Jurnal Teknologi Dan Sistem Informasi Bisnis
Vol. 5 No. 4 Oktober 2023 Hal. 377-383
http://jurnal.unidha.ac.id/index.php/jteksis

E-ISSN : **2655-8238**
P-ISSN : **2964-2132**

Implementasi *Web Scraping* untuk Pengambilan Data Pada Website *E-Commerce*

Apriza Zicka Rizquina[1)], Chanifah Indah Ratnasari[2*)]
[1,2]Program Studi Informatika, Fakultas Teknologi Industri, Universitas Islam Indonesia
email: apriza.rizquina@students.uii.ac.id[1], chanifah.indah@uii.ac.id[2]
(*) Corresponding Author

Submitted: 23-06-2023, Reviewed: 04-07-2023, Accepted 18-08-2023
https://doi.org/10.47233/jteksis.v5i4.913

Abstract
Data is a new mind; data is gold; data is a new mine. This is a parable that was as familiar in the digital era as it is today. Data can be utilized, among other things, to improve operational efficiency, spur innovation in a business, understand user needs, and encourage decision-making. The need for data including data from e-commerce websites encourages the emergence of various web scraping methods. To find a method that suits the intended website, it is necessary to experiment with various relevant approaches. One of the methods that can be used to collected e-commerce data is through web scraping techniques using Python and libraries such as Selenium, Beautiful Soup, and Time. The web scraping methods at Shopee and Tokopedia used in this study are HTML Parsing and CSS Selector. This study found that the HTML Parsing and CSS Selector methods cannot be used for web scraping on Shopee owing to bot and CAPTCHA detection mechanisms. However, this method was successfully used on Tokopedia. We conducted 10 web scraping attempts on two product pages, which resulted in around 160–166 data points each time, with 3–18 duplications of data. The average execution time of a program is 1 minute and 0.5 seconds.
Keywords: *Web Scraping, E-commerce, Data Mining, HTML Parsing, CSS Selector*

Abstrak
Data is new mind; data adalah emas; data adalah tambang baru; merupakan perumpamaan yang sudah tidak asing di era digitalisasi seperti saat ini. Data dapat dimanfaatkan di antaranya untuk meningkatkan efisiensi operasional, memacu inovasi dalam suatu bisnis, memahami kebutuhan pengguna, dan mendorong pengambilan keputusan. Kebutuhan akan data termasuk data dari situs web *e-commerce* mendorong munculnya berbagai metode web scraping. Untuk menemukan metode yang sesuai dengan situs web yang dituju, perlu dilakukan eksperimen dengan berbagai pendekatan yang relevan. Salah satu metode yang dapat digunakan untuk mengambil data *e-commerce* yaitu melalui teknik *web scraping* dengan menggunakan Python dan *library* seperti Selenium, Beautifulsoup, dan Time. Metode *web scraping* pada Shopee dan Tokopedia yang digunakan pada penelitian ini adalah HTML *Parsing* dan CSS *Selector*. Penelitian ini menemukan bahwa metode HTML *Parsing* dan CSS *Selector* tidak dapat digunakan untuk *web scraping* pada Shopee dikarenakan adanya mekanisme deteksi bot dan CAPTCHA. Namun, metode ini berhasil digunakan pada Tokopedia. Dilakukan sebanyak 10 kali percobaan *web scraping* pada dua halaman produk yang menghasilkan sekitar 160-166 data pada setiap kali percobaannya, dengan duplikasi sebanyak 3-18 data. Waktu eksekusi program rata-rata adalah 1 menit 0,5 detik.
Keywords: *Web Scraping, E-commerce,* Penambangan Data, HTML *Parsing,* CSS *Selector*

PENDAHULUAN

Data memainkan peran yang sangat penting dalam perkembangan bisnis saat ini. Data telah menjadi aset yang berharga yang dapat dimanfaatkan untuk mendorong pengambilan keputusan, meningkatkan efisiensi operasional, memahami kebutuhan pengguna, serta memacu inovasi dalam suatu bisnis. Agar dapat dimanfaatkan dengan baik, data perlu diolah dan dianalisis terlebih dahulu, salah satunya dengan bantuan sistem atau aplikasi yang melibatkan *machine learning*, *deep learning*, dan *data mining* [1]. Kebutuhan pengolahan data disesuaikan dengan kebutuhan dari analisis data itu sendiri. Sebelum dilakukan pengolahan, *dataset* atau kumpulan data yang akan diolah harus tersedia terlebih dahulu. Jenis *dataset* ada dua, yaitu: (1) *private dataset*, merupakan *dataset* yang diambil dari organisasi/perusahaan/instansi pemilik data tersebut, contohnya, data rumah sakit, bank, sekolah, perusahaan, dan lain sebagainya; (2) *public dataset*, merupakan *dataset* yang diambil dari repositori publik. Selain hal tersebut, terdapat juga data yang pengambilannya harus dilakukan dengan cara penarikan (*crawling*), contohnya data Twitter.

Transformasi digital telah menciptakan platform baru dalam berbisnis dan jual-beli, yaitu *e-commerce*. *E-commerce* atau *electronic commerce* merupakan kegiatan transaksi atau jual-beli dengan menggunakan sarana media elektronik, dalam hal ini yang dimaksud adalah internet. *E-commerce* juga sebagai media untuk memasarkan dan

Jurnal Teknologi Dan Sistem Informasi Bisnis
Vol. 5 No. 4 Oktober 2023 Hal. 377-383
http://jurnal.unidha.ac.id/index.php/jteksis
E-ISSN : **2655-8238**
P-ISSN : **2964-2132**

mempromosikan produk [2]. Tercatat bahwa pertumbuhan pengunjung situs web *e-commerce* di Indonesia yang diakumulasikan dari periode kuartal III/2019 hingga kuartal II/2022 memiliki rata-rata 158.3 juta pengunjung [3]. Shopee dan Tokopedia merupakan *e-commerce* yang populer di Indonesia, dengan jumlah penggunanya yang besar. Kedua platform tersebut menyediakan beragam produk dengan harga yang bervariasi dari setiap toko. Oleh karena itu, penting untuk melakukan analisis data dari *e-commerce* tersebut untuk pengambilan pengetahuan dan keputusan yang beragam. Contohnya, analisis data untuk membantu dalam merancang strategi penjualan produk yang sesuai di masa mendatang.

Salah satu cara pengambilan data dari situs web *e-commerce* adalah *web scraping*. *Web scraping* merupakan teknik yang digunakan untuk mengekstrak data dalam jumlah besar dari situs web dan menyimpannya dalam format file lokal atau basis data dalam bentuk tabel [4]. Proses *web scraping* termasuk dalam tahapan *data mining*, yang melibatkan integrasi berbagai bidang ilmu seperti *machine learning*, pengenalan pola, statistik, basis data, dan visualisasi [5]. Tujuan dari *data mining* adalah mengambil informasi yang tersembunyi dalam kumpulan data/*dataset* [6].

Terdapat berbagai metode *web scraping* yang dapat digunakan, mulai dari metode manual seperti *copy-pasting* hingga metode otomatis. Metode *copy-pasting* adalah metode yang sederhana dan mudah digunakan, dilakukan dengan membuka *browser* dan menyalin data secara manual untuk ditempelkan ke media lain [7]. Akan tetapi metode ini tidak efisien untuk pengambilan data dalam jumlah besar karena membutuhkan waktu yang cukup lama.

Metode lain yang dapat digunakan untuk mengambil data secara otomatis atau disebut dengan *automated web scraping* di antaranya HTML Parsing, Regex, DOM Parsing, dan Xpath [8]. Metode ini dilakukan dengan pengambilan data secara otomatis dengan cara menemukan pola ekstraksi dari satu atau banyak halaman web yang diinginkan [9]. Hal ini tentunya dapat meminimalisir waktu yang dibutuhkan dalam pengambilan data.

Dalam penelitian ini dilakukan percobaan untuk mengambil dan mengekstraksi data *e-commerce* pada Shopee dan Tokopedia. Data yang diambil meliputi nama produk, harga, jumlah produk terjual, rating, lokasi, dan nama toko dari produk yang tersedia pada tahun 2023. Shopee dan Tokopedia yang digunakan adalah Shopee dan Tokopedia Indonesia.

METODE PENELITIAN

Tahapan yang dilakukan dalam penelitian ini ditunjukkan pada Gambar 1.

Gambar 1. Metodologi Penelitian

2.1 Studi Literatur

Web scraping digunakan untuk mengekstrak data yang dari banyak situs web dan akan disimpan dalam bentuk *spreadsheets* atau *database* [10]. *Web scraping* yang juga dikenal dengan istilah *web data extraction*, *web data scraping*, *web harvesting*, atau *screen scraping*, bertujuan untuk mengekstraksi informasi dari situs web menjadi data yang dapat dipahami seperti *spreadsheets*, basis data, atau file *Comma-Separated Values* (CSV) [11]. Untuk menetukan metode *web scraping* yang digunakan dalam penelitian ini, terlebih dahulu dilakukan studi literatur guna mendapat gambaran mengenai masing-masing metode dan mengkaji metode yang digunakan pada penelitian-penelitian terdahulu.

Dewi et al. (2019) melakukan penelitian yang bertujuan untuk mengambil data dari media sosial seperti Facebook dan Twitter dengan menggunakan API. *Application Programming Interface* (API) digunakan di banyak situs untuk mengakses sebagian besar informasi dengan mudah. Pada penelitian tersebut juga digunakan *Regular Expression* (Regex) untuk menemukan teks yang sesuai dengan pola yang ditentukan. Namun demikian, berdasarkan pada penelitian Han dan Anderson (2021), disebutkan bahwa API sering kali sulit diakses, memiliki jangka waktu yang terbatas, serta memerlukan biaya untuk mengaksesnya.

Penelitian lain dilakukan oleh Rizaldi dan Arief (2017), berdasarkan hasil evaluasi performa *web scraping*, diperoleh temuan bahwa metode Regex memiliki penggunaan CPU dan memori yang paling sedikit. Lalu, metode XPath menggunakan waktu yang tersingkat dibandingkan metode lainnya. Sementara itu, untuk penggunaan *bandwidth* paling kecil yaitu dengan menggunakan metode CSS Selector. Selain itu, Elveny et al. (2021) melakukan penelitian untuk mengambil data produk *e-commerce* dengan menggunakan HTML *Parsing* yang dapat mendeteksi nama produk, harga produk, angka ulasan, dan URL produk pertama hingga keempat. *Library* yang digunakan yaitu

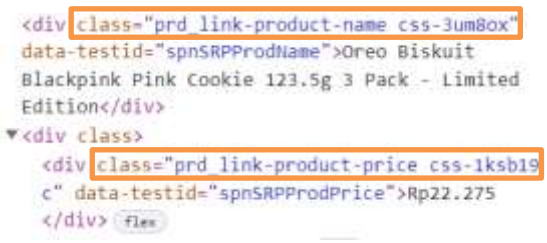
Jurnal Teknologi Dan Sistem Informasi Bisnis
Vol. 5 No. 4 Oktober 2023 Hal. 377-383
http://jurnal.unidha.ac.id/index.php/jteksis
E-ISSN : **2655-8238**
P-ISSN : **2964-2132**

Beautifulsoup yang merupakan salah satu *library* Python yang memliki kemampuan HTML *Parsing*.

Pada penelitian ini dilakukan percobaan pengumpulan data melalui *web scraping* pada *e-commerce* Shopee dan Tokopedia. Berdasarkan studi literatur yang dilakukan, ditentukan metode yang akan digunakan dalam melakukan *web scraping*, yaitu HTML *Parsing* yang memiliki kemampuan untuk ekstraksi data dengan bantuan CSS *Selector* untuk memilih elemen-elemen yang spesifik dalam struktur HTML.

2.2 Identifikasi Halaman Web yang Akan Di-*Scraping*

Pengambilan data pada halaman web yang dilakukan akan melibatkan elemen-elemen dari kode sumber web. Kode sumber halaman web akan ditampilkan dan diidentifikasi elemen-elemen yang dibutuhkan untuk melakukan *scraping*. Gambar 2 menunjukkan contoh beberapa elemen yang akan digunakan pada situs web yang dituju.



Gambar 2. *Class* dari Elemen Sumber Web *E-commerce*

2.3 Membuat Kode *Scraping*

a. *Library* Python

Penelitian ini menggunakan bahasa pemograman Python. Beberapa *library* Python akan digunakan dalam percobaan pengambilan data *web scraping*.

1. Beautifulsoup

   Beautifulsoup adalah *library* Python yang populer yang digunakan untuk mengambil informasi dari dari halaman HTML [8]. *Library* ini dapat mengekstrak informasi seperti teks, atribut, *link*, atau gambar dari halaman web dengan cara *parsing* dari dokumen HTML [15].

2. Selenium

   Selenium merupakan alat pengujian populer yang digunakan untuk pengujian aplikasi [16]. Selenium 2.0 atau Selenium Webdriver merupakan salah satu versi dari Selenium yang dapat digunakan untuk melakukan *web scraping* dikarenakan memiliki kemampuan untuk otomatisasi web dan berinteraksi dengan elemen web seperti klik pada tombol, mengisi form, membuka tab baru, membuka halaman web, dan lain-lain [17]. Selenium Webdriver saat ini mendukung sebagian besar *browser* populer, seperti Chrome, Firefox, Opera, dan lain-lain.

3. Pandas

   Merupakan *open-source library* yang menyediakan alat dan struktur data yang sederhana sehingga mudah untuk digunakan. Pandas dapat digunakan untuk memuat, menyiapkan, dan memanipulasi data, serta membuat model yang akan dianalisis [18].

4. Time

   Library Time merupakan modul utilitas standar Python yang sudah ada jadi tidak memerlukan penginstalan secara eksternal. Modul ini menyediakan fungsi terkait dengan waktu seperti mengukur selang waktu, menghentikan atau menunda eksekusi program. Dalam penelitian ini, fungsi yang digunakan yaitu '`time.sleep(sec)`' yang berfungsi untuk menunda eksekusi program selama jumlah detik (sec) yang ditentukan.

b. Ekstraksi Informasi

Metode yang digunakan pada penelitian ini adalah HTML *Parsing*, berfungsi untuk mengambil konten teks dan mengekstraksi data dari tag atau atribut tertentu. Hal ini digunakan untuk mengakses dan menganalisis konten serta struktur halaman web dengan mengambil data seperti *web scraping*.

Dalam penelitian ini juga melibatkan metode *Cascading Style Sheet* (CSS) *Selector*, yaitu metode yang digunakan untuk mencari elemen HTML pada situs web. Metode ini melibatkan pemilihan elemen berdasarkan kelas CSS, id, nama tag, dan atribut lainnya. CSS Selector memiliki pola yang singkat dan mudah untuk ditulis [9]. *Pseudo code* untuk pengambilan data *web scraping* pada penelitian ini ditunjukkan pada Gambar 3.

Gambar 3. *Pseudo Code Web Data Scraping*

Pseudo code tersebut menggambarkan proses *web scraping* sederhana menggunakan Selenium dan BeautifulSoup untuk mengekstraksi data dari halaman pencarian Tokopedia. Proses *web scraping* dimulai dengan mengatur URL dan membuka *browser*. Informasi berupa nama produk, harga, rating, jumlah barang terjual, lokasi, dan nama toko akan diekstrasi sebanyak 2 kali iterasi dan dimasukkan kedalam data list yang telah dibuat sebelumnya.

Data yang telah diekstraksi akan dimasukkan ke dalam dataframe. Dataframe adalah struktur data

Jurnal Teknologi Dan Sistem Informasi Bisnis
Vol. 5 No. 4 Oktober 2023 Hal. 377-383
http://jurnal.unidha.ac.id/index.php/jteksis

E-ISSN : **2655-8238**
P-ISSN : **2964-2132**

dari library Pandas yang digunakan untuk proses penyimpanan data dalam bentuk tabular. *Dataset* tersebut akan ditransformasi menjadi data yang terstruktur dengan format CSV untuk penyimpanan internal.

2.4 Percobaan *Web Scraping*

Pada percobaan ini, akan dilakukan *web scraping* pada dua web *e-commerce* yang berbeda, yaitu Shopee dan Tokopedia. Kedua *e-commerce* tersebut akan diuji menggunakan kode yang sama untuk menjaga kesamaan dalam proses pengujian. Adapun alamat URL dari *e-commerce* yang diuji ditunjukkan pada Tabel 1.

Tabel 1. URL Web E-commerce

E-commerce	URL
Shopee	https://shopee.co.id/
Tokopedia	https://www.tokopedia.com/

2.5 Pengukuran Waktu Eksekusi

Waktu eksekusi mengacu pada durasi yang diperlukan dalam menjalankan kode *web scraping* dan menyelesaikan tugasnya untuk mengekstraksi data dari situs web. Pengukuran waktu eksekusi dilakukan dengan menginisialisasi variabel start_time sebelum eksekusi kode dan end_time setelah eksekusi kode, kemudian melakukan operasi reduksi (end_time–start_time). *Pseudo code* untuk menampilkan waktu eksekusi *web scraping* pada penelitian ini ditunjukkan pada Gambar 4.

```
start_time = current_time()
run_code()
end_time = current_time()
execution_time = end_time - start_time
print("Execution time:",execution_time, "secs")
```

Gambar 4. *Pseudo Code* Pengukuran Waktu Eksekusi *Web Scraping*

HASIL DAN PEMBAHASAN

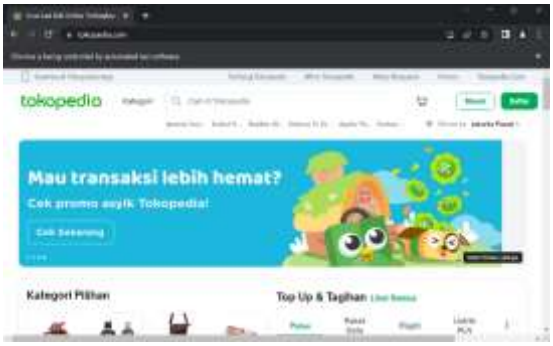
Pada setiap percobaan *web scraping* yang dilakukan, dilakukan pencatatan kemudian dianalisis.

1. Percobaan Menggunakan URL *E-commerce* Shopee

Gambar 5. Halaman Shopee

Gambar 5 merupakan tampilan halaman *e-commerce* Shopee yang terbuka pada webdriver Chrome. Seperti banyak situs web lainnya, Shopee menggunakan mekanisme deteksi bot, termasuk CAPTCHA, untuk mencegah akses otomatis. Mekanisme ini dapat mengganggu kemampuan Selenium WebDriver untuk berinteraksi dengan situs web. Maka dari itu web Shopee tidak dapat terbuka dengan menggunakan cara ini, yaitu Selenium Webdriver, yang mana hal ini mengakibatkan proses *web scraping* tidak dapat dilanjutkan.

2. Percobaan Menggunakan URL *E-commerce* Tokopedia



Gambar 6. Halaman Tokopedia

Gambar 6 menunjukkan halaman Tokopedia yang berhasil terbuka secara aman dengan menggunakan Selenium Webdriver, sehingga memungkinkan dilakukannya proses *web scraping*.

Proses *web scraping* ini melibatkan penggunaan *library* seperti Selenium dan BeautifulSoup untuk melakukan penelusuran serta ekstraksi informasi dari halaman web *e-commerce*. Dilakukan seleksi elemen HTML yang mengandung informasi data yang ingin kita kumpulkan, seperti nama produk, harga, rating, jumlah barang terjual, lokasi, dan nama toko. Setelah itu, menggunakan teknik seperti parsing HTML *Parsing* dan *CSS Selector* untuk mengekstraksi nilai-nilai yang diinginkan dari elemen tersebut. Data yang telah

Jurnal Teknologi Dan Sistem Informasi Bisnis
Vol. 5 No. 4 Oktober 2023 Hal. 377-383
http://jurnal.unidha.ac.id/index.php/jteksis

E-ISSN : **2655-8238**
P-ISSN : **2964-2132**

diekstraksi disimpan ke dalam format yang sesuai, seperti DataFrame atau CSV.

Gambar 7. Hasil *Web Scraping* dalam Bentuk Dataframe

Gambar 8. Hasil *Web Scraping* dalam Bentuk CSV

Gambar 7 dan 8 merupakan hasil *web scraping* yang telah dilakukan dan telah ditrasformasikan ke dalam dataframe dan file CSV.

Tabel 2. Hasil Percobaan Pada Tokopedia

Percobaan ke-	Nama Produk	Banyak Data	Banyaknya Data Duplikasi
1	Oreo	163	14
2	Chitato	163	6
3	Momogi	160	16
4	Tango	160	5
5	Qtela	163	18
6	Malkist	163	13
7	Pringles	166	8
8	Biskuat	163	14
9	Chocolatos	163	16
10	Monde	160	3

Tabel 2 merupakan hasil percobaan *web scraping* produk makanan ringan dengan menggunakan *e-commerce* Tokopedia. Dalam percobaan ini, *web scraping* dilakukan pada 2 halaman, yang tiap halamannya berisi 80 produk. Hasil yang diperoleh menunjukkan variasi jumlah produk yang diambil setiap kali eksekusi. Sepuluh produk tersebut memiliki banyak data yang hampir serupa, dengan jumlah sekitar 160 hingga 166 data. *Web scraping* ini dapat diulang sebanyak yang diperlukan, sesuai dengan kebutuhan jumlah data dan jenis produk yang diinginkan.

Setiap kali melakukan *web scraping*, sering kali terdapat data yang terduplikasi yang perlu dihapus agar hasil analisis menjadi lebih akurat. Hal ini juga terlihat pada Tabel 2, di mana terdapat banyak data yang memiliki duplikat, dengan jumlah data antara 3 hingga 18 data. Oleh karena itu, hasil *web scraping* perlu dilakukan *pre-processing* terlebih dahulu sebelum dilakukan analisis data. *Pre-processing dataset* hasil *web scraping* melibatkan langkah-langkah berikut.

a. Data Cleaning

Dalam tahapan ini, selain menghapus data yang terduplikat, data dengan nilai kosong atau *null* dituliskan dengan 'NaN' pada kolom 'Lokasi' dan 'Toko' juga akan dihapus untuk menjaga konsistensi data. Sementara itu dalam kolom 'Rating' nilai kosong diisi dengan nilai 0 karena memiliki tipe data *float*.

Regular Expression (Regex) digunakan untuk mengubah pola data dalam kolom 'Harga' dan 'Terjual' dengan menghilangkan karakter yang tidak diperlukan sehingga hanya meninggalkan nilai angka.

b. Data Transformation

Dalam tahapan ini, tipe data *string* dari kolom 'Harga' dan 'Terjual' akan dikonversi menjadi tipe data *integer* dimana berisi angka yang dapat dioperasikan sesuai dengan kebutuhan analisis data.

c. Data Reduction

Dalam tahapan ini, *dataset* yang memiliki 7 fitur akan di seleksi untuk mengurangi dimensi data yang diperlukan untuk menyederhanakan analisis dan mengurangi resiko *overfitting*. Fitur atau kolom yang akan dihilangkan pada *dataset* ini yaitu kolom pertama yang berisi urutan baris yang tidak diperlukan.

Hasil *pre-processing* ini dapat dilihat pada Gambar 9.

Gambar 9. *Dataset* Hasil *Pre-processing*

Dengan melakukan *pre-processing*, data akan menjadi lebih akurat dan dapat memberikan hasil analisis yang lebih valid serta dapat dipastikan bahwa hanya data unik yang digunakan dalam analisis, sehingga mencegah adanya pengaruh yang tidak diinginkan.

3. Pengukuran Waktu Eksekusi

Tabel 3. Pengukuran Waktu Eksekusi

Percobaan ke-	Nama Produk	Waktu
1	Oreo	1m 0.6s
2	Chitato	1m 0.6s
3	Momogi	1m 0.4s
4	Tango	1m 0.5s
5	Qtela	1m 0.4s
6	Malkist	1m 0.5s
7	Pringles	1m 0.6s

Jurnal Teknologi Dan Sistem Informasi Bisnis
Vol. 5 No. 4 Oktober 2023 Hal. 377-383
http://jurnal.unidha.ac.id/index.php/jteksis

E-ISSN : **2655-8238**
P-ISSN : **2964-2132**

8	Biskuat	1m 0.8s
9	Chocolatos	1m 0.6s
10	Monde	1m 0.5s
Avg	**-**	**1m 0.5s**

Pada Tabel 3, ditunjukkan waktu yang diperlukan untuk setiap eksekusi pada setiap produk. Rata-rata waktu yang dibutuhkan adalah sekitar 1 menit 0,5 detik. Waktu dalam *web scraping* dipengaruhi oleh kecepatan internet yang digunakan. Kondisi yang tidak stabil dari koneksi internet dapat mempengaruhi proses membuka dan mengambil data dari situs web.

Penggunaan *library* Time dengan fungsi `'time.sleep(sec)'` berguna untuk menunda waktu eksekusi program. Fungsi ini dapat digunakan untuk memberikan jeda waktu antara perintah-perintah dalam program. Dengan menggunakan fungsi ini dapat dipastikan bahwa halaman situs web telah terbuka sepenuhnya sebelum program melanjutkan eksekusi baris selanjutnya. Hal ini membantu mengatasi masalah ketidakstabilan koneksi internet dan memastikan bahwa proses *web scraping* berjalan dengan baik.

Bedasarkan penelitian yang dilakukan, diperoleh temuan bahwa metode HTML *Parsing* dan CSS *Selector* tidak dapat digunakan untuk *web scraping* pada *e-commerce* Shopee. Hal ini dikarenakan pada Shopee memiliki mekanisme deteksi bot, termasuk CAPTCHA, yang bertujuan untuk mencegah akses otomatis. Namun, metode tersebut berhasil digunakan untuk *web scraping* produk pada *e-commerce* Tokopedia. Pada *web scraping* ini, HTML *Parsing* digunakan untuk mengekstrak informasi, sedangkan CSS *Selector* digunakan untuk memilih elemen yang akan digunakan dalam proses *web scraping*.

SIMPULAN

Pada penelitian ini dilakukan percobaan pengumpulan data melalui *web scraping* pada *e-commerce* Shopee dan Tokopedia Indonesia. *Web scraping* dengan menggunakan HTML *Parsing* dan CSS *Selector* tidak dapat digunakan pada Shopee, namun berhasil pada Tokopedia. Percobaan *web scraping* dilakukan sebanyak 10 kali pada 2 halaman produk Tokopedia. Pada setiap kali percobaannya, jumlah data yang diperoleh hampir sama, yaitu sekitar 160-166. Waktu rata-rata yang diperlukan untuk pengambilan data adalah 1 menit 0,5 detik.

Hasil dari *web scraping* ini memiliki kekurangan berupa adanya duplikasi data yang dapat mempengaruhi hasil analisis data nantinya. Oleh karena itu, sebelum data tersebut digunakan untuk proses analisis, perlu dilakukan *pre-processing* guna menghasilkan data yang bersih dan siap digunakan.

DAFTAR PUSTAKA

[1] P. Thota and E. Ramez, "Web Scraping of COVID-19 News Stories to Create Datasets for Sentiment and Emotion Analysis," *ACM Int. Conf. Proceeding Ser.*, pp. 306–314, 2021, doi: 10.1145/3453892.3461333.

[2] E. S. Sulistiyawati and A. Widayani, "Marketplace Shopee Sebagai Media Promosi Penjualan UMKM di Kota Blitar," *J. Pemasar. Kompetitif*, vol. 4, no. 1, p. 133, 2020, doi: 10.32493/jpkpk.v4i1.7087.

[3] A. Ahdiat, "Ini Pertumbuhan Pengunjung Tokopedia sampai Kuartal II 2022," *databoks*, 2022. https://databoks.katadata.co.id/datapublish/2022/11/21/ini-pertumbuhan-pengunjung-tokopedia-sampai-kuartal-ii-2022

[4] T. Rizaldi and H. Arief, "Perbandingan Metode Web Scraping Menggunakan CSS Selector dan Xpath Selector," *Teknika*, vol. 6, no. 1, pp. 43–46, 2017, doi: 10.34148/teknika.v6i1.56.

[5] M. Afdhal, V. Ariandi, and R. Rita, "Memprediksi Penjualan Pada Toko Hanifah Metode C.45," *J. Teknol. Dan Sist. Inf. Bisnis*, vol. 4, no. 2, pp. 248–255, 2022, doi: 10.47233/jteksis.v4i1.460.

[6] N. N. Hasanah and A. S. Purnomo, "Implementasi Data Mining Untuk Pengelompokan Buku Menggunakan Algoritma K-Means Clustering (Studi Kasus : Perpustakaan Politeknik LPP Yogyakarta)," *J. Teknol. Dan Sist. Inf. Bisnis*, vol. 4, no. 2, pp. 300–311, 2022, doi: 10.47233/jteksis.v4i2.499.

[7] R. Gunawan, A. Rahmatulloh, I. Darmawan, and F. Firdaus, "Comparison of Web Scraping Techniques : Regular Expression, HTML DOM and Xpath," vol. 2, no. IcoIESE 2018, pp. 283–287, 2019, doi: 10.2991/icoiese-18.2019.50.

[8] I. Onyenwe, E. Onyedinma, C. Nwafor, and O. Agbata, "Developing Products Update-Alert System for E-Commerce Websites Users using Html Data and Web Scraping Technique," *Int. J. Nat. Lang. Comput.*, vol. 10, no. 5, pp. 01–07, 2021, doi: 10.5121/ijnlc.2021.10501.

[9] E. Uzun, "A regular expression generator based on CSS selectors for efficient extraction from HTML pages," *Turkish J. Electr. Eng. Comput. Sci.*, vol. 28, no. 6, pp. 3389–3401, 2020, doi: 10.3906/ELK-2004-67.

[10] M. El Asikri, S. Krit, and H. Chaib, "Using Web Scraping In A Knowledge Environment To Build Ontologies Using Python And Scrapy," *Eur. J. Transl. Clin. Med.*, vol. 07, no. 03, pp. 433–442, 2020, [Online]. Available: https://www.researchgate.net/publication/346215371

[11] D. B. Pratama, A. Sofwan, and Y. A. A. Soetrisno, "Implementasi Teknik Web Scraping dan Fitur Data Eksternal pada Sistem Informasi Dosen Penelitian dan Pengabdian Dosen Fakultas Teknik Universitas Diponegoro," vol. 10, no. 2, pp. 292–299, 2021.

[12] L. C. Dewi, Meiliana, and A. Chandra, "Social media web scraping using social media developers API and regex," *Procedia Comput. Sci.*, vol. 157, pp. 444–449, 2019, doi: 10.1016/j.procs.2019.08.237.

[13] S. Han and C. K. Anderson, "Web Scraping for Hospitality Research: Overview, Opportunities, and Implications," *Cornell Hosp. Q.*, vol. 62, no. 1, pp. 89–104, 2021, doi: 10.1177/1938965520973587.

[14] M. Elveny, S. M. Hardi, I. Jaya, and P. Gundari, "Web-based E-Commerce Products Grouping," *J. Phys. Conf. Ser.*, vol. 1898, no. 1, 2021, doi: 10.1088/1742-6596/1898/1/012018.

[15] A. Purnomo, "Impementasi Web Scraping Pada OJS Dengan Metode CSS Selector," *REDOLUSI Rekayasa Tek. Inform. dan Inf.*, vol. 3, no. 2, pp. 176–191, 2022.

[16] S. Nyamathulla, P. Ratnababu, N. S. Shaik, and B. L. N, "A Review on Selenium Web Driver with Python," *Ann. Rom. Soc. Cell Biol.*, vol. 25, no. 4, pp. 16760–16768, 2021, [Online]. Available: http://annalsofrscb.ro

[17] M. Levi, H. N. Palit, and S. Rostianingsih, "Perbandingan

E-ISSN : **2655-8238**
P-ISSN : **2964-2132**

Performa Tools Web Scraping pada Website dengan Data Statis dan Dinamis," 2020.

[18] A. Nagpal and G. Gabrani, "Python for Data Analytics, Scientific and Technical Applications," *Proc. - 2019 Amity Int. Conf. Artif. Intell. AICAI 2019*, pp. 140–145, 2019, doi: 10.1109/AICAI.2019.8701341.