

BAB 3

ANALISIS DAN PERANCANGAN SISTEM

3.1 Analisis Sistem

Analisis mengenai pengelompokan aktivitas pengembangan sistem informasi inventori dilakukan untuk mendapatkan kelompok aktivitas pengembangan sistem informasi inventori yang optimal dan efektif. Aktivitas pengembangan merupakan rencana pengurutan kerja untuk menyelesaikan suatu pekerjaan dengan sasaran khusus dengan saat penyelesaian yang jelas (Ardan & Anisa Putri, 2021). Dalam membuat jadwal proyek pengembangan sistem informasi inventori terdapat urutan-urutan yang harus dilakukan untuk mendapatkan jadwal yang efisien, optimal dan efektif. Berikut adalah urutan-urutan yang harus dilakukan:

1. Analisis kebutuhan: analisis kebutuhan dilakukan untuk memahami kebutuhan pengguna. Kebutuhan pengguna meliputi desain sistem, fitur-fitur yang akan diimplementasikan kedalam sistem, dan lain-lain. Dalam menentukan fitur-fitur yang akan digunakan ketika melakukan pengembangan sistem, dibutuhkan pengamatan fitur dari beberapa aplikasi yang telah ada atau jurnal penelitian terdahulu. Fitur-fitur tersebut akan dimasukkan kedalam daftar fitur yang nanti akan diimplementasikan pada sistem informasi inventori.
2. Penentuan prioritas fitur dan beban kerja fitur: penentuan prioritas fitur dan beban kerja fitur dilakukan untuk menentukan fitur yang harus dikerjakan terlebih dahulu karena fitur tersebut memiliki peran penting dalam sistem informasi inventori. Dalam menentukan prioritas dan beban kerja fitur, dibutuhkan daftar fitur-fitur yang akan digunakan dan model algoritma untuk membantu pengembang dalam menentukan prioritas dan beban kerja dari suatu fitur. Dari prioritas fitur dan beban kerja fitur akan diubah menjadi prioritas dan beban kerja aktivitas pengembangan fitur. Terdapat beberapa model yang dapat digunakan diantaranya adalah metode Moscow Prioritization.

3. Penentuan durasi proyek pengembangan sistem informasi inventori: pada tahapan ini menentukan durasi pengembangan sistem informasi inventori berdasarkan durasi implementasi fitur atau beban kerja dari fitur yang akan diimplementasikan. Dalam menentukan durasi proyek, dibutuhkan kelompok aktivitas pengembangan yang terdapat prioritas dan beban kerja pengembangan fitur untuk membantu tim pengembang dalam mengalokasikan waktu dan sumber daya secara efektif, dan memastikan proyek dapat selesai tepat waktu tanpa adanya penundaan aktivitas yang berlebihan.
4. Perancangan jadwal proyek pengembangan sistem informasi inventori: pada tahapan ini merupakan langkah penting dalam memastikan kelancaran dan efektivitas proyek dengan melibatkan visualisasi urutan tugas, dan durasi masing-masing tugas berdasarkan hasil tahapan analisis, penentuan prioritas dan beban kerja fitur sebelumnya. Dalam perancangan jadwal proyek pengembangan sistem informasi inventori dibutuhkan susunan atau daftar dari prioritas fitur beserta beban kerja dari masing-masing fitur yang didapatkan dari tahapan sebelumnya.

Permasalahan yang sering dihadapi dalam pembuatan jadwal proyek sistem informasi inventori yang efektif adalah ketidakjelasan dalam menentukan prioritas dan beban kerja aktivitas pengembangan yang akan diimplementasikan. Dalam menentukan aktivitas pengembangan sistem informasi inventori, dibutuhkan pengamatan fitur pada beberapa aplikasi yang sudah publik atau pengamatan fitur pada jurnal penelitian terdahulu. Setelah melakukan pengamatan fitur, selanjutnya adalah menyusun daftar prioritas aktivitas pengembangan fitur dan beban kerja aktivitas pengembangan fitur untuk mengetahui tingkat urgensi dari masing-masing fitur sehingga tim pengembang dapat mengetahui fitur yang harus diimplementasikan terlebih dahulu.

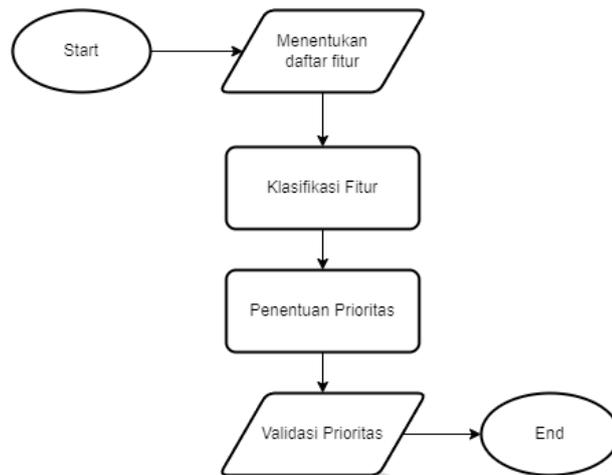
Sesuai dengan analisis sistem yang telah dilakukan, terdapat permasalahan dalam perancangan jadwal proyek sistem informasi inventori yaitu tidak adanya kelompok aktivitas pengembangan sistem informasi inventori. Permasalahan

tersebut dapat diselesaikan dengan menggunakan sistem pengelompokan aktivitas pengembangan sistem informasi inventori.

3.2 Hasil Analisis Sistem

Ketidajelasan atau tidak adanya proses dalam menentukan kelompok aktivitas pengembangan sistem informasi inventori sering kali menjadi hambatan dalam perancangan jadwal proyek pengembangan sistem informasi inventori. Hal ini dapat mengakibatkan penundaan aktivitas, dan kurangnya efisien dalam penggunaan sumber daya. Untuk mengatasi permasalahan tersebut, sistem pengelompokan aktivitas pengembangan sistem informasi inventori dikembangkan. Sistem pengelompokan aktivitas pengembangan sistem informasi inventori dikembangkan untuk mengatasi permasalahan dalam menentukan prioritas dan beban kerja dari aktivitas pengembangan yang akan digunakan kedalam perancangan jadwal proyek sistem informasi inventori. Sistem tersebut akan menghasilkan kelompok prioritas aktivitas pengembangan sistem informasi inventori yang terdiri dari prioritas Tinggi, Sedang, dan Rendah yang sudah terdapat beban kerja atau durasi pengembangan dari masing-masing fitur, dan kelompok prioritas tersebut dapat diimplementasikan kedalam perancangan jadwal proyek pengembangan sistem informasi inventori.

Pengembangan sistem pengelompokan fitur jadwal proyek sistem informasi inventori dikembangkan dengan menggunakan metode penentuan prioritas moscow, uji korelasi pearson, dan algoritma k-means clustering. Metode Moscow (*Must have, Should have, Could have, Wont have*) digunakan untuk menentukan prioritas fitur pada perancangan jadwal proyek sistem informasi inventori karena metode ini mudah dan cepat dalam mengurutkan daftar kebutuhan atau prioritas serta mampu menghasilkan urutan yang akurat (Marthasari & Hayatin, 2020). Berikut adalah tahapan dalam menggunakan metode moscow untuk menentukan prioritas fitur sistem informasi inventori:



Gambar 3.1 *Flowchart* Metode Moscow

1. Menentukan daftar fitur:

Membuat daftar fitur yang akan diimplementasikan ke dalam sistem. Fitur yang akan diimplementasikan berdasarkan referensi pada aplikasi yang sudah ada dan penelitian terdahulu. Daftar fitur terdapat pada tabel fitur dari sub 2.3 Fitur Aplikasi Inventori

Tabel Fitur dari sub 2.3 Fitur Aplikasi Inventori

No	Fitur
1	Tambah barang
2	Edit Barang
3	Hapus Barang
4	List Barang
5	Tambah Penjualan
6	Edit Penjualan
7	Hapus Penjualan
8	List Penjualan
9	Tambah Pembelian
10	Edit Pembelian
11	Hapus Pembelian
12	List Pembelian
13	Tambah Vendor

No	Fitur
14	Edit Vendor
15	Hapus Vendor
16	List Vendor
17	Hubungi Vendor
18	Hubungi Owner

2. Klasifikasi fitur:

Mengelompokkan fitur berdasarkan kategori moscow, yaitu Must Have, Should Have, Could Have dan Won't Have. Pengelompokan fitur dilakukan dengan menyebarkan kuesioner kepada pengguna.

Tabel 3.1 Klasifikasi fitur

No	Fitur	Klasifikasi
1	Tambah barang	<i>Must Have</i>
2	Edit Barang	<i>Must Have</i>
3	Hapus Barang	<i>Must Have</i>
4	List Barang	<i>Must Have</i>
5	Tambah Penjualan	<i>Must Have</i>
6	Edit Penjualan	<i>Should Have</i>
7	Hapus Penjualan	<i>Wont Have</i>
8	List Penjualan	<i>Must Have</i>
9	Tambah Pembelian	<i>Must Have</i>
10	Edit Pembelian	<i>Should Have</i>
11	Hapus Pembelian	<i>Should Have</i>
12	List Pembelian	<i>Must Have</i>
13	Tambah Vendor	<i>Should Have</i>
14	Edit Vendor	<i>Should Have</i>
15	Hapus Vendor	<i>Should Have</i>
16	List Vendor	<i>Should Have</i>
17	Hubungi Vendor	<i>Could Have</i>

No	Fitur	Klasifikasi
18	Hubungi Owner	<i>Could Have</i>

3. Penentuan prioritas

Mengurutkan fitur dalam setiap kategori berdasarkan kebutuhan pengguna dan urgensinya.

Tabel 3.2 Penentuan prioritas fitur

No	Fitur	Klasifikasi
1	Tambah barang	<i>Must Have</i>
2	Edit Barang	<i>Must Have</i>
3	Hapus Barang	<i>Must Have</i>
4	List Barang	<i>Must Have</i>
5	Tambah Penjualan	<i>Must Have</i>
6	List Penjualan	<i>Must Have</i>
7	Tambah Pembelian	<i>Must Have</i>
8	List Pembelian	<i>Must Have</i>
9	Edit Penjualan	<i>Should Have</i>
10	Edit Pembelian	<i>Should Have</i>
11	Hapus Pembelian	<i>Should Have</i>
12	Tambah Vendor	<i>Should Have</i>
13	Edit Vendor	<i>Should Have</i>
14	Hapus Vendor	<i>Should Have</i>
15	List Vendor	<i>Should Have</i>
16	Hubungi Vendor	<i>Could Have</i>
17	Hubungi Owner	<i>Could Have</i>
18	Hapus Penjualan	<i>Wont Have</i>

4. Validasi prioritas

Memvalidasi prioritas fitur berdasarkan suara terbanyak dari hasil kuesioner. Pada tahapan ini menentukan fitur berdasarkan hasil kuesioner

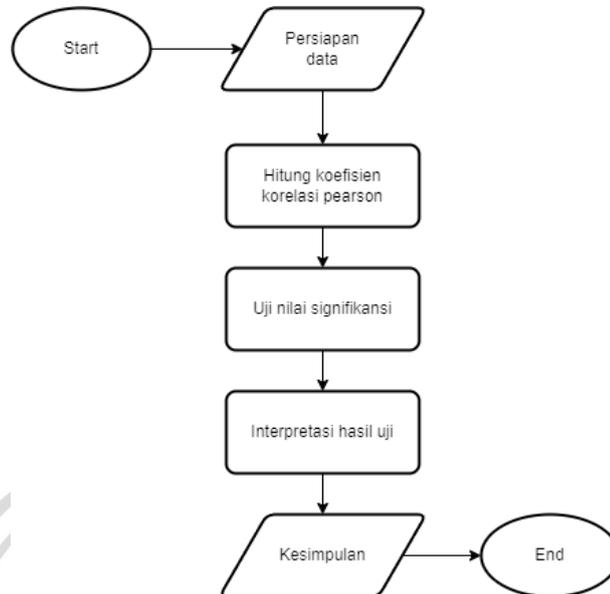
terbanyak. Seperti pada Tabel 3.3 terdapat 30 suara pada prioritas *Must*, 0 suara di prioritas *Should*, *Could*, dan *Wont* di fitur Tambah barang, maka fitur Tambah barang memiliki prioritas fitur *Must have*.

Tabel 3.3 Validasi prioritas fitur

No	Fitur	Klasifikasi	<i>Must</i>	<i>Should</i>	<i>Could</i>	<i>Wont</i>
1	Tambah barang	<i>Must Have</i>	30	0	0	0
2	Edit Barang	<i>Must Have</i>	17	7	3	3
3	Hapus Barang	<i>Must Have</i>	16	3	8	3
4	List Barang	<i>Must Have</i>	30	0	0	0
5	Tambah Penjualan	<i>Must Have</i>	30	0	0	0
6	List Penjualan	<i>Must Have</i>	30	0	0	0
7	Tambah Pembelian	<i>Must Have</i>	25	3	2	0
8	List Pembelian	<i>Must Have</i>	22	4	1	3
9	Edit Penjualan	<i>Should Have</i>	3	20	5	2
10	Edit Pembelian	<i>Should Have</i>	5	17	5	3
11	Hapus Pembelian	<i>Should Have</i>	2	3	8	17
12	Tambah Vendor	<i>Should Have</i>	5	5	16	4
13	Edit Vendor	<i>Should Have</i>	4	5	17	4
14	Hapus Vendor	<i>Should Have</i>	6	2	17	5
15	List Vendor	<i>Should Have</i>	7	2	20	1
16	Hubungi Vendor	<i>Could Have</i>	5	6	0	19
17	Hubungi Owner	<i>Could Have</i>	20	3	0	7
18	Hapus Penjualan	<i>Wont Have</i>	6	0	22	2

Uji korelasi pearson digunakan untuk mengetahui hubungan antar variabel data hasil kuesioner metode moscow dengan variabel data pengamatan pada aplikasi yang telah publik, dan melakukan eliminasi pada hasil uji korelasi pearson dengan nilai koefisien terendah untuk mengetahui kelayakan data yang akan digunakan ke proses selanjutnya yaitu penentuan fitur dan beban kerja fitur

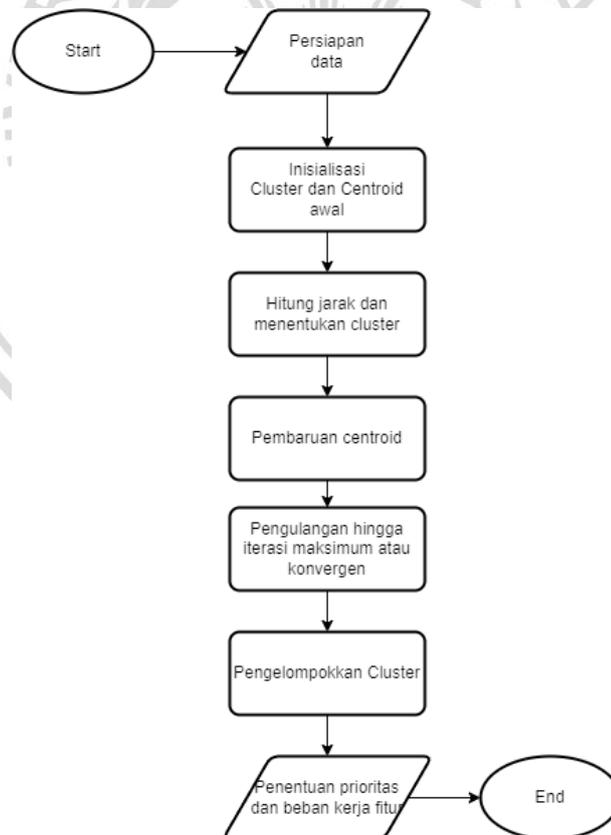
menggunakan algoritma k-means clustering. Diagram alur atau *flowchart* uji korelasi pearson terdapat pada Gambar 3.2.



Gambar 3.2 *Flowchart* Uji Korelasi Pearson

1. Persiapan data
Mempersiapkan data yang akan digunakan, yaitu data pengamatan pada beberapa aplikasi yang telah publik dan variabel dari metode moscow (*must have, should have, could have, wont have*).
2. Hitung koefisien korelasi pearson
Menghitung nilai koefisien korelasi pearson secara manual atau dengan bantuan software SPSS.
3. Uji nilai signifikansi
Melakukan uji nilai signifikansi dengan membuat hipotesa nol dan hipotesa alternatif.
4. Interpretasi hasil
Menginterpretasikan nilai koefisien korelasi pearson yang berkisar antara -1 dan 1.
5. Kesimpulan
Membuat kesimpulan dari hasil interpretasi koefisien korelasi pearson dan melakukan eliminasi pada variabel moscow yang memiliki nilai interpretasi terendah.

Algoritma k-means *clustering* digunakan untuk menentukan prioritas tinggi, sedang dan rendahnya aktivitas pengembangan sistem informasi inventori dengan mengimplementasikan beban kerja aktivitas pengembangan pada masing-masing prioritas aktivitas pengembangan. Data yang digunakan untuk membuat rancangan jadwal proyek sistem informasi inventori didapatkan dari hasil uji korelasi pearson dengan *centroid* sebanyak 3 sesuai dengan prioritas fitur yang akan diterapkan kedalam jadwal proyek. Data yang digunakan untuk mendapatkan kelompok aktivitas pengembangan sistem informasi inventori didapatkan dari hasil uji korelasi pearson yang sudah melakukan eliminasi pada variabel moscow dengan nilai korelasi terendah dan prioritas yang akan diterapkan dalam kelompok aktivitas pengembangan sistem informasi inventori sebanyak 3 prioritas, yaitu tinggi, sedang, dan rendah. Diagram alur atau *flowchart* dari penerapan k-means *clustering* untuk mendapatkan kelompok aktivitas pengembangan sistem informasi inventori terdapat pada Gambar 3.3.



Gambar 3.3 *Flowchart* K-Means *Clustering*

1. Persiapan data

Mempersiapkan data yang akan digunakan, yaitu hasil dari uji korelasi pearson yaitu pengamatan, dan metode moscow yang tidak tereliminasi.

2. Inisialisasi *cluster* dan *centroid* awal

Menentukan jumlah *cluster* sesuai dengan jumlah prioritas yang akan digunakan sebagai rancangan pengelompokan aktivitas pengembangan sistem informasi inventori yaitu sebanyak 3, dan menentukan nilai *centroid* awal secara acak.

3. Hitung jarak dan menentukan *cluster*

Menghitung jarak setiap titik data ke semua *centroid* dan menghitung rata-rata dari semua titik data dalam setiap *cluster* untuk menentukan *cluster* baru.

4. Pengulangan hingga iterasi maksimum atau konvergen

Mengulangi proses menghitung jarak dan menentukan *cluster* hingga *centroid* tidak berubah lagi atau konvergen dengan perubahan *centroid* di bawah ambang batas tertentu.

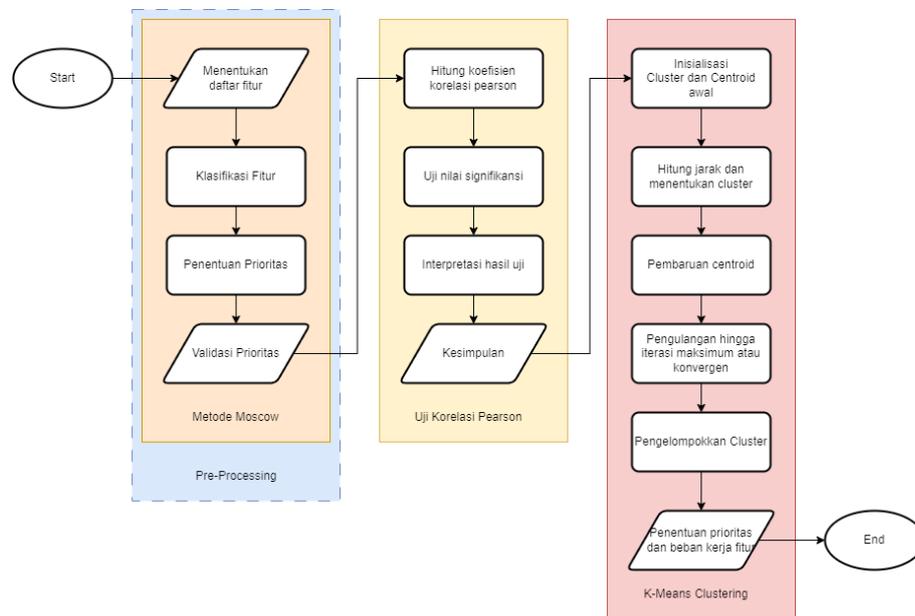
5. Pengelompokan *cluster*

Melakukan pengelompokan *cluster* dari masing-masing fitur untuk tahap selanjutnya.

6. Prioritas dan beban kerja aktivitas pengembangan

Menentukan prioritas tinggi, sedang, atau rendahnya aktivitas pengembangan sistem informasi inventori dengan beban kerja sesuai dengan *deadline* proyek sistem informasi inventori,

Flowchart untuk keseluruhan dari pengembangan sistem pengelompokan aktivitas pengembangan sistem informasi inventori dengan menggunakan 3 metode yang digunakan adalah sebagai berikut:



Gambar 3.4 Flowchart pengembangan sistem pengelompokkan aktivitas pengembangan sistem informasi inventori

Data yang akan digunakan didapatkan dari tahapan *pre-processing* yaitu setelah melakukan menentukan prioritas aktivitas pengembangan sistem informasi inventori dan telah divalidasi dengan kuesioner. Data validasi prioritas digunakan ke dalam perhitungan korelasi pearson untuk mendapatkan nilai signifikan dari metode Moscow dan Pengamatan fitur pada jurnal atau penelitian terdahulu. Pada tahapan kesimpulan pada uji korelasi pearson, melakukan eliminasi prioritas dari salah satu metode moscow untuk mendapatkan tingkat presisi atau akurasi yang akurat dalam proses selanjutnya yaitu pengelompokkan aktivitas pengembangan sistem informasi inventori menggunakan *k-means clustering*. Setelah membuat kesimpulan prioritas aktivitas pengembangan sistem informai inventori yang dieliminasi, selanjutnya adalah melakukan pengelompokkan aktivitas pengembangan sistem informai inventori dengan beban kerja menggunakan *k-means clustering*. Data yang digunakan pada tahapan *clustering* didapatkan dari hasil kesimpulan uji korelasi pearson, dalam hasil uji korelasi pearson terdapat beberapa variabel yang akan digunakan yaitu variabel Pengamatan, dan variabel metode Moscow yang telah dieliminasi sesuai dengan nilai signifikansi terendah.

Hasil yang didapatkan dari tahapan ini adalah Prioritas aktivitas pengembangan yang sudah memiliki beban kerja, yaitu tinggi, sedang, dan rendah.

3.3 Representasi Model

Uji korelasi pearson digunakan untuk mengetahui hubungan antar variabel yang akan digunakan dalam menentukan prioritas dan beban kerja aktivitas pengembangan sistem informasi inventori. Berikut adalah tahapan uji korelasi pearson dalam menentukan prioritas dan beban kerja fitur sistem informasi inventori:

1. Persiapan data

Data yang akan digunakan pada uji korelasi pearson adalah data pengamatan pada jurnal penelitian terdahulu yang telah dilakukan pada bab 2.3 Fitur-fitur aplikasi inventori, dan data hasil validasi prioritas metode moscow dari tabel 3 Validasi Prioritas Fitur. Jurnal-jurnal penelitian terdahulu antara lain adalah Jurnal Perancangan dan Pembuatan Website Inventori Barang Dengan Metode Waterfall (Studi Kasus Sakti Brem), Jurnal Perancangan Sistem Informasi Manajemen Gudang PT. X. dan Jurnal Rancang Bangun Sistem Informasi Inventori Bahan Bangunan Berbasis Web Pada PB Adi Jaya.

Tabel 3.4 hasil dari pengamatan pada fitur sistem informasi inventori.

No	Fitur	Pengamatan Fitur Pada Penelitian Terdahulu
1	Tambah barang	3
2	Edit Barang	3
3	Hapus Barang	3
4	List Barang	3
5	Tambah Penjualan	3
6	List Penjualan	3
7	Tambah Pembelian	3
8	List Pembelian	3
9	Edit Penjualan	2
10	Edit Pembelian	2

No	Fitur	Pengamatan Fitur Pada Penelitian Terdahulu
11	Hapus Pembelian	1
12	Tambah Vendor	1
13	Edit Vendor	1
14	Hapus Vendor	1
15	List Vendor	1
16	Hubungi Vendor	1
17	Hubungi Owner	1
18	Hapus Penjualan	2

Tabel 3.5 hasil validasi prioritas metode moscow

No	Fitur	Klasifikasi	<i>Must</i>	<i>Should</i>	<i>Could</i>	<i>Wont</i>
1	Tambah barang	<i>Must Have</i>	30	0	0	0
2	Edit Barang	<i>Must Have</i>	17	7	3	3
3	Hapus Barang	<i>Must Have</i>	16	3	8	3
4	List Barang	<i>Must Have</i>	30	0	0	0
5	Tambah Penjualan	<i>Must Have</i>	30	0	0	0
6	List Penjualan	<i>Must Have</i>	30	0	0	0
7	Tambah Pembelian	<i>Must Have</i>	25	3	2	0
8	List Pembelian	<i>Must Have</i>	22	4	1	3
9	Edit Penjualan	<i>Should Have</i>	3	20	5	2
10	Edit Pembelian	<i>Should Have</i>	5	17	5	3
11	Hapus Pembelian	<i>Should Have</i>	2	3	8	17
12	Tambah Vendor	<i>Should Have</i>	5	5	16	4
13	Edit Vendor	<i>Should Have</i>	4	5	17	4
14	Hapus Vendor	<i>Should Have</i>	6	2	17	5
15	List Vendor	<i>Should Have</i>	7	2	20	0
16	Hubungi Vendor	<i>Could Have</i>	5	6	0	19
17	Hubungi Owner	<i>Could Have</i>	20	3	0	7
18	Hapus Penjualan	<i>Wont Have</i>	6	0	22	2

Tabel 3.6 gabungan tabel hasil Pengamatan 3.4 dan hasil validasi metode moscow 3.5.

No	Fitur	Pengamatan	<i>Must</i>	<i>Should</i>	<i>Could</i>	<i>Wont</i>
1	Tambah barang	3	30	0	0	0
2	Edit Barang	3	17	7	3	3
3	Hapus Barang	3	16	3	8	3
4	List Barang	3	30	0	0	0
5	Tambah Penjualan	3	30	0	0	0
6	List Penjualan	3	30	0	0	0
7	Tambah Pembelian	3	25	3	2	0
8	List Pembelian	3	22	4	1	3
9	Edit Penjualan	2	3	20	5	2
10	Edit Pembelian	2	5	17	5	3
11	Hapus Pembelian	1	2	3	8	17
12	Tambah Vendor	1	5	5	16	4
13	Edit Vendor	1	4	5	17	4
14	Hapus Vendor	1	6	2	17	5
15	List Vendor	1	7	2	20	0
16	Hubungi Vendor	1	5	6	0	19
17	Hubungi Owner	1	20	3	0	7
18	Hapus Penjualan	2	6	0	22	2

2. Hitung koefisien korelasi

Menghitung nilai koefisien korelasi dari masing-masing metode moscow (Must have, Should have, Could have, dan Wont have) yang terdapat pada tabel 3.5 dengan hasil pengamatan pada tabel 3.4. Proses perhitungan koefisien korelasi dilakukan dengan menggunakan aplikasi SPSS untuk mendapatkan hasil yang akurat dan mempercepat proses perhitungan

koefisien korelasi pearson. Hasil dari perhitungan koefisien korelasi menggunakan aplikasi SPSS adalah sebagai berikut:

Tabel 3.7 Uji Korelasi Pearson

	Pengamatan	<i>Must Have</i>	<i>Should Have</i>	<i>Could Have</i>	<i>Won't Have</i>
Pengamatan Pearson	1	.782**	-.151	-.564*	-.583*
<i>Correlation</i>					
Sig. (2-tailed)		.000	.549	.015	.011
N	18	18	18	18	18

***. Correlation is significant at the 0.01 level*

(2-tailed).

**. Correlation is significant at the 0.05 level (2-*

tailed).

3. Uji nilai signifikansi

Pada tabel uji korelasi pearson terdapat kolom Sig, kolom tersebut merupakan nilai signifikansi dari masing-masing variabel moscow. Pada tahap ini menentukan apakah terdapat hubungan korelasi antara dua variabel secara statistik signifikan. Cara menentukannya adalah dengan membuat hipotesa nol (H0) dan hipotesa alternatif (H1). Berikut adalah penjelasannya:

a. Hipotesa nol (H0):

Tidak ada korelasi antara variabel X dan Y atau nilai signifikan sama dengan nol.

b. Hipotesa alternatif (H1):

Terdapat korelasi antara variabel X dan Y atau nilai signifikan tidak sama dengan nol.

Pada uji nilai signifikansi diketahui bahwa variabel X (Pengamatan) dengan variabel Y (Metode Moscow) masing-masing memiliki nilai signifikansi sendiri sehingga cara menentukan hipotesanya adalah dengan

melakukan perbandingan dengan nilai signifikansi standar yaitu 0,05 atau 5%. Berikut adalah hasil uji nilai signifikansi dari masing-masing variabel.

a. Variabel Pengamatan dan Variabel MustHave:

Pada tabel uji korelasi pearson, nilai signifikansi antara variabel pengamatan dengan variabel musthave adalah 0,00, sehingga menolak hasil hipotesa nol (H_0) karena terdapat korelasi antara kedua variabel tersebut karena nilai signifikansi nya kurang dari 0,05 atau 5%.

b. Variabel Pengamatan dan Variabel ShouldHave:

Pada tabel uji korelasi pearson, nilai signifikansi antara variabel pengamatan dengan variabel shouldhave adalah 0,549, sehingga menerima hasil hipotesa nol (H_0) karena tidak ada korelasi antara kedua variabel tersebut karena nilai signifikansi nya lebih dari 0,05 atau 5%.

c. Variabel Pengamatan dan Variabel CouldHave:

Pada tabel uji korelasi pearson, nilai signifikansi antara variabel pengamatan dengan variabel musthave adalah 0,15, sehingga menolak hasil hipotesa nol (H_0) karena terdapat korelasi antara kedua variabel tersebut karena nilai signifikansi nya kurang dari 0,05 atau 5%.

d. Variabel Pengamatan dan Variabel WontHave:

Pada tabel uji korelasi pearson, nilai signifikansi antara variabel pengamatan dengan variabel musthave adalah 0,011, sehingga menolak hasil hipotesa nol (H_0) karena terdapat korelasi antara kedua variabel tersebut karena nilai signifikansi nya kurang dari 0,05 atau 5%.

4. Interpretasi hasil uji

Pada tahap ini melakukan perbandingan nilai koefisien korelasi dengan interpretasi kekuatan korelasi. Tujuan dari tahapan ini adalah untuk menilai nilai signifikansi yang telah dilakukan sebelumnya, sehingga dapat mengetahui bahwa hasil dari uji nilai signifikansi tidak terjadi secara kebetulan. Berikut adalah cara menentukan interpretasi kekuatan korelasi:

a. Membuat tabel interpretasi korelasi:

Tabel interpretasi hasil uji korelasi adalah alat bantu yang digunakan untuk menginterpretasikan nilai koefisien korelasi Pearson.

Berikut adalah tabel dari interpretasi hasil uji korelasi pearson

Tabel 3.8 Interpretasi hasil uji

No	Korelasi Pearson	Interpretasi
1	0,00 s/d 0,20	Tidak ada
2	0,21 s/d 0,40	Lemah
3	0,41 s/d 0,60	Sedang
4	0,61 s/d 0,80	Kuat
5	0,81 s/d 1,00	Sempurna

b. Melakukan interpretasi hasil uji:

Langkah selanjutnya adalah membandingkan masing-masing nilai koefisien dari variabel moscow.

- i. Nilai Koefisien Variabel Pengamatan dan Variabel Must Have: 0,782, maka interpretasi nya adalah Kuat dengan nilai Positif.
- ii. Nilai Koefisien Variabel Pengamatan dan Variabel Should Have: -0,151, maka interpretasinya adalah Tidak ada dengan nilai Negatif.
- iii. Nilai Koefisien Variabel Pengamatan dan Variabel Could Have: -0,564, maka interpretasinya adalah Sedang dengan nilai Negatif.
- iv. Nilai Koefisien Variabel Pengamatan dan Variabel Won't Have: -0,583, maka interpretasinya adalah Sedang dengan nilai Negatif.

5. Kesimpulan

Pada tahapan ini mendapatkan hasil uji korelasi pearson yang telah dilakukan dan mengeliminasi salah satu variabel moscow dengan nilai korelasi terendah. Berikut adalah kesimpulan dari uji korelasi pearson yang telah dilakukan, "Berdasarkan hasil uji korelasi pearson dengan menggunakan variabel Pengamatan sebagai X dan variabel metode Moscow

sebagai Y, maka dapat disimpulkan bahwa nilai koefisien dari uji variabel Pengamatan dan variabel Should Have tidak akan digunakan atau dieliminasi untuk tahap selanjutnya karena memiliki nilai interpretasi dan nilai korelasi terendah diantara metode Moscow lain”. Sehingga variabel data yang akan digunakan pada tahap selanjutnya adalah variabel Pengamatan, variabel Must Have, variabel Could Have, dan variabel Won’t Have.

Algoritma K-Means Clustering digunakan untuk menentukan prioritas dan beban kerja fitur pada jadwal proyek pengembangan sistem informasi inventori. Pada tahapan ini data yang digunakan adalah data setelah melewati uji korelasi pearson, lalu kemudian akan di kelompokkan sesuai dengan prioritas Tinggi, Sedang, dan Rendah yang sudah termasuk beban kerja masing-masing fitur, seperti prioritas Tinggi memiliki beban kerja atau durasi pengerjaan 7 hari, prioritas Sedang memiliki beban kerja 5 hari, dan prioritas Rendah memiliki beban kerja 3 hari. Berikut adalah penerapan algoritma k-means clustering untuk menentukan prioritas dan beban kerja fitur:

1. Persiapan data

Data yang digunakan pada tahap ini adalah data hasil dari uji korelasi pearson. Data yang akan digunakan terdapat pada Tabel 3.9:

Tabel 3.9 Data hasil uji korelasi pearson

No	Fitur	Pengamatan	<i>Must</i>	<i>Could</i>	<i>Wont</i>
1	Tambah barang	3	30	0	0
2	Edit Barang	3	17	3	3
3	Hapus Barang	3	16	8	3
4	List Barang	3	30	0	0
5	Tambah Penjualan	3	30	0	0
6	List Penjualan	3	30	0	0
7	Tambah Pembelian	3	25	2	0
8	List Pembelian	3	22	1	3
9	Hubungi Vendor	1	5	0	19
10	Hubungi Owner	1	20	0	7

No	Fitur	Pengamatan	<i>Must</i>	<i>Could</i>	<i>Wont</i>
11	Hapus Penjualan	2	6	22	2

2. Inisialisasi cluster dan centroid awal

Langkah selanjutnya adalah inisialisasi *cluster* dan *centroid* awal. Karena dalam pengelompokan aktivitas pengembangan sistem informasi inventori dibutuhkan 3 (tiga) buah prioritas, maka menggunakan 3 (tiga) *cluster* dengan nilai titik awal pada data ke-3, ke-7, dan ke-9, nilai ambang batas atau T sebesar 0,1, dan nilai awal sebesar 1000.

Tabel 3.10 Inisialisasi *cluster* dan *centroid*

No	Fitur	Pengamatan	<i>Must</i>	<i>Could</i>	<i>Wont</i>	<i>Centroid</i>
1	Tambah barang	3	30	0	0	
2	Edit Barang	3	17	3	3	
3	Hapus Barang	3	16	8	3	<i>Centroid 1</i>
4	List Barang	3	30	0	0	
5	Tambah Penjualan	3	30	0	0	
6	List Penjualan	3	30	0	0	
7	Tambah Pembelian	3	25	2	0	Centroid 2
8	List Pembelian	3	22	1	3	
9	Hubungi Vendor	1	5	0	19	Centroid 3
10	Hubungi Owner	1	20	0	7	
11	Hapus Penjualan	2	6	22	2	

Tabel 3.11 Titik *centroid* awal

Data ke	Centroid	Pengamatan	<i>Must</i>	<i>Could</i>	<i>Wont</i>
3	K1	3	16	8	3
7	K2	3	25	2	0
9	K3	1	5	0	9

3. Menghitung jarak dan pembaruan *centroid*

Langkah selanjutnya setelah inisialisasi *cluster* dan *centroid* awal adalah menghitung jarak masing-masing titik ke *centroid* terdekat (K) dan memperbarui nilai *centroid* baru untuk digunakan ke iterasi selanjutnya jika masih terdapat perubahan *cluster* pada data. Menghitung jarak masing-masing titik ke *centroid* terdekat dengan menggunakan rumus euclidean *distance* (Pers 2.1). Berikut adalah hasil dari menghitung jarak dengan menggunakan rumus euclidean *distance* yang terdapat pada persamaan 2.1.

Tabel 3.12 jarak ke *centorid* terdekat

Data ke-	Jarak titik ke <i>centroid</i>			Min	Cluster
	K1	K2	K3		
1	16,40	5,39	26,65	5,39	2
2	5,10	8,60	13,89	5,10	1
3	0,00	11,22	15,00	0,00	1
4	16,40	5,39	26,65	5,39	2
5	16,40	5,39	26,65	5,39	2
6	16,40	5,39	26,65	5,39	2
7	11,22	0,00	22,11	0,00	2
8	9,22	4,36	18,17	4,36	2
9	21,10	27,73	10,00	10,00	3
10	10,00	9,06	15,13	9,06	2
11	17,26	27,68	23,13	17,26	1

Selanjutnya melakukan perhitungan centroid yang baru untuk setiap cluster berdasarkan data yang bergabung pada setiap clusternya.

Tabel 3.13 Cluster 1

Data Ke-	Pengamatan	Must	Could	Won't
2	3	17	3	3
3	3	16	8	3
11	2	6	22	2

Data Ke-	Pengamatan	Must	Could	Won't
Rata-rata	2,67	13,00	11,00	2,67

Tabel 3.14 *Cluster 2*

Data Ke-	Pengamatan	Must	Could	Won't
1	3	30	0	0
4	3	30	0	0
5	3	30	0	0
6	3	30	0	0
7	3	25	2	0
8	3	22	1	3
10	1	20	0	7
Rata-rata	2,71	26,71	0,43	1,43

Tabel 3.15 *Cluster 3*

Data Ke-	Pengamatan	Must	Could	Won't
9	1	5	0	19
Rata-Rata	1,00	5,00	0,00	19,00

Tabel 3.16 Centroid baru

Data ke	Centroid	Pengamatan	Must	Could	Wont
3	K1	2,67	13,00	11,00	2,67
7	K2	2,71	26,71	0,43	1,43
9	K3	1,00	5,00	0,00	9,00

Setelah mendapatkan nilai centroid baru, langkah selanjutnya adalah menemukan nilai fungsi objektif. Fungsi objektif digunakan untuk menilai konvergensi algoritma, jika perubahan J atau fungsi objektif dibawah ambang batas tertentu (0,1) algoritma dianggap telah mencapai konvergensi dan dapat dihentikan. Pada tahap ini, nilai awal fungsi objektif yang digunakan adalah

sebesar 1000 dengan nilai ambang batas sebesar 0,1. Sehingga didapatkan hasil seperti berikut:

Tabel 3.17 Fungsi objektif

Data	<i>C1</i>	<i>C2</i>	<i>C3</i>	<i>J</i>	Perubahan
1		3,62			
2	8,96				
3	4,27				
4		3,62			
5		3,62			
6		3,62		57,43	942,57
7		2,74			
8		5,01			
9			0,00		
10		8,90			
11	13,07				

Nilai *J* atau Fungsi objektif pada iterasi ke-1 adalah 57,43. Setelah mendapatkan nilai *J*, selanjutnya adalah mengetahui perubahan fungsi objektif dengan mengurangi nilai awal dengan nilai *J* pada iterasi saat ini yaitu $1000 - 57,43 = 942,57$.

4. Pengulangan hingga iterasi maksimum

Pada tahapan ini, melanjutkan dari tahapan sebelumnya apabila nilai *J* atau fungsi objektif masih melebihi nilai ambang batas yang telah ditentukan. Jika nilai *J* atau fungsi objektif masih melebihi nilai ambang batas, maka perhitungan dilanjutkan dengan cara yang sama tetapi menggunakan nilai centroid baru yang terdapat pada Tabel 3.16. Berikut adalah hasil *centroid* iterasi terakhir karena tidak ada perubahan pada nilai fungsi objektif.

Tabel 3.18 Cluster 1

Data Ke-	Pengamatan	Must	Could	Won't
2	3	17	3	3
3	3	16	8	3
11	2	6	22	2
Rata-rata	2,67	13,00	11,00	2,67

Tabel 3.19 Cluster 2

Data Ke-	Pengamatan	Must	Could	Won't
1	3	30	0	0
4	3	30	0	0
5	3	30	0	0
6	3	30	0	0
7	3	25	2	0
8	3	22	1	3
10	1	20	0	7
Rata-rata	2,71	26,71	0,43	1,43

Tabel 3.20 Cluster 3

Data Ke-	Pengamatan	Must	Could	Won't
9	1	5	0	19
Rata-Rata	1,00	5,00	0,00	19,00

Tabel 3.21 Centroid baru

Data ke	Centroid	Pengamatan	Must	Could	Wont
3	K1	2,67	13,00	11,00	2,67
7	K2	2,71	26,71	0,43	1,43
9	K3	1,00	5,00	0,00	9,00

Tabel 3.22 Fungsi objektif iterasi terakhir

Data	<i>C1</i>	<i>C2</i>	<i>C3</i>	<i>J</i>	<i>Perubahan</i>
1		3,62		57,43	0
2	8,96				
3	4,27				
4		3,62			
5		3,62			
6		3,62			
7		2,74			
8		5,01			
9			0,00		
10		8,90			
11	13,07				

Pada tabel 16 nilai fungsi objektif yang didapatkan adalah 57,43 dan nilai perubahan fungsi objektif yang didapatkan adalah $57,43 - 57,43 = 0$, yang berarti perubahan fungsi objektif pada iterasi tersebut sudah dibawah ambang batas yang telah ditentukan maka proses perhitungan dihentikan karena kondisi *cluster* sudah mencapai konvergen.

5. Pengelompokkan *cluster*

Langkah selanjutnya adalah pengelompokkan cluster berdasarkan nilai rata-rata dari tiap *cluster*. *Cluster* yang digunakan adalah *cluster* yang terakhir yaitu pada Tabel 3.21. Pada Tabel 3.21 masing-masing *centroid* akan dihitung rata-rata untuk mendapatkan urutan prioritas berdasarkan nilai tertinggi ke nilai terendah. Setelah itu menentukan beban kerja dengan durasi atau tenggat waktu pengembangan proyek sistem informasi inventori. Untuk menentukan beban kerja aktivitas pengembangan fitur, didapatkan dari beberapa penelitian atau buku yang telah terbit, diantaranya adalah *Software Engineering* karya Sommerville, I., *Software Engineering: A Practitioner's Approach* karya Pressman, R. S., dan *Testing Computer Software* karya

Kaner, C., Falk, J., Nguyen, Q. Tabel beban kerja tiap prioritas aktivitas pengembangan terdapat pada Tabel 3.23.

Tabel 3.23 Beban kerja tiap prioritas

Prioritas dan Beban Kerja		
Tinggi	Analisis dan Desain	5 Hari
	Pengembangan	5 Hari
	Pengujian	3 Hari
Sedang	Analisis dan Desain	4 Hari
	Pengembangan	4 Hari
	Pengujian	2 Hari
Rendah	Analisis dan Desain	3 Hari
	Pengembangan	3 Hari
	Pengujian	1 Hari

Langkah selanjutnya adalah menghitung rata-rata dari tiap *centroid* atau K untuk mengurutkan dari nilai tertinggi ke nilai terendah untuk menentukan prioritas fitur yaitu Tinggi, Sedang dan Rendah. Untuk menghitung nilai rata-rata adalah sebagai berikut.

$$\frac{\text{Pengamatan} + \text{Must} + \text{Could} + \text{Wont}}{4} \quad \text{Pers 1.1 Rata-rata centroid}$$

Tabel 3.24 Rata-rata nilai centroid

Variabel	Pengamatan	Must	Could	Wont	Rata-rata
K1	2,67	13,00	11,00	2,67	7,33
K2	2,71	26,71	0,43	1,43	7,82
K3	1,00	5,00	0,00	9,00	3,75

6. Prioritas dan beban kerja fitur

Langkah selanjutnya adalah menentukan prioritas dan beban kerja fitur yang telah didapatkan dari menghitung rata-rata dari masing-masing nilai *centroid* atau K.

Tabel 3.25 Prioritas dan beban kerja fitur

Prioritas dan Beban Kerja			
K2	Tinggi	Analisis dan Desain	5 Hari
		Pengembangan	5 Hari
		Pengujian	3 Hari
K1	Sedang	Analisis dan Desain	4 Hari
		Pengembangan	4 Hari
		Pengujian	2 Hari
K3	Rendah	Analisis dan Desain	3 Hari
		Pengembangan	3 Hari
		Pengujian	1 Hari

Tabel 3.26 Daftar Kelompok Aktivitas Pengembangan Sistem Informasi Inventori

No	Aktivitas Pengembangan	Cluster	Prioritas	Beban Kerja
1	Tambah barang	2	Tinggi	Analisis dan Desain: 5, Pengembangan: 5, Pengujian: 3
2	Edit Barang	1	Sedang	Analisis dan Desain: 4, Pengembangan: 4, Pengujian: 2
3	Hapus Barang	1	Sedang	Analisis dan Desain: 4, Pengembangan: 4, Pengujian: 2
4	List Barang	2	Tinggi	Analisis dan Desain: 5, Pengembangan: 5, Pengujian: 3
5	Tambah Penjualan	2	Tinggi	Analisis dan Desain: 5, Pengembangan: 5, Pengujian: 3

No	Aktivitas Pengembangan	Cluster	Prioritas	Beban Kerja
6	List Penjualan	2	Tinggi	Analisis dan Desain: 5, Pengembangan: 5, Pengujian: 3
7	Tambah Pembelian	2	Tinggi	Analisis dan Desain: 5, Pengembangan: 5, Pengujian: 3
8	List Pembelian	2	Tinggi	Analisis dan Desain: 5, Pengembangan: 5, Pengujian: 3
9	Hubungi Vendor	3	Rendah	Analisis dan Desain: 3, Pengembangan: 3, Pengujian: 1
10	Hubungi Owner	2	Tinggi	Analisis dan Desain: 5, Pengembangan: 5, Pengujian: 3
11	Hapus Penjualan	1	Sedang	Analisis dan Desain: 4, Pengembangan: 4, Pengujian: 2

Prioritas dan beban kerja fitur yang telah didapatkan dari algoritma k-means clustering dapat diimplementasikan kedalam perancangan jadwal proyek pengembangan sistem informasi inventori. Hasil dari penentuan prioritas dan beban kerja fitur terdapat durasi minimal dan maksimal, tujuan dari adanya durasi minimal dan maksimal adalah untuk memaksimalkan alokasi sumber daya pada pengembangan fitur, sehingga jika pengembangan fitur selesai pada hari ke-4, maka pengembang dapat melakukan debugging pada fitur tersebut selama batas waktu maksimal yang telah ditentukan, sehingga pemanfaatan waktu dalam pengembangan tiap fitur terjaga dan bisa selesai tepat pada durasi pengembangan proyek yang telah ditentukan.

3.4 Perancangan Sistem

Setelah melakukan analisis sistem dilakukan maka proses selanjutnya adalah perancangan sistem penjadwalan pengembangan proyek sistem informasi inventori. Perancangan sistem ini dikembangkan dengan menggunakan bahasa pemrograman python dengan memanfaatkan python library yang sudah tersedia dan disusun dalam bentuk flowchart, diagram konteks, diagram hirarki, dan data flow diagram.

3.4.1 Flowchart

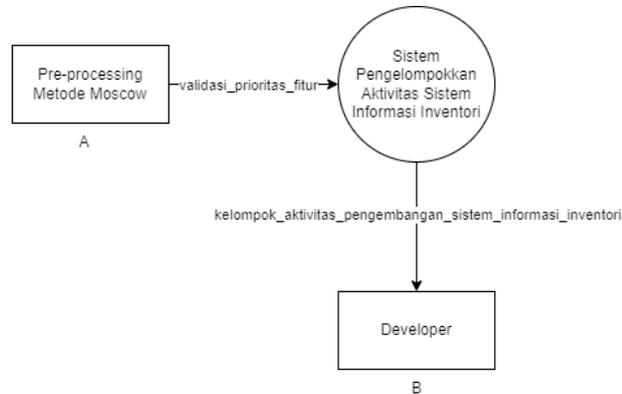
Berikut adalah urutan flowchart yang menggambarkan langkah-langkah dari sistem penjadwalan pengembangan proyek sistem informasi inventori:



Gambar 3.5 Flowchart sistem penjadwalan pengembangan proyek sistem informasi inventori

1. Pengembang memasukkan data dengan format yang terdapat pada tabel 3.6, yaitu kolom pengamatan, *must*, *should*, *could*, dan *won't*.
2. Proses perhitungan uji korelasi pearson dan clustering menggunakan metode K-Means Clustering untuk mendapatkan prioritas dan beban kerja fitur atau aktivitas pengembangan fitur.
3. Mendapatkan kelompok aktivitas pengembangan sistem informasi inventori dengan prioritas dan beban kerja yang dapat digunakan dalam menyusun jadwal pengembangan proyek sistem informasi inventori.

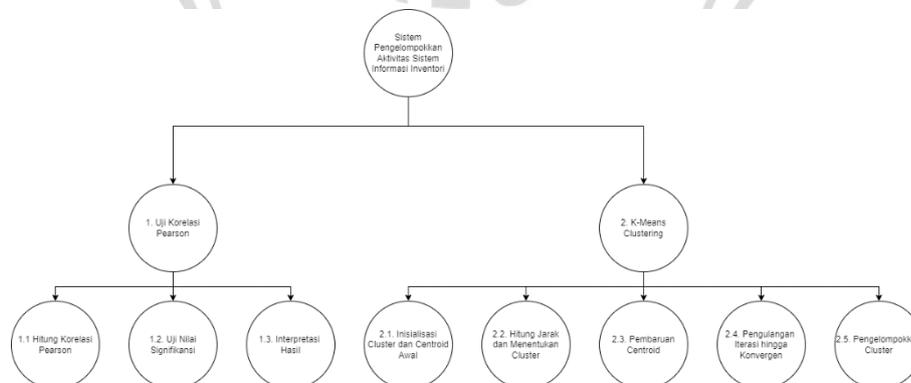
3.4.2 Diagram Konteks



Gambar 3.6 Diagram konteks sistem penjadwalan pengembangan proyek sistem informasi inventori

Data validasi prioritas fitur merupakan hasil dari *pre-processing* dengan menggunakan metode moscow untuk melakukan klasifikasi prioritas fitur. Validasi prioritas fitur diinputkan kedalam Sistem Pengelompokan Aktivitas Pengembangan Sistem Informasi Inventori dan akan dilakukan proses perhitungan menggunakan uji korelasi pearson dan metode k-means *clustering*. Data validasi prioritas akan menjadi dasar dalam melakukan proses perhitungan dalam uji korelasi dan pengelompokan prioritas dan beban kerja aktivitas pengembangan. Setelah proses perhitungan selesai maka developer mendapatkan jadwal untuk pengembangan sistem informasi inventori yang sudah terdapat prioritas dan beban kerja fitur.

3.4.3 Diagram Hirarki



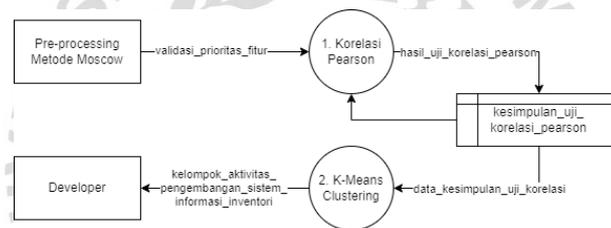
Gambar 7 Diagram Hirarki sistem penjadwalan pengembangan sistem informasi inventori

Pada gambar 3.7 dapat dilihat secara keseluruhan dari proses yang akan dilakukan pada sistem penjadwalan pengembangan proyek sistem informasi inventori menggunakan uji korelasi pearson dan k-means clustering. Berikut adalah penjelasannya:

1. Level Atas: Sistem Pengelompokkan Aktivitas Pengembangan Sistem Informasi Inventori.
2. Level 1: Berisi proses yang dilakukan di dalam sistem pengelompokkan aktivitas pengembangan sistem informasi inventori, yaitu proses perhitungan menggunakan uji korelasi pearson dan k-means clustering.
3. Level 2: Proses dari masing-masing perhitungan atau tahapan-tahapan dalam menggunakan uji korelasi pearson dan k-means clustering.

3.4.4 Data Flow Diagram

3.4.4.1 Data Flow Diagram Level 1



Gambar 3.8 Data flow diagram level 1 sistem pengelompokkan aktivitas pengembangan sistem informasi inventori

Dari gambar 3.8 merupakan gambar *data flow diagram* atau DFD *level 1* yang memiliki beberapa proses yaitu:

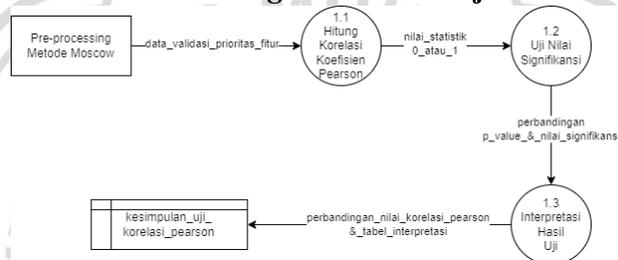
1. Data hasil dari *pre-processing* moscow yaitu data validasi prioritas fitur digunakan sebagai data yang akan dibandingkan dengan data pengamatan fitur dengan menggunakan uji korelasi pearson.
2. Data hasil uji korelasi pearson disimpan untuk digunakan di proses selanjutnya yaitu K-Means Clustering. Data tersebut berisi variabel prioritas moscow yang telah dieliminasi dan yang akan digunakan ke tahap selanjutnya. Variabel prioritas moscow dieliminasi jika nilai interpretasi diantara 4 variabel moscow

(Must Have, Should Have, Could Have, dan Won't Have) dengan variabel pengamatan bernilai terendah.

3. Data kesimpulan uji korelasi pearson digunakan di proses clustering menggunakan k-means clustering untuk mendapatkan kelompok aktivitas pengembangan sistem informasi inventori.
4. Hasil dari k-means clustering adalah jadwal pengembangan proyek sistem informasi inventori yang dapat digunakan oleh developer dalam mengembangkan proyek sistem informasi inventori.

3.4.4.2 Data Flow Diagram Level 2

3.4.4.2.1. Data Flow Diagram Level 2 Uji Korelasi Pearson

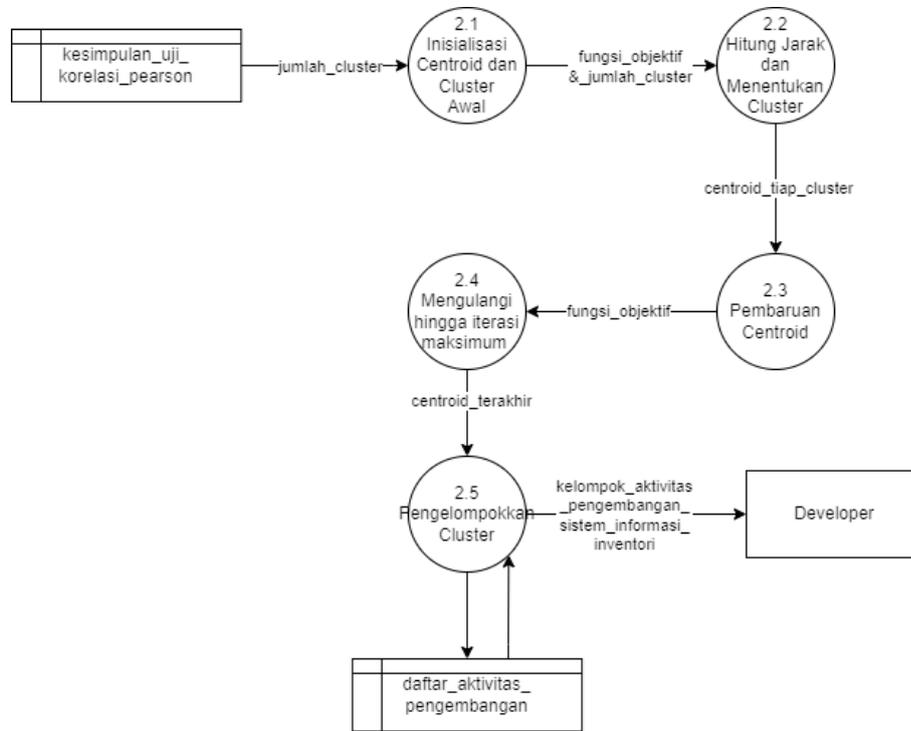


Gambar 3.9 DFD Level 2 Uji Korelasi Pearson

Dari gambar 3.9 merupakan gambar data flow diagram level 2 uji korelasi pearson yang terdapat beberapa proses antara lain:

1. Data validasi prioritas fitur dari metode moscow dan data pengamatan fitur digunakan sebagai data awal untuk melakukan proses menghitung korelasi koefisien pearson.
2. Proses 1.1 menghitung korelasi koefisien pearson.
3. Proses 1.2 menghitung uji nilai signifikansi korelasi pearson.
4. Proses 1.3 menghitung interpretasi hasil uji.
5. Menyimpan hasil kesimpulan uji korelasi pearson untuk digunakan pada proses selanjutnya yaitu k-means clustering.

3.4.4.2.2. Data Flow Diagram Level 2 K-Means Clustering

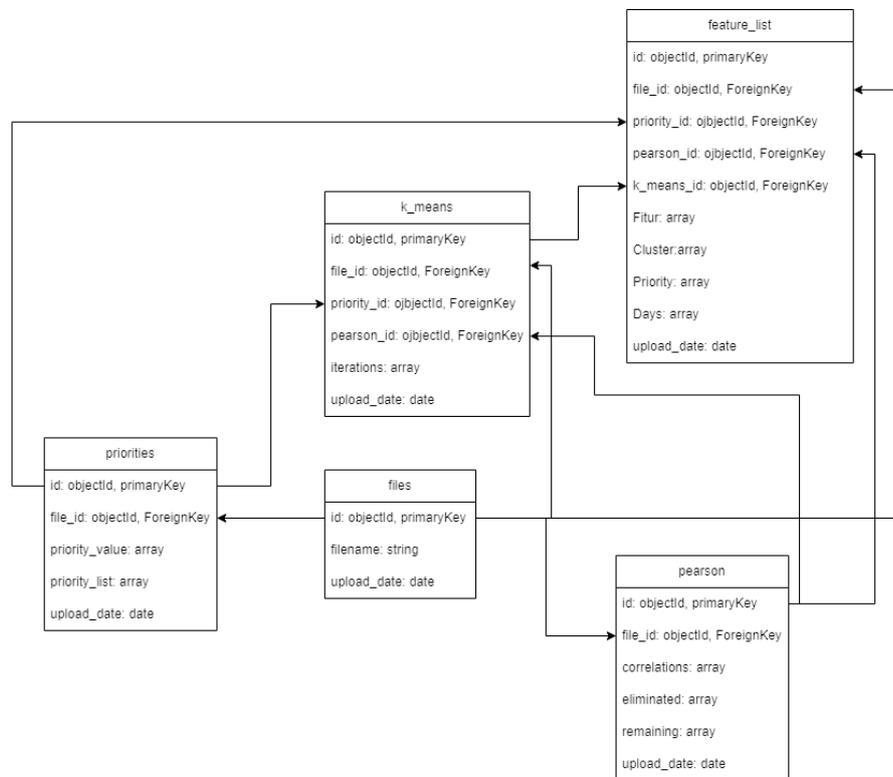


Gambar 3.10 DFD Level 2 K-Means Clustering

Dari gambar 3.10 merupakan *data flow diagram level 2* proses k-means clustering, masing-masing proses tersebut antara lain:

1. Data kesimpulan uji korelasi pearson digunakan sebagai data dan menentukan jumlah cluster
2. Proses 2.1 menentukan inisialisasi centroid, cluster, fungsi objektif, dan nilai ambang batas awal.
3. Proses 2.2 menghitung jarak dan menentukan cluster baru
4. Proses 2.3 memperbarui titik centroid baru
5. Proses 2.4 melakukan perulangan hingga iterasi maksimum atau konvergen
6. Proses 2.5 Pengelompokkan cluster
7. Pengelompokkan cluster menghasilkan prioritas dan beban kerja fitur yang dapat digunakan developer dalam melakukan pengembangan sistem informasi inventori.

3.5 Perancangan Basis Data



Gambar 3.11 Entity Relationship Diagram Sistem Pengelompokan Aktivitas Pengembangan Proyek SI Inventori

Pada sistem pengelompokan aktivitas pengembangan proyek sistem informasi inventori mempunyai sebuah *database* untuk menyimpan data-data yang digunakan. Dalam database terdapat struktur tabel yang merupakan susunan tabel pada *database* yang berfungsi sebagai rak agar data tersimpan rapi. Model untuk menggambarkan hubungan antara tabel dan memodelkan struktur data dari tabel terdapat pada gambar 3.11.

3.5.1 Tabel Files

Tabel *files* dibuat agar pengguna dapat menyimpan file yang diunggah oleh pengguna kedalam folder *upload*. Data *file* yang diunggah oleh pengguna tersimpan di dalam tabel *files*. Struktur dari tabel *files* dapat dilihat pada Tabel 3.26.

Tabel 3.27 Tabel *files*

No	Field	Type	Length	Key
1	<i>id</i>	<i>ObjectId</i>		<i>Primary Key</i>

No	Field	Type	Length	Key
2	<i>filename</i>	<i>String</i>	50	
3	<i>upload_date</i>	<i>date</i>		

3.5.2 Tabel Priorities

Tabel *priorities* adalah sebuah tabel yang menyimpan *prioritas* dan durasi hari dari masing-masing *prioritas*. Tabel ini memiliki relasi dengan tabel *files* untuk mengambil data *file*, seperti *filename* dan *upload date* dari *file* yang diunggah oleh pengguna. Struktur dari tabel *priorities* dapat dilihat pada Tabel 3.27.

Tabel 3.28 Tabel priorities

No	Field	Type	Length	Key
1	<i>id</i>	<i>ObjectId</i>		<i>Primary Key</i>
2	<i>file_id</i>	<i>ObjectId</i>		<i>Foreign Key</i>
3	<i>priority_value</i>	<i>array</i>		
4	<i>priority_list</i>	<i>array</i>		
5	<i>upload_date</i>	<i>date</i>		

3.5.3 Tabel Pearson

Tabel *pearson* adalah sebuah tabel yang menyimpan hasil uji korelasi *pearson*. Tabel ini memiliki relasi dengan tabel *files* untuk mengambil data *file*, seperti *filename* dan *upload date* dari *file* yang diunggah oleh pengguna. Struktur dari tabel *pearson* dapat dilihat pada Tabel 3.28.

Tabel 3.29 Tabel pearson

No	Field	Type	Length	Key
1	<i>id</i>	<i>ObjectId</i>		<i>Primary Key</i>
2	<i>file_id</i>	<i>ObjectId</i>		<i>Foreign Key</i>
3	<i>correlations</i>	<i>array</i>		
4	<i>eliminated</i>	<i>array</i>		
5	<i>remaining</i>	<i>array</i>		
6	<i>upload_date</i>	<i>date</i>		

3.5.4 Tabel K_means

Tabel *k_means* adalah sebuah tabel yang menyimpan data hasil k-means *clustering*. Tabel ini memiliki relasi dengan tabel *files*, tabel *priorities*, dan tabel *pearson* untuk mengambil masing-masing data dari masing-masing tabel, seperti data *filename* dari tabel *files*, data *priority* dari tabel *priorities*, dan data *pearson* dari tabel *pearsons*. Struktur dari tabel *k_means* dapat dilihat pada Tabel 3.29

Tabel 3.30 Tabel *k_means*

No	Field	Type	Length	Key
1	<i>id</i>	<i>ObjectId</i>		<i>Primary Key</i>
2	<i>file_id</i>	<i>ObjectId</i>		<i>Foreign Key</i>
3	<i>priority_id</i>	<i>ObjectId</i>		<i>Foreign Key</i>
4	<i>pearson_id</i>	<i>ObjectId</i>		<i>Foreign Key</i>
5	<i>iterations</i>	<i>ObjectId</i>		<i>Foreign Key</i>
6	<i>upload_date</i>	<i>date</i>		

3.5.5 Tabel *feature_list*

Tabel *feature_list* adalah tabel yang menyimpan prioritas dan beban kerja dari aktivitas *pengembangan* fitur sistem informasi inventori yang akan digunakan dalam menyusun jadwal pengembangan sistem informasi inventori. Tabel ini memiliki relasi dengan tabel *files*, tabel *priorities*, tabel *pearson*, dan tabel *k_means*, seperti data *filename* dari tabel *files*, data prioritas dari tabel *priorities*, data *pearson* dari tabel *pearsons*, dan data k-means *clustering* dari tabel *k_means*. Struktur dari tabel *feature_list* dapat dilihat pada tabel 3.30.

Tabel 3.31 Tabel *feature_list*

No	Field	Type	Length	Key
1	<i>id</i>	<i>ObjectId</i>		<i>Primary Key</i>
2	<i>file_id</i>	<i>ObjectId</i>		<i>Foreign Key</i>
3	<i>priority_id</i>	<i>ObjectId</i>		<i>Foreign Key</i>
4	<i>pearson_id</i>	<i>ObjectId</i>		<i>Foreign Key</i>
5	<i>k_means</i>	<i>ObjectId</i>		<i>Foreign Key</i>

No	Field	Type	Length	Key
6	Fitur	array		
7	Cluster	array		
8	Priority	array		
9	Days	array		
10	upload_date	date		

3.6 Perancangan Antarmuka

3.6.1 Antarmuka Halaman Awal



Sistem Pengelompokan Aktivitas
Pengembangan Proyek Sistem Informasi Inventori

Choose File No File Chosen

lorem ipsum dolor sit amet

Input 1

Input 2

Select ▼

Button

Gambar 3. 12 Antarmuka halaman awal

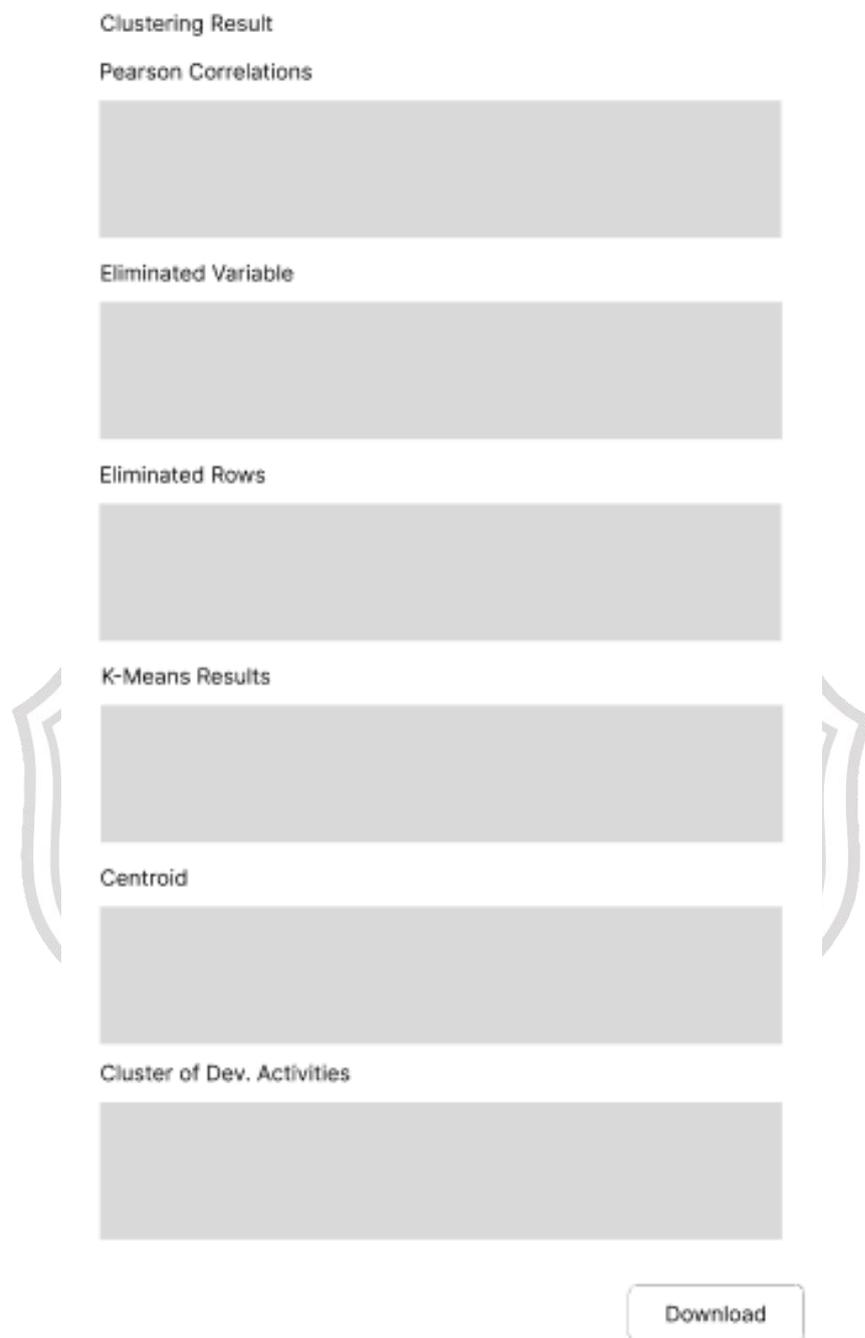
Halaman awal adalah halaman yang dimuat pertama kali oleh sistem. *Pada* halaman awal pengguna harus memasukkan *file excel* yang akan digunakan untuk mendapatkan prioritas dan beban kerja aktivitas pengembangan sistem informasi inventori. Untuk tampilan antarmuka halaman awal terdapat pada Gambar 3.12.

3.6.2 Antarmuka Hasil Perhitungan

Antarmuka hasil perhitungan setelah pengguna menekan tombol submit. *Pada* tampilan ini menampilkan hasil dari perhitungan uji korelasi pearson, dan k-

means clustering. Pada halaman ini pengguna dapat mendownload file nya dalam bentuk excel dengan prioritas dan beban kerja fitur yang sudah didapatkan dari hasil uji korelasi pearson dan k-means clustering, sehingga fitur-fitur tersebut dapat digunakan dalam jadwal pengembangan proyek sistem informasi inventori. Gambar desain antarmuka hasil perhitungan terdapat pada Gambar 3.13.





Gambar 3.13 Antarmuka hasil perhitungan uji korelasi pearson dan k-means *clustering*

3.7 Perencanaan Pengujian Sistem

Dalam pengujian sistem penjadwalan pengembangan proyek sistem informasi inventori terdapat langkah-langkah yang akan dilakukan dalam prosesnya sebagai berikut:

1. Mempersiapkan daftar fitur yang akan digunakan dalam sistem informasi inventori dan diklasifikasikan menggunakan metode moscow dan mempersiapkan data pengamatan pada jurnal penelitian terdahulu seperti pada tabel 3.6 dengan format excel atau xlsx.
2. Memasukkan atau mengunggah file daftar fitur ke dalam sistem penjadwalan dan memasukkan jumlah cluster yang diinginkan.
3. Kemudian memasukkan titik pusat setiap cluster atau centroid sesuai dengan banyak dari baris fitur yang terdapat di dalam file yang telah diunggah, memasukkan jumlah hari dari masing-masing centroid untuk menentukan prioritas tinggi, sedang dan rendah, dan jika semua kolom terisi tekan tombol submit untuk melakukan perhitungan uji korelasi pearson dan k-means clustering.
4. Kemudian didapatkan list atau daftar tabel korelasi pearson, variabel pearson yang dieliminasi beserta alasannya, tabel k-means clustering, tabel centroid yang berisi urutan prioritas centroid dari prioritas tinggi hingga rendah, dan tabel prioritas dan beban kerja dari masing-masing fitur. Hasil dari perhitungan tersebut dapat diunduh dalam format excel atau xlsx dan dapat digunakan dalam penyusunan jadwal pengembangan sistem informasi inventori.

3.8 Spesifikasi Pengembangan Sistem

1. Hardware: Macbook Air M3
2. Operating System: MacOS
3. Processor: Apple M3 Chip 8-Core
4. Ram: 8GB
5. Software: VSCode, Chrome