

## BAB 4

### IMPLEMENTASI DAN PENGUJIAN SISTEM

#### 4.1. Implementasi

Implementasi sistem dilakukan dengan menggunakan bahasa pemrograman python dan basis data mongodb. Bahasa pemrograman python digunakan karena sudah terdapat *library* untuk melakukan uji korelasi pearson dan k-means *clustering* sehingga dapat mempermudah dan mempersingkat waktu pengembangan sistem, dan basis data mongodb digunakan karena penggunaannya lebih fleksibel untuk digunakan dengan bahasa pemrograman python. Berikut adalah *source code* dari implmentasi sistem menggunakan bahasa pemrogaman python dan basis data mongodb berbasis *website*.

Pada halaman hasil perhitungan dikembangkan dengan *HTML*, *CSS library* tailwind css, flowbite, python dengan bantuan *library* flask, dan penyimpanan data kedalam basis data mongodb. Untuk *source code* halaman hasil perhitungan adalah sebagai berikut.

```
<!DOCTYPE html>
<html lang="en">

  <head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width,
    initial-scale=1.0">
    <title>Clustering Results</title>
    <link
      href="https://cdn.jsdelivr.net/npm/flowbite@2.5.2/dist/flowbite.min.css" rel="stylesheet" />
  </head>

  <body class="py-5">
    <div class="container mx-auto flex flex-col space-y-3">
      <h1 class="mt-5 text-3xl font-bold">Clustering
      Results</h1>

      <!-- Pearson Correlations -->
      <h2 class="mt-4 text-2xl font-semibold">Pearson
      Correlations of Moscow Variable with Pengamatan</h2>
      <table class="w-full divide-y divide-gray-200
      border">
        <thead class="bg-gray-50">
          <tr>
```

```

        <th class="px-6 py-3 text-left text-xs font-
medium text-gray-500 uppercase tracking-
wider">Variable</th>
        <th class="px-6 py-3 text-left text-xs font-
medium text-gray-500 uppercase tracking-
wider">Correlation</th>
    </tr>
</thead>
<tbody class="bg-white divide-y divide-gray-200">
    {% for var, corr in correlations.items() %}
    <tr>
        <td class="px-6 py-4 whitespace nowrap">{{ var
}}</td>
        <td class="px-6 py-4 whitespace nowrap">{{ corr
}}</td>
    </tr>
    {% endfor %}
</tbody>
</table>

<!-- Eliminated Variables -->
<h2 class="mt-4 text-2xl font-semibold">Eliminated
Variables</h2>
<table class="w-full border divide-y divide-gray-
200">
    <thead class="bg-gray-50">
        <tr>
            <th class="px-6 py-3 text-left text-xs font-
medium text-gray-500 uppercase tracking-
wider">Variable</th>
            <th class="px-6 py-3 text-left text-xs font-
medium text-gray-500 uppercase tracking-
wider">Correlation</th>
            <th class="px-6 py-3 text-left text-xs font-
medium text-gray-500 uppercase tracking-
wider">Reason</th>
        </tr>
    </thead>
    <tbody class="bg-white divide-y divide-gray-200">
        {% for item in eliminated_vars %}
        <tr>
            <td class="px-6 py-4 whitespace nowrap">{{ item.
variable }}</td>
            <td class="px-6 py-4 whitespace nowrap">{{ item.
correlation }}</td>
            <td class="px-6 py-4 whitespace nowrap">{{ item.
reason }}</td>
        </tr>
        {% endfor %}
    </tbody>
</table>

```

```

<!-- Eliminated Rows -->
<h2 class="mt-4 text-2xl font-semibold">Eliminated
Rows</h2>
<table class="w-full border divide-y divide-gray-
200">
  <thead class="bg-gray-50">
    <tr>
      {% for col in rows_to_remove[0].keys() %}
      <th class="px-6 py-3 text-left text-xs font-
medium text-gray-500 uppercase tracking-wider">{{ col
}}</th>
      {% endfor %}
    </tr>
  </thead>
  <tbody class="bg-white divide-y divide-gray-200">
    {% for row in rows_to_remove %}
    <tr>
      {% for col in row.values() %}
      <td class="px-6 py-4 whitespace nowrap">{{ col
}}</td>
      {% endfor %}
    </tr>
    {% endfor %}
  </tbody>
</table>

<!-- K-Means Results -->
<h2 class="mt-4 text-2xl font-semibold">K-Means
Clustering Results</h2>
<table class="w-full border divide-y divide-gray-
200">
  <thead class="bg-gray-50">
    <tr>
      <th class="px-6 py-3 text-left text-xs font-
medium text-gray-500 uppercase tracking-
wider">Iteration</th>
      <th class="px-6 py-3 text-left text-xs font-
medium text-gray-500 uppercase tracking-
wider">Centroids</th>
      <th class="px-6 py-3 text-left text-xs font-
medium text-gray-500 uppercase tracking-wider">Dev.
Activities
      </th>
      <th class="px-6 py-3 text-left text-xs font-
medium text-gray-500 uppercase tracking-
wider">Clusters</th>
      <th class="px-6 py-3 text-left text-xs font-
medium text-gray-500 uppercase tracking-
wider">Objective Function
      Values</th>
    </tr>
  </thead>

```

```

<tbody class="bg-white divide-y divide-gray-200">
    {%
        for i in
        range(kmeans_results['centroids'].length) %
    <tr>
        <td class="px-6 py-4 whitespace-nowrap">{{ i +
        1 }}</td>
        <td class="px-6 py-4 whitespace-nowrap">
            {%
                for centroid in
                kmeans_results['centroids'][i] %
            <strong>Centroid {{ loop.index
            }}:</strong><br>
                {{ centroid|join(', ')}<br><br>
                {%
                    endfor %
                }
            </td>
            <td class="px-6 py-4 whitespace-nowrap">
                {%
                    for value in
                    kmeans_results['column_a'][i] %
                {{ value }}<br>
                {%
                    endfor %
                }
            </td>
            <td class="px-6 py-4 whitespace-nowrap">
                {%
                    for cluster in
                    kmeans_results['clusters'][i] %
                {{ cluster + 1 }}<br>
                {%
                    endfor %
                }
            </td>
            <td class="px-6 py-4 whitespace-nowrap">
                <strong>Current:</strong> {{
                kmeans_results['objective_function_values'][i][0]
                }}<br>
                <strong>Delta:</strong> {{
                kmeans_results['objective_function_values'][i][1] }}
            </td>
        </tr>
        {%
            endfor %
        }
    </tbody>
</table>

<!-- Centroids -->
<h2 class="mt-4 text-2xl font-
semibold">Centroids</h2>
<table class="w-full border divide-y divide-gray-
200">
    <thead class="bg-gray-50">
        <tr>
            {%
                for col in df_centroids[0].keys() %
            <th class="px-6 py-3 text-left text-xs font-
            medium text-gray-500 uppercase tracking-wider">{{ col
            }}</th>
            {%
                endfor %
            }
        </tr>
    </thead>

```

```

<tbody class="bg-white divide-y divide-gray-200">
  {%
    for row in df_centroids %
  <tr>
    {%
      for col in row.values() %
    <td class="px-6 py-4 whitespace-nowrap">{{ col
    }}</td>
    {%
      endfor %
    </tr>
    {%
      endfor %
    </tbody>
  </table>

  <!-- Jadwal -->
  <h2 class="mt-4 text-2xl font-semibold">Cluster of
  Dev. Activities</h2>
  <table class="w-full border divide-y divide-gray-
  200">
    <thead class="bg-gray-50">
      <tr>
        {%
          for col in df_jadwal[0].keys() %
        <th class="px-6 py-3 text-left text-xs font-
        medium text-gray-500 uppercase tracking-wider">{{
        'Dev. Activity'
          if col == 'Fitur' else col }}</th>
        {%
          endfor %
        </tr>
      </thead>
      <tbody class="bg-white divide-y divide-gray-200">
        {%
          for row in df_jadwal %
        <tr>
          {%
            for col in row.values() %
          <td class="px-6 py-4 whitespace-nowrap">{{ col
          }}</td>
          {%
            endfor %
          </tr>
          {%
            endfor %
          </tbody>
        </table>

        <!-- Download Link -->
        <a href="{{ url_for('static',
        filename=download_link.split('/')[-1]) }}"
        class="mt-4 text-white underline text-right"><span
        class="bg-gray-900 px-4 py-2 rounded">Download
        the results</span></a>
      </div>

      <script
        src="https://cdn.jsdelivr.net/npm/flowbite@2.5.2/dist
        /flowbite.min.js"></script>
    </body>

```

```
</html>
```

Hasil dari implementasi kode diatas dapat dilihat pada Gambar 4.1.

Gambar 4.1 Halaman Hasil Perhitungan

Setelah implementasi desain antarmuka halaman *website*, selanjutnya implementasi rumus perhitungan uji korelasi pearson (Pers 2.1) dan rumus perhitungan k-means *clustering* (Pers 2.2) kedalam sistem pengelompokkan. *Source code* untuk implementasi rumus perhitungan uji korelasi pearson kedalam sistem adalah sebagai berikut.

```
import pandas as pd
import numpy as np
from sklearn.cluster import KMeans
from scipy.spatial import distance
```

Kode diatas berfungsi untuk mengimpor beberapa *library* yang digunakan dalam melakukan perhitungan korelasi pearson dan k-means *clustering*.

```
def load_data(file_path):
    df = pd.read_excel(file_path)
    df.rename(columns={"Feature": "Dev. Activity"}, inplace=True)
    return df
```

Kode diatas befungsi untuk memuat data dari file Excel dan mengganti nama kolom "Feature" menjadi "Dev. Activity".

```
def compute_pearson_correlation(df):
    corr_with_pengamatan = df.iloc[:, 2:].corrwith(df["Pengamatan"])
    corr_with_pengamatan = corr_with_pengamatan.round(2)
```

```

corr_with_pengamatan =
corr_with_pengamatan.reset_index()
corr_with_pengamatan.columns = ["Variable", "Pearson
Correlate"]
corr_with_pengamatan.index = np.arange(1,
len(corr_with_pengamatan) + 1)
return corr_with_pengamatan

```

Kode diatas befungsi untuk menghitung korelasi Pearson antara kolom "Pengamatan" dengan kolom lainnya. Dan untuk *round(2)* digunakan untuk membulatkan nilai ke dua angka desimal.

```

def interpret_corr(corr):
    if 0 <= corr <= 0.20:
        return "None"
    elif 0.21 <= corr <= 0.40:
        return "Weak"
    elif 0.41 <= corr <= 0.60:
        return "Moderate"
    elif 0.61 <= corr <= 0.80:
        return "Strong"
    else:
        return "Perfect"

```

Menginterpretasikan nilai korelasi Pearson dengan nilai "None" sampai nilai "Perfect".

```

def interpret_correlations(corr_with_pengamatan):
    correlations = {}
    for variable, correlation in
corr_with_pengamatan.items():
        if variable != "Pengamatan":
            correlations[variable] = correlation
    return correlations

```

Kode diatas befungsi untuk menginterpretasikan semua korelasi Pearson, yaitu nilai korelasi variabel Must Have, Should Have, Could Have, dan Wont Have dengan variabel Pengamatan.

```

def filter_data(df, correlations):
    abs_correlations = {key: abs(value) for key, value
in correlations.items()}
    min_abs_corr = min(abs_correlations.values())
    columns_to_drop = [
        key for key, value in abs_correlations.items()
        if value == min_abs_corr
    ]

    rows_to_remove = df[df["Classification"] ==
columns_to_drop[0]]
    df = df.drop(columns=columns_to_drop, axis=1)
    variable_to_remove = columns_to_drop[0]

```

```

df = df[df["Classification"] != variable_to_remove].reset_index(drop=True)
df.index = np.arange(1, len(df) + 1)

eliminated_vars = []
for col in columns_to_drop:
    eliminated_vars.append(
    {
        "variable": col,
        "correlation": correlations[col],
        "reason": "Lowest correlation",
    }
)

return df, eliminated_vars, df.columns.tolist(),
rows_to_remove

```

Kode diatas befungsi untuk menyaring data berdasarkan korelasi terendah dan menghapus variabel serta baris terkait.

Setelah implementasi rumus perhitungan uji korelasi pearson, selanjutnya adalah implementasi rumus perhitungan k-means *clustering*. *Source code* untuk implementasi rumus perhitungan k-means *clustering* adalah sebagai berikut.

```

def initialize_centroid(row_number, df, num_features):
    if not isinstance(row_number, int) or row_number < 0 or row_number >= df.shape[0]:
        raise ValueError(f"Invalid row number: {row_number}")
    centroid_data = df.iloc[row_number, 2:].values
    if len(centroid_data.shape) == 1:
        centroid = centroid_data
    else:
        centroid = centroid_data[:num_features]
    return centroid

```

Kode diatas befungsi untuk menginisialisasi centroid berdasarkan nomor baris data sesuai dengan yang diinput pada halaman awal.

```

def initial_centroids(df, row_numbers):
    num_features = len(df.columns) - 2
    centroids = []
    for row_number in row_numbers:
        adjusted_row_number = row_number - 1
        centroid =
        initialize_centroid(adjusted_row_number, df,
        num_features)
        centroids.append(centroid)
    centroids = np.vstack(centroids)
    return centroids

```

Kode diatas befungsi untuk menginisialisasi centroid awal berdasarkan nomor baris data sesuai dengan yang diinput pada halaman awal.

```
def assign_priorities(num_clusters):  
    if num_clusters == 2:  
        return ["High", "Low"]  
    elif num_clusters == 3:  
        return ["High", "Medium", "Low"]  
    else:  
        raise ValueError(f"Unsupported number of  
clusters: {num_clusters}")
```

Kode diatas befungsi untuk menentukan prioritas berdasarkan jumlah cluster.

Pada kode ini dibatasi minimal 2 cluster dan maksimal 3 cluster.

```
def get_priority_workload(priority):  
    workload = {  
        "High": {"Analysis and Design": 5,  
"Development": 5, "Testing": 3},  
        "Medium": {"Analysis and Design": 4,  
"Development": 4, "Testing": 2},  
        "Low": {"Analysis and Design": 3, "Development":  
3, "Testing": 1},  
    }  
    return ",\n".join([f"{key}: {value}" for key, value  
in workload[priority].items()])
```

Kode diatas befungsi untuk menentukan dan mendapatkan beban kerja berdasarkan prioritas. Pada kode ini dilakukan implementasi beban kerja prioritas secara statis, karena beban kerja prioritas sudah ditentukan sebelumnya.

```
def run_kmeans(df, centroids, num_clusters):  
    df_numeric = df.iloc[:, 2:]  
    kmeans = KMeans(  
        n_clusters=num_clusters,  
        init=centroids,  
        n_init="auto",  
        max_iter=100,  
        tol=0.1,  
        random_state=1000,  
    )  
    kmeans.fit(df_numeric)  
    return kmeans
```

Kode diatas befungsi untuk menjalankan algoritma K-Means *clustering* dengan nilai toleransi 0.1, nilai awal 1000.

```
def update_clusters(df, kmeans, centroids,  
priority_list, priority_values):  
    clusters = kmeans.labels_  
    df_with_clusters = df.copy()  
    df_with_clusters["Cluster"] = clusters + 1
```

```

df_with_clusters["Centroids"] =
str(centroids.round(2)).replace("\n", ", ")
df_with_clusters.index = np.arange(1,
len(df_with_clusters) + 1)
df_numeric = df.iloc[:, 2:]
distances = distance.cdist(df_numeric.values,
centroids, "euclidean")
for j in range(len(centroids)):
    df_with_clusters[f"K{j+1}"] = distances[:, j]
df_with_clusters["Min"] = df_with_clusters[
    [f"K{i+1}" for i in range(len(centroids))]].min(axis=1)
return df_with_clusters, clusters

```

Kode diatas befungsi untuk memperbarui cluster berdasarkan hasil perhitungan iterasi k-means.

```

def save_results(
    df_all_iterations,
    df_objective_function_values,
    df_centroids,
    df_jadwal,
    file_name,
    initial_data,
    eliminated_data,
    pearson_correlation,
):
    with pd.ExcelWriter(file_name) as writer:
        initial_data.to_excel(writer, index=True,
        sheet_name="Initial Data")
        pearson_correlation.to_excel(
            writer, index=True, sheet_name="Pearson
Correlation")
        eliminated_data.to_excel(writer, index=True,
        sheet_name="Eliminated Data")
        df_all_iterations.round(2).to_excel(writer,
        index=True, sheet_name="Iteration")
        df_objective_function_values.round(2).to_excel(
            writer, index=True, sheet_name="Objective
Function")
        df_centroids.round(2).to_excel(writer,
        index=True, sheet_name="Priority")
        df_jadwal.round(2).to_excel(writer, index=True,
        sheet_name="Jadwal")
    print("Results saved to:", file_name)

```

Kode diatas befungsi untuk menyimpan hasil perhitungan uji korelasi pearson dan k-means clustering ke dalam file Excel yang dapat di download oleh pengguna.

Setelah melakukan implementasi rumus perhitungan, selanjutnya adalah implementasi basis data kedalam sistem pengelompokan aktivitas pengembangan sistem informasi inventori. *Source code* nya adalah sebagai berikut.

```
import os

UPLOAD_FOLDER = "uploads"
STATIC_FOLDER = "static"

if not os.path.exists(UPLOAD_FOLDER):
    os.makedirs(UPLOAD_FOLDER)

if not os.path.exists(STATIC_FOLDER):
    os.makedirs(STATIC_FOLDER)
```

Kode diatas befungsi untuk mengimpor modul os yang digunakan untuk membuat sebuah folder jika folder upload dan static tidak ada.

```
from pymongo import MongoClient
from datetime import datetime
```

Kode diatas befungsi untuk mengimpor MongoClient dan datetime untuk dapat berinteraksi dengan client mongodb dan mendapatkan tanggal dan waktu saat ini.

```
def save_to_mongodb(
    correlations,
    kmeans_results,
    eliminated_vars,
    remaining_vars,
    db_name="penjadwalan",
    file_path=None,
    priority_values=None,
    priority_list=None,
    df_jadwal=None,
):
    client = MongoClient("mongodb://localhost:27017/")
    db = client[db_name]
```

Kode diatas befungsi untuk menyimpan data ke MongoDB dengan nama database “penjadwalan” dan menghubungkan basis data mongodb dengan mongourl agar bisa berinteraksi dengan basis data.

```
file_collection = db["files"]
file_data = {
    "filename": file_path,
    "upload_date": datetime.now(),
}
file_id =
file_collection.insert_one(file_data).inserted_id
```

Kode diatas befungsi untuk menyimpan informasi file yang diunggah oleh pengguna kedalam tabel atau collection files.

```
pearson_collection = db["pearson"]
correlation_data = {
    "file_id": file_id,
    "correlations": correlations,
    "eliminated": eliminated_vars,
    "remaining": remaining_vars,
    "upload_date": datetime.now(),
}
pearson_id =
pearson_collection.insert_one(correlation_data).inserted_id
```

Kode diatas befungsi untuk menyimpan data hasil perhitungan uji korelasi pearson kedalam tabel atau collection pearson.

```
priority_collection = db["priorities"]
priority_data = {
    "file_id": file_id,
    "priority_values": priority_values,
    "priority_list": priority_list,
    "upload_date": datetime.now(),
}
priority_id =
priority_collection.insert_one(priority_data).inserted_id
```

Kode diatas befungsi untuk menyimpan data prioritas kedalam tabel atau collection priorities.

```
kmeans_collection = db["k_means"]
kmeans_data = {
    "pearson_id": pearson_id,
    "priority_id": priority_id,
    "file_id": file_id,
    "iterations": [],
    "upload_date": datetime.now(),
}
```

Kode diatas befungsi untuk menyimpan hasil K-Means kedalam tabel atau collection k\_means.

```
for i, (centroids, clusters, objective_function_values)
in enumerate(
    zip(
        kmeans_results["centroids"],
        kmeans_results["clusters"],
        kmeans_results["objective_function_values"],
    )
):
    iteration_data = {
```

```

        "iteration": i + 1,
        "centroids": centroids,
        "clusters": clusters,
        "objective_function_values":
        objective_function_values,
    }
    kmeans_data["iterations"].append(iteration_data)

    k_means_id =
    kmeans_collection.insert_one(kmeans_data).inserted_id

```

Kode diatas befungsi untuk menambahkan data iterasi K-Means kedalam kolom iterations.

```

jadwal_collection = db["feature_list"]
jadwal_data = {
    "file_id": file_id,
    "pearson_id": pearson_id,
    "priority_id": priority_id,
    "kmeans_id": k_means_id,
    "Dev. Activity": df_jadwal.iloc[:, 0].tolist(),
    "Cluster": df_jadwal["Cluster"].tolist(),
    "Priority": df_jadwal["Priority"].tolist(),
    "Priorities and Workloads":
    df_jadwal["Priorities and Workloads"].tolist(),
    "upload_date": datetime.now(),
}
jadwal_collection.insert_one(jadwal_data)

print("Data saved to MongoDB")

```

Kode diatas befungsi untuk menyimpan data aktivitas pengembangan sistem informasi inventori kedalam tabel atau collection feature\_list.

Setelah melakukan implementasi basis data dengan sistem, selanjutnya adalah membuat rute untuk sistem pengelompokan aktivitas pengembangan sistem informasi inventori. Rute ini digunakan supaya pengguna dapat mengakses halaman secara berurutan. *Source code* implementasi rute adalah sebagai berikut.

```

from flask import Blueprint, request, render_template
from utils import (
    load_data,
    compute_pearson_correlation,
    interpret_correlations,
    filter_data,
    initial_centroids,
    assign_priorities,
    get_priority_workload,
    run_kmeans,
    update_clusters,
    save_results,
)

```

```
)  
from database import save_to_mongodb  
from config import UPLOAD_FOLDER, STATIC_FOLDER  
import os  
from datetime import datetime  
import numpy as np  
import pandas as pd
```

Kode diatas befungsi untuk mengimpor beberapa fungsi yang telah dibuat sebelumnya dan beberapa library.

```
main_bp = Blueprint("main", __name__)
```

Kode diatas befungsi untuk membuat *Blueprint* untuk rute utama.

```
@main_bp.route("/")  
def index():  
    return render_template("index.html")
```

Kode diatas befungsi untuk membuat rute untuk halaman utama, menggunakan file yang terdapat dalam folder templates/index.html.

```
@main_bp.route("/submit", methods=["POST"])  
def submit():  
    file = request.files["file"]  
    num_clusters = int(request.form["num_clusters"])
```

Kode diatas befungsi untuk membuat rute untuk mengirimkan file dan memproses data.

```
current_date = datetime.now().strftime("%Y-%m-%d")  
new_filename = f"{current_date}_{file.filename}"  
file_path = os.path.join("uploads", new_filename)
```

Kode diatas befungsi untuk menyimpan file yang diunggah dengan nama baru, dengan format tanggal\_namafile, dan menyimpan kedalam folder uploads.

```
df = load_data(file_path)  
initial_data = df.copy()  
initial_data.index = np.arange(1,  
len(initial_data) + 1)  
corr_with_pengamatan =  
compute_pearson_correlation(df)  
correlations = interpret_correlations()  
  
corr_with_pengamatan.set_index("Variable")  
["Pearson Correlate"]  
)  
df, eliminated_vars, remaining_vars,  
rows_to_remove = filter_data(df, correlations)
```

Kode diatas befungsi untuk memuat dan memproses data yang telah diupload oleh pengguna dan memanggil fungsi perhitungan korelasi pearson.

```

row_numbers =
[int(request.form[f"row_number_{i+1}"])]
for i in range(num_clusters):
    centroids = initial_centroids(df,
row_numbers)

```

Kode diatas befungsi untuk menginisialisasi centroid sesuai dengan yang diinputkan oleh pengguna.

```

priority_list =
assign_priorities(num_clusters)
priority_values = {
priority:
get_priority_workload(priority) for
priority in priority_list
}

```

Kode diatas befungsi untuk menentukan prioritas dan beban kerja dengan memanggil fungsi yang telah dibuat.

```

all_iterations = []
objective_function_values = []
j_initial = 1000
previous_cluster_labels = None

kmeans_results = {
"centroids": [],
"clusters": [],
"objective_function_values": [],
"column_a": []
}

for i in range(100):
    kmeans = run_kmeans(df, centroids, num_clusters)
    centroids = kmeans.cluster_centers_.copy()
    df_with_clusters, clusters = update_clusters(
        df, kmeans, centroids, priority_list,
        priority_values
    )
    if previous_cluster_labels is not None:
        df_with_clusters["Status"] = np.where(
            df_with_clusters["Cluster"] ==
previous_cluster_labels,
            "No Change",
            "Moved",
        )
    else:
        df_with_clusters["Status"] = "Initial
Assignment"
    previous_cluster_labels =
df_with_clusters["Cluster"].copy()
    all_iterations.append(df_with_clusters.copy())
    nan_df = pd.DataFrame()

```

```

        [""] * len(df_with_clusters.columns),
index=df_with_clusters.columns
).T
nan_df.index = [None]
all_iterations.append(nan_df)
j_current = kmeans.inertia_
delta_j = j_initial - j_current
objective_function_values.append(
    (round(j_initial, 2), round(j_current, 2),
round(delta_j, 2))
)
j_initial = j_current

kmeans_results["centroids"].append(np.round(centroids,
, 2).tolist())

kmeans_results["clusters"].append(clusters.tolist())

kmeans_results["objective_function_values"].append(
    (round(j_current, 2), round(delta_j, 2))
)
kmeans_results["column_a"].append(df["Dev.
Activity"].tolist())
if kmeans.n_iter_ == 1 and i > 0:
    break

df_all_iterations = pd.concat(all_iterations)

```

Kode diatas befungsi untuk menjalankan algoritma K-Means dengan nilai ambang batas atau toleransi 0.1, dan nilai awal sebesar 1000. Dan proses akan berhenti ketika data sudah menetap atau tidak ada perubahan cluster.

```

df_centroids = pd.DataFrame(centroids,
columns=df.columns[2:]).round(2)
df_centroids["Cluster"] = np.arange(1,
len(df_centroids) + 1)
df_centroids["Mean"] = (
df_centroids[df_centroids.columns.difference(["Cluste
r"])].mean(axis=1).round(2)
)
df_centroids = df_centroids.sort_values(by="Mean",
ascending=False).reset_index(
    drop=True
)
df_centroids["Priority"] = priority_list
df_centroids["Priorities and Workloads"] =
df_centroids["Cluster"].map(
    dict(zip(df_centroids["Cluster"],
priority_values.values())))
)

```

```

df_centroids.index = np.arange(1, len(df_centroids)
+ 1)

df_objective_function_values = pd.DataFrame(
{
    "Iteration": range(1, len(objective_function_values) + 1),
    "Initial J": [round(val[0], 2) for val in
objective_function_values],
    "Objective Function": [
        round(val[1], 2) for val in
objective_function_values
    ],
    "Delta J": [round(val[2], 2) for val in
objective_function_values],
}
)
df_objective_function_values.index = np.arange(
    1, len(df_objective_function_values) + 1
)

df_jadwal = df.iloc[:, [0]].copy()
df_jadwal.index = np.arange(1, len(df_jadwal) + 1)
df_jadwal["Cluster"] = clusters + 1
df_jadwal["Priority"] = df_jadwal["Cluster"].map(
    dict(zip(df_centroids["Cluster"],
df_centroids["Priority"]))
)
df_jadwal["Priorities and Workloads"] =
df_jadwal["Cluster"].map(
    dict(zip(df_centroids["Cluster"],
priority_values.values())))
)

```

Kode diatas befungsi untuk menyimpan hasil centroid dan kelompok aktivitas pengembangan sistem informasi inventori.

```

static_file_path = os.path.join(
    STATIC_FOLDER, new_filename
)
save_results(
    df_all_iterations,
    df_objective_function_values,
    df_centroids,
    df_jadwal,
    static_file_path,
    initial_data=initial_data,
    eliminated_data=rows_to_remove,
    pearson_correlation=corr_with_pengamatan,
)

```

Kode diatas befungsi untuk menyimpan hasil ke dalam folder statis agar bisa didownload oleh pengguna.

```
    save_to_mongodb(
        correlations,
        kmeans_results,
        eliminated_vars,
        remaining_vars,
        file_path=file_path,
        priority_values=priority_values,
        priority_list=priority_list,
        df_jadwal=df_jadwal,
    )

    static_file_path =
        static_file_path.replace("\\\\", "/")
```

Kode diatas befungsi untuk menyimpan hasil ke dalam basis data mongodb.

```
    return render_template(
        "result.html",
        correlations=correlations,
        kmeans_results=kmeans_results,
        eliminated_vars=eliminated_vars,
        remaining_vars=remaining_vars,
        df_centroids=df_centroids.round(2).to_dict(orient="records"),
        df_jadwal=df_jadwal.round(2).to_dict(orient="records"),
        rows_to_remove=rows_to_remove.round(2).to_dict(orient="records"),
        download_link=static_file_path,
    )
```

Kode diatas befungsi untuk menampilkan halaman *results* dari *templates/results.html*.

#### 4.2. Pengujian Sistem

Pengujian sistem dilakukan dengan berbagai teknik atau metode untuk memastikan bahwa sistem yang dikembangkan berfungsi dengan baik sesuai dengan spesifikasi dan memenuhi kebutuhan pengguna. Cara yang digunakan adalah dengan mengubah nilai *parameter* titik *centroid*, nilai awal atau *seed*, dan nilai ambang batas atau *tolerance*. Berikut adalah hasil pengujian sistem dengan menggunakan berbagai teknik atau metode dengan mengubah *parameter* k-means *clustering*.

#### 4.2.1. Pengujian Inisialisasi Manual.

##### 4.2.1.1. Nilai Awal 1300 dan Ambang Batas 0,1.

Pada pengujian ini dilakukan dengan menggunakan data yang terdapat pada Tabel 3.6. Pengujian ini dilakukan dengan cara menentukan titik *centroid* secara manual dengan menginputkan titik *centroid* secara manual yaitu berada pada titik data ke 3, 7, dan 9, lalu menentukan nilai awal sebesar 1300, dan nilai ambang batas 0,1. Berikut adalah hasil pengujian dengan menggunakan parameter tersebut.

#### **Sistem Pengelompokan Aktivitas Pengembangan Proyek Sistem Informasi Inventori**

Upload file

**Choose File** test skripsi.xlsx

XLSX file with the following columns:

Column	Description
Column A	Feature
Column B	Classification
Column C	Pengamatan
Column D	Must Have
Column E	Should Have
Column F	Could Have
Column G	Wont Have

Number of Clusters  
3

Number of clusters must be between 2 and 3.

Seed  
1300

Seed must be at least 0 and at most  $2^{32} - 1$ .

Tolerance  
0,1

Tolerance must be at least  $1e-10$  and at most  $1e-1$ .

Centroid Initialization Method

Manual

Manual centroid initialization requires you to input the row numbers for each cluster.

Row Number for Cluster 1  
3

Row Number for Cluster 2  
7

Row Number for Cluster 3  
9

**Submit**

Gambar 4.2 Inisialisasi *centroid* manual

Pada Gambar 4.2 menggunakan parameter nilai *seed* atau nilai awal sebesar 1300, nilai ambang batas atau *tolerance* sebesar 0,1, dan inisialisasi *centroid* secara manual pada titik data ke 3, 7, dan 9.

**Clustering Dev. Activities Results**

With manual method, seed 1300, tolerance 0.1

Pearson Correlations of Moscow Variable with Pengamatan

VARIABLE	CORRELATION
Must Have	0.78
Should Have	-0.15
Could Have	-0.56
Wont Have	-0.58

Eliminated Variables		
VARIABLE	CORRELATION	REASON
Should Have	-0.15	Lowest correlation

DEV. ACTIVITY	CLASSIFICATION	PENGAMATAN	MUST HAVE	SHOULD HAVE	COULD HAVE	WONT HAVE
Edit Penjualan	Should Have	2	3	20	5	2
Edit Pembelian	Should Have	2	5	17	5	3
Hapus Pembelian	Should Have	1	2	3	8	17
Tambah Vendor	Should Have	1	5	5	16	4
Edit Vendor	Should Have	1	4	5	17	4
Hapus Vendor	Should Have	1	6	2	17	5
List Vendor	Should Have	1	7	2	20	0

Gambar 4.3 Hasil uji korelasi pearson dan variabel yang tereliminasi

Pada Gambar 4.3 merupakan hasil uji korelasi pearson antara variabel Pengamatan dengan masing-masing variabel Moscow (*Must have*, *Should have*, *Could have*, dan *Wont have*). Dari hasil uji korelasi pearson, variabel yang memiliki nilai terendah akan dieliminasi beserta fitur-fitur yang memiliki klasifikasi yang sama dengan variabel tersebut. Pada Gambar 4.3 variabel yang memiliki nilai korelasi terendah adalah variabel *Should have*, oleh karena itu variabel *Should have* dieliminasi beserta fitur-fitur yang memiliki klasifikasi *Should have*.

K-Means Clustering Results

ITERATION	CENTROIDS	DEV. ACTIVITIES	CLUSTERS	OBJECTIVE FUNCTION VALUES
1	<b>Centroid 1:</b> 2.67, 13.0, 11.0, 2.67  <b>Centroid 2:</b> 2.71, 26.71, 0.43, 1.43  <b>Centroid 3:</b> 1.0, 5.0, 0.0, 19.0	Tambah barang Edit Barang Hapus Barang List Barang Tambah Penjualan List Penjualan Tambah Pembelian List Pembelian Hubungi Vendor Hubungi Owner Hapus Penjualan	2 1 1 2 2 2 2 3 2 1	<b>Current:</b> 433.62 <b>Delta:</b> 566.38
2	<b>Centroid 1:</b> 2.67, 13.0, 11.0, 2.67  <b>Centroid 2:</b> 2.71, 26.71, 0.43, 1.43  <b>Centroid 3:</b> 1.0, 5.0, 0.0, 19.0	Tambah barang Edit Barang Hapus Barang List Barang Tambah Penjualan List Penjualan Tambah Pembelian List Pembelian Hubungi Vendor Hubungi Owner Hapus Penjualan	2 1 1 2 2 2 2 3 2 1	<b>Current:</b> 433.62 <b>Delta:</b> 0.0

Centroids

PENGAMATAN	MUST HAVE	COULD HAVE	WONT HAVE	CLUSTER	MEAN	PRIORITY	PRIORITIES AND WORKLOADS
2.71	26.71	0.43	1.43	2	7.82	High	Analysis and Design: 5, Development: 5, Testing: 3
2.67	13.0	11.0	2.67	1	7.34	Medium	Analysis and Design: 4, Development: 4, Testing: 2
1.0	5.0	0.0	19.0	3	6.25	Low	Analysis and Design: 3, Development: 3, Testing: 1

Gambar 4.4 Iterasi k-means clustering dan centroid terakhir yang sudah memiliki prioritas dan beban kerja

Pada Gambar 4.4 merupakan hasil k-means *clustering* menjalankan 2 iterasi dengan nilai *centroid* yang sudah diperbarui pada setiap iterasi. Pada Gambar 4.4 juga sudah terdapat *centroid* tiap cluster yang akan digunakan untuk mendapatkan urutan prioritas dari tinggi, sedang, atau rendah dengan menghitung rata-rata dari setiap *centroid* dengan menggunakan rumus perhitungan (Pengamatan + Must Have + Could Have + Wont Have) / 4. Pada tabel *centroid* yang terdapat pada Gambar 4.4 juga sudah memiliki prioritas dan beban kerja sesuai pada Tabel 3.25.

Cluster of Dev. Activities			
DEV. ACTIVITY	CLUSTER	PRIORITY	PRIORITIES AND WORKLOADS
Tambah barang	2	High	Analysis and Design: 5, Development: 5, Testing: 3
Edit Barang	1	Medium	Analysis and Design: 4, Development: 4, Testing: 2
Hapus Barang	1	Medium	Analysis and Design: 4, Development: 4, Testing: 2
List Barang	2	High	Analysis and Design: 5, Development: 5, Testing: 3
Tambah Penjualan	2	High	Analysis and Design: 5, Development: 5, Testing: 3
List Penjualan	2	High	Analysis and Design: 5, Development: 5, Testing: 3
Tambah Pembelian	2	High	Analysis and Design: 5, Development: 5, Testing: 3
List Pembelian	2	High	Analysis and Design: 5, Development: 5, Testing: 3
Hubungi Vendor	3	Low	Analysis and Design: 3, Development: 3, Testing: 1
Hubungi Owner	2	High	Analysis and Design: 5, Development: 5, Testing: 3
Hapus Penjualan	1	Medium	Analysis and Design: 4, Development: 4, Testing: 2

**Pearson Computation Time:** 0.003 seconds  
**KMeans Computation Time:** 0.015 seconds  
**Total Computation Time:** 0.018 seconds

[Download the results](#)

Gambar 4.5 Daftar kelompok aktivitas pengembangan yang sudah memiliki prioritas dan beban kerja

Pada Gambar 4.5 merupakan daftar kelompok aktivitas pengembangan sistem informasi inventori yang sudah memiliki prioritas dan beban kerja. Pada luaran pengujian ini berbentuk daftar kelompok aktivitas pengembangan sistem informasi inventori yang dapat digunakan untuk penyusunan jadwal pengembangan sistem informasi inventori. Berikut adalah luaran atau *output* daftar kelompok aktivitas pengembangan yang telah didownload dalam bentuk file excel.

Tabel 4.1 *Output* pengujian manual (3,7,9), 1300, 0,1

No	Dev. Activity	Cluste r	Priorit y	Priorities and Workloads
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

No	Dev. Activity	Cluster	Priority	Priorities and Workloads
5	Tambah Penjualan	2	Tinggi	Analisis dan Desain: 5, Pengembangan: 5, Pengujian: 3
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

#### 4.2.1.2. Nilai Awal 1700 dan Ambang Batas 0,9

Pada pengujian ini dilakukan dengan menggunakan data yang terdapat pada Tabel 3.6. Pengujian ini dilakukan dengan cara menentukan titik *centroid* secara manual dengan menginputkan titik *centroid* secara manual yaitu berada pada titik data ke 3, 7, dan 9, lalu menentukan nilai awal sebesar 1700, dan nilai ambang batas 0,9. Berikut adalah hasil pengujian dengan menggunakan parameter tersebut.

## Sistem Pengelompokan Aktivitas Pengembangan Proyek Sistem Informasi Inventori

Upload file

**Choose File** test skripsi.xlsx

XLSX file with the following columns:

Column	Description
Column A	Feature
Column B	Classification
Column C	Pengamatan
Column D	Must Have
Column E	Should Have
Column F	Could Have
Column G	Wont Have

Number of Clusters

3

Number of clusters must be between 2 and 3.

Seed

1700

Seed must be at least 0 and at most  $2^{32} - 1$ .

Tolerance

0.9

Tolerance must be at least  $1e-10$  and at most  $1e-1$ .

Centroid Initialization Method

Manual

Manual centroid initialization requires you to input the row numbers for each cluster.

Row Number for Cluster 1

3

Row Number for Cluster 2

7

Row Number for Cluster 3

9

**Submit**

Gambar 4.6 Inisialisasi centroid manual

Pada Gambar 4.6 menggunakan parameter nilai *seed* atau nilai awal sebesar 1700, nilai ambang batas atau *tolerance* sebesar 0,9, dan inisialisasi *centroid* secara manual pada titik data ke 3, 7, dan 9.

### Clustering Dev. Activities Results

With manual method, seed 1700, tolerance 0.9

#### Pearson Correlations of Moscow Variable with Pengamatan

VARIABLE	CORRELATION
Must Have	0.78
Should Have	-0.15
Could Have	-0.56
Wont Have	-0.58

#### Eliminated Variables

VARIABLE	CORRELATION	REASON
Should Have	-0.15	Lowest correlation

#### Eliminated Rows

DEV. ACTIVITY	CLASSIFICATION	PENGAMATAN	MUST HAVE	SHOULD HAVE	COULD HAVE	WONT HAVE
Edit Penjualan	Should Have	2	3	20	5	2
Edit Pembelian	Should Have	2	5	17	5	3
Hapus Pembelian	Should Have	1	2	3	8	17
Tambah Vendor	Should Have	1	5	5	16	4
Edit Vendor	Should Have	1	4	5	17	4
Hapus Vendor	Should Have	1	6	2	17	5
List Vendor	Should Have	1	7	2	20	0

Gambar 4.7 Hasil uji korelasi pearson

Pada Gambar 4.7 merupakan hasil uji korelasi pearson antara variabel Pengamatan dengan masing-masing variabel Moscow (*Must have, Should have, Could have, dan Wont have*). Dari hasil uji korelasi pearson, variabel yang memiliki nilai terendah akan dieliminasi beserta fitur-fitur yang memiliki klasifikasi yang sama dengan variabel tersebut. Pada Gambar 4.7 variabel yang memiliki nilai korelasi terendah adalah variabel *Should have*, oleh karena itu variabel *Should have* dieliminasi beserta fitur-fitur yang memiliki klasifikasi *Should have*.

## K-Means Clustering Results

ITERATION	CENTROIDS	DEV. ACTIVITIES	CLUSTERS	OBJECTIVE FUNCTION VALUES
1	<b>Centroid 1:</b> 2.67, 13.0, 11.0, 2.67	Tambah barang	2	
		Edit Barang	1	
		Hapus Barang	1	
		List Barang	2	
	<b>Centroid 2:</b> 2.71, 26.71, 0.43, 1.43	Tambah Penjualan	2	
		List Penjualan	2	
	<b>Centroid 3:</b> 1.0, 5.0, 0.0, 19.0	Tambah Pembelian	2	<b>Current:</b> 433.62 <b>Delta:</b> 566.38
		List Pembelian	2	
		Hubungi Vendor	3	
		Hubungi Owner	2	
		Hapus Penjualan	1	
2	<b>Centroid 1:</b> 2.67, 13.0, 11.0, 2.67	Tambah barang	2	
		Edit Barang	1	
		Hapus Barang	1	
		List Barang	2	
	<b>Centroid 2:</b> 2.71, 26.71, 0.43, 1.43	Tambah Penjualan	2	
		List Penjualan	2	
	<b>Centroid 3:</b> 1.0, 5.0, 0.0, 19.0	Tambah Pembelian	2	<b>Current:</b> 433.62 <b>Delta:</b> 0.0
		List Pembelian	2	
		Hubungi Vendor	3	
		Hubungi Owner	2	
		Hapus Penjualan	1	

## Centroids

Centroid 3							
PENGAMATAN	MUST HAVE	COULD HAVE	WONT HAVE	CLUSTER	MEAN	PRIORITY	PRIORITIES AND WORKLOADS
2.71	26.71	0.43	1.43	2	7.82	High	Analysis and Design: 5, Development: 5, Testing: 3
2.67	13.0	11.0	2.67	1	7.34	Medium	Analysis and Design: 4, Development: 4, Testing: 2
1.0	5.0	0.0	19.0	3	6.25	Low	Analysis and Design: 3, Development: 3, Testing: 1

Gambar 4.8 Hasil k-means clustering

Pada Gambar 4.8 merupakan hasil k-means *clustering* menjalankan 2 iterasi dengan nilai *centroid* yang sudah diperbarui pada setiap iterasi. Pada Gambar 4.8 juga sudah terdapat *centroid* tiap cluster yang akan digunakan untuk mendapatkan urutan prioritas dari tinggi, sedang, atau rendah dengan menghitung rata-rata dari setiap *centroid* dengan menggunakan rumus perhitungan (Pengamatan + Must Have + Could Have + Wont Have) / 4. Pada tabel *centroid* yang terdapat pada Gambar 4.8 juga sudah memiliki prioritas dan beban kerja sesuai pada Tabel 3.25.

Cluster of Dev. Activities			
DEV. ACTIVITY	CLUSTER	PRIORITY	PRIORITIES AND WORKLOADS
Tambah barang	2	High	Analysis and Design: 5, Development: 5, Testing: 3
Edit Barang	1	Medium	Analysis and Design: 4, Development: 4, Testing: 2
Hapus Barang	1	Medium	Analysis and Design: 4, Development: 4, Testing: 2
List Barang	2	High	Analysis and Design: 5, Development: 5, Testing: 3
Tambah Penjualan	2	High	Analysis and Design: 5, Development: 5, Testing: 3
List Penjualan	2	High	Analysis and Design: 5, Development: 5, Testing: 3
Tambah Pembelian	2	High	Analysis and Design: 5, Development: 5, Testing: 3
List Pembelian	2	High	Analysis and Design: 5, Development: 5, Testing: 3
Hubungi Vendor	3	Low	Analysis and Design: 3, Development: 3, Testing: 1
Hubungi Owner	2	High	Analysis and Design: 5, Development: 5, Testing: 3
Hapus Penjualan	1	Medium	Analysis and Design: 4, Development: 4, Testing: 2

**Pearson Computation Time:** 0.003 seconds  
**KMeans Computation Time:** 0.012 seconds  
**Total Computation Time:** 0.015 seconds

[Download the results](#)

Gambar 4.9 Daftar kelompok aktivitas pengembangan

Pada Gambar 4.9 merupakan daftar kelompok aktivitas pengembangan sistem informasi inventori yang sudah memiliki prioritas dan beban kerja. Pada luaran pengujian ini berbentuk daftar kelompok aktivitas pengembangan sistem informasi inventori yang dapat digunakan untuk penyusunan jadwal pengembangan sistem informasi inventori. Berikut adalah luaran atau *output* daftar kelompok aktivitas pengembangan yang telah didownload dalam bentuk file excel.

Tabel 4.2 *Output* pengujian manual (3,7,9), 1700, 0,9

No	Dev. Activity	Cluster	Priority	Priorities and Workloads
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

No	Dev. Activity	Cluster	Priority	Priorities and Workloads
5	Tambah Penjualan	2	Tinggi	Analisis dan Desain: 5, Pengembangan: 5, Pengujian: 3
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

#### 4.2.2. Pengujian Inisialisasi *Random*.

##### 4.2.2.1. Nilai Awal 1300 dan Ambang Batas 0,1

Pada pengujian ini dilakukan dengan menggunakan data yang sama seperti pengujian sebelumnya, yaitu seperti pada Tabel 3.6. Pengujian ini dilakukan dengan cara menentukan titik centroid secara acak atau random, lalu menentukan nilai awal sebesar 1300, dan nilai ambang batas 0,1. Berikut adalah hasil pengujian dengan menggunakan parameter tersebut.

## Sistem Pengelompokkan Aktivitas Pengembangan Proyek Sistem Informasi Inventori

Upload file

**Choose File**

test skripsi.xlsx

XLSX file with the following columns:

Column	Description
Column A	Feature
Column B	Classification
Column C	Pengamatan
Column D	Must Have
Column E	Should Have
Column F	Could Have
Column G	Wont Have

Number of Clusters

3

Number of clusters must be between 2 and 3.

Seed

1300

Seed must be at least 0 and at most  $2^{32} - 1$ .

Tolerance

0,1

Tolerance must be at least  $1e-10$  and at most  $1e-1$ .

Centroid Initialization Method

Random

Manual centroid initialization requires you to input the row numbers for each cluster.

**Submit**

**Gambar 4.10 Inisialisasi *centroid random***

Pada Gambar 4.10 menggunakan parameter nilai *seed* atau nilai awal sebesar 1300, nilai ambang batas atau *tolerance* sebesar 0,1, dan inisialisasi *centroid* secara *random*.

### Clustering Dev. Activities Results

With random method, seed 1300, tolerance 0.1

#### Pearson Correlations of Moscow Variable with Pengamatan

VARIABLE	CORRELATION
Must Have	0.79
Should Have	-0.15
Could Have	-0.56
Wont Have	-0.58

#### Eliminated Variables

VARIABLE	CORRELATION	REASON
Should Have	-0.15	Lowest correlation

#### Eliminated Rows

DEV. ACTIVITY	CLASSIFICATION	PENGAMATAN	MUST HAVE	SHOULD HAVE	COULD HAVE	WONT HAVE
Edit Penjualan	Should Have	2	3	20	5	2
Edit Pembelian	Should Have	2	5	17	5	3
Hapus Pembelian	Should Have	1	2	3	8	17
Tambah Vendor	Should Have	1	5	5	16	4
Edit Vendor	Should Have	1	4	5	17	4
Hapus Vendor	Should Have	1	6	2	17	5
List Vendor	Should Have	1	7	2	20	0

Gambar 4.11 Hasil uji korelasi pearson

Pada Gambar 4.11 merupakan hasil uji korelasi pearson antara variabel Pengamatan dengan masing-masing variabel Moscow (*Must have, Should have, Could have, dan Wont have*). Dari hasil uji korelasi pearson, variabel yang memiliki nilai terendah akan dieliminasi beserta fitur-fitur yang memiliki klasifikasi yang sama dengan variabel tersebut. Pada Gambar 4.11 variabel yang memiliki nilai korelasi terendah adalah variabel *Should have*, oleh karena itu variabel *Should have* dieliminasi beserta fitur-fitur yang memiliki klasifikasi *Should have*.

K-Means Clustering Results

ITERATION	CENTROIDS	DEV. ACTIVITIES	CLUSTERS	OBJECTIVE FUNCTION VALUES
1	<b>Centroid 1:</b> 1.5, 5.5, 11.0, 10.5	Tambah barang	2	Current: 486.45 Delta: 513.55
		Edit Barang	3	
		Hapus Barang	3	
	<b>Centroid 2:</b> 3.0, 29.0, 0.4, -0.0	List Barang	2	
		Tambah Penjualan	2	
		List Penjualan	2	
	<b>Centroid 3:</b> 2.5, 18.75, 3.0, 4.0	Tambah Pembelian	2	
		List Pembelian	3	
		Hubungi Vendor	1	
2	<b>Centroid 1:</b> 1.5, 5.5, 11.0, 10.5	Hubungi Owner	3	Current: 486.45 Delta: 0.0
		Hapus Penjualan	1	
			1	
	<b>Centroid 2:</b> 3.0, 29.0, 0.4, -0.0		2	
			2	
			2	
	<b>Centroid 3:</b> 2.5, 18.75, 3.0, 4.0		3	
			1	
			3	
3	<b>Centroid 1:</b> 1.5, 5.5, 11.0, 10.5	Hapus Penjualan	1	Current: 486.45 Delta: 0.0
			1	
			1	
	<b>Centroid 2:</b> 3.0, 29.0, 0.4, -0.0		2	
			2	
			2	
	<b>Centroid 3:</b> 2.5, 18.75, 3.0, 4.0		3	
			1	
			3	

Centroids

PENGAMATAN	MUST HAVE	COULD HAVE	WONT HAVE	CLUSTER	MEAN	PRIORITY	PRIORITIES AND WORKLOADS
3.0	29.0	0.4	-0.0	2	8.1	High	Analysis and Design: 5, Development: 5, Testing: 3
1.5	5.5	11.0	10.5	1	7.12	Medium	Analysis and Design: 4, Development: 4, Testing: 2
2.5	18.75	3.0	4.0	3	7.06	Low	Analysis and Design: 3, Development: 3, Testing: 1

Gambar 4.12 Hasil k-means clustering

Pada Gambar 4.12 merupakan hasil k-means *clustering* menjalankan 2 iterasi dengan nilai *centroid* yang sudah diperbarui pada setiap iterasi. Pada Gambar 4.12 juga sudah terdapat *centroid* tiap cluster yang akan digunakan untuk mendapatkan urutan prioritas dari tinggi, sedang, atau rendah dengan menghitung rata-rata dari masing-masing *centroid*. Pada tabel *centroid* yang terdapat pada Gambar 4.12 juga sudah memiliki prioritas dan beban kerja sesuai pada Tabel 3.25.

Cluster of Dev. Activities

DEV. ACTIVITY	CLUSTER	PRIORITY	PRIORITIES AND WORKLOADS
Tambah barang	2	High	Analysis and Design: 5, Development: 5, Testing: 3
Edit Barang	3	Low	Analysis and Design: 3, Development: 3, Testing: 1
Hapus Barang	3	Low	Analysis and Design: 3, Development: 3, Testing: 1
List Barang	2	High	Analysis and Design: 5, Development: 5, Testing: 3
Tambah Penjualan	2	High	Analysis and Design: 5, Development: 5, Testing: 3
List Penjualan	2	High	Analysis and Design: 5, Development: 5, Testing: 3
Tambah Pembelian	2	High	Analysis and Design: 5, Development: 5, Testing: 3
List Pembelian	3	Low	Analysis and Design: 3, Development: 3, Testing: 1
Hubungi Vendor	1	Medium	Analysis and Design: 4, Development: 4, Testing: 2
Hubungi Owner	3	Low	Analysis and Design: 3, Development: 3, Testing: 1
Hapus Penjualan	1	Medium	Analysis and Design: 4, Development: 4, Testing: 2

**Pearson Computation Time:** 0.002 seconds

**KMeans Computation Time:** 0.013 seconds

**Total Computation Time:** 0.015 seconds

[Download the results](#)

Gambar 4.13 Daftar kelompok aktivitas pengembangan

Pada Gambar 4.13 merupakan daftar kelompok aktivitas pengembangan sistem informasi inventori yang sudah memiliki prioritas dan beban kerja. Pada luaran pengujian ini berbentuk daftar kelompok aktivitas pengembangan sistem informasi inventori yang dapat digunakan untuk penyusunan jadwal pengembangan sistem informasi inventori. Berikut adalah luaran atau *output* daftar kelompok aktivitas pengembangan yang telah *download* dalam bentuk file excel.

Tabel 4.3 Output pengujian random, 1300, 0,1

No	Dev. Activity	Cluster	Priority	Priorities and Workloads
1	Tambah barang	1	Tinggi	Analisis dan Desain: 5, Pengembangan: 5, Pengujian: 3
2	Edit Barang	3	Rendah	Analisis dan Desain: 3, Pengembangan: 3, Pengujian: 1
3	Hapus Barang	3	Rendah	Analisis dan Desain: 3, Pengembangan: 3, Pengujian: 1
4	List Barang	1	Tinggi	Analisis dan Desain: 5, Pengembangan: 5, Pengujian: 3
5	Tambah Penjualan	1	Tinggi	Analisis dan Desain: 5, Pengembangan: 5, Pengujian: 3
6	List Penjualan	1	Tinggi	Analisis dan Desain: 5, Pengembangan: 5, Pengujian: 3
7	Tambah Pembelian	1	Tinggi	Analisis dan Desain: 5, Pengembangan: 5, Pengujian: 3
8	List Pembelian	3	Rendah	Analisis dan Desain: 3, Pengembangan: 3, Pengujian: 1
9	Hubungi Vendor	3	Rendah	Analisis dan Desain: 3, Pengembangan: 3, Pengujian: 1
10	Hubungi Owner	3	Rendah	Analisis dan Desain: 3, Pengembangan: 3, Pengujian: 1

No	Dev. Activity	Cluster	Priority	Priorities and Workloads
11	Hapus Penjualan	2	Sedang	Analisis dan Desain: 4, Pengembangan: 4, Pengujian: 2

#### 4.2.2.2. Nilai Awal 1700 dan Ambang Batas 0,9

Pada pengujian ini dilakukan dengan menggunakan data yang sama seperti pengujian sebelumnya, yaitu seperti pada Tabel 3.6. Pengujian ini dilakukan dengan cara menentukan titik centroid secara acak atau random, lalu menentukan nilai awal sebesar 1700, dan nilai ambang batas 0,9. Berikut adalah hasil pengujian dengan menggunakan parameter tersebut.



## Sistem Pengelompokkan Aktivitas Pengembangan Proyek Sistem Informasi Inventori

Upload file

**Choose File**

test skripsi.xlsx

XLSX file with the following columns:

Column	Description
Column A	Feature
Column B	Classification
Column C	Pengamatan
Column D	Must Have
Column E	Should Have
Column F	Could Have
Column G	Wont Have

Number of Clusters

3

Number of clusters must be between 2 and 3.

Seed

1700

Seed must be at least 0 and at most  $2^{32} - 1$ .

Tolerance

0,9

Tolerance must be at least  $1e-10$  and at most  $1e-1$ .

Centroid Initialization Method

Random

Manual centroid initialization requires you to input the row numbers for each cluster.

**Submit**

Gambar 4.14 Inisialisasi *centroid random*

Pada Gambar 4.14 menggunakan parameter nilai *seed* atau nilai awal sebesar 1700, nilai ambang batas atau tolerance sebesar 0,9, dan inisialisasi *centroid* secara *random*.

**Clustering Dev. Activities Results**  
 With random method, seed 1700, tolerance 0.9  
**Pearson Correlations of Moscow Variable with Pengamatan**

VARIABLE	CORRELATION
Must Have	0.78
Should Have	-0.15
Could Have	-0.56
Wont Have	-0.58

**Eliminated Variables**

VARIABLE	CORRELATION	REASON
Should Have	-0.15	Lowest correlation

**Eliminated Rows**

DEV. ACTIVITY	CLASSIFICATION	PENGAMATAN	MUST HAVE	SHOULD HAVE	COULD HAVE	WONT HAVE
Edit Penjualan	Should Have	2	3	20	5	2
Edit Pembelian	Should Have	2	5	17	5	3
Hapus Pembelian	Should Have	1	2	3	8	17
Tambah Vendor	Should Have	1	5	5	16	4
Edit Vendor	Should Have	1	4	5	17	4
Hapus Vendor	Should Have	1	6	2	17	5
List Vendor	Should Have	1	7	2	20	0

Gambar 4.15 Hasil uji korelasi pearson

Pada Gambar 4.15 merupakan hasil uji korelasi pearson antara variabel Pengamatan dengan masing-masing variabel Moscow (*Must have, Should have, Could have, dan Wont have*). Dari hasil uji korelasi pearson, variabel yang memiliki nilai terendah akan dieliminasi beserta fitur-fitur yang memiliki klasifikasi yang sama dengan variabel tersebut. Pada Gambar 4.15 variabel yang memiliki nilai korelasi terendah adalah variabel *Should have*, oleh karena itu variabel *Should have* dieliminasi beserta fitur-fitur yang memiliki klasifikasi *Should have*.

K-Means Clustering Results

ITERATION	CENTROIDS	DEV. ACTIVITIES	CLUSTERS	OBJECTIVE FUNCTION VALUES
1	<b>Centroid 1:</b> 2.71, 26.71, 0.43, 1.43	Tambah barang Edit Barang Hapus Barang List Barang Tambah Penjualan List Penjualan Tambah Pembelian List Pembelian Hubungi Vendor Hubungi Owner Hapus Penjualan	1 3 3 1 1 1 1 2 1 3	<b>Current:</b> 433.62 <b>Delta:</b> 566.38
2	<b>Centroid 1:</b> 2.71, 26.71, 0.43, 1.43	Tambah barang Edit Barang Hapus Barang List Barang Tambah Penjualan List Penjualan Tambah Pembelian List Pembelian Hubungi Vendor Hubungi Owner Hapus Penjualan	1 3 3 1 1 1 1 1 2 1 3	<b>Current:</b> 433.62 <b>Delta:</b> 0.0

Centroids

PENGAMATAN	MUST HAVE	COULD HAVE	WONT HAVE	CLUSTER	MEAN	PRIORITY	PRIORITIES AND WORKLOADS
2.71	26.71	0.43	1.43	1	7.82	High	Analysis and Design: 5, Development: 5, Testing: 3
2.67	13.0	11.0	2.67	3	7.34	Medium	Analysis and Design: 4, Development: 4, Testing: 2
1.0	5.0	0.0	19.0	2	6.25	Low	Analysis and Design: 3, Development: 3, Testing: 1

Gambar 4.16 Hasil k-means clustering

Pada Gambar 4.16 merupakan hasil k-means clustering menjalankan 2 iterasi dengan nilai centroid yang sudah diperbarui pada setiap iterasi. Pada Gambar 4.12 juga sudah terdapat centroid tiap cluster yang akan digunakan untuk mendapatkan urutan prioritas dari tinggi, sedang, atau rendah dengan menghitung rata-rata dari masing-masing centroid. Pada tabel centroid yang terdapat pada Gambar 4.12 juga sudah memiliki prioritas dan beban kerja sesuai pada Tabel 3.25.

Cluster of Dev. Activities			
DEV. ACTIVITY	CLUSTER	PRIORITY	PRIORITIES AND WORKLOADS
Tambah barang	1	High	Analysis and Design: 5, Development: 5, Testing: 3
Edit Barang	3	Medium	Analysis and Design: 4, Development: 4, Testing: 2
Hapus Barang	3	Medium	Analysis and Design: 4, Development: 4, Testing: 2
List Barang	1	High	Analysis and Design: 5, Development: 5, Testing: 3
Tambah Penjualan	1	High	Analysis and Design: 5, Development: 5, Testing: 3
List Penjualan	1	High	Analysis and Design: 5, Development: 5, Testing: 3
Tambah Pembelian	1	High	Analysis and Design: 5, Development: 5, Testing: 3
List Pembelian	1	High	Analysis and Design: 5, Development: 5, Testing: 3
Hubungi Vendor	2	Low	Analysis and Design: 3, Development: 3, Testing: 1
Hubungi Owner	1	High	Analysis and Design: 5, Development: 5, Testing: 3
Hapus Penjualan	3	Medium	Analysis and Design: 4, Development: 4, Testing: 2

**Pearson Computation Time:** 0.003 seconds  
**KMeans Computation Time:** 0.014 seconds  
**Total Computation Time:** 0.017 seconds

[Download the results](#)

Gambar 4.17 Daftar kelompok aktivitas pengembangan

Pada Gambar 4.17 merupakan daftar kelompok aktivitas pengembangan sistem informasi inventori yang sudah memiliki prioritas dan beban kerja. Pada luaran pengujian ini berbentuk daftar kelompok aktivitas pengembangan sistem informasi inventori yang dapat digunakan untuk penyusunan jadwal pengembangan sistem informasi inventori. Berikut adalah luaran atau *output* daftar kelompok aktivitas pengembangan yang telah didownload dalam bentuk file excel.

Tabel 4.4 *Output* pengujian random, 1700, 0,9

No	Dev. Activity	Cluster	Priority	Priorities and Workloads
1	Tambah barang	1	Tinggi	Analisis dan Desain: 5, Pengembangan: 5, Pengujian: 3
2	Edit Barang	3	Sedang	Analisis dan Desain: 4, Pengembangan: 4, Pengujian: 2
3	Hapus Barang	3	Sedang	Analisis dan Desain: 4, Pengembangan: 4, Pengujian: 2
4	List Barang	1	Tinggi	Analisis dan Desain: 5, Pengembangan: 5, Pengujian: 3

No	Dev. Activity	Cluster	Priority	Priorities and Workloads
5	Tambah Penjualan	1	Tinggi	Analisis dan Desain: 5, Pengembangan: 5, Pengujian: 3
6	List Penjualan	1	Tinggi	Analisis dan Desain: 5, Pengembangan: 5, Pengujian: 3
7	Tambah Pembelian	1	Tinggi	Analisis dan Desain: 5, Pengembangan: 5, Pengujian: 3
8	List Pembelian	1	Tinggi	Analisis dan Desain: 5, Pengembangan: 5, Pengujian: 3
9	Hubungi Vendor	2	Rendah	Analisis dan Desain: 3, Pengembangan: 3, Pengujian: 1
10	Hubungi Owner	1	Tinggi	Analisis dan Desain: 5, Pengembangan: 5, Pengujian: 3
11	Hapus Penjualan	3	Sedang	Analisis dan Desain: 4, Pengembangan: 4, Pengujian: 2

#### 4.2.3. Pengujian Inisialisasi K-Means++.

##### 4.2.3.1. Nilai Awal 1300 dan Ambang Batas 0,1

Pada pengujian ini dilakukan dengan menggunakan data yang terdapat pada Tabel 3.6. Pengujian ini dilakukan dengan cara menentukan titik centroid secara manual dengan menginputkan titik centroid menggunakan metode k-means++ untuk mendapatkan hasil clustering yang lebih stabil, lalu menentukan nilai awal sebesar 1300, dan nilai ambang batas 0,1. Berikut adalah hasil pengujian dengan menggunakan parameter tersebut.

## Sistem Pengelompokkan Aktivitas Pengembangan Proyek Sistem Informasi Inventori

Upload file

**Choose File** test skripsi.xlsx

XLSX file with the following columns:

Column	Description
Column A	Feature
Column B	Classification
Column C	Pengamatan
Column D	Must Have
Column E	Should Have
Column F	Could Have
Column G	Wont Have

Number of Clusters

3

Number of clusters must be between 2 and 3.

Seed

1300

Seed must be at least 0 and at most  $2^{32} - 1$ .

Tolerance

0,1

Tolerance must be at least  $1e-10$  and at most  $1e-1$ .

Centroid Initialization Method

KMeans++

Manual centroid initialization requires you to input the row numbers for each cluster.

**Submit**

Gambar 4.18 Inisialisasi centroid kmeans++

Pada Gambar 4.18 menggunakan parameter nilai *seed* atau nilai awal sebesar 1300, nilai ambang batas atau *tolerance* sebesar 0,1, dan inisialisasi *centroid* menggunakan metode k-means++.

### Clustering Dev. Activities Results

With kmeans++ method, seed 1300, tolerance 0.1

#### Pearson Correlations of Moscow Variable with Pengamatan

VARIABLE	CORRELATION
Must Have	0.78
Should Have	-0.15
Could Have	-0.56
Wont Have	-0.58

#### Eliminated Variables

VARIABLE	CORRELATION	REASON
Should Have	-0.15	Lowest correlation

#### Eliminated Rows

DEV. ACTIVITY	CLASSIFICATION	PENGAMATAN	MUST HAVE	SHOULD HAVE	COULD HAVE	WONT HAVE
Edit Penjualan	Should Have	2	3	20	5	2
Edit Pembelian	Should Have	2	5	17	5	3
Hapus Pembelian	Should Have	1	2	3	8	17
Tambah Vendor	Should Have	1	5	5	16	4
Edit Vendor	Should Have	1	4	5	17	4
Hapus Vendor	Should Have	1	6	2	17	5
List Vendor	Should Have	1	7	2	20	0

Gambar 4.19 Hasil uji korelasi pearson

Pada Gambar 4.19 merupakan hasil uji korelasi pearson antara variabel Pengamatan dengan masing-masing variabel Moscow (*Must have, Should have, Could have, dan Wont have*). Dari hasil uji korelasi pearson, variabel yang memiliki nilai terendah akan dieliminasi beserta fitur-fitur yang memiliki klasifikasi yang sama dengan variabel tersebut. Pada Gambar 4.19 variabel yang memiliki nilai korelasi terendah adalah variabel *Should have*, oleh karena itu variabel *Should have* dieliminasi beserta fitur-fitur yang memiliki klasifikasi *Should have*.

K-Means Clustering Results

ITERATION	CENTROIDS	DEV. ACTIVITIES	CLUSTERS	OBJECTIVE FUNCTION VALUES	
1	<b>Centroid 1:</b> 2.78, 24.44, 1.56, 1.78  <b>Centroid 2:</b> 1.0, 5.0, 0.0, 19.0  <b>Centroid 3:</b> 2.0, 6.0, 22.0, 2.0	Tambah barang	1	<b>Current:</b> 383.56 <b>Delta:</b> 616.44	
		Edit Barang	1		
		Hapus Barang	1		
		List Barang	1		
		Tambah Penjualan	1		
		List Penjualan	1		
		Tambah Pembelian	1		
		List Pembelian	1		
		Hubungi Vendor	2		
		Hubungi Owner	1		
2	<b>Centroid 1:</b> 2.78, 24.44, 1.56, 1.78  <b>Centroid 2:</b> 1.0, 5.0, 0.0, 19.0  <b>Centroid 3:</b> 2.0, 6.0, 22.0, 2.0	Hapus Penjualan	3	<b>Current:</b> 383.56 <b>Delta:</b> 0.0	
		Tambah barang	1		
		Edit Barang	1		
		Hapus Barang	1		
		List Barang	1		
		Tambah Penjualan	1		
		List Penjualan	1		
		Tambah Pembelian	1		
		List Pembelian	1		
		Hubungi Vendor	2		
<b>Centroids</b>		Hubungi Owner	1		
		Hapus Penjualan	3		

Gambar 4.20 Hasil k-means clustering

Pada Gambar 4.20 merupakan hasil k-means *clustering* menjalankan 2 iterasi dengan nilai *centroid* yang sudah diperbarui pada setiap iterasi. Pada Gambar 4.20 juga sudah terdapat *centroid* tiap cluster yang akan digunakan untuk mendapatkan urutan prioritas dari tinggi, sedang, atau rendah dengan menghitung rata-rata dari setiap *centroid* dengan cara menjumlah seluruh data atau fitur pada *centroid* kemudian dibagi dengan banyak data atau fitur. Pada tabel *centroid* yang terdapat pada Gambar 4.20 juga sudah memiliki prioritas dan beban kerja sesuai pada Tabel 3.25.

Cluster of Dev. Activities				
DEV. ACTIVITY	CLUSTER	PRIORITY	PRIORITIES AND WORKLOADS	
Tambah barang	1	Medium	Analysis and Design: 4, Development: 4, Testing: 2	
Edit Barang	1	Medium	Analysis and Design: 4, Development: 4, Testing: 2	
Hapus Barang	1	Medium	Analysis and Design: 4, Development: 4, Testing: 2	
List Barang	1	Medium	Analysis and Design: 4, Development: 4, Testing: 2	
Tambah Penjualan	1	Medium	Analysis and Design: 4, Development: 4, Testing: 2	
List Penjualan	1	Medium	Analysis and Design: 4, Development: 4, Testing: 2	
Tambah Pembelian	1	Medium	Analysis and Design: 4, Development: 4, Testing: 2	
List Pembelian	1	Medium	Analysis and Design: 4, Development: 4, Testing: 2	
Hubungi Vendor	2	Low	Analysis and Design: 3, Development: 3, Testing: 1	
Hubungi Owner	1	Medium	Analysis and Design: 4, Development: 4, Testing: 2	
Hapus Penjualan	3	High	Analysis and Design: 5, Development: 5, Testing: 3	

**Pearson Computation Time:** 0.005 seconds  
**KMeans Computation Time:** 0.023 seconds  
**Total Computation Time:** 0.028 seconds

[Download the results](#)

Gambar 4.21 Daftar kelompok aktivitas pengembangan

Pada Gambar 4.21 merupakan daftar kelompok aktivitas pengembangan sistem informasi inventori yang sudah memiliki prioritas dan beban kerja. Pada luaran pengujian ini berbentuk daftar kelompok aktivitas pengembangan sistem informasi inventori yang dapat digunakan untuk penyusunan jadwal pengembangan sistem informasi inventori. Berikut adalah luaran atau *output* daftar kelompok aktivitas pengembangan yang telah didownload dalam bentuk file excel.

Tabel 4.5 *Output* pengujian kmeans++, 1300, 0,1

No	Dev. Activity	Cluster	Priority	Priorities and Workloads
1	Tambah barang	1	Sedang	Analisis dan Desain: 4, Pengembangan: 4, Pengujian: 2
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	1	Sedang	Analisis dan Desain: 4, Pengembangan: 4, Pengujian: 2

No	Dev. Activity	Cluster	Priority	Priorities and Workloads
5	Tambah Penjualan	1	Sedang	Analisis dan Desain: 4, Pengembangan: 4, Pengujian: 2
6	List Penjualan	1	Sedang	Analisis dan Desain: 4, Pengembangan: 4, Pengujian: 2
7	Tambah Pembelian	1	Sedang	Analisis dan Desain: 4, Pengembangan: 4, Pengujian: 2
8	List Pembelian	1	Sedang	Analisis dan Desain: 4, Pengembangan: 4, Pengujian: 2
9	Hubungi Vendor	2	Rendah	Analisis dan Desain: 3, Pengembangan: 3, Pengujian: 1
10	Hubungi Owner	1	Sedang	Analisis dan Desain: 4, Pengembangan: 4, Pengujian: 2
11	Hapus Penjualan	3	Tinggi	Analisis dan Desain: 5, Pengembangan: 5, Pengujian: 3

#### 4.2.3.2. Nilai Awal 1700 dan Ambang Batas 0,9

Pada pengujian ini dilakukan dengan menggunakan data yang terdapat pada Tabel 3.6. Pengujian ini dilakukan dengan cara menentukan titik centroid secara manual dengan menginputkan titik centroid menggunakan metode k-means++ untuk mendapatkan hasil clustering yang lebih stabil, lalu menentukan nilai awal sebesar 1700, dan nilai ambang batas 0,9. Berikut adalah hasil pengujian dengan menggunakan parameter tersebut.

## **Sistem Pengelompokan Aktivitas Pengembangan Proyek Sistem Informasi Inventori**

Upload file

**Choose File**

test skripsi.xlsx

XLSX file with the following columns:

Column	Description
Column A	Feature
Column B	Classification
Column C	Pengamatan
Column D	Must Have
Column E	Should Have
Column F	Could Have
Column G	Wont Have

Number of Clusters

3

Number of clusters must be between 2 and 3.

Seed

1700

Seed must be at least 0 and at most  $2^{32} - 1$ .

Tolerance

0,9

Tolerance must be at least  $1e-10$  and at most  $1e-1$ .

Centroid Initialization Method

KMeans++

Manual centroid initialization requires you to input the row numbers for each cluster.

**Submit**

**Gambar 4.22 Inisialisasi *centroid* K-Means++**

Pada Gambar 4.22 menggunakan parameter nilai *seed* atau nilai awal sebesar 1700, nilai ambang batas atau *tolerance* sebesar 0,9, dan inisialisasi *centroid* menggunakan metode k-means++.

### Clustering Dev. Activities Results

With kmeans++ method, seed 1700, tolerance 0.9

#### Pearson Correlations of Moscow Variable with Pengamatan

VARIABLE	CORRELATION
Must Have	0.78
Should Have	-0.15
Could Have	-0.56
Wont Have	-0.58

#### Eliminated Variables

VARIABLE	CORRELATION	REASON
Should Have	-0.15	Lowest correlation

#### Eliminated Rows

DEV_ACTIVITY	CLASSIFICATION	PENGAMATAN	MUST HAVE	SHOULD HAVE	COULD HAVE	WONT HAVE
Edit Penjualan	Should Have	2	3	20	5	2
Edit Pembelian	Should Have	2	5	17	5	3
Hapus Pembelian	Should Have	1	2	3	8	17
Tambah Vendor	Should Have	1	5	5	16	4
Edit Vendor	Should Have	1	4	5	17	4
Hapus Vendor	Should Have	1	6	2	17	5
List Vendor	Should Have	1	7	2	20	0

Gambar 4.23 Hasil uji korelasi pearson

Pada Gambar 4.23 merupakan hasil uji korelasi pearson antara variabel Pengamatan dengan masing-masing variabel Moscow (*Must have, Should have, Could have, dan Wont have*). Dari hasil uji korelasi pearson, variabel yang memiliki nilai terendah akan dieliminasi beserta fitur-fitur yang memiliki klasifikasi yang sama dengan variabel tersebut. Pada Gambar 4.23 variabel yang memiliki nilai korelasi terendah adalah variabel *Should have*, oleh karena itu variabel *Should have* dieliminasi beserta fitur-fitur yang memiliki klasifikasi *Should have*.

K-Means Clustering Results

ITERATION	CENTROIDS	DEV. ACTIVITIES	CLUSTERS	OBJECTIVE FUNCTION VALUES
1	<b>Centroid 1:</b> 1.0, 5.0, 0.0, 19.0  <b>Centroid 2:</b> 2.78, 24.44, 1.56, 1.78  <b>Centroid 3:</b> 2.0, 6.0, 22.0, 2.0	Tambah barang	2	
		Edit Barang	2	
		Hapus Barang	2	
2	<b>Centroid 1:</b> 1.0, 5.0, 0.0, 19.0  <b>Centroid 2:</b> 2.78, 24.44, 1.56, 1.78  <b>Centroid 3:</b> 2.0, 6.0, 22.0, 2.0	List Barang	2	
		Tambah Penjualan	2	
		List Penjualan	2	<b>Current:</b> 383.56 <b>Delta:</b> 616.44
	<b>Centroid 1:</b> 1.0, 5.0, 0.0, 19.0  <b>Centroid 2:</b> 2.78, 24.44, 1.56, 1.78  <b>Centroid 3:</b> 2.0, 6.0, 22.0, 2.0	Tambah Pembelian	2	
		List Pembelian	2	
		Hubungi Vendor	1	
	<b>Centroid 1:</b> 1.0, 5.0, 0.0, 19.0  <b>Centroid 2:</b> 2.78, 24.44, 1.56, 1.78  <b>Centroid 3:</b> 2.0, 6.0, 22.0, 2.0	Hubungi Owner	2	
		Hapus Penjualan	3	

Centroids

PENGAMATAN	MUST HAVE	COULD HAVE	WONT HAVE	CLUSTER	MEAN	PRIORITY	PRIORITIES AND WORKLOADS
2.0	6.0	22.0	2.0	3	8.0	High	Analysis and Design: 5, Development: 5, Testing: 3
2.78	24.44	1.56	1.78	2	7.64	Medium	Analysis and Design: 4, Development: 4, Testing: 2
1.0	5.0	0.0	19.0	1	6.25	Low	Analysis and Design: 3, Development: 3, Testing: 1

Gambar 4.24 Hasil k-means *clustering*

Pada Gambar 4.24 merupakan hasil k-means *clustering* menjalankan 2 iterasi dengan nilai *centroid* yang sudah diperbarui pada setiap iterasi. Pada Gambar 4.24 juga sudah terdapat *centroid* tiap cluster yang akan digunakan untuk mendapatkan urutan prioritas dari tinggi, sedang, atau rendah dengan menghitung rata-rata dari setiap *centroid* dengan cara menjumlah seluruh data atau fitur pada *centroid* kemudian dibagi dengan banyak data atau fitur. Pada tabel *centroid* yang terdapat pada Gambar 4.24 juga sudah memiliki prioritas dan beban kerja sesuai pada Tabel 3.25.

Cluster of Dev. Activities			
DEV. ACTIVITY	CLUSTER	PRIORITY	PRIORITIES AND WORKLOADS
Tambah barang	1	High	Analysis and Design: 5, Development: 5, Testing: 3
Edit Barang	3	Medium	Analysis and Design: 4, Development: 4, Testing: 2
Hapus Barang	3	Medium	Analysis and Design: 4, Development: 4, Testing: 2
List Barang	1	High	Analysis and Design: 5, Development: 5, Testing: 3
Tambah Penjualan	1	High	Analysis and Design: 5, Development: 5, Testing: 3
List Penjualan	1	High	Analysis and Design: 5, Development: 5, Testing: 3
Tambah Pembelian	1	High	Analysis and Design: 5, Development: 5, Testing: 3
List Pembelian	1	High	Analysis and Design: 5, Development: 5, Testing: 3
Hubungi Vendor	2	Low	Analysis and Design: 3, Development: 3, Testing: 1
Hubungi Owner	1	High	Analysis and Design: 5, Development: 5, Testing: 3
Hapus Penjualan	3	Medium	Analysis and Design: 4, Development: 4, Testing: 2

**Pearson Computation Time:** 0.003 seconds  
**KMeans Computation Time:** 0.014 seconds  
**Total Computation Time:** 0.017 seconds

[Download the results](#)

Gambar 4.25 Daftar kelompok aktivitas pengembangan

Pada Gambar 4.25 merupakan daftar kelompok aktivitas pengembangan sistem informasi inventori yang sudah memiliki prioritas dan beban kerja. Pada luaran pengujian ini berbentuk daftar kelompok aktivitas pengembangan sistem informasi inventori yang dapat digunakan untuk penyusunan jadwal pengembangan sistem informasi inventori. Berikut adalah luaran atau *output* daftar kelompok aktivitas pengembangan yang telah didownload dalam bentuk file excel.

Tabel 4.6 Output pengujian kmeans++, 1700, 0,9

No	Dev. Activity	Cluster	Priority	Priorities and Workloads
1	Tambah barang	2	Medium	Analysis and Design: 4, Development: 4, Testing: 2
2	Edit Barang	2	Medium	Analysis and Design: 4, Development: 4, Testing: 2
3	Hapus Barang	2	Medium	Analysis and Design: 4, Development: 4, Testing: 2
4	List Barang	2	Medium	Analysis and Design: 4, Development: 4, Testing: 2

No	Dev. Activity	Cluster	Priority	Priorities and Workloads
5	Tambah Penjualan	2	Medium	Analysis and Design: 4, Development: 4, Testing: 2
6	List Penjualan	2	Medium	Analysis and Design: 4, Development: 4, Testing: 2
7	Tambah Pembelian	2	Medium	Analysis and Design: 4, Development: 4, Testing: 2
8	List Pembelian	2	Medium	Analysis and Design: 4, Development: 4, Testing: 2
9	Hubungi Vendor	1	Low	Analysis and Design: 3, Development: 3, Testing: 1
10	Hubungi Owner	2	Medium	Analysis and Design: 4, Development: 4, Testing: 2
11	Hapus Penjualan	3	High	Analysis and Design: 5, Development: 5, Testing: 3

#### 4.3. Analisis Hasil Pengujian Sistem

Dengan menggunakan 3 teknik pengujian data fitur untuk mendapatkan kelompok aktivitas pengembangan sistem informasi inventori yang terdapat prioritas dan beban kerja dari masing-masing aktivitas seperti pada Tabel 3.25. Hasil pengujian pertama menggunakan data pada Tabel 3.6 dengan menggunakan *centroid* manual yang terletak pada titik data ke 3, 7, dan 9 menghasilkan daftar aktivitas pengembangan sistem informasi inventori seperti pada Gambar 4.5 dan Gambar 4.9.

Tabel 4.7 Daftar kelompok aktivitas pengembangan dengan inisialisasi manual pada uji 4.2.1.1

Manual (3,7,9), 1300, 0,1				
No	Dev. Activity	Cluster	Priority	Priorities and Workloads
1	Tambah barang	2	Tinggi	Analisis dan Desain: 5, Pengembangan: 5,

Manual (3,7,9), 1300, 0,1				
No	Dev. Activity	Cluster	Priority	Priorities and Workloads
				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
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

Tabel 4.8 Daftar kelompok aktivitas pengembangan dengan inisialisasi manual pada uji 4.2.1.2

Manual (3,7,9), 1700, 0,9				
No	Dev. Activity	Cluster	Priority	Priorities and Workloads
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
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,

Manual (3,7,9), 1700, 0,9				
No	Dev. Activity	Cluster	Priority	Priorities and Workloads
				Pengujian: 2

Pengujian selanjutnya menggunakan data yang sama yaitu data pada Tabel 3.6 tetapi menggunakan teknik inisialisasi *centroid* secara acak atau *random*, nilai awal atau *seed* dan nilai ambang batas atau *tolerance* yang berbeda dengan pengujian pertama, yaitu menggunakan teknik inisialisasi *centroid* secara acak yang terletak pada titik data ke 11,9,8, nilai awal 1300 nilai ambang batas 0,1, dan titik data ke 7, 10, 4, nilai awal 1700, nilai ambang batas 0,9. Hasil pengujian berbentuk tabel daftar kelompok aktivitas pengembangan sistem informasi inventori seperti pada Gambar 4.13 dan Gambar 4.17.

Tabel 4.9 Daftar kelompok aktivitas pengembangan dengan inisialisasi manual pada uji 4.2.2.1

Random, 1300, 0,1				
No	Dev. Activity	Cluster	Priority	Priorities and Workloads
1	Tambah barang	1	Tinggi	Analisis dan Desain: 5, Pengembangan: 5, Pengujian: 3
2	Edit Barang	3	Rendah	Analisis dan Desain: 3, Pengembangan: 3, Pengujian: 1
3	Hapus Barang	3	Rendah	Analisis dan Desain: 3, Pengembangan: 3, Pengujian: 1
4	List Barang	1	Tinggi	Analisis dan Desain: 5, Pengembangan: 5, Pengujian: 3
5	Tambah Penjualan	1	Tinggi	Analisis dan Desain: 5, Pengembangan: 5, Pengujian: 3
6	List Penjualan	1	Tinggi	Analisis dan Desain: 5, Pengembangan: 5, Pengujian: 3
7		1	Tinggi	Analisis dan Desain: 5,

Random, 1300, 0,1				
No	Dev. Activity	Cluster	Priority	Priorities and Workloads
	Tambah Pembelian			Pengembangan: 5, Pengujian: 3
8	List Pembelian	3	Rendah	Analisis dan Desain: 3, Pengembangan: 3, Pengujian: 1
9	Hubungi Vendor	3	Rendah	Analisis dan Desain: 3, Pengembangan: 3, Pengujian: 1
10	Hubungi Owner	3	Rendah	Analisis dan Desain: 3, Pengembangan: 3, Pengujian: 1
11	Hapus Penjualan	2	Sedang	Analisis dan Desain: 4, Pengembangan: 4, Pengujian: 2

Tabel 4.10 Daftar kelompok aktivitas pengembangan dengan inisialisasi manual pada uji 4.2.2.2

Random, 1700, 0,9				
No	Dev. Activity	Cluster	Priority	Priorities and Workloads
1	Tambah barang	1	Tinggi	Analisis dan Desain: 5, Pengembangan: 5, Pengujian: 3
2	Edit Barang	3	Sedang	Analisis dan Desain: 4, Pengembangan: 4, Pengujian: 2
3	Hapus Barang	3	Sedang	Analisis dan Desain: 4, Pengembangan: 4, Pengujian: 2
4	List Barang	1	Tinggi	Analisis dan Desain: 5, Pengembangan: 5, Pengujian: 3
5	Tambah Penjualan	1	Tinggi	Analisis dan Desain: 5, Pengembangan: 5, Pengujian: 3

Random, 1700, 0,9				
No	Dev. Activity	Cluster	Priority	Priorities and Workloads
6	List Penjualan	1	Tinggi	Analisis dan Desain: 5, Pengembangan: 5, Pengujian: 3
7	Tambah Pembelian	1	Tinggi	Analisis dan Desain: 5, Pengembangan: 5, Pengujian: 3
8	List Pembelian	1	Tinggi	Analisis dan Desain: 5, Pengembangan: 5, Pengujian: 3
9	Hubungi Vendor	2	Rendah	Analisis dan Desain: 3, Pengembangan: 3, Pengujian: 1
10	Hubungi Owner	1	Tinggi	Analisis dan Desain: 5, Pengembangan: 5, Pengujian: 3
11	Hapus Penjualan	3	Sedang	Analisis dan Desain: 4, Pengembangan: 4, Pengujian: 2

Pengujian terakhir menggunakan data yang sama seperti pengujian sebelumnya yaitu data pada Tabel 3.6 tetapi menggunakan metode inisialisasi *centroid* k-means++, nilai awal atau *seed* dan nilai ambang batas atau *tolerance* yang berbeda dengan pengujian pertama dan kedua, yaitu menggunakan metode inisialisasi *centroid* k-means++ yang terletak pada titik data ke 11, 7, dan 9, nilai awal 1500 dan nilai ambang batas 0,001. Hasil pengujian berbentuk tabel daftar kelompok aktivitas pengembangan sistem informasi inventori seperti pada Gambar 4.21 dan Gambar 4.25.

Tabel 4.11 Daftar kelompok aktivitas pengembangan dengan inisialisasi manual pada uji 4.2.3.1

Kmeans++, 1300, 0,1				
No	Dev. Activity	Cluster	Priority	Priorities and Workloads
1	Tambah barang	1	Sedang	Analisis dan Desain: 4, Pengembangan: 4,

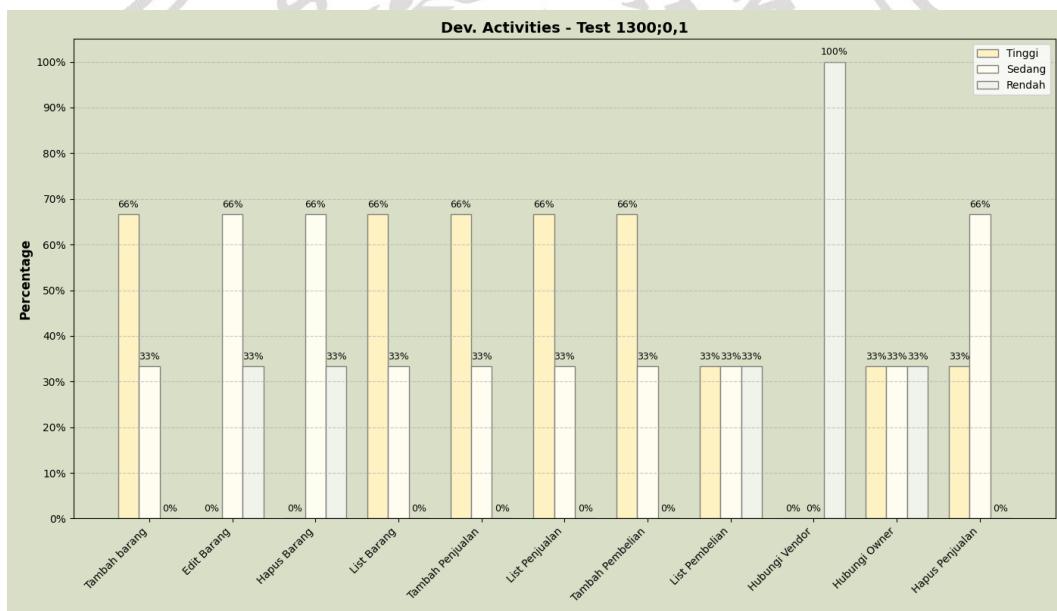
Kmeans++, 1300, 0,1				
No	Dev. Activity	Cluster	Priority	Priorities and Workloads
				Pengujian: 2
2	Edit Barang	1	Sedang	Analisis dan Desain: 4, Pengembangan: 4, Pengujian: 2
3	Hapus Barang	1	Sedang	Analisis dan Desain: 4, Pengembangan: 4, Pengujian: 2
4	List Barang	1	Sedang	Analisis dan Desain: 4, Pengembangan: 4, Pengujian: 2
5	Tambah Penjualan	1	Sedang	Analisis dan Desain: 4, Pengembangan: 4, Pengujian: 2
6	List Penjualan	1	Sedang	Analisis dan Desain: 4, Pengembangan: 4, Pengujian: 2
7	Tambah Pembelian	1	Sedang	Analisis dan Desain: 4, Pengembangan: 4, Pengujian: 2
8	List Pembelian	1	Sedang	Analisis dan Desain: 4, Pengembangan: 4, Pengujian: 2
9	Hubungi Vendor	2	Rendah	Analisis dan Desain: 3, Pengembangan: 3, Pengujian: 1
10	Hubungi Owner	1	Sedang	Analisis dan Desain: 4, Pengembangan: 4, Pengujian: 2
11	Hapus Penjualan	3	Tinggi	Analisis dan Desain: 5, Pengembangan: 5, Pengujian: 3

Tabel 4.12 Daftar kelompok aktivitas pengembangan dengan inisialisasi manual pada uji 4.2.3.2

Kmeans++, 1700, 0,9				
No	Dev. Activity	Cluster	Priority	Priorities and Workloads
1	Tambah barang	2	Sedang	Analisis dan Desain: 4, Pengembangan: 4, Pengujian: 2
2	Edit Barang	2	Sedang	Analisis dan Desain: 4, Pengembangan: 4, Pengujian: 2
3	Hapus Barang	2	Sedang	Analisis dan Desain: 4, Pengembangan: 4, Pengujian: 2
4	List Barang	2	Sedang	Analisis dan Desain: 4, Pengembangan: 4, Pengujian: 2
5	Tambah Penjualan	2	Sedang	Analisis dan Desain: 4, Pengembangan: 4, Pengujian: 2
6	List Penjualan	2	Sedang	Analisis dan Desain: 4, Pengembangan: 4, Pengujian: 2
7	Tambah Pembelian	2	Sedang	Analisis dan Desain: 4, Pengembangan: 4, Pengujian: 2
8	List Pembelian	2	Sedang	Analisis dan Desain: 4, Pengembangan: 4, Pengujian: 2
9	Hubungi Vendor	1	Rendah	Analisis dan Desain: 3, Pengembangan: 3, Pengujian: 1
10	Hubungi Owner	2	Sedang	Analisis dan Desain: 4, Pengembangan: 4, Pengujian: 2
11	Hapus Penjualan	3	Tinggi	Analisis dan Desain: 5, Pengembangan: 5,

Kmeans++, 1700, 0,9						
No	Dev. Activity	Cluster	Priority	Priorities and Workloads		
			Pengujian: 3			

Berdasarkan hasil pengujian yang telah dilakukan, Tabel 4.1 dan Tabel 4.2 hasilnya sama dengan hasil pengujian manual yang terdapat pada Tabel 3.26. Pengujian *random* dan kmeans++ memiliki hasil yang berbeda karena adanya perbedaan pada inisialisasi *centroid* awal, nilai awal atau *seed*, dan nilai ambang batas atau *tolerance* dapat mempengaruhi hasil pengelompokan aktivitas pengembangan sistem informasi inventori. Dari hasil pengujian sistem, dilakukan analisis hasil pengujian sistem dengan berbagai cara atau metode dengan menggunakan visualisasi grafik. Berikut adalah hasil analisis prioritas dari hasil pengujian menggunakan nilai awal 1300, nilai ambang batas 0,1, dan nilai awal 1700, nilai ambang batas 0,9.

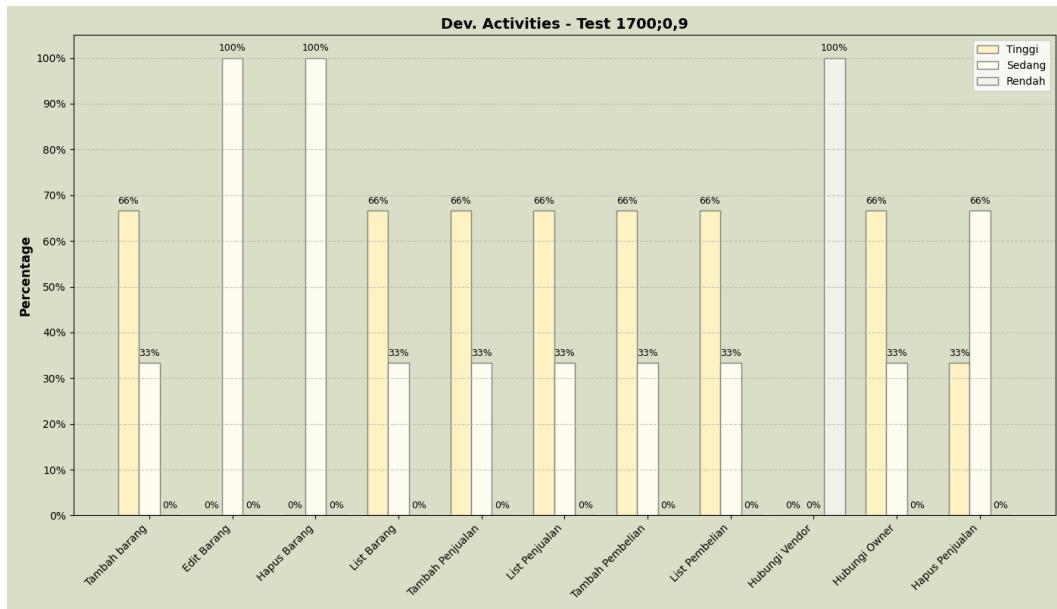


Gambar 4.26 Hasil pengujian nilai awal 1300, nilai ambang batas 0,1

Pada Gambar 4.26 menunjukkan hasil pengujian sistem menggunakan nilai awal 1300, dan nilai ambang batas 0,1 dengan tiga metode inisialisasi centroid yaitu manual, random, dan kmeans++. Hasil pengujian ini dianalisis untuk menentukan prioritas dari setiap aktivitas pengembangan. Persentase dari setiap prioritas dihitung berdasarkan frekuensi kemunculan dalam ketiga metode pengujian.

Diagram batang pada gambar menunjukkan persentase kemunculan setiap prioritas (Tinggi, Sedang, Rendah) berdasarkan hasil dari ketiga metode pengujian. Berikut adalah hasil analisis pengujian sistem menggunakan nilai awal 1300, dan nilai ambang batas 0,1.

1. Tambah Barang, List Barang, Tambah Penjualan, List Penjualan, Tambah Pembelian: aktivitas tersebut memiliki prioritas tinggi dengan nilai persentase sebesar 66%, dan prioritas sedang dengan nilai persentase sebesar 33% menunjukkan bahwa aktivitas tersebut dianggap penting dalam pengembangan sistem informasi inventori sehingga proses pengembangan wajib didahulukan.
2. Edit Barang dan Hapus Barang: aktivitas tersebut memiliki prioritas sedang dengan nilai persentase sebesar 66%, dan prioritas rendah dengan nilai persentase sebesar 33% menunjukkan bahwa aktivitas tersebut dianggap kurang penting dibandingkan dengan aktivitas lainnya dalam pengembangan sistem informasi inventori.
3. List Pembelian dan Hubungi Owner: aktivitas tersebut menunjukkan distribusi prioritas yang lebih merata, dengan nilai persentase masing-masing prioritas tinggi, sedang, dan rendah sebesar 33%. Aktivitas tersebut menunjukkan adanya ketidakpastian dalam penilaian pentingnya kedua aktivitas ini dalam pengembangan sistem informasi inventori.
4. Hubungi Vendor: aktivitas tersebut memiliki prioritas rendah dengan nilai persentase sebesar 100%, menunjukkan bahwa aktivitas ini dianggap paling tidak penting dalam pengembangan sistem informasi inventori.
5. Hapus Penjualan: aktivitas tersebut memiliki prioritas sedang dengan nilai persentase sebesar 67% dan prioritas tinggi dengan nilai persentase sebesar 33%. Aktivitas tersebut menunjukkan bahwa aktivitas dianggap cukup penting namun tidak sepenting aktivitas dengan prioritas sedang lainnya dalam pengembangan sistem informasi inventori.



Gambar 4.27 Hasil pengujian nilai awal 1700, nilai ambang batas 0,9

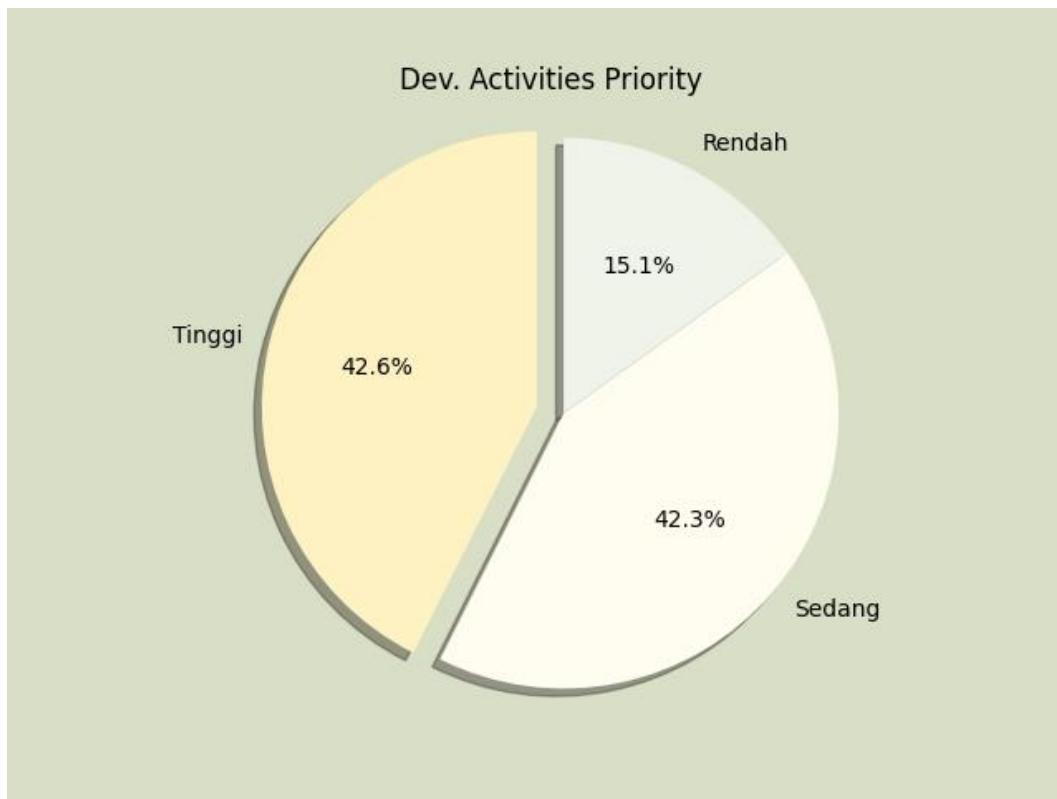
Pada Gambar 4.27 menunjukkan hasil pengujian sistem menggunakan nilai awal 1700, dan nilai ambang batas 0,9 dengan tiga metode inisialisasi centroid yaitu manual, random, dan kmeans++. Hasil pengujian ini dianalisis untuk menentukan prioritas dari setiap aktivitas pengembangan. Persentase dari setiap prioritas dihitung berdasarkan frekuensi kemunculan dalam ketiga metode pengujian. Diagram batang pada gambar menunjukkan persentase kemunculan setiap prioritas (Tinggi, Sedang, Rendah) berdasarkan hasil dari ketiga metode pengujian. Berikut adalah hasil analisis pengujian sistem menggunakan nilai awal 1700, dan nilai ambang batas 0,9.

1. Tambah Barang, List Barang, Tambah Penjualan, List Penjualan, Tambah Pembelian: Aktivitas-aktivitas tersebut memiliki prioritas tinggi sebesar 66% dan prioritas sedang sebesar 33%. Hal ini menunjukkan bahwa aktivitas-aktivitas tersebut dianggap sangat penting dalam pengembangan sistem informasi inventori.
2. Edit Barang dan Hapus Barang: Kedua aktivitas ini memiliki prioritas sedang sebesar 66% dan prioritas rendah sebesar 33%. Hal ini menunjukkan bahwa aktivitas-aktivitas ini dianggap kurang penting

dibandingkan dengan aktivitas lainnya dalam pengembangan sistem informasi inventori.

3. List Pembelian dan Hubungi Owner: Aktivitas-aktivitas tersebut memiliki prioritas yang lebih bervariasi dengan masing-masing prioritas sebesar 33%. Hal ini menunjukkan adanya ketidakpastian dalam penilaian pentingnya aktivitas-aktivitas tersebut dalam pengembangan sistem informasi inventori.
4. Hubungi Vendor: Aktivitas ini memiliki prioritas rendah sebesar 100%, menunjukkan bahwa aktivitas ini dianggap paling tidak penting dalam pengembangan sistem informasi inventori.
5. Hapus Penjualan: Aktivitas ini memiliki prioritas sedang sebesar 66% dan prioritas tinggi sebesar 33%. Ini menunjukkan bahwa aktivitas ini dianggap cukup penting namun tidak sepenting aktivitas dengan prioritas tinggi lainnya.

Setelah membandingkan hasil persentase prioritas dari masing-masing hasil pengujian sistem, selanjutnya adalah membandingkan hasil persentase prioritas dari kedua analisis perbandingan yang telah dilakukan. Perbandingan dilakukan dengan menggunakan persentase prioritas pengujian sistem dengan nilai awal 1300, nilai ambang batas 0,1 dengan persentase prioritas pengujian sistem dengan nilai awal 1700, ambang batas 0,9. Dengan membandingkan hasil persentase prioritas pengujian sistem, dapat dipahami bagaimana perubahan nilai awal dan ambang batas mempengaruhi hasil prioritas yang dihasilkan oleh ketiga metode inisialisasi centroid. Berikut adalah hasil perbandingan persentase pengujian dengan nilai awal 1300, nilai ambang batas 0,1 dengan nilai awal 1700, nilai ambang batas 0,9.



Gambar 4.28 Hasil analisis prioritas kelompok aktivitas pengembangan

Hasil analisis pada Gambar 4.28 merupakan hasil analisis dari keseluruhan prioritas kelompok aktivitas pengembangan sistem informasi inventori pada Gambar 4.28. Analisis ini dilakukan untuk mengetahui prioritas dari pengembangan sistem informasi inventori memiliki tingkat prioritas tinggi, sedang, atau rendah untuk didiskusikan dengan pihak-pihak terkait atau user. Sehingga pada Gambar 4.28 yaitu hasil analisis prioritas pengembangan sistem informasi inventori ini berada pada tingkat tinggi dengan nilai persentase sebesar 42.6%. Nilai persentase tersebut didapatkan dari rata-rata perhitungan dari seluruh aktivitas yang terdapat pada Gambar 4.27. Sehingga pengembangan proyek sistem informasi inventori dianggap sangat penting, dan perlu perencanaan yang sangat baik untuk memastikan bahwa semua aktivitas dapat diselesaikan sesuai dengan jadwal yang ditetapkan.

Hasil analisis hasil pengujian sistem digunakan untuk menyusun timeline atau jadwal pengembangan sistem informasi inventori yang lebih realistik dan terstruktur. Dengan mengetahui prioritas dari setiap aktivitas, tim pengembangan

dapat mengalokasikan waktu dan sumber daya secara efektif, sehingga proyek dapat diselesaikan tepat waktu. Berikut adalah *timeline* atau jadwal pengembangan sistem informasi inventori yang didapatkan dari analisis hasil pengujian sistem dengan menerapkan prioritas dan beban kerja aktivitas pengembangan seperti pada

Tabel 3.25.

Gambar 4.29 Jadwal pengembangan sistem informasi inventori