

**PENERAPAN *WEB CRAWLING* MENGGUNAKAN ALGORITMA *RAITA*
PADA PENGUMPULAN INFORMASI PENGINAPAN DI BERASTAGI**

SKRIPSI

Diajukan Untuk Melengkapi Tugas Dan Memenuhi Syarat Memperoleh Ijazah
Sarjana Komputer

KANIA ULFA LUBIS

181401059



**PROGRAM STUDI S-1 ILMU KOMPUTER
FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI
UNIVERSITAS SUMATERA UTARA**

MEDAN

2024

**PENERAPAN *WEB CRAWLING* MENGGUNAKAN ALGORITMA *RAITA*
PADA PENGUMPULAN INFORMASI PENGINAPAN DI BERASTAGI**

SKRIPSI

Diajukan Untuk Melengkapi Tugas Dan Memenuhi Syarat Memperoleh Ijazah
Sarjana Komputer

KANIA ULFA LUBIS

181401059



**PROGRAM STUDI S-1 ILMU KOMPUTER
FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI
UNIVERSITAS SUMATERA UTARA**

MEDAN

2024

PERSETUJUAN

Judul	:	PENERAPAN <i>WEB CRAWLING</i> MENGGUNAKAN ALGORITMA <i>RAITA</i> PADA PENGUMPULAN INFORMASI PENGINAPAN DI BERASTAGI
Kategori	:	SKRIPSI
Nama	:	KANIA ULFA LUBIS
Nomor Induk Mahasiswa	:	181401059
Program Studi	:	SARJANA (S-1) ILMU KOMPUTER
Fakultas	:	ILMU KOMPUTER DAN TEKNOLOGI INFORMASI UNIVERSITAS SUMATERA UTARA

Telah diuji dan dinyatakan lulus di Medan, 29 April 2024

Dosen Pembimbing II



Dr. Jos Timanta Tarigan, S.Kom., M.Sc
NIP. 198501262015041001

Dosen Pembimbing I



Dr. Amalia S.T., M.T.
NIP. 197812212014042001

Diketahui/Disetujui Oleh
Ketua Program Studi S-1 Ilmu Komputer



Dr. Amalia S.T., M.T.
NIP. 197812212014042001

PERNYATAAN

**PENERAPAN *WEB CRAWLING* MENGGUNAKAN ALGORITMA *RAITA* PADA
PENGUMPULAN INFORMASI PENGINAPAN DI BERASTAGI**

SKRIPSI

Saya mengakui bahwa skripsi ini adalah hasil karya saya sendiri, kecuali beberapa kutipan dan ringkasan yang masing-masing telah disebutkan sumbernya.

Medan, 17 Februari 2024

A handwritten signature in black ink, appearing to read 'Kania', with a stylized flourish underneath.

Kania Ulfa Lubis

181401059

KATA PENGANTAR

Puji dan syukur penulis panjatkan ke hadirat Allah SWT, karena dengan limpahan rahmat dan petunjuk-Nya, penulis dapat menyelesaikan penulisan skripsi ini sebagai bagian dari persyaratan untuk meraih gelar Sarjana Komputer di Program Studi S-1 Ilmu Komputer, Fakultas Ilmu Komputer dan Teknologi Informasi, Universitas Sumatera Utara.

Penulis ingin menyatakan rasa hormat dan terima kasih yang sangat besar kepada:

1. Bapak Dr. Muryanto Amin, S.Sos., M.Si. selaku Rektor Universitas Sumatera Utara.
2. Ibu Dr. Maya Silvi Lydia B.Sc., M.Sc. selaku Dekan Fakultas Ilmu Komputer dan Teknologi Informasi Universitas Sumatera Utara.
3. Ibu Dr. Amalia ST., M.T. selaku Ketua Program Studi S-1 Ilmu Komputer Universitas Sumatera Utara sekaligus Dosen Pembimbing I yang telah memberikan arahan, kritik, dan saran kepada penulis selama proses pengerjaan skripsi.
4. Bapak Dr. Jos Timanta Tarigan, S.Kom., M.Sc. selaku Dosen Pembimbing II yang telah memberikan arahan, kritik, dan saran yang bermanfaat kepada penulis.
5. Ibu Hayatunnufus, S.Kom, M.Kom selaku Dosen Penguji I yang telah memberi arahan dan saran yang bermanfaat.
6. Ibu Sri Melvani Hardi S.Kom., M.Kom. selaku Sekretaris Program Studi S-1 Ilmu Komputer Universitas Sumatera Utara sekaligus Dosen Penguji II yang telah memberi arahan dan saran yang bermanfaat.
7. Seluruh dosen dan pegawai Program Studi S-1 Ilmu Komputer Fakultas Ilmu Komputer dan Teknologi Informasi Universitas Sumatera Utara.
8. Kedua orang tua tercinta, Ayahanda Ikhwan Ispianda Lubis dan Ibunda Nana Ashary Siregar yang selalu memberikan doa, dukungan, motivasi dan semangat kepada penulis sehingga penulis dapat menyelesaikan skripsi ini.
9. Saudara-saudara kandung penulis, Karina Lubis dan Kandy Farisi Lubis serta keluarga besar yang senantiasa memberikan dukungan selama proses penyelesaian skripsi.

10. Sahabat seperjuangan Ayu Afreza Siregar yang telah menemani penulis sejak awal kuliah hingga saat ini dan selalu memberikan dukungan, bantuan dan semangat kepada penulis.
11. Seluruh teman Stambuk 2018 S-1 Ilmu Komputer Universitas Sumatera Utara yang telah bersama selama masa perkuliahan.
12. Dan semua pihak yang telah membantu dalam penulisan skripsi ini yang tidak dapat disebutkan satu per satu.

Medan, 17 Februari 2024

A handwritten signature in black ink, appearing to read 'Kanis', with a stylized flourish underneath.

Penulis

ABSTRAK

Dalam era digital, kemudahan akses informasi penginapan menjadi sangat penting. Metode pengumpulan data otomatis melalui *web crawling* menjadi alternatif, namun tantangan terkait akurasi dan efisiensi tetap ada. Penelitian ini mengeksplorasi penggunaan algoritma *Raita* untuk meningkatkan akurasi dan efisiensi pengumpulan data penginapan di Berastagi. Algoritma *Raita* adalah algoritma pencocokan *string* dengan susunan karakter pada *string* yang dicocokkan dengan jumlah atau urutan karakter dalam *string* yang sama. Algoritma *Raita* membandingkan karakter terakhir dari pola karakter yang terdapat paling kanan *window* hingga kecocokan terjadi. Dengan memilih *seed* URL yang tepat, mengembangkan *web crawler*, dan menganalisis struktur web serta karakteristik informasi penginapan, penelitian ini menemukan bahwa algoritma *Raita* dapat mengurangi informasi yang redundan dan mempercepat proses pengumpulan data. Hasil dari pengujian sistem dengan kata kunci “hotel” didapatkan hasil 15 hotel, kata kunci “*villa*” berjumlah 40 *villa*, kata kunci “*homestay*” berjumlah 11 *homestay*, kata kunci “*cottage*” berjumlah 3 *cottage*, dan pengujian tanpa kata kunci menghasilkan seluruh penginapan di Berastagi yang terdapat pada *website* tiket.com yaitu sebanyak 106 penginapan. Dari 106 informasi penginapan yang telah diekstraksi, persentase informasi penginapan yang lengkap adalah 54% dan yang tidak lengkap adalah 46%. Keterangan dari informasi persentase ini adalah kelengkapan data informasi meliputi nama, kesan, *rating*, harga, dan alamat penginapan.

Kata kunci: *algoritma Raita, informasi penginapan, tiket.com, web crawling.*

IMPLEMENTATION OF WEB CRAWLING USING RAITA ALGORITHM FOR LODGING INFORMATION GATHERING IN BERASTAGI

ABSTRACT

In the digital era, easy access to lodging information is very important. Automated data collection methods through web crawling are an alternative, but challenges related to accuracy and efficiency remain. This research explores the use of Raita algorithm to improve the accuracy and efficiency of lodging data collection in Berastagi. Raita algorithm is a string matching algorithm with the arrangement of characters in the string being matched with the number or order of characters in the same string. Raita algorithm compares the last character of the character pattern contained in the rightmost window until a match occurs. By selecting the right seed URL, developing a web crawler, and analyzing the web structure and characteristics of lodging information, this research found that Raita's algorithm can reduce redundant information and speed up the data collection process. The results of testing the system with the keyword "hotel" resulted in 15 hotels, the keyword "villa" amounted to 40 villas, the keyword "homestay" amounted to 11 homestays, the keyword "cottage" amounted to 3 cottages, and testing without keywords resulted in all lodgings in Berastagi found on the tiket.com website, namely 106 lodgings. Of the 106 lodging information that has been extracted, the percentage of complete lodging information is 54% and the incomplete is 46%. The description of this percentage information is the completeness of information data including name, impression, rating, price, and address of the inn.

Keywords: *Raita algorithm, accommodation information, tiket.com, web crawling.*

DAFTAR ISI

PERSETUJUAN	i
PERNYATAAN	ii
KATA PENGANTAR	iii
ABSTRAK	v
ABSTRACT	vi
DAFTAR ISI	vii
DAFTAR TABEL	ix
DAFTAR GAMBAR	x
DAFTAR LAMPIRAN	xi
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	3
1.3 Batasan Masalah	4
1.4 Tujuan Penelitian	4
1.5 Manfaat Penelitian	4
1.6 Metodologi Penelitian	4
BAB II LANDASAN TEORI	6
2.1 Website	6
2.1.1 HTML (<i>HyperText Markup Language</i>)	6
2.1.2 CSS (<i>Cascading Style Sheets</i>)	6
2.2 tiket.com	7
2.3 Web Crawling	7
2.3.1 URL (<i>Uniform Resource Locator</i>)	8
2.3.2 Seed URL	8
2.3.3 Cara Kerja Web Crawling	8
2.4 Web Scraping	9
2.5 String Matching	10
2.6 Algoritma Raita	10
2.6.1 Fase <i>Preprocessing</i>	11
2.6.2 Fase Pencarian	12
2.7 Penelitian yang Relevan	13

BAB III ANALISIS DAN PERANCANGAN SISTEM	17
3.1 Arsitektur Umum	17
3.2 Proses Algoritma <i>Raita</i>	19
3.3 Proses <i>Web Crawling</i>	21
3.4 Proses <i>Testing</i>	25
3.5 Perancangan Sistem	25
BAB IV IMPLEMENTASI DAN PENGUJIAN SISTEM	26
4.1 Implementasi Sistem	26
4.1.1. Spesifikasi Perangkat Keras	26
4.1.2. Spesifikasi Perangkat Lunak	26
4.2 Implementasi <i>Website</i> tiket.com	26
4.3 Implementasi Proses Filter Algoritma <i>Raita</i>	27
4.4 Pengujian Sistem	30
4.4.1. Pengujian Pertama	31
4.4.2. Pengujian Kedua	33
4.4.3. Pengujian Ketiga.....	39
4.4.4. Pengujian Keempat.....	42
4.4.5. Pengujian Kelima.....	43
BAB V KESIMPULAN DAN SARAN	46
5.1 Kesimpulan	46
5.2 Saran	46
DAFTAR PUSTAKA	48
LAMPIRAN.....	A-1

DAFTAR TABEL

Tabel 2.1 Indeks <i>Pattern</i>	11
Tabel 2.2 Hasil <i>BmBc (Pattern)</i>	12
Tabel 3.1 <i>Source Code</i> Proses Algoritma <i>Raita</i>	20
Tabel 3.2 <i>Source Code</i> Proses <i>Web Crawling</i>	21
Tabel 4.1 Indeks <i>Pattern</i>	29
Tabel 4.2 Hasil <i>BmBc</i>	29
Tabel 4.3 Hasil Pengujian Pertama	32
Tabel 4.4 Hasil Pengujian Kedua	34
Tabel 4.5 Hasil Pengujian Ketiga	40
Tabel 4.6 Hasil Pengujian Keempat	42
Tabel 4.7 Persentase Infomasi Penginapan Hasil <i>Crawling</i>	45

DAFTAR GAMBAR

Gambar 2.1 Alur Proses <i>Web Crawling</i>	9
Gambar 2.2 Langkah-Langkah Pencarian pada Teks	12
Gambar 3.1 Arsitektur Umum	17
Gambar 3.2 <i>Flowchart</i> Sistem	18
Gambar 4.1 Halaman tiket.com	27
Gambar 4.2 <i>Google Trends</i>	27
Gambar 4.3 Pencocokan <i>String</i> Algoritma <i>Raita</i> (1)	28
Gambar 4.4 Pencocokan <i>String</i> Algoritma <i>Raita</i> (2)	28
Gambar 4.5 Langkah-Langkah Pencarian pada Teks	30
Gambar 4.7 Proses Pengujian Pertama	31
Gambar 4.8 Proses Pengujian Kedua	34
Gambar 4.9 Proses Pengujian Ketiga	40
Gambar 4.10 Proses Pengujian Keempat	42
Gambar 4.11 Informasi Penginapan Berastagi dari <i>Website</i> tiket.com (1)	43
Gambar 4.12 Informasi Penginapan Berastagi dari <i>Website</i> tiket.com (2)	44
Gambar 4.13 Informasi Penginapan Berastagi dari <i>Website</i> tiket.com (3)	44
Gambar 4.14 Informasi Penginapan Berastagi dari <i>Website</i> tiket.com (4)	45

DAFTAR LAMPIRAN

Lampiran <i>Listing Program</i>	A-1
---------------------------------------	-----

BAB I

PENDAHULUAN

1.1 Latar Belakang

Informasi merupakan hal yang dibutuhkan dalam berbagai aspek kehidupan. Informasi merupakan sebuah data yang dikelola menjadi sesuatu yang lebih bernilai tinggi bagi penerima guna untuk membantu membuat sebuah pengambilan keputusan (Tukino, 2020). Saat ini, internet merupakan sumber terbaik untuk memperoleh informasi karena beragamnya informasi yang dapat diakses dengan mudah. Salah satu informasi yang dapat kita cari adalah informasi penginapan di Berastagi dari berbagai *website*, seperti tiket.com, Agoda, Traveloka, PegiPegi, dan Booking.com.

Untuk memperoleh informasi secara efektif dan efisien, *web crawling* atau teknik pengambilan data otomatis dari *website* telah menjadi alat yang krusial. *Web crawler*, atau dikenal juga sebagai *web spider* atau *web robot*, adalah suatu program atau *script* untuk melakukan penjelajahan dan pengambilan informasi dari suatu *website*. Umumnya *web crawler* menelusuri halaman-halaman web dalam bahasa *markup* HTML atau XHTML (Pradana, 2022).

Web scraping, *web harvesting*, atau *web data extraction* merupakan proses pengumpulan data web terstruktur secara otomatis yang dilakukan oleh aplikasi khusus atau kode pemrograman. Data yang terkumpul bisa berasal dari ribuan, jutaan bahkan milyaran data di dunia maya (Yondra et al., 2022).

Penelitian yang dilakukan oleh (Jeong et al., 2020) dengan judul “*Deep Learning-based Tourism Recommendation System using Social Network Analysis*” bertujuan untuk menghasilkan rekomendasi tempat wisata di Korea yang sesuai dengan pengguna dengan memanfaatkan kriteria rekomendasi yang jelas. Penelitian ini juga menerapkan *web crawling* untuk mengumpulkan 1.000 data dari *website* Naver dan blog dengan menggunakan kata kunci. Perbedaannya, data yang diambil hanya berupa nama tempat wisata dengan frekuensi tertinggi.

Penelitian yang dilakukan oleh (Shamrat et al., 2020) yang berjudul “*An Effective Implementation of Web Crawling Technology to Retrieve Data from the World Wide Web (WWW)*” juga menerapkan *web crawling* untuk mengekstrak data dari *website*, akan tetapi lebih terfokus pada mengusulkan sebuah *web crawler* yang

menggunakan serangkaian *query* dari daftar kata kunci untuk melakukan *crawling* melalui halaman web daripada mengindeks semua halaman web di internet.

Pada penelitian yang dilakukan oleh (Hien et al., 2020) yang berjudul “*Web Crawler: Design and Implementation for Extracting Article-Like Contents*”, sebuah model *web crawler* dirancang dan dikembangkan yang berfungsi untuk mengekstrak dari 495 *website* menjadi konten yang mirip dengan artikel. Perbedaannya, model ini menggunakan model SVM (*Support Vector Machine*), *Grid Search*, dan *Cross Validation*. Dari ketiga model tersebut, *Grid Search* merupakan model yang paling optimal dengan akurasi sebesar 92,01%

Penelitian yang dilakukan oleh (Ou et al., 2023) dengan judul “*Design and Implementation of a Recommendation System for Buying Fresh Foods Online Based on Web Crawling*” menggunakan *web crawling* untuk mengumpulkan informasi dari tiga *website* makanan segar, dimana halaman web tersebut berisi informasi produk secara detail. Perbedaannya, penelitian ini menggunakan *library* lanjutan dari tesaurus *Jieba* untuk membantu mengenali kata-kata yang digunakan dalam bahasa Tionghoa.

Penelitian yang dilakukan oleh (Prismana et al, 2020) yang berjudul “*The Design and Implementation of Web Crawler Distributed News Domain Detection System*” menggunakan sebuah *crawler* yang difokuskan pada distribusi untuk mengumpulkan data berita secara efisien. Penelitian ini menggunakan *Get Metadata* untuk mendeteksi adanya judul berita yang sama. Jika ada, berita akan diperiksa kembali untuk menghasilkan data berita yang relevan.

Pada tahun 2021, penelitian dengan judul “Penerapan *Web Crawling* Menggunakan Algoritma *Horspool* sebagai Filter pada Pengumpulan Informasi Restoran di Kota Medan” oleh Ikhwan Arjuna Lubis. Pengujian sistem dilakukan sebanyak empat kali yaitu dengan menggunakan kata kunci “ayam penyet”, “nasi goreng”, dan “mie aceh” dengan tingkat akurasi data sebesar 100% dan pengujian untuk menentukan jumlah persentase restoran yang memiliki *review* dan tidak memiliki *review*.

Berdasarkan hasil penelitian yang dilakukan oleh Dedi Rudi Bawanto dan Nidia Rosmawanti pada tahun 2017 dengan judul “Perbandingan Algoritma *Binary Search* dan *Raita* dalam Pencarian Data”, yang membandingkan keakuratan kedua algoritma tersebut berdasarkan kecepatan dan ketepatan dalam pencarian, algoritma

Raita lebih cepat 3% dibandingkan algoritma *Binary Search* dalam pencarian data dari keseluruhan data sampel yang diuji dengan total waktu 2,5116 ms.

Algoritma *Raita* merupakan algoritma pencocokan *string* dengan susunan karakter pada *string* yang dicocokkan dengan jumlah atau urutan karakter dalam *string* yang sama. Algoritma *Raita* membandingkan karakter terakhir dari pola karakter paling kanan *window* sampai kecocokan terjadi. Setelah itu dilanjutkan dengan membandingkan karakter pertama dari pola paling kiri *window*. Jika cocok, karakter tengah pada pola di *window* dibandingkan juga. Jika benar-benar cocok, algoritma akan membandingkan karakter lain, mulai dari karakter kedua hingga kedua terakhir dan mungkin membandingkan dengan karakter tengah lagi (Ginting, 2019).

Berdasarkan uraian di atas, karena banyaknya informasi yang tersedia pada *website*, maka diperlukan sebuah sistem untuk pencarian dan pengambilan data yakni *web crawling*. *Web crawling* bertujuan untuk melakukan pencarian dan pengambilan informasi dengan cepat melalui proses *crawling* data pada *website* tertentu. Pada penelitian ini, penulis menerapkan sebuah sistem *web crawling* untuk mengumpulkan informasi penginapan di Berastagi pada *website* tiket.com dengan beberapa kata kunci sebagai *input* untuk pencocokan *string* dengan menggunakan algoritma *Raita* agar pengambilan informasi sesuai dengan kata kunci.

1.2 Rumusan Masalah

Dalam mengumpulkan informasi tentang penginapan, meskipun masih memungkinkan untuk melakukannya secara manual, penggunaan teknik *web crawling* menjadi pilihan yang lebih efisien. Hal ini disebabkan oleh kecepatan dalam memperoleh informasi yang dimiliki oleh *web crawling*. Meskipun telah tersedia layanan pencarian dan pemesanan penginapan otomatis seperti di tiket.com, terdapat tantangan dalam menyaring informasi yang relevan dan sesuai dengan kebutuhan. Walaupun fitur pencarian lanjutan memungkinkan untuk menetapkan kriteria, masih terdapat kendala dalam menemukan informasi dengan cepat dan efisien jika dilakukan secara manual. Maka dari itu dilakukan pencarian secara otomatis dengan menerapkan teknik *web crawling* dan algoritma *Raita* sebagai algoritma pencocokan *string*.

1.3 Batasan Masalah

Batasan masalah pada penelitian ini adalah sebagai berikut:

1. Data penginapan yang akan diambil berasal dari *website* tiket.com.
2. Data yang diambil yakni data pada November hingga Desember 2023.
3. Metode yang digunakan untuk mengekstrak data adalah *web scraping*.
4. Algoritma *Raita* sebagai filter agar data yang diekstrak lebih terfokus.
5. *Output* sistem berupa dokumen *Excel* yang berisi data yang telah diekstrak.
6. Menggunakan *library Selenium*.
7. IDE yang digunakan untuk membuat sistem adalah *Jupyter Notebook* dengan bahasa pemrograman *Python 3.12*.

1.4 Tujuan Penelitian

Tujuan penelitian ini adalah menerapkan sistem yang dapat mengekstrak informasi penginapan di Berastagi dari *website* tiket.com dengan menggunakan teknik *web crawling* dan algoritma *Raita* sebagai filter. Hal ini dilakukan dengan tujuan agar data yang diekstrak lebih terfokus pada kata kunci yang relevan.

1.5 Manfaat Penelitian

Manfaat yang diharapkan dapat diperoleh dari penelitian ini adalah:

1. Digunakan sebagai *tools*/alat untuk mencari dan mengekstraksi data dari web tertentu.
2. Hasil dari sistem dapat digunakan oleh data *scientist* atau data *analyst* sebagai bahan penelitian selanjutnya.
3. Sebagai bahan perbandingan dan menjadi referensi dalam pengembangan sistem *crawler* bagi penulis selanjutnya yang tertarik dengan *web crawler*.

1.6 Metodologi Penelitian

Metode penelitian yang dilakukan dalam penelitian ini adalah:

1. Studi Pustaka

Pada tahap ini, penelitian dimulai dengan mencari referensi dari berbagai sumber dan melakukan peninjauan pustaka melalui buku, jurnal, *e-book*, artikel ilmiah, makalah ataupun situs internet yang berhubungan dengan *web crawling* dan algoritma *Raita*.

2. Analisa Sistem

Pada tahap ini, dilakukan analisis dan pengumpulan data yang dibutuhkan untuk penelitian dengan menggunakan *web crawling* dan algoritma *Raita*.

3. Perancangan dan Implementasi Sistem

Pada tahap ini, dilakukan perancangan berdasarkan hasil analisis sistem dan melakukan implementasi dari perancangan yang telah dibuat dengan menggunakan bahasa pemrograman *Python*.

4. Pengujian Sistem

Pada tahap ini, proses pengujian dilakukan untuk mengetahui apakah sistem yang telah dirancang sudah berjalan dengan baik dan sesuai dengan kebutuhan yang telah ditentukan.

5. Dokumentasi

Pada tahap ini, penelitian yang telah dilakukan didokumentasikan dalam bentuk skripsi, mulai dari tahap analisa hingga pengujian.

Sistematika dalam penulisan skripsi ini terdiri dari beberapa bagian utama sebagai berikut:

BAB I : PENDAHULUAN

Pada Bab satu, berisi tentang latar belakang penelitian, rumusan masalah, tujuan penelitian, batasan masalah, manfaat penelitian, metodologi penelitian, dan sistematika penulisan.

BAB II : LANDASAN TEORI

Pada Bab kedua, terdiri dari teori-teori yang diperlukan untuk memahami permasalahan pada penelitian ini. Teori-teori yang berhubungan dengan *web crawling*, *web scraping*, dan algoritma *Raita*.

BAB III : ANALISIS DAN PERANCANGAN SISTEM

Pada Bab ketiga, menjelaskan tentang analisis dan perancangan sistem yang akan dibangun dengan menggunakan *web crawling* dan algoritma *Raita*. Pada bab ini dijabarkan tentang arsitektur umum dan perancangan sistem.

BAB IV : IMPLEMENTASI DAN PENGUJIAN SISTEM

Pada Bab keempat, menampilkan hasil implementasi sistem yang telah dibangun sesuai dengan analisis dan perancangan, serta pengujian terhadap sistem untuk memastikan bahwa sistem berjalan sesuai dengan tujuan yang telah ditetapkan dalam penelitian ini.

BAB V : KESIMPULAN DAN SARAN

Pada Bab kelima, berisi tentang kesimpulan yang diperoleh dari hasil penelitian yang telah dilakukan dan saran untuk pengembangan penelitian selanjutnya.

BAB II

LANDASAN TEORI

2.1 Website

Menurut Elgamar (2020), *website* adalah suatu media yang terdiri dari beberapa halaman yang saling berkaitan satu sama lain, dan berfungsi sebagai media untuk menampilkan suatu informasi, baik berbentuk gambar, video, teks, suara, ataupun gabungan dari semuanya. *Website* bersifat *multiplatform* yang artinya dapat dibuka dari segala perangkat atau *device* yang terhubung dengan jaringan internet.

Pengertian *website* menurut (Romadhon et al., 2021) adalah sekumpulan informasi atau *page* yang bisa diakses lewat jalur internet. Setiap orang di berbagai tempat dan segala waktu bisa menggunakannya selama terhubung secara *online* di jaringan internet. Secara teknis, *website* adalah kumpulan dari *page*, yang tergabung ke dalam suatu *domain* atau *subdomain* tertentu.

Menurut Sebok, Vermat, dan tim dalam Achmad Munawar (2022) *website* adalah kumpulan halaman yang saling terhubung yang di dalamnya terdapat beberapa item seperti dokumen dan gambar yang tersimpan di dalam *web server*.

2.1.1 HTML (*HyperText Markup Language*)

HTML adalah singkatan dari *HyperText Markup Language* yaitu bahasa pemrograman standar yang digunakan untuk membuat sebuah halaman web, yang kemudian dapat diakses untuk menampilkan berbagai informasi di dalam sebuah penjelajah web internet (*browser*). HTML dapat juga digunakan sebagai *link* antara file-file dalam situs atau dalam komputer dengan menggunakan *localhost*, atau *link* yang menghubungkan antar situs dalam dunia internet. Fungsi HTML adalah untuk mengelola serangkaian data dan informasi sehingga suatu dokumen dapat diakses dan ditampilkan di internet melalui layanan web (Lengkong et al., 2019).

2.1.2 CSS (*Cascading Style Sheets*)

CSS atau *Cascading Style Sheets* adalah dokumen yang berisi aturan-aturan yang digunakan untuk memisahkan isi dari tata letak dalam halaman-halaman web. Dengan menggunakan CSS, pengguna dapat menentukan gaya atau tampilan dari elemen-elemen di halaman web secara terpisah dari konten

sebenarnya. CSS memungkinkan pengguna untuk menciptakan “*template*” gaya yang dapat digunakan secara konsisten di seluruh situs web, dan memungkinkan penulisan kode yang lebih mudah untuk desain halaman web. Salah satu keunggulan CSS adalah kemampuannya untuk menciptakan halaman web yang memiliki tampilan yang konsisten dan responsif, bahkan saat dilihat pada layar dengan resolusi yang berbeda, tanpa harus mengandalkan penggunaan tabel seperti pada HTML klasik (Febry Ahmad, 2021).

2.2 tiket.com

PT Global Tiket Network merupakan perusahaan yang memiliki fokus pada sebuah web dengan situs tiket.com dengan fitur *booking* dan *ticketing online* yang memberikan pelayanan reservasi tiket. Dilansir oleh tiket.com pada tahun 2017, tiket.com berafiliasi dengan Blibli dan pada tahun 2021 menjadi anak perusahaan terkonsolidasi dari Blibli. tiket.com merupakan salah satu agen travel terbesar di Indonesia. tiket.com terfokus layanan pemesanan tiket transportasi, penginapan, dan hiburan. Fitur yang beragam tersebut menjadikan tiket.com memiliki informasi yang lengkap. Salah satu fitur dalam tiket.com adalah *booking* berbagai jenis penginapan, seperti hotel, *homestay*, *resort*, *cottage*, dan sebagainya (PT. Global Tiket Network, 2023).

Menurut data *Top Brand Index*, tiket.com merupakan salah satu merek yang mendapatkan *Top Brand* pada bidang *online booking* tiket pesawat dan travel dengan posisi kedua pada tahun 2016, posisi ketiga pada tahun 2017, dan posisi keempat pada tahun 2018. *Top Brand Index* merupakan hasil riset terhadap konsumen yang dilakukan melalui survei dari *Frontier Consulting Group* di 11 kota besar di Indonesia dan menggunakan parameter yang diukur berdasarkan *top of mind share*, *top of market share* dan *top of commitment share* (Havidz et al., 2020).

2.3 Web Crawling

Web crawler yang juga disebut *web spider* adalah sebuah program yang menjelajahi web dengan cara yang metodis untuk mengumpulkan informasi. *Web crawler* digunakan untuk mengumpulkan data atau menyalin halaman *website* apa pun yang dikunjunginya. Namun yang paling penting, *web crawler* digunakan untuk mengumpulkan beberapa data spesifik dari sebuah *website*. (Shamrat et al., 2020).

2.3.1 URL (*Uniform Resource Locator*)

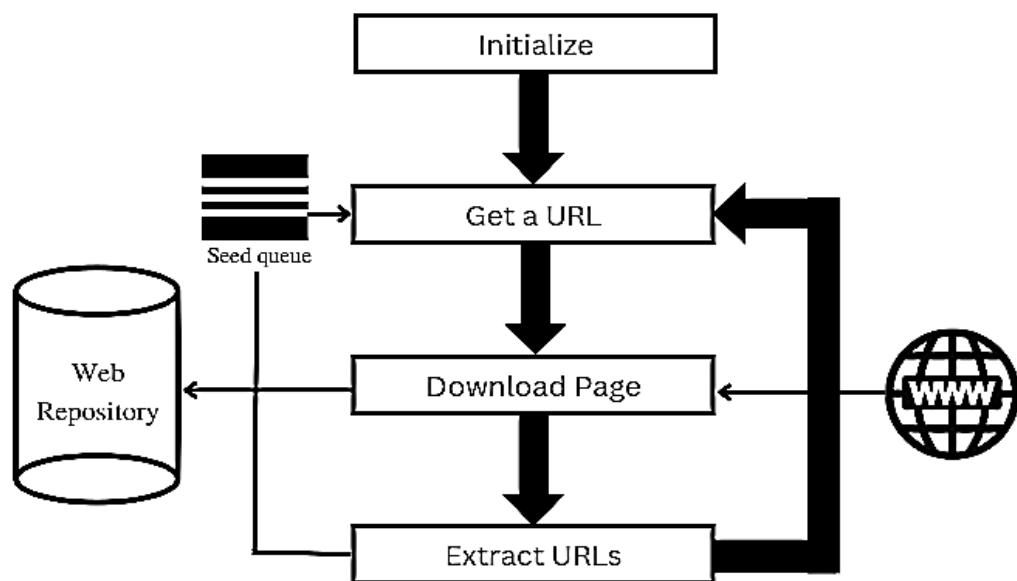
Uniform Resource Locator atau disingkat URL adalah sebuah alamat yang menyimpan sebuah data atau informasi yang berhubungan secara langsung dengan *website* tersebut. Secara umum manfaat atau fungsi dari sebuah URL itu untuk mengidentifikasi alamat data yang disimpan oleh *website*, baik berupa data gambar atau angka yang akan ditampilkan halaman *website* tersebut (Hidayatullah, 2022).

2.3.2 Seed URL

Seed URL atau URL awal dalam *web crawling* adalah URL dari mana sebuah *web crawler* akan mulai menjelajahi sebuah situs. Setelah *crawler* berada di *seed URL*, *crawler* akan mengekstrak data dari halaman tersebut dan mencari semua tautan ke halaman tambahan. Jika sebuah *crawler* diatur untuk menjelajahi seluruh domain, *crawler* secara sistematis akan mengikuti setiap tautan pada setiap halaman, mengekstrak data dari setiap halaman yang dihasilkan. Jalur dari *seed URL* sering kali dipengaruhi oleh file robots.txt dari sebuah *website*, yang menentukan bagaimana pemilik situs ingin bot menjelajahi situs tersebut (Cook, 2024).

2.3.3 Cara Kerja Web Crawling

Proses dari *web crawling* dimulai dengan *seed URL*. *Crawler* bergerak dari satu *link* ke *link* lainnya, mengekstrak *link* baru yang tersedia di halaman yang diunduh. Kemudian halaman yang diambil disimpan dalam *database* yang diindeks dengan baik sehingga nantinya dapat diambil kembali jika diperlukan. URL yang diekstrak dari halaman yang diunduh dikonfirmasi untuk mengetahui apakah dokumen terkaitnya telah diunduh atau belum. Jika belum diunduh, proses pengunduhan dokumen diulang sampai seluruh URL terunduh (Kausar et al., 2013).



Gambar 2.1 Alur Proses *Web Crawling* (Kausar et al., 2013)

Cara kerja *web crawler* dapat didefinisikan sebagai langkah-langkah berikut:

1. Memilih URL atau *seed* URL.
2. Mengambil halaman web yang sesuai dengan URL yang dipilih.
3. Mem-*parsing* halaman web yang sama untuk menemukan *link* URL baru.
4. Menambah URL baru ke *database*.

Ulangi proses mulai dari langkah kedua sampai tidak ada lagi URL yang tersisa.

2.4 *Web Scraping*

Web scraping adalah metode untuk mengekstraksi informasi dari situs web, sehingga menjadi data yang dapat dianalisis dan dimanfaatkan untuk berbagai tujuan (Djufri, M, 2020).

Langkah-langkah dasar yang digunakan untuk melakukan *web scraping* menggunakan bahasa pemrograman menurut (Almaqbal et al., 2019) antara lain:

1. Temukan dan periksa halaman web yang akan di-*scrape*.
2. Identifikasi data yang diperlukan yang ingin diekstrak.
3. Tulis kode yang sesuai menggunakan Bahasa *Python*.
4. Jalankan kode untuk mengekstrak data yang diperlukan dan disimpan dalam format yang diperlukan.

2.5 String Matching

Menurut (Marbun et al., 2019) *string matching* adalah proses pencarian semua kemunculan *query* yang selanjutnya disebut *pattern* ke dalam *string* yang lebih panjang atau teks. *String matching* dirumuskan menjadi $x = x[0..m - 1]$ dan $y = y[0..n - 1]$ dimana x adalah *pattern*, m adalah panjang *pattern*, y adalah teks, dan n adalah panjang teks.

String matching dibagi menjadi dua yaitu *exact matching* dan *heuristic matching*. *Exact matching* digunakan untuk menemukan pola yang berasal dari satu teks. Berdasarkan arah pencocokan *string*, algoritma *exact matching* diklasifikasikan menjadi tiga bagian, yaitu:

1. Pencocokan *string* dari arah kiri ke arah kanan, algoritma yang termasuk dalam kategori ini adalah *Brute Force* dan *Knuth Morris Pratt*.
2. Pencocokan *string* dari arah kanan ke arah kiri, algoritma yang termasuk dalam kategori ini adalah *Boyer Moore*, *Turbo Boyer-Moore*, *Tuned Boyer-Moore*, *Raita*, dan *Zhu-Takaoka*.
3. Arah pencarian atau pencocokan *string* yang ditentukan oleh program, algoritma yang termasuk dalam kategori ini adalah *Colussi* dan *Crochemore-Perrin*.

Heuristic atau *statistical matching* digunakan untuk menghubungkan dua data terpisah ketika *exact matching* tidak mampu mengatasi karena pembatasan pada saat yang tersedia. *Heuristic matching* dapat dilakukan dengan perhitungan *distance* antara pola dengan teks. *Exact* dan *heuristic matching* memiliki kelemahan dalam menemukan kata yang memiliki kemiripan makna tetapi berbeda tulisan.

2.6 Algoritma Raita

Algoritma *Raita* ialah bagian dari algoritma *exact string matching* yakni pencocokan *string* sesuai dengan susunan karakter yang terdapat dalam *string* dicocokkan dan memiliki jumlah maupun urutan karakter yang sama dalam *string*. *Raita* dalam merancang sebuah algoritma dengan membandingkan karakter terakhir dari pola karakter yang terdapat paling kanan *window*. Apabila mereka cocok, lalu karakter pertama dari pola teks kiri dari *window* juga dapat dibandingkan. Dan apabila mereka cocok, akan dibandingkan pada karakter tengah pola dengan karakter teks tengah *window*. Jika pada akhirnya benar-benar cocok, algoritma membandingkan dari

karakter lain, karakter kedua ke karakter kedua terakhir serta mungkin membandingkan kembali dengan karakter tengah (Ginting, 2019).

Algoritma *Raita*, menurut (Hudaa et al., 2020) terdiri atas dua fase yaitu:

2.6.1 Fase *Preprocessing*

Fase *preprocessing* pada algoritma *Raita* bertujuan untuk menghitung *bad-character* dari algoritma *Boyer Moore*. *Bad-character* adalah kumpulan karakter yang mewakili suatu pola (*pattern*). Fungsi pergeseran *bad-character* dari algoritma *Boyer Moore* disimpan dalam tabel *BmBc*.

Sebagai contoh dilakukan pencarian pola “HOMESTAY” pada teks “NACHELLE HOMESTAY”. Menurut (Marbun et al., 2019) proses perhitungannya adalah sebagai berikut:

Text : NACHELLE HOMESTAY

Pattern : HOMESTAY

Tabel 2.1 Indeks *Pattern*

<i>Index</i>	0	1	2	3	4	5	6	7
<i>Pattern</i>	H	O	M	E	S	T	A	Y

Tahap berikutnya dibuatlah tabel *BmBc* untuk melakukan perhitungan dengan rumus persamaan sebagai berikut:

$$m - 2 \dots \dots \dots (1)$$

$$m = \text{Pattern} = 8$$

$$m = \text{panjang pattern}$$

$$m = 8 - 2 = 6$$

$$m - 1 - i \dots \dots \dots (2)$$

$$BmBc(\text{pattern}) \ m - 1 - i$$

$$8 - 1 - 0 = 7 \text{ maka nilai diletakkan pada index ke-0 dengan pattern H}$$

$$8 - 1 - 1 = 6 \text{ maka nilai diletakkan pada index ke-1 dengan pattern O}$$

$$8 - 1 - 2 = 5 \text{ maka nilai diletakkan pada index ke-2 dengan pattern M}$$

$$8 - 1 - 3 = 4 \text{ maka nilai diletakkan pada index ke-3 dengan pattern E}$$

$$8 - 1 - 3 = 3 \text{ maka nilai diletakkan pada index ke-3 dengan pattern S}$$

$$8 - 1 - 3 = 2 \text{ maka nilai diletakkan pada index ke-3 dengan pattern T}$$

$$8 - 1 - 3 = 1 \text{ maka nilai diletakkan pada index ke-3 dengan pattern A}$$

Nilai Y adalah 8 sesuai dengan panjang pola, karena karakter yang tidak ada pada tabel diinisialisasikan dengan tanda (*) kemudian nilainya sesuai dengan panjang pola.

Setelah melakukan perhitungan di atas, maka diperoleh hasil *BmBc* yang dapat dilihat pada **Tabel 2.2**

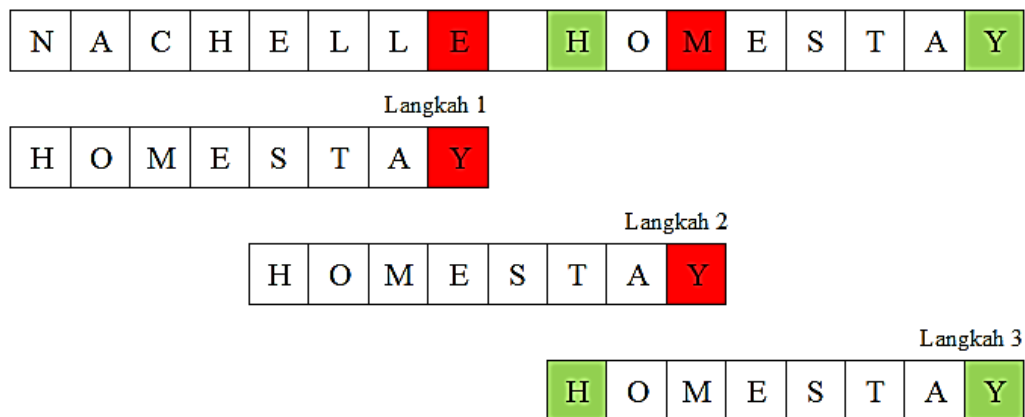
Tabel 2.2 Hasil *BmBc* (*Pattern*)

<i>Index</i>	0	1	2	3	4	5	6	
<i>Pattern</i>	H	O	M	E	S	T	A	*
<i>BmBc</i>	7	6	5	4	3	2	1	8

Setelah memperoleh hasil *BmBc* pada **Tabel 2.2** maka tahap selanjutnya adalah melakukan proses pencocokan karakter pola terhadap karakter teks.

2.6.2 Fase Pencarian

Proses pencocokan pola dengan teks dimulai dari sisi kiri teks. Proses pencarian dilakukan blok demi blok untuk mencari pola yang diinginkan. Proses akan berlanjut selama teks tersedia. Jika blok teks yang dicari telah ditemukan, pencarian akan berhenti.



Gambar 2.2 Langkah-Langkah Pencarian pada Teks

Dari **Gambar 2.2** dapat diketahui langkah-langkah pencarian pola pada teks sebagai berikut:

1. Langkah 1

Terjadi ketidakcocokan antara karakter “E” dan “Y”, maka dilakukan pergeseran karakter pola sebanyak nilai *BmBc* (*) yaitu 4 langkah.

2. Langkah 2

Terjadi ketidakcocokan antara karakter “M” dan “Y”, maka dilakukan pergeseran karakter pola sebanyak nilai $BmBc$ (*) yaitu 5 langkah.

3. Langkah 3

Terjadi kecocokan antara karakter “Y” pada teks dengan karakter “Y” pada pola sehingga selanjutnya dibandingkan dengan karakter awal pola yaitu “H” dengan karakter “H” pada teks dan terjadi kecocokan, sehingga dilakukan perbandingan karakter di tengah pola dan terjadi kecocokan. Maka proses pencocokan karakter telah selesai.

2.7 Penelitian yang Relevan

Berikut merupakan beberapa penelitian terdahulu yang relevan dengan penelitian yang akan dilakukan oleh penulis, antara lain:

1. Ikhwan Arjuna Lubis dengan judul “Penerapan *Web Crawling* Menggunakan Algoritma *Horspool* sebagai Filter pada Pengumpulan Informasi Restoran di Kota Medan” (2021). Pengujian sistem dilakukan sebanyak empat kali yaitu dengan menggunakan kata kunci “ayam penyet”, “nasi goreng”, dan “mie aceh” dengan tingkat akurasi data sebesar 100% dan pengujian untuk menentukan jumlah persentase restoran yang memiliki *review* dan tidak memiliki *review*.
2. Berdasarkan hasil penelitian yang dilakukan oleh Dedi Rudi Bawanto dan Nidia Rosmawanti dengan judul “Perbandingan Algoritma *Binary Search* dan *Raita* dalam Pencarian Data” (2017), yang membandingkan keakuratan kedua algoritma tersebut berdasarkan kecepatan dan ketepatan dalam pencarian, algoritma *Raita* lebih cepat 3% dibandingkan algoritma *Binary Search* dalam pencarian data dari keseluruhan data sampel yang diuji dengan total waktu 2,5116 ms.
3. Ryan Ridho Valba dengan judul “Perbandingan Algoritma *Horspool* dan Algoritma *Raita* pada Aplikasi Kamus Bahasa Indonesia-Jerman Berbasis Web” (2017). Dari hasil penelitian, algoritma *Raita* memiliki rata-rata hasil *running time* untuk pencarian kata yang lebih cepat yaitu 0,2336 ms dibandingkan algoritma *Horspool* dengan *running time* 0,2381 ms. Untuk hasil kompleksitas algoritma *Horspool* adalah $T(n) = \Theta(mn)$ sedangkan untuk algoritma *Raita* adalah $T(n) = \Theta(m)$, sehingga pencarian dengan menggunakan algoritma *Raita* lebih

cepat meskipun kedua algoritma ini tidak terlihat jauh berbeda saat diimplementasikan.

4. Berdasarkan penelitian yang dilakukan oleh (Hudaa et al., 2020) yang berjudul “*Data Search using Raita Algorithm*”. Dari hasil penelitian ini, algoritma *Raita* merupakan algoritma yang sangat baik untuk pencocokan *string*. Algoritma ini bekerja cepat dengan membandingkan blok demi blok di ruang *linkup* pencarian dan menghasilkan data yang akurat. Kecepatan proses pencocokan *pattern* dengan teks tergantung pada panjang *pattern* yang digunakan, semakin panjang *pattern* maka semakin cepat proses pencocokan dan sebaliknya.
5. Berdasarkan penelitian yang dilakukan oleh (Rahim et al., 2018) yang berjudul “*Searching Process with Raita Algorithm and its Application*”. Pada penelitian ini, dilakukan pencarian pada folder dengan nama “makalah” yang berisi 8.132 *files* dengan ukuran 5,32 GB. Proses pencarian dengan menggunakan algoritma *Raita* dengan *keyword* a dan jenis file PDF didapatkan total sebanyak 423 *files* dan membutuhkan waktu selama 3,650 *ms*. Implementasi algoritma *Raita* untuk pencarian file dilakukan sebanyak 5 kali dengan tipe yang sama dan *keyword* yang berbeda. Aplikasi pencarian file dengan menerapkan algoritma *Raita* mendapatkan hasil yang baik dengan waktu yang cukup cepat.
6. Berdasarkan penelitian yang dilakukan oleh Judith Hillen dengan judul “*Web Scraping for Food Price Research*” (2019). Pada penelitian ini, dapat disimpulkan bahwa *web scraping* adalah metode pengumpulan data berbiaya rendah yang menjanjikan untuk penelitian harga makanan. Data yang diperoleh melalui *web scraping* dapat membantu menganalisis bagaimana harga ditetapkan dalam ritel makanan *online* yang sedang berkembang.
7. Berdasarkan penelitian yang dilakukan oleh (Nandwani et al., 2021) yang berjudul “*Data Analysis by Web Scraping using Python*”. Berdasarkan hasil penelitian, hasil keseluruhan proyek sangat membantu untuk dipahami. *Web scraping* dapat mengekstrak data dengan lebih mudah dan hasil pengambilan data disimpan dalam format file *csv*.
8. Berdasarkan penelitian yang dilakukan oleh Shivani Jain dan A.L.N Rao yang berjudul “*A Comparative Performance Analysis of Approximate String Matching*” (2013). Pada penelitian ini, peneliti melakukan perbandingan antar beberapa jenis algoritma *string matching* yang di antaranya *Knuth Morris Pratt*, *Boyer Moore*,

Raita, *Horspool* dengan bahasa pemrograman PHP. Berdasarkan hasil penelitian tersebut, algoritma *Raita* terbukti jauh lebih cepat dalam prakteknya dibandingkan tiga algoritma lainnya.

9. Penelitian yang dilakukan oleh (Jeong et al., 2020) dengan judul “*Deep Learning-based Tourism Recommendation System using Social Network Analysis*” bertujuan untuk menghasilkan rekomendasi tempat wisata di Korea yang sesuai dengan pengguna dengan memanfaatkan kriteria rekomendasi yang jelas. Penelitian ini juga menerapkan *web crawling* untuk mengumpulkan 1.000 data dari *website* Naver dan blog dengan menggunakan kata kunci. Perbedaannya, data yang diambil hanya berupa nama tempat wisata dengan frekuensi tertinggi.
10. Penelitian yang dilakukan oleh (Shamrat et al., 2020) yang berjudul “*An Effective Implementation of Web Crawling Technology to Retrieve Data from the World Wide Web (WWW)*” juga menerapkan *web crawling* untuk mengekstrak data dari *website*, akan tetapi lebih berfokus pada mengusulkan sebuah *web crawler* yang menggunakan serangkaian *query* dari daftar kata kunci untuk melakukan *crawling* melalui halaman web daripada mengindeks semua halaman web di internet.
11. Pada penelitian yang dilakukan oleh (Hien et al., 2020) yang berjudul “*Web Crawler: Design and Implementation for Extracting Article-Like Contents*”, sebuah model *web crawler* dirancang dan dikembangkan yang berfungsi untuk mengekstrak dari 495 *website* menjadi konten yang mirip dengan artikel. Perbedaannya, model ini menggunakan model SVM (*Support Vector Machine*), *Grid Search*, dan *Cross Validation*. Dari ketiga model tersebut, *Grid Search* merupakan model yang paling optimal dengan akurasi sebesar 92,01%.
12. Penelitian yang dilakukan oleh (Ou et al., 2023) dengan judul “*Design and Implementation of a Recommendation System for Buying Fresh Foods Online Based on Web Crawling*” menggunakan *web crawling* untuk mengumpulkan informasi dari tiga *website* makanan segar, dimana halaman web tersebut berisi informasi produk secara detail. Perbedaannya, penelitian ini menggunakan *library* lanjutan dari tesaurus *Jieba* untuk membantu mengenali kata-kata yang digunakan dalam bahasa Tionghoa.
13. Penelitian yang dilakukan oleh (Prismana et al, 2020) yang berjudul “*The Design and Implementation of Web Crawler Distributed News Domain Detection System*” menggunakan sebuah *crawler* yang difokuskan pada distribusi untuk

mengumpulkan data berita secara efisien. Penelitian ini menggunakan *Get Metadata* untuk mendeteksi adanya judul berita yang sama. Jika ada, berita akan diperiksa kembali untuk menghasilkan data berita yang relevan.

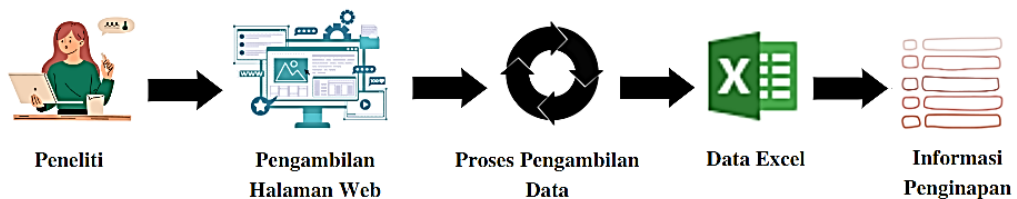
BAB III

ANALISIS DAN PERANCANGAN SISTEM

3.1 Arsitektur Umum

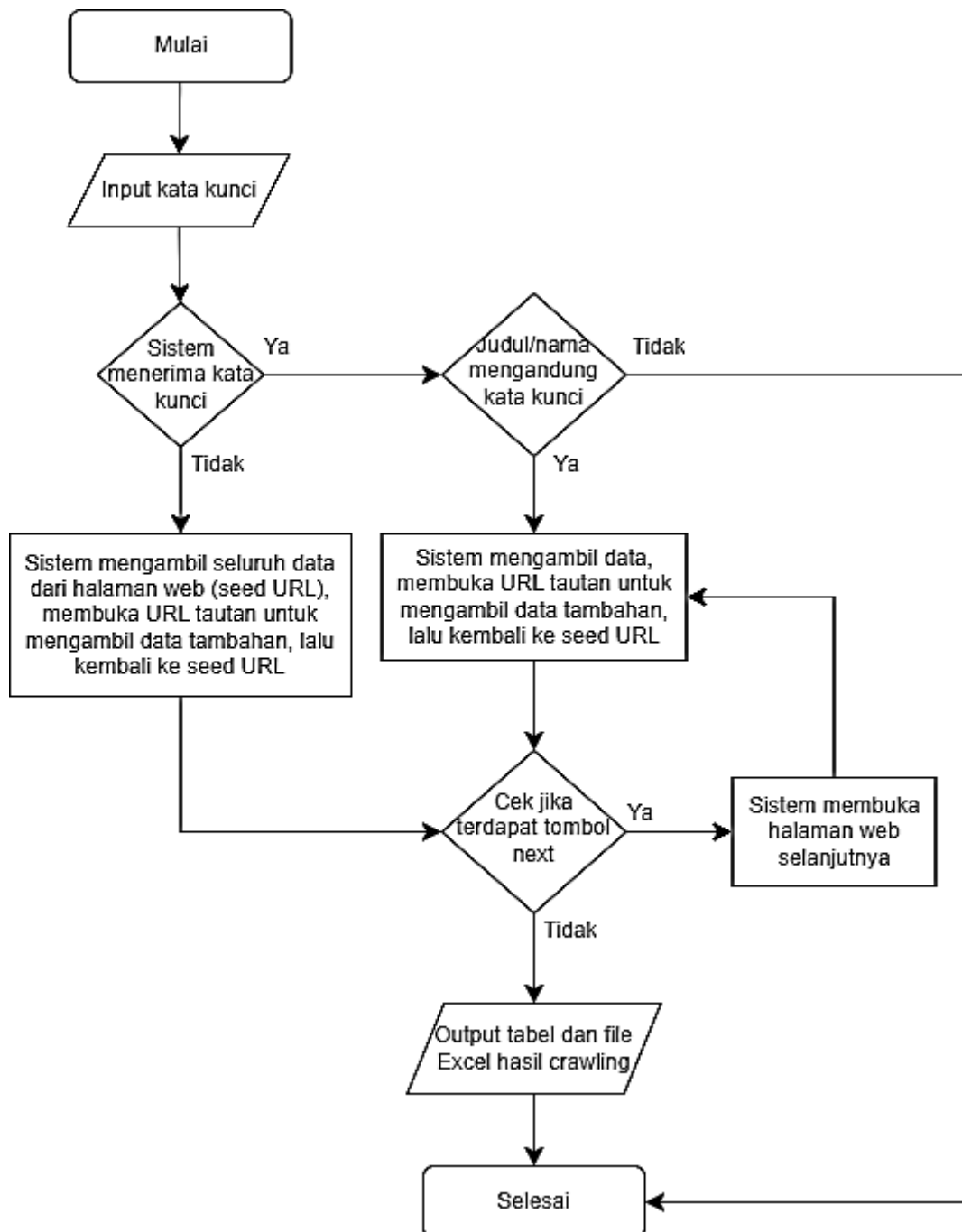
Dalam penelitian ini, informasi akan diekstraksi menggunakan *web crawler* dan algoritma *Raita* sebagai filter, sehingga hasil ekstraksi lebih terfokus pada kata kunci. *Web crawler* berfungsi untuk mengambil *hyperlink* dari halaman web serta mengunjungi *link-link* tersebut satu per satu. Sedangkan algoritma *Raita* berfungsi sebagai kata kunci dalam proses pengambilan informasi yang diinginkan. Adapun pengujian sistem ini yaitu dilakukan uji coba pengambilan informasi penginapan di Berastagi dari *website* tiket.com. *Seed URL* pada sistem ini adalah <https://www.tiket.com/hotel/indonesia/area/berastagi-108001534490301065> yaitu *URL website* tiket.com yang menampilkan penginapan di Berastagi. *Seed URL* tersebut berperan untuk membatasi sistem agar hanya mengambil informasi dari *website* tersebut. Setelah itu, sistem akan menerima *input* sebagai kata kunci, sehingga informasi yang diekstrak sesuai dengan kebutuhan. Jika sistem tidak menerima *input*, sistem secara otomatis akan melakukan proses *crawling* seluruh data yang ditampilkan. Karena beberapa informasi yang dibutuhkan terdapat pada halaman lain, maka sistem harus berpindah halaman *URL* untuk mendapatkan informasi tersebut. Oleh karena itu, sistem ini menggunakan *WebDriver* dan *library Selenium* untuk melakukan pindah halaman. Sistem hanya akan mengekstraksi informasi dari penginapan yang terkait dengan kata kunci yang telah dimasukkan ke dalam sistem sebelumnya.

Setelah informasi sudah selesai diambil, maka sistem akan menampilkan tabel dan menghasilkan sebuah dokumen *Excel* yang berisi hasil dari proses *crawling* sebelumnya. Adapun data penginapan yang diambil oleh sistem ini yaitu nama, kesan, *rating*, harga, dan alamat dari penginapan tersebut. Arsitektur umum pada penelitian ini dapat dilihat pada **Gambar 3.1**.



Gambar 3.1 Arsitektur Umum

Sistem akan dimulai dengan memasukkan kata kunci yang ingin dicari ke dalam sistem, setelah itu sistem akan mengunjungi halaman web penginapan satu per satu. Jika kata kunci sudah sesuai dengan nama penginapan, sistem akan mengambil data dari penginapan tersebut. Proses ini akan terus diulang sampai semua halaman yang ada pada web tersebut telah dikunjungi. *Flowchart* sistem dapat dilihat pada **Gambar 3.2**.



Gambar 3.2 *Flowchart* Sistem

Adapun *flowchart* sistem pada **Gambar 3.2** memiliki penjelasan sebagai berikut:

1. Input Kata Kunci dari Pengguna

Pengguna diminta untuk memasukkan kata kunci pada kolom pencarian, misalnya hotel, villa atau homestay. *WebDriver* akan membuka halaman web tiket.com yang berisi daftar penginapan di daerah Berastagi.

2. Logika untuk Menangani Kata Kunci Kosong

Jika pengguna tidak memasukkan kata kunci, maka semua data penginapan akan diekstraksi. Jika kata kunci dimasukkan, algoritma *Raita* akan digunakan untuk pencocokan *string*.

3. Pencocokan pada Kata Kunci

Memeriksa apakah nama penginapan mengandung kata kunci yang telah dimasukkan.

4. Pengumpulan Data dari *Seed* URL

Melakukan iterasi melalui elemen-elemen HTML untuk mendapatkan informasi tentang setiap penginapan. Informasi yang dicari meliputi nama, kesan, dan harga.

5. Pembukaan URL baru untuk Pengambilan Data Tambahan

Klik tautan hotel untuk membuka halaman detail hotel dalam tab baru. Pada tab baru, informasi tambahan yaitu *rating* dan alamat diambil dari elemen-elemen HTML yang sesuai, lalu sistem akan kembali ke halaman utama.

6. Cek Ketersediaan Halaman Selanjutnya

Cek apakah halaman selanjutnya tersedia, jika tidak tersedia atau proses *crawling* sudah selesai sistem akan berhenti.

7. Output Tabel dan File *Excel* Hasil *Crawling*

Sistem menampilkan tabel yang berisi informasi penginapan yang sudah diekstraksi antara lain nama, kesan, *rating*, harga, dan alamat penginapan, serta menyimpan informasi tersebut dalam format dokumen *Excel*.

3.2 Proses Algoritma *Raita*

Proses ini bertujuan untuk memperoleh informasi yang sesuai dengan kata kunci yang dimasukkan ke sistem. Pada penelitian ini algoritma *Raita* digunakan untuk proses *string matching*. *Source code* proses algoritma *Raita* dapat dilihat pada **Tabel 3.1**.

Tabel 3.1 *Source Code* Proses Algoritma Raita

```
def preBmBc(pat):
    # Fungsi untuk membuat tabel bad character
    lpat = len(pat)
    bmBc = [lpat] * 256
    for i in range(lpat - 1):
        bmBc[ord(pat[i].lower())] = lpat - i - 1
    return bmBc

def raita(pat, s):
    # Fungsi utama pencarian dengan algoritma Raita
    pat = pat.lower() # Konversi pola menjadi lowercase
    s = s.lower()     # Konversi teks menjadi lowercase

    lpat = len(pat)
    n = len(s)
    bmBc = preBmBc(pat) # Buat tabel bad character

    # Quick edge cases
    if lpat == 0 or lpat > n:
        return []

    if lpat == 1:
        return [i for i, c in enumerate(s) if c == pat[0]]

    # Inisialisasi hasil pencarian
    result = []
    firstCh, middleCh, lastCh = pat[0], pat[lpat // 2], pat[lpat - 1]
    j = 0

    # Proses pencarian
    while j <= n - lpat:
        # Periksa keseluruhan pola pada posisi j
        if s[j:j+lpat].lower() == pat:
            result.append(j)
            j += bmBc[ord(s[j + lpat - 1].lower())]
        else:
            j += 1

    return result
```

Adapun penjelasan pada *source code* proses algoritma Raita yaitu:

1. Fungsi lowercase() yaitu untuk mengubah seluruh pola dan teks menjadi huruf kecil.
2. Selanjutnya, fungsi dari lpat() yaitu untuk mendapatkan panjang pola dan panjang teks.
3. Pemisahan pola dan teks dilakukan dengan bmBc = preBmBc(pat) yakni, membuat tabel *bad character* menggunakan fungsi preBmBc.
4. Jika panjang pola = 0 atau lebih besar dari panjang teks n (input), maka fungsi mengembalikan list kosong [].
5. Jika panjang pola = 1, maka fungsi mengembalikan indeks semua kemunculan karakter tersebut dalam teks.

6. Hasil akan menyimpan indeks kemunculan pola, karakter yang dihasilkan yakni karakter pertama, tengah dan terakhir dari pola dengan fungsi `firstCh`, `middleCh`, `lastCh = pat[0]`, `pat[lpat // 2]`, `pat[lpat - 1]`.
7. `j = 0` berfungsi untuk inisialisasi variabel `j`, melacak posisi saat mencari.
8. `if s[j:j+lpat].lower() == pat` berfungsi untuk mengecek apakah *substring* pada posisi `j` hingga `j + lpat` dalam teks sama dengan pola. Jika ya, maka indeks `j` ditambahkan ke dalam list `result`.
9. `j += bmBc[ord(s[j + lpat - 1].lower())]` berfungsi untuk menggeser posisi pencarian sejauh nilai yang telah ditentukan dalam tabel *bad character*.
10. Setelah itu, didapatkan hasil pencocokan karakter teks dan pola.

3.3 Proses Web Crawling

Dalam penelitian ini, digunakan *Selenium* sebagai *library* untuk pengambilan informasi kumpulan *link* halaman yang berada pada *website* tiket.com. Proses *web crawling* menggunakan *Webdriver* bertujuan untuk membuka *website* dengan bantuan *library Selenium*, dan *Webdriver* yang digunakan pada *browser Google Chrome* adalah *Chromedriver*. Adapun kata kunci pada dimasukkan dalam penelitian ini meliputi “hotel”, “villa”, “homestay”, “cottage”, dan tanpa kata kunci dengan filter pada *website* tiket.com dengan lokasi di Berastagi. *Source code* proses *web crawling* dapat dilihat pada **Tabel 3.2**.

Tabel 3.2 *Source Code* Proses Web Crawling

```
# Input pola dari pengguna
print("Masukkan Kata Kunci")
pattern = input()

# Inisialisasi WebDriver
driver = webdriver.Chrome(options=options)

# Buka URL halaman web yang ingin Anda scraping
driver.get("https://www.tiket.com/hotel/indonesia/area/berastagi-108001534490301065")

# Inisialisasi list kosong untuk menyimpan data
data = []

# Fungsi untuk mengklik tombol "Selanjutnya"
def click_next_button():
    try:
        time.sleep(3) # Tunggu beberapa detik untuk memastikan halaman selesai dimuat
        next_button = driver.find_element(By.CSS_SELECTOR, 'a.btn-pages.active + a.btn-pages')
        next_button.click()
```

```

        return True
    except NoSuchElementException:
        return False

# Logika untuk menangani pola kosong
if not pattern:
    print("Pola kosong, scraping semua data...")
    pattern_check = lambda title: True
else:
    pattern_check = lambda title: raita(pattern, title)

while True:
    # Temukan semua elemen "seo-search-result"
    seo_search_results = driver.find_elements(By.CLASS_NAME, "seo-search-result")

    for seo_search_result in seo_search_results:
        # Temukan semua elemen "hotel-card" dalam "seo-search-result"
        hotel_card_elements =
seo_search_result.find_elements(By.CLASS_NAME, "hotel-card")

        for hotel_card_element in hotel_card_elements:
            # Temukan tautan <a> di dalam div
            link_element =
hotel_card_element.find_element(By.TAG_NAME, "a")

            title_element = hotel_card_element.find_element(By.CLASS_NAME,
'title.ellipsis')
            title = title_element.text

            # Periksa apakah title hotel cocok dengan pola
            if pattern_check(title):
                try:
                    tiket_impresi =
hotel_card_element.find_element(By.CSS_SELECTOR, 'div.tiket-
impression')

                    impresi = tiket_impresi.text
                except NoSuchElementException:
                    impresi = "Tidak tersedia"

                try:
                    harga_hotel =
hotel_card_element.find_element(By.CLASS_NAME, 'after-price')
                    harga = harga_hotel.text
                except NoSuchElementException:
                    harga = "Tidak tersedia"

                # Ambil URL tautan
                link_url = link_element.get_attribute("href")

                # Cetak URL tautan
                print("Link Hotel:", link_url)

                # Menggulir ke elemen agar terlihat

            driver.execute_script("arguments[0].scrollIntoView();", link_element)
            time.sleep(1) # Tunggu sebentar setelah menggulir

            # Klik tautan untuk membuka tab baru
            driver.execute_script("arguments[0].click();",
link_element)

```

```

        # Tunggu hingga tab baru terbuka (Anda bisa
menambahkan waktu tunggu sesuai kebutuhan)
        time.sleep(3) # Tunggu beberapa detik untuk
memastikan tab baru terbuka

        # Alihkan ke tab/jendela baru
        new_window = driver.window_handles[-1]
        driver.switch_to.window(new_window)

        # Di sini Anda dapat menambahkan kode untuk mengambil
data dari tab/jendela baru
        try:
            rating_element =
driver.find_element(By.CSS_SELECTOR,
'p.HcPVsG_text.HcPVsG_variant_highEmphasis')
            rating = rating_element.text
        except NoSuchElementException:
            rating = "Tidak tersedia"

        try:
            lokasi_element =
driver.find_element(By.CSS_SELECTOR,
'span.LocationSection_address_V8gdU')
            lokasi = lokasi_element.text
        except NoSuchElementException:
            lokasi = "Tidak tersedia"

        # Tambahkan data ke dalam list
        data.append([title, impresi, rating, harga, lokasi])

        # Menutup tab/jendela baru
        driver.close()

        # Kembali ke tab/jendela sebelumnya
        driver.switch_to.window(driver.window_handles[0])

    if not click_next_button():
        print("Halaman selanjutnya tidak tersedia atau sudah
selesai.")
        break # Keluar dari loop jika tombol "Selanjutnya" tidak
ditemukan

# Tutup WebDriver
driver.quit()

```

Adapun penjelasan pada *source code web crawling* yaitu:

1. Input Pola dari Pengguna

Pengguna diminta untuk memasukkan pola pencarian melalui input.

2. Inisialisasi *WebDriver*

WebDriver Chrome diinisialisasi menggunakan opsi yang mungkin telah didefinisikan sebelumnya.

3. Buka URL Halaman Web

WebDriver membuka halaman web tiket.com yang berisi daftar hotel di daerah Berastagi, Indonesia.

4. Inisialisasi List Kosong untuk Menyimpan Data
List 'data' dibuat untuk menyimpan hasil *scraping*.
5. Fungsi 'click_next_button()'
 - a. Fungsi ini digunakan untuk mengklik tombol "Selanjutnya" pada halaman web.
 - b. Tunggu beberapa detik untuk memastikan halaman selesai dimuat.
 - c. Cari elemen tombol "Selanjutnya" dengan menggunakan *selector* CSS.
 - d. Jika tombol ditemukan, klik tombol tersebut dan kembalikan 'True', jika tidak, kembalikan 'False'.
6. Logika untuk Menangani Pola Kosong:
 - a. Jika pengguna tidak memasukkan pola, maka semua data hotel akan di-*scrape*.
 - b. Jika pola dimasukkan, fungsi 'raita' akan digunakan untuk memeriksa apakah judul hotel cocok dengan pola.
7. *Loop* Utama untuk Pencarian Data
Loop utama ini terus berjalan selama ada halaman selanjutnya.
8. *Loop* Pengumpulan Data
 - a. Melakukan iterasi melalui elemen-elemen HTML untuk mendapatkan informasi tentang setiap hotel.
 - b. Informasi yang dicari meliputi judul, impresi, dan harga.
9. Pencocokan Pada Pola
Memeriksa apakah judul hotel cocok dengan pola yang diberikan oleh pengguna.
10. Pembukaan Tab Baru untuk Detail Hotel
 - a. Klik tautan hotel untuk membuka halaman detail hotel dalam tab baru.
 - b. Tunggu beberapa detik untuk memastikan tab baru terbuka.
11. Pengambilan Data Tambahan dari Tab Baru
Pada tab baru, ambil informasi tambahan seperti *rating* dan lokasi dari elemen-elemen HTML yang sesuai.
12. Penambahan Data ke Dalam List 'data'
Setelah mengumpulkan semua informasi yang dibutuhkan, tambahkan data ke dalam list 'data'.

13. Penutupan Tab/Jendela Baru dan Kembali ke Tab/Jendela Sebelumnya
Setelah mengumpulkan data dari halaman detail hotel, tutup tab/jendela baru dan kembali ke tab/jendela utama.
14. *Loop* untuk Halaman Selanjutnya
 - a. Cek apakah halaman selanjutnya tersedia dengan memanggil fungsi `'click_next_button()'`.
 - b. Jika halaman selanjutnya tidak tersedia atau *scraping* sudah selesai, keluar dari *loop* utama.
15. Penutupan *WebDriver*
Setelah selesai *scraping*, tutup *WebDriver*.

3.4 Proses Testing

Pada penelitian ini dilakukan pengujian pada *website* tiket.com yaitu pengambilan informasi penginapan di Berastagi dengan menggunakan *web crawler* dan algoritma yang digunakan yaitu *Raita* untuk melihat apakah sistem yang dibangun berfungsi dengan baik. Proses pada sistem dimulai dengan mengambil *seed* URL yaitu <https://www.tiket.com/hotel/indonesia/area/berastagi-108001534490301065>. Sebelum data di-*crawl* oleh sistem, pengguna akan diminta memasukkan sebuah *input* atau kata kunci ke sistem yang akan melakukan *filtering* dengan algoritma *Raita* dengan pencocokan *string* untuk menemukan nama penginapan yang hanya mengandung kata kunci saja. Kemudian sistem akan menampilkan nama, kesan, *rating*, harga, dan alamat dari penginapan yang telah diekstraksi tersebut. Selanjutnya informasi yang telah diperoleh disimpan dalam format dokumen *Excel*.

3.5 Perancangan Sistem

Dalam penelitian ini, pengembangan sistem akan dilakukan menggunakan *Jupyter Notebook* dengan bahasa pemrograman *Python Library* yang akan digunakan adalah *Selenium*, dengan hasil akhir yang disimpan dalam format dokumen *Excel*.

BAB IV

IMPLEMENTASI DAN PENGUJIAN SISTEM

Setelah melakukan analisis dan perancangan sistem, langkah selanjutnya adalah implementasi. Pada bagian ini, akan dibahas mengenai hasil analisis dari proses *web crawling* dengan algoritma *Raita* sebagai filter untuk mengambil informasi penginapan di Berastagi dari *website* tiket.com.

4.1 Implementasi Sistem

Dalam penelitian ini, pengembangan sistem dilakukan dengan menggunakan IDE *Jupyter Notebook*, menggunakan bahasa pemrograman *Python*, dan menghasilkan dokumen *Excel* sebagai *output*.

4.1.1. Spesifikasi Perangkat Keras

Perangkat keras yang digunakan dalam penelitian ini adalah komputer dengan spesifikasi sebagai berikut:

1. Processor Intel Celeron CPU N3050 @ 1.60Hz 1.60 GHz
2. Intel HD Graphics
3. Memory RAM 4 GB
4. Kapasitas Harddisk 500 GB

4.1.2. Spesifikasi Perangkat Lunak

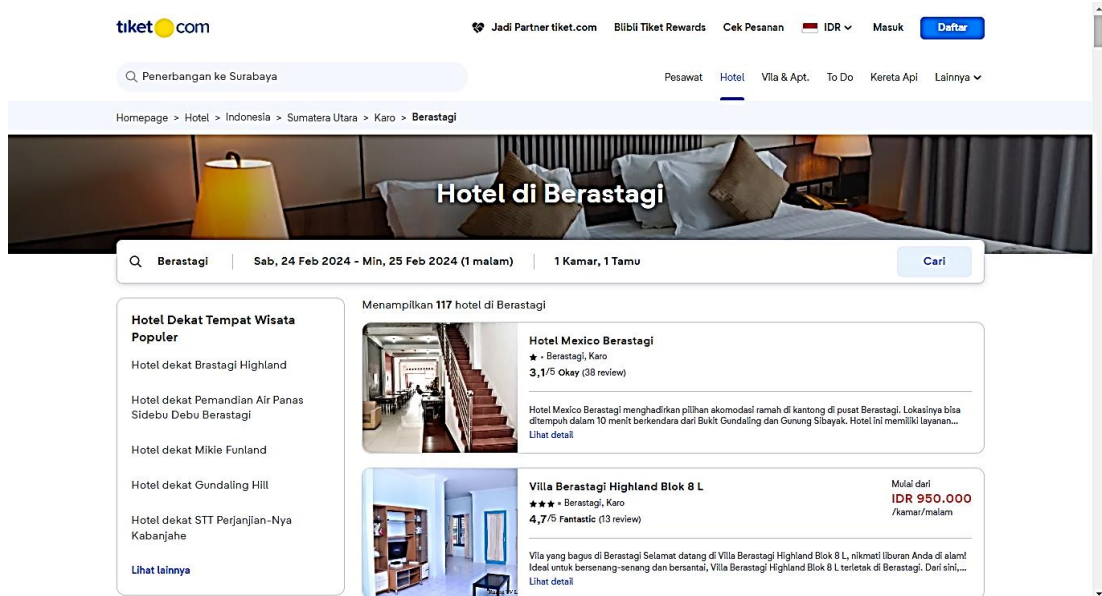
Adapun beberapa perangkat lunak yang digunakan dalam penelitian ini adalah sebagai berikut:

1. Sistem Operasi Windows 10 Home Single Language
2. Google Chrome & ChromeDriver
3. Jupyter Notebook
4. Python 3.12
5. Microsoft Excel 2010
6. Library: selenium, openpyxl, dan pandas.

4.2 Implementasi Website tiket.com

Penelitian ini menggunakan *website* tiket.com sebagai sumber informasi penginapan. Filter yang digunakan yaitu berdasarkan lokasi di “Berastagi” Provinsi Sumatera Utara. Dengan sistem tersebut didapatkan hasil sejumlah 106 informasi penginapan

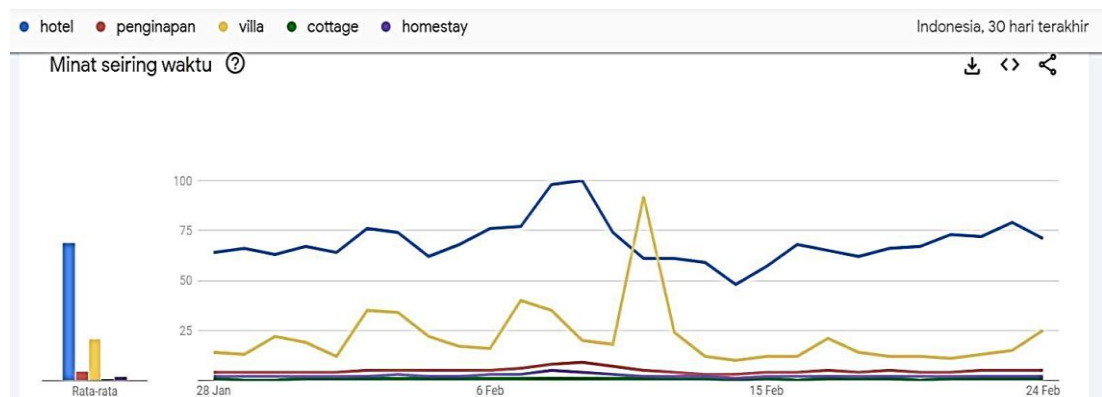
Berastagi yang tersedia pada *website* tiket.com yang diekstraksi pada November hingga Desember 2023.



Gambar 4.1 Halaman tiket.com

4.3 Implementasi Proses Filter Algoritma *Raita*

Sistem yang dikembangkan dalam penelitian ini menggunakan algoritma *Raita* sebagai filter. Implementasi algoritma dimulai dengan pencarian tabel *bad character* dan pengambilan informasi menggunakan *Webdriver Selenium*. Pada penelitian ini dilakukan pengujian sebanyak 5 kali dengan kata kunci yang berbeda yaitu “hotel”, “villa”, “homestay”, “cottage”, dan tanpa kata kunci. Pengambilan kata kunci diambil berdasarkan informasi kata kunci yang paling banyak digunakan untuk kata kunci yang relevan dari *Google Trends*.



Gambar 4.2 Google Trends

Adapun uji coba pencocokan *string* dengan menggunakan algoritma *Raita* untuk menemukan pola “hotel” pada teks “Grand Mutiara Hotel Berastagi” dapat dilihat pada **Gambar 4.3** dan **Gambar 4.4** di mana pola “hotel” ditemukan pada indeks ke-14 pada teks.

```

✓ 15s ▶ def preBmBc(pat):
    # Fungsi untuk membuat tabel bad character
    lpat = len(pat)
    bmBc = [lpat] * 256
    for i in range(lpat - 1):
        bmBc[ord(pat[i].lower())] = lpat - i - 1
    return bmBc

def raita_match(pat, s):
    # Fungsi utama pencarian dengan algoritma Raita
    pat = pat.lower() # Konversi pola menjadi lowercase
    s = s.lower()     # Konversi teks menjadi lowercase

    lpat = len(pat)
    n = len(s)
    bmBc = preBmBc(pat) # Buat tabel bad character

    # Quick edge cases
    if lpat == 0 or lpat > n:
        return []

    if lpat == 1:
        return [i for i, c in enumerate(s) if c == pat[0]]

    # Inisialisasi hasil pencarian
    result = []
    firstCh, middleCh, lastCh = pat[0], pat[lpat // 2], pat[lpat - 1]
    j = 0

```

Gambar 4.3 Pencocokan *String* Algoritma *Raita* (1)

```

✓ 15s ▶ # Proses pencarian
while j <= n - lpat:
    if lastCh == s[j + lpat - 1] and middleCh == s[j + lpat // 2] and firstCh == s[j]:
        # Jika karakter terakhir, tengah, dan pertama cocok, tambahkan ke hasil
        result.append(j)
        j += bmBc[ord(s[j + lpat - 1])]

    return result

# Input dari pengguna
pattern = input("Masukkan pola: ")
text = input("Masukkan teks: ")

# Pencocokan pola
matches = raita_match(pattern, text)
print(f"Pencocokan pola '{pattern}' pada teks '{text}' pada posisi: {matches}")

```

Masukkan pola: hotel
 Masukkan teks: Grand Mutiara Hotel Berastagi
 Pencocokan pola 'hotel' pada teks 'Grand Mutiara Hotel Berastagi' pada posisi: [14]

Gambar 4.4 Pencocokan *String* Algoritma *Raita* (2)

Adapun langkah-langkah dari proses pencocokan *string* algoritma *Raita* adalah sebagai berikut:

1. Fase *Preprocessing*

Pada fase *preprocessing*, dilakukan perhitungan untuk mendapatkan tabel *BmBc*.

Text : GRAND MUTIARA HOTEL BERASTAGI

Pattern : HOTEL

Tabel 4.1 Indeks *Pattern*

<i>Index</i>	0	1	2	3	4
<i>Pattern</i>	H	O	T	E	L

Tahap berikutnya dibuatlah tabel *BmBc* untuk melakukan perhitungan dengan rumus persamaan sebagai berikut:

$$m - 2 \dots \dots \dots (1)$$

$$m = \text{Pattern} = 5$$

$$m = \text{panjang pattern}$$

$$m = 5 - 2 = 3$$

$$m - 1 - i \dots \dots \dots (2)$$

$$BmBc(\text{pattern}) \ m - 1 - i$$

$$5 - 1 - 0 = 4 \text{ maka nilai diletakkan pada index ke-0 dengan pattern H}$$

$$5 - 1 - 1 = 3 \text{ maka nilai diletakkan pada index ke-1 dengan pattern O}$$

$$5 - 1 - 2 = 2 \text{ maka nilai diletakkan pada index ke-2 dengan pattern T}$$

$$5 - 1 - 3 = 1 \text{ maka nilai diletakkan pada index ke-3 dengan pattern E}$$

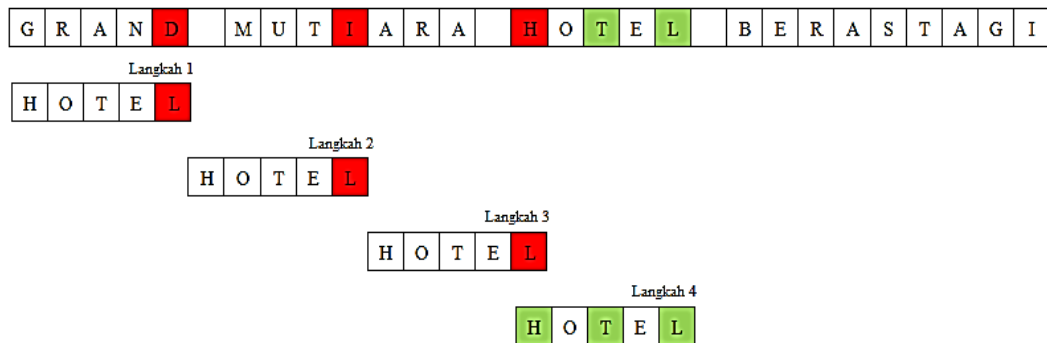
Nilai L adalah 5 sesuai dengan panjang pola, karena karakter yang tidak ada pada tabel diinisialisasikan dengan tanda (*) kemudian nilainya sesuai dengan panjang pola. Maka, diperoleh hasil *BmBc* yang dapat dilihat pada **Tabel 4.2**

Tabel 4.2 Hasil *BmBc*

<i>Index</i>	0	1	2	3	
<i>Pattern</i>	H	O	T	E	*
<i>BmBc</i>	4	3	2	1	5

2. Fase Pencarian

Pada fase pencarian, dilakukan proses pencocokan pola dengan teks dari sisi kiri teks seperti pada **Gambar 4.5**.



Gambar 4.5 Langkah-Langkah Pencarian pada Teks

Langkah-langkah pencarian pola pada teks adalah sebagai berikut:

1. Langkah 1

Terjadi ketidakcocokan antara karakter "D" dan "L", maka dilakukan pergeseran karakter pola sebanyak nilai $BmBc (*)$ yaitu 5 langkah.

2. Langkah 2

Terjadi ketidakcocokan antara karakter "I" dan "L", maka dilakukan pergeseran karakter pola sebanyak nilai $BmBc (*)$ yaitu 5 langkah.

3. Langkah 3

Terjadi ketidakcocokan antara karakter "H" dan "L", maka dilakukan pergeseran karakter pola sebanyak nilai $BmBc (*)$ yaitu 4 langkah.

4. Langkah 4

Terjadi kecocokan antara karakter "L" pada teks dengan karakter "L" pada pola sehingga selanjutnya dibandingkan dengan karakter awal pola yaitu "H" dengan karakter "H" pada teks dan terjadi kecocokan, sehingga dilakukan perbandingan karakter di tengah pola yaitu karakter "T" dan terjadi kecocokan. Maka proses pencocokan karakter telah selesai.

4.4 Pengujian Sistem

Pengujian dilakukan untuk mengetahui apakah sistem yang telah dibangun dapat bekerja dengan baik dalam mengekstrak informasi yang terdapat pada *website* tiket.com dan informasi yang diekstraksi sesuai dengan kata kunci yang diberikan pada sistem. Pengujian dilakukan terhadap 3 *page* yang terdiri dari 106 informasi

penginapan di Berastagi. Pengujian dilakukan dengan mengambil informasi pada November hingga Desember 2023.

4.4.1. Pengujian Pertama

Pengujian pertama ini dilakukan pencarian untuk menemukan penginapan di Berastagi dengan kata kunci “hotel”.

```
Masukkan Kata Kunci
hotel
Link Hotel: https://www.tiket.com/hotel/indonesia/grand-mutiara-hotel-berastagi-108001534490300668?checkin=2023-11-30&checkout=2023-12-01&room=1&adult=1&soldOut=
Link Hotel: https://www.tiket.com/hotel/indonesia/uluna-hotel-412001639108211657?checkin=2023-11-30&checkout=2023-12-01&room=1&adult=1&soldOut=
Link Hotel: https://www.tiket.com/hotel/indonesia/hotel-miranda-at-des-wisata-berastagi-412001640080854093?checkin=2023-11-30&checkout=2023-12-01&room=1&adult=1&soldOut=
Link Hotel: https://www.tiket.com/hotel/indonesia/hotel-mexico-berastagi-108001534519898710?checkin=2023-11-30&checkout=2023-12-01&room=1&adult=1&soldOut=
Link Hotel: https://www.tiket.com/hotel/indonesia/pariban-hotel-606001687316453689?checkin=2023-11-30&checkout=2023-12-01&room=1&adult=1&soldOut=
Link Hotel: https://www.tiket.com/hotel/indonesia/hotel-kalang-ulu-berastagi-412001639994113896?checkin=2023-11-30&checkout=2023-12-01&room=1&adult=1&soldOut=
Link Hotel: https://www.tiket.com/hotel/indonesia/hotel-sibayak-internasional-berastagi-206001560503799303?checkin=2023-11-30&checkout=2023-12-01&room=1&adult=1&soldOut=
Link Hotel: https://www.tiket.com/hotel/indonesia/rudang-berastagi-hotel-108001534490318017?checkin=2023-11-30&checkout=2023-12-01&room=1&adult=1&soldOut=
Link Hotel: https://www.tiket.com/hotel/indonesia/3-berastagi-hotel-405001622169017034?checkin=2023-11-30&checkout=2023-12-01&room=1&adult=1&soldOut=
Link Hotel: https://www.tiket.com/hotel/indonesia/green-garden-hotel-berastagi-108001534490387141?checkin=2023-11-30&checkout=2023-12-01&room=1&adult=1&soldOut=
Link Hotel: https://www.tiket.com/hotel/indonesia/hotel-new-dieng-by-helocus-607001689131427338?checkin=2023-11-30&checkout=2023-12-01&room=1&adult=1&soldOut=
Link Hotel: https://www.tiket.com/hotel/indonesia/as-cafe-resto-hotel-507001656611493244?checkin=2023-11-30&checkout=2023-12-01&room=1&adult=1&soldOut=
Link Hotel: https://www.tiket.com/hotel/indonesia/suite-pakar-hotel-209001567494906233?checkin=2023-11-30&checkout=2023-12-01&room=1&adult=1&soldOut=
Link Hotel: https://www.tiket.com/hotel/indonesia/oyo-2322-dien-karona-hotel-212001576052105936?checkin=2023-11-30&checkout=2023-12-01&room=1&adult=1&soldOut=
Link Hotel: https://www.tiket.com/hotel/indonesia/hotel-grand-orri-berastagi-509001662427184106?checkin=2023-11-30&checkout=2023-12-01&room=1&adult=1&soldOut=
Halaman selanjutnya tidak tersedia atau sudah selesai.
```

Gambar 4.6 Proses Pengujian Pertama

Hasil pengujian pertama pada **Gambar 4.7** menggunakan kata kunci “hotel” sebagai parameter pencarian dengan *seed* URL yang digunakan yaitu “https://www.tiket.com/hotel/indonesia/area/berastagi-108001534490301065”. Hasil yang diperoleh menemukan sebanyak 15 hotel setelah proses *crawling* dan pencocokan *string* nama.

Tabel 4.3 Hasil Pengujian Pertama

No	Nama	Kesan	Rating	Harga	Alamat
1	Grand Mutiara Hotel Berastagi	Good	4,38/5	IDR 742.000	Jl. Peceren No.168, Berastagi, Kab. Karo, North Sumatra, Indonesia
2	Uluna Hotel	Good	4,23/5	Tidak tersedia	Jl. Jamin Ginting No 10
3	Hotel Miranda at Desa Wisata Berastagi	Good	4,08/5	IDR 104.979	Jalan Pendidikan, No.no 9, Jaranguda, Kecamatan Berastagi
4	Hotel Mexico Berastagi	Okay	3,08/5	IDR 194.775	Jl. Veteran No.18, Tambak Lau Mulgap I, Berastagi, Kabupaten Karo, Sumatera Utara, Indonesia
5	Pariban Hotel	Good	4,23/5	IDR 382.725	Jaranguda, Kec. Merdeka, Kabupaten Karo, Sumatera Utara 20354
6	Hotel Kalang Ulu Berastagi	Good	4,12/5	IDR 439.900	Jl. Mumpin Tua No.99 (Arah Vihara Dharma Shanti), Desa Sempa Jaya, Berastagi - Karo
7	Hotel Sibayak International Berastagi	Good	4,01/5	IDR 699.602	Jl. Merdeka, Berastagi - Sumatera Utara
8	Rudang Hotel & Resort Berastagi	Good	3,80/5	IDR 373.915	Jl. Jamin Ginting, No. 16, Berastagi
9	3 Berastagi Hotel	Okay	3,50/5	IDR 193.556	Jl. Jamin Ginting No.134, Sempajaya, Kec. Berastagi, Kabupaten Karo, Sumatera Utara
10	Green Garden Hotel Berastagi	Good	3,73/5	IDR 505.885	Jl. Peceran No.30, Berastagi, Tanah Karo, Sumatera Utara
11	Hotel New Dieng By Helocus	Good	3,74/5	IDR 237.500	Jl. Udara No.8, Gundaling I, Kec. Berastagi,

					Kabupaten Karo, Sumatera Utara, 22153 Berastagi, Indonesia
12	As Cafe Resto Hotel	Okay	3,0/5	IDR 12.825.0 00	Desa Jaranguda Simpang Tiga, gunung sibayak, Kec. Berastagi, Sumatera Utara
13	Suite Pakar Hotel	Bad	2,3/5	IDR 8.400.00 0	Jalan Letjen Jamin 2 Ginting, Desa Merdeka, Berastagi, Sumatera Utara, ID
14	OYO 2322 Dien Karona Hotel	Tidak tersedia	Akomo dasi ini lagi penuh	Tidak tersedia	148, Jl. Jaranguda No.148, Merdeka, Sumatera Utara, Kabupaten Karo, Berastagi, ID
15	Hotel Grand Orri Berastagi	Tidak tersedia	Akomo dasi ini lagi penuh	Tidak tersedia	6G36+6PF, Jalan Orri (from, Jl. Jamin Ginting Jl. Lau Gumba, Sempajaya, Berastagi, Karo Regency

Hasil pengujian pertama pada **Tabel 4.3**, *web crawling* telah berhasil dilakukan. Dengan 15 data hotel yang berhasil di-*crawling* dengan kata kunci “hotel”. Adapun informasi yang diambil yakni nama, kesan, *rating*, harga, dan alamat. Terdapat 3 dari 15 data hotel yang tidak memiliki keterangan yang lengkap dikarenakan hotel tersebut sedang penuh pada saat proses *crawling* sehingga menampilkan kalimat “Tidak Tersedia” dan “Akomodasi ini lagi penuh”. Keterangan yang tidak lengkap tersebut meliputi kesan, *rating* dan harga. Didapati hasil hotel terbaik berdasarkan *rating* dari 15 hotel yang berhasil di-*crawling* yakni Grand Mutiara Hotel Berastagi dengan *rating* 4,38.

4.4.2. Pengujian Kedua

Pengujian kedua ini dilakukan pencarian untuk menemukan penginapan di Berastagi dengan kata kunci “*villa*”.

Masukkan Kata Kunci

villa

Link Hotel: <https://www.tiket.com/homes/indonesia/villa-berastagi-highland-8kk-201001547180292829?checkin=2023-11-30&checkout=2023-12-01&room=1&adult=1&soldOut=>

Link Hotel: <https://www.tiket.com/homes/indonesia/villa-berastagi-highland-blok-8-1-402001614221306696?checkin=2023-11-30&checkout=2023-12-01&room=1&adult=1&soldOut=>

Link Hotel: <https://www.tiket.com/homes/indonesia/villa-bukit-indah-4br-brastagi-berastagi-401001611307813658?checkin=2023-11-30&checkout=2023-12-01&room=1&adult=1&soldOut=>

Link Hotel: <https://www.tiket.com/homes/indonesia/villa-berastagi-highland-112001544792469084?checkin=2023-11-30&checkout=2023-12-01&room=1&adult=1&soldOut=>

Link Hotel: <https://www.tiket.com/homes/indonesia/villa-berastagi-highland-blok-9kk-408001628849724828?checkin=2023-11-30&checkout=2023-12-01&room=1&adult=1&soldOut=>

Link Hotel: <https://www.tiket.com/homes/indonesia/galaxy-villa-manage-by-pals-berastagi-505001652966123683?checkin=2023-11-30&checkout=2023-12-01&room=1&adult=1&soldOut=>

Link Hotel: <https://www.tiket.com/homes/indonesia/villa-mutiara-asri-by-dinda-no-8bb-504001650840445565?checkin=2023-11-30&checkout=2023-12-01&room=1&adult=1&soldOut=>

Link Hotel: <https://www.tiket.com/homes/indonesia/villa-manchester-blok-1-h-506001654161717002?checkin=2023-11-30&checkout=2023-12-01&room=1&adult=1&soldOut=>

Link Hotel: <https://www.tiket.com/homes/indonesia/villa-mutiara-asri-by-dinda-no-8d-504001650348926224?checkin=2023-11-30&checkout=2023-12-01&room=1&adult=1&soldOut=>

Link Hotel: <https://www.tiket.com/hotel/indonesia/villa-keluarga-bre-batunanggar-307001595622165074?checkin=2023-11-30&checkout=2023-12-01&room=1&adult=1&soldOut=>

Link Hotel: <https://www.tiket.com/homes/indonesia/villa-mutiara-asri-by-dinda-no-18a-504001650842178896?checkin=2023-11-30&checkout=2023-12-01&room=1&adult=1&soldOut=>

Link Hotel: <https://www.tiket.com/homes/indonesia/villa-lutze-gundaling-berastagi-401001611641947684?checkin=2023-11-30&checkout=2023-12-01&room=1&adult=1&soldOut=>

Link Hotel: <https://www.tiket.com/homes/indonesia/villa-mutiara-asri-by-dinda-no-18g-504001651053651138?checkin=2023-11-30&checkout=2023-12-01&room=1&adult=1&soldOut=>

Link Hotel: <https://www.tiket.com/homes/indonesia/villa-berastagi-highland-7-11-412001639390432776?checkin=2023-11-30&checkout=2023-12-01&room=1&adult=1&soldOut=>

Link Hotel: <https://www.tiket.com/hotel/indonesia/caesarea-villa-604001681113650192?checkin=2023-11-30&checkout=2023-12-01&room=1&adult=1&soldOut=>

Link Hotel: <https://www.tiket.com/homes/indonesia/villa-berastagi-resort-v86-605001684275624211?checkin=2023-11-30&checkout=2023-12-01&room=1&adult=1&soldOut=>

Link Hotel: <https://www.tiket.com/homes/indonesia/villa-highland-berastagi-no-9a-awe-villa-506001655221976847?checkin=2023-11-30&checkout=2023-12-01&room=1&adult=1&soldOut=>

Link Hotel: <https://www.tiket.com/homes/indonesia/villa-bukit-mas-best-seller-b3-507001657748138985?checkin=2023-11-30&checkout=2023-12-01&room=1&adult=1&soldOut=>

Link Hotel: <https://www.tiket.com/hotel/indonesia/sunroom-villa-bukit-mas-605001685370469139?checkin=2023-11-30&checkout=2023-12-01&room=1&adult=1&soldOut=>

Gambar 4.7 Proses Pengujian Kedua

Hasil pengujian kedua pada **Gambar 4.8** menggunakan kata kunci “*villa*” sebagai parameter pencarian dengan *seed* URL yang digunakan yaitu “<https://www.tiket.com/hotel/indonesia/area/berastagi-108001534490301065>”. Hasil yang diperoleh menemukan sebanyak 40 *villa* setelah proses *crawling* dan pencocokan *string* nama.

Tabel 4.4 Hasil Pengujian Kedua

No	Nama	Kesan	Rating	Harga	Alamat
1	Villa Berastagi Highland 8KK	Good	4,1/5	IDR 849.915	Jl. Lau Gumba, Sempajaya, Berastagi, Kabupaten Karo, Sumatera Utara, Indonesia
2	Villa Berastagi Highland Blok 8 L	Fantastic	4,7/5	IDR 940.500	Jalan Lau Gumba Blok 8L, Sempajaya, Kab. Karo. Berastagi, Berastagi, Karo

3	Villa Bukit Indah 4BR Brastagi / Berastagi	Good	3,7/5	IDR 841.500	Jl. Mimpin Tua, Sempajaya, Kec. Berastagi, Kabupaten Karo, Sumatera Utara 22152, Indonesia
4	Villa Berastagi Highland	Good	4,5/5	IDR 875.000	jalan lau gumba no 9R
5	Villa Berastagi Highland Blok 9KK	Good	4,3/5	IDR 891.000	Jl. Lau Gumba Blok 9KK , Berastagi, Karo, Sumatera Utara
6	Galaxy Villa manage by Pal's Berastagi	Good	3,8/5	Tidak tersedia	Jl. Kenangan, Gundaling I, Kec. Berastagi, Kabupaten Karo, Sumatera Utara 22152
7	Villa Mutiara Asri by Dinda No 8BB	Fantastic	5,0/5	IDR 914.544	Jl Jamin Ginting, Sempajaya, Berastagi, Karo Regency, North Sumatra 22152
8	Villa Manchester Blok 1 H	Fantastic	5,0/5	IDR 843.374	Jl. Mimpin Tua No.63, Sempajaya, Kec. Berastagi, Kabupaten Karo, Sumatera Utara 22152, Indonesia
9	Villa Mutiara Asri by Dinda No 8D	Good	3,7/5	IDR 914.544	Jl. Jamin Ginting , Sempajaya
10	Villa Keluarga Bre Batunanggar	Good	4,4/5	IDR 304.586	Jl. Kolam Renang, Gundaling I, Berastagi, Kabupaten Karo, Sumatera Utara
11	Villa Mutiara Asri by Dinda No 18A	Okay	3,4/5	IDR 923.781	Jl Jamin Ginting , Sempajaya, Berastagi, Karo Regency, North Sumatra 22152
12	Villa Lutze Gundaling Berastagi	Tidak tersedia	Akomo dasi ini lagi penuh	Tidak tersedia	Jalan Gundaling No. 184 , Berastagi , Sumatera Utara , Indonesia

13	Villa Mutiara Asri by Dinda No. 18G	Fantastic	5,0/5	IDR 831.402	Jl Jamin Ginting , Sempajaya, Berastagi, Karo Regency, North Sumatra 22152
14	Villa Berastagi Highland 7 LL	Good	3,8/5	IDR 801.900	JALAN LAU GUMBA BLOK 7LL , Berastagi, Karo, Sumatera Utara
15	Caesarea Villa	Fantastic	4,8/5	IDR 334.854	Jl. Kenangan No.10, Gundaling I, Kec. Berastagi, Kabupaten Karo, Sumatera Utara 22153
16	Villa Berastagi Resort V86	Tidak tersedia	Akomo dasi ini lagi penuh	Tidak tersedia	Tidak tersedia
17	Villa Highland Berastagi no 9A (Awe Villa)	Tidak tersedia	Akomo dasi ini lagi penuh	Tidak tersedia	Tidak tersedia
18	Villa Bukit Mas Best Seller B3	Tidak tersedia	Akomo dasi ini lagi penuh	Tidak tersedia	Tidak tersedia
19	Sunroom Villa Bukit Mas	Tidak tersedia	Akomo dasi ini lagi penuh	Tidak tersedia	Jalan Besar Medan Berastagi, Sempajaya, Berastagi, North Sumatra, ID
20	Villa Bukit Indah	Tidak tersedia	Akomo dasi ini lagi penuh	Tidak tersedia	Tidak tersedia
21	Villa Bukit Mas Berastagi-Strategic Location, New	Tidak tersedia	Akomo dasi ini lagi penuh	Tidak tersedia	Tidak tersedia
22	Ulina Villa Berastagi	Tidak tersedia	Yuk, pilih tanggal yang tersedia	IDR 872.640	Jl. Lau Gumba, Sempajaya, Kec. Berastagi, Kabupaten Karo, Sumatera Utara

			di bawah ini atau ubah tanggal nginep mu.		
23	Villa Ncole	Tidak tersedia	Akomodasi ini lagi penuh	Tidak tersedia	Jl. Doulu, Daulu, Kec. Berastagi, Kabupaten Karo, Sumatera Utara
24	Villa Mutiara Asri by Dinda No 18E	Okay	3,4/5	IDR 795.000	Jl Jamin Ginting, Sempajaya, Berastagi, Karo Regency, North Sumatra
25	Villa Mutiara Asri by Dinda No 28K	Fantastic	5,0/5	IDR 914.544	Jl Jamin Ginting , Sempajaya, Berastagi, Karo Regency, North Sumatra 22152
26	Villa Brastagi Resort Jalan Mimpin Tua Blok C No. 41	Good	4,3/5	Tidak tersedia	Villa Brastagi Resort Jalan Mimpin Tua Blok C No. 41 Villa Brastagi Resort, Berastagi, Berastagi, Indonesia
27	VILLA BERASTAGI HIGHLAND BLOK 9X	Tidak tersedia	Akomodasi ini lagi penuh	Tidak tersedia	Tidak tersedia
28	Villa Berastagi Resort depan Kolam Renang V25	Tidak tersedia	Akomodasi ini lagi penuh	Tidak tersedia	Tidak tersedia
29	Villa Berastagi Resort V25 depan Kolam Renang	Terrible	1,4/5	Tidak tersedia	Tidak tersedia
30	Villa Lutze Gundaling Berastagi	Fantastic	5,0/5	Tidak tersedia	Jalan Gundaling, 22152 Berastagi, Indonesia

31	Villa Highland (8V) berastagi	Good	3,6/5	Tidak tersedia	Tidak tersedia
32	Villa Mahong & Dentist	Fantastic	5,0/5	IDR 879.254	Tidak tersedia
33	VILLA BERASTAGI BUKIT INDAH BLOK D10	Tidak tersedia	Akomodasi ini lagi penuh	Tidak tersedia	Tidak tersedia
34	Villa Berastagi Mas	Fantastic	5,0/5	IDR 1.215.000	Villa Berastagi Mas, Berastagi, Indonesia
35	VILLA BERASTAGI BUKIT INDAH BLOK E1	Tidak tersedia	Akomodasi ini lagi penuh	Tidak tersedia	Tidak tersedia
36	Villa Highland No 7 P Berastagi View	Tidak tersedia	Akomodasi ini lagi penuh	Tidak tersedia	Komplek Villa Highland Berastagi
37	Villa Bukit Indah Brastagi Ternyaman	Fantastic	5,0/5	Tidak tersedia	Jl. Mimpin Tua, Brastagi, Tanah Karo, Sumatera Utara Jl. Mimpin Tua, Brastagi, Tanah Karo, Sumatera Utara, Berastagi, Indonesia
38	Villa Highland No 7 M Berastagi View	Tidak tersedia	Akomodasi ini lagi penuh	Tidak tersedia	Komplek Villa Highland Berastagi
39	Villa Highland Berastagi No 9C	Tidak tersedia	Akomodasi ini lagi penuh	Tidak tersedia	Tidak tersedia
40	Villa Bukit Mas Berastagi-Strategic Location, New	Tidak tersedia	Akomodasi ini lagi penuh	Tidak tersedia	Tidak tersedia

Hasil pengujian kedua pada **Tabel 4.4**, *web crawling* telah berhasil dilakukan. Dengan 40 data *villa* yang berhasil di-*crawling* dengan kata kunci “*villa*”. Adapun informasi yang diambil yakni nama, kesan, *rating*, harga, dan alamat. Terdapat 17 dari 40 data *villa* yang tidak memiliki keterangan yang lengkap dikarenakan *villa* tersebut sedang penuh pada saat proses *crawling* sehingga menampilkan kalimat “Tidak Tersedia”, “Akomodasi ini lagi penuh”, dan “Yuk, pilih tanggal yang tersedia di bawah ini atau ubah tanggal nginepmu.”. Keterangan yang tidak lengkap tersebut meliputi kesan, *rating* dan harga. Didapati hasil *villa* terbaik berdasarkan *rating* dari 40 *villa* yang berhasil di-*crawling* dengan *rating* 5 antara lain:

- a. Villa Mutiara Asri by Dinda No 8BB
- b. Villa Manchester Blok 1 H
- c. Villa Mutiara Asri by Dinda No. 18G
- d. Villa Mutiara Asri by Dinda No 28K
- e. Villa Lutze Gundaling Berastagi
- f. Villa Mahong & Dentist
- g. Villa Berastagi Mas
- h. Villa Bukit Indah Brastagi Ternyaman

4.4.3. Pengujian Ketiga

Pengujian ketiga ini dilakukan pencarian untuk menemukan penginapan di Berastagi dengan kata kunci “*homestay*”.

Masukkan Kata Kunci

homestay

Link Hotel: <https://www.tiket.com/hotel/indonesia/berastagi-nachelle-homestay-203001551897283930?checkin=2023-11-30&checkout=2023-12-01&room=1&adult=1&soldOut=>

Link Hotel: <https://www.tiket.com/hotel/indonesia/nachelle-homestay-404001617246135350?checkin=2023-11-30&checkout=2023-12-01&room=1&adult=1&soldOut=>

Link Hotel: <https://www.tiket.com/homes/indonesia/vulkaan-homestay-609001693767239365?checkin=2023-11-30&checkout=2023-12-01&room=1&adult=1&soldOut=>

Link Hotel: <https://www.tiket.com/hotel/indonesia/vulkaan-homestay-610001696560033342?checkin=2023-11-30&checkout=2023-12-01&room=1&adult=1&soldOut=>

Link Hotel: <https://www.tiket.com/hotel/indonesia/oyo-90209-dk-singarimbun-berastagi-homestay-412001639107884565?checkin=2023-11-30&checkout=2023-12-01&room=1&adult=1&soldOut=>

Link Hotel: <https://www.tiket.com/hotel/indonesia/friendjoss-homestay-laugumba-redpartner-411001637613123213?checkin=2023-11-30&checkout=2023-12-01&room=1&adult=1&soldOut=>

Link Hotel: <https://www.tiket.com/hotel/indonesia/rtm-stable-homestay-411001636699633077?checkin=2023-11-30&checkout=2023-12-01&room=1&adult=1&soldOut=>

Link Hotel: <https://www.tiket.com/homes/indonesia/berastagi-mountain-view-homestay-pizzeria-311001604557539552?checkin=2023-11-30&checkout=2023-12-01&room=1&adult=1&soldOut=>

Link Hotel: <https://www.tiket.com/hotel/indonesia/kaesa-homestay-301001579900189141?checkin=2023-11-30&checkout=2023-12-01&room=1&adult=1&soldOut=>

Link Hotel: <https://www.tiket.com/homes/indonesia/smileys-homestay-601001674361477753?checkin=2023-11-30&checkout=2023-12-01&room=1&adult=1&soldOut=>

Link Hotel: <https://www.tiket.com/homes/indonesia/ulina-homestay-berastagi-610001697942169933?checkin=2023-11-30&checkout=2023-12-01&room=1&adult=1&soldOut=>

Halaman selanjutnya tidak tersedia atau sudah selesai.

Gambar 4.8 Proses Pengujian Ketiga

Hasil pengujian ketiga pada **Gambar 4.9** menggunakan kata kunci “homestay” sebagai parameter pencarian dengan *seed* URL yang digunakan yaitu “<https://www.tiket.com/hotel/indonesia/area/berastagi-108001534490301065>”. Hasil yang diperoleh menemukan sebanyak 11 *homestay* setelah proses *crawling* dan pencocokan *string* nama.

Tabel 4.5 Hasil Pengujian Ketiga

No	Nama	Kesan	Rating	Harga	Alamat
1	Berastagi Nachele Homestay	Good	4,3/5	IDR 153.900	Simpang Ujung Aji, Complex BBC No. 1, Berastagi, North Sumatra, ID
2	Nachele Homestay	Good	4,6/5	IDR 215.000	Jalan Jamin Ginting. Simpang Ujung Aji, Komplek BBC
3	Vulkaan Homestay	Tidak tersedia	Akomo dasi ini lagi penuh	Tidak tersedia	Tidak tersedia
4	Vulkaan Homestay	Tidak tersedia	Udah punya akun belum?	IDR 279.311	Jln.udara,semangat, Kec.Merdeka,Kab.K aro
5	OYO 90209	Bad	2,2/5	Tidak	OYO 90209 DK

	DK Singarimbun Berastagi Homestay			tersedia	Singarimbun Berastagi Homestay
6	Friendjoss Homestay Laugumba RedPartner	Good	4,0/5	IDR 105.802	Jl. Lau Gumba No.37, Sempajaya, Kec. Berastagi, Kabupaten Karo, Sumatera Utara 22152, Indonesia
7	RTM Stable & Homestay	Good	4,0/5	IDR 337.500	Jaranguda, Merdeka, Berastagi, Kabupaten Karo, Sumatera utara
8	Berastagi Mountain View Homestay & Pizzeria	Fantastic	5,0/5	IDR 202.500	Jalan Ujung Aji Gang Sampang Rukur 398, Berastagi, Berastagi, Indonesia
9	Kaesa Homestay	Fantastic	4,7/5	IDR 230.001	Jalan Kaliaga 3
10	Smiley's Homestay	Tidak tersedia	Akomodasi ini lagi penuh	Tidak tersedia	Jalan Sekolah Kelurahan Gundaling I, Berastagi, Berastagi, Berastagi, Indonesia
11	Ulina Homestay Berastagi	Tidak tersedia	Akomodasi ini lagi penuh	Tidak tersedia	Tidak tersedia

Hasil pengujian ketiga pada **Tabel 4.5**, *web crawling* telah berhasil dilakukan. Dengan 11 data *homestay* yang berhasil di-*crawling* dengan kata kunci “*homestay*”. Adapun informasi yang diambil yakni nama, kesan, *rating*, harga, dan alamat. Terdapat 17 dari 11 data *homestay* yang tidak memiliki keterangan yang lengkap dikarenakan *homestay* tersebut sedang penuh pada saat proses *crawling* sehingga menampilkan kalimat “Tidak Tersedia”, “Akomodasi ini lagi penuh”, dan “Udah punya akun belum?”. Keterangan yang tidak lengkap tersebut meliputi kesan, *rating* dan harga. Didapati hasil *homestay* terbaik berdasarkan *rating* dari 11 *homestay* yang berhasil di-*crawling* yakni Berastagi Mountain View Homestay & Pizzeria dengan *rating* 5.

4.4.4. Pengujian Keempat

Pengujian keempat ini dilakukan pencarian untuk menemukan penginapan di Berastagi dengan kata kunci “cottage”.

```
Masukkan Kata Kunci
cottage
Link Hotel: https://www.tiket.com/homes/indonesia/para-para-cafe-cottage-605001
683516104894?checkin=2023-11-30&checkout=2023-12-01&room=1&adult=1&soldOut=
Link Hotel: https://www.tiket.com/hotel/indonesia/danau-toba-international-cott
age-berastagi-212001575883808048?checkin=2023-11-30&checkout=2023-12-01&room=1&
adult=1&soldOut=
Link Hotel: https://www.tiket.com/hotel/indonesia/alloyna-country-cottage-10800
1534490378463?checkin=2023-11-30&checkout=2023-12-01&room=1&adult=1&soldOut=
Halaman selanjutnya tidak tersedia atau sudah selesai.
```

Gambar 4.9 Proses Pengujian Keempat

Hasil pengujian keempat pada **Gambar 4.10** menggunakan kata kunci “cottage” sebagai parameter pencarian dengan *seed* URL yang digunakan yaitu “https://www.tiket.com/hotel/indonesia/area/berastagi-108001534490301065”.

Hasil yang diperoleh menemukan sebanyak 3 *cottage* setelah proses *crawling* dan pencocokan *string* nama.

Tabel 4.6 Hasil Pengujian Keempat

No	Nama	Kesan	Rating	Harga	Alamat
1	Para para café & cottage	Tidak tersedia	Akomo dasi ini lagi penuh	Tidak tersedia	Tidak tersedia
2	Danau Toba Internationa l Cottage Berastagi	Good	3,87/5	IDR 263.940	Jl. Gundaling No.66
3	Alloyna Country Cottage	Good	3,84/5	IDR 517.262	Jl. Letjend Jamin Ginting, No. 007, Sempajaya, Karo Regency, North Sumatra, Indonesia

Hasil pengujian keempat pada **Tabel 4.6**, *web crawling* telah berhasil dilakukan. Dengan 3 data *cottage* yang berhasil di-*crawling* dengan kata kunci “cottage”. Adapun informasi yang diambil yakni nama, kesan, *rating*, harga, dan alamat. Terdapat 1 dari 3 data *cottage* yang tidak memiliki keterangan yang lengkap dikarenakan *cottage* tersebut sedang penuh pada saat proses *crawling* sehingga menampilkan kalimat “Tidak Tersedia” dan “Akomodasi ini lagi

penyutuh”. Keterangan yang tidak lengkap tersebut meliputi kesan, *rating* dan harga. Didapati hasil *cottage* terbaik berdasarkan *rating* dari 3 *cottage* yang berhasil di-*crawling* yakni Danau Toba International Cottage Berastagi dengan *rating* 3,87.

4.4.5. Pengujian Kelima

Pengujian kelima ini dilakukan pencarian untuk menemukan penginapan di Berastagi tanpa memasukkan kata kunci. Didapatkan hasil sebanyak 106 data penginapan di Berastagi yang tersedia pada *website* tiket.com yang sudah disimpan dalam format dokumen *Excel* seperti diberikan pada **Gambar 4.11**, **Gambar 4.12**, **Gambar 4.13**, dan **Gambar 4.14**.

	Title	Impresi	Rating	Harga	Lokasi
0	Grand Mutiara Hotel Berastagi	Good	4.38/5	IDR 742.000	Jl. Peceren No.168, Berastagi, Kab. Karo, Nor
1	RedDoorz Sapo Rumbia 2 Berastagi	Good	4.30/5	IDR 166.531	Jl. Kenangan No.1, Karo, Kec. Berastagi, Kabu
2	Penginapan puncak Dua RedPartner	Good	4.16/5	IDR 127.940	Desa, Semangat Gunung, Kec. Berastagi, Kab
3	Villa Berastagi Highland Blok 9KK	Good	4.33/5	IDR 810.000	Jl. Lau Gumba Blok 9KK, Berastagi, Karo, Sur
4	RedDoorz near Mikie Holiday Funland Berastagi	Good	3.51/5	IDR 169.580	Jl. Jamin Ginting No.10, Sempajaya, Kec. Ber
5	Villa Berastagi Highland	Good	4.50/5	IDR 787.500	Jalan lau gumba no 9R
6	Uluna Hotel	Good	4.23/5	Tidak tersedia	Jl. Jamin Ginting No 10
7	Villa Berastagi Highland Blok 8L	Fantastic	4.66/5	IDR 950.000	Jalan Lau Gumba Blok 8L, Sempajaya, Kab. Ki
8	Villa Bukit Indah 48R Brastagi / Berastagi	Good	3.72/5	IDR 738.820	Jl. Mimpin Tua, Sempajaya, Kec. Berastagi, Ki
9	Villa Berastagi Highland 8KK	Good	4.05/5	Tidak tersedia	Jl. Lau Gumba, Sempajaya, Berastagi, Kabupe
10	Airangga Home Stay Syariah Berastagi RedPa Okay	Okay	3.20/5	IDR 104.979	JALAN KENANGA Gg. Surya Indah No.22, Gur
11	Pemandian dan Penginapan Gunung Sibayak	Tidak terse	Akomodas	Tidak tersedia	6GF7+CM7, Jaranguda, Merdeka, Karo Reger
12	Pariama Inn	Good	4.05/5	Tidak tersedia	Desa Semangat Gunung Kec. Merdeka Kab. K
13	Hotel Mexico Berastagi	Okay	3.08/5	IDR 185.037	Jl. Veteran No.18, Tambak Lau Mulgap I, Beri
14	Hotel Miranda at Desa Wisata Berastagi	Good	4.08/5	IDR 104.979	Jalan Pendidikan, No.no 9, Jaranguda, Kecam
15	Pariban Hotel	Good	4.17/5	IDR 382.725	Jaranguda, Kec. Merdeka, Kabupaten Karo, S
16	Mikie Holiday Resort	Good	4.30/5	IDR 843.729	Jl. Letjend Jamin Ginting, Berastagi, Karo, Sur
17	Hotel Kalang Ulu Berastagi	Good	4.12/5	IDR 439.900	Jl. Mumpin Tua No.99 (Arah Vihara Dharma S
18	Galaxy Villa manage by Pal's Berastagi	Good	3.75/5	Tidak tersedia	Jl. Kenangan, Gundaling I, Kec. Berastagi, Kab
19	Sinabung Hills Berastagi	Good	4.09/5	IDR 791.820	Jl. Kolam Renang, Gundaling I, Kec. Berastagi
20	Panorama Guesthouse Syariah	Good	4.34/5	IDR 202.354	Gundaling 1, Berastagi, Karo
21	Villa Mutiara Asri by Dinda No 888	Fantastic	5.00/5	IDR 905.489	Jl Jamin Ginting, Sempajaya, Berastagi, Karo
22	Villa Manchester Blok 1 H	Fantastic	5.00/5	IDR 843.374	Jl. Mimpin Tua No.63, Sempajaya, Kec. Beras
23	Rudang Hotel & Resort Berastagi	Good	3.78/5	IDR 373.915	Jl. Jamin Ginting, No. 16, Berastagi
24	Villa Mutiara Asri by Dinda No 8D	Good	4.10/5	IDR 914.635	Jl. Jamin Ginting, Sempajaya
25	Villa Keluarga Bre Batunanggar	Good	4.35/5	IDR 307.189	Jl. Kolam Renang, Gundaling I, Berastagi, Kab
26	Hotel New Dieng By Helocus	Good	3.70/5	IDR 263.061	Jl. Udara No.8, Gundaling I, Kec. Berastagi, Ki
27	Villa Mutiara Asri by Dinda No 18A	Okay	3.40/5	IDR 923.781	Jl Jamin Ginting, Sempajaya, Berastagi, Karo
28	Green Garden Hotel Berastagi	Good	3.73/5	IDR 505.885	Jl. Peceren No.30, Berastagi, Tanah Karo, Sun
29	Nachelle Homestay	Good	4.57/5	IDR 215.000	Jalan Jamin Ginting. Simpang Ujung Aji, Kom
30	Hotel Sibayak International Berastagi	Good	3.99/5	IDR 699.602	Jl. Merdeka, Berastagi - Sumatera Utara

Gambar 4.10 Informasi Penginapan Berastagi dari *Website* tiket.com (1)

31	Villa Mutiara Asri by Dinda No. 18G	Fantastic	5.00/5	IDR 923.781	Jl Jamin Ginting , Sempajaya, Berastagi, Karo
32	Vulkaan Homestay	Tidak terse	Udah	puny IDR 279.311	Jln.udara,semangat,Kec.Merdeka,Kab.Karo
33	Berastagi Nachele Homestay	Good	4.32/5	IDR 170.999	Simpang Ujung Aji, Complex BBC No. 1, Bera
34	Villa Lutze Gundaling Berastagi	Tidak terse	Akomodas	Tidak tersedia	Jalan Gundaling No. 184 , Berastagi , Sumate
35	Caesarea Villa	Fantastic	4.80/5	IDR 301.369	Jl. Kenangan No.10, Gundaling I, Kec. Berast
36	Villa Berastagi Highland 7 LL	Good	3.80/5	IDR 909.000	JALAN LAU GUMBA BLOK 7LL , Berastagi, Kar
37	Villa Berastagi Resort V86	Tidak terse	Akomodas	Tidak tersedia	Tidak tersedia
38	Villa Highland Berastagi no 9A (Awe Villa)	Tidak terse	Akomodas	Tidak tersedia	Tidak tersedia
39	3 Berastagi Hotel	Good	3.53/5	IDR 193.556	Jl. Jamin Ginting No.134, Sempajaya, Kec. Be
40	Traditional room family para para	Tidak terse	Akomodas	Tidak tersedia	Tidak tersedia
41	Para para cafe & cottage	Tidak terse	Akomodas	Tidak tersedia	Tidak tersedia
42	Vulkaan Homestay	Tidak terse	Akomodas	Tidak tersedia	Tidak tersedia
43	OYO 1696 Sapo Rumbia	Okay	3.44/5	IDR 165.812	1, Jl. Sekolah, Karo
44	Penginapan Imanuel Lau Gara by ZUZU	Tidak terse	Akomodas	Tidak tersedia	Jl.Jaranguda,Ds.Semangat Gunung,Merdeka,
45	OYO 1701 Bagus Inn	Good	3.70/5	Tidak tersedia	5, Jl. Jamin Ginting, Sempajaya, Kec. Berast
46	Kamuna Garden 2 Sidebuk Debuk	Okay	3.40/5	IDR 12.825.000	6G7F+MJ9, Jaranguda, Merdeka, Karo Regen
47	Villa Bukit Mas Best Seller B3	Tidak terse	Akomodas	Tidak tersedia	Tidak tersedia
48	Sunroom Villa Bukit Mas	Tidak terse	Akomodas	Tidak tersedia	Jalan Besar Medan Berastagi, Sempajaya, Be
49	Villa Bukit Indah	Tidak terse	Akomodas	Tidak tersedia	Tidak tersedia
50	RedDoorz @ Jalan Ujung Aji Berastagi	Tidak terse	Udah	puny IDR 180.992	Jalan Ujung Aji, Gang Sampang Rukur 398, Ru
51	Penginapan SEJOLIE Sibayak	Okay	3.40/5	IDR 309.096	Jl. Lau Gara, Desa Semangat Gunung Kec. Me
52	Villa Ncole	Tidak terse	Akomodas	Tidak tersedia	Jl. Doulu, Daulu, Kec. Berastagi, Kabupaten K
53	Penginapan Imanuel Lau Gara	Okay	3.00/5	Tidak tersedia	Desa Semangat Gunung, Kec. Merdeka, Kabu
54	Penginapan Kamuna Garden 3	Tidak terse	Akomodas	IDR 151.464	Jalan Pertamina Desa Semangat Gunung, Kec
55	OYO 90209 DK Singarimbun Berastagi Homes	Bad	2.20/5	Tidak tersedia	OYO 90209 DK Singarimbun Berastagi Homes
56	Bangkit Nan Jaya	Okay	Akomodas	IDR 101.251	Jl. Jaranguda, Jaranguda, Merdeka, Berastagi
57	Kalincayo Guesthouse Sidebuk Debuk	Tidak terse	Akomodas	IDR 14.392.500	6G7F+98X Sidebuk Debuk Thermal Baths Ber
58	OYO 1440 New Sejarah	Okay	3.00/5	Tidak tersedia	No. 1, Jl., Trimurti, Berastagi, ID
59	As Cafe Resto Hotel	Okay	3.00/5	IDR 12.825.000	Desa Jaranguda Simpang Tiga, gunung sibaya
60	Sapo Karo Rest House	Good	3.90/5	IDR 214.637	Jalan Pervira, Gundaling I, Berastagi, Gundal
61	Penginapan Biringta	Tidak terse	Akomodas	Tidak tersedia	DESA SEMANGAT GUNUNG, KECAMATAN MI
62	Suite Pakar Hotel	Bad	2.33/5	IDR 8.400.000	Jalan Letjen Jamin 2 Ginting, Desa Merdeka,

Gambar 4.11 Informasi Penginapan Berastagi dari *Website* tiket.com (2)

63	Penginapan Kamuna Garden 1	Good	3.60/5	IDR 12.825.000	6G7F+MJ9, Jaranguda, Merdeka, Karo Regen
64	Kaliaga Bungalow	Tidak terse	Udah	puny IDR 232.804	Jl. Gundaling 1, Jl Pervira no 219, Berastagi
65	Danau Toba International Cottage Berastagi	Good	3.86/5	IDR 263.940	Jl. Gundaling No.66
66	OYO 2322 Dien Karona Hotel	Tidak terse	Akomodas	Tidak tersedia	Jl. Jaranguda No.148, Jaranguda, Merdeka
67	Ulina Villa Berastagi	Tidak terse	Yuk, pilih t	IDR 864.000	Jl. Lau Gumba, Sempajaya, Kec. Berastagi, Ka
68	OYO 93062 Sibayak Multinational Rest House	Okay	3.00/5	IDR 128.833	08, Jl. Pendidikan No 08, Jaranguda, Merdeka
69	Sibayak Multi National Rest House	Good	4.10/5	IDR 177.758	Jl. Pendidikan No. 08, Jaranguda Berastagi, Ir
70	RedDoorz @ Permata Ivana Berastagi	Okay	3.18/5	IDR 185.602	Jl. Jamin Ginting, Sempajaya, Kec. Berastagi,
71	Alloyna Country Cottage	Good	3.84/5	IDR 477.020	Jl. Letjend Jamin Ginting, No. 007, Sempajaya
72	OYO 581 Serdika Pavilion	Okay	2.78/5	IDR 87.371	Jl. Pervira No.227, Gundaling I, Berastagi, Ka
73	Shohwah Syariah	Good	3.90/5	IDR 231.822	Jl. Pervira No.219A, Gundaling I, Kec. Berast
74	Friendjoss Homestay Laugumba RedPartner	Good	3.99/5	IDR 105.802	Jl. Lau Gumba No.37, Sempajaya, Kec. Berast
75	Villa Mutiara Asri by Dinda No 18E	Okay	3.40/5	IDR 795.000	Jl Jamin Ginting, Sempajaya, Berastagi, Karo
76	Penginapan Puncak DP	Okay	3.40/5	IDR 195.889	Jl. Jaranguda, Jaranguda, Merdeka, Kabupaten
77	Villa Mutiara Asri by Dinda No 28K	Fantastic	5.00/5	IDR 914.635	Jl Jamin Ginting , Sempajaya, Berastagi, Karo
78	RedDoorz near Bukit Kubu Berastagi	Okay	3.45/5	IDR 195.139	Jalan Jamin Ginting No.5, Sempajaya, Karo Ru
79	RTM Stable & Homestay	Good	4.02/5	IDR 337.500	Jaranguda, Merdeka, Berastagi, Kabupaten
80	Berastagi Mountain View Homestay & Pizzeri	Fantastic	5.00/5	IDR 202.500	Jalan Ujung Aji Gang Sampang Rukur 398, Be
81	WELCOMETO WISMA SUNRISE GUEST HOUSE	Good	4.06/5	Tidak tersedia	Jl. Pervira Atas No. 5 Berastagi
82	Villa Brastagi Resort Jalan Mimpin Tua Blok C	Good	4.31/5	Tidak tersedia	Villa Brastagi Resort Jalan Mimpiin Tua Blok C
83	VILLA BERA TAGI HIGHLAND BLOK 9X	Tidak terse	Akomodas	Tidak tersedia	Tidak tersedia
84	Kaesa Homestay	Fantastic	4.71/5	IDR 230.001	Jalan Kaliaga 3
85	Villa Berastagi Resort depan Kolam Renang V	Tidak terse	Akomodas	Tidak tersedia	Tidak tersedia
86	Villa Berastagi Resort V25 depan Kolam Rena	Terrible	1.40/5	Tidak tersedia	Tidak tersedia
87	Villa Lutze Gundaling Berastagi	Fantastic	5.00/5	Tidak tersedia	Jalan Gundaling, 22152 Berastagi, Indonesia
88	OYO 1543 Sibayak Ncole	Bad	1.60/5	Tidak tersedia	Jl. Doulu Rajabemeh Desa Doulu, Jaranguda,
89	Villa Highland (8V) berastagi	Good	3.60/5	Tidak tersedia	Tidak tersedia
90	Villa Mahong & Dentist	Fantastic	5.00/5	IDR 879.254	Tidak tersedia
91	VILLA BERA TAGI BUKIT INDAH BLOK D10	Tidak terse	Akomodas	Tidak tersedia	Tidak tersedia
92	Hotel Grand Orri Berastagi	Tidak terse	Akomodas	Tidak tersedia	6G36+6PF, Jalan Orri (from, Jl. Jamin Ginting
93	Villa Berastagi Mas	Fantastic	5.00/5	IDR 1.215.000	Villa Berastagi Mas, Berastagi, Indonesia
94	Penginapan puncak Dua	Tidak terse	Udah	puny IDR 130.163	Semangat

Gambar 4.12 Informasi Penginapan Berastagi dari *Website* tiket.com (3)

95	VILLA BERASTAGI BUKIT INDAH BLOK E1	Tidak terse Akomodas Tidak tersedia	Tidak tersedia
96	Villa Highland No 7 P Berastagi View	Tidak terse Akomodas Tidak tersedia	Tidak tersedia
97	Villa Bukit Indah Brastagi Ternyaman	Fantastic 5.00/5 Tidak tersedia	Jl. Mimpin Tua, Brastagi, Tanah Karo, Sumate
98	Smiley's Homestay	Tidak terse Akomodas Tidak tersedia	Jalan Sekolah Kelurahan Gundaling I, Berasta
99	Villa Highland No 7 M Berastagi View	Tidak terse Akomodas Tidak tersedia	Tidak tersedia
100	Villa Highland Berastagi No 9C	Tidak terse Akomodas Tidak tersedia	Tidak tersedia
101	Ulina Homestay VIP Berastagi	Tidak terse Uдах punya IDR 202.500	Tidak tersedia
102	Gintingta Homstay	Tidak terse Akomodas Tidak tersedia	Tidak tersedia
103	Ron's Hostel	Tidak terse Akomodas Tidak tersedia	Jl. Perwira gg Kaliaga No. 58 Berastagi
104	Villa Bukit Mas Berastagi- Strategic Location,	Tidak terse Akomodas Tidak tersedia	Tidak tersedia
105	Ulina Homestay Berastagi	Tidak terse Akomodas Tidak tersedia	Tidak tersedia

Gambar 4.13 Informasi Penginapan Berastagi dari *Website* tiket.com (4)

Hasil pengujian kelima pada **Gambar 4.11**, **Gambar 4.12**, **Gambar 4.13**, dan **Gambar 4.14**, *web crawling* telah berhasil dilakukan. Dengan 106 data penginapan yang berhasil di-*crawling* tanpa kata kunci. Adapun informasi yang diambil yakni nama, kesan, *rating*, harga, dan alamat. Terdapat 49 dari 106 data penginapan yang tidak memiliki keterangan yang lengkap dikarenakan penginapan tersebut belum memiliki informasi yang lengkap atau sedang penuh pada saat proses *crawling* sehingga menampilkan kalimat “Tidak Tersedia”, “Akomodasi ini lagi penuh”, “Yuk, pilih tanggal yang tersedia di bawah ini atau ubah tanggal nginepmu.”, dan “Udah punya akun belum?”. Keterangan yang tidak lengkap tersebut meliputi kesan, *rating* dan harga.

Tabel 4.7 Persentase Infomasi Penginapan Hasil *Crawling*

Informasi Penginapan	
Infomasi Lengkap	Infomasi Tidak Lengkap
54%	46%

Pengujian sistem pada penelitian ini dilakukan sebanyak 5 kali dengan memasukkan kata kunci yang berbeda-beda antara lain “hotel”, “villa”, “homestay”, “cottage”, dan tanpa kata kunci. Adapun total data hasil *crawling* yaitu 106 penginapan. Pada **Tabel 4.7** terdapat infomasi penginapan yang lengkap memiliki persentase 54% dan yang tidak lengkap yakni 46%. Adapun kelengkapan data yang dimaksud informasi tersebut meliputi nama, kesan, *rating*, harga, dan alamat penginapan. Hasil yang diperoleh dari sistem yang telah dikembangkan sesuai dengan informasi yang tertera di *website* sehingga dapat disimpulkan bahwa sistem dapat bekerja dengan baik dan benar.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil analisis, perancangan, implementasi, dan pengujian sistem, maka dapat disimpulkan sebagai berikut:

1. Hasil yang diperoleh dari sistem yang telah dikembangkan sesuai dengan informasi yang tertera di *website* tiket.com, sistem berkerja dengan baik dan benar.
2. Hasil dari pengujian sistem dengan kata kunci “hotel” didapatkan hasil 15 hotel, kata kunci “*villa*” berjumlah 40 *villa*, kata kunci “*homestay*” berjumlah 11 *homestay*, kata kunci “*cottage*” berjumlah 3 *cottage*, dan pengujian tanpa kata kunci menghasilkan seluruh penginapan yang ada di Berastagi.
3. Total data hasil *crawling* berjumlah 106 penginapan dengan persentase informasi penginapan yang lengkap yaitu 54% dan yang tidak lengkap yaitu 46%. Kelengkapan data informasi meliputi nama, kesan, *rating*, harga, dan alamat penginapan.
4. Algoritma *Raita* bekerja dengan pencocokan *string* sesuai dengan susunan karakter yang terdapat dalam *string* dicocokkan dan memiliki jumlah maupun urutan karakter yang sama dalam *string* dapat bekerja pada data berskala kecil.
5. Sistem *web crawling* pada *website* tiket.com dapat digunakan untuk pengambilan informasi penginapan. Akan tetapi dikarenakan adanya perubahan pada *source code website* tiket.com, penulis tidak dapat mengekstraksi data terbaru dari tiket.com sehingga data yang digunakan pada penelitian ini adalah data hasil *crawling* pada November hingga Desember 2023.

5.2 Saran

Adapun beberapa saran yang dapat menjadi pertimbangan dalam penelitian selanjutnya adalah sebagai berikut:

1. Untuk pengembangan tahap selanjutnya, diharapkan menggunakan *seed URL* lebih banyak, sehingga informasi yang dihasilkan juga lebih banyak.

2. Untuk penelitian berikutnya, diharapkan agar sistem dapat disesuaikan dengan *source code* dari *website* yang menjadi target *crawling* sehingga memungkinkan pengambilan data yang selalu terkini dari *website* yang dituju.
3. Untuk pengembangan penelitian selanjutnya, diharapkan agar sistem divisualisasikan lebih baik.
4. Untuk kedepannya, penelitian diharapkan menggunakan spesifikasi komputer dan jaringan yang lebih baik.

DAFTAR PUSTAKA

- Almaqbal, I.S.H., Al, F.M.A., Khufairi, Khan, M.S., Bhat, A.Z., & Ahmed, I. (2019). Web Scrapping: Data Extraction from Websites. *Journal of Student Research*, 2. <https://doi.org/10.47611/jsr.vi.942>
- Bawanto, D. R., & Rosmawanti, N. (2017). Perbandingan Algoritma Binary Search dan Raita dalam Pencarian Data. *Jurnal Ilmiah Teknik Informatika dan Sistem Informasi (JUTISI)*, 6(1), 1311-1448. <http://dx.doi.org/10.35889/jutisi.v6i1.225>
- Cook, Merrill. (2024). Seed URL. <https://blog.diffbot.com/knowledge-graph-glossary/seed-url/>
- Djufri, M. (2020). Penerapan Teknik Web Scraping untuk Penggalan Potensi Pajak (Studi Kasus pada Online Market Place Tokopedia, Shopee dan Bukalapak). *Jurnal BPPK*, 13(2), 66. <https://doi.org/10.48108/jurnalbppk.v13i2.636>
- Ginting, G.L. (2019). Analisa Algoritma Raita pada Automatic Hyperlink. *Prosiding Seminar Nasional Riset Information Science (SENARIS)*, 886-893. <http://dx.doi.org/10.30645/senaris.v1i0.96>
- Havidz, H.B.H., Hudaya, A., & Ali, H. (2020). Model of Consumer Trust on Travel Agent Online: Analysis of Perceived Usefulness and Security on Re-Purchase Interest (Case Study tiket.com). *Dinasti International Journal of Economics, Finance & Accounting (DIJEFA)*, 1(1), 112. <https://doi.org/10.38035/dijefa.v1i1.227>
- Hidayatullah, Taufiq. (2022). Implementasi Algoritma Base-64 dalam Mengamankan URL (Uniform Resource Locator) Website Layanan Pengaduan Masyarakat Desa Bojongraharja. *Jurnal Media Infotama*, 18(2), 337. <https://doi.org/10.37676/jmi.v18i2.2937>
- Hien, N.L.H., Tien, T.Q., & Hieu, N.V. (2020). Web Crawler: Design and Implementation for Extracting Article-Like Contents. *Cybernetics and Physics*, 9(3), 144-149.
- Hudaa, S., Nguyen, P.T., Lestari, S.P., Gunawan, G., & Supiyandi, S. (2020). Data Search using Raita Algorithm. *Journal of Critical Reviews*, 7(1), 72. <http://dx.doi.org/10.22159/jcr.07.01.14>
- Jain, S., & Rao, A.L.N. (2013). A Comparative Performance Analysis of

- Approximate String Matching. *International Journal of Innovative Technology and Exploring Engineering (IJITEE)*, 3(5), 126-128.
- Jeong, C.S., Ryu, K.H., Lee, J.Y., & Jung, K.D. (2020). Deep Learning-based Tourism Recommendation System using Social Network Analysis. *International Journal of Internet, Broadcasting and Communication*, 12(2), 116-117. <https://doi.org/10.7236/IJIBC.2020.12.2.113>
- Lengkong, C.M., Sengkey, M., & Brave A.S. (2019). Sistem Informasi Pariwisata Berbasis Web di Kabupaten Minahasa. *Jurnal Teknik Informatika*, 14(1), 16. <https://doi.org/10.35793/jti.14.1.2019.23780>
- Lubis. Ikhwan Arjuna. (2021). *Penerapan Web Crawling Menggunakan Algoritma Horspool sebagai Filter pada Pengumpulan Informasi Restoran di Kota Medan*. Medan: Universitas Sumatera Utara. <https://repositori.usu.ac.id/handle/123456789/44718>
- Marbun, N., Zarlis, M., Hartama, D., Mesran, & Sitompul, B.J.D. (2019). Implementasi Algoritma Raita pada Pencarian Katalog Alkes. *Seminar Nasional Sains & Teknologi Informasi (SENSASI)*. 521.
- Munawar, A., Raharjo, E.H.B., & Setiyowati, L. (2022). Perancangan Prototype Sistem Informasi Penyewaan Mesin Fotocopy Pada Cv. Faida Gemilang Bekasi. *Jurnal Ilmiah ILKOMINFO-Jurnal Ilmu Komputer dan Informatika*, 5(2), 81. <https://doi.org/10.47324/ilkominfo.v5i2.152>
- Nandwani, U., Mishra, R., Patil, A., & Siddiqui, W. (2021). Data Analysis by Web Scraping using Python. *International Journal for Research in Engineering Application & Management (IJREAM)*, 7, 15.
- Ou, T.Y., Lee, Y.C., Chang, T.H., Lee, S.H., & Tsai, W.L. (2023). Design and Implementation of a Recommendation System for Buying Fresh Foods Online Based on Web Crawling. *Journal of Advanced Computational Intelligence and Intelligent Informatics*, 27(2), 273-276. <https://doi.org/10.20965/jaciii.2023.p0271>
- Pradana, Egi Gusniawan. (2022). Implementasi Web Crawler untuk Mencari Harga Barang Termurah dari berbagai Situs E-Commerce Indonesia. *Jurnal Teknologi Pintar*, 2(9), 1-2.
- Prismana, G.L.P.E., Prehanto, D.R., & Nuryana, K.D. (2020). The Design and Implementation of Web Crawler Distributed News Domain Detection System.

- International Joint Conference on Science and Engineering*, 196. 96.
<https://doi.org/10.2991/aer.k.201124.017>
- Rahim, R., et al. (2018). Searching Process with Raita Algorithm and its Application. *Journal of Physics Conference Series*, 1007.
<https://www.doi.org/10.1088/1742-6596/1007/1/012004>
- Romadhon, M.H., Yudhistira, Y., & Mukrodin. (2021). Sistem Informasi Rental Mobil Berbasis Android dan Website Menggunakan Framework Codeigniter 3 Studi Kasus : CV Kopja Mandiri. *Jurnal Sistem Informasi dan Teknologi Peradaban (JSITP)*, 2(1). 31.
- Shamrat, F.M.J.M, Tasnim, Z., Rahman, A.K.M.S., Nobel, N.I., & Hossain, S.A. (2020). An Effective Implementation of Web Crawling Technology to Retrieve Data from the World Wide Web (WWW). *International Journal of Scientific & Technology Research*, 9(1). 1253.
- tiket.com. (2011). Tentang tiket.com. <https://www.tiket.com/info/about-us>
- Tukino, T. (2020). Rancang Bangun Sistem Informasi E-Marketing Pada PT Pulau Cahaya Terang. *Computer Based Information System Journal*, 8(1), 25-33.
- Valba, Ryan Ridho. (2017). *Perbandingan Algoritma Horspool dan Algoritma Raita pada Aplikasi Kamus Bahasa Indonesia – Jerman Berbasis Web*. Medan: Universitas Sumatera Utara.
<http://repositori.usu.ac.id/handle/123456789/2227>
- Yondra, A.S., Triyanto, D., & Bahri, S. (2022). Implementasi Web Scraping untuk Mengumpulkan Informasi Produk dari Situs E-Commerce dan Marketplace dengan Teknik Pemrosesan Paralel. *Coding : Jurnal Komputer Dan Aplikasi*, 10(1), 93-102. <https://dx.doi.org/10.26418/coding.v10i01.52722>

LAMPIRAN

Lampiran *Listing Program*

1. Instalasi *Library*

```
# install selenium and openpyxl
!pip install selenium
!pip install openpyxl
```

2. *Import Library* dan Modul

```
# set options to be headless, ..
import time
import pandas as pd
from selenium import webdriver
from selenium.webdriver.common.by import By
from selenium.webdriver.common.keys import Keys
from selenium.webdriver.support.ui import WebDriverWait
from selenium.webdriver.support import expected_conditions as EC
from selenium.common.exceptions import NoSuchElementException
```

3. *String Matching* dengan Algoritma *Raita*

```
def preBmBc(pat):
    # Fungsi untuk membuat tabel bad character
    lpat = len(pat)
    bmBc = [lpat] * 256
    for i in range(lpat - 1):
        bmBc[ord(pat[i].lower())] = lpat - i - 1
    return bmBc

def raita(pat, s):
    # Fungsi utama pencarian dengan algoritma Raita
    pat = pat.lower() # Konversi pola menjadi lowercase
    s = s.lower()     # Konversi teks menjadi lowercase

    lpat = len(pat)
    n = len(s)
    bmBc = preBmBc(pat) # Buat tabel bad character

    # Quick edge cases
    if lpat == 0 or lpat > n:
        return []

    if lpat == 1:
        return [i for i, c in enumerate(s) if c == pat[0]]

    # Inisialisasi hasil pencarian
    result = []
    firstCh, middleCh, lastCh = pat[0], pat[lpat // 2], pat[lpat - 1]
    j = 0

    # Proses pencarian
    while j <= n - lpat:
```



```

        # Periksa keseluruhan pola pada posisi j
        if s[j:j+lpat].lower() == pat:
            result.append(j)
            j += bmBc[ord(s[j + lpat - 1].lower())]

    return result

```

5. Konfigurasi *WebDriver*

```

options = webdriver.ChromeOptions()
options.add_argument('--headless')
options.add_argument('--no-sandbox')
options.add_argument('--disable-dev-shm-usage')
options.add_argument("user-agent=Mozilla/5.0 (Windows NT 10.0; Win64;
x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/118.0.0.0
Safari/537.36")

```

6. Metode *Web Crawling*

```

# Input pola dari pengguna
print("Masukkan Kata Kunci")
pattern = input()

# Inisialisasi WebDriver
driver = webdriver.Chrome(options=options)

# Buka URL halaman web yang ingin Anda scraping
driver.get("https://www.tiket.com/hotel/indonesia/area/berastagi-108001534490301065")

# Inisialisasi list kosong untuk menyimpan data
data = []

# Fungsi untuk mengklik tombol "Selanjutnya"
def click_next_button():
    try:
        time.sleep(3) # Tunggu beberapa detik untuk memastikan halaman selesai dimuat
        next_button = driver.find_element(By.CSS_SELECTOR, 'a.btn-pages.active + a.btn-pages')
        next_button.click()
        return True
    except NoSuchElementException:
        return False

# Logika untuk menangani pola kosong
if not pattern:
    print("Pola kosong, scraping semua data...")
    pattern_check = lambda title: True
else:
    pattern_check = lambda title: raita(pattern, title)

while True:
    # Temukan semua elemen "seo-search-result"
    seo_search_results = driver.find_elements(By.CLASS_NAME, "seo-search-result")

```

```

        for seo_search_result in seo_search_results:
            # Temukan semua elemen "hotel-card" dalam "seo-search-result"
            hotel_card_elements =
seo_search_result.find_elements(By.CLASS_NAME, "hotel-card")

            for hotel_card_element in hotel_card_elements:
                # Temukan tautan <a> di dalam div
                link_element = hotel_card_element.find_element(By.TAG_NAME,
"a")

                title_element =
hotel_card_element.find_element(By.CLASS_NAME, 'title.ellipsis')
                title = title_element.text

                # Periksa apakah title hotel cocok dengan pola
                if pattern_check(title):
                    try:
                        tiket_impresi =
hotel_card_element.find_element(By.CSS_SELECTOR, 'div.tiket-impression')
                        impresi = tiket_impresi.text
                    except NoSuchElementException:
                        impresi = "Tidak tersedia"

                    try:
                        harga_hotel =
hotel_card_element.find_element(By.CLASS_NAME, 'after-price')
                        harga = harga_hotel.text
                    except NoSuchElementException:
                        harga = "Tidak tersedia"

                    # Ambil URL tautan
                    link_url = link_element.get_attribute("href")

                    # Cetak URL tautan
                    print("Link Hotel:", link_url)

                    # Menggulir ke elemen agar terlihat
                    driver.execute_script("arguments[0].scrollIntoView();",
link_element)

                    time.sleep(1) # Tunggu sebentar setelah menggulir

                    # Klik tautan untuk membuka tab baru
                    driver.execute_script("arguments[0].click();",
link_element)

                    # Tunggu hingga tab baru terbuka (Anda bisa menambahkan
waktu tunggu sesuai kebutuhan)
                    time.sleep(3) # Tunggu beberapa detik untuk memastikan
tab baru terbuka

                    # Alihkan ke tab/jendela baru
                    new_window = driver.window_handles[-1]
                    driver.switch_to.window(new_window)

```

```

        # Di sini Anda dapat menambahkan kode untuk mengambil
data dari tab/jendela baru
        try:
            rating_element =
driver.find_element(By.CSS_SELECTOR,
'p.HcPVsG_text.HcPVsG_variant_highEmphasis')
            rating = rating_element.text
        except NoSuchElementException:
            rating = "Tidak tersedia"

        try:
            lokasi_element =
driver.find_element(By.CSS_SELECTOR,
'span.LocationSection_address__V8gdU')
            lokasi = lokasi_element.text
        except NoSuchElementException:
            lokasi = "Tidak tersedia"

        # Tambahkan data ke dalam list
data.append([title, impresi, rating, harga, lokasi])

        # Menutup tab/jendela baru
driver.close()

        # Kembali ke tab/jendela sebelumnya
driver.switch_to.window(driver.window_handles[0])

    if not click_next_button():
        print("Halaman selanjutnya tidak tersedia atau sudah selesai.")
        break # Keluar dari loop jika tombol "Selanjutnya" tidak
ditemukan

# Tutup WebDriver
driver.quit()

```

7. Menampilkan dan Menyimpan Hasil *Crawling*

```

# Nama kolom-kolom
columns = ['Name', 'Impression', 'Rating', 'Price', 'Location']

# Buat DataFrame
df = pd.DataFrame(data, columns=columns)

df.dropna(subset=['Name'], inplace=True)

df.head(100)

df.to_excel('data_penginapan.xlsx')

```