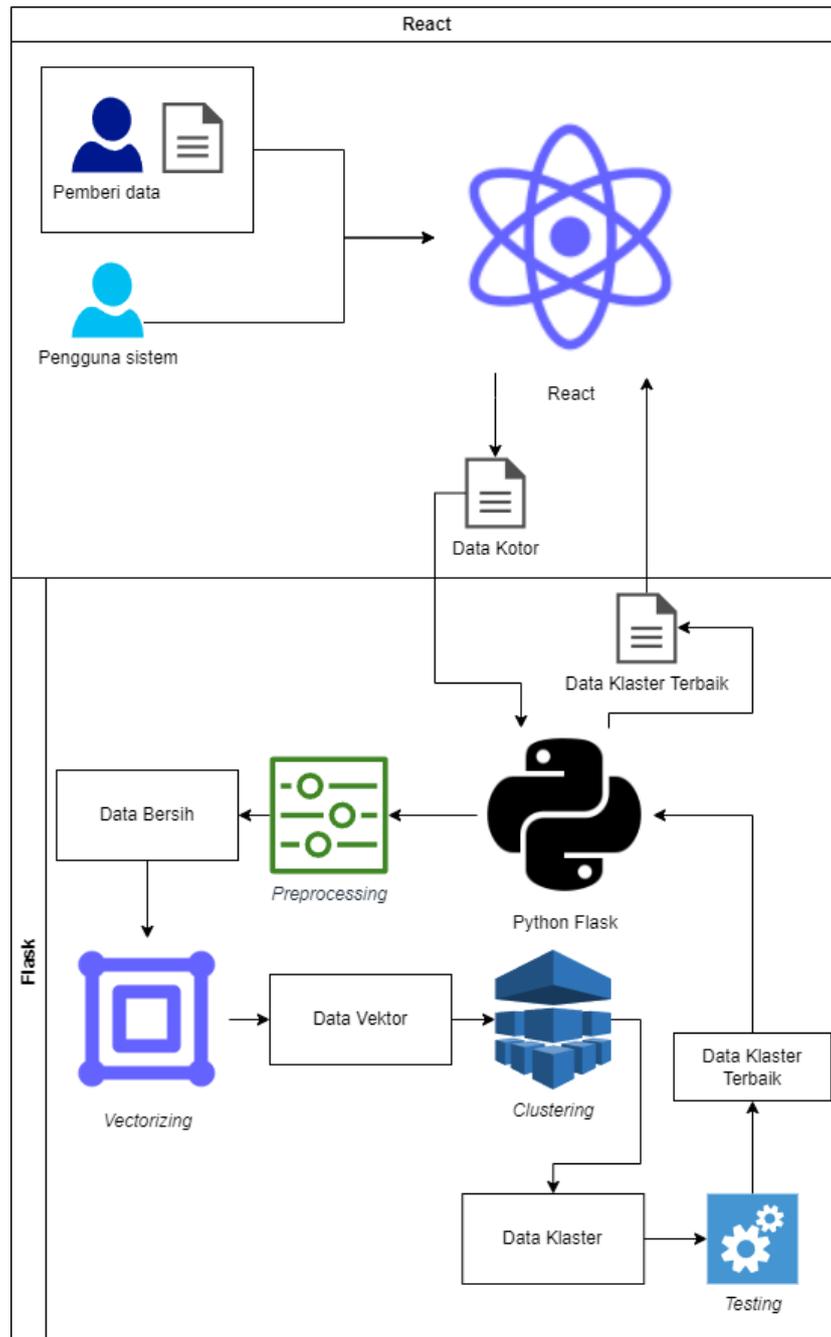
- D. Selanjutnya berdasarkan grafik yang telah dibuat akan diambil data jumlah klaster dengan nilai *silhouette* tertinggi yang akan menjadi data klaster terbaik.

#### **4.4.3. Perancangan Kebutuhan Fungsional**

Pada perancangan kebutuhan fungsional peneliti akan memaparkan hal-hal yang diperlukan untuk memenuhi kebutuhan fungsional sistem seperti, perancangan arsitektur dan perancangan *database* sistem.

1. Perancangan Arsitektur

Berdasarkan analisis kebutuhan fungsional, dibentuklah perancangan arsitektur sistem dapat dilihat pada gambar 4.4.



Gambar 4. 4. Arsitektur Sistem

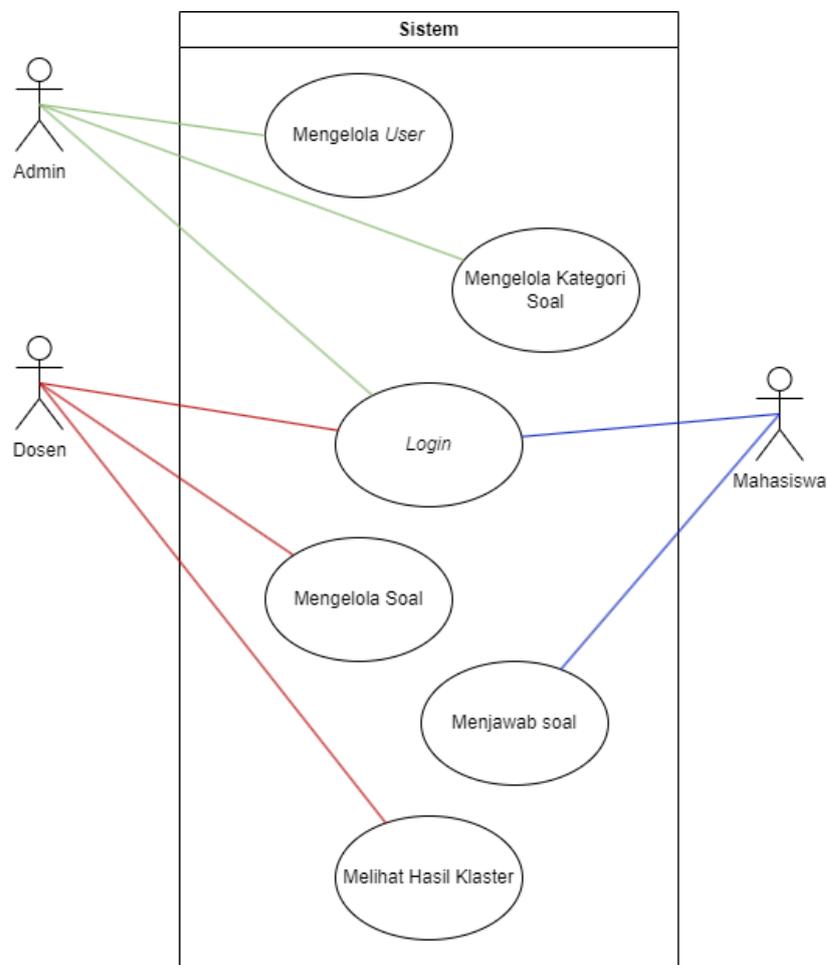
Pada gambar arsitektur sistem diatas dapat dilihat bahwa sistem yang akan dibuat akan menggunakan react dan flask sebagai *framework*, dengan flask python sebagai tempat untuk mengolah data utama dan reactjs sebagai tempat proses-proses pendukung dan untuk menampilkan data.

## 2. Perancangan Diagram

Pada perancangan diagram ini akan dipaparkan diagram-diagram yang diharapkan dapat membantu memahami proses realisasi kebutuhan fungsional user. Diagram yang digunakan terdiri dari dua jenis yaitu use case diagram dan activity diagram.

#### A. Use Case Diagram

Use case diagram berfungsi untuk menggambarkan siapa saja pengguna dan apa saja yang bisa pengguna lakukan. Untuk gambaran use case diagram dapat dilihat pada gambar 4.5.

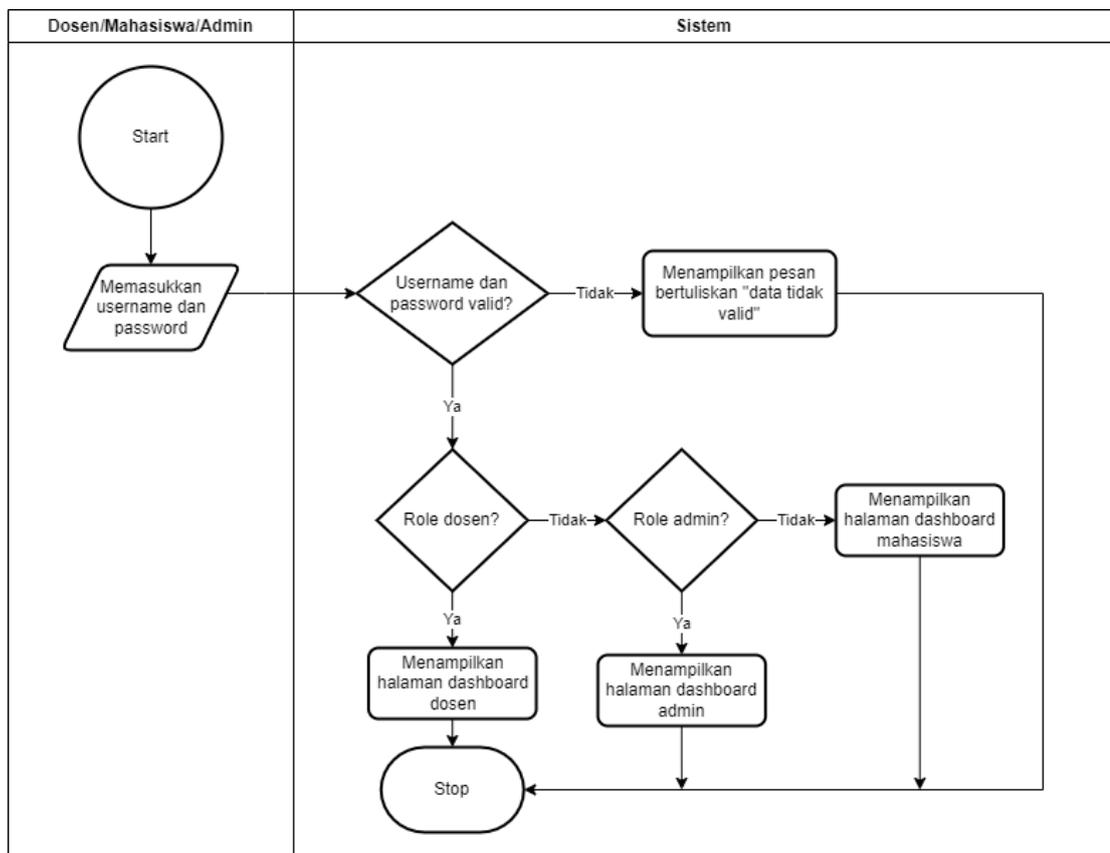


Gambar 4. 5. Use Case Diagram

#### B. Activity Diagram Login

Activity diagram login berfungsi untuk menggambarkan apa saja langkah-langkah yang dilakukan

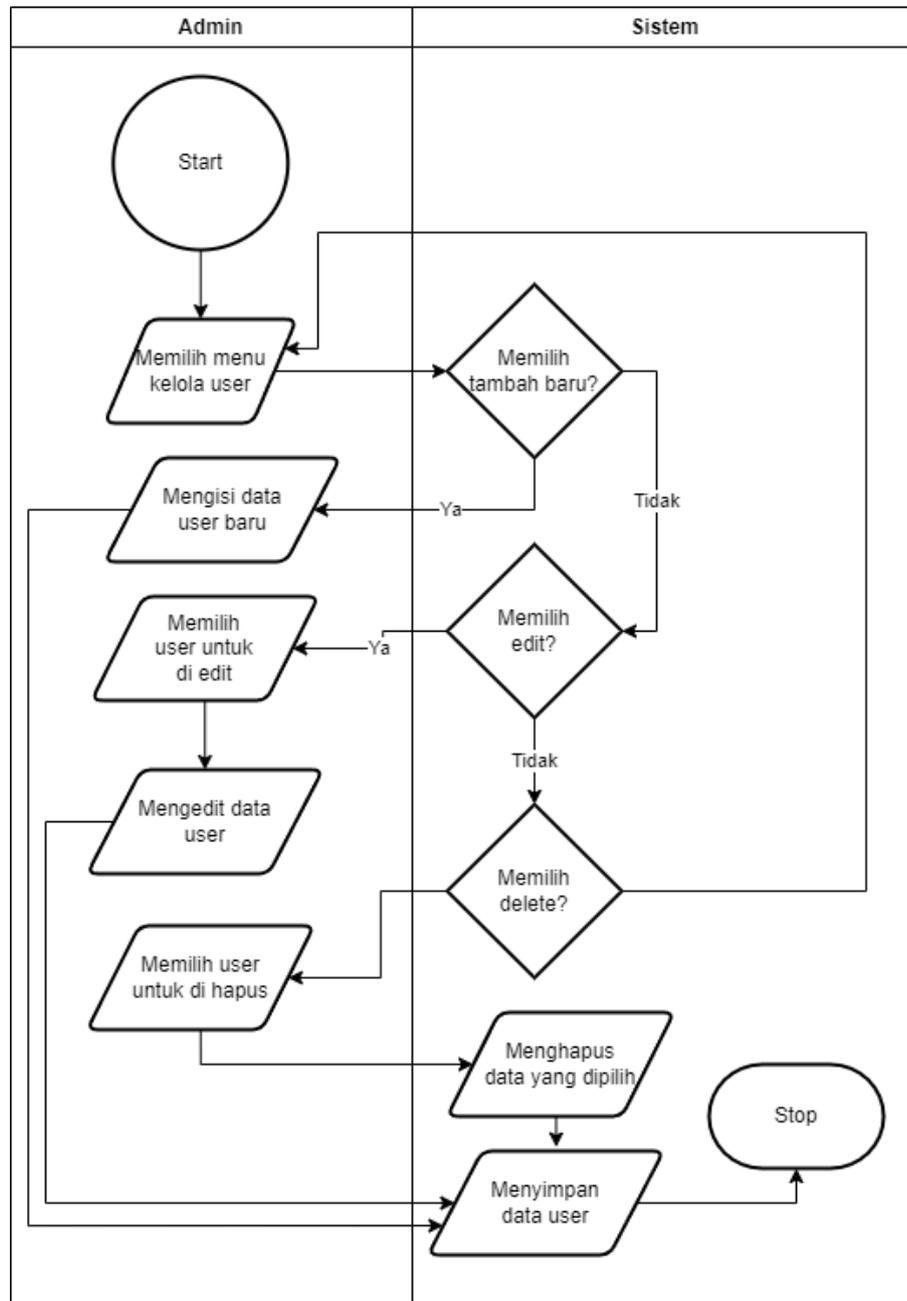
pada proses login. Untuk gambaran use case diagram dapat dilihat pada gambar 4.6.



Gambar 4. 6. Activity Diagram Login

### C. Activity Diagram Kelola User

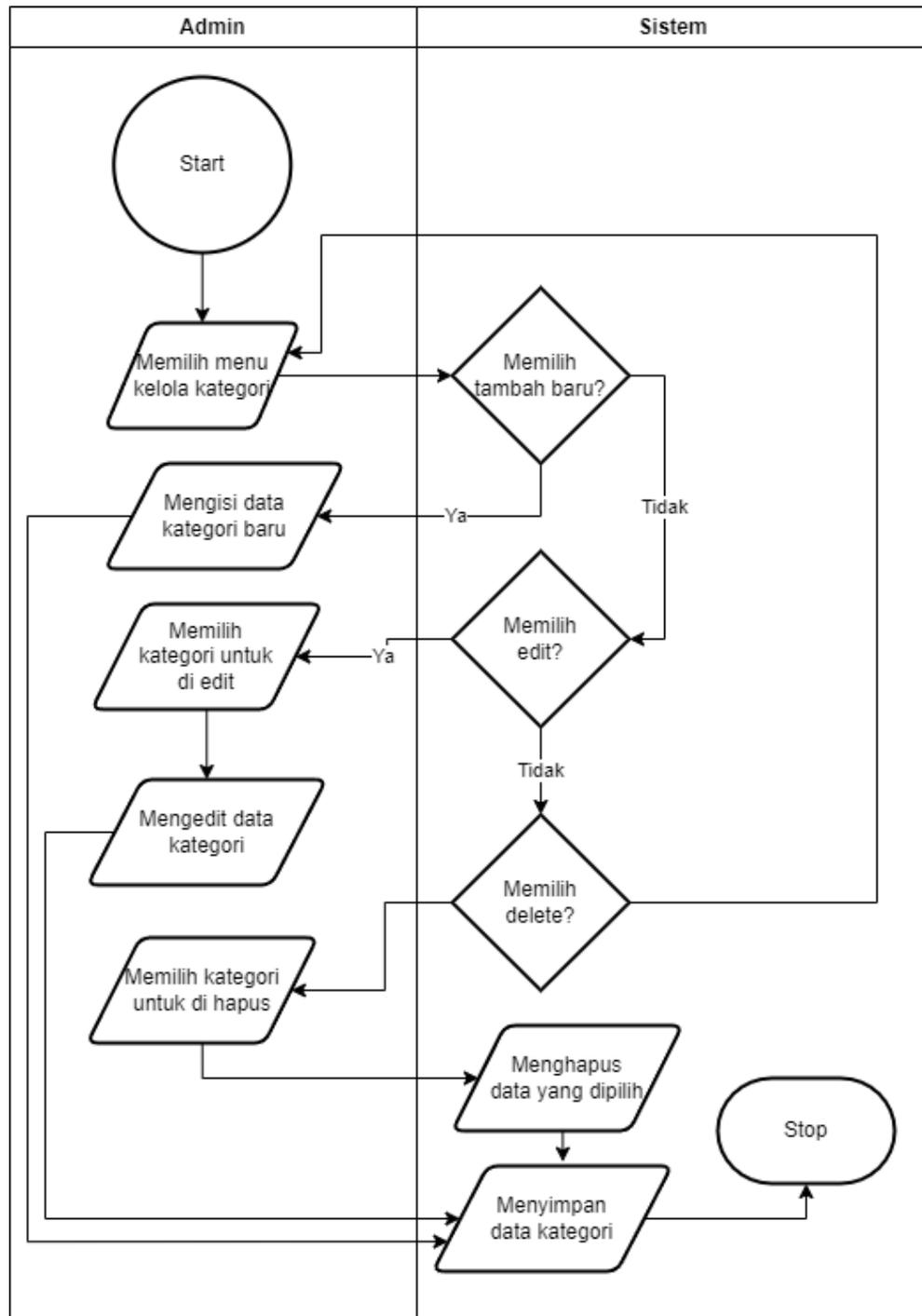
Activity diagram kelola user berfungsi untuk menggambarkan apa saja langkah-langkah yang dilakukan pada proses kelola user. Untuk gambaran use case diagram dapat dilihat pada gambar 4.7.



Gambar 4. 7. Activity Diagram Kelola User

#### D. Activity Diagram Kelola Kategori

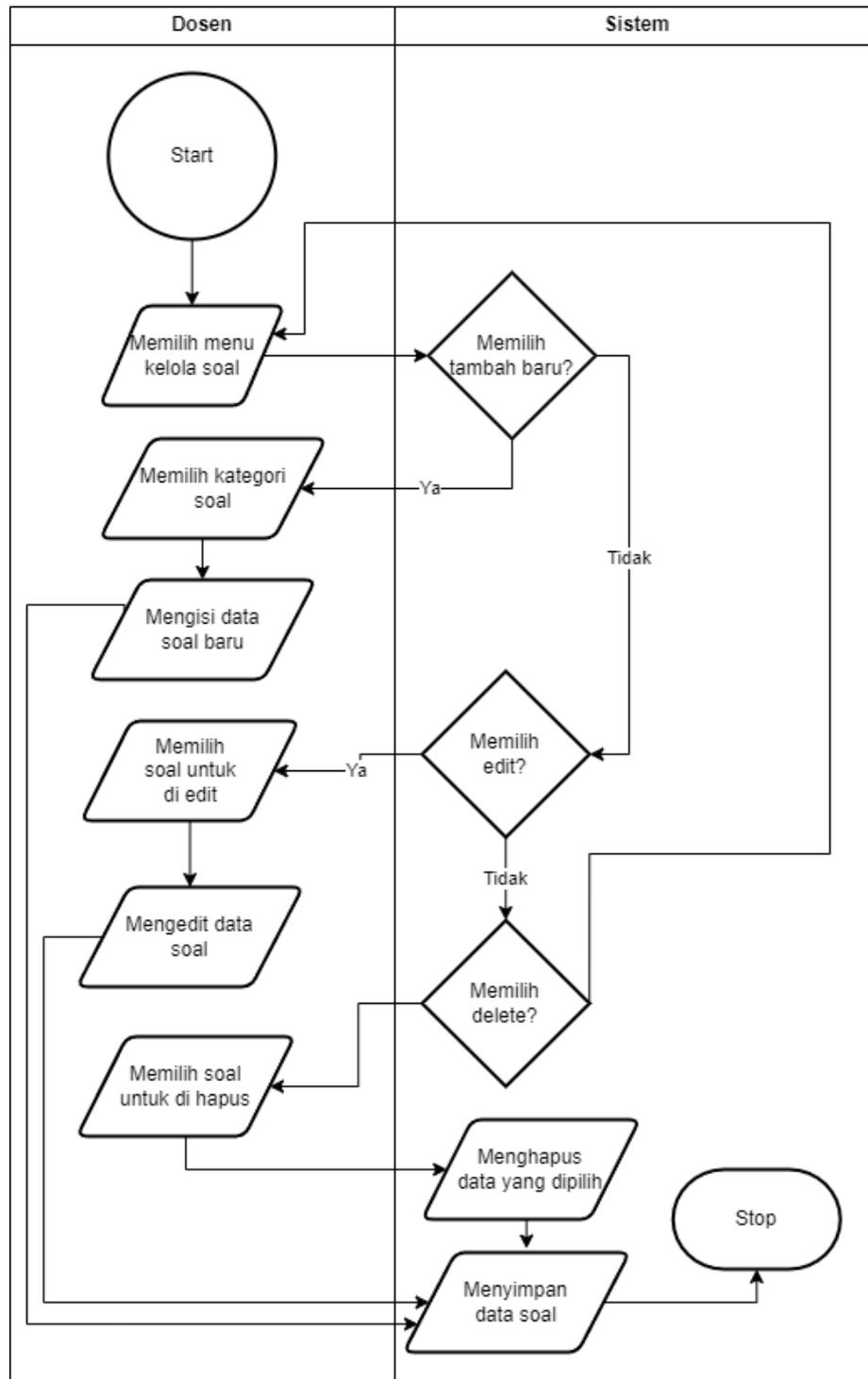
Activity diagram kelola kategori berfungsi untuk menggambarkan apa saja langkah-langkah yang dilakukan pada proses kelola kategori. Untuk gambaran use case diagram dapat dilihat pada gambar 4.8.



Gambar 4. 8. Activity Diagram Kelola Kategori

#### E. Activity Diagram Kelola Soal

Activity diagram kelola soal berfungsi untuk menggambarkan apa saja langkah-langkah yang dilakukan pada proses kelola soal. Untuk gambaran use case diagram dapat dilihat pada gambar 4.9.

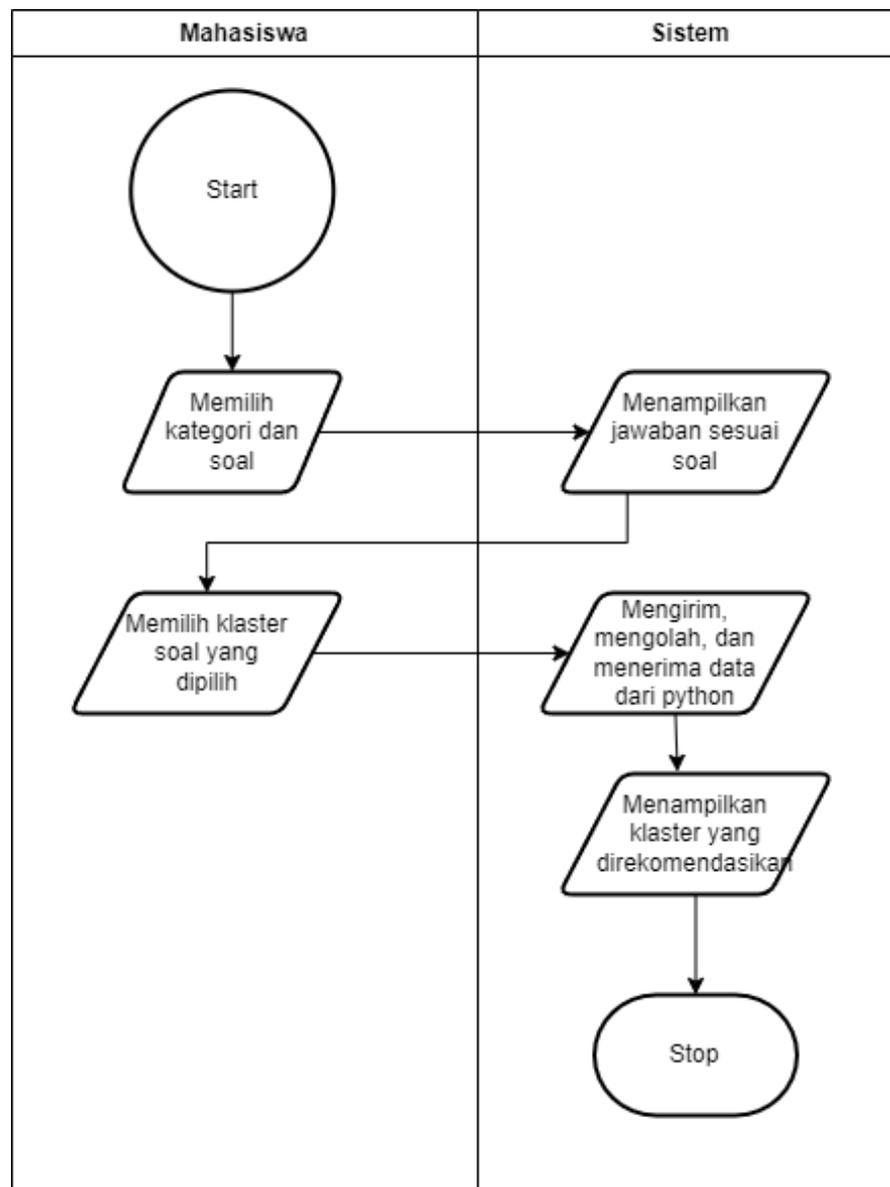


Gambar 4. 9. Activity Diagram Kelola Soal

#### F. Activity Diagram Lihat Kluster

Activity diagram lihat kluster berfungsi untuk menggambarkan apa saja langkah-langkah yang dilakukan

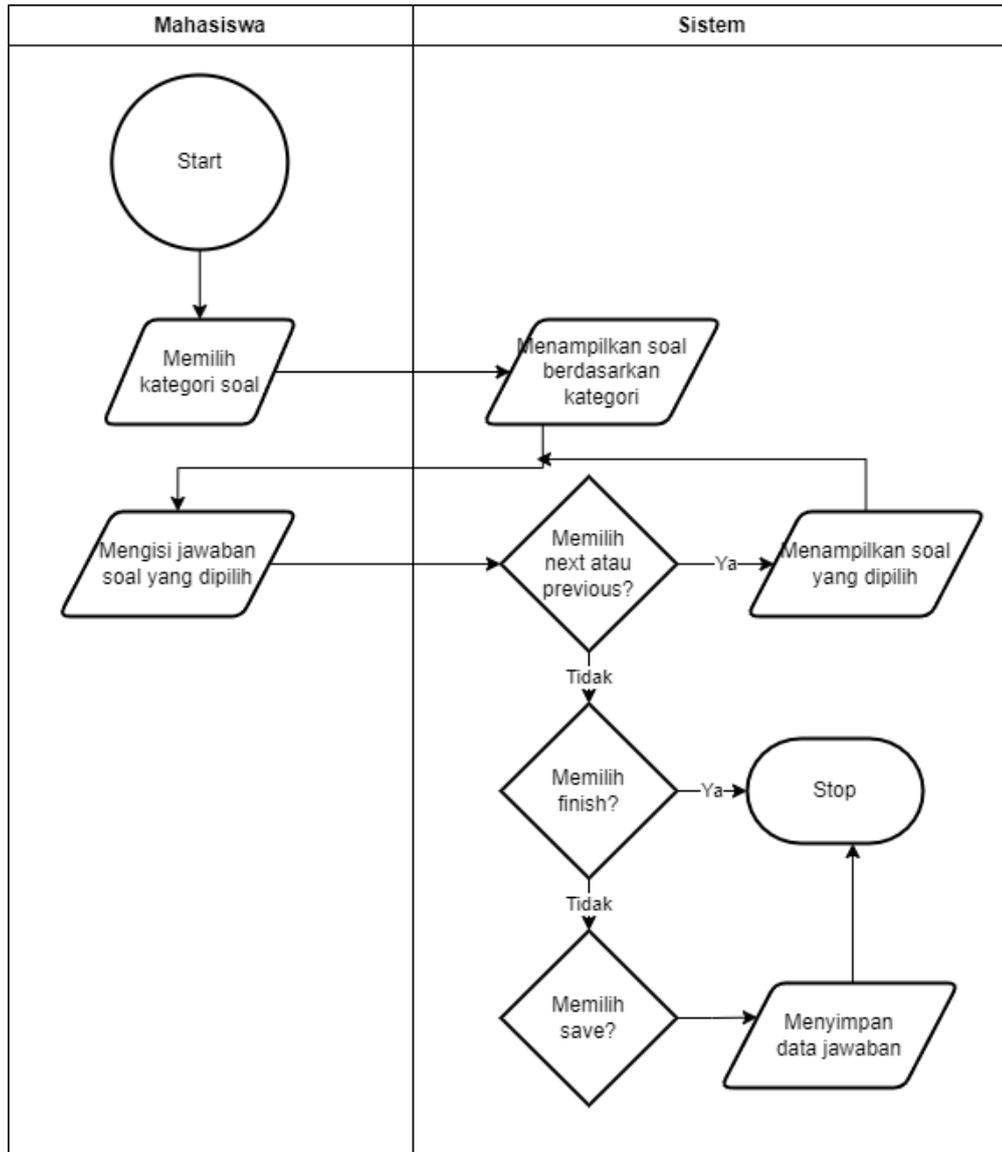
pada proses lihat kluster. Untuk gambaran use case diagram dapat dilihat pada gambar 4.10.



Gambar 4. 10. Activity Diagram Lihat Kluster

#### G. Activity Diagram Jawab Soal

Activity diagram jawab soal berfungsi untuk menggambarkan apa saja langkah-langkah yang dilakukan pada proses jawab soal. Untuk gambaran use case diagram dapat dilihat pada gambar 4.11.



Gambar 4. 11. Activity Diagram Jawab Soal

### 3. Perancangan Database

Pada perancangan database disini peneliti menggunakan database mysql dengan bahasa pemrograman javascript. Berdasarkan analisis kebutuhan fungsional setidaknya ada lima tabel yang diperlukan oleh sistem yaitu tabel untuk *user*, kategori, soal, jawaban, dan *session*.

#### A. Tabel Users

Tabel user berguna untuk menyimpan data user atau pengguna dari sistem. Disini pengguna membutuhkan id dengan tipe data integer, dan uuid dengan tipe data varchar sebagai kode unik, nama dengan tipe data varchar sebagai identitas utama, username

dan password dengan tipe data varchar untuk keperluan login, role dengan tipe data varchar untuk membedakan jenis pengguna dan createdAt, updatedAt dengan tipe data datetime untuk dokumentasi waktu ketika ada perubahan data. Untuk rincian tabel user dapat dilihat pada tabel 4.10.

Tabel 4. 10. Rancangan Tabel Users

users	
Kolom	Tipe Data
id	int
uuid	varchar
name	varchar
username	varchar
password	varchar
role	varchar
createdAt	datetime
updatedAt	datetime

#### B. Tabel Kategoris

Tabel kategoris berguna untuk menyimpan data kategori soal atau. Disini kategori membutuhkan id dengan tipe data integer, dan uuid dengan tipe data varchar sebagai kode unik, nama dengan tipe data varchar sebagai atribut utama, userId dengan tipe data integer sebagai foreign key untuk id dari tabel user dan createdAt, updatedAt dengan tipe data datetime untuk dokumentasi waktu ketika ada perubahan data. Untuk rincian tabel user dapat dilihat pada tabel 4.11.

Tabel 4. 11. Rancangan Tabel Kategoris

kategoris	
Kolom	Tipe Data
id	int
uuid	varchar

name	varchar
userId	int
createdAt	datetime
updatedAt	datetime

### C. Tabel Soals

Tabel soals berguna untuk menyimpan data soal dosen. Disini soal membutuhkan id dengan tipe data integer, dan uuid dengan tipe data varchar sebagai kode unik, nama dengan tipe data varchar sebagai atribut utama, userId dengan tipe data integer sebagai foreign key dari tabel user yang berguna untuk mengambil data user, kategoriId dengan tipe data integer sebagai foreign key dari tabel kategoris yang berguna untuk mengambil data kategori, dan createdAt, updatedAt dengan tipe data datetime untuk dokumentasi waktu ketika ada perubahan data. Untuk rincian tabel user dapat dilihat pada tabel 4.12.

Tabel 4. 12. Rancangan Tabel Soals

soals	
Kolom	Tipe Data
id	int
uuid	varchar
name	varchar
userId	int
kategoriId	int
createdAt	datetime
updatedAt	datetime

### D. Tabel Jawabs

Tabel jawabs berguna untuk menyimpan data jawaban mahasiswa. Disini jawaban membutuhkan id dengan tipe data integer, dan uuid dengan tipe data varchar sebagai kode unik, nama dengan tipe data text sebagai atribut utama, userId dengan tipe data

integer sebagai foreign key dari tabel user yang berguna untuk mengambil data user, kategoriId dengan tipe data integer sebagai foreign key dari tabel kategoris yang berguna untuk mengambil data kategori, soalId dengan tipe data integer sebagai foreign key dari tabel soals yang berguna untuk mengambil data soal, dan createdAt, updatedAt dengan tipe data datetime untuk dokumentasi waktu ketika ada perubahan data. Untuk rincian tabel user dapat dilihat pada tabel 4.13.

Tabel 4. 13. Rancangan Tabel Jawabs

jawabs	
Kolom	Tipe Data
id	int
uuid	varchar
name	text
userId	int
kategoriId	int
soalId	int
createdAt	datetime
updatedAt	datetime

#### E. Tabel Sessions

Tabel sessions berguna untuk menyimpan data user yang sedang login atau aktif didalam sistem. Disini session membutuhkan sid dengan tipe data varchar sebagai kode unik, expires dengan tipe data datetime sebagai waktu berakhirnya seorang pengguna aktif, data dengan tipe data text sebagai atribut utama, dan createdAt, updatedAt dengan tipe data datetime untuk dokumentasi waktu ketika ada perubahan data. Untuk rincian tabel user dapat dilihat pada tabel 4.14.

Tabel 4. 14. Rancangan Tabel Sessions

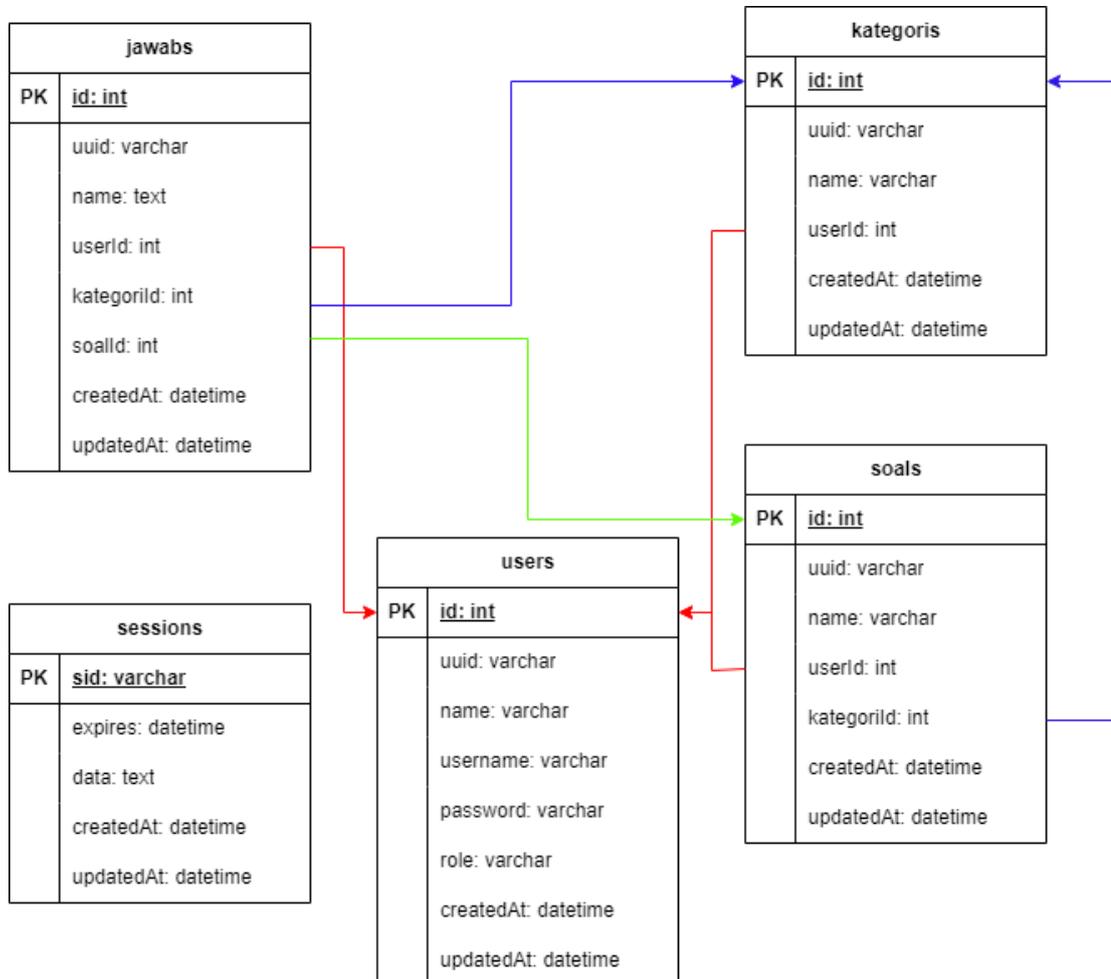
sessions	
Kolom	Tipe Data
sid	varchar
expires	datetime
data	text
createdAt	datetime
uddatedAt	datetime

#### F. Relasi Tabel

Untuk relasi tabel pada sistem terdiri dari enam relasi dengan satu jenis relasi yang sama yaitu one to many. Berikut rincian dari relasi.

- Users has many kategoris  
Relasi ini terjadi karena satu user pasti memiliki atau bisa memiliki lebih dari satu kategori
- Users has many soals  
Relasi ini terjadi karena satu user pasti memiliki atau bisa memiliki lebih dari satu soal
- Users has many jawabs  
Relasi ini terjadi karena satu user pasti memiliki atau bisa memiliki lebih dari satu jawaban
- Kategoris has many soals kategoris has many jawabs  
Relasi ini terjadi karena satu kategori pasti memiliki atau bisa memiliki lebih dari satu soal
- Kategoris has many soals kategoris has many jawabs  
Relasi ini terjadi karena satu kategori pasti memiliki atau bisa memiliki lebih dari satu jawaban
- Soal has many jawabs  
Relasi ini terjadi karena satu soal pasti memiliki atau bisa memiliki lebih dari satu jawaban

Untuk gambaran relasi antar tabel dapat dilihat pada gambar 4.12.



Gambar 4. 12. Perancangan Relasi Tabel

#### 4.4.4. Perancangan Antarmuka

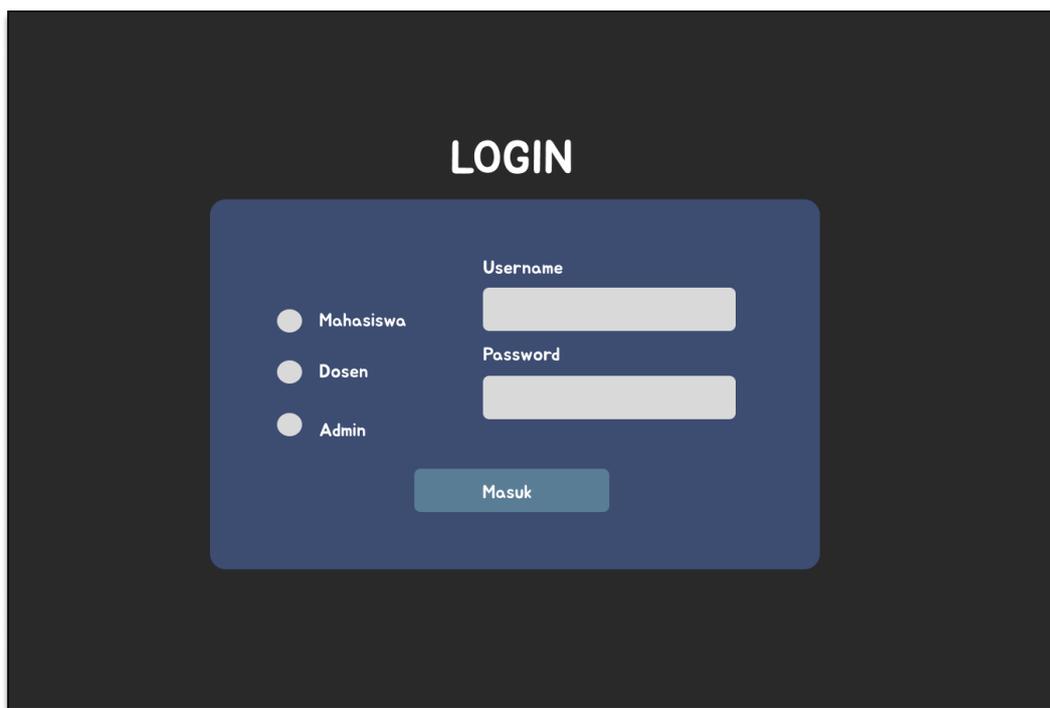
Sistem yang akan dibuat berbasis website oleh karena itu sebagai tampilan antarmuka akan menggunakan laman website. Antarmuka sistem ini dibangun menggunakan bahasa pemrograman Javascript, HTML, dan CSS dibungkus dengan framework reactjs karena cukup banyak kebutuhan fungsional yang akan dilakukan oleh sistem. Laman ini berfungsi untuk tempat memilih data mana yang akan dikelola.

Dari hasil analisis user, untuk tampilan website akan terkesan minimalis karena lebih mengedepankan fungsi. Berikut beberapa halaman menu dari sistem

yang akan dibuat. Menu yang ditampilkan adalah halaman login, kelola user, kelola kategori, kelola soal, jawab soal, dan lihat hasil klaster.

### 1. Login

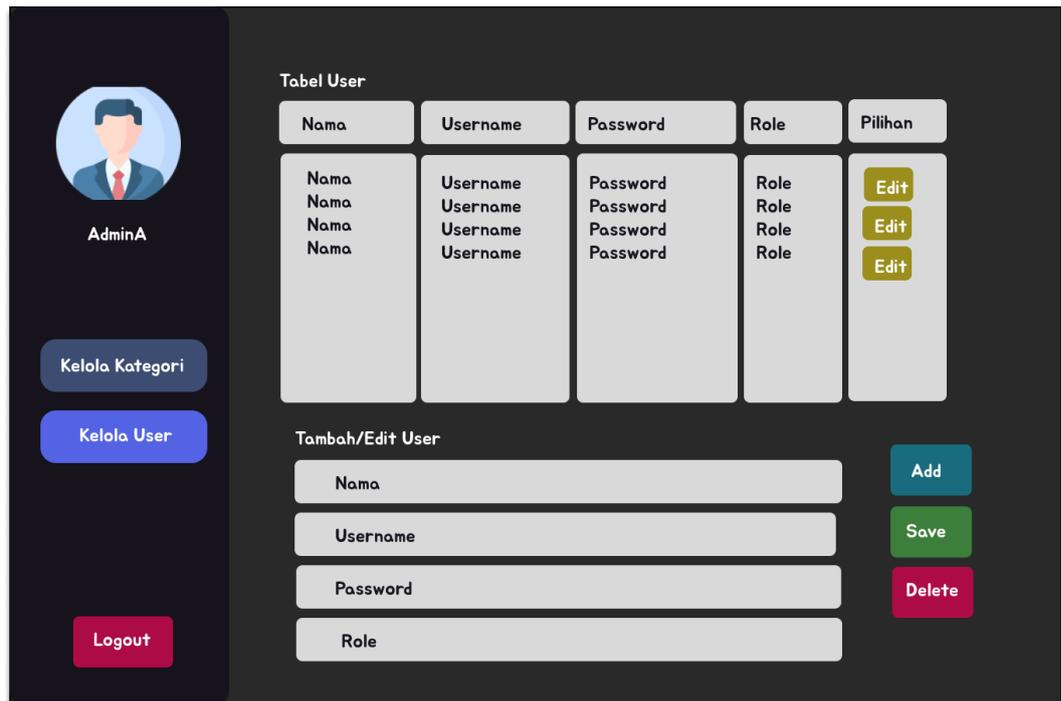
Halaman ini berfungsi untuk masuknya pengguna kedalam sistem, dimana ada 3 jenis pengguna yaitu admin, dosen, dan mahasiswa. Untuk gambar antarmuka login dapat dilihat pada gambar 4.13.



Gambar 4. 13. Perancangan Antarmuka Login

### 2. Kelola User

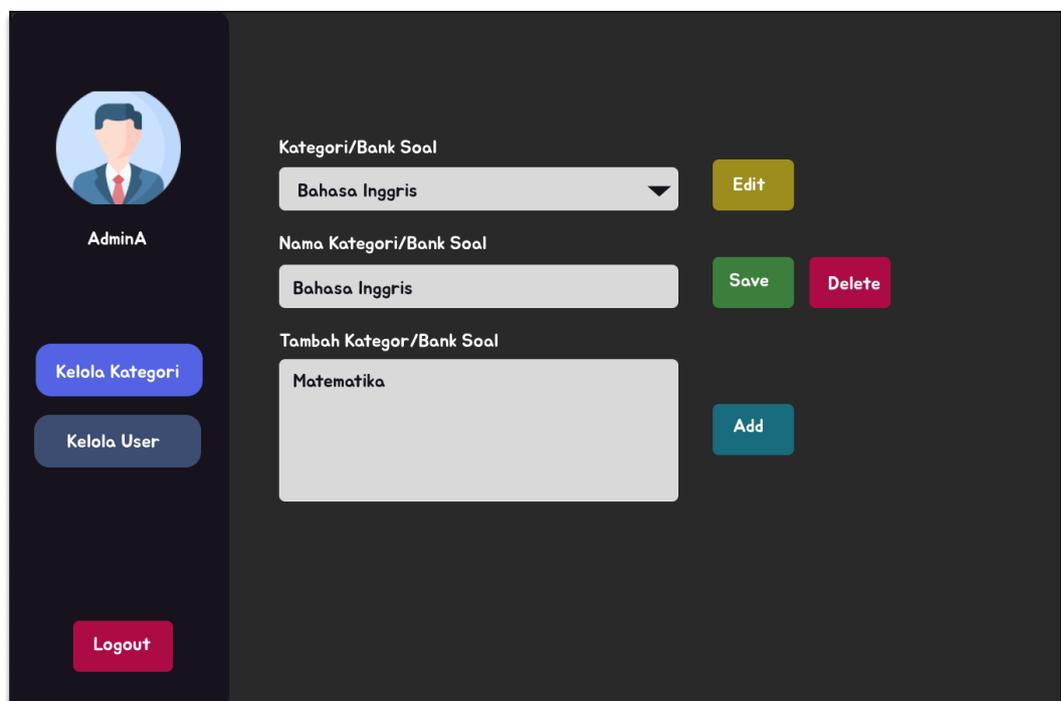
Halaman ini berfungsi sebagai tempat admin mengelola user. Untuk gambar antarmuka kelola user dapat dilihat pada gambar 4.14.



Gambar 4. 14. Perancangan Antarmuka Kelola User

### 3. Kelola Kategori

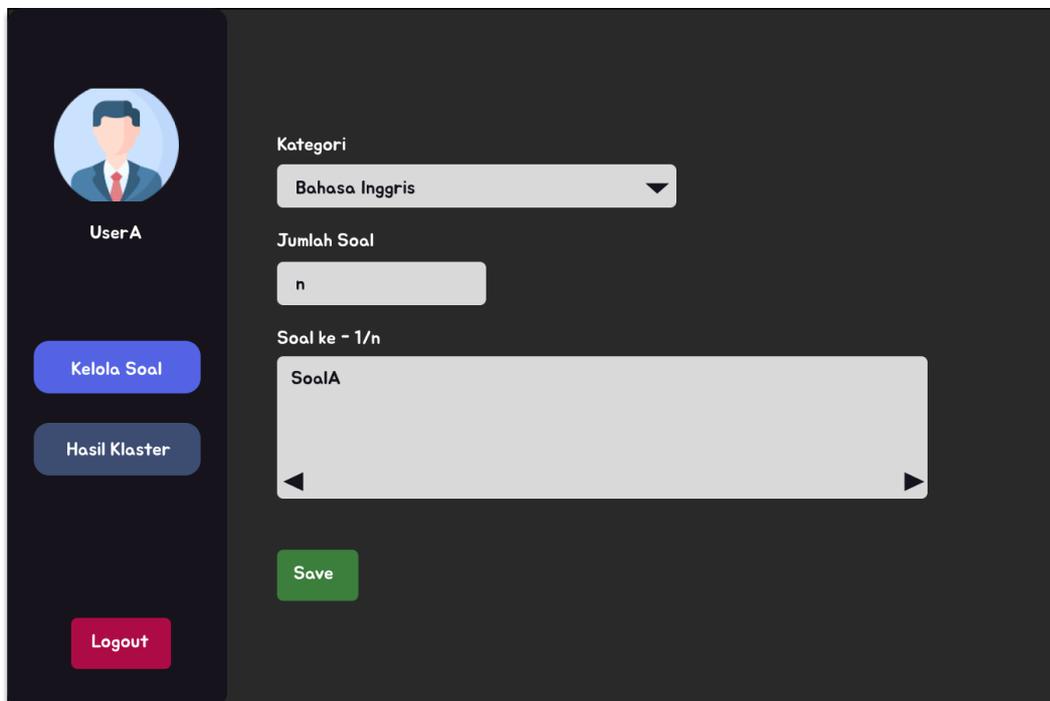
Halaman ini berfungsi sebagai tempat admin mengelola kategori. Untuk gambar antarmuka kelola kategori dapat dilihat pada gambar 4.15.



Gambar 4. 15. Perancangan Antarmuka Kelola Kategori

#### 4. Kelola Soal

Halaman ini berfungsi sebagai tempat dosen mengelola soal. Untuk gambar antarmuka kelola soal dapat dilihat pada gambar 4.16.



The screenshot displays the 'Kelola Soal' (Manage Questions) interface. On the left sidebar, there is a user profile icon labeled 'UserA', a blue button labeled 'Kelola Soal', a grey button labeled 'Hasil Kluster', and a red button labeled 'Logout'. The main content area features a 'Kategori' dropdown menu set to 'Bahasa Inggris', a 'Jumlah Soal' input field with the value 'n', a 'Soal ke - 1/n' label, and a large text area containing 'SoalA'. A green 'Save' button is located below the text area.

Gambar 4. 16. Perancangan Antarmuka Kelola Soal

#### 5. Lihat Kluster

Halaman ini berfungsi sebagai tempat dosen melihat kluster. Untuk gambar antarmuka lihat kluster dapat dilihat pada gambar 4.17.

Gambar 4. 17. Perancangan Antarmuka Lihat Kluster

## 6. Jawab Soal

Halaman ini berfungsi sebagai tempat mahasiswa menjawab soal.

Untuk gambar antarmuka jawab soal dapat dilihat pada gambar 4.18.

Gambar 4. 18. Perancangan Antarmuka Jawab Soal

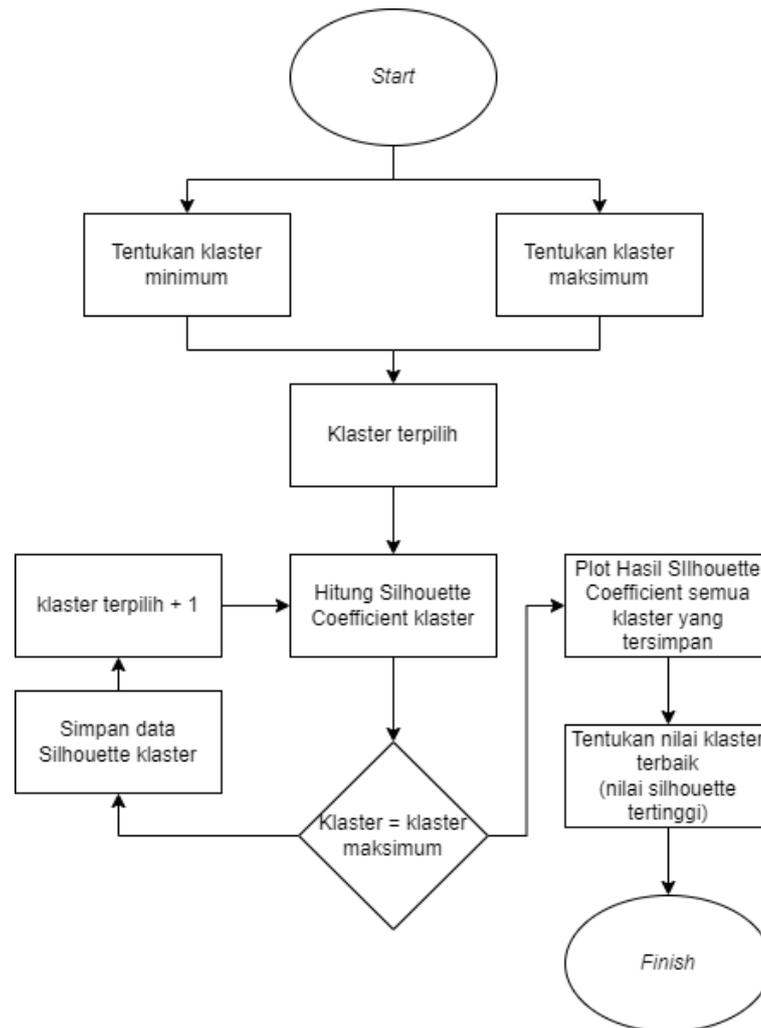
#### 4.4.5. Perancangan Pengujian

Pada Pengujian system akan dilakukan menggunakan use case scenario, klaster, dan efektifitas sedangkan untuk uji coba klaster menggunakan silhouette coefficient

##### 1. Perancangan Pengujian Klaster

Pengujian ini berfungsi untuk mengetahui bagaimana performa setiap klaster sekaligus mengetahui klaster mana yang paling optimal. Pengujian ini akan menggunakan Silhouette coefficient yang akan mengukur seberapa baik setiap data yang ada pada klaster, hasil dari silhouette berkisar antara  $-1 - 1$  dimana semakin besar nilai silhouette nya maka semakin bagus klaster tersebut.

Hasil dari *clustering* ini selanjutnya akan masuk ke proses *testing* menggunakan *silhouette coefficient*. Hasil *testing* ini dilakukan untuk mencari tau jumlah klaster terbaik yang akan direkomendasikan sistem untuk ditampilkan pada *user* sehingga *user* tidak perlu menentukan jumlah klaster. Alur algoritma *silhouette coefficient* dapat dilihat pada gambar 4.19.



Gambar 4. 19. Proses *Silhouette Coefficient*

Pada gambar diatas dilakukan *testing* menggunakan *silhouette coefficient* yang akan menghitung jarak rata-rata data antar kluster dibagi dengan jarak rata rata terdekat data dengan kluster. Proses dari algoritma *silhouette coefficient* kurang lebih sebagai berikut.

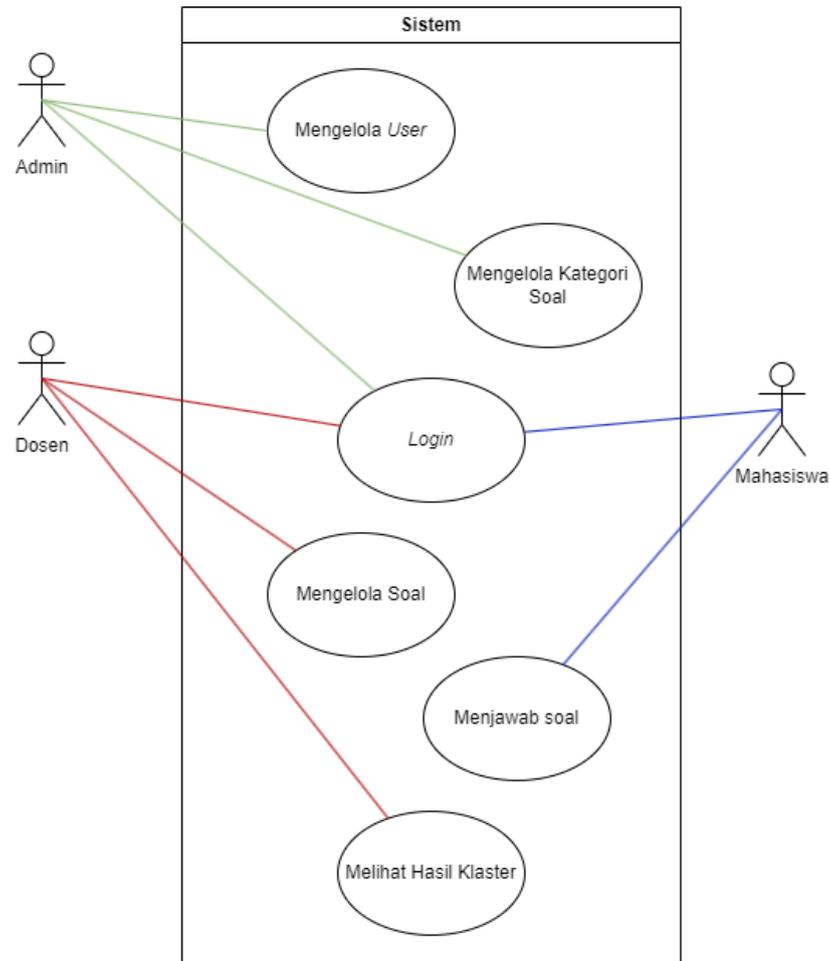
- A. Menentukan jumlah kluster minimum dan maksimum, hal ini berguna untuk proses pengulangan dimana pada proses *testing* semua jumlah kluster akan diproses.
- B. Setelah didapat jumlah kluster terpilih, sistem akan menghitung nilai *silhouette*
- C. Setelah itu akan ada kondisi yang memeriksa apakah jumlah kluster terpilih sama dengan kluster maksimum? Jika tidak maka data akan disimpan dan data jumlah kluster terpilih akan

ditambah 1, lalu dilakukan perhitungan *silhouette* kembali menggunakan data jumlah klaster yang baru. Jika kondisi terpenuhi maka dilakukan *plotting* atau pembuatan grafik dari semua data yang tersimpan

Selanjutnya berdasarkan grafik yang telah dibuat akan diambil data jumlah klaster dengan nilai *silhouette* tertinggi yang akan menjadi data klaster terbaik.

## 2. Perancangan Pengujian Sistem

Pengujian Sistem akan dilakukan menggunakan *use case scenario*. Pengujian ini akan dilakukan oleh rekan peneliti yang tidak ikut serta dalam pembuatan sistem. Disini penguji diharapkan dapat menguji sistem sesuai dengan tabel yang sudah disediakan, menambahkan catatan bila perlu atau ada proses yang gagal, dan menyatakan sukses atau gagalnya proses tersebut dengan cara menambahkan warna kuning pada tempat yang sudah disediakan. Pengujian dilakukan pada tanggal 17 Juli 2023. Untuk use case diagram dapat dilihat pada gambar 4.20.



Gambar 4. 20. Use Case Diagram

Pada use case dapat dilihat ada tiga role dan enam proses yang dapat dilakukan sistem. Terdapat role admin yang dapat login, mengelola user, dan mengelola kategori soal. Role dosen yang dapat login, mengelola soal, dan melihat hasil klaster, dan Role mahasiswa yang dapat login dan menjawab soal. Untuk skenario beserta tabel pengujian sistem dapat dilihat pada tabel 4.15 Sampai 4.21.

#### 1. Login

Tabel 4. 15. Skenario Login

Aktor: Admin, Dosen, Mahasiswa	
Deskripsi: User masuk kedalam sistem	
Pra-Kondisi: User mengetahui keperluan login (username & password)	
Proses Utama	Respon

1. Akses <a href="http://libclusterjum.my.id/">http://libclusterjum.my.id/</a>	Menampilkan halaman login	
2. Memasukkan data username dan password yang sesuai lalu klik tombol masuk	Menampilkan dasbor	
Proses Alternatif 1	Respon	
2. Memasukkan data username dan password tidak sesuai lalu klik tombol masuk	Menampilkan peringatan user tidak ditemukan	
Proses Alternatif 2	Respon	
2. Memasukkan data username tidak sesuai, dan password sesuai lalu klik tombol masuk	Menampilkan peringatan user tidak ditemukan	
Proses Alternatif 3	Respon	
2. Memasukkan data username sesuai, dan password tidak sesuai lalu klik tombol masuk	Menampilkan peringatan password salah	
Catatan	Pada Proses ke	
Hasil		
Gagal Mayor	Gagal Minor	Sukses

## 2. Kelola User

Tabel 4. 16 Skenario Kelola User

Aktor: Admin	
Deskripsi: Admin menambahkan, mengedit, dan menghapus data user	
Pra-Kondisi: Login kedalam sistem sebagai admin, terdapat data user pada database	
Proses Utama	Respon
1. Akses dasbor admin lalu klik menu kelola user pada sidebar	Menampilkan halaman kelola user

2. Klik tombol tambah user baru	Menampilkan halaman tambah user
3. Memasukkan data dengan sesuai dan memilih role lalu klik tombol tambah	Menampilkan peringatan user berhasil dibuat dan data baru muncul pada tabel
4. Kembali ke menu kelola user, pilih salah satu user lalu klik tombol edit	Menampilkan halaman edit user
5. Memasukkan data dengan sesuai lalu klik tombol edit	Menampilkan peringatan user berhasil diedit dan data muncul pada tabel
6. Kembali ke menu kelola user, pilih salah satu user lalu klik tombol delete	Data yang dihapus hilang
Proses Alternatif 1	Respon
3. Memasukkan data dengan sesuai dan tidak memilih role lalu klik tombol tambah	Menampilkan peringatan proses gagal
Proses Alternatif 2	Respon
3. Memasukkan data password dan confirm password tidak sesuai dan memilih role lalu klik tombol tambah	Menampilkan peringatan data password dan confirm password tidak sama
Proses Alternatif 3	Respon
3. Memasukkan data password dan confirm password tidak sesuai dan tidak memilih role lalu klik tombol tambah	Menampilkan peringatan data password dan confirm password tidak sama
Proses Alternatif 4	Respon
3. Tidak mengisi data lalu klik tombol tambah	Menampilkan peringatan proses gagal
Proses Alternatif 5	Respon

5. Memasukkan data password dan confirm password tidak sesuai lalu klik tombol edit	Menampilkan peringatan data password dan confirm password tidak sama	
Proses Alternatif 6	Respon	
5. Tidak mengisi data lalu klik tombol simpan	Menampilkan peringatan proses gagal	
Catatan	Pada Proses ke	
Hasil		
Gagal Mayor	Gagal Minor	Sukses

### 3. Kelola Kategori

Tabel 4. 17. Skenario Kelola Kategori

Aktor: Admin	
Deskripsi: Admin menambahkan, mengedit, dan menghapus data kategori	
Pra-Kondisi: Login kedalam sistem sebagai admin, terdapat data user dan kategori pada database	
Proses Utama	Respon
1. Akses dasbor admin lalu klik menu kelola kategori pada sidebar	Menampilkan halaman kelola kategori
2. Klik tombol tambah kategori baru	Menampilkan halaman tambah kategori
3. Memasukkan data lalu klik tombol tambah	Menampilkan peringatan kategori berhasil dibuat dan data baru muncul pada tabel
4. Kembali ke menu kelola kategori, pilih salah satu kategori lalu klik tombol edit	Menampilkan halaman edit kategori

5. Memasukkan data lalu klik tombol simpan	Menampilkan peringatan kategori berhasil diedit dan data muncul pada tabel	
6. Kembali ke menu kelola kategori, pilih salah satu kategori lalu klik tombol hapus	Data yang dihapus hilang	
Proses Alternatif 1	Respon	
3. Tidak mengisi data lalu klik tombol simpan	Menampilkan peringatan proses gagal	
Proses Alternatif 2	Respon	
5. Tidak mengisi data lalu klik tombol tambah	Menampilkan peringatan proses gagal	
Catatan	Pada Proses ke	
Hasil		
Gagal Mayor	Gagal Minor	Sukses

#### 4. Kelola Soal

Tabel 4. 18. Skenario Kelola Soal

Aktor: Dosen	
Deskripsi: Dosen menambahkan, mengedit, dan menghapus data soal	
Pra-Kondisi: Login kedalam sistem sebagai dosen, terdapat data user, kategori, dan soal pada database	
Proses Utama	Respon
1. Akses dasbor dosen lalu klik menu kelola soal pada sidebar	Menampilkan halaman kelola soal
2. Pilih kategori lalu klik tombol tambah soal baru	Menampilkan halaman tambah kategori
3. Memasukkan data lalu klik tombol tambah	Menampilkan peringatan soal berhasil dibuat dan data baru muncul pada tabel

4. Kembali ke halaman kelola soal, pilih salah satu soal lalu klik tombol edit	Menampilkan halaman edit soal	
5. Memasukkan data lalu klik tombol simpan	Menampilkan peringatan soal berhasil diedit dan data muncul pada tabel	
6. Kembali ke halaman kelola soal, pilih salah satu soal lalu klik tombol hapus	Menampilkan peringatan data berhasil dihapus dan data yang dihapus hilang	
Proses Alternatif 1	Respon	
2. Tidak pilih kategori lalu klik tombol tambah soal baru	Menampilkan peringatan harap pilih kategori	
Proses Alternatif 2	Respon	
3. Tidak mengisi data lalu klik tombol tambah	Menampilkan peringatan proses gagal	
Proses Alternatif 3	Respon	
5. Tidak mengisi data lalu klik tombol simpan	Menampilkan peringatan proses gagal	
Catatan	Pada Proses ke	
Hasil		
Gagal Mayor	Gagal Minor	Sukses

### 5. Lihat Klaster

Tabel 4. 19. Skenario Lihat Klaster

Aktor: Dosen	
Deskripsi: Dosen melihat data klaster	
Pra-Kondisi: Login kedalam sistem sebagai dosen, terdapat data user, kategori, soal, dan jawaban pada database	
Proses Utama	Respon

1. Akses dasbor dosen lalu klik menu lihat klaster pada sidebar	Menampilkan halaman lihat klaster
2. Pilih kategori	Menampilkan data yang dipilih
3. Pilih salah satu soal lalu klik tombol pilih	Menampilkan halaman klaster soal
4. Klik tombol klaster jawaban	Menampilkan tabel klaster rekomendasi sistem
Proses Alternatif 1	Respon
3. Pilih salah satu soal lalu klik tombol pilih	Menampilkan peringatan tidak ada data
Catatan	Pada Proses ke
Hasil	
Gagal Mayor	Gagal Minor
	Sukses

## 6. Jawab Soal

Tabel 4. 20. Skenario Jawab Soal

Aktor: Mahasiswa	
Deskripsi: Mahasiswa menambahkan data jawaban	
Pra-Kondisi: Login kedalam sistem sebagai mahasiswa, terdapat data user, kategori, soal, pada database	
Proses Utama	Respon
1. Akses dasbor mahasiswa lalu klik menu jawab soal pada sidebar	Menampilkan halaman pilih kategori soal
2. Pilih salah satu kategori lalu klik tombol pilih	Menampilkan halaman jawab soal
3. Mengisi data jawaban lalu klik tombol simpan	Menampilkan peringatan jawaban disimpan lalu data jawaban akan disimpan
4. Mengisi data jawaban lalu klik tombol next/previous	Menampilkan soal sebelum/sesudahnya dengan data jawaban ada pada textarea

5. Klik tombol next/previous di soal awal/akhir	Menampilkan peringatan and sudah di pertanyaan pertama/terakhir	
6. Mengisi data jawaban lalu klik tombol simpan dan klik finish/waktu habis	Menyimpan data jawaban dan menampilkan halaman pilih kategori soal	
Proses Alternatif 1	Respon	
2. Pilih salah satu kategori tanpa soal lalu klik tombol pilih	Menampilkan halaman kosong	
Proses Alternatif 2	Respon	
3. Tidak mengisi data jawaban lalu klik tombol simpan	Menampilkan peringatan gagal menyimpan jawaban	
Proses Alternatif 3	Respon	
4. Tidak mengisi data jawaban lalu klik tombol next/previous	Menampilkan soal sebelum/sesudahnya dengan textarea kosong	
Proses Alternatif 4	Respon	
6. Mengisi data jawaban tidak klik tombol simpan dan klik finish/waktu habis	Tidak menyimpan data jawaban dan menampilkan halaman pilih kategori soal	
Catatan	Pada Proses ke	
Hasil		
Gagal Mayor	Gagal Minor	Sukses

## 7. Logout

Tabel 4. 21 Skenario Logout

Aktor: Admin, Dosen, Mahasiswa	
Deskripsi: User keluar dari sistem	
Pra-Kondisi: User sedang didalam sistem	
Proses Utama	Respon
1. Akses dasbor salah satu user lalu klik tombol keluar	Menampilkan halaman login
Proses Alternatif 1	Respon
1. Akses dasbor salah satu user lalu klik tombol keluar	Menampilkan halaman dasbor

Catatan	Pada Proses ke	
Hasil		
Gagal Mayor	Gagal Minor	Sukses

### 3. Perancangan Pengujian Efektifitas

Pengujian Efektifitas sistem akan dilakukan menggunakan perbandingan hasil pengelompokkan jawaban secara manual dan pengelompokkan jawaban menggunakan sistem yang dibuat, untuk pengelompokkan secara manual dilakukan oleh salah satu dosen Jurusan Teknologi Informasi, Politeknik Negeri Malang dimana pengujiannya sendiri dilakukan pada tanggal 17 Juli 2023, berikut adalah kemungkinan yang dapat timbul dari hasil pengujian efektifitas.

1. Tidak efektif: Jika persentase  $< 25\%$  atau perbandingan durasi antara proses manual dengan sistem adalah  $> 1/1$
2. Kurang efektif: jika persentase  $> 25\%$  dan  $< 50\%$  atau perbandingan durasi antara proses manual dengan sistem adalah  $< 1/1$  dan  $> 1/2$
3. Cukup efektif: jika persentase  $> 50\%$  dan  $< 75\%$  atau perbandingan durasi antara proses manual dengan sistem adalah  $< 1/2$
4. Sangat efektif: jika persentase  $> 75\%$  dan perbandingan durasi antara proses manual dengan sistem adalah  $< 1/2$

Pada uji coba ini jumlah klaster akan ditentukan berdasarkan jumlah klaster yang memiliki nilai *silhouette coefficient* terbesar dan yang akan menjadi indikator pembanding adalah persentase ketepatan klaster dan durasi dilakukannya pengelompokkan. Untuk contoh tabel pengujian efektifitas sistem dapat dilihat pada tabel 4.22.

Tabel 4. 22. Perancangan Pengujian Efektifitas

Klaster = 3 (dimisalkan)				
Soal	Jawaban	Kelompok/ Klaster (A-C)		Hasil (Sesuai/Tidak)
		Manual	Sistem	

Sebutkan definisi komputer.	Komputer adalah perangkat elektronik yang digunakan untuk mengolah data!	A/B/C	A/B/C	Sesuai/Tidak
	Komputer merupakan alat bantu yang memproses informasi dengan cepat.	A/B/C	A/B/C	Sesuai/Tidak
	Komputer dapat menjalankan program sesuai dengan instruksi yang diberikan.	A/B/C	A/B/C	Sesuai/Tidak
	Komputer memiliki kemampuan untuk menyimpan dan mengambil data.	A/B/C	A/B/C	Sesuai/Tidak
	Komputer terdiri dari beberapa komponen seperti CPU, RAM, dan hard disk!	A/B/C	A/B/C	Sesuai/Tidak
Durasi		menit/	menit	
Persentase				%