

## **Transformasi Data Statistik Menjadi Visual Interaktif Menggunakan Streamlit: Studi Kasus BPS Kota Mojokerto**

Kamilia Nabila Oktaviarini<sup>1\*</sup>, Eka Dyar Wahyuni<sup>1</sup>, Reisa Permata Sari<sup>1</sup>

<sup>1</sup>Universitas Pembangunan Nasional "Veteran" Jawa Timur, Surabaya, Indonesia

[20082010105@student.upnjatim.ac.id](mailto:20082010105@student.upnjatim.ac.id)\*

| Received: 04/12/2024 | Revised: -/-/ | Accepted: 30/12/2024 |

*Copyright©2024 by authors, all rights reserved. Authors agree that this article remains permanently open access under the terms of the Creative Commons Attribution License 4.0 International License*

### **Abstrak**

Penelitian ini bertujuan untuk mentransformasi penyajian data statistik Badan Pusat Statistik (BPS) Kota Mojokerto agar lebih interaktif dan mudah diakses oleh masyarakat. Penyajian data sebelumnya melalui menu grafik pada aplikasi mobile Dakocantik masih bergantung pada visualisasi Excel, yang dianggap kurang efektif. Untuk mengatasi kendala ini, penelitian mengembangkan dashboard berbasis web menggunakan Streamlit, yang menghadirkan visualisasi data secara interaktif dan informatif. Metode penelitian melibatkan studi literatur, wawancara, visualisasi data, dan integrasi sistem. Proses pengembangan meliputi identifikasi sumber data, ekstraksi, pembersihan, pemetaan, hingga penyimpanan data sebelum divisualisasikan menggunakan Streamlit. Dashboard ini diintegrasikan dengan aplikasi mobile Dakocantik melalui fitur webview, sehingga pengguna dapat mengakses data statistik interaktif langsung dari perangkat mobile. Hasil penelitian menunjukkan bahwa dashboard yang dikembangkan berhasil meningkatkan aksesibilitas dan pemahaman data statistik oleh masyarakat. Berdasarkan umpan balik dari 13 responden, 92,3% menyatakan bahwa dashboard mudah dipahami, dan 84,6% menilai fitur serta tampilannya mempermudah akses data. Beberapa masukan dari pengguna mencakup kebutuhan untuk meningkatkan waktu pemuatan (loading time) dan menyediakan pembaruan data otomatis guna mendukung pengalaman pengguna yang lebih baik.

Kata kunci: visualisasi data, Streamlit, BPS Kota Mojokerto, dashboard interaktif

### **Abstract**

*This study aims to transform the presentation of statistical data at the Central Bureau of Statistics (BPS) of Mojokerto City to be more interactive and accessible to the public. The previous data presentation through the graph menu on the Dakocantik mobile application relied on Excel-based visualizations, which were deemed less effective. To address this issue, this research developed a web-based dashboard using Streamlit, providing interactive and informative data visualization. The research methods included literature studies, interviews, data visualization, and system integration. The development process encompassed data source*

*identification, extraction, cleaning, mapping, and storage before visualization using Streamlit. The dashboard was integrated with the Dakocantik mobile application via the web view feature, enabling users to access interactive statistical data directly from their mobile devices. The study's findings indicate that the developed dashboard successfully enhanced data accessibility and understanding among the public. Feedback from 13 respondents revealed that 92.3% found the dashboard easy to understand, and 84.6% assessed its features and interface as user-friendly for accessing data. Suggestions from users included improving loading times and enabling automatic data updates to enhance user experience further.*

*Keywords: data visualization, Streamlit, Mojokerto City BPS, interactive dashboard*

## 1. Pendahuluan

Badan Pusat Statistik (BPS) adalah lembaga pemerintah yang bertanggung jawab menyediakan data statistik untuk mendukung pembangunan di tingkat nasional maupun regional. BPS Kota Mojokerto, menyajikan data statistik dalam bentuk *hardcopy* dan *softcopy* melalui perpustakaan dan situs web. Terbitan data statistik dari tahun ke tahun memiliki jumlah yang sangat banyak. Jumlah ini sering mengakibatkan kesulitan bagi instansi lain dan masyarakat umum untuk melihat dan memahami data statistik yang ada (Badan Pusat Statistik, 2023). Menurut hasil wawancara peneliti dengan Kasubbag Umum BPS Kota Mojokerto, masyarakat kesulitan memahami dan memanfaatkan data dengan optimal.

Untuk mengatasi masalah akses data dalam kemudahan layanan, BPS Kota Mojokerto telah membangun aplikasi mobile penyedia data statistik Kota Mojokerto bernama Aplikasi Dakocantik yang dimana, masyarakat lebih mudah untuk mendapatkan informasi statistik Kota Mojokerto. Namun, aplikasi ini masih menghadapi kendala seperti pembaruan visualisasi data yang menggunakan Excel dan visualisasi yang kurang efektif. Dalam era digital saat ini, kebutuhan akan akses informasi yang cepat dan akurat semakin meningkat, sehingga pengembangan platform yang dapat menyajikan data statistik secara interaktif menjadi sangat penting. Untuk meningkatkan kualitas layanan, diperlukan transformasi dalam cara penyajian data statistik menggunakan teknologi yang lebih modern. Dashboard ini diusulkan untuk dikembangkan menggunakan Streamlit, sebuah platform yang memungkinkan pembuatan aplikasi website interaktif dengan mudah.

Penggunaan dashboard berbasis teknologi interaktif, seperti Streamlit, memberikan kemudahan dalam memvisualisasikan data secara real-time dan lebih menarik. Penggunaan dashboard dapat meningkatkan interaksi pengguna dengan data, memungkinkan mereka untuk memperoleh informasi yang dibutuhkan dengan lebih cepat. Dengan visualisasi yang lebih jelas, data dapat disajikan dalam bentuk yang lebih mudah dipahami, sehingga meningkatkan kualitas pengambilan keputusan dan pemahaman informasi oleh masyarakat (Meliyana & Lathifah, 2022).

Penggunaan Streamlit sebagai platform untuk visualisasi data statistik dipilih karena keunggulannya dalam menyajikan data secara interaktif dan menarik (Meliyana & Lathifah, 2022). Hasil penelitian menunjukkan bahwa Streamlit dapat meningkatkan efektivitas penyampaian informasi melalui visualisasi data yang sederhana namun informatif. Streamlit terintegrasi dengan baik dengan pustaka Python seperti Pandas, Matplotlib, dan Seaborn. Integrasi

ini memungkinkan Streamlit untuk memanfaatkan kekuatan pustaka-pustaka tersebut dalam visualisasi dan integrasi data (Stevens, Smith dan Johnson, 2019). Selain itu, Streamlit memungkinkan pembaruan data secara real-time, sehingga informasi yang disajikan selalu akurat dan terkini (Shen, Lee dan Park, 2020). Fleksibilitas dan skalabilitas Streamlit memungkinkan pengembangan aplikasi web yang dapat disesuaikan dengan kebutuhan pengguna, menjadikannya pilihan yang tepat untuk berbagai jenis aplikasi (Huang, Wang, dan Liu, 2019).

Dengan adanya dashboard yang digunakan untuk menyajikan visualisasi data, diharapkan proses input, visualisasi dan integrasi data statistik dapat dilakukan secara real-time, sehingga memudahkan masyarakat untuk mengakses informasi data statistik yang dibutuhkan dengan cepat dan akurat. Dengan demikian, layanan digital BPS Kota Mojokerto dapat lebih optimal dan dapat memenuhi kebutuhan informasi statistik masyarakat secara lebih efektif.

## **2. Metodologi Penelitian**

### **2.1 Waktu dan Tempat Penelitian**

Penelitian ini dilaksanakan di Kota Mojokerto, dengan fokus pada data statistik yang disediakan oleh Badan Pusat Statistik (BPS) Kota Mojokerto. Waktu penelitian berlangsung dari Januari hingga Desember 2024, meliputi tahapan pengumpulan data, analisis, dan implementasi hasil penelitian ke dalam dashboard berbasis Streamlit dan aplikasi Kodular.

### **2.2 Alat dan Bahan**

Alat dan bahan yang digunakan dalam penelitian ini terdiri dari:

#### **1. Perlengkapan Hardware**

Hardware yang digunakan dalam pembuatan dashboard visualisasi dan integrasi ke aplikasi mobile adalah sebuah laptop ASUS X441M Intel Celeron N4000- 4GB/DDR4 - SSD 128GB.

#### **2. Perlengkapan Software**

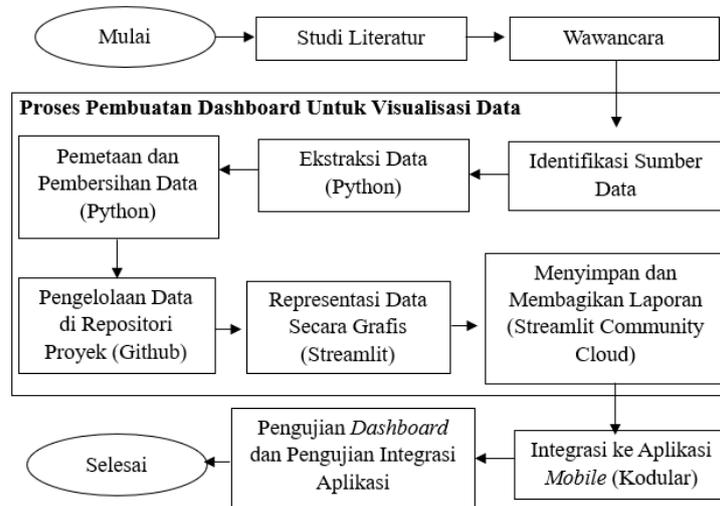
Software yang digunakan dalam pembuatan dashboard visualisasi dan integrasi ke aplikasi mobile adalah :

- a. Sistem Operasi Windows 11 64-bit
- b. Visual Studio Code
- c. Google Chrome
- d. Streamlit
- e. Streamlit Community Cloud
- f. Github
- g. Kodular
- h. Kodular Companion

### **2.3 Jenis Penelitian**

Penelitian ini menggunakan metodologi terstruktur yang fokus pada penggunaan platform Streamlit untuk visualisasi dan integrasi data statistik di BPS Kota Mojokerto. Metodologi ini meliputi studi literatur, wawancara, visualisasi data, integrasi aplikasi, serta pengujian dashboard dan aplikasi. Setiap langkah bertujuan mengolah dan menyajikan data dalam

format visual yang mudah dipahami. Berikut adalah alur penyelesaian yang digambarkan Gambar 1 di bawah ini:



Gambar 1 Diagram Alir Penelitian

a. Studi Literatur

Studi literatur merupakan tahap awal yang penting dalam penelitian ini, yang dilakukan melalui pencarian sistematis di berbagai sumber seperti website, jurnal, dan buku terkait visualisasi dan integrasi data statistik. Fokus utama literatur yang digunakan adalah pengolahan data, teknik visualisasi, serta platform Streamlit dan Kodular, dengan perhatian khusus pada aplikasi mobile "Dakocantik" yang dikembangkan menggunakan Kodular.

b. Wawancara

Wawancara merupakan bagian penting dalam penelitian ini untuk mengumpulkan informasi mengenai kebutuhan dan masalah BPS Kota Mojokerto terkait visualisasi dan integrasi data statistik. Metode yang digunakan adalah wawancara semi-terstruktur, yang memungkinkan fleksibilitas dalam menggali informasi lebih mendalam.

Pihak yang diwawancarai adalah Bapak Didin, Kasubbag Umum BPS, dan Ibu Mimik Nurjanti, Kepala BPS Kota Mojokerto. Wawancara dengan Kasubbag Umum berfokus pada tantangan operasional, sedangkan wawancara dengan Kepala BPS lebih menitikberatkan pada kebutuhan strategis dan visi jangka panjang terkait pengelolaan data. Informasi yang dikumpulkan, termasuk masalah dalam aplikasi "Dakocantik" dan harapan terhadap solusi visualisasi data, sangat penting untuk menyesuaikan metodologi penelitian dengan kebutuhan BPS Kota Mojokerto.

c. Proses Pembuatan Dashboard Untuk Visualisasi Data

Proses identifikasi sumber data merupakan langkah pertama yang sangat penting dalam penelitian ini. Identifikasi ini bertujuan untuk menentukan data yang relevan dan akan digunakan dalam visualisasi dan integrasi data statistik. Sumber data yang digunakan diambil dari dokumen internal BPS Kota Mojokerto, yang mencakup data sensus dan survei selama beberapa tahun.

Tahap ekstraksi data dilakukan untuk mengambil data mentah dari sumber yang telah diidentifikasi. Proses ini menggunakan pustaka Python seperti pandas untuk mengekstraksi data dari berbagai format file, seperti .xlsx dan .xls. Data yang telah diekstraksi kemudian disiapkan untuk diproses lebih lanjut, termasuk pembersihan dan pemetaan data. Berikut adalah proses ekstraksi data yang digambarkan Gambar 2 di bawah ini:

```
def load_data_pdrb(file_path):  
    # Tentukan engine berdasarkan ekstensi file  
    if file_path.endswith('.xlsx'):  
        data = pd.read_excel(file_path, engine='openpyxl')  
    elif file_path.endswith('.xls'):  
        data = pd.read_excel(file_path, engine='xlrd')  
  
    # Baca file Excel dengan engine yang sesuai  
    else:  
        data = pd.read_html(io.BytesIO(file_path))  
        data = data[0]
```

Gambar 2 Code Load Data Excel

Setelah data diekstraksi, tahap berikutnya adalah pemetaan dan pembersihan data. Proses pemetaan data mencakup konversi data mentah ke dalam format yang seragam dan konsisten. Pembersihan data bertujuan untuk menghilangkan kesalahan atau anomali yang dapat mempengaruhi kualitas hasil visualisasi. Proses ini mencakup pengecekan format data, konsistensi antar kolom, serta validasi nilai untuk memastikan bahwa data siap digunakan dalam visualisasi. Berikut adalah proses pemetaan data yang digambarkan Gambar 3:

```
for col in data.columns[1:]:  
    # if data[col].dtype == 'object': # Check if column type is object (usually string)  
    # data[col] = data[col].replace('-',100) # Menampilkan 0  
  
    # jika semua elemen kolom adalah string  
    if data[col].apply(lambda x: isinstance(x, str)).all():  
        data[col] = data[col].str.replace(  
            ',', '', regex=False) # Remove thousands separator  
        # Replace decimal comma with dot  
        data[col] = data[col].str.replace('.', '', regex=False)  
        data[col] = data[col].str.replace(' ', '') # Remove all spaces  
  
        try:  
            # if data[col] != '-':  
            data[col] = pd.to_numeric(  
                data[col], errors='coerce') # Convert to numeric  
        except ValueError as e:  
            st.error(f"Error converting data in column {col}: {e}")  
    else:  
        try:  
            # Convert to numeric, setting invalid parsing as NaN  
            data[col] = pd.to_numeric(data[col], errors='coerce')  
        except ValueError as e:  
            st.error(f"Error converting data in column {col}: {e}")
```

Gambar 3 Code Map Data

Setelah proses pemetaan dan pembersihan data, data yang telah dibersihkan disimpan di repositori proyek. Pengelolaan data yang terstruktur dengan baik memudahkan akses dan penggunaan data untuk visualisasi lebih lanjut. Dengan menggunakan repositori proyek, data dapat diakses dan diubah dengan mudah sesuai kebutuhan. Data yang telah disiapkan dan disimpan di repositori ini memastikan bahwa data yang digunakan di platform Streamlit selalu relevan dan terkini.

Platform Streamlit digunakan untuk membuat visualisasi data yang dinamis dan interaktif. Dengan menggunakan berbagai pustaka visualisasi data seperti Matplotlib dan Seaborn, peneliti dapat membuat grafik seperti diagram batang, garis, dan pie chart. Interaksi pengguna dengan visualisasi memungkinkan mereka memilih rentang waktu, kategori data, dan jenis grafik yang diinginkan, untuk mendapatkan wawasan yang lebih mendalam.

d. Menyimpan dan Membagikan Laporan

Setelah visualisasi data selesai dibuat, langkah berikutnya adalah menyimpan dan membagikan laporan hasil visualisasi. Dashboard yang telah dibuat akan diintegrasikan dengan aplikasi mobile "Dakocantik" untuk mempermudah akses pengguna. Proses ini dimulai dengan mengunggah proyek ke GitHub untuk memastikan semua file tersedia secara publik. Selanjutnya, dashboard dideploy menggunakan Streamlit Community Cloud untuk memungkinkan akses publik. Link yang dihasilkan dari deploy ini akan diintegrasikan ke dalam aplikasi "Dakocantik," memungkinkan pengguna untuk mengakses data dan visualisasi secara langsung melalui perangkat mobile.

e. Integrasi ke Aplikasi Mobile

Setelah data divisualisasikan menggunakan Streamlit dan dideploy ke Streamlit Community Cloud, langkah berikutnya adalah mengintegrasikan visualisasi data ke aplikasi mobile "Dakocantik." Proses ini melibatkan pengambilan link publik dari dashboard Streamlit dan menampilkannya dalam antarmuka pengguna di aplikasi mobile. Aplikasi "Dakocantik," yang dibangun dengan platform Kodular, menyediakan antarmuka yang ramah pengguna, yang memungkinkan pengguna untuk mengakses visualisasi data secara langsung. Dengan integrasi ini, informasi yang relevan dapat diakses secara dinamis oleh pemangku kepentingan.

f. Pengujian Dashboard dan Pengujian Integrasi Aplikasi

Pengujian dilakukan untuk memastikan bahwa dashboard visualisasi data yang telah dibuat dapat berfungsi dengan baik dan terintegrasi dengan aplikasi "Dakocantik." Pengujian dilakukan dalam beberapa tahap, termasuk pengujian untuk mengunggah data, menampilkan data dalam bentuk grafik interaktif, dan menguji integrasi antara dashboard dan aplikasi mobile.

### 3. Hasil dan Pembahasan

#### 3.1 Hasil Proses Visualisasi Data

Proses visualisasi data dimulai dengan langkah-langkah sesuai diagram alir untuk membuat dashboard menggunakan Streamlit. Langkah-langkah tersebut meliputi identifikasi sumber data, ekstraksi, pemetaan, pembersihan, unggah ke server, representasi grafis, serta menyimpan dan membagikan laporan.

##### 3.1.1 Proses Ekstraksi Data

a. Penentuan Engine Pembacaan:

```
def load_data_pdrb(file_path):  
    # Tentukan engine berdasarkan ekstensi file  
    if file_path.endswith('.xlsx'):  
        data = pd.read_excel(file_path, engine='openpyxl')  
    elif file_path.endswith('.xls'):  
        data = pd.read_excel(file_path, engine='xlrd')  
  
    # Baca file Excel dengan engine yang sesuai  
    else:  
        data = pd.read_html(io.BytesIO(file_path))  
        data = data[0]
```

Gambar 4 Code Engine Pembacaan

Gambar 4 menunjukkan fungsi untuk menentukan jenis file berdasarkan ekstensi yang diberikan. Jika file berformat .xlsx, maka digunakan engine openpyxl, sementara untuk format .xls, digunakan engine xlrd. dan untuk file HTML, fungsi (io.BytesIO(file\_path)) diterapkan.

b. Pembacaan Data:

```
def load_data_pdrb(file_path):  
    # Tentukan engine berdasarkan ekstensi file  
    if file_path.endswith('.xlsx'):  
        data = pd.read_excel(file_path, engine='openpyxl')  
    elif file_path.endswith('.xls'):  
        data = pd.read_excel(file_path, engine='xlrd')  
  
    # Baca file Excel dengan engine yang sesuai  
    else:  
        data = pd.read_html(io.BytesIO(file_path))  
        data = data[0]
```

Gambar 5 Code Pembacaan Data

Setelah menentukan engine, data dibaca dari file menggunakan fungsi yang sesuai. Gambar 5 menjelaskan untuk file Excel, fungsi pd.read\_excel digunakan, dan untuk file HTML, fungsi pd.read\_html(io.BytesIO(file\_path)) diterapkan.

c. Pembersihan dan Konversi Data

```
for col in data.columns[1:]:  
    # if data[col].dtype == 'object': # Check if column type is object (usually string)  
    # data[col] = data[col].replace('-',100) # Menampilkan 0  
  
    # jika semua elemen kolom adalah string  
    if data[col].apply(lambda x: isinstance(x, str)).all():  
        data[col] = data[col].str.replace(  
            ',', '', regex=False) # Remove thousands separator  
        # Replace decimal comma with dot  
        data[col] = data[col].str.replace('.', '', regex=False)  
        data[col] = data[col].str.replace(' ', '') # Remove all spaces  
  
        try:  
            # if data[col] != '-':  
            data[col] = pd.to_numeric(  
                data[col], errors='coerce') # Convert to numeric  
        except ValueError as e:  
            st.error(f"Error converting data in column {col}: {e}")  
    else:  
        try:  
            # Convert to numeric, setting invalid parsing as NaN  
            data[col] = pd.to_numeric(data[col], errors='coerce')  
        except ValueError as e:  
            st.error(f"Error converting data in column {col}: {e}")
```

Gambar 6 Code Pembersihan dan Konversi Data

Setelah data dibaca, proses pembersihan dilakukan pada setiap kolom yang terdeteksi. Gambar 6 menunjukkan jika semua elemen dalam kolom tersebut berupa string, maka langkah-langkah selanjutnya adalah menghapus pemisah ribuan, mengganti koma desimal dengan titik untuk memastikan format yang konsisten, dan menghapus spasi yang tidak perlu. Data kemudian diubah menjadi format numerik dengan menggunakan pd.to\_numeric, yang mengonversi nilai yang tidak valid menjadi NaN untuk memastikan akurasi data.

d. Penanganan Kesalahan

```
except ValueError as e:  
    st.error(f"Error converting data in column {col}: {e}")
```

Gambar 7 Code Penanganan Kesalahan

Gambar 7 menunjukkan fungsi ini juga mencakup penanganan kesalahan yang memungkinkan, yang memberikan umpan balik jika ada masalah dalam konversi data. Setelah

proses ekstraksi, data yang berhasil dikumpulkan terdiri dari tiga kategori: PDRB, kependudukan, dan Indeks Pembangunan Manusia (IPM). Total data yang diekstraksi adalah 33, terdiri dari 11 data PDRB, 9 data kependudukan, dan 13 data IPM. Data yang terkumpul siap untuk dianalisis lebih lanjut dalam tahap pembersihan dan visualisasi.

### 3.1.2 Hasil Pemetaan dan Pembersihan Data

```
# Cek judul data dan tambahkan logika pemformatan
if file_path.endswith('PDRB Pengeluaran Publikasi Softfile/Distribusi
PDRB Atas Dasar Harga Berlaku Menurut Komponen Pengeluaran Kota Mojokerto
2019-2023.xlsx'): # Ubah judul sesuai kebutuhan
    # Jika nilai desimal, kalikan dengan 100 dan format dengan simbol %
    if data[col].dtype == 'float64': # Pastikan tipe data float
        data[col] = data[col] * 100 # Ubah ke persen
        data[col] = data[col].astype(str) + '%' # Tambahkan simbol %
```

Gambar 8 Code Pemetaan dan Pembersihan Data

Gambar 8 menunjukkan logika yang melakukan pengecekan terhadap file yang diunggah untuk memastikan bahwa judul data sesuai dengan yang diharapkan. Jika file yang diunggah adalah file untuk "Distribusi PDRB Atas Dasar Harga Berlaku Menurut Komponen Pengeluaran Kota Mojokerto, 2019-2023", maka sistem melakukan pemformatan pada kolom yang bertipe data float. Dengan mengalikan nilai desimal dengan 100 dan menambahkan simbol '%', proses ini membantu dalam mengonversi data menjadi format persentase yang lebih mudah dipahami oleh pengguna.

```
# Muat data berdasarkan judul yang dipilih
if selected_data_title:
    # Ambil file path pertama
    file_path = data_files[data_type][selected_data_title][0]
    if file_path: # Pastikan file_path tidak kosong
        data = load_data_pdrb(file_path)

    # Cek apakah data berhasil dimuat
    if data is not None and not data.empty:
        st.title(f"{selected_data_title}")
```

Gambar 9 Code Memuat Data dari File yang Dipilih Oleh Pengguna

Setelah pemetaan dan pemformatan data dilakukan, tahap selanjutnya adalah pemuatan data yang telah dipetakan sebelumnya. Gambar 9 berfungsi untuk memuat data dari file yang dipilih oleh pengguna. Dengan memanfaatkan file path yang telah ditentukan, streamlit mengakses file Excel dan memuatnya ke dalam DataFrame menggunakan fungsi `load_data_pdrb(file_path)`.

### 3.1.3 Pengelolaan Data di Repository Proyek

```
# Dictionary untuk memetakan jenis data ke judul dan file path
data_files = {
    'PDRB': [
        'PDRB Atas Dasar Harga Berlaku Menurut Komponen Pengeluaran Kota
Mojokerto, 2019-2023': [
            'PDRB Pengeluaran Publikasi Softfile/PDRB Atas Dasar Harga Berlaku
Menurut Komponen Pengeluaran, Kota Mojokerto 2019-2023.xlsx',
        ],
        'Distribusi PDRB Atas Dasar Harga Berlaku Menurut Komponen Pengeluaran
Kota Mojokerto, 2019-2023': [
            'PDRB Pengeluaran Publikasi Softfile/Distribusi PDRB Atas Dasar Harga
Berlaku Menurut Komponen Pengeluaran Kota Mojokerto 2019-2023.xlsx',
        ],
    ],
}
```

Gambar 10 Code Dictionary Untuk Memetakan Jenis Data ke Judul dan File Path

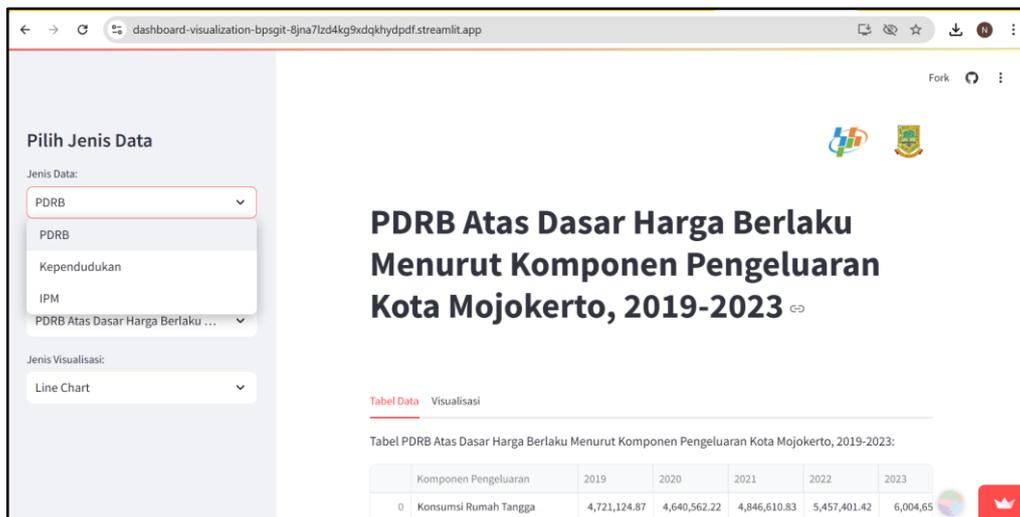
Pada tahap pengelolaan data di repositori proyek, hasil dari proses pemetaan dan pembersihan data yang telah dilakukan sebelumnya disimpan dalam bentuk file Excel di dalam folder proyek. Data yang telah dibersihkan ini diakses langsung dalam aplikasi menggunakan pustaka Python, khususnya pandas, yang memungkinkan peneliti untuk mengimpor data dari file Excel. Gambar 10, data berhasil dipetakan berdasarkan jenis data ke judul dan file path yang sesuai.

### 3.1.4 Hasil Representasi Data Secara Grafis

```
# Menampilkan dropdown jenis visualisasi tergantung pada judul data yang dipilih
# Visualisasi Data PDRB
if selected_data_title == 'PDRB Atas Dasar Harga Berlaku Menurut Komponen
Pengeluaran Kota Mojokerto, 2019-2023':
    visual_type = st.sidebar.selectbox(
        "Jenis Visualisasi:", ["Line Chart"])
elif selected_data_title == 'Distribusi PDRB Atas Dasar Harga Berlaku Menurut
Komponen Pengeluaran Kota Mojokerto, 2019-2023':
    visual_type = st.sidebar.selectbox(
        "Jenis Visualisasi:", ["Line Chart"])
elif selected_data_title == 'Laju Pertumbuhan PDRB ADHK 2010 Menurut Komponen
Pengeluaran Kota Mojokerto, 2019-2023':
    visual_type = st.sidebar.selectbox("Jenis Visualisasi:", ["Bar Chart"])
```

Gambar 11 Code Jenis Visualisasi

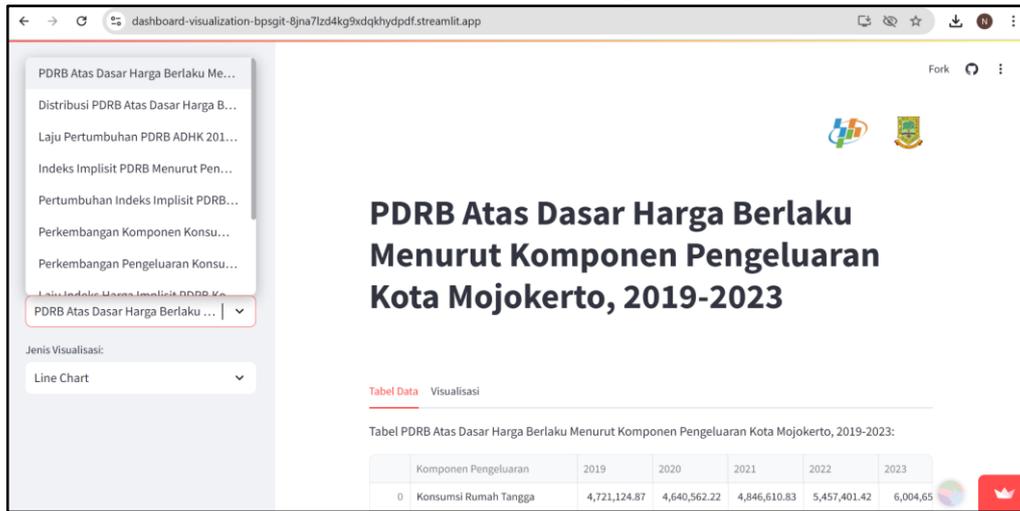
Dalam tahapan representasi data secara grafis, peneliti menggunakan kode Python untuk membuat visualisasi interaktif yang memberikan wawasan lebih dalam mengenai jenis data PDRB, kependudukan dan IPM. Kode yang digunakan berfokus pada pengambilan data dari file Excel dan penyajiannya dalam bentuk tabel dan grafik yang menarik. Gambar 11 menampilkan code mengenai pilihan dropdown untuk jenis visualisasi tergantung pada judul data yang dipilih pada masing-masing proses visualisasi data pada judul data yang telah digunakan pada dashboard.



Gambar 12 Tampilan Fitur Pilih Jenis Data

Gambar 12 menampilkan fitur pemilihan jenis data pada sidebar aplikasi visualisasi berbasis Streamlit. Sidebar ini dirancang untuk memberikan fleksibilitas kepada pengguna dalam memilih kategori data yang ingin ditampilkan pada dashboard. Menu dropdown "Pilih Jenis Data" menyediakan beberapa pilihan, seperti "PDRB," "Kependudukan," dan "IPM," yang mewakili kelompok data utama yang tersedia untuk analisis. Fungsi ini memungkinkan pengguna untuk

dengan mudah mengganti fokus analisis sesuai kebutuhan, tanpa harus meninggalkan halaman utama. Pilihan jenis data yang responsif ini memberikan pengalaman interaktif.



Gambar 13 Tampilan Fitur Pilih Judul Data

Gambar 13 menyoroti fitur pemilihan judul data yang lebih spesifik setelah kategori utama dipilih. Pada menu dropdown kedua, pengguna dapat memilih sub-kategori atau judul data tertentu yang relevan dengan kategori yang telah dipilih sebelumnya. Sebagai contoh, untuk kategori "PDRB," pengguna dapat memilih judul data seperti "PDRB Atas Dasar Harga Berlaku Menurut Komponen Pengeluaran Kota Mojokerto, 2019-2023." Fitur ini membantu pengguna menyaring dan memusatkan perhatian pada topik atau dataset tertentu yang ingin divisualisasikan, sehingga mendukung analisis mendalam dengan data yang lebih terfokus.



Gambar 14 Tampilan Fitur Jenis Visualisasi

Gambar 14 menunjukkan fitur pemilihan jenis visualisasi data yang fleksibel. Misalnya, pengguna dapat memilih line chart untuk memvisualisasikan tren PDRB Kota Mojokerto dari waktu ke waktu. Opsi ini memungkinkan pengguna untuk secara cepat mengidentifikasi pola

pertumbuhan atau penurunan PDRB serta membandingkan kontribusi masing-masing komponen pengeluaran.

Tabel PDRB Atas Dasar Harga Berlaku Menurut Komponen Pengeluaran Kota Mojokerto, 2019-2023

	Komponen Pengeluaran	2019	2020	2021	2022	2023
0	Konsumsi Rumah Tangga	4,721,124.87	4,640,562.22	4,846,610.83	5,457,401.42	6,004,65
1	Konsumsi LNPRT	59,547.93	60,735.12	63,026.04	69,224.88	75,12
2	Konsumsi Pemerintah	1,166,741.4	1,060,615.21	1,103,129.25	1,042,001.28	1,089,460
3	Pembentukan Modal Tetap Bruto	2,421,753.8	2,325,805.86	2,392,008.09	2,622,362.96	2,713,43
4	Perubahan Inventori	675.62	248.82	318.27	332.04	366
5	Ekspor Netto	-1,568,147.03	-1,499,909.52	-1,469,460.15	-1,554,298.51	-1,844,389
6	Total PDRB	6,801,696.59	6,588,057.71	6,935,632.33	7,637,024.08	8,038,657

Gambar 15 Tampilan Tab Tabel Data

Gambar 15 menyajikan tab untuk tabel data. Misalnya, pada data PDRB Kota Mojokerto dalam format yang terstruktur dan mudah dipahami. Dengan menggunakan tabel, informasi mengenai komponen pengeluaran dan perubahannya dari waktu ke waktu disajikan secara ringkas dan sistematis. Struktur baris dan kolom yang jelas memungkinkan pembaca dengan cepat membandingkan kontribusi masing-masing komponen.



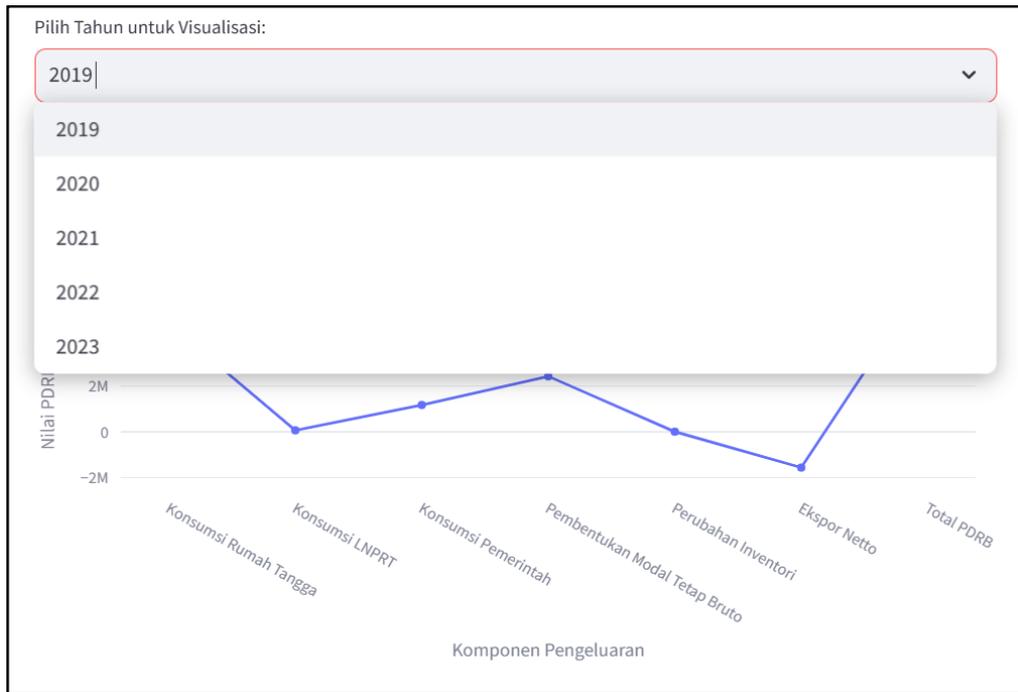
Gambar 16 Tampilan Fitur Pilih Komponen yang Ingin Divisualisasikan

Gambar 16 menunjukkan fitur dropdown "Pilih Komponen Pengeluaran" yang memungkinkan pengguna untuk secara interaktif memilih komponen spesifik dari Produk Domestik Regional Bruto (PDRB) Kota Mojokerto yang ingin mereka visualisasikan. Dengan fitur ini, pengguna dapat dengan mudah memfokuskan analisis pada komponen tertentu, seperti konsumsi rumah tangga, konsumsi pemerintah, atau pembentukan modal tetap bruto. Setelah memilih komponen, sistem akan secara otomatis menampilkan data dalam bentuk grafik garis (line chart), sehingga pengguna dapat melihat tren dan perubahan nilai komponen tersebut dari waktu ke waktu.



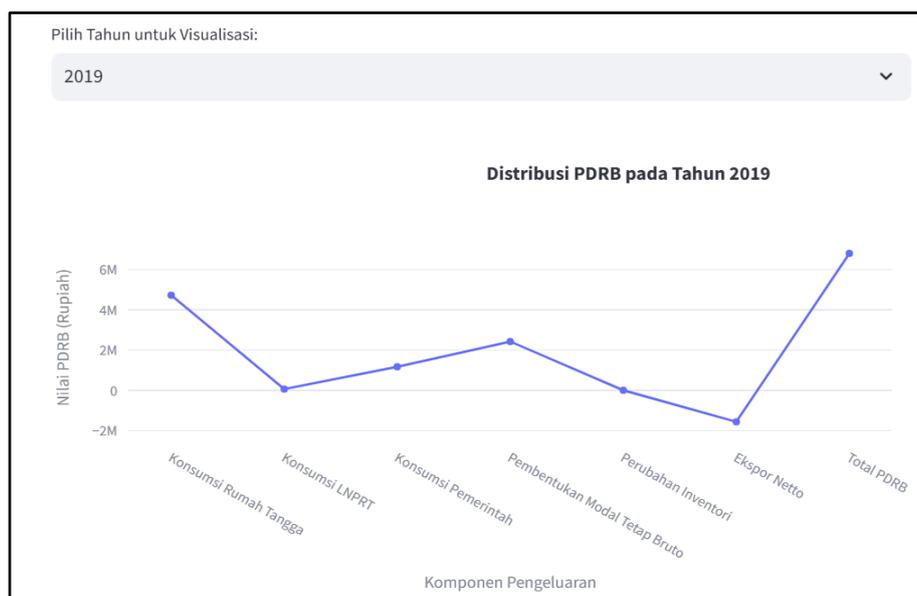
Gambar 17 Tampilan Grafik dari Komponen yang Ingin Divisualisasikan

Gambar 17 menunjukkan fitur grafik visualisasi data pada gambar tersebut memungkinkan pengguna untuk melihat secara visual tren perubahan nilai PDRB (Produk Domestik Regional Bruto) Kota Mojokerto berdasarkan komponen pengeluarannya dari tahun ke tahun. Dengan menggunakan grafik garis (line chart), pengguna dapat dengan mudah mengidentifikasi pola pertumbuhan atau penurunan pada komponen-komponen tertentu, seperti konsumsi rumah tangga atau pembentukan modal tetap bruto. Selain itu, perbandingan antara komponen-komponen ini dapat dilakukan secara visual, sehingga memudahkan dalam menganalisis kontribusi masing-masing komponen terhadap total PDRB.



Gambar 18 Tampilan Fitur Pilih Tahun yang Ingin Divisualisasikan

Gambar 18 menunjukkan fitur dropdown "Pilih Tahun untuk Visualisasi" memungkinkan pengguna untuk secara interaktif memilih tahun tertentu yang ingin mereka lihat visualisasinya. Dengan kata lain, pengguna dapat memfokuskan analisis pada data PDRB Kota Mojokerto di tahun yang spesifik. Setelah memilih tahun, grafik akan secara otomatis memperbarui tampilannya untuk menunjukkan data PDRB pada tahun yang telah dipilih. Fitur ini sangat berguna untuk membandingkan perkembangan PDRB dari tahun ke tahun dan melihat tren pertumbuhan atau penurunan pada komponen-komponen pengeluaran yang berbeda.

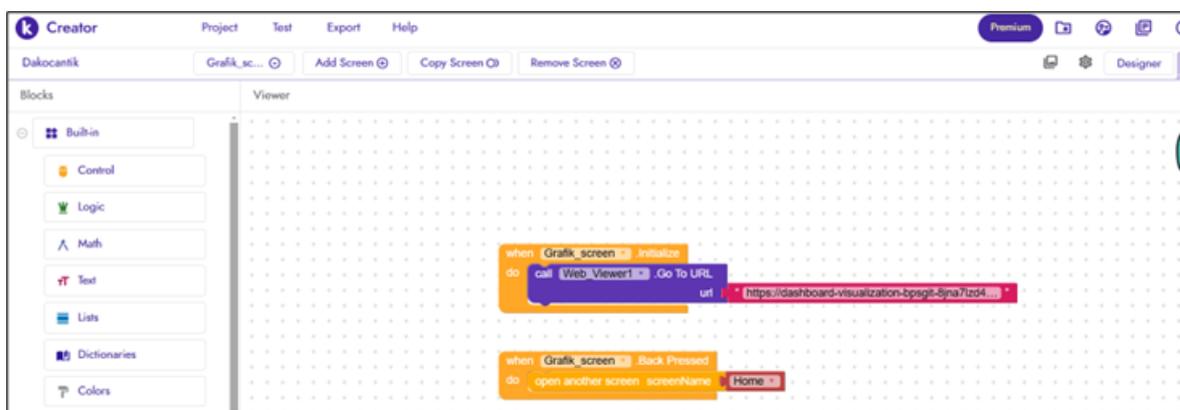


Gambar 19 Tampilan Grafik dari Tahun yang Ingin Divisualisasikan

Gambar 19 menunjukkan fitur grafik visualisasi data yang terhubung dengan dropdown "Pilih Tahun" memungkinkan pengguna untuk melihat secara visual distribusi Produk Domestik Regional Bruto (PDRB) Kota Mojokerto pada tahun tertentu. Dengan memilih tahun yang diinginkan, grafik akan secara otomatis memperbarui tampilannya untuk menampilkan kontribusi masing-masing komponen pengeluaran terhadap total PDRB pada tahun tersebut. Hal ini memungkinkan pengguna untuk menganalisis perubahan struktur ekonomi Kota Mojokerto dari waktu ke waktu dan mengidentifikasi komponen-komponen yang mengalami pertumbuhan atau penurunan yang signifikan pada tahun yang dipilih.

### 3.1.5 Menyimpan dan Membagikan Laporan

Setelah tahap pembuatan visualisasi data berhasil dilakukan, langkah berikutnya adalah menyimpan dan membagikan hasil tersebut agar dapat diakses oleh pengguna melalui aplikasi "Dakocantik." Dalam penelitian ini, proses penyimpanan dan pembagian laporan dilakukan melalui integrasi antara platform Streamlit dan aplikasi mobile "Dakocantik" yang dikembangkan menggunakan Kodular. Proses penyimpanan laporan diawali dengan mengunggah seluruh folder proyek ke GitHub. Dashboard hasil visualisasi dideploy menggunakan Streamlit Community Cloud. Platform ini menyediakan link publik yang berfungsi sebagai tautan untuk mengakses dashboard secara langsung yang ditunjukkan pada yang ada Gambar 4.245. Link publik dashboard bisa diakses melalui link ini <https://dashboard-visualization-bpsgit-8jna7lzd4kg9xdqkhydpdf.streamlit.app/>.



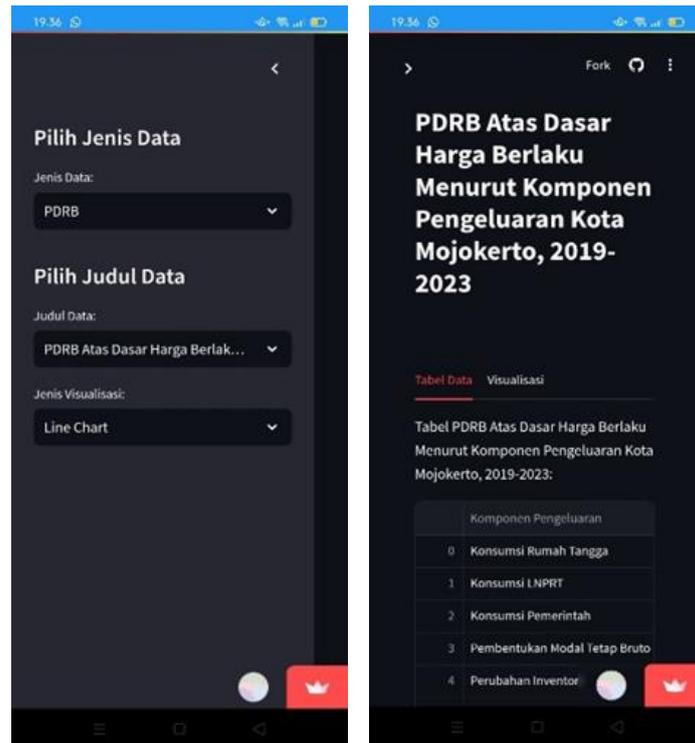
Gambar 20 Menempelkan Link Publik Dashboard ke Aplikasi Dakocantik Pada Kodular

Peneliti memasukkan tautan tersebut ke dalam platform Kodular, di mana proses integrasi dilakukan menggunakan blok pemrograman. Tautan dari *dashboard Streamlit* ditempelkan ke dalam blok pemrograman pada Kodular yang ditunjukkan Gambar 20 yang berfungsi untuk menghubungkan aplikasi "Dakocantik" dengan *dashboard* tersebut. Dengan integrasi ini, aplikasi "Dakocantik" dapat menampilkan data visualisasi dan tabel informasi secara langsung dari *dashboard Streamlit* yang telah dibuat.

### 3.2 Hasil Integrasi ke Aplikasi Mobile

Proses integrasi visualisasi data ke dalam aplikasi "Dakocantik" selesai dilakukan, hasilnya menunjukkan bahwa dashboard visualisasi berhasil diakses dengan baik melalui antarmuka aplikasi mobile. Proses ini memungkinkan pengguna untuk melihat data dan grafik

interaktif secara langsung melalui aplikasi, memberikan pengalaman yang lebih mudah dan cepat dalam memahami informasi statistik dari BPS Kota Mojokerto. Dashboard Streamlit yang telah dideploy pada Streamlit Community Cloud berhasil diintegrasikan ke aplikasi Dakocantik melalui Kodular. Hasil integrasi ini memungkinkan pengguna untuk mengakses link publik dashboard secara lancar dari aplikasi dalam bentuk web view. Dengan adanya visualisasi data dalam aplikasi mobile "Dakocantik", pengguna dapat mengakses informasi yang lebih terstruktur dan informatif, yang mendukung pengambilan keputusan yang lebih baik berdasarkan data.



Gambar 21 Tampilan Sidebar Dashboard Gambar 22 Tampilan Tabel Dashboard

Gambar 21 menunjukkan tampilan sidebar menu pada dashboard yang dapat digunakan oleh pengguna untuk memilih jenis data, memilih judul data, dan menampilkan jenis visualisasi data. Gambar 22 menunjukkan tampilan webview pada layar mobile pengguna yang mengakses dashboard. Tampilan utamanya berupa tab visualisasi data.



Gambar 23 Tampilan Grafik Dashboard

Jika pengguna mengklik menu grafik yang ada pada aplikasi, maka pengguna dapat melihat grafik dan tabel dengan jelas seperti yang ditunjukkan Gambar 23, dimana dashboard sesuai dengan yang ditampilkan di platform Streamlit, sehingga informasi yang disampaikan tetap akurat dan tidak terdistorsi pada layar aplikasi mobile.

### 3.3 Pengujian Dashboard dan Integrasi ke Aplikasi Dakocantik

Pengujian dashboard dan integrasinya ke dalam aplikasi Dakocantik dilakukan untuk memastikan bahwa semua fungsi berjalan sesuai dengan kebutuhan pengguna dan tidak menimbulkan kendala teknis. Proses ini mencakup pengujian pada setiap alur bisnis, mulai dari pengunggahan data hingga visualisasi interaktif, serta integrasi antara dashboard yang dibuat menggunakan Streamlit dengan aplikasi Dakocantik. berikut ini merupakan tabel pengujian yang dilakukan oleh penulis.

Tabel 1. Tabel Proses Pengujian Mengunggah dan Memetakan Data dari Excel ke Streamlit

No	Skenario	Hasil yang Diharapkan	Hasil Sebenarnya	Kesimpulan
1.	Normal flow	Sistem dapat memuat dan memetakan data dari file Excel secara otomatis ke dashboard.	Data dari file Excel berhasil dimuat dan dipetakan tanpa kendala. Sesuai	Sesuai
2.	Alternate flow	Jika format file tidak sesuai atau data rusak, sistem menampilkan pesan error "Data tidak dapat dimuat."	Pesan error "Data tidak dapat dimuat" muncul saat format file tidak sesuai.	Sesuai

Tabel 2. Tabel Proses Pengujian Menampilkan dan Memilih Jenis Data di Dashboard Melalui Dropdown di Sidebar

No	Skenario	Hasil yang Diharapkan	Hasil Sebenarnya	Kesimpulan
1.	Normal flow	Pengguna dapat memilih jenis data tertentu melalui dropdown, dan data yang sesuai ditampilkan di dashboard.	Data yang dipilih berhasil ditampilkan sesuai dengan pilihan dropdown.	Sesuai
2.	Alternate flow	Jika data tidak ada, sistem menampilkan pesan "Data tidak tersedia."	Pesan "Data tidak tersedia" muncul ketika memilih data yang tidak ada.	Sesuai

Tabel 3. Tabel Proses Pengujian Menampilkan Visualisasi Data di Dashboard Dalam Bentuk Tabel dan Grafik Interaktif di Dashboard

No	Skenario	Hasil yang Diharapkan	Hasil Sebenarnya	Kesimpulan
1.	Normal flow	Sistem menampilkan tabel dan grafik interaktif sesuai data yang dipilih pengguna melalui dropdown.	Tabel dan grafik interaktif muncul dengan benar sesuai pilihan pengguna.	Sesuai
2.	Alternate flow	Jika data tidak lengkap, sistem menampilkan pesan "Data tidak cukup untuk visualisasi."	Pesan "Data tidak cukup untuk visualisasi" muncul ketika data tidak lengkap.	Sesuai

Tabel 4. Tabel Integrasi Dashboard Visualisasi Data di Streamlit dengan Aplikasi Mobile Dakocantik

No	Skenario	Hasil yang Diharapkan	Hasil Sebenarnya	Kesimpulan
1.	Normal flow	Dashboard visualisasi data terbuka di aplikasi Dakocantik saat pengguna mengakses menu Grafik.	Menu Grafik di aplikasi Dakocantik membuka dashboard visualisasi dengan baik, tampilan dapat dilihat dan diakses.	Sesuai
2.	Verifikasi URL	URL dashboard yang dideploy di Streamlit Community Cloud dapat diakses dari aplikasi Dakocantik.	URL dashboard dapat diakses melalui aplikasi, tanpa hambatan atau kesalahan koneksi.	Sesuai
3	Error handling	Jika URL dashboard tidak aktif atau terjadi gangguan jaringan, aplikasi menampilkan pesan kesalahan "Network Error".	Aplikasi menampilkan pesan kesalahan saat URL tidak aktif atau saat jaringan tidak tersedia.	Sesuai
4	Alternate flow	Jika menu Grafik gagal merespons, aplikasi tetap stabil dan tidak mengalami crash.	Aplikasi tetap stabil saat menu Grafik gagal merespons, tidak ada crash atau pengaruh ke fungsi lainnya.	Sesuai
5	Interaksi dengan Fitur	Pengguna dapat berinteraksi dengan fitur dropdown pada dashboard untuk memilih data yang diinginkan dari aplikasi Dakocantik.	Pengguna berhasil memilih data yang diinginkan menggunakan dropdown tanpa kendala di aplikasi Dakocantik.	Sesuai

#### 4. Kesimpulan

Penelitian ini berhasil memenuhi tujuan yang telah ditetapkan, yaitu mengembangkan dashboard berbasis Streamlit, mengintegrasikannya dengan aplikasi mobile Dakocantik, dan memungkinkan pembaruan visualisasi data secara berkala. Berdasarkan hasil yang diperoleh, kesimpulan yang dapat ditarik adalah sebagai berikut:

Pembuatan Dashboard Interaktif Penelitian ini berhasil menciptakan dashboard berbasis Streamlit yang menyajikan data statistik BPS Kota Mojokerto secara interaktif dan informatif, menggantikan format statis yang sebelumnya digunakan. Hal ini sesuai dengan tujuan awal untuk meningkatkan efektivitas visualisasi data.

Integrasi dengan Aplikasi Mobile Dakocantik Dashboard Streamlit berhasil diintegrasikan dengan aplikasi mobile Dakocantik, yang dikembangkan dengan Kodular, sehingga meningkatkan efektivitas visualisasi dan akses data bagi pengguna. Pembaruan Visualisasi Data Aplikasi Dakocantik kini dapat memperbarui visualisasi data secara otomatis

melalui integrasi dengan dashboard Streamlit, memungkinkan masyarakat untuk mengakses data statistik yang terkini dan relevan. Secara keseluruhan, penelitian ini telah berhasil menyediakan solusi interaktif yang efektif untuk menyajikan data statistik, meningkatkan aksesibilitas, dan mendukung pengambilan keputusan yang lebih baik di BPS Kota Mojokerto.

### **Daftar Pustaka**

- Badan Pusat Statistik Kota Mojokerto. 2023. Kota Mojokerto dalam Angka. Badan Pusat Statistik Kota Mojokerto. Diakses pada 9 Juni 2024, dari <https://mojokertokota.bps.go.id/>
- Huang, B., Wang, X., & Liu, Y. 2019. *Streamlit: A flexible and scalable framework for building data science applications*. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining* (pp. 2334-2342). <https://doi.org/10.1145/3292500.3330721>
- Meliyana, D., & Latifah, K. 2022, December 15. *Pengelolaan dan visualisasi data pada Dinas Komunikasi dan Informatika Kabupaten Pati*. In *Science and Engineering National Seminar 7 (SENS 7)*, Semarang, Indonesia: Universitas PGRI Semarang.
- Shen, J., Lee, H., & Park, S. 2020. *Real-time data streaming with Streamlit*. arXiv preprint arXiv:2005.14040. Retrieved from <https://arxiv.org/abs/2005.14040>
- Stevens, R., Smith, J., & Johnson, M. 2019. Streamlit: A Python package for building interactive data science applications. *Journal of Open Source Software*, 4(40), 1042. <https://doi.org/10.21105/joss.01042>