

RESEARCH ARTICLE

Perancangan Program Pencarian Kost Pada Website Mamikost Berbasis *Robotic Process Automation* (Rpa)

Ahmad Izharul Ihsan, Kris Sujatmoko* and Iwan Iwut Tritoasmoro

Fakultas Teknik Elektro, Universitas Telkom, Bandung, 40257, Jawa Barat, Indonesia

* Corresponding author: ismankrn@telkomuniversity.ac.id

Received on 10 August 2023; accepted on 05 September 2023

Abstrak

Setiap tahunnya, Telkom University menerima ribuan mahasiswa baru serta meluluskan ribuan lainnya. Mahasiswa baru akan datang dan mencari tempat tinggal di area kampus, begitu pula dengan mahasiswa yang telah lulus akan mencari tempat tinggal disekitaran kantor tempat mereka bekerja. Namun, kegiatan ini sangat sulit dilakukan dikarenakan para mahasiswa masih asing dengan daerah yang akan mereka datangi yang mengakibatkan pencarian mahasiswa akan sangat terbatas. Dikarenakan sulitnya dalam mencari kost, banyak website yang menyediakan daftar kost yang dapat disewa. Namun, website tersebut hanya menyediakan informasi terkait kost tersebut dan tidak menyediakan rekomendasi kost yang cocok untuk mahasiswa. Hal ini akan memakan waktu yang sangat lama dikarenakan ada puluhan bahkan ratusan kost yang ada disekitaran kampus maupun tempat kerja mahasiswa tersebut. Kegiatan yang berulang tersebut dapat diotomasi sehingga waktu yang diperlukan untuk mencari kost dapat dipercepat. Dengan otomasi juga setiap kost yang telah disortir bisa diberi *scoring* dengan parameter - parameter tertentu. Hal ini akan membuat mahasiswa bisa fokus ke kegiatan lain yang seperti mengumpulkan syarat - syarat masuk kuliah dan kerja.

Key words: Otomasi, Robot, Kost, *Scoring*.

Pendahuluan

Kost adalah bangunan atau Gedung yang terdiri dari kamar dan fasilitas penunjang yang dihuni oleh seseorang untuk jangka waktu tertentu dengan dipungut atau tidak dipungut bayaran [1]. Menyewa sebuah kamar kost memang menjadi sebuah keharusan bagi setiap mahasiswa atau para pekerja kantoran yang beraktivitas jauh dari tempat asalnya. Pencarian kost akan menjadi sangat sulit apalagi mahasiswa tersebut tidak mengenal daerah yang akan ia datangi [2]. Pada saat ini, sudah banyak website penyedia informasi mengenai kostan khusus pada daerah tertentu dan bahkan seluruh Indonesia salah satunya adalah mamikos. Namun, kegiatan mencari kost merupakan kegiatan yang dapat memakan waktu yang lama dikarenakan banyaknya rekomendasi kost yang diberikan oleh website tersebut.

Belum lagi mahasiswa harus menyortir sendiri data dari website pencari kost dikarenakan website tersebut hanya menyediakan informasi mengenai sebuah kost saja dan tidak dilengkapi dengan rekomendasi kost yang paling cocok untuk mahasiswa tersebut. Kegiatan inilah yang akan terus berulang sampai mahasiswa tersebut menemukan kost yang sesuai dengan kriterianya. Kegiatan mencari kost ini dapat diotomasi menggunakan RPA untuk mempercepat proses pencarian. Hal ini disebabkan karena RPA dapat meniru aktivitas manusia didalam

komputer dengan sangat cepat dan memiliki akurasi hingga 100% [3]. Proses otomasi tersebut akan mencakup mencari kost pada website mamikost, membuatkan dokumen yang berisikan informasi dari kost yang telah ter-filter dan juga memberikan *scoring* berdasarkan kecocokan dengan kriteria yang telah ditentukan oleh mahasiswa. Dengan menggunakan proses otomasi, maka kegiatan mencari kost tidak akan memakan waktu.

Tinjauan Pustaka

Mamikos

Mamikos merupakan salah satu website yang menyediakan informasi tentang kost di Indonesia. Saat ini mamikos memiliki lebih dari 2 juta kamar kos yang tersebar lebih dari 140 kota di seluruh Indonesia [4]. Mamikos menyediakan data informasi kost seperti fasilitas kost, foto kamar dan virtual tour, harga kost dan lain sebagainya yang membuat calon penghuni mengetahui detail kost tanpa harus datang berkunjung terlebih dahulu.

Table 1. Daftar Perangkat Lunak yang Digunakan

Nama Aplikasi	Jenis	Keterangan
UiPath	Aplikasi Windows	Perangkat lunak utama pada penelitian ini yang digunakan untuk membuat proses otomasi
Jupyter Notebook	Aplikasi Windows	Digunakan untuk merancang proses <i>scoring</i> yang nantinya akan dijalankan oleh robot &
Google Chrome	Aplikasi Windows	Aplikasi peramban yang digunakan pada penelitian kali ini.
Microsoft Excel	Aplikasi Windows	Digunakan untuk menyimpan data hasil <i>scraping</i> pada website mamikos
Microsoft Outlook	Aplikasi Windows	Digunakan untuk mengirim email kepada user

Robotic Process Automation (RPA)

RPA adalah teknologi perangkat lunak yang mempermudah proses perancangan, pengimplementasian dan pengelolaan robot yang dapat meniru tindakan manusia dengan digital sistem atau perangkat lunak [5]. Robot yang diciptakan dapat melakukan kegiatan yang *repetitive* seperti *login*, memindahkan data dan lain sebagainya. Berbeda dengan manusia, robot yang diciptakan oleh RPA dapat melakukan kegiatan - kegiatan tersebut dengan sangat cepat.

Selain mempercepat proses yang ingin di otomasi, RPA juga memiliki manfaat lain seperti menaikkan produktifitas. Hal ini dikarenakan kegiatan - kegiatan yang tadinya harus kita lakukan secara manual akan otomatis terlaksana oleh robot yang telah dibuat. Misalkan, setiap pagi hari kita harus melakukan absensi berbentuk *form*, maka *form* ini akan otomatis diisi oleh robot setiap hari pada pagi harinya. Lalu, kita bisa melakukan hal lain yang tidak *repetitive*.

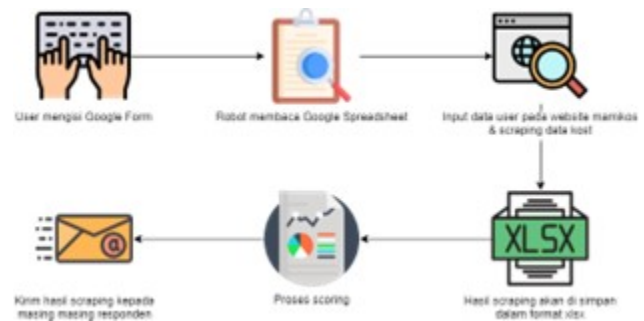
Rule-Based System

Rule-based system adalah sistem yang menerapkan aturan buatan manusia untuk menyimpan, menyortir, dan memanipulasi data [6]. *Rule-based system* akan mengeksekusi data yang masuk sesuai dengan *rules* yang telah di tentukan di awal. Sistem pun hanya akan menerapkan *rules* tersebut untuk semua data yang ada. Jika ada data yang tidak bisa dieksekusi menggunakan *rules* yang ada, maka sistem akan berhenti atau *error*. *Rule-based system* juga tidak akan belajar dari kesalahan, sistem akan terus mengulangi kesalahan yang sama terus menerus. Maka dari itu, untuk membuat sistem menjadi lebih pintar ditambahkan *rules* baru secara manual.

Pada penelitian ini, *rule-based system* digunakan pada proses *scoring*. Sistem akan diberikan *rules* yang akan menghitung skor berdasarkan hasil *scraping* yang telah dilakukan. Dengan menggunakan *rule-based system*, semua data kos hasil *scraping* akan diperlakukan sama sehingga akan didapatkan skor yang adil.

Penelitian Kuantitatif

Penelitian Kuantitatif menurut Cresswell (1994), "Penelitian kuantitatif adalah metode-metode untuk menguji teori-teori tertentu dengan cara meneliti hubungan antar variabel. Biasanya, variabel tersebut diukur dengan instrumen penelitian sehingga data yang terdiri dari angka-angka dapat dianalisis berdasarkan prosedur statistik." [7]. Pada

**Gambar 1.** Gambaran Umum Sistem.**Table 2.** Kebutuhan Perangkat Lunak

Spesifikasi	Minimum	Perangkat Peneliti
CPU Cores	2 × 1.8GHz32-bit (x86)	6 × 3.0GHz64-bit (x64)
RAM	4 GB	16 GB
Disk Space	3.5 GB	Mencukupi

penelitian kuantitatif terdapat 2 jenis penelitian, yaitu penelitian survei dan penelitian eksperimen. Pada penelitian ini jenis penelitian survei yang dimana akan menggunakan kuisioner sebagai alat pengumpulan data. Kemudian, skala yang digunakan pada kuisioner tersebut adalah skala *Likert*.

Skala *Likert* adalah suatu skala psikometrik yang umum digunakan dalam angket dan merupakan skala yang paling banyak digunakan dalam riset berupa survei. Responden akan mengisi kuisioner sesuai dengan pendapat mereka terhadap suatu pertanyaan.

Metodologi Penelitian

Gambaran Umum Sistem

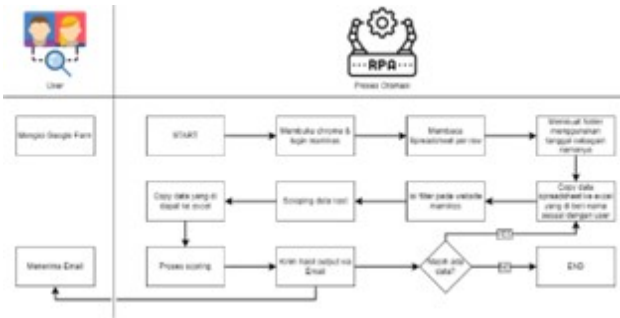
Berikut ini adalah diagram untuk gambaran umum dari perancangan yang dilakukan. Proses akan dimulai pada sisi user, diperlukan untuk mengisi *form* berupa data diri mereka dan juga kriteria kos yang ingin mereka cari. Setelah itu robot akan membaca *google spreadsheet* yang berisikan informasi dari form yang kemudian akan diisikan pada website mamikos sebagai filter kos. Kemudian setelah muncul kos dengan kriteria tersebut robot akan mulai melakukan data *scraping*. Hasil data *scraping* akan disimpan dalam format *.xls yang nantinya akan diproses kembali untuk proses *scoring*. Setelah dilakukan proses *scoring*, maka data tersebut akan dikirim secara otomatis oleh robot kepada *user*.

Analisis Kebutuhan Sistem

Kebutuhan Data

Data yang dibutuhkan merupakan data dari *user* yang telah mengisi *google form* yang telah disediakan. Data tersebut nantinya akan diproses oleh robot UiPath. Berikut adalah data yang harus diisi oleh *user*:

1. Nama
2. Email
3. Lokasi kos yang diinginkan
4. Alamat yang dituju
5. Tipe kos
6. Waktu bayar sewa
7. Harga maksimum



Gambar 2. Alur Perancangan Program.

Table 3. Daftar Activity yang Digunakan Untuk Web Scaping

No	Jenis Data	Activity yang Digunakan	Hasil Scraping Setelah Preprocessing Data
1	Nama kost	Extract Data Table	Kost Blok M Tipe Standart Kebayoran Baru Jakarta Selatan
2	Harga kost	Get Attribute	Rp1.750.000/bulan
3	Link mamiko	Get Attribute	https://mamikos.com/room/kost-kota-jakarta-selatan-kost-putra-eksklusif-kost-blok-m-tipe-standart-kebayoran-baru-jakarta-selatan?redirection_source=list%20kos%20result
4	Alamat kost	Get Attribute	Jl Mesjid darusalam raya no. 27 A, Blok A Rt 003 Rw 005 Kebayoran Baru jakarta Selatan 12140 .
5	Link google maps	Get Text	https://maps.google.com/maps?z=7q=-6.257763,106.795165
6	Fasilitas kamar	Extract Data Table	AC,Kasur,Lemari Baju,Bantal,Guling,Jendela,
7	Fasilitas bersama	Extract Data Table	WiFi,R. Cuci,Kulkas,R. Tamu,Mesin Cuci,Penjaga Kos,R. Makan,R. Jemur,Dispenser,Dapur,
8	Jarak ke kantor	Get Text	2,9 km

8. Fasilitas kamar
9. Fasilitas umum
10. Faktor yang diutamakan

Kebutuhan Perangkat Lunak

Berikut adalah daftar perangkat lunak yang dibutuhkan pada penelitian ini: 1

Kebutuhan Perangkat Lunak

Untuk menjalankan program yang dibutuhkan untuk perancangan pada UiPath dibutuhkan spesifikasi sebagai berikut: 2

Table 4. Hasil Scoring

No. Kost	Nama Kost	Score Harga	Score Kamar	Score Umum	Score Jarak
0	Kost Blok M Tipe Standart Kebayoran Baru Jakarta Selatan	12.5	6	10	2.9
1	Kost Blok M Tipe Superior Kebayoran Baru Jakarta Selatan	11.0	9	10	3.8
2	Kost Blok M Tipe Standart 2 Kebayoran Baru Jakarta Selatan	12.0	8	10	2.9
	Kost Blok M Tipe Deluxe Kebayoran Baru	5.0	8	10	

Kebutuhan User

Pada penelitian kali ini user dibutuhkan untuk:

1. Mengisi google form yang sudah disediakan.
2. Memiliki alamat email yang aktif untuk menerima email berupa data rekomendasi kos yang akan dikirimkan oleh robot.
3. Memiliki aplikasi pembaca PDF jenis apapun untuk membaca file yang dikirimkan melalui email Gambar 2 merupakan proses bisnis yang berjalan pada penelitian kali ini. Proses dimulai dengan adanya input dari user pada google form. Kemudian pada saat robot dijalankan, robot akan membaca spreadsheet yang dihasilkan oleh google form. Setelah itu robot akan membuat folder dengan nama folder tanggal pada saat robot dijalankan. Untuk setiap row yang ada di spreadsheet akan di extract ke dalam bentuk excel pada folder yang sudah dibuat sebelumnya.

Kemudian robot akan mulai memasukan input user yang berupa kriteria kos yang dicari kedalam filter website mamikos. Setelah selesai di input, robot akan mulai melakukan data dengan sejumlah kos yang ditentukan sebelumnya oleh robot. Hasil dari scraping akan disimpan sesuai dengan excel masing masing user. Setelah semua informasi berhasil didapatkan, robot akan menjalankan program python untuk melakukan scoring terhadap kos yang berada pada excel yang hasil outputnya akan berupa file berformat .pdf . Kemudian, setelah selesai dilakukan scoring robot akan mengirimi hasil output scoring kepada user melalui e-mail.



Gambar 3. Daftar *Dependency Package* yang digunakan.

Pemilihan UiPath *Package*

Dependency Package merupakan *library* dari *Activity* yang dapat kita lakukan di UiPath. Pada penelitian kali ini, *package* yang digunakan adalah *package* bawaan dari UiPath, *Excel Activities*, *Google Workspace* dan *Mail Activities* 3.

Hasil dan Pembahasan

Implementasi Sistem

Setelah dilakukannya perancangan, gambar 4 merupakan *workflow* keseluruhan dari program yang dibuat pada aplikasi UiPath. Pada saat program dijalankan, program akan membuka *Google Chrome* dan *login* pada website mamikos. Setelah berhasil *login*, program akan membaca *Spreadsheets* dan apabila ditemukan *input* dari *user* maka program akan membuat folder dengan nama sesuai dengan tanggal dijalanannya program. Setelah itu, *user* akan dibuatkan *excel* khusus yang nantinya akan diisi dengan hasil web *scraping*. Kemudian program akan melakukan *input filter* kost yang diinginkan oleh *user* ke website mamikos. *Filter* yang akan diisikan adalah lokasi kost, jenis kost, harga kost, fasilitas kamar dan fasilitas umum. Setelah daftar kost muncul maka program akan mulai melakukan web *scraping* untuk nama dan juga harga kost. Setelah nama kost berhasil didapatkan, program akan masuk ke halaman kost dan melakukan web *scraping* untuk URL kost tersebut, alamat kost, link *Google Maps* dari kost, fasilitas kamar dan fasilitas umum. Berikutnya program akan membuka link *Google Maps* yang telah di-*scraping* sebelumnya untuk mencari jarak kost dengan kantor atau tujuan *user*. Setelah mendapatkan jarak, program akan *looping* untuk *scraping* data kost lainnya. Pada saat *looping* berhenti, *excel* yang berisi hasil web *scraping* akan disalin ke folder sementara untuk dilakukan proses *scoring* karena salinan tersebutlah yang akan dibaca oleh program. Proses *scoring* akan mengeluarkan dokumen baru berformat *.pdf yang kemudian akan disalin kembali ke folder untuk tanggal tersebut. Langkah terakhir dari program adalah mengirimkan file pdf tersebut ke *user* yang bersangkutan. Jika robot tidak mendeteksi adanya *input* lain dari *user* pada *Spreadsheets* maka program akan berhenti.

Implementasi Web *Scraping*

Untuk melakukan web *scraping* dibutuhkan *activity* yang sesuai untuk masing masing data. Dapat dilihat *workflow* pada gambar 5 yang merupakan proses web *scraping* yang dilakukan oleh robot UiPath. Selain melakukan *scraping*, program juga melakukan *preprocessing* data agar nantinya data yang *discraping* dapat dengan mudah disalin kedalam *excel* dan juga mempermudah proses *scoring*.

Setiap data membutuhkan *activity* yang berbeda sesuai dengan jenisnya. Berikut adalah daftar *activity* yang digunakan untuk masing

Table 5. Hasil untuk Masing - Masing Pembobotan

No. Kost	Nama Kost	Pembo botan Harga	Pembo botan Fasilitas Samar	Pembo botan Fasilitas Smum	Pembo botan Jarak
0	Kost Blok M Tipe Standart Kebayoran Baru Jakarta Selatan	10.06	6.16	8.56	6.52
1	Kost Blok M Tipe Superior Kebayoran Baru Jakarta Selatan	9.22	8.02	8.62	6.34
2	Kost Blok M Tipe Standart 2 Kebayoran Baru Jakarta Selatan	9.91	7.51	8.71	6.97
3	Kost Blok M Tipe Deluxe Kebayoran Baru Jakarta Selatan	5.01	6.81	8.01	4.87
4	Kost Home Sweet Home Kost 79A Tipe B Mampang Prapatan Jakarta Selatan	9.71	8.51	8.51	4.77

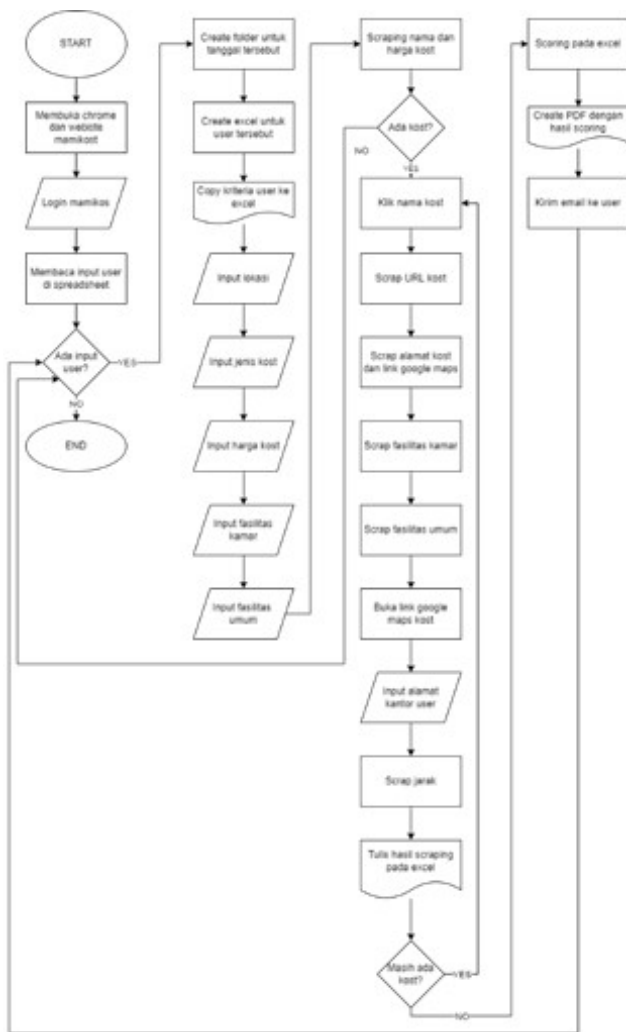
- masing data serta contoh data setelah dilakukan *preprocessing* data: 3 Hasil dari *scraping* tersebut akan disimpan dalam format *.xlsx pada *sheet* "Rekomendasi kost". Tampilan dari *.xlsx dapat dilihat pada gambar 6.

Implementasi *Scoring*

Setelah web *scraping* selesai dilakukan, data akan diproses menggunakan *python* untuk proses *scoring*. Berikut adalah formula yang digunakan:

1. Skor Harga

```
#menampilkan harga yang di input user
in_harga = df1['Harga']
in_harga = int(in_harga)
in_harga
#menampilkan harga hasil scraping
scrap_harga = df2['Harga Kost'].str.replace('Rp—/bulan;', '')
scrap_harga = scrap_harga.str.replace('.',',')
scrap_harga = scrap_harga.astype(int)
scrap_harga
#menghitung skor harga
selisih = in_harga - scrap_harga
score_harga = selisih / 100000
```



Gambar 4. Workflow Lengkap Sistem

```

score_harga
2. Skor Fasilitas Kamar
score kamar = df2['Fasilitas
Kamar'].str.count(',')
score kamar.astype (float)

Skor Fasilitas Umum
score_umum = df2['Fasilitas
Bersama'].str.count(',')
score umum. astype (float)
3. Skor Jarak jarak = df2['Jarak Ke Kantor']
jarak = jarak.map (lambda x : x .rstrip('km'))
jarak = jarak.str.replace ('+', '.')
score_jarak = jarak.astype (float)
score jarak
4. Total Skor
score = score harga + score kamar + score umum score.jarak
Setelah dilakukan implementasi untuk menentukan komponen
skor pada tabel 4, berikut adalah hasil dari proses scoring awal
yang dilakukan: 4
Setelah didapatkan komponen awal untuk scoring, akan dilaku-
kan perhitungan lanjutan menggunakan pembobotan sesuai

```

Table 6. Hasil Pengujian Alpha

No	Skenario	Kesimpulan
1	Initiate program	Diterima
2	Pembuatan folder dan excel	Diterima
3	Preprocessing data hasil <i>scraping</i> sebelum ditulis kedalam excel	Diterima
4	Pemberian skor dan membuat grafik skor	Diterima
5	Mengirim pdf via email ke pengguna	Diterima

Table 7. Pengujian Data Alpha

No	Data yang diuji	Kesimpulan
1	Pengisian kriteria pada website mamikos	Diterima
2	Hasil web <i>scraping</i>	Diterima
3	Hasil proses <i>scoring</i>	Diterima

dengan faktor yang diutamakan oleh *user*. Faktor tersebut akan diberikan bobot 70% sedangkan yang lain akan diberi bobot masing - masing 10%. Berikut adalah formula yang digunakan untuk pembobotan:

- a. **Selisih Harga**
 $score = (score_harga \cdot 0.7) + (score_kamar \cdot 0.1) + (score_umum \cdot 0.1) - (\bar{s} \text{ core_jarak} \cdot 0.1)$
 - b. **Fasilitas Kamar**
 $Score = (score_harga \cdot 0.1) + (score_kamar \cdot 0.7) + (score_umum \cdot 0.1) - (score_jarak \cdot 0.1)$
 - c. **Fasilitas Bersama**
 $Score = (score_harga \cdot 0.1) + (score_kamar \cdot 0.1) + (score_umum \cdot 0.7) - (score_jarak \cdot 0.1)$
- Jarak**
 $Score = ((score \text{ harga} \cdot 0.1) + (score \text{ kamar} \cdot 0.1) + (score \text{ umu} \cdot m \cdot 0.1)) \cdot 3 - (score_jarak \cdot 0.7)$

Berikut adalah skor akhir yang akan didapatkan *user* untuk masing - masing botbot yang tersedia: 10

Pengujian Alpha

Pengujian alpha merupakan pengujian yang dilakukan untuk mengetahui apakah rangkaian program berjalan dengan sebagaimana mestinya. Pengujian ini berfokus pada tingkat keberhasilan pengambilan data yang dilakukan oleh web *scraping*. Tabel 6 adalah daftar skenario pengujian alpha yang dilakukan. Pada tabel 7, dapat dilihat bahwa setiap skenario yang harus dikerjakan atau dilakukan oleh robot dapat berjalan dengan lancar tanpa adanya *error*. Kemudian setelah pengujian alpha dilakukan, proses berikutnya adalah pengujian data. Tabel 7 adalah pengujian data alpha yang dilakukan.

Dapat dilihat pada tabel 7 bahwa setiap data telah diproses dengan benar oleh robot UiPath. Pada dua pengujian tersebut dapat dilihat bahwa robot dapat berjalan dengan lancar dan juga memproses data

Table 8. Rating Scale

Kategori	Persentase
Sangat Puas	81 – 100%
Puas	61 – 80%
Cukup Puas	41 – 60%
Kurang Puas	21 – 40%
Sangat Tidak Puas	0 – 20%

dengan tepat. Maka dari itu dapat disimpulkan bahwa robot berhasil dalam mencari dan juga memberikan *scoring* untuk setiap kost.

Pengujian Mean Opinion Score

Kuisisioner menggunakan survei *likert* untuk menentukan tingkat kepuasan dari setiap pengguna. Pada kuisisioner tersebut, *user* akan diberikan hasil *output* dari robot UiPath yang telah dibuat. Dari gambar 7, responden dapat memilih skor 1 sampai dengan 5. 1 merupakan skor paling rendah yang dapat pilih apabila responden merasa sangat tidak setuju dengan pertanyaan yang diajukan. Sedangkan skor tertinggi adalah 5 yang menandakan responden sangat setuju dengan pertanyaan yang diajukan. Setelah semua responden mengisi kuisisioner, maka akan dilakukan perhitungan menggunakan rumus:

$$p = \frac{f}{n} \times 100\%$$

Keterangan:

p = Nilai akhir

f = Jumlah skor responden

n = Jumlah skor maksimal

Rumus tersebut akan digunakan untuk menghitung tingkat kepuasan responden terhadap pertanyaan yang diajukan. Setelah dihitung, setiap pertanyaan akan dimasukkan kedalam kategori berupa rating *scale* berdasarkan hasil perhitungan. Tabel 7 label rating *scale* yang digunakan pada pengujian beta. Menurut 34 *responden*, nilai yang didapat untuk masing - masing pertanyaan yang diberikan adalah sebagai berikut: 9. Dari tabel 9, rata - rata persentase yang didapat adalah 84,12%. Dengan begitu maka hasil dari pengujian sistem ini masuk kedalam kategori "Sangat Puas".

Perbandingan Waktu

Tabel 10 merupakan perbandingan waktu antara proses manual dengan proses otomatisasi: 10 Berdasarkan tabel 7, dapat disimpulkan bahwa proses manual dapat dipercepat dengan menggunakan sistem yang dirancang pada penelitian kali ini. Hal ini dapat menghemat waktu *user*

sehingga *user* dapat fokus pada registrasi kuliah atau onboarding di kantor mereka.

Kesimpulan

Setelah sistem dirancang dan juga dilakukan pengujian maka dapat ditarik kesimpulan bahwa:

1. Sistem pencarian kost pada website mamikos berbasis *Robotic Process Automation* (RPA) telah berhasil dibuat dan dapat mengerjakan *scenario* yang sama dengan proses manual yang ada.
2. Sistem dapat melakukan web *scraping* dengan tepat dan juga dapat memberikan skor terhadap kost yang telah di-*scraping* sesuai dengan hasil dari pengujian alpha yang telah dilakukan
3. Dari hasil pengujian beta dapat disimpulkan bahwa sistem yang dibuat mendapatkan kategori "Sangat Puas" dari 34 responden yang ada.
4. Waktu yang didapat pada proses manual adalah 25 detik untuk membuka website dan *login* serta 1 menit 51 detik untuk *scraping* data untuk 1 kost. Sedangkan waktu yang didapat pada saat menggunakan sistem adalah 12 detik untuk membuka website dan *login* serta 35 detik untuk *scraping* data untuk 1 kost. Dengan demikian dapat disimpulkan bahwa waktu yang dipangkas setelah menggunakan sistem adalah 13 detik untuk membuka website dan *login* serta 1 menit 16 detik untuk *scraping* data untuk 1 kost.

Daftar Pustaka

1. Ikhsan A. Desain Rumah Kos Berdasarkan Persepsi Penghuni. 2019;1(1):47.
2. Abidin B. SISTEM INFORMASI RUMAH KOST ONLINE. 2012;1(1).
3. Maulid LNMPRAH. Lappybot: Aplikasi Chatbot Untuk Informasi Pemilihan Laptop Menggunakan Metode Natural Language Processing (NLP). 2020;1(1).
4. Mamikos;. [Accessed 28 July 2022]. Available from: <https://mamikos.com/>.
5. What is robotic process automation?;. [Accessed 25 July 2022]. Available from: <https://www.uipath.com/rpa/robotic-process-automation>.
6. Doe J. Profesional AI; 2020. [Accessed 22 April 2020]. Available from: <https://www.professional-ai.com/rule-based-systems.html>.
7. Laily IN. Pengertian Penelitian Kuantitatif, Karakteristik dan Jenisnya; 2022. [Accessed August 2022]. Available from: <https://katadata.co.id/iftitah/ekonopedia/6295749c7fdd7/pengertian-penelitian-kuantitatif-karakteristik-dan-jenisnya>.

Table 9. Hasil Perhitungan Kuisisioner

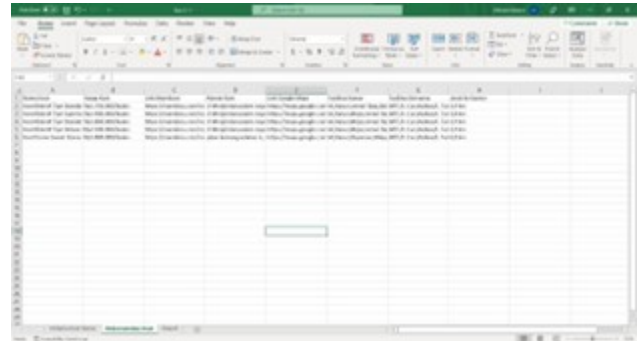
No	Pertanyaan	Skor	Persentase	Rating Scale
1	Apakah form yang dipakai untuk proses input data mudah untuk dipahami dan diisi?	146	85, 88%	Sangat Puas
2	Apakah output dari program dapat mudah dipahami?	138	81, 18%	Sangat Puas
3	Menurut anda, apakah daftar kos yang ada pada output tersebut sesuai dengan input diawal?	144	84, 71%	Sangat Puas
4	Menurut anda, apakah grafik yang disediakan dapat membantu dalam membaca hasil?	139	81, 76%	Sangat Puas
5	Menurut anda, apakah skor yang ditampilkan dapat merepresentasikan kualitas dari kos?	143	84,12%	Sangat Puas
6	Menurut anda, apakah program ini membantu dalam menentukan kost mana yang akan disewa?	148	87,06%	Sangat Puas
7	Bagaimana menurut anda efisiensi waktu yang dipangkas oleh program ini?	143	84,12%	Sangat Puas

Table 10. Perbandingan Waktu Proses Manual dengan Proses otomasi

Skenario	Metode yang digunakan	
	Manual	Robot UiPath
Membuka website dan <i>login</i>	00 m : 25 s	00 m : 12 s
<i>Scraping</i> data untuk 1 kost	01 m : 51 s	00 m : 35 s



Gambar 5. Implementasi Web Scraping



Gambar 6. Skala Jawaban Pada Kuisiонер



Gambar 7. Skala Jawaban Pada Kuisiонер