

Berikut adalah penjelasan detail dan contoh kode lengkap untuk file Flask (`app.py`) berdasarkan langkah-langkah yang disebutkan:

a. Import Modul dan Buat Aplikasi

Tambahkan modul yang dibutuhkan dan inialisasi aplikasi Flask:

```
from flask import Flask, render_template, jsonify
import pandas as pd
import plotly.express as px

# Inialisasi aplikasi Flask
app = Flask(__name__)
```

Instalasi modul
pip install Flask pandas plotly folium

b. Load Dataset

Tambahkan endpoint untuk halaman utama (`/`) yang memuat data dan merender template HTML:

```
@app.route('/')
def home():
    # Muat data SMA dan SMK
    sma_data = pd.read_csv('./data/data_aggregated_sma.csv')
    smk_data = pd.read_csv('./data/data_aggregated_smk.csv')

    # Kirim ke template jika data perlu ditampilkan (opsional)
    return render_template('index.html', sma_data=sma_data.head().to_html(),
                           smk_data=smk_data.head().to_html())
```

Penjelasan:

- `pd.read_csv()` digunakan untuk membaca dataset CSV.
- Data `sma_data` dan `smk_data` akan digunakan untuk analisis dan visualisasi.
- Parameter opsional `sma_data` dan `smk_data` bisa dikirim ke template HTML untuk debugging atau pratinjau.

c. Endpoint untuk Visualisasi

Endpoint ini memungkinkan visualisasi interaktif dengan menggunakan Plotly:

```
@app.route('/visualize/<cluster_type>')
def visualize(cluster_type):
    # Pilih data berdasarkan tipe klaster
    if cluster_type == 'SMA':
        data = pd.read_csv('./data/data_aggregated_sma.csv')
    elif cluster_type == 'SMK':
        data = pd.read_csv('./data/data_aggregated_smk.csv')
    else:
        return jsonify({"error": "Invalid cluster type"}), 400

    # Buat scatter plot menggunakan Dimensi PCA (Dim1, Dim2) dan warna berdasarkan Cluster
    fig = px.scatter(data, x='Dim1', y='Dim2', color='Cluster',
                    title=f'Cluster Visualization ({cluster_type})',
                    labels={'Dim1': 'Principal Component 1', 'Dim2': 'Principal Component 2'},
                    template='plotly_white')

    # Return plot sebagai JSON untuk digunakan di front-end
    return jsonify(fig.to_json())
```

Penjelasan:

- **cluster_type**: Parameter URL menentukan apakah data SMA atau SMK yang akan divisualisasikan.
- **px.scatter()**: Digunakan untuk membuat scatter plot berdasarkan dua dimensi utama (PCA).
- **fig.to_json()**: Mengonversi visualisasi menjadi JSON agar bisa digunakan di browser.

d. Jalankan Flask

Tambahkan fungsi utama untuk menjalankan aplikasi:

```
if __name__ == '__main__':
    app.run(debug=True)
```

File `app.py` Lengkap

Berikut adalah versi lengkapnya:

```
from flask import Flask, render_template, jsonify
import pandas as pd
import plotly.express as px

app = Flask(__name__)

# Endpoint untuk halaman utama
@app.route('/')
def home():
    # Muat data SMA dan SMK
    sma_data = pd.read_csv('./data/data_aggregated_sma.csv')
    smk_data = pd.read_csv('./data/data_aggregated_smk.csv')
    return render_template('index.html', sma_data=sma_data.head().to_html(),
                           smk_data=smk_data.head().to_html())

# Endpoint untuk visualisasi klaster
@app.route('/visualize/<cluster_type>')
def visualize(cluster_type):
    if cluster_type == 'SMA':
        data = pd.read_csv('./data/data_aggregated_sma.csv')
    elif cluster_type == 'SMK':
        data = pd.read_csv('./data/data_aggregated_smk.csv')
    else:
        return jsonify({"error": "Invalid cluster type"}), 400

    # Buat scatter plot
    fig = px.scatter(data, x='Dim1', y='Dim2', color='Cluster',
                     title=f'Cluster Visualization ({cluster_type})',
                     labels={'Dim1': 'Principal Component 1', 'Dim2': 'Principal Component 2'},
                     template='plotly_white')
    return jsonify(fig.to_json())

# Jalankan aplikasi
if __name__ == '__main__':
    app.run(debug=True)
```

Cara Menjalankan:

1. Simpan file ini dengan nama `app.py`.

Jalankan aplikasi Flask:

```
python app.py
```

- 2.
3. Buka browser dan akses:
 - <http://127.0.0.1:5000/> untuk halaman utama.
 - <http://127.0.0.1:5000/visualize/SMA> untuk visualisasi SMA.
 - <http://127.0.0.1:5000/visualize/SMK> untuk visualisasi SMK.

Jika memerlukan bantuan tambahan, beri tahu saya!