Implementasi Web *Scraping* Untuk Ekstraksi Data Penjual dan Produk Panel Surya Di *E-Commerce*

Muhammad Adhit Dwi Yuda[1]

[1]Program Studi Teknik Informatika, Universitas Cendekia Abditama

E-mail: adhit@uca.ac.id[1]

Abstrak. Perkembangan *e-commerce* di Indonesia telah mendorong peningkatan penjualan produk berbasis energi terbarukan, seperti panel surya. Namun, keterbatasan akses terhadap data produk dan penjual di *marketplace* seperti Tokopedia menimbulkan tantangan dalam analisis pasar. Penelitian ini menerapkan metode *web scraping* untuk mengotomatisasikan ekstraksi data penjual dan produk panel surya di Tokopedia. studi ini menggunakan *BeautifulSoap* dan *Selenium* sebagai pustaka utama untuk *scraping*, serta *pandas* untuk pembersihan data dan analisis awal dalam *Jupyter Notebook*. *BeautifulSoap* digunakan untuk *parsing* HTML statis, sedangkan *Selenium* menangani pemuatan konten dinamis dan interaksi dengan elemen berbasis *Javascript*. Hasil penelitian ini menunjukkan bahwa metode ini efektif dalam pengumpulan informasi terkait harga, lokasi penjual, dan spesifikasi produk. Namun, tantangan seperti konten dinamis, perlindungan *anti-bot* dari Tokopedia, serta verifikasi *captcha* menjadi hambatan yang perlu diatasi dengan teknik khusus. Penelitian ini berkontribusi dalam menyediakan dataset yang dapat digunakan untuk analisis tren pasar dan pengambilan keputusan bisnis di sektor energi terbarukan.

Kata kunci: *Web Scraping; Solar panels; BeautifulSoap; Selenium; Data extraction*.

Abstract. The development of e-commerce in indonesia has driven an increase in sales of renewable energy-based products such as solar panels. However, limited access to product and seller data in marketplaces like Tokopedia poses challenges for market analysis. This study implements a web scraping method to automate the extraction of seller and product data of solar panels on Tokopedia. The study employs BeautifulSoap and Selenium as the main libraries for scraping, and Pandas for data cleaning and initial analysis in jupyter notebook. BeautifulSoap is used for parsing static HTML, while Selenium handles dynamic content loading and interactions with JavaScript-based elements. The research results indicate that this method is effective in collecting information related to pricing, seller location, and product specification. However, challenges such as dynamic content, anti-bot protection from Tokopedia, and CAPTCHA verification pose obstacles that need to be addressed with specific techniques, This study contributes to providing dataset that can be used for market trend analysis and business decision-making in the renewable energy sector.

Keywords: Web Scraping; Solar panels; BeautifulSoap; Selenium; Data extraction.

1. Pendahuluan

Pasar *e-commerce* di Indonesia mengalami pertumbuhan pesat dalam beberapa tahun terakhir, seiring dengan meningkatnya adopsi teknologi digital dan perubahan pola konsumsi masyarakat. salah satu sektor yang mengalami perkembangan signifikan adalah penjualan produk energi baru terbarukan,

termasuk panel surya. Panel surya menjadi solusi yang makin diminati sebagai alternatif sumber energi ramah lingkungan untuk rumah tangga maupun industri. Namun, meskipun permintaan terhadap produk ini terus meningkat akan kesadaran energi baru terbarukan, keterbatasan data yang komprehensif mengenai penjual dan produk panel surya di platform *e-commerce* masih terbatas. Keterbatasan akses terhadap data tersebut menyulitkan berbagai pihak, termasuk peneliti dan pelaku bisnis, dan pemerintah, dalam melakukan analisis pasar, pemetaan kompetisi, serta perencanaan strategi pemasaran yang berbasis data.

Untuk Mengatasi tantangan ini, penelitian ini mengimplementasikan teknik *web scraping* sebagai metode otomatisasi dalam pengambilan data dari platform *e-commerce*. Dalam konteks penelitian ini, *web scraping* digunakan untuk mengumpulkan informasi mengenai penjual dan produk panel surya yang tersedia di Tokopedia, salah satu *marketplace* terbesar di Indonesia. Dengan memanfaatkan teknik ini, data dapat diperoleh secara sistematis dan dalam jumlah besar tanpa harus menggunakan *API* publik yang sering kali memiliki keterbatasan akses dan fitur.

Web Scraping merupakan teknik pengumpulan data secara otomatis dari situs web menggunakan skrip pemrograman. Teknik ini memungkinkan ekstraksi data dalam skala besar dengan cara menelusuri dan mengambil elemen-elemen tertentu dari struktur *HTML* suatu halaman web. Data yang diperoleh kemudian dapat diolah lebih lanjut untuk berbagai keperluan, seperti analisis tren pasar, pemetaan persaingan hingga pengambilan keputusan berbasis data [1]. Dalam penelitian ini, *web scraping* digunakan untuk memperoleh informasi terkait produk panel surya, termasuk harga, *rating*, jumlah ulasan, serta lokasi penjual. Dengan demikian, penelitian ini diharapkan dapat memberikan kontribusi dalam penyediaan data yang lebih akurat dan terkini guna mendukung pengembangan sektor energi baru terbarukan di Indonesia. strak elemen yang diinginkan.

Web *scraping* pada *e-commerce* telah banyak digunakan dalam pemantauan harga, inventori, dan analisis konsumen[2]. Namun, penerapannya dalam konteks produk niche seperti panel surya masing jarang dieksplorasi, terutama di Indonesia. Studi terdahulu cenderung fokus pada penggunaan API resmi untuk kategori produk populer, sedangkan pendekatan *scraping* dinamis untuk toko atau produk EBT belum banyak diteliti.

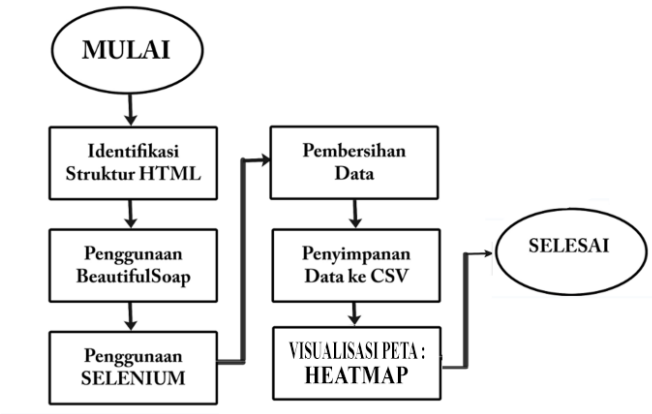
Tokopedia dipilih karena memiliki pangsa pasar yang besar, sekitar 35% dari GMV *e-commerce* Indonesia per tahun 2022, menempatkan sebagai platform lokal terdepan, hanya selisih tipis dari Shopee, dan jauh melampaui pesaing seperti Lazada dan Bukalapak [3]. Pemilihan ini mempertimbangkan representativitas data dan aksesibilitas struktur HTML-nya dibandingkan platform lain.

Beberapa komponen utama dalam *web scraping* :

a. HTML *Parsing*: Menggunakan *BeautifulSoap* untuk membaca dan mengambil elemen dari dokumen HTML [4].
b. Automasi Peramban : *Selenium* digunakan untuk menangani situs web dinamis yang memerlukan interaksi dengan *JavaScript* [5].
c. Pembersihan Data : Menggunakan *Pandas* untuk membersihkan dan menstrukturkan data agar siap dianalisis [6].
d. Visualisasi *Heatmap* : Menggunakan *Folium* untuk membuat peta interaktif, intensitas warna pada *heatmap* menunjukkan jumlah setiap wilayah [7]

2. Metode

Penelitian ini menggunakan metode *web scraping* berbasis *Python* dengan *library BeautifulSoap* dan *Selenium* di lingkungan *Jupyter Notebook.* Proses *scraping* dilakukan dalam beberapa tahap ditunjukkan pada Gambar 1.



***Gambar* 1.** Metode Penelitian

Proses *scraping* dilakukan melalui beberapa sistematis :

a. Identifikasi Struktur *HTML* : Menganalisis elemen web Tokopedia yang berisi data penjual dan produk [8].
b. Penggunaan *BeautifulSoap* : Mengurai *HTML* statis untuk mengekstraksi data yang tersedia tanpa eksekusi *Javascript* [9].
c. Penggunaan *Selenium* : Mengakses dan menavigasi elemen dinamis yang memerlukan eksekusi *javascript*, seperti *pagination* dan *load-on-scroll* [10].
d. Pembersihan data : Menggunakan *Pandas* untuk menghapus data duplikat, menangani *missing value*, dan standarisasi format data [11].
e. Penyimpanan Data ke *CSV* : Dataset yang dikumpulkan disimpan dalam format CSV untuk bisa dianalisis lebih lanjut [12].
f. Visualisasi Peta *heatmap* : Tahap ini Menampilkan distribusi geografis melalui peta interaktif berbasis *heatmap* [13].

2.1. Identifikasi Struktur HTML

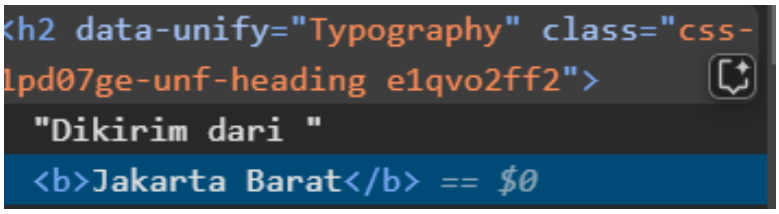
Tahap identifikasi Struktur *HTML* diawali dengan menganalisis kode sumber halaman web menggunakan *developer tool* di browser (misalnya, *Google Chrome*) [14]. Langkah ini mencakup inspeksi elemen-elemen *HTML* yang memuat informasi penting seperti nama produk, harga, lokasi penjual, *rating* dan jumlah ulasan. Gambar 2, 3, 4, dan 5 menunjukkan contoh beberapa elemen tersebut pada situs web Tokopedia.

```
<h1 data-expanded="false" class="css-1xfe
dof" data-testid="lblPDPDetailProductNam
e">ECOFLOW Solar Panel Solar Cell Panel
Bundle Tenaga Matahari Surya 160W
</h1> == $0
```

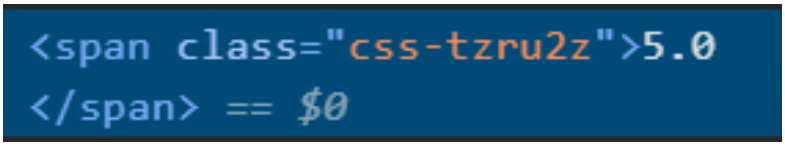
Gambar 2. Elemen sumber web tokopedia untuk judul produk

```
<div class="price" data-testid="lblPDPDet
ailProductPrice">Rp9.098.000</div> == $0
```

Gambar 3. Elemen sumber web tokopedia untuk harga



Gambar 4. Elemen sumber web tokopedia untuk lokasi penjual



Gambar 5. Elemen sumber web tokopedia untuk *rating*

2.2. Penggunaan BeautifulSoap

Pada tahap ini, *BeautifulSoap* digunakan untuk mengurai (*parsing*) dokumen *HTML* yang diperoleh dari permintaan *HTTP* ke halaman Tokopedia. Karena Tokopedia menggunakan *JavaScript* untuk memuat sebagian besar kontennya secara dinamis, metode ini hanya efektif untuk mengekstrak data yang tersedia dalam *HTML* statis, proses ekstraksi mengirim permintaan *HTTP* menggunakan pustaka *requests*, kemudian hasil *HTML* yang diperoleh diuraikan menggunakan *BeautifulSoap*. Identifikasi elemen dilakukan dengan metode seperti *find()* dan *find_all()* untuk mengambil semua *link* produk, seperti , , atau <a>, yang berisi informasi produk, harga, dan nama penjual. Gambar 6, 7 dan 8 menunjukkan contoh algoritma penggunaan pustaka dan metode untuk identifikasi elemen.

```
# HTTP Request
webpage = requests.get(URL, headers=HEADERS)
```

Gambar 6. Algoritma data diambil dari situs web tokopedia

```
# Fetch links as List of Tag Objects
links = soup.find_all
("a", attrs={'class':
            'oQ94Awb6LlTiGByQZo8Lyw== IM26HEnTb-krJayD-R0OHw=='})

links

[<a class="oQ94Awb6LlTiGByQZo8Lyw== IM26HEnTb-krJayD-R0OHw==" data-
lar-cell-panel-bundle-tenaga-matahari-surya-160w?extParam=ivf%3Dfal
="><div class="FsRHU-HdyqHfg3GrIQNUWA=="><div class="Het-JIVjmDQQce
ine-block;opacity:1;border:0;margin:0;padding:0;width:initial;heigh
```

Gambar 7. Algoritma identifikasi elemen untuk mengambil semua link produk

```
new_soup.find
("div", attrs=
{"data-testid":
 'lblPDPDetailProductPrice'}).text

'Rp9.098.000'
```

Gambar 8. Algoritma mengambil elemen pada harga

2.3. Penggunaan Selenium

Pada tahap ini, *Selenium* digunakan untuk mengakses dan menavigasi elemen-elemen dinamis yang memerlukan eksekusi *javascript*, seperti *pagination, load and scroll*, dan pemuatan data produk secara asinkron. Tokopedia memuat sebagian besar kontennya secara dinamis, sehingga permintaan *HTTP* biasa tidak cukup untuk mengambil semua data yang dibutuhkan. *Selenium* memungkinkan interaksi langsung dengan halaman web layaknya penggunaan manusia, termasuk menggulir halaman untuk memicu pemuatan tambahan data (*lazy loading*) dan mengeklik tombol 'muat lebih banyak' pada halaman pencarian produk. proses ini di mulai dengan mengisialisasi *Selenium WebDriver*, kemudian menggunakan *driver.get(url)* untuk memuat halaman. Dengan bantuan *loop*, Selenium dapat mengambil banyak halaman secara otomatis. Gambar 9, 10, dan 11 menunjukkan contoh algoritma penggunaan selenium.

```
# Setup driver
driver = webdriver.Chrome()
```

Gambar 9. Algoritma *setup webdriver*

```
# Loop untuk halaman 1-100
for page in range(1, 101):
    print(f"Scraping Halaman {page}...")

    URL = f"https://www.tokopedia.com/search?st=&page={page}
    driver.get(URL)
```

Gambar 10. Algoritma *loop* mengambil 100 halaman pada tokopedia

```
# Simpan semua link produk
links_list = [link.get('href')
              for link in product_links if link.get('href')]

# Loop untuk mengambil detail produk dari tiap link
for link in links_list:
    try:
        driver.get(link)  # Buka halaman produk
        time.sleep(3)  # Tunggu agar halaman termuat
```

Gambar 11. Algoritma simpan semua link produk dan *loop* setiap detail produk link

2.4. Perbandingan Tools dan Evaluasi Performa

Selain menggunakan *BeautifulSoap* dan *Selenium*, penulis juga melakukan studi perbandingan dengan dua *tools* populer lainnya, yaitu *Scrapy* dan *Playwright*, untuk mengevaluasi efektivitas *scraping* terhadap elemen dinamis Tokopedia. Tabel 1 menyajikan perbandingan berdasarkan tiga aspek utama : kemudahan *scraping* elemen dinamis, kemampuan mengatasi *captcha*/anti bot, dan kecepatan eksekusi *scraping*.

Tabel 3. Perbandingan tools *Scrapy*, *Playwright*, dan *Selenium*

Tools	Dukungan Elemen Dinamis	Captcha Handling	Kecepatan Eksekusi (100 Halaman)
Selenium	Sangat Baik	Terbatas (manual bypass)	± 3.2 detik/halaman
Scrapy	Terbatas	Tidak mendukung	± 1.1 detik/halaman
Playwright	Baik (multi-tab & JS)	Mendukung (headless stealth)	± 2.5 detik/halaman

Pemilihan *Selenium* dalam penelitian ini didasarkan pada kebutuhan untuk menangani konten yang dimuat secara dinamis melalui *JavaScript*, serta kebutuhan untuk berinteraksi langsung dengan elemen halaman web (scroll, klik, dll.).

2.5. Pembersihan Data

Setelah data berhasil diekstrak, tahap selanjutnya adalah pembersihan data menggunakan pustaka *pandas* untuk memastikan kualitas dan konsistensi dataset [15]. Proses ini diawali dengan identifikasi data duplikat seperti contoh pada gambar 12.

```
#mencari total duplikat pada tabel title
ps["title"].duplicated().sum()

279
```

Gambar 12. Algoritma mencari data terduplikasi

yang memungkinkan pendeteksian entri yang muncul lebih dari satu kali dalam dataset. tabel yang dihapus data terduplikasi oleh penulis yaitu tabel *title*, tabel yang berisikan judul produk. Pada dataset tabel *title* ditemukan 279 data terduplikasi, maka baris tersebut dihapus menggunakan algoritma drop seperti contoh gambar 13.

```
#hapus duplikasi
ps.drop_duplicates(subset="title", inplace=True)

ps["title"].duplicated().sum()

0
```

Gambar 13. Algoritma hapus data terduplikasi

Selanjutnya, tahapan pencarian nilai kosong dilakukan dengan menghitung jumlah data yang kosong setiap kolomnya. Gambar 14 menunjukkan contoh pada menghitung dataset tersebut.

```
#Mencari dataset kosong
ps.isnull().sum()

nama_toko         1
domisili          1
title             0
price             0
rating          439
reviews         439
availability      0
dtype: int64
```

Gambar 14. Algoritma mencari data kosong

Bahwa ditemukan data yang kosong pada kolom nama_toko, domisili, *rating*, dan *review*. Untuk kolom yang lain tidak ditemukan ada data yang kosong. Kemudian menangani data yang kosong, maka harus diganti data yang kosong dengan isian tanda (-) pada kolom nama_toko, domisili, *rating*, *reviews*, agar data yang kosong terisi. Gambar 15 menunjukkan contoh mengisi data kosong.

```
#Mereplace Data Kosong dengan tanda (-)
ps.nama_toko = ps.nama_toko.fillna('-')
ps.domisili = ps.domisili.fillna('-')
ps.rating = ps.rating.fillna('-')
ps.reviews = ps.reviews.fillna('-')

#hasil isi data kosong
ps.isnull().sum()

nama_toko       0
domisili        0
title           0
price           0
rating          0
reviews         0
availability    0
dtype: int64
```

Gambar 15. Algoritma mengisi data kosong dengan (-)

2.6. Penyimpanan Data

Setelah data melewati ekstraksi dan pembersihan, tahap selanjutnya adalah penyimpanan dalam format *csv* agar dapat diakses dan dianalisis lebih lanjut. Penyimpanan dalam format *csv* memiliki keunggulan karena sifatnya yang ringan, mudah dibaca, serta kompatibel dengan berbagai perangkat lunak analisis data seperti *Python, R,* dan *excel*. Proses Penyimpanan diakukan dengan menggunakan *library pandas* melalui fungsi *to_csv()*, yang memungkinkan penyimpanan data dalam format yang terstruktur tanpa menyertakan indeks baris yang tidak diperlukan [16]. Gambar 16 menunjukkan contoh algoritma simpan data ke format *csv*.

```
#simpan dataset setelah cleansing
ps.to_csv('panelsurya_tokopediafinal.csv',
          index=False)
```

Gambar 16. Algoritma simpan data ke format *csv*

2.7. Visualisasi Heatmap

Setelah dataset berhasil dibersihkan dan disimpan, tahapan selanjutnya melakukan visualisasi spasial menggunakan peta *heatmap* untuk menampilkan distribusi geografis penjual panel surya di Indonesia. Visualisasi ini menggunakan pustaka *Folium* pada lingkungan Jupiter Notebook, dengan metode *geocoding* untuk mengubah nama kota pada domisili menjadi koordinat latitude dan longitude dengan fungsi lambda. Gambar 17 menunjukkan contoh algoritma lamda mengubah kota menjadi koordinat latitude dan longitude.

```
# Tambahkan koordinat ke data utama
df['latitude'] = df['kota'].map(lambda x: koordinat_dict.get(x, (None, None))[0])
df['longitude'] = df['kota'].map(lambda x: koordinat_dict.get(x, (None, None))[1])
```

Gambar 17. Algoritma Lamda Mengubah kota menjadi koordinat latitude dan longitude

Hasil dari proses *geocoding* menghasilkan dua kolom baru, yaitu latitude dan longitude, yang berisi koordinat geografis masing-masing kota. Penambahan kolom ini diperlukan agar data memenuhi persyaratan visualisasi peta, khususnya untuk pemetaan spasial menggunakan pustaka *Folium*. Gambar 18 menampilkan hasil akhir dataset dilengkapi dengan kolom *latitude* dan *longitude*.

df

	nama_toko	domisili	title	price	rating	reviews	availability	kota	popup_info	latitude	longitude
0	Panel Suryaku	Dikirim dari Jakarta Barat	Paket 2pcs Solar Panel Surya 100wp Poly + Cont...	Rp1.243.000	4.9	24.0	166.0	Jakarta Barat	\n Nama Toko: Panel Suryaku \n ...	-6.161569	106.743891
1	Panel Suryaku	Dikirim dari Jakarta Barat	Paket 2pcs Solar Panel Surya 120wp Mono + SCC ...	Rp1.293.000	4.9	112.0	166.0	Jakarta Barat	\n Nama Toko: Panel Suryaku \n ...	-6.161569	106.743891
2	Panel Suryaku	Dikirim dari Jakarta Barat	Solar Panel Surya 120wp Mono Solar Cell 120wp ...	Rp543.000	4.9	189.0	122.0	Jakarta Barat	\n Nama Toko: Panel Suryaku \n ...	-6.161569	106.743891
3	Reseller HARGA Distributor ~TERIMA SILPA DANA ...	Dikirim dari Jakarta Barat	PAKET PANEL SURYA 240WP 12PCS FREE PAKING KAYU...	Rp15.500.000	5.0	2.0	118.0	Jakarta Barat	\n Nama Toko: Reseller HARGA Distrib...	-6.161569	106.743891

Gambar 18. Menampilkan hasil akhir dataset lengkap dengan kolom latitude dan longitude

Setiap titik pada peta merepresentasikan setiap kota tempat toko berada, dengan intensitas warna (gradasi merah-kuning) yang menunjukkan jumlah penjual di wilayah tersebut. Selain itu, setiap titik juga dilengkapi dengan *tooltip* interaktif yang menampilkan informasi detail seperti nama toko, nama produk, harga, *rating*, ulasan, dan ketersediaan stok. Hal ini memberikan pengalaman eksploratif yang mendalam dalam analisis spasial *e-commerce* panel surya. Gambar 19 menampilkan algoritma visualisasi peta interaktif berbasis *heatmap* dan gambar 20 menampilkan hasil visualisasi *heatmap* yang dibuat dalam Jupiter Notebook menggunakan data hasil *scraping* yang telah diproses.

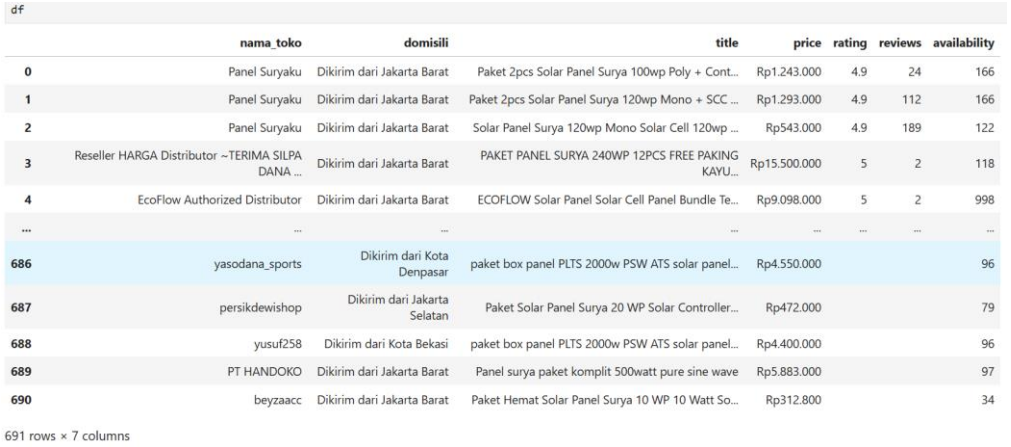
```
# Buat peta Indonesia
m = folium.Map(location=[-2.5, 118], zoom_start=5)

# Cluster marker agar rapi
marker_cluster = MarkerCluster().add_to(m)

# Tambahkan marker per toko
for _, row in df.iterrows():
    folium.Marker(
        location=[row['latitude'], row['longitude']],
        popup=folium.Popup(row['popup_info'], max_width=350),
        tooltip=row['nama_toko']
    ).add_to(marker_cluster)

m
```

Gambar 19. Algoritma Visualisasi peta interaktif berbasis *heatmap*.

Gambar 20. Visualisasi *Heatmap* distribusi penjual panel surya berdasarkan kota di indonesia.

3. Hasil dan Pembahasan

Dari hasil *scraping* yang dilakukan pada Tokopedia, berhasil dikumpulkan sebanyak 690 data produk panel surya dari berbagai penjual. Data yang diperoleh mencakup informasi seperti nama toko, nama produk, harga, lokasi penjual, *rating*, *review* dan ketersediaan barang. Gambar 21 menunjukkan hasil *scraping* sebelum dilakukan pembersihan data.



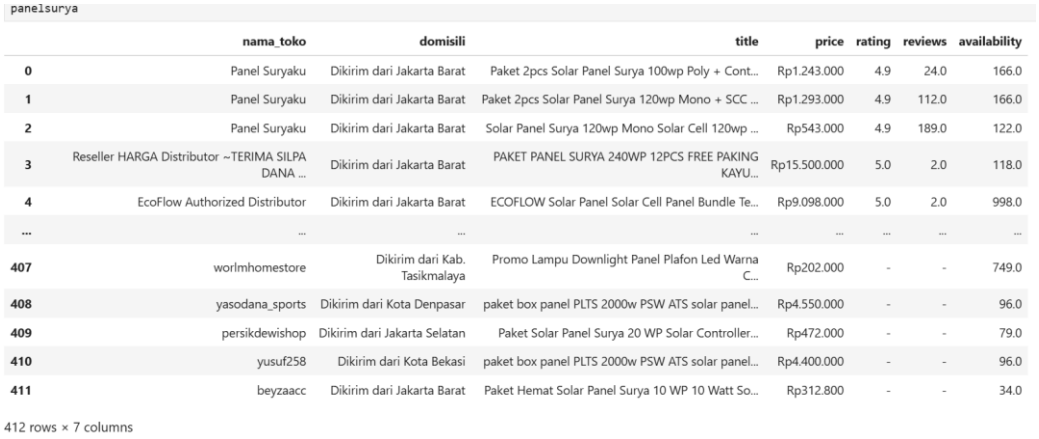
df

	nama_toko	domisili	title	price	rating	reviews	availability
0	Panel Suryaku	Dikirim dari Jakarta Barat	Paket 2pcs Solar Panel Surya 100wp Poly + Cont...	Rp1.243.000	4.9	24	166
1	Panel Suryaku	Dikirim dari Jakarta Barat	Paket 2pcs Solar Panel Surya 120wp Mono + SCC ...	Rp1.293.000	4.9	112	166
2	Panel Suryaku	Dikirim dari Jakarta Barat	Solar Panel Surya 120wp Mono Solar Cell 120wp ...	Rp543.000	4.9	189	122
3	Reseller HARGA Distributor ~TERIMA SILPA DANA ...	Dikirim dari Jakarta Barat	PAKET PANEL SURYA 240WP 12PCS FREE PAKING KAYU...	Rp15.500.000	5	2	118
4	EcoFlow Authorized Distributor	Dikirim dari Jakarta Barat	ECOFLOW Solar Panel Solar Cell Panel Bundle Te...	Rp9.098.000	5	2	998
...	...	...	...	...	...	...	...
686	yasodana_sports	Dikirim dari Kota Denpasar	paket box panel PLTS 2000w PSW ATS solar panel...	Rp4.550.000			96
687	persikdewishop	Dikirim dari Jakarta Selatan	Paket Solar Panel Surya 20 WP Solar Controller...	Rp472.000			79
688	yusuf258	Dikirim dari Kota Bekasi	paket box panel PLTS 2000w PSW ATS solar panel...	Rp4.400.000			96
689	PT HANDOKO	Dikirim dari Jakarta Barat	Panel surya paket komplit 500watt pure sine wave	Rp5.883.000			97
690	beyzaacc	Dikirim dari Jakarta Barat	Paket Hemat Solar Panel Surya 10 WP 10 Watt So...	Rp312.800			34

691 rows × 7 columns

Gambar 21. hasil dataset yang telah di *scraping* sebelum pembersihan data

Adapun setelah dataset dilakukan pembersihan data, dataset tersebut menjadi sebanyak 411 data produk panel surya dari berbagai penjual. Gambar 22 menunjukkan hasil dataset setelah dilakukan pembersihan data.



panelsurya

	nama_toko	domisili	title	price	rating	reviews	availability
0	Panel Suryaku	Dikirim dari Jakarta Barat	Paket 2pcs Solar Panel Surya 100wp Poly + Cont...	Rp1.243.000	4.9	24.0	166.0
1	Panel Suryaku	Dikirim dari Jakarta Barat	Paket 2pcs Solar Panel Surya 120wp Mono + SCC ...	Rp1.293.000	4.9	112.0	166.0
2	Panel Suryaku	Dikirim dari Jakarta Barat	Solar Panel Surya 120wp Mono Solar Cell 120wp ...	Rp543.000	4.9	189.0	122.0
3	Reseller HARGA Distributor ~TERIMA SILPA DANA ...	Dikirim dari Jakarta Barat	PAKET PANEL SURYA 240WP 12PCS FREE PAKING KAYU...	Rp15.500.000	5.0	2.0	118.0
4	EcoFlow Authorized Distributor	Dikirim dari Jakarta Barat	ECOFLOW Solar Panel Solar Cell Panel Bundle Te...	Rp9.098.000	5.0	2.0	998.0
...	...	...	...	...	...	...	...
407	worlmhomestore	Dikirim dari Kab. Tasikmalaya	Promo Lampu Downlight Panel Plafon Led Warna C...	Rp202.000	-	-	749.0
408	yasodana_sports	Dikirim dari Kota Denpasar	paket box panel PLTS 2000w PSW ATS solar panel...	Rp4.550.000	-	-	96.0
409	persikdewishop	Dikirim dari Jakarta Selatan	Paket Solar Panel Surya 20 WP Solar Controller...	Rp472.000	-	-	79.0
410	yusuf258	Dikirim dari Kota Bekasi	paket box panel PLTS 2000w PSW ATS solar panel...	Rp4.400.000	-	-	96.0
411	beyzaacc	Dikirim dari Jakarta Barat	Paket Hemat Solar Panel Surya 10 WP 10 Watt So...	Rp312.800	-	-	34.0

412 rows × 7 columns

Gambar 22. hasil dataset yang telah dilakukan pembersihan data

3.1. Tantangan dalam Proses Scraping

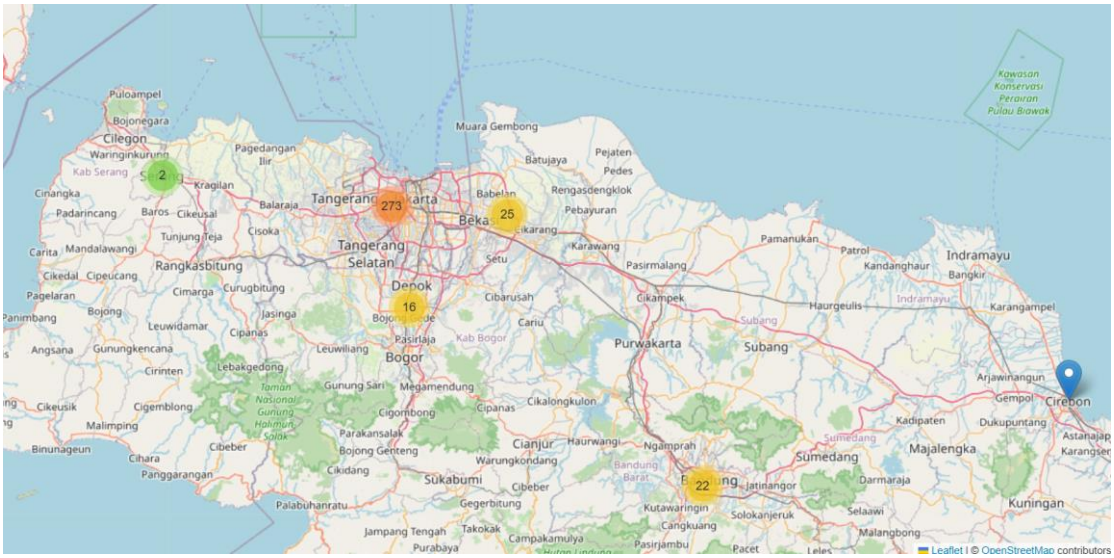
Selama proses scraping, terdapat beberapa tantangan yang dihadapi:

a. *Dynamic content*: Halaman produk Tokopedia menggunakan teknik *lazy-loading* yang mengharuskan penggunaan *selenium* untuk memuat seluruh data sebelum diekstraksi.
b. Proteksi *Anti-Bot*: Tokopedia memiliki mekanisme *anti-scraping* seperti *CAPTCHA* dan pembatasan *ip*, yang mengharuskan penggunaan strategi *rotating proxies* dan *user-agent spoofing*.
c. Data Tidak Terstruktur: beberapa elemen produk memiliki format yang berbeda seperti, seperti harga yang ditampilkan dalam format teks yang perlu dikonversi ke format numerik sebelum analisis.

3.2. Manfaat Data untuk Visualisasi Heatmap

Dataset hasil *scraping* yang telah diperoleh dapat dimanfaatkan untuk distribusi geografis penjual, untuk mengetahui persebaran penjual panel surya di berbagai kota di Indonesia untuk mendeteksi pusat distribusi utama dan wilayah dengan potensi pasar tinggi.

Wilayah Tangerang dan DKI Jakarta menunjukkan potensi pasar yang tinggi dalam distribusi penjual panel surya, berdasarkan hasil visualisasi *heatmap*. Konsentrasi penjual yang dominan di wilayah tersebut mengindikasikan tingginya aktivitas penjualan dan permintaan pasar. Gambar 23 memperlihatkan visualisasi persebaran ini secara spasial.



Gambar 23. *Heatmap* distribusi penjual panel surya dengan titik konsentrasi tertinggi berada di DKI Jakarta dan Tangerang.

4. Kesimpulan

Penelitian ini Membuktikan bahwa *web scraping* menggunakan *BeautifulSoap* dan *Selenium* di lingkungan Jupyter Notebook, efektif untuk mengotomatisasi pengumpulan data penjual dan produk panel surya dari platform *e-commerce* Tokopedia. Proses *scraping* yang mencakup identifikasi struktur *HTML*, penanganan konten dinamis, hingga visualisasi spasial, telah menghasilkan *dataset* yang kaya dan siap digunakan untuk analisis pasar energi baru terbarukan (EBT).

Hasil *scraping* menunjukkan bahwa wilayah DKI Jakarta dan Tangerang memiliki konsentrasi penjual tertinggi, mengindikasikan potensi pasar yang signifikan di kawasan tersebut. Visualisasi peta *heatmap* memperkuat temuan ini dengan menampilkan distribusi geografis yang dapat dimanfaatkan sebagai dasar pemetaan strategis oleh pelaku industri.

Secara akademis, penelitian ini memberikan kerangka kerja *scraping* yang dapat di replikasi untuk produk-produk niche lainnya di *e-commerce*, terutama yang belum memiliki dukungan *API* resmi. Secara praktis, data yang diperoleh dapat dijadikan rujukan bagi UMKM atau pelaku usaha dalam menentukan harga, lokasi pemasaran, dan prefensi produk.

Kedepannya, penelitian ini dapat dikembangkan lebih lanjut dengan menerapkan analisis sentimen pada ulasan produk, integrasi prediksi harga menggunakan *machine learning*, serta visualisasi dalam bentuk dashboard interaktif untuk mendukung pengambilan keputusan berbasis data.

5. Referensi

[1] A. Abodayeh, R. Hejazi, W. Najjar, L. Shihadeh, dan R. Latif, "Web Scraping for Data Analytics: A BeautifulSoup Implementation," dalam *Proceedings of the 2023 Sixth International Conference of Women in Data Science at Prince Sultan University (WiDS PSU)*, Mar. 2023, hal. 65–69. IEEE.

[2] D. Y. Praptiwi et al., "Analisis sentimen online review pengguna e-commerce menggunakan metode Support Vector Machine dan Maximum Entropy (studi kasus: review Bukalapak pada Google Play)," *Jurnal Ilmiah*, 2018.

[3] A. K. Saumi dan D. Rachmawati, "TikTok Tokopedia Geser Shopee, Laju Blibli Bukalapak Makin

Sengit 2024," *Bisnis.com*, Dec. 30, 2023. [Online].
[4] N. Nirsal, A. Apriyanto, F. M. Sinaga, Y. M. Saragih, S. Y. Kusumastuti, L. Judijanto, dan K. H. Rambe, *Big Data: Panduan dan Peluang di Era Digital*. PT Green Pustaka Indonesia, Apr. 2024.
[5] A. Z. Rizquina dan C. I. Ratnasari, "Implementasi Web Scraping untuk Pengambilan Data Pada Website E-Commerce," *Jurnal Teknologi Dan Sistem Informasi Bisnis*, vol. 5, no. 4, hal. 377–383, 2023.
[6] S. L. Rosa, E. A. Kadir, dan A. Kurniawan, "Penggunaan Email dan Internet dalam Menunjang Administrasi di Pemerintahan," *Jurnal Pengabdian Masyarakat dan Penerapan Ilmu Pengetahuan*, vol. 1, no. 2, hal. 1–10, 2020.
[7] R. S. Narastu, M. Wiranata, T. A. Lubis, dan J. Parhusip, "Analisis data eksplorasi dataset gempa bumi Indonesia," *Journal Sains Student Research*, vol. 3, no. 1, hal. 136–142, 2025.
[8] R. Zulfiqri, B. N. Sari, dan T. N. Padilah, "Analisis Sentimen Ulasan Pengguna Aplikasi Media Sosial Instagram pada Situs Google Play Store Menggunakan Naïve Bayes Classifier," *Jurnal Informatika dan Teknik Elektro Terapan*, vol. 12, no. 3, 2024.
[9] A. S. Yondra, D. Triyanto, dan S. Bahri, "Implementasi Web Scraping untuk Mengumpulkan Informasi Produk dari Situs E-Commerce dan Marketplace dengan Teknik Pemrosesan Paralel," *Coding: Jurnal Komputer dan Aplikasi*, vol. 10, no. 1, hal. 93–102, 2022.
[10] Y. A. Hafiz dan E. Sudarmilah, "Implementasi Web Scraping pada Portal Berita Online," *Inisiasi*, vol. 12, no. 1, hal. 55–60, 2023. DOI: 10.59344/inisiasi.v12i1.120.
[11] C. Umri, A. F. Pulungan, A. Halimatusyaddiah, I. E. Sinaga, L. A. Tarigan, N. Aini, dan S. M. Mazaya, "EduSearch: Web Pencarian Cerdas Berbasis Semantik untuk Mencari Data Seluruh Sekolah Formal di Kota Medan," *Jurnal Minfo Polgan*, vol. 13, no. 2, hal. 2699–2713, 2024.
[12] A. Nugroho dan H. M. Haris, "Analisis Efektivitas Teknik Imputasi pada LSTM untuk Meningkatkan Kualitas Data pada Peramalan Curah Hujan," *JIRE*, vol. 7, no. 2, hal. 301–1, Nov. 2024.
[13] F. Sulianta, *Visualisasi Data untuk Pemula*. Surabaya: Feri Sulianta, 2024.
[14] R. H. Nufus dan U. Surapati, "Analisis Sentimen Persepsi Masyarakat Terhadap Timnas Indonesia U-23 dalam AFC-23 Asian Cup 2024 pada Media Sosial X Menggunakan Metode Naïve Bayes Classifier," *Jurnal Indonesia: Manajemen Informatika dan Komunikasi*, vol. 5, no. 3, hal. 2647–2657, 2024. DOI: 10.35870/jimik.v5i3.964.
[15] A. Noviantoro, A. B. Silviana, R. R. Fitriani, dan H. P. Permatasari, "Rancangan dan Implementasi Aplikasi Sewa Lapangan Badminton Wilayah Depok Berbasis Web," *Jurnal Teknik dan Science*, vol. 1, no. 2, hal. 88–103, 2022.
[16] N. E. Febriyanty, *Deteksi Berita Hoax dari Media Online Indonesia Menggunakan Algoritma Naïve Bayes dan Support Vector Machine*, Disertasi Doktor, Universitas Islam Negeri Maulana Malik Ibrahim, 2023.