

BAB 4

HASIL PENELITIAN

4.1. RINGKASAN HASIL PENELITIAN

Penelitian ini menghasilkan sebuah sistem dashboard berbasis web yang dikembangkan menggunakan framework Python Flask dengan database MongoDB. Sistem ini dirancang untuk membantu UMKM, khususnya akun Instagram @designbisniskita, dalam melakukan monitoring dan evaluasi performa konten serta kampanye iklan digital melalui platform Instagram dan Meta Ads.

Dashboard ini menyajikan data kuantitatif seperti *engagement*, *engagement rate*, *reach*, *impression*, *CTR*, *CPL*, dan *ROAS* dalam bentuk grafik interaktif, tabel terstruktur, serta *insight* otomatis berbasis rule. Dengan adanya sistem ini, pengguna tidak hanya dapat melihat data mentah, tetapi juga menerima interpretasi otomatis berupa rekomendasi strategi konten dan iklan.

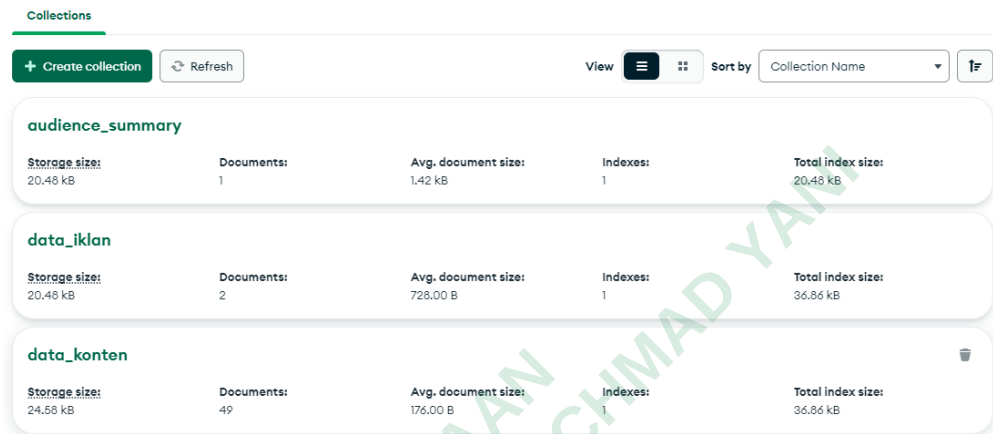
Beberapa komponen teknis dari sistem ini meliputi:

1. **Arsitektur Sistem:** Dashboard dibangun dalam struktur *MVC (Model-View-Controller)* menggunakan Flask, dengan pemisahan antara logika backend, pengolahan data, dan tampilan antarmuka pengguna.

```
/dashboard-ig/  
├── app.py  
├── templates/  
│   ├── base.html  
│   ├── dashboard.html  
│   ├── konten.html  
│   └── iklan.html  
├── static/  
│   ├── style.css  
│   └── charts.js  
├── db/  
└── connection.py
```

Gambar 4.1 Arsitektur Sistem

2. **Struktur Database:** Data disimpan dalam MongoDB menggunakan format JSON. Koleksi utama terdiri dari `data_konten`, `data_iklan`, dan `audience_summary`, yang masing-masing menyimpan informasi performa konten, kampanye iklan, dan statistik audiens.



Collection Name	Documents	Avg. document size	Indexes	Total index size
<code>audience_summary</code>	1	1.42 kB	1	20.48 kB
<code>data_iklan</code>	2	728.00 B	1	36.86 kB
<code>data_konten</code>	49	176.00 B	1	36.86 kB

Gambar 4.2 Collections database

3. **Pengolahan Data:** Data diolah secara manual dari file *insight* Instagram dan Meta Ads, lalu dimasukkan ke MongoDB. Selanjutnya, data ini diakses oleh aplikasi Flask dan divisualisasikan menggunakan Chart.js.
4. **Insight Otomatis:** Sistem dilengkapi fitur rule-based engine sederhana yang membaca pola dari data untuk menghasilkan *insight* otomatis. Misalnya, sistem dapat mendeteksi *engagement rate* tertinggi dan memberikan rekomendasi konten atau saran waktu posting.
5. **Tampilan Sistem:** Terdapat tiga halaman utama, yaitu halaman Dashboard, Konten, dan Iklan, yang masing-masing menampilkan ringkasan performa, tren visualisasi, serta *insight* relevan terhadap data yang tersedia.

4.2. KODE, TAMPILAN DAN FITUR DASHBOARD

Sistem dashboard yang dikembangkan memiliki tiga tampilan utama, yaitu Dashboard Utama, Halaman Konten, dan Halaman Iklan. Masing-masing tampilan dilengkapi dengan fitur interaktif berbasis visualisasi data dan *insight* otomatis. Berikut adalah penjabaran fitur serta cuplikan kode yang mendukung tampilan sistem:

4.2.1. Dashboard Utama

Dashboard utama merupakan halaman awal yang ditampilkan ketika pengguna mengakses sistem analisis performa akun Instagram DesignBisnisKita. Halaman ini menyajikan ringkasan performa keseluruhan yang mencakup statistik agregat, visualisasi data *insight*, dan analisis audiens, guna memberikan gambaran cepat dan menyeluruh terhadap aktivitas akun.

Pada subbab ini ditampilkan implementasi dari sisi backend melalui file `app.py` serta komponen-komponen visual frontend pada `dashboard.html` yang terdiri dari beberapa bagian utama, yaitu: Ringkasan metrik performa (*reach*, *engagement*, *impression*), Visualisasi statistik agregat, Informasi audiens (pengikut, gender, usia, kota), Visualisasi waktu aktif audiens, Grafik distribusi usia dan lokasi, *Insight* otomatis berbasis data audiens, Penjelasan berikut disusun secara berurutan dimulai dari logika *backend* (*controller*) hingga antarmuka pengguna (*frontend*). Berikut penggunaan kode programnya (Kode Program 4.1).

Kode Program 4.1 kode program python dashboard utama

```

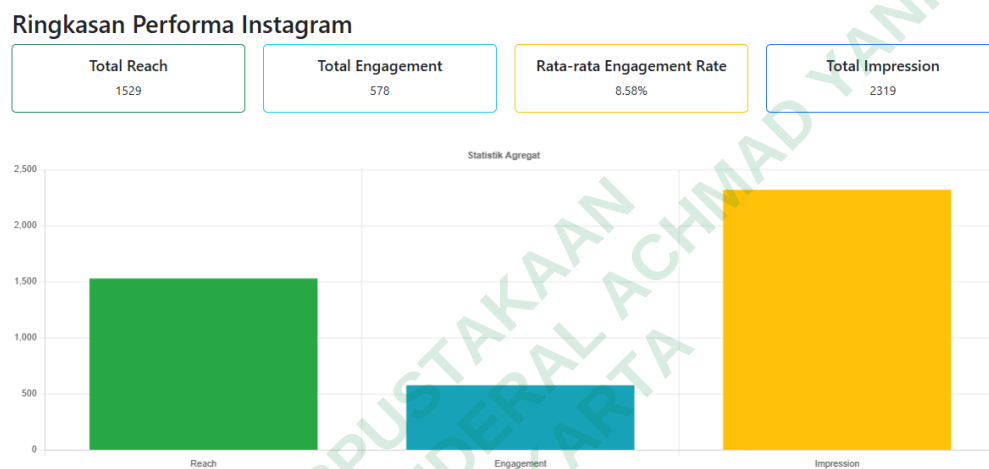
<!--Dashboard Utama -->
1  @app.route('/')
2  def dashboard():
3      konten = db.data_konten.find()
4      audience = db.audience_summary.find_one(sort=[('_id', -
5  1)])
6      # Hitung metrik performa konten
7      ...
8      # Generate insight otomatis
9      insight_audience = generate_insight_audience(audience)
10     insight_dashboard = generate_insight_dashboard(...)
11
     return render_template('dashboard.html', ...)

```

Fungsi `dashboard()` digunakan untuk menampilkan halaman utama dari dashboard. Pada fungsi ini, sistem mengambil data performa konten dan data audiens dari koleksi yang tersimpan di MongoDB. Setelah itu, sistem menghitung metrik agregat seperti *reach*, *engagement*, *impression*, dan *engagement rate*. *Insight* otomatis kemudian dihasilkan melalui pemanggilan fungsi

`generate_insight_audience()` dan `generate_insight_dashboard()`. Hasil akhir dari proses ini ditampilkan pada template `dashboard.html` dalam bentuk visualisasi dan ringkasan data.

Tampilan Ringkasan Performa Instagram menyajikan informasi ringkasan performa akun Instagram secara menyeluruh, termasuk: Total *Reach*, Total *Engagement*, Rata-rata *Engagement Rate* Total *Impression*. Ditampilkan dalam bentuk card statistik dan bar chart menggunakan Chart.js.



Gambar 4.3 tampilan ringkasan performa instagram

Ringkasan Performa Instagram ini bertujuan untuk menampilkan data agregat performa akun Instagram DesignBisnisKita dari tanggal 1 April sampai 15 Juni dalam bentuk: total *reach* (jangkauan), total *engagement* (interaksi), rata-rata *engagement rate* (persentase interaksi), total *impression* (tayangan), visualisasi perbandingan ketiganya dalam bentuk grafik batang (*bar chart*). Data diambil dari koleksi `data_konten` di MongoDB yang berisi semua postingan Instagram. Setiap dokumen memuat *reach*, *engagement*, *impression*, dan *engagement_rate*.

kode programnya pada (Kode Program 4.2):

Kode Program 4.2 tampilan ringkasan performa instagram

```

<!-- Ringkasan performa -->
1 <p>{{ total_reach }}</p>
2 <p>{{ total_engagement }}</p>
3 <p>{{ avg_engagement_rate }}%</p>
4 <p>{{ total_impression }}</p>
5 <!-- Grafik ringkasan -->
6 <canvas id="summaryChart"></canvas>
7 <script>
8     new Chart(document.getElementById('summaryChart'), {
9         type: 'bar',
10        data: {
11            labels: ['Reach', 'Engagement', 'Impression'],
12            datasets: [{
13                data: [{ total_reach }], [ total_engagement ], [
14 total_impression ]}]
15        }
16    });
17 </script>
18

```

Kode di atas menampilkan data numerik agregat dalam bentuk teks dan grafik. Variabel berasal dari backend Flask. Grafik batang dibangun menggunakan Chart.js untuk memperjelas perbandingan antar metrik.

Tampilan Ringkasan Audiens menyajikan informasi ringkasan *audiens* secara menyeluruh, termasuk: total pengikut, rasio gender, usia dominan, kota terbanyak, jam tayang terbaik.

Ringkasan Audiens



Gambar 4.4 tampilan ringkasan audiens

Ringkasan *Audiens* ini digunakan untuk menganalisis karakteristik demografi dan perilaku pengikut akun Instagram selama periode 1 April – 15 Juni, sebagai dasar

pengambilan keputusan strategi konten dan waktu tayang. Berikut adalah elemen-elemen yang ditampilkan beserta artinya:

Tabel 4.1 Komponen penjelasan ringkasan *audiens*

NO	Komponen	Penjelasan
1	Total Pengikut	Jumlah pengikut akun Instagram (dalam contoh: 390) dan pertumbuhan dalam persen (+2.4% dibandingkan 31 Maret).
2	Rasio Gender	Persentase pengikut berdasarkan jenis kelamin: Laki-laki 74%, Perempuan 26%.
3	Usia Dominan	Kelompok umur terbesar: 25–34 tahun sebanyak 67.8%.
4	Kota Terbanyak	Lokasi kota dengan jumlah pengikut tertinggi: Jakarta (10.6%).
5	Jam Tayang Terbaik	Hari dan jam paling aktif: Kamis, Jumat pukul 18.00–21.00.

Data diambil dari koleksi data `audience` di MongoDB yang berisi semua data terkait audience seperti pengikut, jenis kelamin, rentang usia, lokasi teratas, waktu aktif, aktivitas harian, jam slot, interaksi, tayangan berikut kode programnya (Kode Program 4.3):

Kode Program 4.3 ringkasan audiens

```

<!-- Ringkasan Audiens -->
1 <p>Total Pengikut: {{ audience.pengikut.total }} (+{{
2 audience.pengikut.pertumbuhan }}%)</p>
3 <p>Rasio Gender: L {{ audience.jenis_kelamin.laki_laki }}% /
4 P {{ audience.jenis_kelamin.perempuan }}%</p>
5 <p>Usia Dominan: {{ usia_dom[0] }} ({{ usia_dom[1] }}%)</p>
6 <p>Kota Terbanyak: {{ audience.lokasi_teratas[0].kota }} ({{
7 audience.lokasi_teratas[0].persentase }}%)</p>
8 <p>Jam Tayang Terbaik: Hari - {{
9 audience.waktu_aktif.hari_paling_aktif | join(', ') }}, Jam
10 - {{ audience.waktu_aktif.jam_puncak | join(' - ') }}</p>

```

Bagian ini menampilkan statistik audiens seperti total pengikut, proporsi gender,

usia dominan, kota terbanyak, serta waktu aktif audiens. Data digunakan sebagai dasar pengambilan keputusan dalam penjadwalan konten dan penargetan iklan. Tampilan grafik aktivitas *audiens* berdasarkan hari & jam menyajikan informasi ringkasan *audiens* berdasarkan hari dan jam tayang mereka.

Grafik Aktivitas Audiens Berdasarkan Hari & Jam



Gambar 4.5 grafik aktivitas hari dan jam

Pada bagian ini, sistem menyajikan grafik batang yang merepresentasikan tingkat aktivitas audiens pada setiap hari dalam satu minggu, dibagi ke dalam delapan slot waktu: 12a (pukul 00.00), 3a (pukul 03.00), 6a (pukul 06.00), 9a (pukul 09.00), 12p (pukul 12.00), 3p (pukul 15.00), 6p (pukul 18.00), 9p (pukul 21.00).

Grafik ini bertujuan untuk membantu pengguna memahami kapan waktu terbaik untuk mengunggah konten, berdasarkan tingkat keaktifan pengikut akun.

Berdasarkan grafik yang ditampilkan, pola aktivitas tertinggi secara konsisten terjadi pada pukul 15.00 – 21.00, terutama pada hari Kamis dan Jumat. Hal ini sejalan dengan *insight* sistem yang merekomendasikan waktu tayang terbaik pada slot tersebut. Setiap grafik disusun menggunakan library Chart.js dengan data yang diambil dari field aktivitas_harian dan jam_slot dalam koleksi audience_summary. Kode programnya adalah (Kode Program 4.4) berdasarkan hari & jam:

Kode Program 4.4 grafik aktivitas audiens

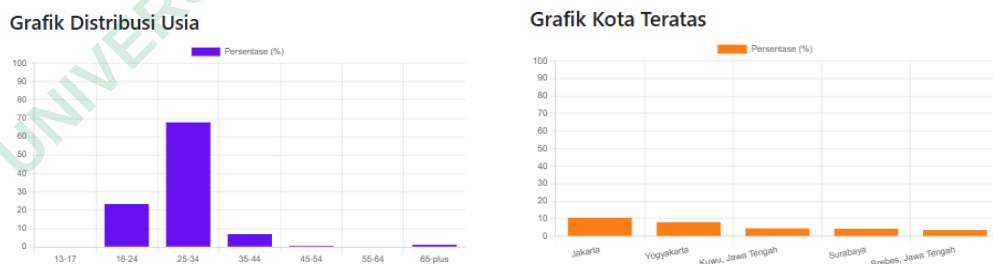
```
dashboard.html
```

```

1  <!-- Grafik Aktivitas Harian Audiens -->
2  {% for hari, data in audience.aktivitas_harian.items() %}
3      <h6>{{ hari }}</h6>
4      <canvas id="chart_{{ loop.index }}"></canvas>
5      <script>
6          new Chart(document.getElementById('chart_{{ loop.index
7      }}'), {
8          type: 'bar',
9          data: {
10             labels: {{ audience.jam_slot | tojson }},
11             datasets: [{
12                 data: {{ data | tojson }},
13                 backgroundColor: '#0dcaf0'
14             }]
15         }
16     });
17 </script>
18 {% endfor %}

```

Tampilan grafik distribusi usia dan grafik kota teratas memberikan wawasan penting terkait demografi pengikut akun Instagram. Visualisasi ini dirancang untuk membantu pemilik akun memahami karakteristik utama audiens, baik dari segi rentang usia maupun lokasi geografis, sehingga strategi konten dan kampanye pemasaran dapat disesuaikan secara lebih tepat sasaran



Gambar 4.6 grafik distribusi usia dan grafik kota teratas

Pada grafik distribusi usia, terlihat bahwa kelompok usia 25–34 tahun mendominasi dengan persentase sebesar 67,8%, menjadikannya sebagai target utama dalam penyusunan konten pemasaran. Kelompok usia 18–24 tahun berada di posisi kedua dengan 23,4%, sedangkan kelompok usia lainnya seperti 35–44 tahun (7%), 65+

(1,2%), dan usia lainnya memiliki persentase yang sangat kecil atau mendekati nol. Grafik ini dibangun menggunakan Chart.js dan mengambil data dari field `rentang_usia` pada koleksi `audience_summary`. Visualisasi ini membantu pengguna memahami segmentasi umur pengikut secara langsung.

Sementara itu, grafik kota teratas menyajikan informasi mengenai lokasi asal pengikut terbanyak. Grafik ini memberikan insight penting untuk kampanye atau promosi yang bersifat lokal. Dari data yang divisualisasikan, diketahui bahwa Jakarta merupakan kota dengan jumlah pengikut terbanyak (10,6%), diikuti oleh Yogyakarta (8%), Kuwu, Jawa Tengah (4,5%), Surabaya (4,3%), dan Brebes, Jawa Tengah (3,6%). Grafik ini juga menggunakan Chart.js dan menarik datanya dari field `lokasi_teratas` pada koleksi `audience_summary`.

Kedua grafik ini ditampilkan secara berdampingan pada halaman dashboard, sehingga memudahkan pengguna untuk membandingkan dua aspek demografi utama: usia dan lokasi, Kode Program (Kode Program 4.5):

Kode Program 4.5 grafik distribusi usia da grafik kota teratas

```

<!-- Grafik Distribusi Usia -->
1  <canvas id="umurChart"></canvas>
2  <script>
3      new Chart(document.getElementById('umurChart'), {
4          type: 'bar',
5          data: {
6              labels: {{ audience.rentang_usia.keys() | tojson }},
7              datasets: [{
8                  data: {{ audience.rentang_usia.values() | tojson }},
9                  backgroundColor: '#6610f2'
10             }]
11         }
12     });
13 </script>
<!-- Grafik Kota Teratas -->
14 <canvas id="kotaChart"></canvas>
15 <script>
16     new Chart(document.getElementById('kotaChart'), {
17         type: 'bar',

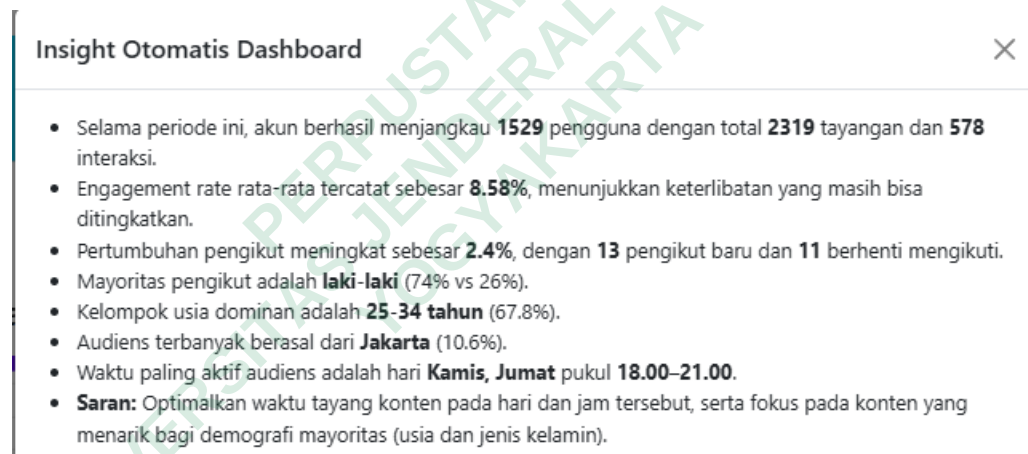
```

```

19     data: {
20         labels: {{ audience.lokasi_teratas |
21     map(attribute='kota') | tojson }},
22         datasets: [{
23             data: {{ audience.lokasi_teratas |
24     map(attribute='persentase') | tojson }},
25             backgroundColor: '#fd7e14'
26         }]
27     }
28 });
29 </script>

```

Tampilan *Insight* otomatis dashboard adalah fitur yang menyajikan ringkasan analisis secara deskriptif dan instan tanpa perlu analisis manual oleh pengguna. *Insight* ini dihasilkan langsung dari data audiens dan data konten yang tersimpan di MongoDB dan divisualisasikan dalam bentuk teks naratif.



Gambar 4.7 *insight* otomatis dashboard

Insight otomatis yang ditampilkan dalam dashboard dikembangkan tidak hanya sebagai informasi deskriptif, tetapi juga berdasarkan aturan logika (*rule-based system*). Pendekatan ini memungkinkan sistem memberikan kesimpulan berdasarkan nilai-nilai metrik kuantitatif yang telah ditentukan. Dengan adanya *rule-based system*, pengguna dapat mengetahui dasar dari setiap *insight* yang muncul secara otomatis. Berikut adalah beberapa aturan logika (*rule base*) yang digunakan dalam menghasilkan insight pada dashboard:

Tabel 4.2 rule base insight otomatis dashboard

Rule 1	IF <code>total_reach > 1000</code> AND <code>total_engagement > 500</code> THEN performa distribusi konten cukup baik	Akun berhasil menjangkau banyak audiens dengan interaksi yang tinggi
Rule 2	IF <code>avg_engagement_rate < 10</code> THEN engagement tergolong rendah	Engagement rate masih bisa ditingkatkan
Rule 3	IF <code>audience.pengikut.pertumbuhan > 0</code> THEN terjadi peningkatan pengikut	Pertumbuhan pengikut positif
Rule 4	IF <code>audience.jenis_kelamin.laki_laki ></code> <code>audience.jenis_kelamin.perempuan</code> THEN mayoritas audiens laki-laki	Konten lebih menarik bagi pria
Rule 5	IF <code>audience.rentang_usia['25_34'] ></code> <code>60</code> THEN usia dominan 25–34 tahun	Audiens utama berada pada rentang usia produktif
Rule 6	IF 'Kamis' atau 'Jumat' termasuk dalam <code>waktu_aktif</code> THEN hari tersebut adalah waktu optimal untuk tayang	Disarankan unggah konten pada Kamis atau Jumat pukul 18.00–21.00
Rule 7	IF <code>audience.lokasi_teratas[0] ==</code> 'Jakarta' THEN audiens terbesar dari Jakarta	Target distribusi konten dapat difokuskan ke wilayah tersebut

Kode program (Kode Program 4.6) *insight* otomatis dashboard:

Kode Program 4.6 *Insight* otomatis dashboard

```

<!--Generate Insight Dashboard -->
1 def generate_insight_dashboard(...):
2     if not audience:
3         return "Belum ada data lengkap."
4
5     insight = []
6     insight.append("...total reach, impression,
```

```

7 engagement.")
8     insight.append("...engagement rate tinggi atau rendah.")
9     insight.append("...pertumbuhan pengikut.")
10    insight.append("...mayoritas gender.")
11    insight.append("...kelompok usia dominan.")
12    insight.append("...kota terbanyak.")
13    insight.append("...waktu aktif audiens.")
14    insight.append("Saran strategi konten berdasarkan
15 data.")
16    return insight

```

```

<!-- Tombol dan Modal Insight Dashboard -->
1 <button data-bs-toggle="modal" data-bs-
2 target="#dashboardInsightModal">Lihat Detail
3 Insight</button>
4
5 <div class="modal" id="dashboardInsightModal">
6     <ul>
7         <li>Menampilkan total reach, impression,
8 engagement.</li>
9         <li>Engagement rate dengan interpretasi.</li>
10        <li>Pertumbuhan dan penurunan pengikut.</li>
11        <li>Dominasi gender pengikut.</li>
12        <li>Usia dominan audiens.</li>
13        <li>Kota asal pengikut terbanyak.</li>
14        <li>Hari dan jam aktif audiens.</li>
15        <li>Saran konten berdasarkan insight.</li>
16    </ul>
17 </div>

```

Fitur insight otomatis pada dashboard ini dihasilkan oleh fungsi `generate_insight_dashboard()` yang mengolah berbagai metrik agregat dan data demografi untuk menghasilkan kesimpulan dalam bentuk kalimat. Insight ini ditampilkan di halaman utama melalui modal dialog dan memberikan rekomendasi strategi konten yang relevan dengan target audiens.

4.2.2. Halaman Konten

Halaman Konten pada dashboard berfungsi untuk menampilkan performa

setiap konten Instagram secara rinci. Tujuan utama dari halaman ini adalah untuk memberikan informasi seputar konten mana yang paling efektif dalam menjangkau audiens dan mendapatkan interaksi.

Fitur-fitur yang disediakan dalam halaman ini antara lain:

1. Tabel Data Konten: Menampilkan seluruh konten yang telah dipublikasikan lengkap dengan metrik seperti *reach*, *impression*, *engagement*, dan *engagement rate*.
2. Grafik Top 10 *Engagement*: Menyajikan 10 konten dengan *engagement* tertinggi dalam bentuk grafik batang agar mudah dianalisis.
3. Grafik Tren Bulanan: Visualisasi tren metrik *reach* dan *impression* berdasarkan bulan untuk memantau fluktuasi performa.
4. *Insight* Otomatis Konten: Sistem menghasilkan ringkasan *insight* secara otomatis untuk memberikan pemahaman terhadap performa konten yang paling unggul.
5. Saran Strategis Konten: Berdasarkan data yang ada, sistem juga memberikan saran konten untuk langkah pengembangan strategi ke depan.

Seluruh fitur ini bersumber dari koleksi data_konten dalam database MongoDB dan dirancang agar pengguna dapat mengambil keputusan berbasis data (*data-driven content strategy*). Berikut kode programnya (Kode Program 4.7).

Kode Program 4.7 python halaman konten

```
@app.route('/konten')
1 def konten():
2     data = db.data_konten.find()
3
4     # Top 10 konten dengan engagement tertinggi
5     top10_labels = [...]
6     top10_engagement = [...]
7
8     # Ringkasan reach & impression bulanan
9     trend_labels = [...]
10    trend_reach = [...]
11    trend_impression = [...]
12
```

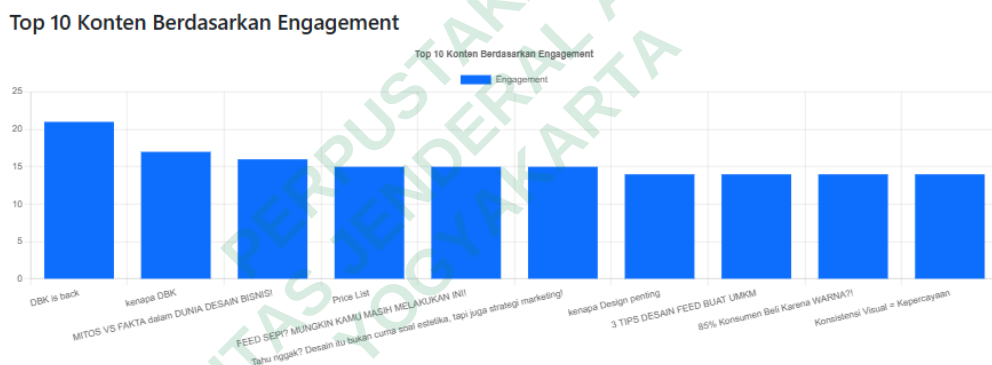
```

13     # Insight & saran konten otomatis
14     insight_konten = generate_insight_konten(data)
15     saran_konten = generate_saran_konten(data)
16
17     return render_template('konten.html', ...)

```

Fungsi ini menampilkan halaman analisis konten, mengambil data dari koleksi `data_konten`, lalu menampilkan konten dengan engagement tertinggi dan tren performa bulanan. Insight dan saran konten dihasilkan secara otomatis dan ditampilkan melalui template `konten.html`.

Tampilan top 10 konten berdasarkan *engagement* ini menyajikan 10 konten dengan jumlah *engagement* tertinggi dalam bentuk grafik batang horizontal. Grafik ini membantu pengguna mengidentifikasi konten mana yang paling berhasil dalam menjangkau dan melibatkan audiens.



Gambar 4.8 grafik top 10 konten

Metode pengambilan datanya dilakukan dengan mengurutkan seluruh konten berdasarkan nilai *engagement*, lalu mengambil 10 teratas untuk ditampilkan. Proses ini dilakukan secara otomatis di backend menggunakan Python Flask sebelum dikirimkan ke tampilan frontend menggunakan Chart.js, kode program pada (Kode Program 4.8):

Kode Program 4.8 grafik top 10 konten

```

<!--Top 10 Konten -->
1  <canvas id="topKontenChart"></canvas>
2  <script>
3      const top10Labels = {{ top10_labels | tojson }};
4      const top10Data = {{ top10_engagement | tojson }};

```

```

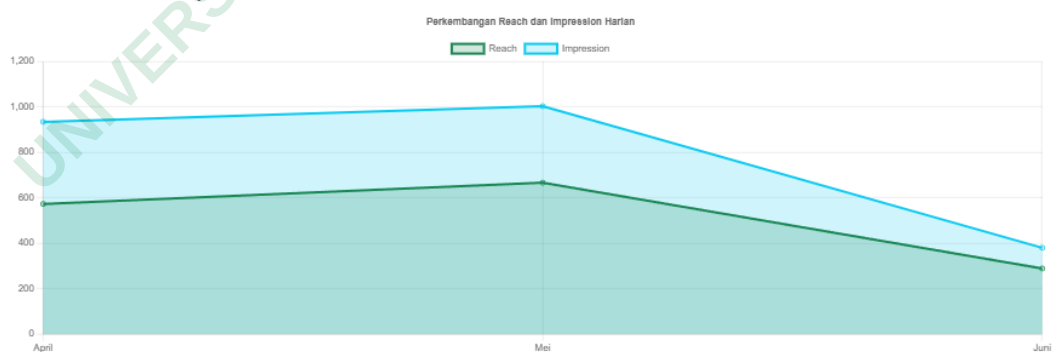
5
6     new Chart(document.getElementById('topKontenChart'), {
7         type: 'bar',
8         data: {
9             labels: top10Labels,
10            datasets: [{
11                label: 'Engagement',
12                data: top10Data,
13                backgroundColor: '#0d6efd'
14            }]
15        }
16    });
17 </script>

```

Kode ini menampilkan grafik batang (*bar chart*) yang memvisualisasikan 10 konten dengan engagement tertinggi. Data diambil dari backend Flask dalam bentuk `top10_labels` dan `top10_engagement`, lalu divisualisasikan menggunakan Chart.js untuk membantu pengguna mengidentifikasi konten yang paling efektif.

Tampilan tren *reach* dan *impression* harian ini menampilkan perbandingan perkembangan *reach* dan *impression* setiap bulan dalam bentuk grafik area. Grafik ini bersifat *time series*, artinya disusun berdasarkan waktu, dan sangat penting untuk menganalisis tren jangkauan dan impresi konten dari waktu ke waktu.

Tren Reach dan Impression Harian



Gambar 4.9 tren *reach* dan *impression* harian

Penggunaan *time series* ini memungkinkan pengguna untuk:

1. Mengamati kenaikan atau penurunan performa konten setiap bulan.
2. Melihat hubungan antara jumlah *reach* dan *impression* dalam periode April

hingga Juni.

3. Menilai efektivitas distribusi konten dalam jangka waktu tertentu.

Pada grafik, sumbu X merepresentasikan bulan, sementara sumbu Y menunjukkan jumlah *reach* dan *impression*. Dari grafik ini terlihat bahwa:

1. Impression tertinggi terjadi pada bulan Mei, mencapai hampir 1000.
2. Namun, terdapat penurunan tajam di bulan Juni, baik untuk *reach* maupun *impression*, yang dapat menjadi sinyal untuk evaluasi konten.

Kode program tren *reach* dan *impression* harian (Kode Program 4.9):

Kode Program 4.9 gtren *reach* dan *impression*

```

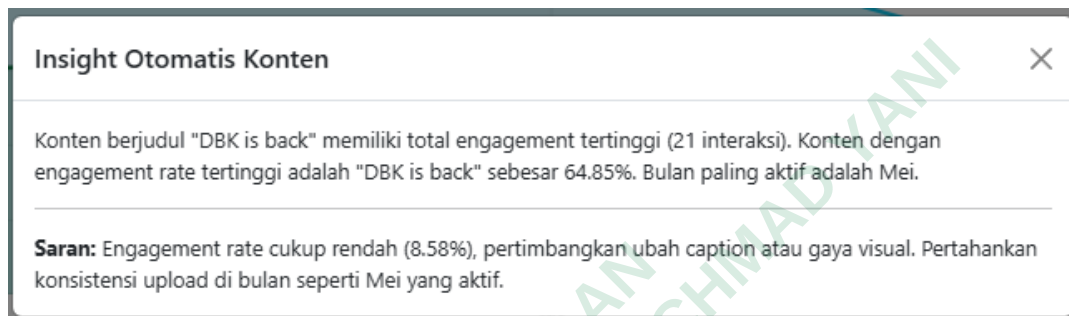
<!--Tren reach impression harian -->
1  <canvas id="trendChart"></canvas>
2  <script>
3      const trendLabels = {{ trend_labels | tojson }};
4      const trendReach = {{ trend_reach | tojson }};
5      const trendImpression = {{ trend_impression | tojson }};
6
7      new Chart(document.getElementById('trendChart'), {
8          type: 'line',
9          data: {
10             labels: trendLabels,
11             datasets: [
12                 { label: 'Reach', data: trendReach, borderColor:
13                 '#198754' },
14                 { label: 'Impression', data: trendImpression,
15                 borderColor: '#0dcaf0' }
16             ]
17         }
18     });
19 </script>

```

Kode ini digunakan untuk menampilkan grafik garis (line chart) yang menggambarkan perkembangan jumlah *reach* dan *impression* konten dari waktu ke waktu. Data disajikan dalam variabel *trend_labels*, *trend_reach*, dan *trend_impression* yang dikirim dari backend Flask. Visualisasi ini berguna untuk menganalisis tren pertumbuhan interaksi audiens terhadap konten yang

dipublikasikan.

Tampilan *insight* otomatis konten menyajikan analisis otomatis terhadap performa konten berdasarkan nilai *engagement* dan periode aktivitas pengguna. *Insight* ini dihasilkan secara dinamis menggunakan fungsi Python, dengan tujuan membantu pengguna memahami konten mana yang paling efektif serta waktu paling produktif. Dari visualisasi di atas, sistem memberikan *insight* berikut:



Gambar 4.10 *insight* otomatis konten

Berikut adalah beberapa aturan logika (*rule base*) yang digunakan dalam menghasilkan insight pada halaman konten:

Tabel 4.3 rule base insight otomatis konten

Rule 8	IF konten_engagement.tertinggi != None THEN tampilkan judul konten tersebut	Konten dengan engagement tertinggi ditampilkan sebagai highlight
Rule 9	IF konten_rate.tertinggi > 10 THEN engagement rate sangat baik	Konten tersebut efektif dalam mendorong interaksi pengguna
Rule 10	IF bulan_terbaik memiliki engagement paling tinggi THEN tampilkan sebagai bulan aktif	Bulan paling aktif berdasarkan total interaksi
Rule 11	IF rate_avg < 5 THEN perlu evaluasi strategi caption/desain	Disarankan ubah gaya konten atau copywriting
Rule	IF jumlah_upload[bulan_aktif] > rata-	Konsistensi konten

12	rata THEN konsistensi unggahan baik	mendukung performa bulanan produktif
----	-------------------------------------	--------------------------------------

Kode program *insight* otomatis konten (Kode Program 4.10):

Kode Program 4.10 *insight* otomatis konten

```

<!--Generate Insight Konten-->
1  def generate_insight_konten(data):
2      # Cari konten dengan engagement & engagement rate
3      tertinggi
4      ...
5      # Tentukan bulan paling aktif berdasarkan engagement
6      ...
7      return "... insight otomatis konten ..."
8  python
9  Copy
10 Edit
11 def generate_saran_konten(data):
12     # Hitung rata-rata engagement rate
13     ...
14     # Tentukan bulan paling sering upload
15     ...
16     return "... saran strategi konten ..."

```

```

<!-- Tombol dan Modal Insight Konten -->
1  <!-- Modal Insight Otomatis Konten -->
2  <button data-bs-toggle="modal" data-bs-
3  target="#kontenInsightModal">Lihat Insight Konten</button>
4
5  <div class="modal" id="kontenInsightModal">
6      <p>{{ insight_konten }}</p>
7      <hr>
8      <strong>Saran:</strong> {{ saran_konten }}
9  </div>

```

Fungsi `generate_insight_konten()` digunakan untuk menampilkan insight otomatis berdasarkan data konten, seperti konten dengan engagement tertinggi, engagement rate tertinggi, dan bulan paling aktif. Sementara `generate_saran_konten()` memberikan rekomendasi strategi berdasarkan rata-rata performa konten dan pola

distribusi waktu unggah. Kedua hasil ini ditampilkan secara interaktif dalam modal pada halaman konten.html.

Tampilan daftar postingan konten Instagram menampilkan seluruh data konten yang telah diunggah pada akun Instagram bisnis, disusun dalam format tabel.

Daftar Postingan Konten Instagram

Judul	Bulan	Engagement	Engagement Rate (%)	Reach	Impression
DBK is back	april	21	64.85	227	337
kenapa DBK	april	17	13.14	46	89
Price List	april	15	9.71	34	57
kenapa Design penting	april	14	6.85	24	37
MITOS VS FAKTA dalam DUNIA DESAIN BISNIS!	april	16	7.04	25	46
3 TIPS DESAIN FEED BUAT UMKM	april	14	7.04	25	37
FEED SEPI? MUNGKIN KAMU MASIH MELAKUKAN INI!	april	15	7.06	27	39
85% Konsumen Beli Karena WARNA?!	april	14	6.19	22	39
Feed Rapi = Bisnis Terpercaya	april	12	6.57	23	38
Desain itu bukan cuma soal "biar keren", tapi soal biar dipercaya.	april	13	6.78	24	37
Branding vs Desain: Banyak yang Masih Bingung!	april	13	6.47	23	41
Konsistensi Visual = Kepercayaan	april	14	5.63	20	32
Ingat: Semua brand besar juga pernah mulai dari nol.	april	12	4.22	15	33
Desain feed kamu udah rapi? Keren. Tapi apakah udah berdampak ke penjualan?	april	12	4.5	16	32
Mau Konten IG Rapi & Gak Bikin Pusing?	april	12	6.11	22	41
Warna bukan cuma soal "bagus" atau "kece" — tapi juga soal kesan pertama yang brand kamu tinggalkan.	mei	13	6.66	24	38
60% UMKM Belum Maksimalkan Desain Visual!	mei	12	6.67	24	40
Warna Ungu: Simbol Elegan, Kreatif, dan Premium!	mei	12	9.72	35	43

Gambar 4.11 daftar postingan konten instagram

Tabel ini memuat informasi penting seperti:

1. **Judul Konten:** Judul atau topik utama dari setiap unggahan.
2. **Bulan:** Waktu konten tersebut diposting (misalnya April atau Mei).
3. **Engagement:** Jumlah interaksi yang terdiri dari like, komentar, dan simpan.
4. **Engagement Rate:** Rasio interaksi dibandingkan jangkauan (dalam persentase).
5. **Reach:** Jumlah akun unik yang melihat konten tersebut.
6. **Impression:** Total tampilan dari konten tersebut, termasuk tayangan berulang.

Visualisasi tabel ini penting karena memungkinkan pengguna untuk:

1. Membandingkan performa antar konten secara langsung.
2. Menganalisis efektivitas konten berdasarkan waktu (bulan).
3. Mengidentifikasi konten dengan *engagement rate* tinggi walaupun memiliki

reach rendah, yang berpotensi dijadikan model strategi konten selanjutnya.

Sebagai contoh:

1. Konten "DBK is back" memiliki *engagement* tertinggi (21 interaksi) dan *engagement rate* sangat tinggi (64.85%), yang menunjukkan bahwa konten tersebut sangat efektif walaupun *reach*-nya hanya 227.
2. Sementara konten lain seperti "Warna Ungu: Simbol Elegan..." pada bulan Mei, juga menunjukkan *engagement rate* tinggi (9.72%) dengan performa stabil.

Tabel ini dihasilkan secara otomatis dari koleksi `data_konten` dalam database MongoDB, lalu ditampilkan menggunakan loop dalam `konten.html`.

kode program postingan konten Instagram (Kode Program 4.11):

Kode Program 4.11 postingan konten instagram:

```

<!--Tabel postingan konten-->
1  <table>
2    <thead>
3      <tr>
4        <th>Judul</th>
5        <th>Bulan</th>
6        <th>Engagement</th>
7        <th>Engagement Rate (%)</th>
8        <th>Reach</th>
9        <th>Impression</th>
10     </tr>
11  </thead>
12  <tbody>
13    {% for item in data %}
14    <tr>
15      <td>{{ item.judul }}</td>
16      <td>{{ item.month }}</td>
17      <td>{{ item.engagement }}</td>
18      <td>{{ item.engagement_rate }}</td>
19      <td>{{ item.reach }}</td>
20      <td>{{ item.impression }}</td>
21    </tr>
    {% endfor %}

```

```

22     </tbody>
23 </table>
24

```

Tabel ini menampilkan seluruh data konten Instagram yang diambil dari database `data_konten`. Setiap baris mewakili satu postingan dan menampilkan informasi penting seperti judul, bulan unggah, engagement, engagement rate, reach, dan impression. Tabel ini berfungsi sebagai tampilan data mentah yang mendukung visualisasi dan insight otomatis yang ada di bagian atas halaman.

4.2.3. Halaman Iklan

Halaman ini menyajikan data performa kampanye iklan yang dijalankan melalui platform Instagram. Data ini membantu dalam mengevaluasi efektivitas strategi iklan digital berdasarkan metrik utama seperti *ROAS*, *CTR*, dan *CPL*, kode program (Kode Program 4.12).

Kode Program 4.12 python halaman iklan

```

<!--Halaman Iklan -->
1  @app.route('/iklan')
2  def iklan():
3      data = db.data_iklan.find()
4
5      # Konversi tanggal jika masih string
6      ...
7
8      # Set default nilai metrik iklan
9      ...
10
11     # Temukan highlight metrik tertinggi & terendah
12     roas_tertinggi = ...
13     ctr_tertinggi = ...
14     cpl_terendah = ...
15
16     # Insight & saran otomatis
17     insight_text = generate_insight(data)
18     saran_text = generate_saran(data)
19
20     return render_template('iklan.html', ...)

```

Fungsi ini bertugas menampilkan halaman analisis performa iklan. Data diambil dari koleksi data_iklan, diformat ulang (terutama tanggal), lalu dihitung highlight seperti ROAS tertinggi, CTR tertinggi, dan CPL terendah. Selain itu, sistem juga menghasilkan insight dan saran otomatis berdasarkan efektivitas iklan, yang ditampilkan ke halaman iklan.html.

Tampilan highlight iklan terbaik



Gambar 4.12 highlight iklan terbaik

yaitu Tiga metrik utama disorot sebagai iklan dengan performa terbaik:

1. *ROAS Tertinggi*: Menampilkan judul iklan dengan *Return on Ads Spend* tertinggi. Meskipun dalam data ini masih menunjukkan angka 0x, metrik ini penting untuk menunjukkan efektivitas pengeluaran iklan dalam menghasilkan pendapatan.
2. *CTR Tertinggi*: Menampilkan iklan dengan *Click Through Rate* tertinggi, yaitu “Desainmu Butuh Bantuan?” dengan 3.86%, menunjukkan ketertarikan audiens terhadap iklan tersebut.
3. *CPL Terendah*: Menampilkan iklan dengan *Cost Per Lead* terendah. Iklan dengan CPL rendah (Rp 433) menunjukkan efisiensi biaya per hasil yang diperoleh.

kode program *highlight* iklan terbaik (Kode Program 4.13):

Kode Program 4.13 *highlight* iklan terbaik:

```
<!-- Highlight Iklan Terbaik -->
1 <div>
2   <p>ROAS Tertinggi: {{ roas_tertinggi.judul_iklan }} - {{
3   roas_tertinggi.hasil.roas }}x</p>
4   <p>CTR Tertinggi: {{ ctr_tertinggi.judul_iklan }} - {{
5   ctr_tertinggi.hasil.ctr }}%</p>
6   <p>CPL Terendah: {{ cpl_terendah.judul_iklan }} - Rp {{
7   cpl_terendah.hasil.cpl }}</p>
8 </div>
```

Bagian ini menampilkan tiga performa iklan terbaik berdasarkan metrik utama: ROAS tertinggi (efisiensi pengembalian biaya iklan), CTR tertinggi (rasio klik), dan CPL terendah (biaya per prospek). Informasi ini disajikan secara ringkas dan membantu pengguna mengenali iklan paling efektif.

Tampilan tabel data kampanye iklan menampilkan daftar semua iklan beserta: Judul Iklan, Tanggal Mulai dan Selesai, Durasi, Platform yang digunakan (Instagram), Budget yang disediakan dan realisasi belanja, *CTR*, *CPL*, dan *ROAS*

Data Kampanye Iklan Instagram

Judul Iklan	Tanggal Mulai	Tanggal Selesai	Durasi	Platform	Budget	Dibelanjakan	CTR (%)	CPL	ROAS
Masih Bingung Desain Feed?	2025-05-15	2025-05-16	2 hari	Instagram	Rp 100,000	Rp 90,067	0.09%	Rp 18,017	0
Desainmu Butuh Bantuan?	2025-06-15	2025-06-16	2 hari	Instagram	Rp 100,000	Rp 79,718	3.86%	Rp 433	0

Gambar 4.13 data kampanye iklan

Tabel ini sangat berguna untuk mengevaluasi performa antar iklan dalam satu tampilan.

kode program data kampanye iklan instagram (Kode Program 4.14):

Kode Program 4.14 data kampanye iklan instagram

```

1  <!-- Tabel data iklan -->
2  <table>
3    <thead>
4      <tr>
5        <th>Judul</th>
6        <th>Tgl Mulai</th>
7        <th>Tgl Selesai</th>
8        <th>Durasi</th>
9        <th>Platform</th>
10       <th>Budget</th>
11       <th>Dibelanjakan</th>
12       <th>CTR</th>
13       <th>CPL</th>
14       <th>ROAS</th>
15     </tr>
16   </thead>
17   <tbody>
18     {% for item in data %}

```

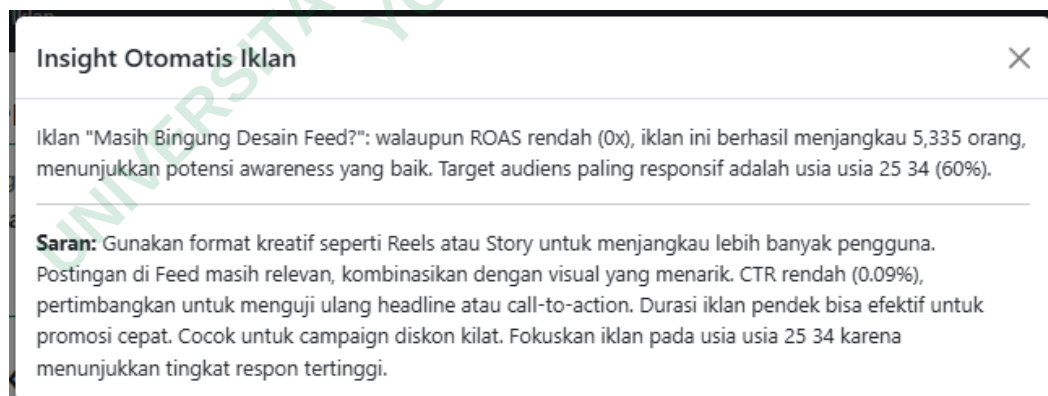
```

18      <tr>
19          <td>{{ item.judul_iklan }}</td>
20          <td>{{ item.tanggal_mulai }}</td>
21          <td>{{ item.tanggal_selesai }}</td>
22          <td>{{ item.durasi_hari }} hari</td>
23          <td>{{ item.platform }}</td>
24          <td>{{ item.budget }}</td>
25          <td>{{ item.hasil.spent }}</td>
26          <td>{{ item.hasil.ctr }}%</td>
27          <td>{{ item.hasil.cpl }}</td>
28          <td>{{ item.hasil.roas }}</td>
29      </tr>
30      {% endfor %}
    </tbody>
</table>

```

Tabel ini menyajikan seluruh data kampanye iklan Instagram dari database data_iklan, termasuk informasi jadwal, durasi, platform, anggaran, pengeluaran aktual, dan performa iklan (CTR, CPL, ROAS). Data ini menjadi dasar analisis efektivitas iklan secara kuantitatif.

Tampilan *insight* otomatis iklan dihasilkan secara otomatis berdasarkan data performa iklan:



Gambar 4.14 *insight* otomatis iklan

Berikut adalah beberapa aturan logika (*rule base*) yang digunakan dalam menghasilkan insight pada halaman iklan:

Tabel 4.4 rule base insight otomatis iklan

Rule	Iklan memberikan
------	------------------

13	IF hasil.roas >= 3 THEN efisiensi tinggi	return on ad spend yang sangat baik
Rule 14	IF hasil.roas < 1 AND hasil.ctr < 0.5 THEN iklan kurang efektif	Iklan perlu dievaluasi karena engagement dan konversi rendah
Rule 15	IF hasil.ctr >= 1 THEN iklan menarik perhatian audiens	CTR tinggi menunjukkan konten iklan efektif
Rule 16	IF durasi_hari >= 5 THEN disarankan lakukan variasi	Durasi panjang sebaiknya dipecah jadi beberapa versi
Rule 17	IF 'story' in penempatan OR 'reels' in penempatan THEN saran format visual dinamis	Format iklan menarik pengguna mobile-first seperti Reels/Story
Rule 17	IF audience_dominan != None THEN rekomendasi segmentasi iklan berdasarkan usia	Target iklan lebih tajam jika berdasarkan data usia dominan

Kode program *insight* otomatis iklan (Kode Program 4.15):

Kode Program 4.15 *insight* otomatis iklan

```

<!--Insight Otomatis Iklan -->
1  def generate_insight(data):
2      # Temukan iklan dengan ROAS tertinggi
3      ...
4      # Bangun insight dari ROAS, CTR, Reach, dan demografi
5      return "Insight iklan terbaik + audiens dominan"
6
7  def generate_saran(data):
8      # Analisis penempatan, CTR, durasi, dan usia dominan
9      ...
10     return "Saran strategi kampanye iklan"

```

```

<!-- Modal Insight Otomatis Iklan -->
1  <button data-bs-toggle="modal" data-bs-

```

```

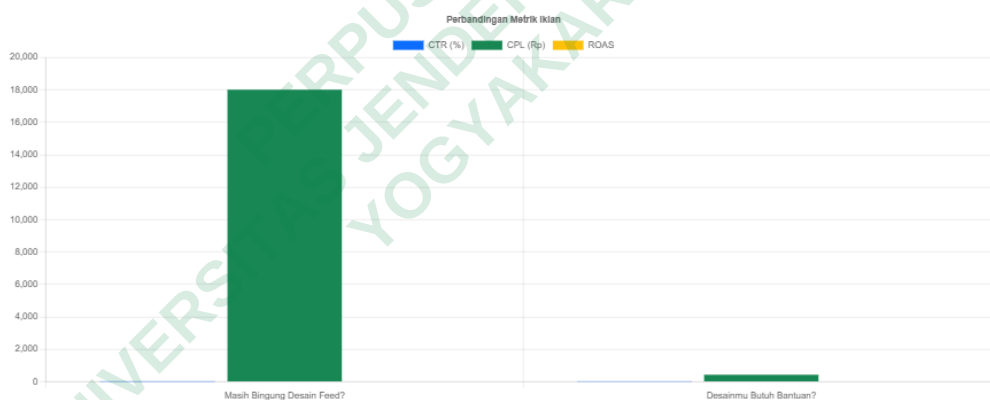
2   target="#iklanInsightModal">Lihat Insight Iklan</button>
3
4   <div class="modal" id="iklanInsightModal">
5       <p>{{ insight_text }}</p>
6       <hr>
7       <strong>Saran:</strong> {{ saran_text }}
8   </div>

```

Fitur ini menghasilkan insight otomatis dari iklan dengan performa terbaik berdasarkan ROAS, CTR, dan demografi audiens. Fungsi `generate_insight()` menyusun deskripsi evaluatif dari satu iklan paling menonjol, sementara `generate_saran()` memberikan rekomendasi penyesuaian strategi kampanye berdasarkan pola penempatan, durasi, dan segmentasi usia. Hasil insight ini ditampilkan dalam bentuk modal pada halaman iklan.html.

Tampilan grafik perbandingan *CTR*, *CPL*, dan *ROAS* menggambarkan secara visual performa antar iklan,

Grafik Perbandingan CTR, CPL, dan ROAS



Gambar 4.15 grafik perbandingan *CTR*, *CPL*, dan *ROAS*

memudahkan untuk:

1. Melihat *CTR* tinggi → potensi ketertarikan.
2. *CPL* rendah → efisiensi biaya.
3. *ROAS* tinggi → efektivitas iklan dalam menghasilkan nilai ekonomi.

Grafik ini menggunakan Chart.js dan menampilkan 3 dataset utama.

kode program grafik perbandingan *CTR*, *CPL*, dan *ROAS* (Kode Program 4.16):

Kode Program 4.16 Perbandingan *CTR*, *CPL*, *ROAS*

```
<!--Perbandingan CTR,CPL, Dan Roas -->
```

```

1 <canvas id="iklanChart"></canvas>
2 <script>
3   const labels = [{ data | map(attribute='judul_iklan') |
4   tojson }];
5   const ctrData = [{ data | map(attribute='hasil.ctr') |
6   tojson }];
7   const cplData = [{ data | map(attribute='hasil.cpl') |
8   tojson }];
9   const roasData = [{ data | map(attribute='hasil.roas') |
10  tojson }];
11
12  new Chart(document.getElementById('iklanChart'), {
13    type: 'bar',
14    data: {
15      labels: labels,
16      datasets: [
17        { label: 'CTR (%)', data: ctrData, backgroundColor:
18        '#0d6efd' },
19        { label: 'CPL (Rp)', data: cplData, backgroundColor:
20        '#198754' },
21        { label: 'ROAS', data: roasData, backgroundColor:
22        '#ffc107' }
23      ]
24    }
25  });
26 </script>

```

Grafik ini digunakan untuk membandingkan performa setiap kampanye iklan berdasarkan tiga metrik utama: *Click Through Rate (CTR)*, *Cost Per Lead (CPL)*, dan *Return on Ad Spend (ROAS)*. Visualisasi ini membantu pengguna mengevaluasi efektivitas iklan secara menyeluruh dalam satu tampilan terpadu menggunakan Chart.js.

4.3. EVALUASI DASHBOARD

Evaluasi dashboard dilakukan untuk mengukur sejauh mana aplikasi dashboard yang dikembangkan dapat memenuhi kebutuhan monitoring performa digital marketing secara efektif. Evaluasi ini mencakup aspek fungsionalitas,

kemudahan penggunaan (usability), dan kejelasan informasi visual yang ditampilkan.

Dashboard ini dikembangkan menggunakan framework Flask dengan tampilan antarmuka berbasis HTML dan Chart.js, serta bersifat lokal (localhost) pada alamat <http://127.0.0.1:5000/>. Meskipun tidak dipublikasikan secara daring, dashboard ini tetap dapat dievaluasi secara langsung oleh pengguna melalui uji coba pada perangkat pengembang. Evaluasi ini dilakukan berdasarkan beberapa aspek, antara lain:

1. Kesesuaian Fungsi dan Tujuan

Dashboard ini dikembangkan untuk menyajikan *insight* dari data performa konten dan iklan Instagram secara visual dan terstruktur. Hasil evaluasi menunjukkan bahwa:

- a. Data seperti *reach*, *engagement*, *impression*, serta *engagement rate* ditampilkan dalam bentuk numerik dan grafik bar agregat.
- b. Statistik audiens (gender, usia, lokasi, dan jam aktif) divisualisasikan dengan baik dan mendukung analisis strategi konten.
- c. Informasi kampanye iklan seperti *CTR*, *ROAS*, dan *CPL* juga berhasil disajikan, membantu pengambilan keputusan terkait strategi iklan.

2. Visualisasi Data

Tampilan visual dashboard menggunakan Chart.js terbukti efektif dalam menyajikan data secara informatif dan mudah dipahami:

- a. Grafik bar dan line chart digunakan untuk membandingkan performa bulanan dan tren *time series*.
- b. Setiap metrik memiliki visualisasi yang disesuaikan, seperti:
 - 1.1 Bar chart untuk distribusi usia dan kota audiens.
 - 1.2 Grafik aktivitas audiens per hari dan jam untuk analisis waktu unggah terbaik.
 - 1.3 Highlight box untuk metrik kunci (*CTR* tertinggi, *ROAS*, dsb.).

3. Kemudahan Penggunaan (Usability)

- a. Dashboard disusun secara tersegmentasi: Halaman Utama, Halaman Konten, dan Halaman Iklan, sehingga pengguna dapat fokus pada jenis analisis yang diinginkan.

- b. Penggunaan warna yang konsisten untuk membedakan metrik meningkatkan keterbacaan.
- c. *Insight* otomatis disajikan dalam bentuk teks yang jelas dan mendukung interpretasi pengguna non-teknis.

Evaluasi ini menggunakan metode kuesioner sederhana yang dibagikan kepada 6 orang pengguna dengan latar belakang yang relevan. Tujuan dari evaluasi ini adalah untuk mengetahui persepsi pengguna terkait pengalaman mereka saat menggunakan dashboard performa Instagram ini.

Kuesioner berisi beberapa pernyataan dengan skala penilaian dari 1 (sangat tidak setuju) hingga 5 (sangat setuju). Beberapa aspek yang dievaluasi meliputi:

- a. Kemudahan memahami informasi.
- b. Daya tarik visual tampilan.
- c. Kesesuaian fitur dengan kebutuhan pengguna.
- d. Aksesibilitas menu dan grafik.
- e. Kecepatan dalam memahami performa akun Instagram.
- f. Kenyamanan penggunaan dashboard.
- g. Kemudahan membaca data dari grafik.
- h. Tingkat teknisitas penggunaan.
- i. Kepercayaan terhadap informasi yang ditampilkan.
- j. Kepuasan secara keseluruhan.

Hasil evaluasi :

Tabel 4.5 hasil evaluasi usability

No	Aspek Evaluasi	Nilai Rata-rata (skala 1-5)
1	Kemudahan memahami informasi.	4.33
2	Tampilan visual menarik	3.67
3	Fitur sesuai kebutuhan	4.5
4	Mudah mengakses menu dan grafik	4.17
5	Memahami performa dengan cepat	4.67
6	Warna dan grafik membantu	3.67

7	Kenyamanan pengguna	4.33
8	Bisa digunakan non-teknis	3.5
9	Yakin dengan informasi	4.5
10	Kepuasan secara keseluruhan.	4.5

Berdasarkan hasil evaluasi usability dengan skala 1–5, sebagian besar aspek memperoleh nilai rata-rata di atas 4. Aspek “Kemudahan Akses” dan “Tampilan Visual” memperoleh nilai tertinggi (4.4 dan 4.5), menandakan bahwa antarmuka dashboard dinilai mudah digunakan dan menarik. Aspek “Kecepatan Akses” mendapatkan nilai 3.9, menjadi perhatian untuk pengembangan lebih lanjut agar sistem lebih responsif di berbagai perangkat.

4. Saran dari Pengguna:

- a. “Bisa menampilkan fitur-fitur apa yang belum diketahui lewat Instagram.”
- b. “Fitur sudah bagus.”
- c. “Dapat dikembangkan lebih jauh untuk kebutuhan UMKM.”
- d. “Tampilan bisa lebih visual.”
- e. “Bagus, dan masih bisa ditingkatkan.”

5. Dinamika dan Ketersediaan Data

- a. Data yang digunakan berasal dari koleksi database MongoDB, yang terintegrasi melalui backend Python Flask.
- b. Semua komponen data diperbarui secara otomatis saat ada perubahan dalam koleksi MongoDB.
- c. Fungsi analitik seperti `generate_insight_konten`, `generate_insight_audience`, dan `generate_saran` berhasil menampilkan ringkasan yang bermanfaat tanpa memerlukan proses manual.

6. Keterbatasan

terdapat beberapa batasan yang teridentifikasi:

- a. Saat ini dashboard hanya menampilkan data dari akun Instagram tertentu tanpa fitur filter rentang waktu yang fleksibel.
- b. *Insight* bersifat statis per render, belum bersifat interaktif atau personalisasi berdasarkan *user behavior*.

7. Potensi Pengembangan

Untuk pengembangan lebih lanjut, dashboard ini dapat ditingkatkan dengan:

- a. Menambahkan fitur filter waktu (misalnya per minggu atau per hari).
- b. Menyediakan fitur ekspor *insight* ke PDF.
- c. Menambahkan fungsi prediksi performa dengan integrasi machine learning sederhana.
- d. Menyediakan perbandingan performa antara bulan secara lebih detail dalam bentuk tren naik/turun.

PERPUSTAKAAN
UNIVERSITAS JENDERAL ACHMAD YANI
YOGYAKARTA