

**PENGEMBANGAN APLIKASI ALAT BANTU ANALISIS  
KORELASI DATA FAKTUAL DAN GOOGLE TRENDS  
UNTUK *MONITORING / SURVEILLANCE***



Disusun Oleh:

N a m a : Muhamad Fajar Setianto  
NIM : 15523156

**PROGRAM STUDI INFORMATIKA – PROGRAM SARJANA  
FAKULTAS TEKNOLOGI INDUSTRI  
UNIVERSITAS ISLAM INDONESIA  
2020**

HALAMAN PENGESAHAN DOSEN PEMBIMBING

**PENGEMBANGAN APLIKASI ALAT BANTU ANALISIS  
KORELASI DATA FAKTUAL DAN GOOGLE TRENDS  
UNTUK *MONITORING / SURVEILLANCE***

**TUGAS AKHIR**



N a m a : Muhamad Fajar Setianto  
NIM : 15523156

الجمعة الاستد الاندو

Yogyakarta, 3 Agustus 2020

Pembimbing,

( Dhomas Hatta Fudholi, S.T., M.Eng., Ph.D. )

## HALAMAN PENGESAHAN DOSEN PENGUJI

**PENGEMBANGAN APLIKASI ALAT BANTU ANALISIS  
KORELASI DATA FAKTUAL DAN GOOGLE TRENDS  
UNTUK *MONITORING / SURVEILLANCE***

## TUGAS AKHIR

Telah dipertahankan di depan sidang penguji sebagai salah satu syarat untuk memperoleh gelar Sarjana Komputer dari Program Studi Informatika - Program Sarjana di Fakultas Teknologi Industri Universitas Islam Indonesia

Yogyakarta, 12 Agustus 2020

Tim Penguji

Dhomas Hatta Fudholi, S.T., M.Eng., Ph.D.

**Anggota 1**

Ahmad Fathan Hidayatullah, S.T., M.Cs.

**Anggota 2**

Kholid Haryono, S.T., M.Kom.

Mengetahui,

Ketua Program Studi Informatika – Program Sarjana

Fakultas Teknologi Industri  
Universitas Islam Indonesia



( Dr. Radeh Teduh Dirgahayu, S.T., M.Sc. )

**HALAMAN PERNYATAAN KEASLIAN TUGAS AKHIR**

Yang bertanda tangan di bawah ini:

Nama : Muhamad Fajar Setianto

NIM : 15523156

Tugas akhir dengan judul:

**PENGEMBANGAN APLIKASI ALAT BANTU ANALISIS  
KORELASI DATA FAKTUAL DAN GOOGLE TRENDS  
UNTUK *MONITORING / SURVEILLANCE***

Menyatakan bahwa seluruh komponen dan isi dalam tugas akhir ini adalah hasil karya saya sendiri. Apabila dikemudian hari terbukti ada beberapa bagian dari karya ini adalah bukan hasil karya sendiri, tugas akhir yang diajukan sebagai hasil karya sendiri ini siap ditarik kembali dan siap menanggung resiko dan konsekuensi apapun.

Demikian surat pernyataan ini dibuat, semoga dapat dipergunakan sebagaimana mestinya.

Yogyakarta, 3 Agustus 2020



( Muhamad Fajar Setianto )

## HALAMAN PERSEMBAHAN

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

*Alhamdulillah Robbil'alamin* Segala puji syukur bagi Allah *Subhanahu wa Ta'ala* atas segala rahmat dan hidayah-Nya yang selalu menunjukkan jalan yang benar, dan keberkahan yang tak terhingga. Atas berkat rahmat dan ridho Allah lah penulis dapat menyelesaikan skripsi ini dengan baik.

Shalawat dan salam penulis sanjungkan kepada Nabi Muhammad *Shalallahu'alaihi Wasallam*, keluarga, dan sahabatnya. Semoga kelak kita semua bisa mendapatkan syafaat Beliau di *yaumul akhir*.

Amin

Karya sederhana ini saya persembahkan untuk:

**Alm Bapak tersayang Ir.Handomo dan Mama ku tercinta Leli Sutini**

Terimakasih atas perhatian, kasih sayang, kesabaran, keikhlasan, nasihat, dan pengorbanan yang selama ini sudah diberikan. Karya sederhana ini tidak akan sebanding dengan semua pengorbanan alm bapak dan mama berikan selama ini. Semoga alm bapak mendapatkan tempat terbaik disisi-Nya dan mama selalu diberikan kesehatan, kebahagiaan, dan selalu dalam lindungan Allah *Subhanahu Wa Ta'ala*.

**HALAMAN MOTO**

***La Ilaha Illallah***  
(Tiada Tuhan Selain Allah)

*“Dan mintalah pertolongan dengan sabar dan shalat.”*

**(Q.S Al-Baqarah : 45)**

*“Ada empat hal yang tidak berguna, yaitu ilmu tanpa amal, kekayaan tanpa diiringi dengan berinfak di jalan Allah, kesalehan hanya untuk dipamerkan yaitu didorong oleh motivasi duniawi, dan umur panjang yang dihabiskan tanpa berbuat amal kebajikan”*

**(Ustman bin Affan)**

*“Kerjakanlah urusan duniawimu seakan-akan kamu hidup selamanya dan laksanakan urusan akhiratmu seakan-akan kamu akan mati besok”*

**(HR.Ibnu Asakir)**

## KATA PENGANTAR

### *Assalamu'alaikum Warrahamatullahi Wabarakatuh*

Puji syukur penulis panjatkan atas kehadirat Allah SWT karena berkat rahmat taufik dan karunia Nya sehingga penulis dapat menyelesaikan laporan tugas akhir ini. Shalawat serta salam semoga senantiasa terlimpah curahkan kepada junjungan nabi besar kita Nabi Muhammad SAW, kepada keluarganya, para sahabatnya, hingga kepada umatnya hingga akhir zaman, *Aamiin*.

Tugas akhir ini merupakan salah satu persyaratan yang harus dipenuhi dalam rangka menyelesaikan pendidikan pada jenjang Strata 1 di Program Studi Informatika – Program Sarjana, Fakultas Teknologi Industri, Universitas Islam Indonesia.

Oleh karena itu, pada kesempatan ini penulis ingin menyampaikan rasa terima kasih kepada:

1. Alm Bapak dan Mama tercinta, yang tidak henti-hentinya memberikan doa, cinta, kasih sayang, dukungan, dan pengorbanan yang telah diberikan sehingga tugas akhir ini dapat terselesaikan.
2. Bapak Dr. Raden Teduh Dirgahayu, S.T., M.Sc. selaku ketua Program Studi Informatika - Program Sarjana.
3. Bapak DThomas Hatta Fudholi, S.T., M.Eng., Ph.D. selaku dosen pembimbing tugas akhir yang telah banyak meluangkan waktu, tenaga dan pikiran dalam membimbing saya menyelesaikan tugas akhir ini.
4. Seluruh Dosen Jurusan Program Studi Informatika – Program Sarjana Universitas Islam Indonesia yang telah memberi ilmu sehingga penulis mampu melangkah hingga pada tahap penyelesaian tugas akhir.
5. Anisa Ariani, terimakasih telah banyak membantu dan meluangkan waktunya. Semoga harapan yang kita rangkai bersama dapat digapai satu persatu.
6. Teman-teman sekaligus rekan kerja CV.Codonesia, terimakasih untuk kerja keras dan kerjasama selama ini.
7. Semua pihak yang telah membantu penulis dalam penyelesaian tugas akhir yang tidak dapat penulis sebutkan satu persatu. Semoga Allah SWT memberikan balasan yang berlipat ganda kepada semuanya.

Penulis menyadari bahwa dalam penyusunan laporan masih terdapat banyak kekurangan. Demi perbaikan selanjutnya, maka dari itu, penulis masih membutuhkan kritik dan saran yang

sifatnya membangun untuk hasil yang lebih baik kedepannya. Penulis berharap laporan ini dapat bermanfaat bagi kita semua.

***Wassalamu'alaikum Warahmatullahi Wabarakatuh***

Yogyakarta, 3 Agustus 2020

Muhammad Fajar Setianto



## SARI

Pada penelitian ini penulis melakukan pengembangan aplikasi analisis korelasi data faktual dan Google Trends untuk *monitoring / surveillance* menggunakan bahasa pemrograman PHP dengan memanfaatkan protocol HTTP *request*. Pengambilan data dari Google Trends dilakukan dengan menerapkan metode *scraping* dengan teknik API. Pada proses pembuatan terdapat beberapa masalah dikarenakan url yang digunakan bukan merupakan API yang diperuntukan untuk umum yaitu, data *time series* yang diskalakan, data harian yang terbatas dan kemungkinan *server* mengembalikan respon 429. Beberapa cara penulis lakukan untuk mengatasi masalah tersebut seperti dengan menerapkan sistem antrian dan proxy rotator. Tahap normalisasi data dilakukan terhadap data yang diperoleh dari Google Trends sebelum proses perhitungan korelasi untuk mengatasi keterbatasan Google Trends dalam menyediakan data. Tahap terakhir dilakukan proses perhitungan korelasi menggunakan korelasi pearson.

Kata kunci: korelasi, google trends, *scraping*, proxy rotator.

## GLOSARIUM

Framework	Kerangka kerja
Monitoring	Suatu kegiatan yang dilakukan untuk mengamati suatu kejadian secara terus menerus
Request	Suatu proses yang terjadi saat komputer menghubungi komputer lainnya\
Parsing	Suatu kegiatan untuk mendapatkan data yang dibutuhkan dari data lainnya
Private	Bersifat pribadi
Scraping	Suatu proses pengambilan data dari suatu website
Surveillance	Suatu kegiatan yang dilakukan untuk mengamati suatu kejadian secara terus menerus diiringi dengan proses pengumpulan data
Server	Sebuah komputer yang menyediakan dan melayani permintaan dari komputer lainnya.
Too Many Request	Suatu respon yang diberikan oleh server dalam hal keamanan dikarekan banyaknya request yang dikirimkan dalam waktu yang singkat
Web	Sebuah alamat yang dapat diakses melalui jaringan internet.

## DAFTAR ISI

HALAMAN JUDUL .....	i
HALAMAN PENGESAHAN DOSEN PEMBIMBING .....	ii
HALAMAN PENGESAHAN DOSEN PENGUJI .....	iii
HALAMAN PERNYATAAN KEASLIAN TUGAS AKHIR.....	iv
HALAMAN PERSEMBAHAN .....	v
HALAMAN MOTO .....	vi
KATA PENGANTAR.....	vii
SARI .....	ix
GLOSARIUM .....	x
DAFTAR ISI .....	xi
DAFTAR TABEL .....	xiii
DAFTAR GAMBAR.....	xiv
<b>BAB I PENDAHULUAN .....</b>	<b>1</b>
1.1 Latar Belakang .....	1
1.2 Rumusan Masalah .....	3
1.3 Batasan Masalah .....	3
1.4 Batasan Masalah .....	3
1.5 Tujuan Penelitian .....	4
1.6 Manfaat Penelitian .....	4
1.7 Metode Penelitian .....	4
1.8 Sistematika Penelitian .....	5
<b>BAB II LANDASAN TEORI .....</b>	<b>6</b>
2.1 Google Trends .....	6
2.2 <i>Web Scraping</i> .....	7
2.2.1 <i>Metode Web Scraping</i> .....	8
2.2.2 <i>Legalitas Web Scraping</i> .....	9
2.2.3 <i>Pemanfaatan Web Scraping</i> .....	10
2.3 Time Series .....	11
2.4 <i>Literature Review</i> .....	11
<b>BAB III METODOLOGI PENELITIAN .....</b>	<b>13</b>
3.1 Identifikasi Masalah .....	14
3.2 Analisis Kebutuhan .....	14
3.2.1 Analisis Kebutuhan Masukan.....	14
3.2.2 Analisis Kebutuhan Proses .....	15
3.2.3 Analisis Kebutuhan Keluaran.....	16
3.3 Perancangan Aplikasi.....	16
3.3.1 <i>Use Case Diagram</i> .....	17
3.3.2 <i>Activity Diagram</i> .....	17
3.3.3 Perancangan Antarmuka.....	19
3.4 Perancangan Pengujian .....	20
3.4.1 Pengujian Fungsionalitas (Black Box) .....	20
3.4.2 Pengujian Usabilitas .....	23
<b>BAB IV IMPLEMENTASI.....</b>	<b>25</b>
4.1 Data Faktual .....	25
4.1.1 Template Data Faktual .....	25
4.1.2 <i>Import</i> dan Validasi Data Faktual .....	26
4.2 Implementasi Scraping.....	28

4.2.1	Analisis Url dan <i>User Interface</i> Google Trends.....	28
4.2.2	Penentuan Teknik Scraping.....	31
4.2.3	Proses Scraping .....	34
4.2.4	Fitur Pendukung .....	43
4.3	Perhitungan Korelasi.....	44
4.4	Implementasi Website.....	45
4.4.1	<i>Model</i> .....	46
4.4.2	<i>View</i> .....	47
4.4.3	<i>Controller</i> .....	52
4.5	Pengujian.....	54
4.5.1	Pengujian Fungsionalitas.....	54
4.5.2	Pengujian Usabilitas .....	61
BAB V KESIMPULAN DAN SARAN .....		65
5.1	Kesimpulan .....	65
5.2	Saran.....	65
DAFTAR PUSTAKA.....		66
LAMPIRAN .....		68

**DAFTAR TABEL**

Tabel 3.1 Skenario Pengujian Menggunakan Black Box .....	20
Tabel 3.2 Rancangan Pertanyaan Kuisisioner .....	23
Tabel 3.3 Kategori Jawaban Responden .....	24
Tabel 3.4 Kategori kelayakan berdasarkan skor usabilitas .....	24
Tabel 4.1 Hasil pengujian fungsionalitas .....	54
Tabel 4.2 Hasil kuisisioner .....	62
Tabel 4.3 Hasil pengujian untuk setiap pertanyaan .....	63

## DAFTAR GAMBAR

Gambar 2.1 Antarmuka Google Trends .....	7
Gambar 3.1 Alur metode penelitian.....	13
Gambar 3.2 <i>Use Case Diagram</i> .....	17
Gambar 3.3 <i>Activity diagram</i> .....	18
Gambar 3.4 Rancangan halaman home .....	19
Gambar 3.5 Rancangan halaman result .....	20
Gambar 4.1 Contoh <i>template</i> data faktual .....	26
Gambar 4.2 Kode program proses <i>import</i> .....	27
Gambar 4.3 <i>Array</i> hasil import dan validasi data faktual .....	28
Gambar 4.4 Tangkapan layar tampilan <i>homepage</i> Google Trends.....	29
Gambar 4.5 Tangkapan layar halaman hasil penelusuran Google Trends.....	30
Gambar 4.6 Http request terhadap url halaman hasil penelusuran Google Trends .....	32
Gambar 4.7 Tangkapan layar analisis lalu lintas data.....	33
Gambar 4.8 Fungsi konstruktor pada kelas GoogleTrend .....	34
Gambar 4.9 Fungsi <i>_getData</i> pada kelas GoogleTrend .....	35
Gambar 4.10 Parameter untuk url “https://trends.google.com/trends/api/explore” .....	35
Gambar 4.11 Kode program fungsi <i>_prepare</i> kelas GoogleTrend .....	36
Gambar 4.12 Contoh respon dari url “https://trends.google.com/trends/api/explore” .....	37
Gambar 4.13 Fungsi <i>getCategories</i> pada kelas GoogleTrend.....	37
Gambar 4.14 Fungsi <i>getMultilineData</i> pada kelas GoogleTrend .....	38
Gambar 4.15 Hasil fungsi <i>getMultilineData</i> kata kunci “ <i>covid-19</i> ” .....	39
Gambar 4.16 Fungsi <i>getRelatedQueries</i> pada kelas GoogleTrend .....	39
Gambar 4.17 Grafik penelusuran 1 Maret 2020 – 30 April 2020 .....	41
Gambar 4.18 Grafik penelusuran 1 April 2020 – 31 Mei 2020 .....	42
Gambar 4.19 Implementasi proxy rotator .....	44
Gambar 4.20 Alur penggunaan website analisis korelasi data faktual dan Google Trends.....	46
Gambar 4.21 Tampilan halaman home .....	47
Gambar 4.22 Contoh tampilan fitur sugestif .....	48
Gambar 4.23 Halaman progress.....	48
Gambar 4.24 Status saat menunggu antrian.....	49
Gambar 4.25 Status pada saat proses <i>scraping</i> .....	49
Gambar 4.26 Status saat selesai proses <i>scraping</i> .....	49

Gambar 4.27 Halaman result .....	50
Gambar 4.28 Hasil analisis korelasi data faktual demam berdarah .....	52
Gambar 4.29 Skenario nomor 1 .....	56
Gambar 4.30 Skenario nomor 2 .....	56
Gambar 4.31 Skenario nomor 3 .....	57
Gambar 4.32 Contoh <i>template</i> data faktual yang salah .....	57
Gambar 4.33 Skenario nomor 4 .....	58
Gambar 4.34 Skenario nomor 5 .....	58
Gambar 4.35 Skenario nomor 6 .....	59
Gambar 4.36 Skenario nomor 7 .....	60
Gambar 4.37 Skenario nomor 8 .....	60
Gambar 4.38 Skenario nomor 9 .....	61

# BAB I

## PENDAHULUAN

### 1.1 Latar Belakang

Saat ini kebutuhan akan internet menjadi hal yang tak terpisahkan dari kehidupan masyarakat Indonesia. Hal ini dibuktikan dengan jumlah pengguna internet aktif di Indonesia sepanjang tahun 2018 sebanyak 123 juta jiwa, yang memosisikan Indonesia sebagai negara keenam dengan jumlah pengguna internet terbanyak setelah negara maju seperti Jepang dan lain sebagainya (Biro Humas Kementerian Komunikasi dan Informatika, 2018). Banyaknya pengguna internet di Indonesia di pengaruhi oleh beberapa faktor salah satunya adalah kebutuhan akan informasi. Menurut (Dan et al., 2013) informasi merupakan kebutuhan yang sangat penting bagi kehidupan manusia untuk mendukung kehidupan sehari-hari. Hal ini mendorong sebagian masyarakat Indonesia berperan aktif mencari informasi melalui internet salah satunya dengan memanfaatkan layanan mesin pencari atau *search engine*.

Mesin pencari adalah salah satu layanan internet yang sering digunakan oleh masyarakat untuk mencari informasi. Mesin pencari sering digunakan untuk berbagai keperluan seperti mencari berita, informasi, tutorial berupa gambar, video, tulisan, aplikasi komputer, lokasi peta bahkan bisa juga sebagai sarana mencari penghasilan (Bunafit, 2016). Kecepatan dan kesesuaian hasil yang diberikan oleh mesin pencari menjadi keunggulan pencarian informasi yang diberikan dibanding menggunakan media informasi lainnya seperti televisi dan media cetak. Saat ini banyak perusahaan yang menyediakan layanan mesin pencari yang dapat digunakan oleh masyarakat umum seperti Google, Bing, Yandex, AOL dan lain sebagainya. Berbagai macam inovasi dilakukan oleh perusahaan penyedia layanan mesin pencari untuk dapat bersaing dengan perusahaan lainnya, seperti pencarian menggunakan gambar, suara dan lain sebagainya.

Penggunaan mesin pencari oleh pengguna internet khususnya masyarakat Indonesia meninggalkan jejak digital dalam setiap pencarian informasi yang dilakukan. Jejak digital ini digunakan oleh perusahaan penyedia layanan mesin pencari untuk menyajikan informasi baru berupa tren istilah pencarian. Salah satu layanan yang menyediakan tren istilah pencarian adalah Google Trends. Google Trends menyediakan informasi berupa tren istilah pencarian dalam kurun waktu dan lokasi tertentu. Dalam perkembangannya Google Trends banyak digunakan oleh perorangan, perusahaan maupun peneliti untuk melakukan riset. Salah satu



penelitian dengan memanfaatkan Google Trends adalah penelitian yang dilakukan oleh (Kang, Zhong, He, Rutherford, & Yang, 2013). Dalam penelitiannya (Kang et al., 2013) melakukan analisis korelasi antara data Google Trends dengan data faktual yang didapatkan dari institusi setempat untuk pemantauan penyakit flu di China selatan. Kesimpulan yang didapatkan oleh (Kang et al., 2013) pada penelitian tersebut adalah bahwa data Google Trends dapat digunakan sebagai sumber data pelengkap untuk pengawasan penyakit flu di China selatan. Penelitian lain yang juga memanfaatkan Google Trends adalah penelitian yang dilakukan oleh (Tkachenko et al., 2017). Pada penelitian tersebut (Tkachenko et al., 2017) melakukan pengujian dengan beberapa skenario pemodelan menggunakan beberapa variabel untuk membuktikan bahwa data Google Trends dapat digunakan untuk meningkatkan hasil pengawasan penyakit diabetes. Dari penelitian yang dilakukan (Tkachenko et al., 2017) menemukan fakta bahwa terdapat korelasi yang kuat antara kata kunci dengan pencarian umum menggunakan istilah diabetes. Di Indonesia sendiri penelitian dengan memanfaatkan Google Trends juga pernah dipublikasikan pada tahun 2019 oleh (Husnayain, Fuad, & Lazuardi, 2019) dengan tema kesehatan. Pada penelitian tersebut dilakukan uji korelasi antara data Google Trends pada demam berdarah dan laporan pengawasan nasional Indonesia. Dari penelitian tersebut ditemukan bahwa Google Trends berpotensi digunakan untuk sistem peringatan dini dan alat baru untuk memantau reaksi publik sebelum peningkatan kasus demam berdarah dan selama wabah.

Dari beberapa penelitian di atas dapat diambil kesimpulan bahwa, tren pencarian untuk beberapa kata kunci yang ada pada Google Trends berpeluang untuk dijadikan data pelengkap yang dapat digunakan berdampingan dengan data faktual guna mendukung sebuah penelitian atau kegiatan yang lainnya seperti *monitoring / surveillance*. Berdasarkan fakta di atas maka, dalam penelitian ini akan dihasilkan aplikasi untuk membantu menganalisis korelasi antara data tren pencarian yang terdapat pada Google Trends dengan data faktual untuk kegiatan *monitoring / surveillance*. Data faktual yang dimaksud adalah data yang bersifat atau didasarkan pada fakta seperti misalnya hasil survei dan lain sebagainya. Pada dasarnya aplikasi ini akan melakukan pengambilan data tren pencarian dari Google Trends berdasarkan kata kunci yang telah ditentukan oleh pengguna, kemudian akan dilakukan proses normalisasi data untuk disesuaikan dengan data faktual yang pengguna miliki dan yang terakhir adalah proses analisis korelasi antara data tren pencarian dengan data faktual. Proses normalisasi data dilakukan karena, adanya kemungkinan perbedaan kisaran nilai antara data tren pencarian yang didapat dari Google Trends yaitu kisaran 0-100 dan data faktual yang dimiliki oleh setiap peneliti yang kemungkinan akan berbeda-beda. Pada proses analisis korelasi aplikasi ini

mampu melakukan analisis korelasi dengan beberapa kata kunci sekaligus sehingga diharapkan akan menghasilkan analisis korelasi yang lebih akurat. Hasil analisis korelasi pada aplikasi ini akan disajikan dalam bentuk grafik dan tabel sehingga memudahkan pengguna dalam membaca hasil analisis. Aplikasi ini juga dilengkapi dengan fitur kata kunci terkait, di mana fitur ini akan menampilkan saran beberapa kata kunci yang berhubungan dengan kata kunci yang sedang dianalisis oleh pengguna. Diharapkan dengan adanya fitur kata kunci terkait ini, kata kunci yang di hasilkan dapat digunakan sebagai alternatif *search terms* seperti misalnya pencarian utama adalah ‘demam berdarah’ *search terms* yang dihasilkan adalah ‘dbd’. Fitur selanjutnya adalah *date range selector*, fitur ini memungkinkan pengguna untuk menentukan interval waktu yang akan dianalisis sesuai dengan ketersediaan data yang ada pada Google Trends dan data faktual. Fitur *range selector* ini dapat digunakan untuk mencari korelasi positif yang lebih tinggi atau sebaliknya dalam kurun waktu tertentu.

## 1.2 Rumusan Masalah

Berdasarkan latar belakang tersebut rumusan masalah yang dapat diangkat adalah bagaimana membangun aplikasi untuk membantu menganalisis korelasi antara data tren pencarian yang terdapat pada Google Trends dengan data faktual untuk kegiatan *monitoring / surveillance*.

## 1.3 Batasan Masalah

Dalam penelitian ini terdapat beberapa batasan masalah yang ditetapkan, yaitu sebagai berikut:

1. Aplikasi yang dibangun hanya untuk melakukan mengujian korelasi pearson.
2. Aplikasi yang dibangun adalah berbasis *web*.
3. Analisis korelasi hanya untuk keseluruhan wilayah Indonesia .
4. Data faktual yang digunakan menggunakan berkas Microsof Excel

## 1.4 Batasan Masalah

Dalam penelitian ini terdapat beberapa batasan masalah yang ditetapkan, yaitu sebagai berikut:

1. Aplikasi yang dibangun hanya untuk melakukan mengujian korelasi *pearson*.
2. Aplikasi yang dibangun adalah berbasis *web*.
3. Analisis korelasi hanya untuk keseluruhan wilayah Indonesia.

4. Data faktual yang digunakan oleh peneliti menggunakan format Microsoft Excel

### **1.5 Tujuan Penelitian**

Tujuan dari penelitian ini yaitu membangun aplikasi untuk membantu menganalisis korelasi antara data tren pencarian yang terdapat pada Google Trends dengan data faktual untuk kegiatan *monitoring / surveillance*.

### **1.6 Manfaat Penelitian**

Adapun manfaat penelitian ini adalah untuk membantu menganalisis korelasi antara data tren pencarian yang terdapat pada Google Trends dengan data faktual untuk kegiatan *monitoring / surveillance*.

### **1.7 Metode Penelitian**

Metode penelitian yang digunakan dalam penelitian tugas akhir ini adalah sebagai berikut:

1. Identifikasi Masalah

Langkah pertama dimulai dengan pencarian masalah yang kemungkinan muncul selama pembangunan sistem.

2. Analisis Kebutuhan

Pada proses ini, dilakukan penganalisan dan pengumpulan kebutuhan sistem yang meliputi informasi kebutuhan *software* maupun *hardware* dan fungsi yang dibutuhkan untuk performansi aplikasi yang akan dibuat. Hasil penganalisaan dan pengumpulan kebutuhan spesifikasi didokumentasikan dan digunakan untuk proses selanjutnya.

3. Perancangan Sistem

Pada proses perancangan sistem, dilakukan penerjemahan kebutuhan sebuah perancangan perangkat lunak yang dapat diperkirakan sebelum berlanjut pada proses implementasi. Proses ini berfokus pada perancangan arsitektur sistem untuk performansi sistem yang akan dibuat.

4. Implementasi Sistem

Implementasi merupakan proses menerjemahkan perancangan sistem ke bentuk yang dapat dimengerti oleh mesin. Aplikasi dibuat dengan menggunakan *framework* yaitu laravel dengan menggunakan bahasa pemrograman PHP, kemudian untuk proses *scraping*

akan dilakukan dengan menggunakan salah satu metode dari beberapa metode yaitu HTML, DOM, XPATH, atau API sesuai dengan percobaan pada proses *scraping*.

## 5. Evaluasi

Setelah proses implementasi selesai, dilanjutkan dengan proses evaluasi di mana evaluasi ini berupa pengujian fungsionalitas dan Proses ini berfungsi untuk memeriksa segala kemungkinan terjadinya kesalahan dan memeriksa apakah hasil dari pengembangan tersebut sesuai dengan hasil yang diinginkan.

## 1.8 Sistematika Penelitian

Sistematika penulisan dalam penelitian ini terdiri dari beberapa pokok pembahasan yaitu sebagai berikut:

### BAB I Pendahuluan

Bab ini berisi latar belakang tentang aplikasi yang akan dikembangkan, rumusan masalah pada penelitian ini, batasan masalah pada pengerjaan penelitian, tujuan pengembangan aplikasi, manfaat aplikasi ini, metode penelitian yang digunakan dan sistematika penulisan.

### BAB II Landasan Teori

Bab ini berisi tentang teori-teori yang menjadi dasar dalam penelitian yang dilakukan dan review beberapa penelitian yang sejenis.

### BAB III Metodologi Penelitian

Bab ini berisi tentang analisis sistem yang dibutuhkan melingkupi identifikasi masalah, rancangan solusi beserta analisis-analisis kebutuhannya dan perancangan sistem yang berasal dari hasil analisis sistem dalam bentuk *Activity Diagram*, dan desain antarmukanya.

### BAB IV Implementasi dan Pengujian Sistem

Bab ini berisi penjelasan dari hasil implementasi yang telah dilakukan beserta hasil evaluasi dari pengujian yang telah diuji oleh mahasiswa.

### BAB V Kesimpulan dan Saran

Bab ini berisi kesimpulan yang ditarik dari proses, hasil maupun evaluasi selama penelitian ini berlangsung beserta saran dari penelitian yang telah dilakukan yang bertujuan untuk perbaikan dan pengembangan aplikasi di masa mendatang

## **BAB II**

### **LANDASAN TEORI**

#### **2.1 Google Trends**

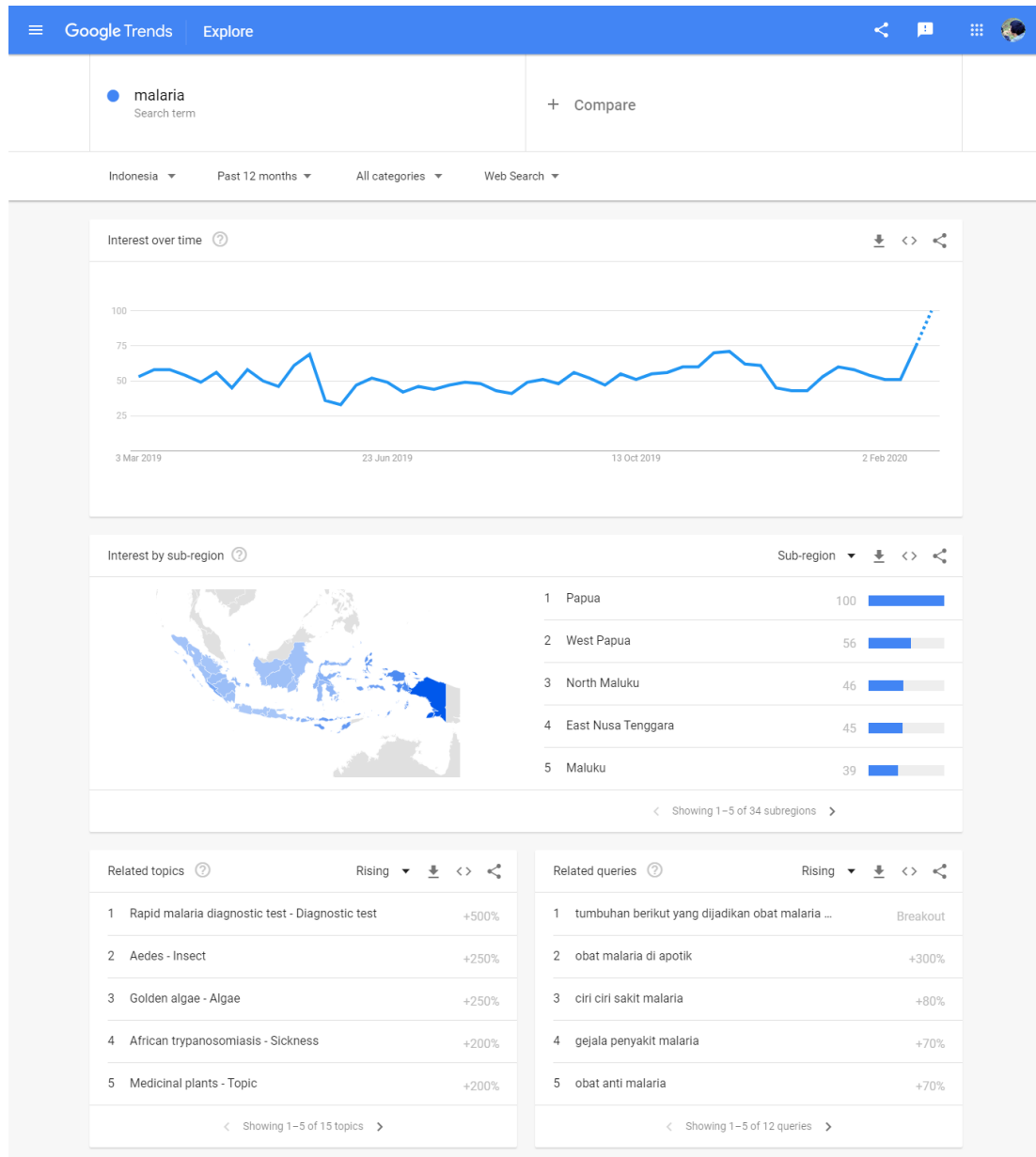
Google Trends adalah layanan dari Google yang diluncurkan pada tahun 2006. Google Trends menyediakan akses untuk bisa mendapatkan sampel data dari hasil pencarian yang dilakukan oleh pengguna mesin pencari Google. Sampel data yang disediakan dalam Google Trends adalah hasil dari perwakilan dataset dari seluruh pencarian yang dilakukan menggunakan mesin pencari Google. Menurut (Google, 2020) keterbatasan dalam menyediakan akses ke seluruh kumpulan data dikarenakan oleh pertimbangan banyaknya data yang ada, karena faktanya mesin pencari Google menangani milyaran pencarian setiap harinya.

Dalam penyajian sampel data, Google Trends melakukan normalisasi dari semua data pencarian yang ada pada mesin pencari Google yang kemudian diskalakan pada kisaran 0 – 100. Nilai tersebut kemudian disajikan dalam bentuk grafik yang dikelompokkan menjadi 4 yaitu, minat seiring waktu, minat berdasarkan wilayah, topik yang berkaitan dan kueri pencarian yang berkaitan. Dalam penggunaannya Google Trends menyediakan fungsi penyaringan yang diklasifikasikan menjadi 4 kategori yaitu wilayah, rentang waktu, kategori pencarian, dan sumber pencarian. Dari ke 4 kategori tersebut pengguna Google Trends dapat mengkombinasikannya sesuai dengan hasil yang diinginkan. Gambar 2.1 adalah tampilan antarmuka Google Trends.

Menurut (Google, 2020) keakuratan data Google Trends dalam mencerminkan aktivitas pencarian memang tidak bisa dikatakan sempurna mengingat adanya pencarian yang tidak alami seperti pencarian otomatis atau kueri yang bersifat spam. Untuk meminimalisir pencarian yang tidak alami dan untuk mendapatkan data yang akurat, Google Trend menyaring jenis pencarian menggunakan beberapa aturan, yaitu:

- a. Pencarian yang dilakukan oleh beberapa orang saja  
Data tren hanya menampilkan pencarian populer, sehingga istilah pencarian dengan volume yang rendah akan bernilai 0.
- b. Pencarian duplikat atau berulang  
Google Trends akan menghilangkan pencarian yang dilakukan oleh orang yang sama dalam kurun waktu yang singkat.
- c. Pencarian dengan karakter khusus

Google Trends akan menyaring kueri yang mengandung karakter khusus seperti tanda kutip dan karakter khusus lainnya.



Gambar 2.1 Antarmuka Google Trends

Sumber: Google Trends (2020)

## 2.2 Web Scraping

Pada proses penulisan landasan teori penulis menemukan beberapa definisi *web scraping* dari berbagai sumber, di antaranya:

- a. Menurut (Turland, 2010) *Web Scraping* adalah proses pengambilan dokumen semi terstruktur dari internet untuk kemudian dianalisis dan dilakukan ekstraksi dengan tujuan untuk mendapatkan data spesifik yang dapat digunakan untuk kepentingan yang lain. Pada umumnya dokumen yang dianalisis berupa halaman web dengan bahasa markup *HyperText Markup Language* (HTML) atau *eXtensible HyperText Markup Language* XHTML.
- b. Menurut (Mooney, Westreich, & El-Sayed, 2015) *Web Scraping* yang juga dikenal sebagai ekstraksi web adalah teknik untuk mengekstraksi data dari *World Wide Web* (WWW) yang kemudian disimpan ke dalam berkas sistem atau disimpan pada basisdata untuk kemudian diambil dan dianalisis lain waktu. Biasanya *web scraping* dilakukan menggunakan protokol *Hyper Text Transfer Protocol* (HTTP) atau melalui peramban web, baik secara manual oleh pengguna maupun secara otomatis oleh bot atau *crawler* web. Dikarenakan WWW menghasilkan data yang sangat banyak secara terus-menerus maka *web scraping* secara luas diakui sebagai teknik yang sangat efisien dan kuat dalam mengumpulkan data yang sangat besar.
- c. Menurut (Haddaway, 2016) *Web Scraping* adalah istilah yang digunakan untuk menggambarkan ekstraksi data dari berkas HTML di internet menggunakan program komputer. Biasanya data ini merupakan data berpola, seperti lis atau tabel. Program yang berinteraksi dengan halaman *web* dan mengekstrak data menggunakan beberapa set perintah dikenal sebagai *Application Programming Interfaces* (API).

### 2.2.1 Metode *Web Scraping*

Saat ini metode *web scraping* baru banyak bermunculan seiring dengan perkembangan teknologi pada layanan web. Salah satu contoh perkembangan metode *web scraping* adalah semenjak diperkenalkannya *Document Object Model* atau DOM, muncul sebuah metode baru yaitu DOM parsing yang dapat menjadi alternatif selain HTML parsing. Berikut beberapa metode *web scraping*, di antaranya:

#### a. *Manual Scraping*

Metode *manual scraping* dilakukan dengan cara manual tanpa menggunakan bantuan program komputer. *Manual scraping* dipilih jika :

- 1 Jumlah data yang akan diekstrak tidak banyak atau cenderung sedikit
- 2 Tidak membutuhkan tugas yang memiliki sifat perulangan atau rekursi.
- 3 Proses pembuatan program *scraping* membutuhkan waktu yang lebih lama jika dibandingkan dengan *scraping* data secara manual.

4 Adanya kendala yang menyebabkan suatu halaman *web* tidak memperbolehkan pengambilan data secara otomatis.

b. HTML parsing

Saat ini *web scraping* tidak hanya digunakan untuk proses ekstraksi data dari sumber data yang terstruktur seperti csv atau json, namun juga dapat digunakan untuk ekstraksi data dari halaman antarmuka yang sebenarnya diperuntukan bagi pengguna suatu website. Salah satu struktur halaman yang dapat dilakukan ekstraksi adalah HTML dengan menggunakan teknik HTML parsing. HTML parsing adalah sebuah teknik dalam *web scraping* dengan cara memanfaatkan elemen-elemen yang ada pada HTML. Biasanya struktur HTML pada sebuah halaman antarmuka memiliki pola yang sama untuk elemen yang sama. Hal ini memungkinkan halaman antarmuka sebuah website menjadi suatu sumber data.

c. DOM parsing

*Document Object Model* (DOM) parsing adalah teknik *web scraping* yang merupakan hasil perkembangan dari teknik *HTML parsing*. Perbedaan mendasar teknik *DOM parsing* dengan *HTML parsing* adalah pada penggunaan *Cascading Stylesheets* (CSS) dan Javascript dalam proses ekstraksi data. Penggunaan CSS dan Javascript menjadikan teknik ini dapat menyeleksi elemen-elemen yang akan dilakukan ekstraksi dengan penanda yang lebih spesifik seperti atribut *id* dan *class*. Kelebihan lain dari teknik ini adalah kemudahan pada proses navigasi saat penyeleksian elemen-elemen.

d. XPATH

*XML Path Language* (XPATH) sama halnya seperti *DOM parsing* teknik ini juga menawarkan kemudahan dalam proses penyeleksian elemen-elemen pada dokumen HTML untuk memudahkan proses ekstraksi data.

e. API

Berbeda dengan teknik *web scraping* lainnya, teknik ini tidak memerlukan *parsing* data yang terlalu rumit. Data yang didapatkan biasanya merupakan data yang terstruktur seperti JSON.

### 2.3 Legalitas Web Scraping

Dalam kegiatan *scraping* perlu diperhatikan beberapa aspek yang harus dipenuhi agar tidak merugikan pihak manapun. Menurut (Draxl, 2018) ada beberapa aspek yang perlu diperhatikan, yaitu:

a. Syarat dan Ketentuan



Banyak situs web yang mencatumkan syarat dan ketentuan penggunaan website mereka. Pada suatu keadaan tertentu pihak situs web dimungkinkan untuk menuntut atas pelanggaran yang mungkin dilakukan pada penggunaan web mereka.

b. Hak Cipta dan Kekayaan Intelektual

*Web scraping* adalah proses yang melibatkan penyalinan data atau informasi, untuk itu perlu diperhatikan kategori konten yang diambil saat proses *web scraping*. Beberapa konten yang dilindungi hak cipta seperti karya sastra, foto dan lain sebagainya.

c. Hak Basisdata

Hak basisdata kemungkinan dilanggar apabila data yang diambil pada proses *scraping* merupakan data yang bersifat rahasia tanpa persetujuan pemilik web.

d. Merek Dagang

Jika pada proses *scraping* data yang diambil mencerminkan merek dagang suatu perusahaan yang memiliki legalitas hukum tanpa persetujuan pemilik, pemilik merek dagang dapat mengklaim atas pelanggaran yang terjadi.

### 2.3.1 Pemanfaatan *Web Scraping*

Pada perkembangan teknologi informasi *web scraping* dapat dimanfaatkan sebagai sumber data baru yang dapat digunakan untuk berbagai macam keperluan. Berikut beberapa pemanfaatan *web scraping* menurut (Draxl, 2018).

a. *Market analysis and research*

Saat ini pengumpulan data menggunakan metode *web scraping* menjadi salah satu strategi perusahaan untuk melakukan penelitian pasar. Data yang dikumpulkan akan dianalisis untuk dapat menjadi bahan pertimbangan strategi pemasaran suatu perusahaan. Metode ini menawarkan respon yang lebih cepat jika dibandingkan dengan metode survei konvensional.

b. *Polling Opini*

*Web scraping* juga dapat dimanfaatkan untuk mengumpulkan data respon terhadap sesuatu. Misalnya, respon publik untuk sebuah judul film.

c. *Social Network Mining*

Media Sosial telah menjadi salah satu sumber data utama untuk penelitian kuantitatif selama beberapa dekade terakhir.

d. *Government Services*

Penggunaan *web scraping* dalam pemerintahan seperti contoh pemantauan kegiatan yang melanggar hukum pada media sosial merupakan sumber informasi penting bagi badan penegak hukum.

## 2.4 Time Series

Menurut (Ashari, 2015) *time series* adalah suatu representasi dari realisasi suatu variabel acak yang biasanya mempunyai interval waktu yang sama dan diamati pada suatu periode tertentu. Secara garis besar, menurut (Ashari, 2015) pola data *time series* dapat dikelompokkan menjadi 2, yaitu:

### a. Level Konstan

Pola konstan menunjukkan data bergerak sekitar rata-rata, dengan variasi jarak yang terkadang 'jauh' dan terkadang 'dekat'. Akan tetapi secara umum data tidak menunjukkan adanya perubahan tren, baik meningkat maupun menurun.

### b. Tren Linear

Pola data yang menunjukkan adanya perubahan rata-rata, tetapi tetap merupakan karakteristik dari pola tren.

## 2.5 Literature Review

Pada penelitian ini penulis menggunakan penelitian-penelitian yang sudah ada sebelumnya sebagai referensi dalam pengembangan aplikasi alat bantu analisis korelasi data faktual dan Google Trends. Adapun penelitian-penelitian yang penulis rangkum yaitu:

- a. Penelitian yang dilakukan oleh (Budiarti, Widyatmoko, Hariadi, & Purnomo, 2017) dengan judul "*Web scraping for automated water quality monitoring system: A case study of PDAM Surabaya*". Dalam penelitian tersebut *web scraping* digunakan untuk mengumpulkan data dari alat sensor pendeteksi kualitas air pada waduk-waduk yang dikelola oleh PDAM Surabaya. Data yang berhasil dikumpulkan kemudian digunakan untuk memantau kualitas air yang ada.
- b. Penelitian yang dilakukan oleh (Kang et al., 2013) dengan judul "*Using Google Trends for Influenza Surveillance in South China*". Dalam penelitiannya (Kang et al., 2013) melakukan analisis korelasi antara data Google Trends dengan data faktual yang didapatkan dari institusi setempat untuk pemantauan penyakit flu di China selatan. Kesimpulan yang didapatkan oleh (Kang et al., 2013) pada penelitian tersebut adalah

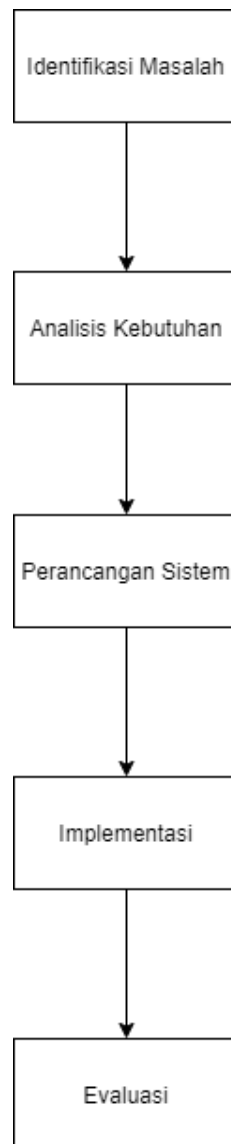
bahwa data Google Trends dapat digunakan sebagai sumber data pelengkap untuk pengawasan penyakit flu di China selatan

- c. Penelitian yang dilakukan oleh (Josi, Abdillah, & Suryayusra, 2014) dengan judul “Penerapan Teknik *Web Scraping* Pada Mesin Pencari Artikel Ilmiah”. Pada penelitian tersebut, *web scraping* digunakan untuk mengumpulkan informasi berupa *bibliography* dari suatu artikel ilmiah yang terdapat pada mesin pencari artikel ilmiah seperti Portal Garuda, *Indonesian Scientific Journal Database (ISJD)* dan Google Scholar.
- d. Penelitian yang dilakukan oleh (Tkachenko et al., 2017) dengan judul “*Google Trends can improve surveillance of Type 2 diabetes*”. Pada penelitian tersebut (Tkachenko et al., 2017) melakukan pengujian dengan beberapa skenario pemodelan menggunakan beberapa variabel untuk membuktikan bahwa data Google Trends dapat digunakan untuk meningkatkan hasil pengawasan penyakit diabetes. Dari penelitian yang dilakukan (Tkachenko et al., 2017) menemukan fakta bahwa terdapat korelasi yang kuat antara kata kunci dengan pencarian umum menggunakan istilah diabetes.
- e. Penelitian yang dilakukan oleh (Husnayain et al., 2019) dengan judul “*Correlation between Google Trends on dengue fever and national surveillance report in Indonesia*”. Dalam penelitian tersebut dilakukan uji korelasi antara data Google Trends pada demam berdarah dan laporan pengawasan nasional Indonesia. Dari penelitian tersebut ditemukan bahwa Google Trends berpotensi digunakan untuk sistem peringatan dini dan alat baru untuk memantau reaksi publik sebelum peningkatan kasus demam berdarah dan selama wabah.

### **BAB III**

## **METODOLOGI PENELITIAN**

Dalam penelitian ini metodologi penelitian penulis bagi menjadi beberapa tahap. Pertama tahap indentifikasi masalah, analisis kebutuhan, perancangan sistem, implementasi, dan evaluasi. Gambar 2.1 menunjukan visualisasi alur metodologi penelitian.



Gambar 3.1 Alur metode penelitian

### 3.1 Identifikasi Masalah

Dari beberapa penelitian pada bagian 1.1 dapat diambil kesimpulan bahwa, tren pencarian untuk beberapa kata kunci yang ada pada Google Trends berpeluang untuk dijadikan data pelengkap yang dapat digunakan berdampingan dengan data faktual guna mendukung sebuah penelitian atau kegiatan yang lainnya seperti *monitoring / surveillance*. Berdasarkan fakta di atas maka, dalam penelitian ini akan dihasilkan aplikasi untuk membantu menganalisis korelasi antara data tren pencarian yang terdapat pada Google Trends dengan data faktual untuk kegiatan *monitoring / surveillance*. Aplikasi ini akan dilengkapi dengan beberapa fitur, yaitu:

a. Kata kunci terkait

Fitur ini akan menampilkan saran beberapa kata kunci yang berhubungan dengan kata kunci yang sedang dianalisis oleh pengguna. Diharapkan dengan adanya fitur kata kunci terkait ini, kata kunci yang di hasilkan dapat digunakan sebagai alternatif *search terms* seperti misalnya pencarian utama adalah ‘demam berdarah’ *search terms* yang dihasilkan adalah ‘dbd’.

b. *Date Range Selector*

Fitur ini memungkinkan pengguna untuk menentukan interval waktu yang akan dianalisis sesuai dengan ketersediaan data yang ada pada Google Trends dan data faktual. Fitur *date range selector* ini dapat digunakan untuk mencari korelasi positif yang lebih tinggi atau sebaliknya dalam kurun waktu tertentu.

### 3.2 Analisis Kebutuhan

Pada penelitian ini analisis kebutuhan dimaksudkan untuk mengetahui data dan informasi apa saja yang dibutuhkan dalam pengembangan aplikasi untuk alat bantu analisis korelasi data faktual dan Google Trends untuk *monitoring / surveillance*. Untuk itu penulis mengumpulkan beberapa referensi yang dapat membantu dalam penelitian ini. Sehingga nantinya aplikasi yang dibuat dapat dikembangkan sesuai dengan apa yang diharapkan.

#### 3.2.1 Analisis Kebutuhan Masukan

Aplikasi ini akan memproses masukan dari pengguna yang kemudian akan diproses untuk mendapatkan informasi berupa hasil korelasi. Masukan yang dibutuhkan untuk sistem ini adalah:

a. Data faktual

Data faktual yang digunakan pada aplikasi ini adalah *time series* dengan pola interval waktu yang konstan dengan minimal interval satu hari. Data *time series* diharapkan dapat merepresentasikan data faktual yang nantinya akan digunakan untuk proses analisis korelasi dengan data *time series* yang diperoleh dari hasil *scraping* Google Trends berdasarkan kata kunci tertentu. Data *time series* ini berupa berkas Microsoft Excel yang memiliki format csv, xlsx atau xls.

b. Kata Kunci

Masukan berikutnya yang dibutuhkan oleh aplikasi adalah kata kunci. Kata kunci pada aplikasi ini diharapkan merepresentasikan variabel yang ada pada data *time series*. Kata kunci ini juga digunakan dalam proses *scraping* Google Trends untuk dapat memperoleh data *time series* yang sesuai, sehingga hasil analisis korelasi diharapkan akan menjadi lebih akurat.

c. Kategori

Masukan kategori ini dimaksudkan untuk dapat membantu proses analisis korelasi menjadi lebih spesifik berdasarkan kategori tertentu.

### 3.2.2 Analisis Kebutuhan Proses

Tahap analisis kebutuhan proses ini dimaksudkan untuk mengidentifikasi proses apa saja yang dibutuhkan dalam aplikasi ini. Berikut beberapa proses yang berhasil penulis indentifikasi.

a. Data Parsing

Pada aplikasi ini data faktual yang digunakan untuk proses analisis korelasi menggunakan format Microsoft Excel, maka dari itu dibutuhkan proses *data parsing*, untuk mengubah data Microsoft Excel ke dalam *data set* berupa *array* agar dapat diproses dalam bahasa pemrograman PHP. Selain itu *data parsing* juga akan digunakan untuk memproses repon yang didapatkan pada saat proses *scraping*.

b. Proses Web Scraping

Google Trends tidak menyediakan protokol komunikasi antar sistem seperti API. Maka dari itu dalam penelitian ini akan digunakan teknik *Web Scraping* dengan memanfaatkan protokol HTTP *Request* yang terdapat pada bahasa pemrograman PHP. Dalam melakukan *web scraping* penulis akan terlebih dahulu melakukan beberapa percobaan *scraping* untuk mengetahui metode *scraping* mana yang mungkin dapat digunakan untuk mengambil atau mengekstrak data dari Google Trends. Adapun metode yang akan penulis coba terapkan

yaitu HTML, DOM, XPATH atau API. Sehingga diharapkan dengan adanya percobaan beberapa metode *scraping*, penulis dapat menentukan metode manakah yang layak dan mungkin dapat digunakan agar proses *scraping* dapat dilakukan dengan lancar.

c. Proses Normalisasi Data

Proses normalisasi data dilakukan karena, adanya kemungkinan perbedaan kisaran nilai antara data tren pencarian yang didapat dari Google Trends yaitu kisaran 0-100 dan data faktual yang dimiliki oleh setiap pengguna yang kemungkinan akan berbeda-beda. Proses normalisasi juga digunakan untuk memproses data *time series* yang didapatkan dari Google Trends. Hal ini dikarenakan Google Trends menghasilkan data yang diskalakan dari 0 – 100 untuk setiap *request* yang diterima.

d. Proses Analisis Korelasi

Aplikasi ini akan menghasilkan informasi berupa hasil analisis korelasi antara data faktual dengan Google Trends. Perhitungan korelasi dalam aplikasi ini adalah menggunakan metode korelasi pearson. Korelasi pearson adalah ukuran korelasi yang digunakan untuk mengukur kekuatan dan arah hubungan linear antara dua variabel. Dua variabel pada aplikasi ini merupakan data faktual dan data Google Trends untuk masing-masing kata kunci.

### 3.2.3 Analisis Kebutuhan Keluaran

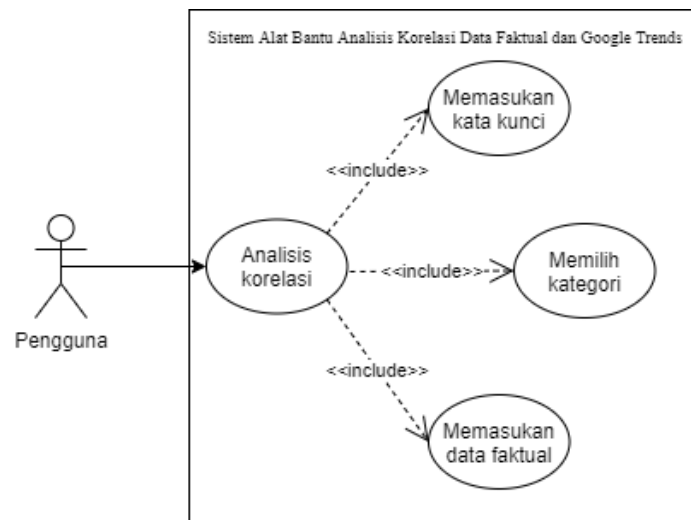
Aplikasi ini akan menghasilkan informasi berupa hasil korelasi antara data faktual dengan Google Trends sehingga dapat digunakan untuk keperluan *monitoring / surveillance*. Salah satu contoh penggunaan aplikasi ini adalah untuk membantu pengawasan penyebaran suatu penyakit di Indonesia.

## 3.3 Perancangan Aplikasi

Pada tahap perancangan aplikasi ini penulis membagi menjadi beberapa tahap yaitu pembuatan *use case diagram*, pembuatan *activity diagram*, dan perancangan antarmuka aplikasi. Tahap ini dilakukan agar dalam proses pembuatan aplikasi dapat sesuai dengan apa yang telah direncanakan sehingga dapat diharapkan dapat menghasilkan aplikasi yang sesuai dengan kebutuhan.

### 3.3.1 Use Case Diagram

Sebuah *Use Case Diagram* menyatakan visualisasi interaksi yang terjadi antara pengguna (aktor) dengan sistem. Diagram ini bisa menjadi gambaran yang bagus untuk menjelaskan konteks dari sebuah sistem sehingga terlihat jelas batasan dari sistem (Larman, 2004).



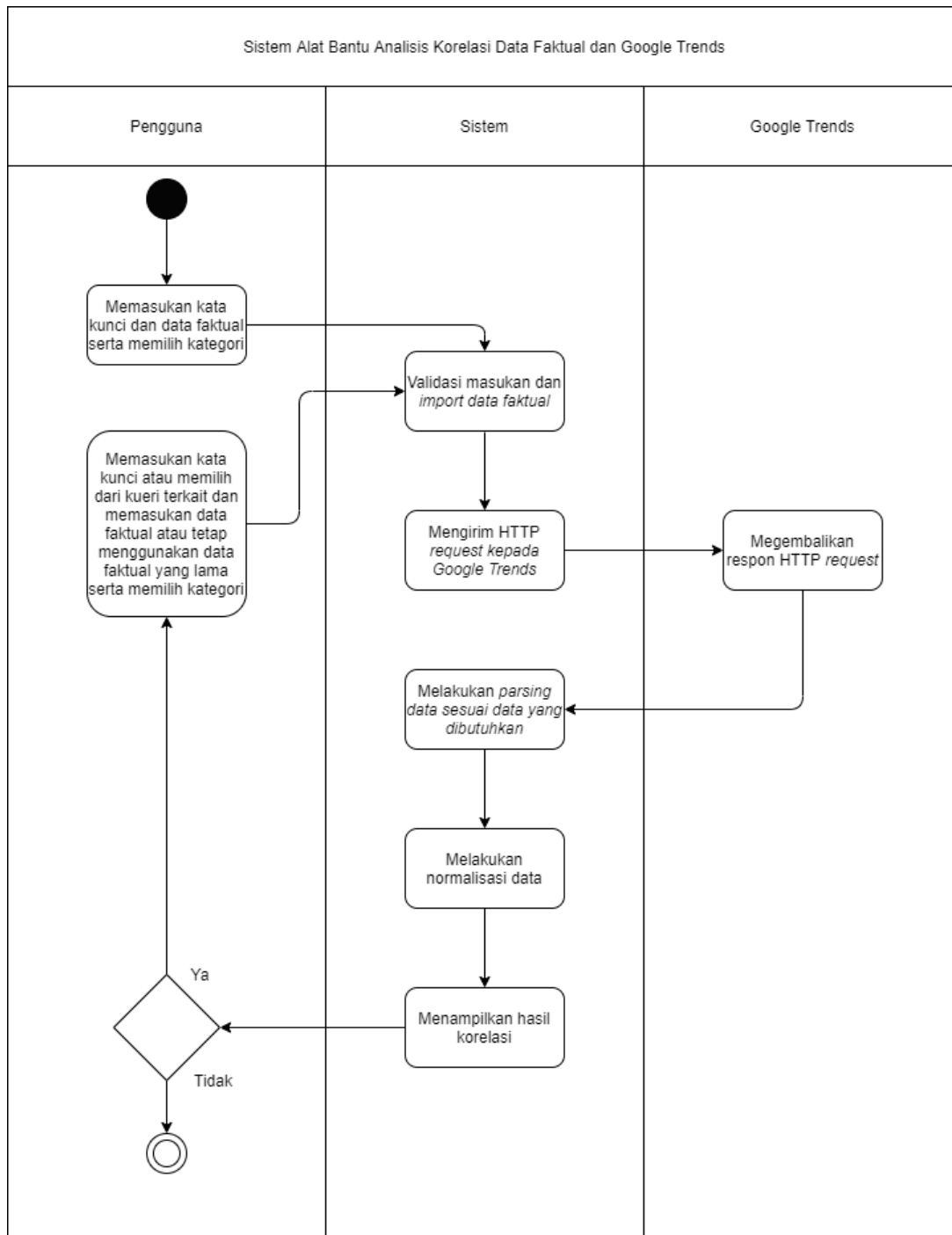
Gambar 3.2 Use Case Diagram

Pada Gambar 3.2 divisualisasikan bagaimana pengguna berinteraksi dengan aplikasi ini. Pengguna dalam hal ini adalah orang yang ingin melakukan analisis korelasi data faktual dan Google Trends. Pengguna memiliki satu aksi yaitu analisis korelasi, yang dapat dilakukan apabila telah melakukan aksi memasukkan kata kunci, memilih kategori dan memasukkan data faktual berupa *time series*. Hal ini disebabkan karena aksi masukan kata kunci, memilih kategori dan aksi memasukkan data *time series* memiliki hubungan *include* dengan aksi analisis korelasi

### 3.3.2 Activity Diagram

*Activity diagram* merupakan sebuah teknik untuk mendeskripsikan logika prosedural, proses bisnis dan aliran kerja dalam banyak kasus. *Activity diagram* digunakan untuk menganalisis *behavior* dengan *use case* yang lebih kompleks dan menunjukkan interaksi interaksi diantara mereka satu sama lain. *Activity diagram* biasanya digunakan untuk menggambarkan aktivitas bisnis yang lebih kompleks, di mana digambarkan hubungan antara satu *use case* dengan *use case* yang lainnya (Anisah & Kuswaya, 2017).





Gambar 3.3 Activity diagram

Pada Gambar 3.3 divisualisasikan bahwa pengguna dapat melakukan analisis korelasi dengan cara mengisi kata kunci dan data *time series*. Kemudian sistem akan melakukan *parsing data* dari data *time series* yang sudah pengguna unggah. Setelah itu sistem akan mengirimkan *HTTP request* kepada Google Trends sesuai dengan kata kunci dan kategori yang telah pengguna masukan sebelumnya. Setelah Google Trends memberikan respon, sistem akan

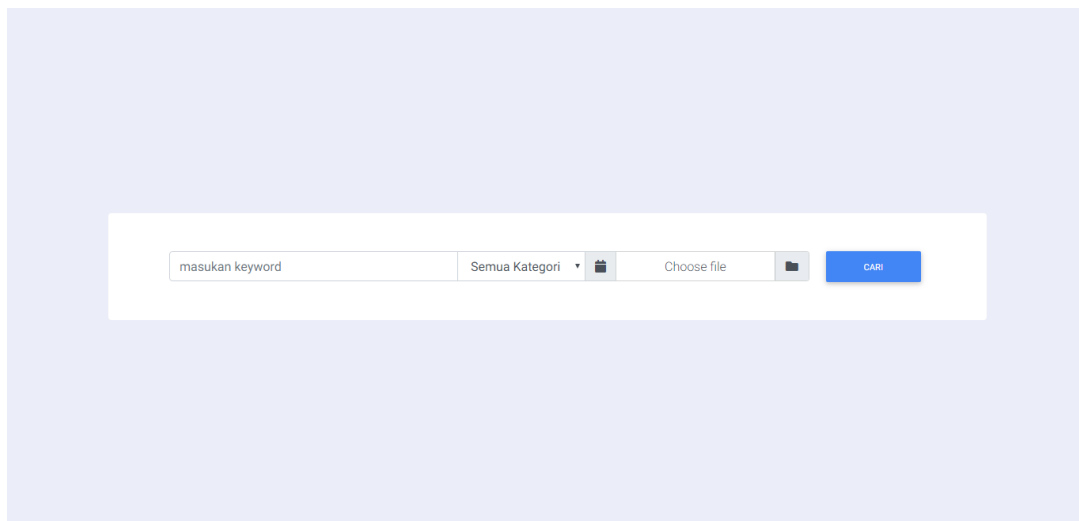
melakukan proses ekstraksi data, yang kemudian dilanjutkan dengan proses normalisasi. Kemudian akan dilakukan proses analisis korelasi untuk ditampilkan kepada pengguna. Terakhir pengguna dapat memilih kata kunci yang berkaitan untuk dilakukan analisis korelasi kembali.

### 3.3.3 Perancangan Antarmuka

Antarmuka pada aplikasi ini dibagi menjadi dua, yaitu halaman home dan result. Untuk mempermudah pada proses pembuatan aplikasi penulis membuat perancangan antarmuka yang akan diimplementasikan pada saat pembuatan aplikasi.

#### a. Halaman home

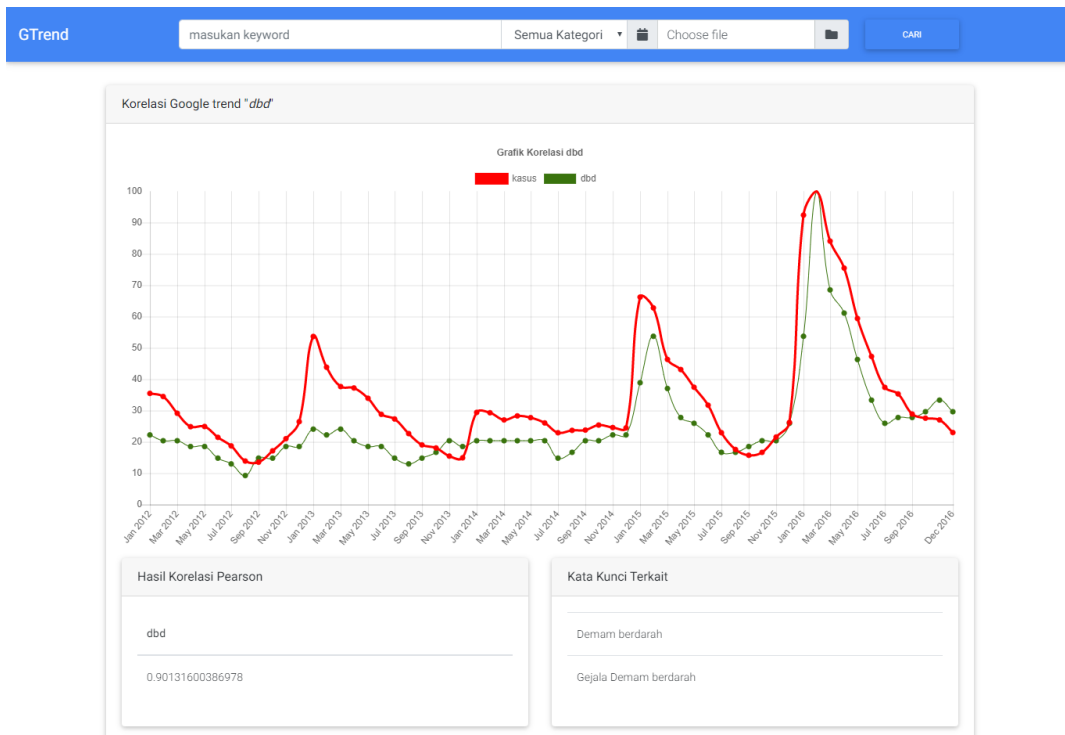
Halaman home digunakan untuk menampilkan formulir untuk proses analisis korelasi. Formulir tersebut berisikan masukan untuk kata kunci, masukan untuk berkas data *time series* dan pilihan kategori untuk kata kunci yang akan di analisis. Rancangan antarmuka halaman home ditunjukkan pada Gambar 3.4



Gambar 3.4 Rancangan halaman home

#### b. Halaman result

Halaman result digunakan untuk menampilkan hasil analisis korelasi beserta formulir yang juga terdapat pada halaman beranda. Hasil analisis akan disajikan dalam bentuk grafik. Halaman hasil analisis juga akan menampilkan kata kunci yang terkait dengan kata kunci yang dianalisis sebelumnya. Pada Gambar 3.5 ditunjukkan rancangan halaman result.



Gambar 3.5 Rancangan halaman result

### 3.4 Perancangan Pengujian

Perancangan pengujian dimaksudkan untuk membuat rancangan pengujian yang akan digunakan dalam proses evaluasi penelitian ini. Pada penelitian ini penulis menggunakan dua metode pengujian yaitu pengujian fungsionalitas dan pengujian usabilitas.

#### 3.4.1 Pengujian Fungsionalitas (Black Box)

Pada tahap evaluasi, salah satu pengujian sistem akan dilakukan untuk mengetahui apakah aplikasi alat bantu analisis korelasi data faktual dan Google Trends dapat mengimplementasikan fitur-fitur yang telah direncanakan. Pengujian ini juga di maksudkan untuk mengetahui apakah aplikasi ini dapat mengatasi kesalahan masukan yang mungkin terjadi saat penggunaan aplikasi ini. Pada Tabel 3.1 di tunjukan skenario pengujian menggunakan metode pengujian black box dan hasil yang diharapkan.

Tabel 3.1 Skenario Pengujian Menggunakan Black Box

No	Skenario Pengujian	Tes Kasus	Hasil yang diharapkan
----	--------------------	-----------	-----------------------

1	Mengkosongkan semua formulir kemudian klik cari	Kata kunci : - Kategori : “semua kategori” Time series : -	Sistem akan mengarahkan ke halaman home dengan menampilkan pesan kesalahan
2	Mengkosongkan kata kunci dengan mengisi data faktual ber-format Microsoft Excel	Kata kunci : - Kategori : “semua kategori” Time series : *.xls	Sistem akan mengarahkan ke halaman home dengan menampilkan pesan kesalahan
3	Mengisi semua formulir, namun data faktual bukan merupakan format Microsoft Excel	Kata kunci : DBD Kategori : “semua kategori” Time series : *.pdf	Sistem akan mengarahkan ke halaman home dengan menampilkan pesan kesalahan
4	Mengisi semua formulir, namun data faktual tidak sesuai <i>template</i> yang dimaksud seperti pada Gambar 4.1	Kata kunci : DBD Kategori : “semua kategori” Time series : *.xls	Sistem akan mengarahkan ke halaman home dengan menampilkan pesan kesalahan
5	Mengisi semua formulir dengan kata kunci berupa simbol	Kata kunci: !@#\$\$%^&*() Kategori : “semua kategori” Time series : *.xls	Sistem tetap memproses permintaan, namun menampilkan pesan bahwa sistem tidak berhasil mendapatkan data dengan kata kunci yang dimaksud.
6	Mengisi semua formulir, dengan kata kunci lebih dari satu	Kata kunci: DBD, OBAT DBD Kategori : “semua kategori” Time series : *.xls	Sistem memproses permintaan dan menampilkan korelasi berdasarkan kedua kata kunci

7	Membuat 2 permintaan secara berurutan dengan rentang waktu yang bersamaan	Kata kunci : DBD Kategori : “semua kategori” Time series : *.xls	Sistem akan menampilkan pesan untuk permintaan yang kedua pada halaman progress dengan menginformasikan jumlah antrian di depannya
8	Memilih kata kunci yang pada kueri yang berkaitan yang terdapat di halaman result	Kata kunci : - Kategori : “semua kategori” Time series : -	Sistem memproses permintaan dan menampilkan hasil analisis
9	Menggeser <i>date range selector</i> pada halaman hasil result	Kata kunci : - Kategori : “semua kategori” Time series : -	Sistem akan menampilkan hasil korelasi sesuai dengan waktu yang dipilih

### 3.4.2 Pengujian Usabilitas

Dalam pengujian usabilitas ini dimaksudkan untuk mengetahui tingkat kepuasan pengguna saat menggunakan aplikasi ini. Untuk dapat mengukur tingkat kepuasan pengguna peneliti membuat pertanyaan berupa kuisisioner dengan berberapa aspek pertanyaan meliputi, efisiensi, *learnability*, *memorability*, *error* dan kepuasan. Pada Tabel 3.2 ditunjukkan rancangan kuisisioner yang akan diajukan kepada pengguna aplikasi ini.

Tabel 3.2 Rancangan Pertanyaan Kuisisioner

No	Aspek	Pertanyaan
1	Efisiensi	Apakah aplikasi alat bantu analisis data faktual dan Google Trends mampu memberikan hasil analisis sesuai dengan apa yang diharapkan?
2		Apakah fitur kueri terkait dan <i>date range selector</i> membantu anda dalam proses analisis?
3	Error	Apakah aplikasi alat bantu analisis data faktual dan Google Trends minim kesalahan?
4		Apakah aplikasi alat bantu analisis data faktual dan Google Trends telah memberikan informasi yang jelas dalam menampilkan pesan kesalahan pengguna?
5	Learnability	Apakah hasil analisis aplikasi alat bantu analisis data faktual dan Google Trends mudah dipahami oleh pengguna?
6		Apakah penggunaan aplikasi alat bantu analisis data faktual dan Google Trends mudah dipahami?
7	Kepuasan	Apakah pengguna merasa puas dengan adanya aplikasi alat bantu analisis korelasi data faktual dan Google Trends?
8		Apakah dengan menggunakan aplikasi alat bantu analisis korelasi data faktual dan Google Trends membantu pengguna dalam proses analisis?
9	Memorability	Apakah pengguna merasa lebih mudah saat menggunakan aplikasi alat bantu analisis korelasi data faktual dan Google Trends saat kedua kalinya?
10		Apakah Menu pada aplikasi alat bantu analisis korelasi data faktual dan Google Trends mudah diingat?

Dari pertanyaan pada Tabel 3.2 kemudian dilakukan pengumpulan data dengan cara memberikan pertanyaan tersebut kepada responden yaitu mahasiswa. Jawaban yang akan

diberikan kepada responden akan dibagi menjadi lima kategori yang masing-masing memiliki bobot tersendiri seperti di tunjukan pada

Tabel 3.3.

Tabel 3.3 Kategori Jawaban Responden

Jawaban	Bobot
Kurang Setuju (KS)	1
Tidak Setuju (TS)	2
Ragu-ragu (R)	3
Setuju (S)	4
Sangat Setuju (SS)	5

Data yang berhasil dikumpulkan dari responden akan dihitung untuk mengukur kelayakan aplikasi ini dari tiap aspek dan diambil kesimpulannya berdasarkan skor usabilitasnya. Adapun klasifikasi kelayakan berdasarkan skor usabilitas dapat dilihat pada Tabel 3.4.

Tabel 3.4 Kategori kelayakan berdasarkan skor usabilitas

Nilai	Keterangan
1,0 – 1,5	Tidak Layak
1,6 – 2,5	Kurang Layak
2,6 – 3,5	Cukup Layak
3,6 – 4,5	Layak
4,6 – 5,0	Sangat layak

## **BAB IV**

### **IMPLEMENTASI**

Pada bab ini penulis akan menjelaskan tahapan implementasi sesuai dengan perancangan yang telah dijelaskan pada Bab III. Untuk dapat lebih memberikan gambaran pada proses implementasi, penulis menggunakan contoh skenario penggunaan aplikasi ini untuk melakukan analisis korelasi terhadap kata kunci “covid-19” antara data faktual dan Google Trends. Adapun data faktual yang penulis gunakan adalah berupa data harian jumlah pasien baru yang terkonfirmasi positif terpapar virus COVID-19 di seluruh wilayah Indonesia dari tanggal 3 Maret 2020 sampai 30 Juli 2020. Data faktual tersebut penulis dapatkan dari website Gugus Tugas Percepatan Penanganan COVID-19 yang dapat diakses pada halaman [“https://covid19.go.id/peta-sebaran”](https://covid19.go.id/peta-sebaran).

#### **4.1 Data Faktual**

Pada aplikasi analisis data faktual dan Google Trends ini, penulis menggunakan berkas Microsoft Excel sebagai sumber dari data faktual. Data faktual yang dimaksud adalah data *time series* dengan minimal interval waktu satu hari. Interval waktu pada data faktual di representasikan menggunakan tanggal mulai dan tanggal selesai.

##### **4.1.1 Template Data Faktual**

Data faktual yang digunakan pada aplikasi ini merupakan data yang secara mandiri dihimpun oleh pengguna dalam bentuk berkas Microsoft Excel. Untuk memudahkan dalam proses *parsing* data, maka penulis menetapkan *template* yang harus diterapkan pada berkas Microsoft Excel agar dapat digunakan pada aplikasi ini. Contoh *template* data faktual dapat dilihat pada Gambar 4.1. Adapun *template* tersebut harus sesuai dengan aturan di bawah ini:

- a. Kolom A dan Kolom B dibuat menggunakan *format date*
- b. Kolom A di isi dengan tanggal mulai
- c. Kolom B di isi dengan tanggal selesai
- d. Kolom C di isi dengan nilai
- e. Data harus di isi mulai dari baris pertama
- f. Minimal rentang waktu adalah satu hari, dengan cara mengisi tanggal mulai dan tanggal selesai dengan tanggal yang sama



g. Lembar yang digunakan adalah lembar pertama.

	A	B	C
1	3/2/2020	3/2/2020	2
2	3/3/2020	3/3/2020	0
3	3/4/2020	3/4/2020	0
4	3/5/2020	3/5/2020	0
5	3/6/2020	3/6/2020	2
6	3/7/2020	3/7/2020	0
7	3/8/2020	3/8/2020	2
8	3/9/2020	3/9/2020	13
9	3/10/2020	3/10/2020	8
10	3/11/2020	3/11/2020	7
11	3/12/2020	3/12/2020	0
12	3/13/2020	3/13/2020	35
13	3/14/2020	3/14/2020	27
14	3/15/2020	3/15/2020	21
15	3/16/2020	3/16/2020	17
16	3/17/2020	3/17/2020	38
17	3/18/2020	3/18/2020	55
18	3/19/2020	3/19/2020	82
19	3/20/2020	3/20/2020	60
20	3/21/2020	3/21/2020	81
21	3/22/2020	3/22/2020	64
22	3/23/2020	3/23/2020	65
23	3/24/2020	3/24/2020	106

Gambar 4.1 Contoh *template* data faktual

Pada Gambar 4.1 dapat dilihat bahwa penulis menggunakan dua penanda waktu yaitu tanggal mulai dan tanggal selesai. Hal tersebut dikarenakan untuk mengakomodir data faktual pengguna yang mungkin dalam rentang waktu harian, mingguan, bulanan, tahunan ataupun jangka waktu yang tidak biasa, sesuai dengan preferensi masing-masing pengguna dengan minimal jangka waktu adalah harian. Pada skenario data faktual covid-19, untuk kolom C pada berkas Microsoft Excel di isi dengan nilai yang merepresentasikan jumlah pasien positif pada rentang waktu yang ada pada kolom A dan Kolom C. Dikarenakan data faktual covid-19 memiliki interval waktu harian, untuk itu tanggal mulai dan tanggal selesai di isi dengan tanggal yang sama.

#### 4.1.2 *Import dan Validasi Data Faktual*

Setelah menetapkan *template* untuk data faktual proses selanjutnya ialah *import* dan validasi. Proses validasi sendiri terbagi menjadi dua tahapan, yaitu validasi berkas dan validasi data. Validasi berkas sendiri bertujuan untuk memastikan bahwa berkas yang diunggah oleh pengguna, sesuai dan tidak melebihi aturan yang telah ditetapkan oleh penulis, dengan detail sebagai berikut:

a. Ukuran berkas tidak melebihi 1024 kilobytes. Validasi ukuran berkas ini bertujuan untuk meminimalisir beban kerja *server*, sehingga *server* dapat melayani beberapa pengguna

dalam kurun waktu yang bersamaan dengan maksimal. Validasi ini juga untuk meminimalisir kinerja memori pada server, karena berkas data faktual nantinya akan disimpan ke dalam sebuah *array*.

b. Berkas harus berekstensi csv, xls atau xlsx

Setelah berkas data faktual berhasil melewati tahap validasi berkas, maka kemudian dilanjutkan dengan proses *import* dan validasi data. Untuk itu penulis membuat sebuah kelas yang bernama *DatasetImport* yang bertanggung jawab untuk melakukan import dan validasi data. Kelas *DatasetImport* ini merupakan kelas yang diimplementasikan dari kelas yang terdapat pada sebuah paket yang dikembangkan oleh Maatwebsite bernama *Laravel-Excel*. Adapun implementasi kode program proses *Import* dapat dilihat pada Gambar 4.2.

```
$import = new DatasetImport();  
Excel::import($import, $request->file('dataset'));
```

Gambar 4.2 Kode program proses *import*

Pada Gambar 4.2 penulis membuat variabel dengan nama *import* yang merupakan instansiasi dari kelas *DatasetImport*. Variabel *import* ini akan digunakan sebagai parameter pada proses *import* menggunakan fungsi *import* pada kelas *Excel*. Setelah proses *import* selesai dilakukan maka dilanjutkan dengan proses validasi data yang terdapat pada berkas data faktual yang diunggah oleh pengguna. Adapun aturan yang penulis terapkan pada proses validasi data yaitu:

- a. Data faktual sekurang kurangnya harus memiliki satu data
- b. Data tanggal mulai dan data tanggal selesai tidak boleh kosong dan harus memiliki *format* tanggal
- c. Tidak diperbolehkan adanya baris data yang kosong antara data satu dengan data yang lainnya
- d. Masing-masing data tidak boleh memiliki tanggal mulai maupun tanggal selesai yang merupakan bagian dari data lainnya
- e. Data nilai tidak boleh kosong dan harus berupa angka.

Data faktual yang telah berhasil divalidasi akan disimpan ke dalam sebuah variabel dengan tipe data *array* sebelum kemudian disimpan ke dalam basisdata. Gambar 4.3

menunjukkan struktur *array* yang merupakan hasil *import* dari data faktual covid-19 yang kemudian akan disimpan ke dalam basisdata .

```

@items: array:152 (▼)
0 => array:3 (▼)
  "start_date" => Carbon\Carbon @1583107200 (#1157 ▼)
  "constructedObjectId" => "990000002290a<0b000000002f5d13ce"
  #localMonthsOverflow: null
  #localYearsOverflow: null
  #localStrictModeEnabled: null
  #localHumanDiffOptions: null
  #localToStringFormat: null
  #localSerializer: null
  #localMacros: null
  #localGenericMacros: null
  #localFormatFunction: null
  #localTranslator: null
  #dumpProperties: array:3 (▼)
  #dumpLocale: null
  date: 2020-03-02 00:00:00.0 UTC (+00:00)
}
"end_date" => Carbon\Carbon @1583107200 (#1159 ▶)
"value" => 0.07527286413248
}
1 => array:3 (▼)
  "start_date" => Carbon\Carbon @1583193600 (#1161 ▶)
  "end_date" => Carbon\Carbon @1583193600 (#1156 ▶)
  "value" => 0.0
}
2 => array:3 (▼)
  "start_date" => Carbon\Carbon @1583280000 (#1158 ▶)
  "end_date" => Carbon\Carbon @1583280000 (#1160 ▶)
  "value" => 0.0
}
3 => array:3 (▼)
  "start_date" => Carbon\Carbon @1583366400 (#1162 ▶)
  "end_date" => Carbon\Carbon @1583366400 (#1163 ▶)
  "value" => 0.0
}
4 => array:3 (▼)
  "start_date" => Carbon\Carbon @1583452800 (#1164 ▶)
  "end_date" => Carbon\Carbon @1583452800 (#1165 ▶)
  "value" => 0.07527286413248
}
5 => array:3 (▶)
6 => array:3 (▶)
7 => array:3 (▶)

```

Gambar 4.3 Array hasil import dan validasi data faktual

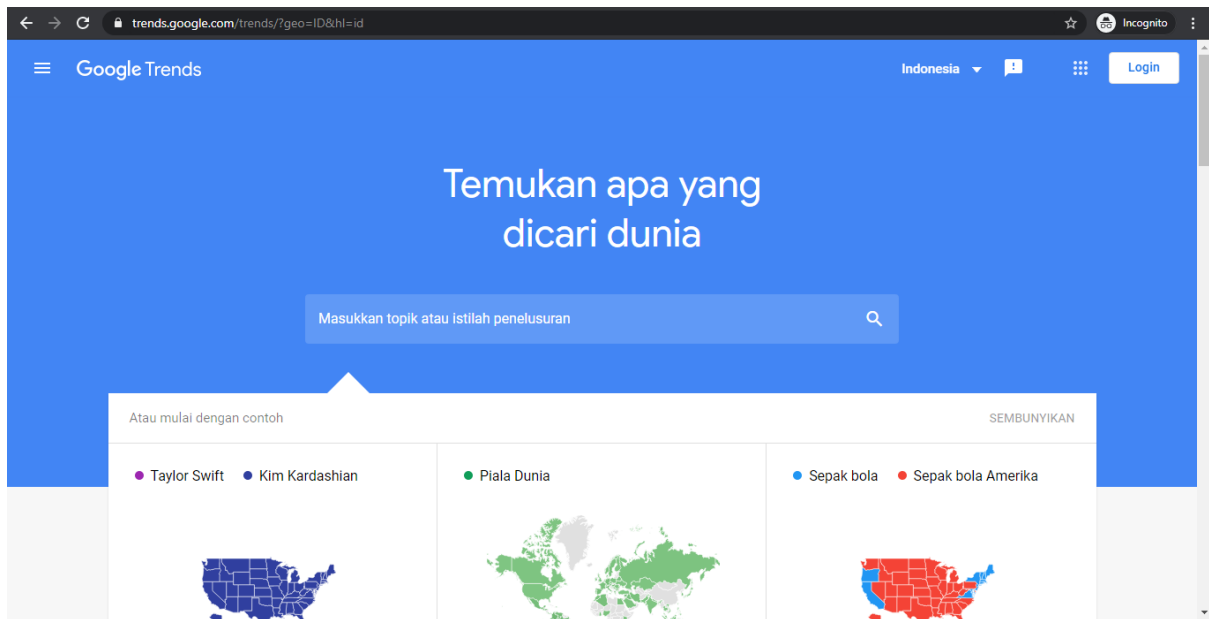
Pada Gambar 4.3 adalah hasil kumpulan *array* dari proses *import* yang berhasil dilakukan. Adapun struktur *array* pada Gambar 4.3 memiliki indeks *start\_date*, *end\_date* dan nilai dari masing-masing data .

## 4.2 Implementasi Scraping

Untuk dapat membangun aplikasi alat bantu analisis korelasi data faktual dan Google Trends untuk *monitoring* / *surveillance* salah satu tahap yang sangat penting yaitu proses *scraping*. Pada tahap ini penulis akan menjelaskan tahapan yang penulis lakukan untuk dapat melakukan *scraping* terhadap Google Trends.

### 4.2.1 Analisis Url dan User Interface Google Trends

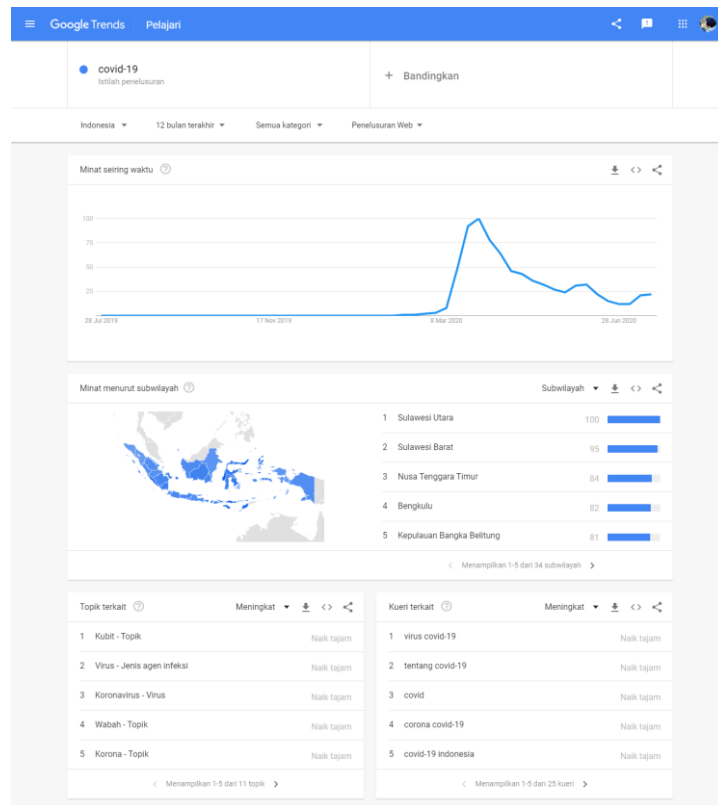
Url (*Uniform Resource Locator*) adalah suatu penanda yang dapat berupa huruf, angka atau simbol tertentu yang merujuk pada suatu *resource*. Sebagai salah satu layanan yang dimiliki oleh Google layaknya Gmail dan Google Maps, url Google Trends dibangun di atas domain yang merupakan subdomain dari domain Google itu sendiri yaitu google.com. Google Trend dapat diakses melalui alamat 'https://trends.google.com' seperti pada Gambar 4.4.



Gambar 4.4 Tangkapan layar tampilan *homepage* Google Trends

Sumber: <https://trends.google.com/trends/?geo=ID&hl=id> (Di akses 7 Juni 2020)

Pada Gambar 4.4 menunjukan tampilan *homepage* Google Trend yang menyajikan beberapa informasi seperti, tren yang sedang terjadi baru – baru ini, contoh penggunaan Google Trends untuk beberapa kata kunci dan informasi lainnya. Pada halaman ini, Google Trends juga menyediakan formulir yang dapat digunakan untuk menelusuri tren berdasarkan kata kunci. Selanjutnya formulir tersebut akan mengarahkan pengguna menuju halaman hasil penelusuran seperti pada Gambar 4.5.



Gambar 4.5 Tangkapan layar halaman hasil penelusuran Google Trends

Pada Gambar 4.5 dapat dilihat bahwa pada halaman hasil penelusuran Google Trends terdapat beberapa informasi yang ditampilkan yaitu grafik tren berdasarkan minat, grafik tren berdasarkan wilayah, topik terkait dan kueri terkait. Halaman hasil penelusuran Google Trend dapat diakses melalui url “<https://trends.google.com/trends/explore>” dengan menggunakan beberapa parameter. Berikut beberapa parameter yang berhasil penulis identifikasi yaitu:

a. q

Parameter ini merepresentasikan kata kunci yang akan dilakukan pencarian trennya. Parameter ini dapat menampung lebih dari satu nilai dengan pemisah berupa tanda koma (.). Jika parameter ini memiliki lebih dari satu nilai maka hasil tren yang ditampilkan adalah berupa hasil perbandingan atau komparasi dari masing-masing kata-kunci.

b. geo

Parameter ini merepresentasikan kode wilayah, di mana hasil trend yang nantinya akan ditampilkan adalah hasil tren yang dihimpun berdasarkan wilayah tersebut. Parameter ini dapat berupa kode negara atau provinsi pada suatu negara.

c. hl

Parameter ini adalah parameter yang digunakan untuk memilih bahasa yang digunakan pada Google Trend. Jika parameter ini tidak digunakan, maka Google Trend secara otomatis akan menggunakan bahasa Inggris-US.

d. date

Parameter ini merepresentasikan jangka waktu yang digunakan. Parameter dapat berupa tanggal dengan aturan [tanggal awal][spasi][tanggal akhir] dengan masing-masing tanggal menggunakan aturan YYYY-MM-DD. Parameter ini juga dapat di isi dengan format lain yang sudah ditentukan oleh Google Trend, seperti:

1. “now 1-H” untuk 1 jam terakhir
2. “now 1-d” untuk 1 hari terakhir
3. “now 1-m” untuk 1 bulan terakhir
4. “now 1-y” untuk 1 tahun terakhir
5. “all” untuk jangka waktu dari tahun 2004 sampai sekarang.

e. cat

Parameter ini berguna untuk mengatur kategori dari kata kunci yang akan ditampilkan hasil trennya. Parameter ini berisi kode dari kategori yang telah ditentukan oleh Google Trends. Jika parameter ini tidak mempunyai nilai, maka Google Trends akan menampilkan hasil tren untuk semua kategori.

f. gprop

Parameter ini berguna untuk mengatur sumber data dari tren yang akan ditampilkan. Sumber data yang di maksud adalah penelusuran web, penelusuran gambar, google shopping, atau penelusuran youtube. Jika parameter ini tidak digunakan maka Google Trends akan menampilkan tren yang bersumber dari penelusuran web.

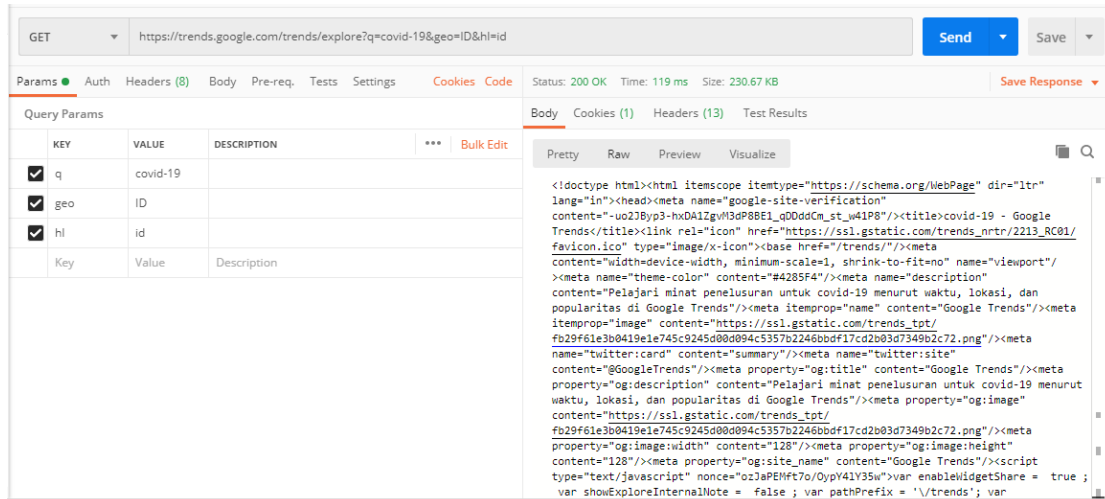
#### 4.2.2 Penentuan Teknik Scraping

Setelah melakukan proses analisis terhadap url dan *user interface* Google Trends, penulis melakukan beberapa percobaan teknik *scraping* yang mungkin dilakukan untuk mengekstrak data dari Google Trends dengan bahasa pemrograman yang penulis gunakan yaitu PHP menggunakan protokol *HTTP Request*.

a. HTML, DOM, dan XPATH

Dengan menggunakan HTML, DOM, dan XPATH penulis mencoba melakukan *parsing* terhadap *user interface* halaman hasil penelusuran (Gambar 4.5). Untuk itu penulis percobaan melakukan *HTTP Request* terhadap url halaman hasil pencarian Google Trends

menggunakan aplikasi Postman, untuk kemudian dilakukan *parsing* menggunakan teknik HTML, DOM, atau XPATH.

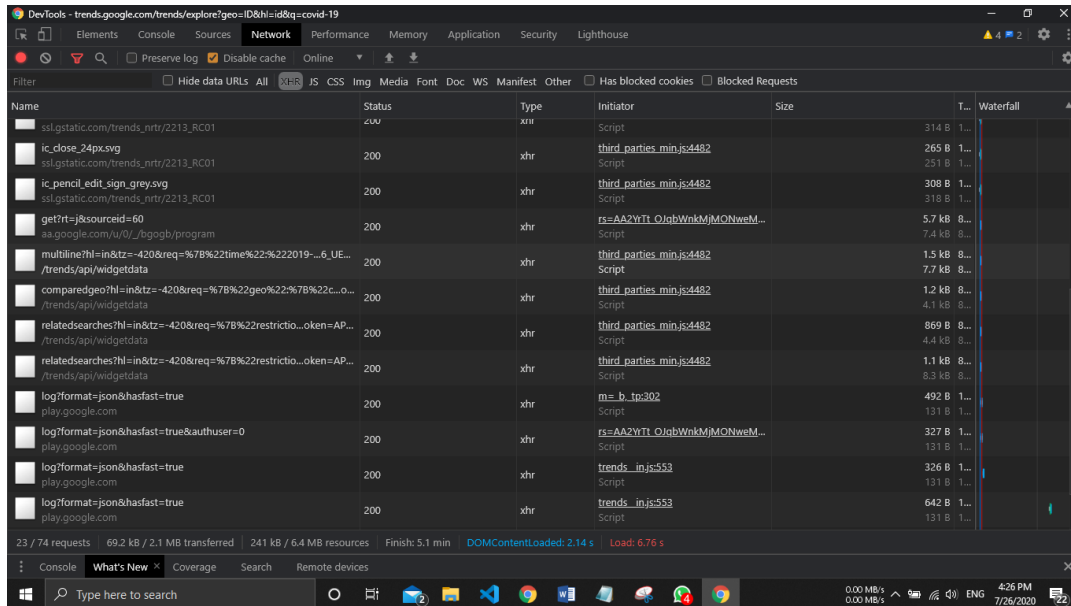


Gambar 4.6 Http request terhadap url halaman hasil penelusuran Google Trends

Dari hasil *request* yang penulis lakukan menggunakan aplikasi Postman, penulis menemukan bahwa beberapa data yang terdapat pada halaman hasil penelusuran Google Trends tidak terdapat pada *response* yang diberikan. Oleh karena itu penulis beranggapan bahwa beberapa data pada halaman hasil penelusuran ditampilkan secara *asynchronous* menggunakan JavaScript. Sehingga metode *parsing* pada halaman hasil penelusuran Google Trends menggunakan teknik HTML, DOM, atau XPATH tidak dapat dilakukan.

#### b. API

Dari fakta yang penulis temukan bahwa sebagian besar data pada halaman hasil pencarian google ditampilkan secara *asynchronous*, maka penulis melakukan analisis lalu lintas data yang terjadi saat mengakses halaman hasil penelusuran menggunakan Google Chrome.



Gambar 4.7 Tangkapan layar analisis lalu lintas data

Dari Gambar 4.7 terlihat adanya beberapa lalu lintas data dengan tipe XML Http Request (XHR). XHR merupakan objek yang terdapat pada JavaScript untuk melakukan *request* terhadap *web server* yang memungkinkan pertukaran data dengan tipe JSON, XML dan HTML. Berikut beberapa lalu lintas data yang berhasil penulis identifikasi berdasarkan fungsinya yang berguna dalam proses pembuatan aplikasi ini.

1. <https://trends.google.com/trends/api/explore>

Url ini digunakan untuk melakukan inisiasi guna mendapatkan token dan parameter yang dibutuhkan untuk mengakses url lainnya.

2. <https://trends.google.com/trends/api/widgetdata/multiline>

Url ini digunakan untuk mendapatkan data berupa *time series* untuk masing-masing kata kunci.

3. <https://trends.google.com/trends/api/explore/pickers/category>

Url ini digunakan untuk mendapatkan kategori berdasarkan bahasa yang diinginkan.

4. <https://trends.google.com/trends/api/autocomplete>

Url ini digunakan untuk mendapatkan saran kata kunci yang memiliki kesesuaian dengan kata kunci lainnya.

5. <https://trends.google.com/trends/api/widgetdata/relatedsearches>

Url ini digunakan untuk mendapatkan kata kunci yang berkaitan dengan suatu kata kunci yang lainnya.



Setelah melakukan beberapa percobaan teknik *scraping* yang akan digunakan, maka penulis menyimpulkan bahwa teknik *scraping* yang akan digunakan adalah menggunakan API. API yang dimaksud bukanlah API yang secara resmi diperuntukan untuk digunakan secara umum, sehingga dalam prosesnya terdapat beberapa fitur yang berfungsi sebagai pendukung agar progress *scraping* berjalan dengan lancar adapun fitur tersebut akan dibahas pada subab 4.2.3.

### 4.2.3 Proses Scraping

Setelah menentukan teknik *scraping* yang digunakan, penulis kemudian mengimplementasikannya ke dalam sebuah kelas dengan nama `GoogleTrend` untuk menampung fungsi-fungsi yang akan digunakan dalam aplikasi ini.

#### a. `__construct`

Fungsi konstruktor pada kelas `GoogleTrend` memiliki parameter berupa kode wilayah, kode bahasa dan `timezone`. Parameter tersebut kemudian akan disimpan ke dalam sebuah properti bernama `options`. Kode program fungsi konstruktor dapat dilihat pada Gambar 4.8.

```

14     private $options;
15
16     public function __construct($geo_code, $lang_code, $timezone)
17     {
18         $this->options = [
19             'geo' => $geo_code,
20             'hl' => $lang_code,
21             'tz' => $timezone
22         ];
23     }

```

Gambar 4.8 Fungsi konstruktor pada kelas `GoogleTrend`

Pada Gambar 4.8 baris 18, penulis menyimpan variabel parameter berupa `geo_code`, `lang_code` dan `timezone` ke dalam sebuah variabel *array* dengan nama `options`. Variabel `options` akan digunakan oleh fungsi lainnya sebagai parameter yang akan dikirimkan kepada url Google Trends.

#### b. `_getData`

Fungsi ini digunakan untuk membantu fungsi lainnya dalam melakukan *request* terhadap url Google Trend dengan menggunakan parameter url dan `params`. Kode program fungsi `_getData` dapat dilihat pada Gambar 4.9.

```

68 public function _getData($url, $params){
69     $client = new GuzzleClient();
70     $jar = new \GuzzleHttp\Cookie\FileCookieJar(storage_path('tmp').'/gtrendcookie.txt', true);
71     $response = $client->request('GET', $url, ['query' => $params, 'cookies' => $jar]);
72     if ($response->getStatusCode() == 200) {
73         return $response->getBody()->getContents();
74     }else{
75         return false;
76     }
77 }

```

Gambar 4.9 Fungsi `_getData` pada kelas `GoogleTrend`

Pada Gambar 4.9 baris 69, penulis membuat variabel `client` yang merupakan instansiasi dari kelas `GuzzleClient`. Kelas `GuzzleClient` sendiri merupakan kelas yang terdapat pada paket `Guzzle`, di mana paket `Guzzle` ini merupakan paket yang digunakan untuk melakukan `HTTP request` dalam bahasa pemrograman PHP. Variabel `client` yang berhasil dibuat akan digunakan sebagai objek untuk melakukan `request` terhadap url yang diberikan pada parameter menggunakan metode `GET`.

c. `_prepare`

Fungsi ini bertanggung jawab untuk menyiapkan token dan parameter yang akan digunakan oleh fungsi `getMultilineData` dan `getRelatedQueries`. Fungsi ini akan melakukan `request` terhadap url <https://trends.google.com/trends/api/explore> dengan menggunakan berapa parameter, seperti yang ditunjukkan pada Gambar 4.10.

```

$params = [
    'hl' => $this->options['hl'],
    'tz' => $this->options['tz'],
    'req' => json_encode([
        'comparisonItem' => [
            [
                'keyword' => $keyword,
                'geo' => $this->options['geo'],
                'time' => $time,
            ],
        ],
        'category' => $category,
    ]),
];

```

Gambar 4.10 Parameter untuk url “<https://trends.google.com/trends/api/explore>”

Adapun penjelasan dari parameter yang terdapat pada Gambar 4.10 adalah sebagai berikut:

1. `hl`, merupakan parameter yang digunakan untuk memilih bahasa yang digunakan. Parameter ini diperoleh dari properti `options` yang terdapat pada kelas `GoogleTrend`.

2. `tz`, merupakan parameter yang menandakan timezone. Parameter ini diperoleh dari properti `options` yang terdapat pada kelas `GoogleTrend`.
3. `req`, merupakan parameter yang digunakan untuk menampung parameter lainnya berupa `comparisonItem` dan kategori.
4. `comparisonItem`, adalah parameter yang digunakan untuk menampung kata kunci, wilayah dan rentang waktu yang akan dicari trennya.

Fungsi ini menerima masukan berupa `widget`, `keyword`, `category` dan `time`. Widget dimaksudkan untuk menandai dari fungsi manakah fungsi ini dijalankan, sehingga dapat memberikan hasil yang sesuai. Contoh, jika yang menjalankan fungsi ini adalah fungsi `getMultilineData` maka token dan parameter yang dikembalikan diambil dari *response* widgets dengan id “TIMESERIES”. Adapun kode program fungsi `_prepare` dapat dilihat pada Gambar 4.11.

```

25 public function prepare($widget,$keyword, $category, $time){
26     $params = [
27         'hl' => $this->options['hl'],
28         'tz' => $this->options['tz'],
29         'req' => json_encode([
30             'comparisonItem' => [
31                 [
32                     'keyword' => $keyword,
33                     'geo' => $this->options['geo'],
34                     'time' => $time,
35                 ],
36             ],
37             'category' => $category,
38         ]),
39     ];
40     $data = $this->_getData('https://trends.google.com/trends/api/explore',$params);
41     if($data){
42         $widgets = collect(json_decode(trim(substr($data, 4)), true)['widgets']);
43         $currentWidget = $widgets->where('id', $widget);
44         if($currentWidget->isNotEmpty()){
45             return [
46                 'token' => $currentWidget->first()['token'],
47                 'req' => json_encode($currentWidget->first()['request'])
48             ];
49         }
50     }
51     return false;
52 }

```

Gambar 4.11 Kode program fungsi `_prepare` kelas `GoogleTrend`

Pada Gambar 4.11 baris 40, penulis melakukan *request* terhadap url <https://trends.google.com/trends/api/explore> melalui fungsi `_getData`. Kemudian melakukan proses *parsing* terhadap respon yang didapatkan untuk mendapatkan token dan parameter yang dibutuhkan sesuai dengan kode widget yang terdapat pada parameter.

Adapun contoh respon yang telah dilakukan proses parsing dapat dilihat pada Gambar 4.12

```

array:6 [▼]
  *widgets* → array:4 [▼]
    0 → array:15 [▼]
      *request* → array:15 [▼]
      *lineAnnotationText* → "minat penelusuran"
      *bullets* → array:1 [▼]
      *showLegend* → false
      *showVerages* → false
      *helpDialog* → array:2 [▼]
      *token* → "APPG_UEAAAAA8zH7H_KZup0GvcXfkuv4NGT80IH-V2"
      *id* → "TIMESERIES"
      *type* → "fc_line_chart"
      *title* → "Minat seiring waktu"
      *template* → "fc"
      *embedTemplate* → "fc_embed"
      *version* → "1"
      *isLong* → true
      *isCurated* → false
    ]
    1 → array:18 [▼]
    2 → array:13 [▼]
      *request* → array:7 [▼]
      *helpDialog* → array:2 [▼]
      *color* → "PALETTE_COLORS"
      *keywordBase* → "covid-19"
      *token* → "APPG_UEAAAAA8zH7H_MfdrXrnfFCyrdR0WuBcozpk14"
      *id* → "RELATED_TOPICS"
      *type* → "fc_related_searches"
      *title* → "Topik terkait"
      *template* → "fc"
      *embedTemplate* → "fc_embed"
      *version* → "1"
      *isLong* → false
      *isCurated* → false
    ]
  ]
  *keywords* → array:1 [▼]
  *timeRanges* → array:1 [▼]
  *examples* → []
  *shareInt* → "Pelajari minat penelusuran untuk covid-19 menurut waktu, lokasi, dan popularitas di Google Trends"
  *shouldShowMultiHeatMapMessage* → false
}

```

Gambar 4.12 Contoh respon dari url “<https://trends.google.com/trends/api/explore>”

Pada Gambar 4.12 dapat dilihat bahwa terdapat *array* dengan indeks “widgets”, yang memiliki sejumlah data diantaranya TIMESERIES dan RELATED\_QUERIES. Dari data inilah kemudian dilakukan *parsing* untuk mendapatkan token dan parameter untuk masing-masing widget.

#### d. getCategories

Fungsi ini bertanggung jawab untuk melakukan *request* terhadap url <https://trends.google.com/trends/api/explore/pickers/category>, dan mengembalikan data yang berisikan kategori-kategori penelusuran yang terdapat pada Google Trends. Kode program fungsi `getCategories` dapat dilihat pada Gambar 4.13.

```

89     public function getCategories()
90     {
91         $params = ['hl' => $this->options['hl']];
92         $data = $this->getData('https://trends.google.com/trends/api/explore/pickers/category', $params);
93         if ($data) {
94             return json_decode(trim(substr($data, 5)), true);
95         }
96         return false;
97     }
98 }

```

Gambar 4.13 Fungsi `getCategories` pada kelas `GoogleTrend`

Pada Gambar 4.13 baris 92, penulis melakukan *request* terhadap url <https://trends.google.com/trends/api/explore/pickers/category> melalui fungsi `_getData` dan kemudian akan dilakukan parsing terhadap respon yang didapatkan.

e. `getMultilineData`

Fungsi ini bertanggung jawab untuk melakukan *request* terhadap url <https://trends.google.com/trends/api/widgetdata/multiline>, dengan menggunakan token dan parameter yang diperoleh dari fungsi `_prepare`. Kode Program fungsi `getMultilineData` dapat dilihat pada Gambar 4.14.

```

54 public function getMultilineData($keyword, $category, $time){
55     $data = $this->prepare('TIMESERIES', $keyword, $category, $time);
56     if($data){
57         $data['hl'] = $this->options['hl'];
58         $data['tz'] = $this->options['tz'];
59         $data = $this->_getData('https://trends.google.com/trends/api/widgetdata/multiline', $data);
60         if($data){
61             return json_decode(trim(substr($data, 5)), true)['default']['timelineData'];
62         }
63     }
64     return false;
65 }

```

Gambar 4.14 Fungsi `getMultilineData` pada kelas `GoogleTrend`

Pada Gambar 4.14 baris 55 penulis memanggil fungsi `_prepare` untuk mendapatkan token dan parameter yang akan digunakan untuk melakukan *request* terhadap url <https://trends.google.com/trends/api/widgetdata/multiline>. Setelah itu penulis melakukan *request* terhadap url <https://trends.google.com/trends/api/widgetdata/multiline> menggunakan token dan parameter yang telah didapatkan melalui fungsi `_prepare`. Terakhir penulis melakukan *parsing* terhadap respon yang didapatkan untuk mendapatkan data *time series*. Adapun contoh data *time series* yang didapatkan dari hasil *request* dapat dilihat pada Gambar 4.15.

```

array:153 [▼]
0 => array:6 [▼]
  "time" => "1583020800"
  "formattedTime" => "1 Mar 2020"
  "formattedAxisTime" => "1 Mar"
  "value" => array:1 [▼]
    0 => 1
  ]
  "hasData" => array:1 [▶]
  "formattedValue" => array:1 [▶]
]
1 => array:6 [▼]
  "time" => "1583107200"
  "formattedTime" => "2 Mar 2020"
  "formattedAxisTime" => "2 Mar"
  "value" => array:1 [▼]
    0 => 4
  ]
  "hasData" => array:1 [▶]
  "formattedValue" => array:1 [▶]
]
2 => array:6 [▼]
  "time" => "1583193600"
  "formattedTime" => "3 Mar 2020"
  "formattedAxisTime" => "3 Mar"
  "value" => array:1 [▼]
    0 => 3
  ]
  "hasData" => array:1 [▶]
  "formattedValue" => array:1 [▶]
]
3 => array:6 [▶]
4 => array:6 [▼]
  "time" => "1583366400"
  "formattedTime" => "5 Mar 2020"
  "formattedAxisTime" => "5 Mar"
  "value" => array:1 [▶]
  "hasData" => array:1 [▶]
  "formattedValue" => array:1 [▶]
]
5 => array:6 [▶]
6 => array:6 [▶]
7 => array:6 [▶]
8 => array:6 [▶]
9 => array:6 [▶]
10 => array:6 [▶]

```

Gambar 4.15 Hasil fungsi `getMultilineData` kata kunci “`covid-19`”

Pada Gambar 4.15 dapat dilihat bahwa fungsi `getMultilineData` mengembalikan nilai dengan tipe `array` untuk merepresentasikan data *time series* dari Google Trends.

#### f. `getRelatedQueries`

Fungsi ini bertanggung jawab untuk melakukan *request* terhadap url “`https://trends.google.com/trends/api/widgetdata/relatedsearches`”, dengan menggunakan token dan parameter yang diperoleh dari fungsi `_prepare`. Kode Program fungsi `getRelatedQueries` dapat dilihat pada Gambar 4.16.

```

88     public function getCategories()
89     {
90         $params = ['hl' => $this->options['hl']];
91         $data = $this->getData('https://trends.google.com/trends/api/explore/pickers/category', $params);
92         if ($data) {
93             return json_decode(trim(substr($data, 5)), true);
94         }
95         return false;
96     }

```

Gambar 4.16 Fungsi `getRelatedQueries` pada kelas `GoogleTrend`

Sama halnya dengan fungsi `getMultilineData`, fungsi `getRelatedQueries` juga membutuhkan token dan parameter yang akan digunakan untuk melakukan *request* terhadap url

<https://trends.google.com/trends/api/widgetdata/relatedsearches>. Sehingga pada Gambar 4.16 baris 91, fungsi ini, terlebih dahulu memanggil fungsi `getRelatedQueries` untuk mendapatkan token dan parameter yang dibutuhkan.

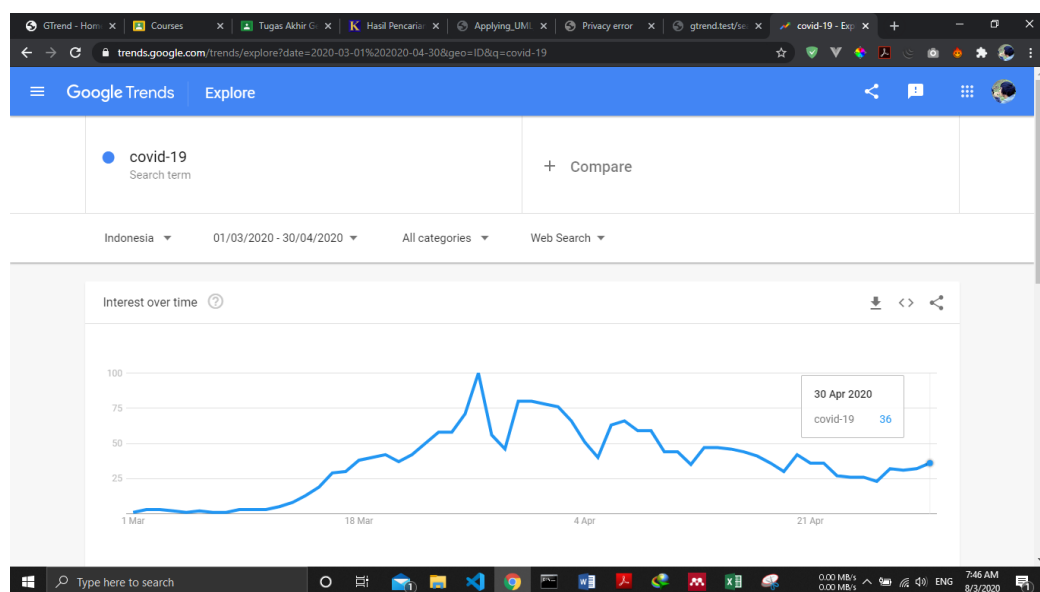
g. `getKeywordSuggestion`

Fungsi ini bertujuan untuk menampilkan hasil pencarian yang mungkin berkaitan dengan kata kunci yang dimasukan oleh pengguna, sehingga diharapkan dapat membantu proses analisis korelasi. Fungsi ini akan melakukan *request* terhadap url <https://trends.google.com/trends/api/autocomplete>. Fungsi ini memiliki satu parameter yaitu `keyword`.

Setelah berhasil mengimplementasikan teknik *scraping* ke dalam kelas `GoogleTrend`, kemudian penulis mulai menyusun skema *scraping* yang akan dijalankan. Mengingat bahwa url yang digunakan pada fungsi `getMultilineData` memiliki keterbatasan dalam mengembalikan data. Keterbatasan yang dimaksud adalah data harian hanya dapat diakses dengan minimal jangka waktu dua hari dan maksimal delapan bulan. Sehingga untuk data faktual yang mungkin memiliki jangka waktu lebih dari delapan bulan, maka proses *scraping* harus dilakukan beberapa kali. Berikut akan dijelaskan skema *scraping* yang dimaksud beserta contohnya:

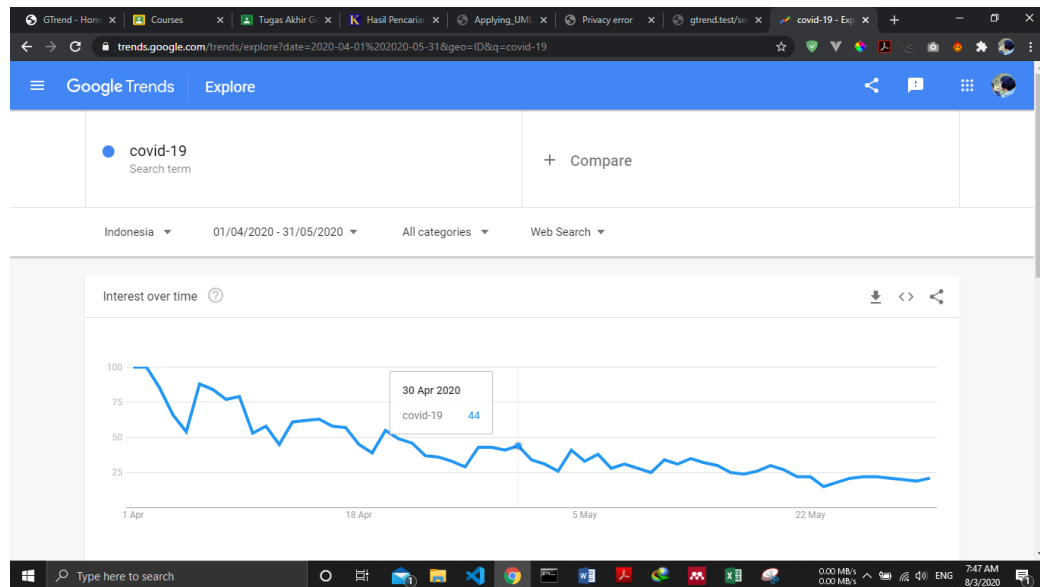
- a. Sistem akan mengidentifikasi tanggal terendah dan tanggal tertinggi dari himpunan data faktual. Kemudian sistem akan membulatkan tanggal terendah menjadi tanggal satu di bulan yang sama dan tanggal terendah menjadi tanggal terakhir di bulan yang sama. Setelah didapatkan hasil pembulatan tanggal mulai dan selesai dilanjutkan dengan membagi tanggal mulai dan tanggal selesai ke dalam periode delapan bulannan dengan irisan empat bulan. Contoh, himpunan data faktual memiliki tanggal terendah 3 Januari 2010 dan data tanggal tertinggi 12 Desember 2010, maka periode yang dihasilkan adalah 1 Januari 2010 – 31 Agustus 2010 dan 1 Mei 2010 – 31 Desember 2010, dengan irisan pada 1 Mei 2010 – 31 Agustus 2010. Untuk skenario covid-19 periode yang dihasilkan hanya ada 1 yaitu tanggal 1 Maret 2020 – 31 Oktober 2020. Periode yang telah ditemukan akan digunakan untuk proses *request* yang ada terdapat pada fungsi `_getMultilineData` pada kelas `GoogleTrends`.
- b. Sistem akan melakukan *request* untuk masing-masing kata kunci berdasarkan periode yang telah diidentifikasi sebelumnya. Untuk periode kedua dan seterusnya, hasil *request* akan melalui tahap normalisasi terlebih dahulu terhadap perbandingan jumlah nilai yang

termasuk dalam irisan dengan periode sebelumnya. Hal ini dilakukan karena Google Trends menghasilkan data yang diskalakan dari 0 – 100 untuk setiap *request* yang diterima. Contoh, pada Gambar 4.17 merupakan penelusuran pada rentang waktu 1 Maret 2020 – 30 April 2020 dan Gambar 4.18 merupakan penelusuran pada rentang waktu 1 April 2020 – 31 Mei 2020 untuk kata kunci yang sama yaitu covid-19. Dapat dilihat bahwa pada tanggal 30 April 2020 untuk penelusuran dengan rentang waktu 1 Maret 2020 – 30 April 2020 memiliki nilai 36 sedangkan untuk rentang waktu 1 April 2020 – 31 Mei 2020 memiliki nilai 44.



Gambar 4.17 Grafik penelusuran 1 Maret 2020 – 30 April 2020





Gambar 4.18 Grafik penelusuran 1 April 2020 – 31 Mei 2020

Dari fakta di atas, maka jika terdapat periode lebih dari 1 periode maka periode kedua dan seterusnya akan dilakukan proses normalisasi terlebih dahulu. Contoh, terdapat dua periode yaitu 1 Januari 2010 – 31 Agustus 2010 dan 1 Mei 2010 – 31 Desember 2010 maka *request* pada periode 1 Mei 2010 – 31 Desember 2010 akan dinormalisasikan terhadap perbandingan jumlah nilai yang termasuk pada tanggal 1 Mei 2010 - 31 Agustus 2010 dengan periode 1 Januari 2010 – 31 Agustus 2010. Untuk skenario covid-19 tidak diperlukan adanya normalisasi dikarenakan hanya terdapat satu periode.

- c. Untuk setiap kata kunci setelah proses *scraping* dilakukan terhadap semua periode maka akan dilakukan proses pemetaan terhadap data faktual. Pemetaan yang dimaksud adalah dengan melakukan proses pencarian tanggal yang termasuk dalam tanggal mulai dan tanggal selesai masing-masing data yang terdapat pada data faktual. Setelah didapatkan data-data yang sesuai maka akan dilakukan proses penjumlahan jika data terdapat lebih dari 1 (berlaku untuk data faktual dengan interval waktu lebih dari 1 hari) yang kemudian akan disimpan ke dalam basisdata. Contoh, data faktual memiliki data dengan interval waktu tiga hari yaitu pada tanggal 1 Januari 2010 sampai 3 Januari 2010, maka akan dilakukan pemetaan terhadap data *time series* dari Google Trends yang termasuk dalam tanggal 1 Januari 2010 sampai tanggal 3 Januari 2010 yaitu tanggal 1, 2, dan 3 Januari 2010. Nilai dari ketiga tanggal tersebut akan dijumlahkan dan hasilnya menjadi nilai dari tanggal 1 Januari 2010 sampai 3 Januari 2010.

- d. Setelah itu dilanjutkan dengan memanggil fungsi `getRelatedQueries` untuk mencari kata kunci yang berkaitan dengan setiap kata kunci untuk rentang waktu dari tanggal terendah sampai tanggal tertinggi yang terdapat pada himpunan data faktual.

#### 4.2.4 Fitur Pendukung

Pada proses implemetasi *scraping* dengan teknik API, penulis menemukan kendala terkait url yang digunakan, mengingat url tersebut peruntukannya untuk penggunaan dengan intensitas normal. Sedangkan pada aplikasi ini tingkat permintaan data terhadap *web server* cenderung tinggi, sehingga dapat memicu server untuk memberikan *response 429 (Too Many Request)*. Untuk dapat mengatasi masalah tersebut, penulis menerapkan beberapa fitur pendukung yaitu:

- a. Sistem antrian (*queue*)

Sistem antrian pada aplikasi ini diterapkan untuk mengatur jumlah *request* terhadap url Google Trends. Di mana semua permintaan analisis korelasi dari pengguna akan diproses satu-persatu, sehingga diharapkan akan mengurangi kemungkinan *web server* merespon dengan kode 429 (*Too Many Request*). Sistem antrian diimplementasikan dengan menggunakan Laravel Queue dan Supervisor.

- b. Proxy Rotator

Proxy adalah suatu mekanisme yang memungkinkan suatu komputer mengakses jaringan internet menggunakan alamat dari komputer lain. Sedangkan Proxy Rotator adalah sebuah layanan yang menyediakan kumpulan alamat proxy server dan dapat diakses melalui satu alamat proxy saja, sehingga pada setiap *request* yang dikirimkan akan menggunakan alamat yang berbeda-beda. Penulis mencoba untuk memanfaatkan layanan proxy rotator dari <https://www.webshare.io/> yang menawarkan masa percobaan selama satu bulan. Dengan ketentuan sebagai berikut:

1. Maksimal *bandwidth* 100 gigabytes
2. Proxy Server yang dapat digunakan berjumlah 10 alamat.
3. Proxy Server yang disediakan merupakan *shared* Proxy Server, sehingga kecepatan akses sangat bergantung pada banyaknya pengguna yang menggunakan alamat tersebut dalam waktu yang bersamaan.

Adapun implementasi Proxy Rotator dapat dilihat pada Gambar 4.19.

```

68     public function _getData($url, $params){
69         $client = new GuzzleClient();
70         $jar = new \GuzzleHttp\Cookie\FileCookieJar(storage_path('tmp').'/gtrendcookie.txt' ,true);
71         $response = $client->request(
72             'GET',
73             $url,
74             [
75                 'query' => $params,
76                 'cookies' => $jar,
77                 'proxy' => 'http://edswurpo-GB-MX-NL-rotate:qh30oorwzasa@45.9.123.11:80'
78             ]);
79         if ($response->getStatusCode() == 200) {
80             return $response->getBody()->getContents();
81         }else{
82             return false;
83         }
84     }

```

Gambar 4.19 Implementasi proxy rotator

Pada Gambar 4.19 baris 77 penulis menambahkan parameter berupa proxy dengan nilai <http://edswurpo-GB-MX-NL-rotate:qh30oorwzasa@45.9.123.11:80>. Adapun penjelasan dari nilai tersebut adalah sebagai berikut:

1. edswurpo-GB-MX-NL-rotate, merupakan nama pengguna yang digunakan dalam proses autentikasi proxy
2. qh30oorwzasa, merupakan kata kunci yang digunakan yang digunakan dalam proses autentikasi proxy
3. 45.9.123.11, alamat ip dari proxy server
4. 80, port yang digunakan oleh proxy server.

### 4.3 Perhitungan Korelasi

Ukuran korelasi yang diterapkan pada aplikasi ini adalah korelasi pearson, di mana korelasi ini digunakan untuk mengukur kekuatan dan arah hubungan linear antara dua variabel. Dua variabel pada aplikasi ini merupakan data faktual dan data Google Trends untuk masing-masing kata kunci. Korelasi pearson akan menghasilkan koefisien korelasi yang berada di antara  $-1 < 0 < 1$  yang dapat diartikan sebagai berikut:

- a. Jika koefisien korelasi bernilai -1 maka dapat diartikan bahwa hubungan dari kedua variabel sangat kuat namun bersifat negatif atau berlawanan.
- b. Jika koefisien korelasi bernilai 0 maka dapat diartikan bahwa kedua variabel tidak memiliki hubungan.
- c. Jika koefisien korelasi bernilai 1 maka dapat diartikan bahwa hubungan dari kedua variabel sangat kuat dan positif atau searah.

Adapun rumus korelasi pearson dapat dilihat pada persamaan (4.1)

$$\text{koefisien korelasi} = \frac{n \sum_{i=1}^n x_i y_i - \sum_{i=1}^n x_i \sum_{i=1}^n y_i}{\sqrt{n \sum_{i=1}^n x_i^2 - (\sum_{i=1}^n x_i)^2} \sqrt{n \sum_{i=1}^n y_i^2 - (\sum_{i=1}^n y_i)^2}} \quad (4.1)$$

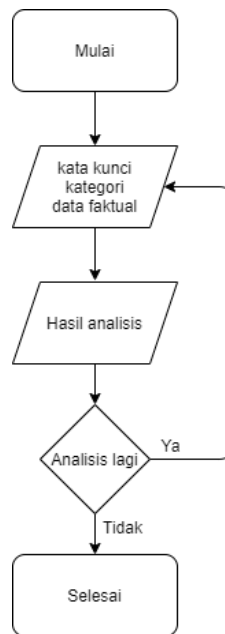
$n$  = jumlah data

$x$  = nilai data faktual

$y$  = nilai data Google Trends

#### 4.4 Implementasi Website

Aplikasi alat bantu analisis korelasi data faktual dan Google Trends adalah aplikasi berbasis web yang dikembangkan menggunakan *framework* Laravel. Laravel adalah *framework* yang mengimplementasikan konsep *Model View Controller* (MVC). MVC adalah konsep di mana aplikasi dibagi menjadi tiga bagian utama yaitu *Model*, *View* dan *Controller*. *Model* sendiri adalah bagian yang terhubung dengan basisdata, sedangkan *View* adalah bagian yang bertanggung jawab untuk menampilkan dan menerima data. *Model* dan *View* dihubungkan melalui *Controller* yang bertugas untuk mengatur jalannya proses bisnis yang terjadi pada sebuah aplikasi. Pada bagian ini, akan dijelaskan *Model*, *View* dan *Controller* yang terdapat pada aplikasi ini untuk mendukung penggunaan website analisis korelasi data faktual dan Google Trends. Adapun alur penggunaan website ini secara umum dapat dilihat pada Gambar 4.20.



Gambar 4.20 Alur penggunaan website analisis korelasi data faktual dan Google Trends

#### 4.4.1 Model

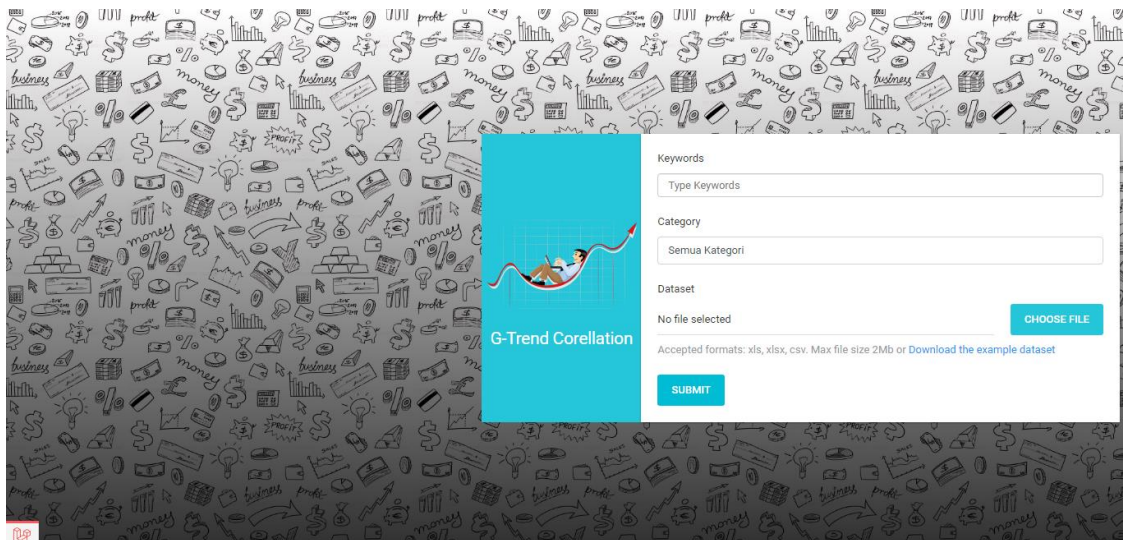
Pada aplikasi ini hanya terdapat satu *model* saja dengan nama *queue*. *Model queue* ini bertugas untuk menyimpan data yang berkaitan saat proses *scraping* berjalan. Adapun data yang disimpan yaitu:

- id, digunakan untuk identifikasi.
- data, digunakan untuk menyimpan data faktual dan data dari google trend yang telah berhasil didapatkan berdasarkan kategori tertentu.
- category, digunakan untuk menyimpan kode kategori yang dipilih oleh pengguna.
- status, digunakan untuk menyimpan status pada proses *scraping*.
- fetches\_keywords, digunakan untuk menandai kata kunci yang telah dilakukan proses *scraping* berdasarkan kategori tertentu.
- keywords, digunakan untuk menyimpan kumpulan kata kunci yang di isi oleh pengguna.
- error\_message, dikarenakan pada aplikasi ini proses *scraping* dijalankan secara *asynchronous*, maka pesan kesalahan pada proses *scraping* disimpan terlebih dahulu sebelum kemudian ditampilkan kembali kepada pengguna.

#### 4.4.2 View

Pada aplikasi alat bantu analisis korelasi data faktual dan Google Trends ini terdapat tiga halaman utama yaitu Home, Progress, dan Result. Adapun penjelasan dan tampilan dari masing-masing halaman adalah sebagai berikut:

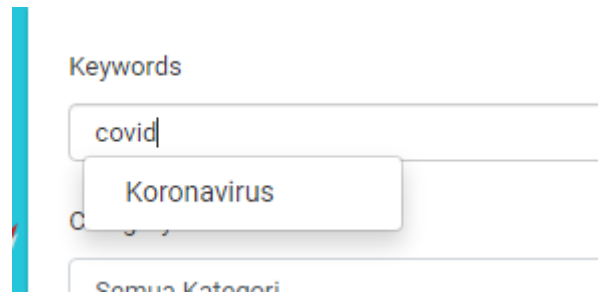
##### a. Home



Gambar 4.21 Tampilan halaman home

Pada Gambar 4.21 dapat dilihat bahwa pada halaman home ini terdapat formulir untuk melakukan analisis korelasi data faktual dan Google Trends. Formlir tersebut terdiri dari masukan berupa kata kunci, kategori dan dataset. Masukan kata kunci pada halaman ini dapat berisi kata kunci dengan jumlah maksimal 10 kata kunci. Kata kunci yang dimaksudkan bersifat *case-sensitive* yang berarti besar kecilnya huruf pada suatu kata kunci akan mempengaruhi data dan hasil yang akan didapatkan.

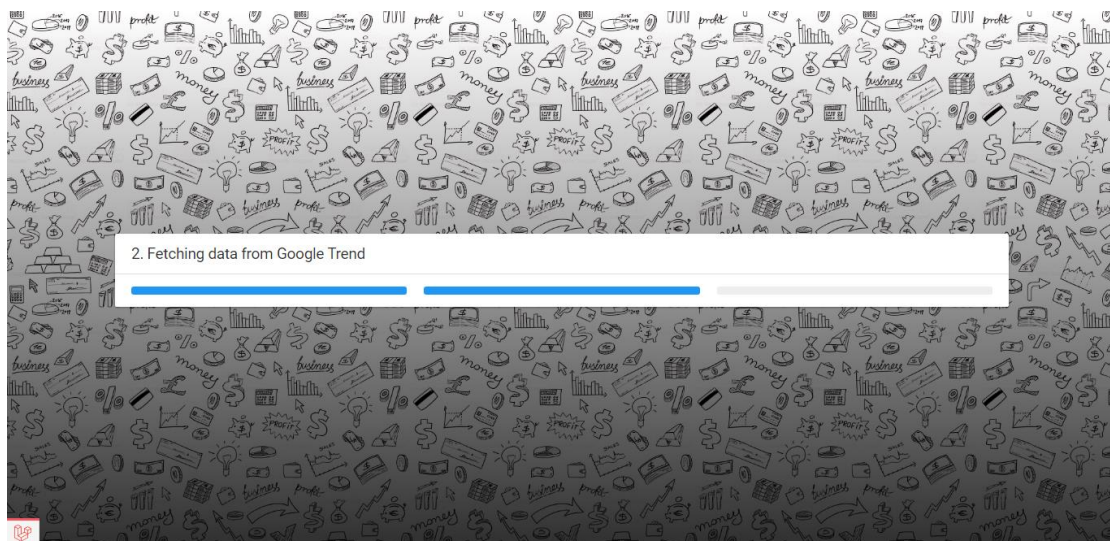
Pada halaman ini juga terdapat fitur sugesti kata kunci, di mana fitur tersebut akan memberikan sugesti untuk kata kunci yang di isi oleh pengguna pada formulir analisis korelasi. Diharapkan dengan adanya fitur ini pengguna dapat mempunyai pilihan ataupun memberikan kata kunci alternatif yang mungkin tidak terpikirkan oleh pengguna sehingga dapat membantu dalam proses analisis korelasi. Adapun kata kunci yang ditampilkan diambil dari url “<https://trends.google.com/trends/api/autocomplete>” melalui fungsi `getKeywordSuggestion` yang terdapat pada kelas `GoogleTrend`. Adapun contoh tampilan fitur sugesti dapat dilihat pada Gambar 4.22.



Gambar 4.22 Contoh tampilan fitur sugesti

Adapun penggunaan fitur sugesti kata kunci seperti pada Gambar 4.22 dapat digunakan dengan cara mengisi kolom kata kunci dengan minimal satu karakter. Jika Google Trends memiliki sugesti kata kunci untuk karakter yang di isi pada kolom kata kunci, maka sistem akan menampilkan daftar kata kunci yang dapat dipilih oleh pengguna.

#### b. Progress

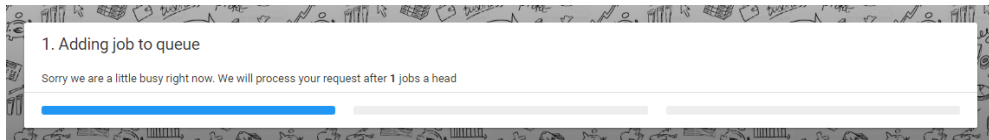


Gambar 4.23 Halaman progress

Halaman ini merupakan halaman yang akan menginformasikan progress *scraping* yang sedang dilakukan. Terdapat tiga tahapan informasi *scraping* yang akan ditampilkan secara berurutan, yaitu:

##### 1. Berhasil ditambahkan dalam antrian

Status ini ditampilkan ketika permintaan analisis korelasi oleh pengguna telah berhasil ditambahkan ke dalam antrian untuk diproses. Halaman ini juga akan menampilkan jumlah antrian yang sedang ditunggu seperti pada Gambar 4.24.

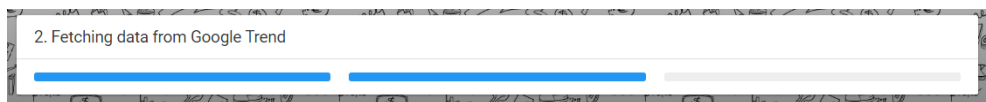


Gambar 4.24 Status saat menunggu antrian

Dalam menampilkan jumlah antrian, halaman ini akan melakukan *request* yang akan diterima oleh fungsi jobs pada *controller* ApplicationController.

2. Sedang melakukan *scraping*

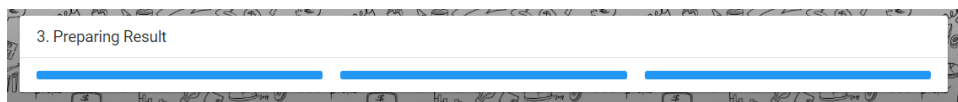
Status ini ditampilkan ketika sistem sedang melakukan proses *scraping* terhadap permintaan dari pengguna, seperti pada Gambar 4.25.



Gambar 4.25 Status pada saat proses *scraping*

3. *Scraping* selesai sedang mempersiapkan hasil.

Status ini akan ditampilkan ketika proses *scraping* yang dibutuhkan telah selesai dilakukan. Halaman ini akan mengarahkan pengguna menuju halaman result, seperti pada Gambar 4.26.



Gambar 4.26 Status saat selesai proses *scraping*

c. Result

Halaman ini akan ditampilkan ketika proses *scraping* telah selesai dilakukan. Halaman ini juga bertugas untuk melakukan perhitungan korelasi dari data. Perhitungan korelasi secara *client side* dipilih untuk meringankan tugas server, sehingga diharapkan server dapat bejalan secara maksimal. Pada halaman ini akan ditampilkan informasi mengenai hasil yang telah didapatkan, yaitu:

1. Grafik

Grafik pada halaman ini akan menampilkan data faktual dan data Google Trends untuk masing-masing kata kunci. Data yang ditampilkan adalah data yang telah



diskalakan dari rentang 0 – 100. Diharapkan dengan adanya grafik ini, pengguna dapat mengetahui perbedaan maupun kesamaan antara data faktual dan Google Trends secara lebih baik.

## 2. Hasil Korelasi

Hasil korelasi ditampilkan berdasarkan kata kunci dan dapat diatur sesuai dengan rentang waktu yang terdapat pada data faktual melalui fitur *date range slider*.

## 3. Kueri Terkait

Kueri terkait ditampilkan berdasarkan kata kunci dan dapat digunakan sebagai kata kunci yang akan dilakukan analisis korelasinya kembali.

Pada halaman ini pengguna juga dapat melakukan analisis korelasi kembali dengan memasukan kata kunci baru maupun memilih kata kunci yang terdapat pada kueri terkait dengan menggunakan data faktual yang baru atau tetap menggunakan data faktual yang digunakan sebelumnya. Adapun contoh tampilan halaman result untuk data faktual covid-19 dapat dilihat pada Gambar 4.27.

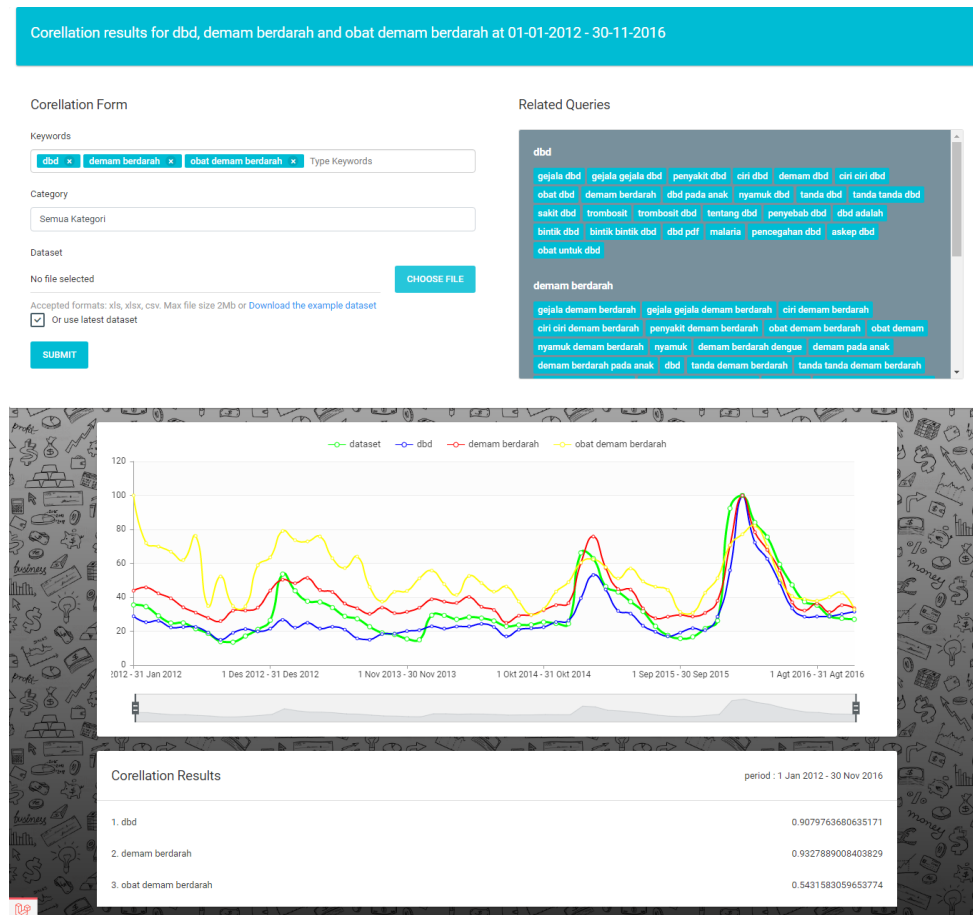


Gambar 4.27 Halaman result

Pada Gambar 4.27 dapat dilihat bahwa hasil korelasi untuk data faktual covid-19 terhadap data Google Trends untuk kata kunci covid-19, corona dan vaksin corona pada kategori “semua kategori” bernilai negatif, maka dapat ditarik kesimpulan bahwa koefisien korelasi bernilai negatif atau berlawanan. Jika diperhatikan pada Gambar 4.27, data tren pada

Google Trends mengalami kenaikan signifikan pada awal bulan Maret 2020 di mana pemerintah baru saja mengumumkan bahwa telah ditemukan pasien yang positif terpapar virus covid-19 untuk pertama kalinya di Indonesia. Kemudian sekitar akhir Maret 2020 tren cenderung menurun hingga akhir bulan Juli 2020. Penulis berasumsi bahwa turunnya tren pencarian untuk kata kunci yang berkaitan dengan “covid-19” salah satunya dikarenakan oleh kecenderungan masyarakat Indonesia yang mulai terbiasa menghadapi covid-19 bahkan ada beberapa yang mulai acuh terbukti dengan masih ada masyarakat Indonesia yang tidak memakai masker pada saat keluar rumah.

Untuk dapat memberikan contoh hasil untuk korelasi positif kepada pembaca, penulis mencoba menggunakan data faktual lainnya yaitu data faktual pasien penyakit demam berdarah di Indonesia yang didapatkan dari penelitian (Husnayain et al., 2019). Data faktual ini memiliki rentang waktu mulai dari tanggal 1 Januari 2012 sampai 30 November 2016 dengan interval waktu satu bulan. Data faktual ini akan penulis gunakan untuk korelasi analisis dengan tren pencarian yang terdapat pada Google Trends untuk kata kunci dbd, demam berdarah, obat demam berdarah. Adapun hasil analisis korelasi untuk data faktual demam berdarah dapat dilihat pada Gambar 4.28.



Gambar 4.28 Hasil analisis korelasi data faktual demam berdarah

Dari hasil analisis yang terdapat pada Gambar 4.28 dapat disimpulkan bahwa data tren yang terdapat pada Google Trends untuk kata kunci dbd, demam berdarah dan obat demam berdarah memiliki koefisien korelasi positif, sehingga data tren pada Google Trends untuk kata kunci dbd, demam berdarah dan obat demam berdarah dapat digunakan sebagai data pendamping data faktual demam berdarah dalam sebuah penelitian.

#### 4.4.3 Controller

Pada aplikasi alat bantu analisis korelasi data faktual dan Google Trends ini penulis membuat satu *controller* yaitu *AppController*. *Controller* ini bertanggung jawab atas jalannya aplikasi ini. Adapun fungsi yang terdapat pada *AppController* adalah sebagai berikut:

a. index

Fungsi ini bertugas untuk menyiapkan data berupa data kategori, yang kemudian akan diarahkan menuju *view* dengan nama Home.

b. search

Fungsi ini bertugas untuk memvalidasi masukan dari pengguna seperti kata kunci, kategori, dan data faktual yang dikirim melalui halaman home maupun halaman result, dengan skema sebagai berikut:

1. Halaman home

Jika permintaan berasal dari halaman home, maka sistem akan membuat data baru pada basisdata menggunakan *model* queue. Kemudian akan memasukan data queue tersebut ke dalam sistem antrian. Terakhir sistem akan mengarahkan pengguna ke halaman result.

2. Halaman result

Jika permintaan berasal dari halaman result, maka sistem akan melihat apakah pengguna menggunakan data faktual baru atau tetap menggunakan data faktual lama. Jika pengguna menggunakan data faktual baru maka sistem akan menghapus data pada kolom data yang terdapat pada *model* queue dan melakukan proses validasi data faktual seperti pada bagian 4.1.2. Sedangkan jika pengguna tetap menggunakan data faktual yang lama, sistem akan melakukan penyaringan kata kunci yang pernah dilakukan proses *scraping* berdasarkan kategorinya. Jika terdapat kata kunci yang belum pernah dilakukan *scraping* sebelumnya, maka sistem akan memasukan queue tersebut ke dalam antrian. Jika semua kata kunci sudah pernah dilakukan *scraping* maka sistem akan mengarahkan pengguna menuju halaman result.

c. Progress

Fungsi ini akan bertugas untuk menampilkan halaman progress, dengan terlebih dahulu melakukan validasi terhadap kolom status yang ada pada data queue. Adapun validasi tersebut adalah untuk menjalankan skenario sebagai berikut:

1. Jika status bernilai 0 yang berarti terdapat kesalahan yang terjadi pada proses *scraping* maka sistem akan mengarahkan pengguna menuju halaman home dengan menampilkan pesan kesalahan yang disimpan kolom *error\_message*.
2. Jika status bernilai 1 atau 2 yang berarti proses *scraping* masih berjalan. Kemudian sistem akan menampilkan halaman progress.
3. Jika status bernilai 3 yang berarti proses *scraping* telah selesai, maka sistem akan mengarahkan pengguna menuju halaman result.

d. Result

Fungsi ini akan bertugas untuk menampilkan halaman result hanya jika status pada data queue bernilai 3.

e. Jobs

Fungsi ini akan bertugas menerima *request* dari halaman progress untuk mengetahui status dari proses *scraping*.

## 4.5 Pengujian

Setelah semua tahap implementasi dilakukan maka penulis melakukan beberapa pengujian yaitu pengujian fungsionalitas dan pengujian usability untuk mengetahui bahwa sistem dapat berjalan sesuai dengan apa yang direncanakan yaitu untuk melakukan analisis korelasi data faktual dan Google Trends untuk *monitoring / surveillance*.

### 4.5.1 Pengujian Fungsionalitas

Pada tahap ini, penulis melakukan pengujian terhadap fungsionalitas sistem. Pengujian dilakukan dengan membuat skenario pengujian dan hasil yang diharapkan. Apabila sistem dapat memberikan hasil seperti hasil yang diharapkan maka pengujian akan bernilai valid. Adapun hasil dari skenario pengujian dapat dilihat pada Tabel 4.1.

Tabel 4.1 Hasil pengujian fungsionalitas

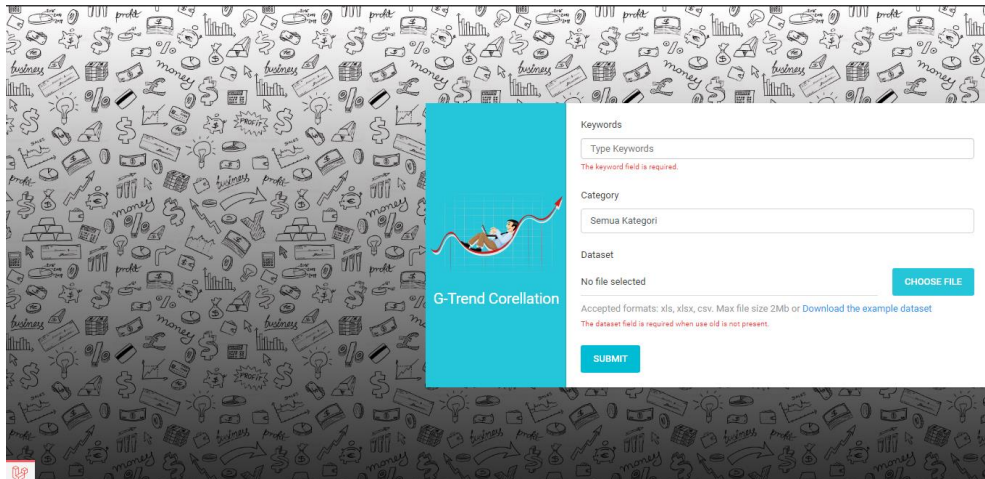
No	Skenario Pengujian	Tes Kasus	Hasil yang diharapkan	Response sistem	Hasil
1	Mengkosongkan semua formulir kemudian klik cari	Kata kunci : - Kategori : “semua kategori” Time series : -	Sistem akan mengarahkan ke halaman home dengan menampilkan pesan kesalahan	Sistem mengarahkan ke halaman home dan menampilkan pesan kesalahan	Valid
2	Mengkosongkan kata kunci dengan mengisi data faktual yang ber-format Microsoft Excel	Kata kunci : - Kategori : “semua kategori” Time series : *.xls	Sistem akan mengarahkan ke halaman home dengan menampilkan pesan kesalahan	Sistem mengarahkan ke halaman home dan menampilkan pesan kesalahan	Valid
3	Mengisi semua formulir, namun data faktual bukan merupakan format Microsoft Excel	Kata kunci : DBD Kategori : “semua kategori” Time series : *.pdf	Sistem akan mengarahkan ke halaman home dengan menampilkan pesan kesalahan	Sistem mengarahkan ke halaman home dan menampilkan pesan kesalahan	Valid
4	Mengisi semua formulir, namun data faktual tidak sesuai <i>template</i> yang dimaksud seperti pada Gambar 4.1	Kata kunci : DBD Kategori : “semua kategori” Time series : *.xls	Sistem akan mengarahkan ke halaman home dengan menampilkan pesan kesalahan	Sistem mengarahkan ke halaman home dan menampilkan pesan kesalahan	Valid

5	Mengisi semua formulir dengan kata kunci berupa simbol	Kata kunci: !@#%\$^&*() Kategori : “semua kategori” Time series : *.xlsx	Sistem tetap memproses permintaan, namun menampilkan pesan bahwa sistem tidak berhasil mendapatkan data dengan kata kunci yang dimaksud.	Sistem mengarahkan ke halaman result dan menampilkan pesan bahwa tidak mendapatkan data yang cukup untuk kata kunci !@#%\$^&*()	Valid
6	Mengisi semua formulir, dengan kata kunci lebih dari satu	Kata kunci: DBD, OBAT DBD Kategori : “semua kategori” Time series : *.xls	Sistem memproses permintaan dan menampilkan korelasi berdasarkan kedua kata kunci	Sistem mengarahkan ke halaman result dan menampilkan hasil kata kunci untuk dbd dan obat dbd	Valid
7	Membuat 2 permintaan secara berurutan dengan rentang waktu yang bersamaan	Kata kunci : DBD Kategori : “semua kategori” Time series : *.xls	Sistem akan menampilkan pesan untuk permintaan yang kedua pada halaman progress dengan menginformasikan jumlah antrian di depannya	Sistem menampilkan pesan menunggu antrian yaitu 1 antrian.	Valid
8	Memilih kata kunci yang pada kueri yang berkaitan yang terdapat di halaman result	Kata kunci : - Kategori : “semua kategori” Time series : -	Sistem memproses permintaan dan menampilkan hasil analisis	Sistem menampilkan hasil korelasi untuk kata kunci yang dipilih dari kueri terkait	Valid
9	Menggeser <i>date range selector</i> pada halaman hasil result	Kata kunci : - Kategori : “semua kategori” Time series : -	Sistem akan menampilkan hasil korelasi sesuai dengan waktu yang dipilih	Sistem menampilkan hasil korelasi berdasarkan rentang waktu yang dipilih	Valid

Adapun penjelasan detail mengenai skenario pengujian pada Tabel 4.1 adalah sebagai berikut:

a. Skenario nomor 1

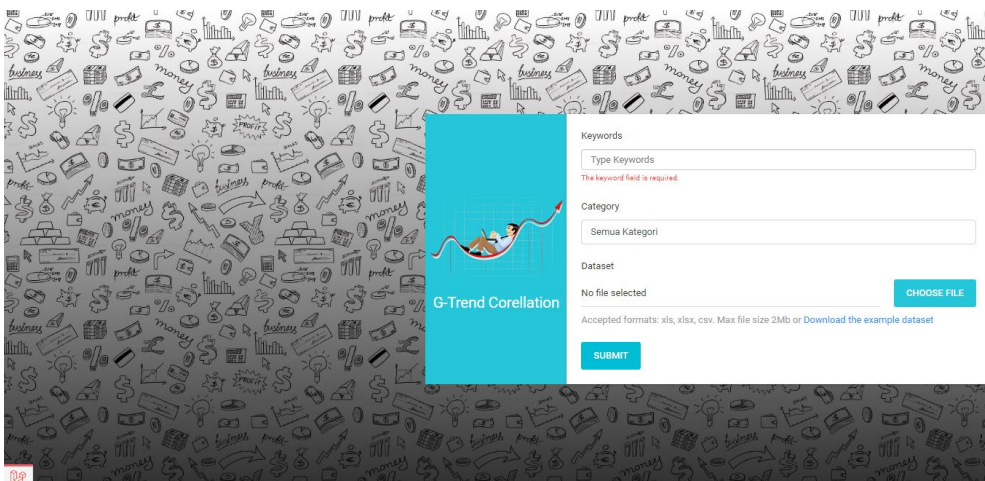
Skenario ini dimaksudkan untuk mengetahui respon aplikasi jika pengguna melakukan analisis tanpa memberikan masukan sama sekali. Dalam pengujian ini ditemukan bahwa aplikasi kembali mengarahkan pengguna menuju halaman home dengan menampilkan pesan kesalahan, seperti pada Gambar 4.29.



Gambar 4.29 Skenario nomor 1

## b. Skenario nomor 2

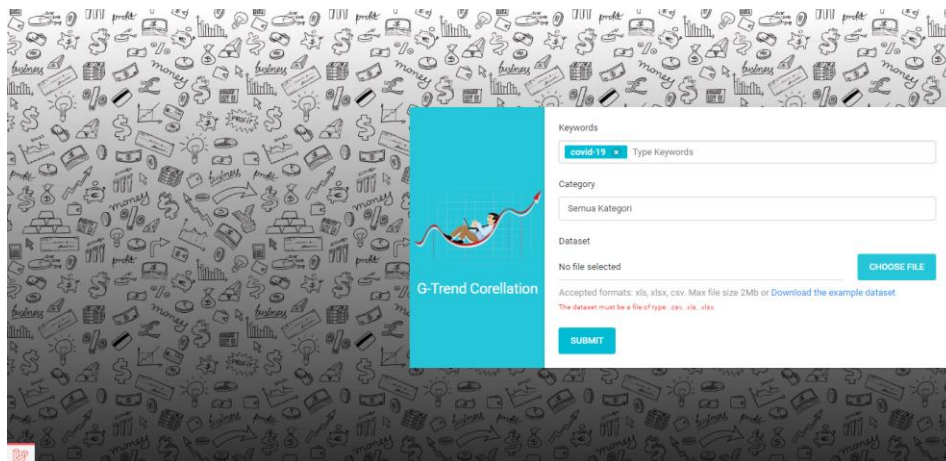
Skenario ini dimaksudkan untuk mengetahui respon aplikasi jika pengguna melakukan analisis dengan hanya memasukan data faktual tanpa memberikan kata kunci. Dalam pengujian ini ditemukan bahwa aplikasi kembali mengarahkan pengguna menuju halaman home dengan menampilkan pesan kesalahan, seperti pada Gambar 4.30.



Gambar 4.30 Skenario nomor 2

## c. Skenario nomor 3

Skenario ini dimaksudkan untuk mengetahui respon aplikasi jika pengguna melakukan analisis dengan memasukan data faktual yang memiliki format pdf. Dalam pengujian ini ditemukan bahwa aplikasi kembali mengarahkan pengguna menuju halaman home dengan menampilkan pesan kesalahan, seperti pada Gambar 4.31.



Gambar 4.31 Skenario nomor 3

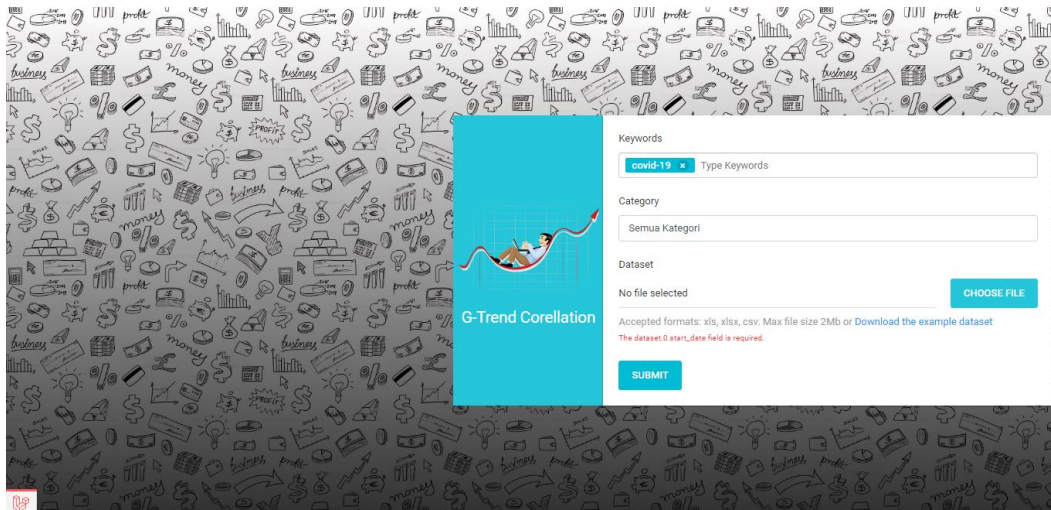
## d. Skenario nomor 4

Skenario ini dimaksudkan untuk mengetahui respon aplikasi jika pengguna melakukan analisis dengan memasukan data faktual dengan tidak mengindahkan *template* yang telah ditentukan pada bagian 4.1.1. Dalam pengujian ini ditemukan bahwa aplikasi kembali mengarahkan pengguna menuju halaman home dengan menampilkan pesan kesalahan. Adapun *template* yang digunakan seperti Gambar 4.32 dan pesan kesalahan pada Gambar 4.33.

	A	B	C	D	E	F
1			3/2/2020	3/2/2020	2	
2			3/3/2020	3/3/2020	0	
3			3/4/2020	3/4/2020	0	
4			3/5/2020	3/5/2020	0	
5			3/6/2020	3/6/2020	2	
6			3/7/2020	3/7/2020	0	
7			3/8/2020	3/8/2020	2	
8			3/9/2020	3/9/2020	13	
9			3/10/2020	3/10/2020	8	
10			3/11/2020	3/11/2020	7	
11			3/12/2020	3/12/2020	0	
12			3/13/2020	3/13/2020	35	
13			3/14/2020	3/14/2020	27	
14			3/15/2020	3/15/2020	21	
15			3/16/2020	3/16/2020	17	
16			3/17/2020	3/17/2020	38	
17			3/18/2020	3/18/2020	55	
18			3/19/2020	3/19/2020	82	
19			3/20/2020	3/20/2020	60	
20			3/21/2020	3/21/2020	81	
21			3/22/2020	3/22/2020	64	
22			3/23/2020	3/23/2020	65	
23			3/24/2020	3/24/2020	106	

Gambar 4.32 Contoh *template* data faktual yang salah

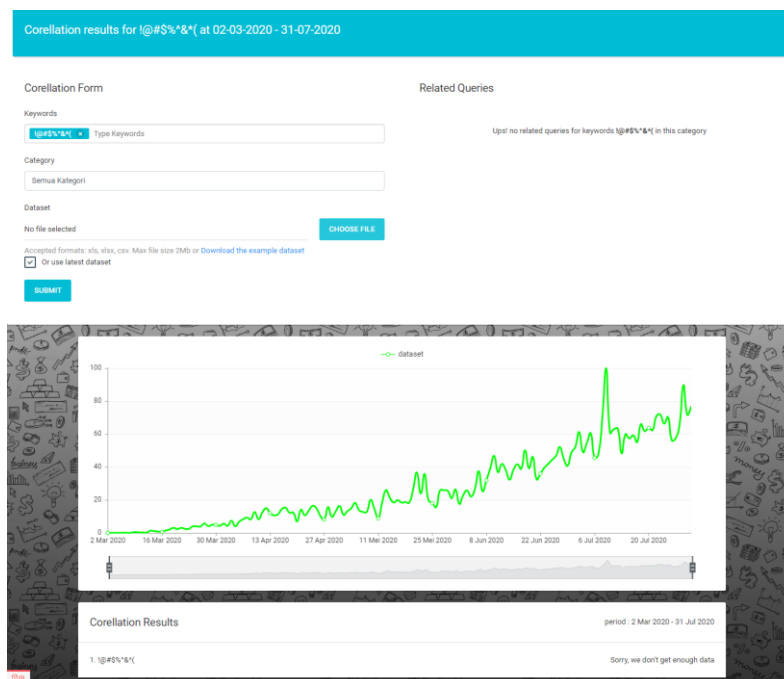




Gambar 4.33 Skenario nomor 4

## e. Skenario nomor 5

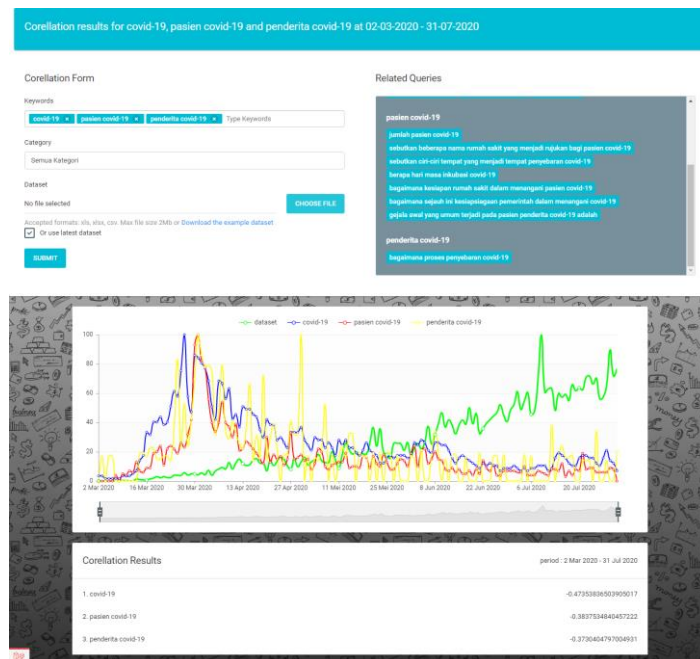
Skenario ini dimaksudkan untuk mengetahui respon aplikasi jika pengguna melakukan analisis dengan memasukan data kata kunci berupa simbol yaitu !@#\$%^&\*(). Dalam pengujian ini ditemukan bahwa aplikasi tetap mengarahkan pengguna menuju halaman result dengan menampilkan pesan bahwa tidak mendapatkan data yang cukup untuk kata kunci !@#\$%^&\*(), seperti pada Gambar 4.34.



Gambar 4.34 Skenario nomor 5

## f. Skenario nomor 6

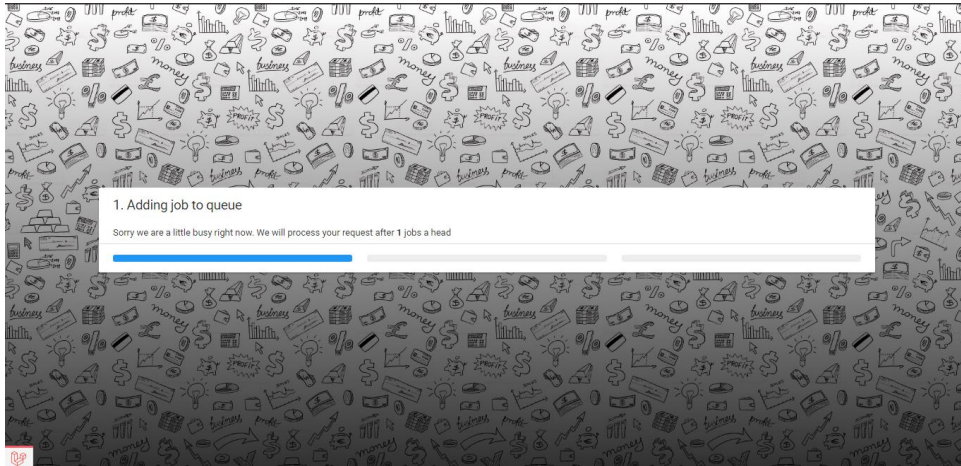
Skenario ini dimaksudkan untuk mengetahui respon aplikasi jika pengguna melakukan analisis dengan memasukan kata kunci lebih dari satu. Dalam pengujian ini ditemukan bahwa aplikasi mengarahkan pengguna menuju halaman result dengan menampilkan hasil analisis dari ketiga kata kunci yaitu covid-19, pasien covid-19, penderita covid-19, seperti pada Gambar 4.35.



Gambar 4.35 Skenario nomor 6

## g. Skenario nomor 7

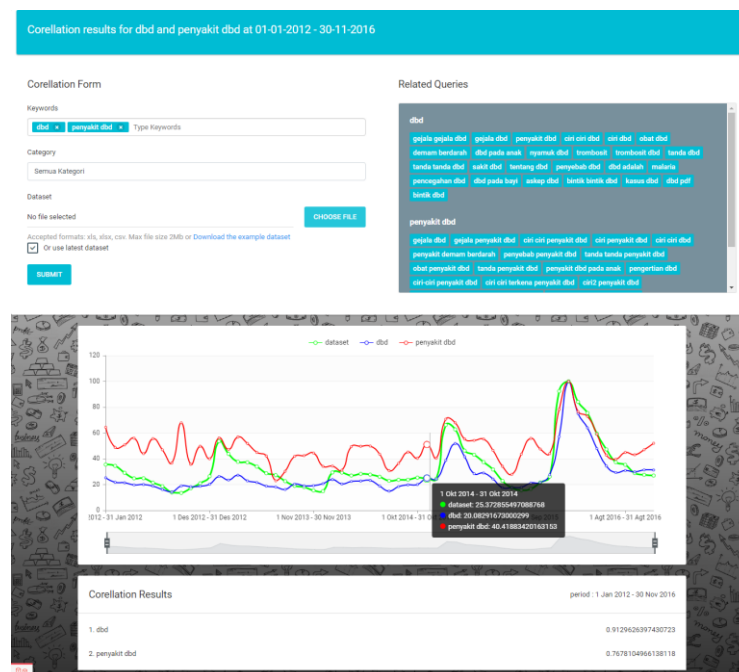
Skenario ini dimaksudkan untuk mengetahui respon aplikasi jika terdapat dua permintaan analisis dalam kurun waktu yang hampir bersamaan. Skenario ini dilakukan untuk menguji sistem antrian yang telah diterapkan. Dalam pengujian ini ditemukan bahwa aplikasi tetap mengarahkan pengguna menuju halaman progress dengan menampilkan pesan menunggu antrian yaitu satu antrian, seperti pada Gambar 4.36.



Gambar 4.36 Skenario nomor 7

#### h. Skenario nomor 8

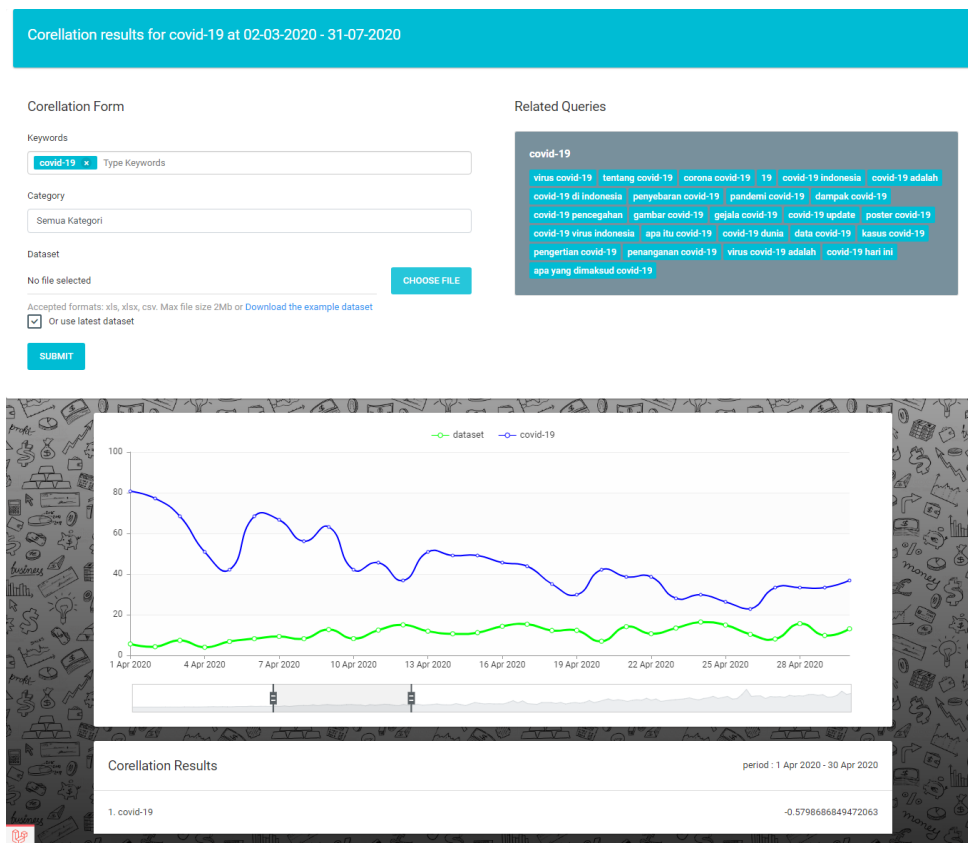
Skenario ini dimaksudkan untuk mengetahui bahwa hasil dari fitur kueri terkait dapat digunakan untuk menghitung korelasi baru. Dalam pengujian ini ditemukan bahwa aplikasi mengarahkan pengguna ke halaman result dengan menampilkan hasil korelasi dari kata kunci yang dipilih dari fitur kueri terkait, seperti pada Gambar 4.36. Adapun kata kunci yang penulis gunakan adalah “penyakit dbd” dari hasil analisis kata kunci “dbd”.



Gambar 4.37 Skenario nomor 8

#### i. Skenario nomor 9

Skenario ini dimaksudkan untuk mengetahui bahwa fitur *date range selector* dapat digunakan untuk menghitung korelasi berdasarkan rentang waktu yang ditentukan pengguna yang terdapat pada data faktual. Dalam skenario ini penulis melakukan analisis korelasi menggunakan data faktual covid-19 yang memiliki rentang waktu dari tanggal 3 Maret 2020 sampai 30 Juli 2020 dengan kata kunci “covid-19”. Kemudian setelah hasil analisis didapatkan penulis mengatur rentang waktu pada fitur *date range slider* menjadi 1 April 2020 sampai 30 April 2020. Dari hasil pengujian tersebut didapatkan kesimpulan bahwa hasil korelasi berubah sesuai dengan rentang waktu yang dipilih, seperti pada Gambar 4.38.



Gambar 4.38 Skenario nomor 9

#### 4.5.2 Pengujian Usabilitas

Pengujian selanjutnya adalah pengujian usabilitas dengan cara membagikan kuisisioner sebagai media penilaian. Adapun sampel responden dari penelitian ini adalah 10 mahasiswa Program Studi Informatika – Program Sarjana Universitas Islam Indonesia. Sebelum para responden mengisi kuisisioner terlebih dahulu dipersilahkan untuk mencoba menggunakan

aplikasi alat bantu analisis korelasi data faktual dan Google Trends dengan data faktual yang disediakan oleh penulis. Adapun data faktual yang digunakan adalah data jumlah pasien baru yang terjangkit covid-19 per-hari untuk seluruh wilayah Indonesia. Data faktual tersebut berasal dari website Gugus Tugas Percepatan Penanganan COVID-19 yang dapat diakses pada halaman <https://covid19.go.id/peta-sebaran>. Tabel 4.2 adalah hasil kuisioner yang telah di isi oleh responden.

Tabel 4.2 Hasil kuisioner

No	Aspek	Pertanyaan	TS	KS	CS	S	SS
1	Efisiensi	Apakah aplikasi alat bantu analisis data faktual dan Google Trends mampu memberikan hasil analisis sesuai dengan apa yang diharapkan?	-	-	2	5	3
2		Apakah fitur kueri terkait dan <i>date range selector</i> membantu anda dalam proses analisis?	-	-	-	5	5
3	Error	Apakah aplikasi alat bantu analisis data faktual dan Google Trends minim kesalahan?	-	1	1	3	5
4		Apakah aplikasi alat bantu analisis data faktual dan Google Trends telah memberikan informasi yang jelas dalam menampilkan pesan kesalahan pengguna?	-	-	1	3	6
5	Learnability	Apakah hasil analisis aplikasi alat bantu analisis data faktual dan Google Trends mudah dipahami oleh pengguna?	-	-	2	4	4
6		Apakah penggunaan aplikasi alat bantu analisis data faktual dan Google Trends mudah dipahami?	-	-	-	6	4
7	Kepuasan	Apakah pengguna merasa puas dengan adanya aplikasi alat bantu analisis korelasi data faktual dan Google Trends?	-	-	2	5	3
8		Apakah dengan menggunakan aplikasi alat bantu analisis korelasi data faktual dan Google Trends membantu pengguna dalam proses analisis?	-	-	2	3	5

9	Memorability	Apakah pengguna merasa lebih mudah saat menggunakan aplikasi alat bantu analisis korelasi data faktual dan Google Trends saat kedua kalinya?	-	-	3	4	3
10		Apakah Menu pada aplikasi alat bantu analisis korelasi data faktual dan Google Trends mudah diingat?	-	-	-	5	5

Setelah hasil kuisisioner didapatkan kemudian dilakukan proses perhitungan nilai untuk masing-masing pertanyaan dengan menggunakan persamaan (4.2). Adapun hasil perhitungan nilai untuk masing-masing pertanyaan dapat dilihat pada Tabel 4.3.

$$P_i = \frac{\sum_{n=1}^5 b_n \cdot h_n}{m} \quad (4.2)$$

$i$  = nomor pertanyaan

$b$  = bobot poin

$h$  = banyai isian

$m$  = jumlah responden

Tabel 4.3 Hasil pengujian untuk setiap pertanyaan

Nomor pertanyaan	Nilai
1	$\frac{(1.0) + (2.0) + (3.2) + (4.5) + (5.3)}{10} = \frac{41}{10} = 4,1$
2	$\frac{(1.0) + (2.0) + (3.0) + (4.5) + (5.5)}{10} = \frac{45}{10} = 4,5$
3	$\frac{(1.0) + (2.1) + (3.1) + (4.3) + (5.5)}{10} = \frac{42}{10} = 4,2$
4	$\frac{(1.0) + (2.0) + (3.1) + (4.3) + (5.6)}{10} = \frac{45}{10} = 4,5$
5	$\frac{(1.0) + (2.0) + (3.2) + (4.4) + (5.4)}{10} = \frac{42}{10} = 4,2$
6	$\frac{(1.0) + (2.0) + (3.0) + (4.6) + (5.4)}{10} = \frac{44}{10} = 4,4$
7	$\frac{(1.0) + (2.0) + (3.2) + (4.5) + (5.3)}{10} = \frac{41}{10} = 4,1$

8	$\frac{(1.0) + (2.0) + (3.2) + (4.3) + (5.5)}{10} = \frac{43}{10} = 4,3$
9	$\frac{(1.0) + (2.0) + (3.3) + (4.4) + (5.3)}{10} = \frac{40}{10} = 4$
10	$\frac{(1.0) + (2.0) + (3.0) + (4.5) + (5.5)}{10} = \frac{45}{10} = 4,5$

Setelah nilai dari masing-masing pertanyaan didapatkan maka langkah selanjutnya adalah menghitung skor usabilitas untuk mengukur kelayakan atas aplikasi analisis data faktual dan Google Trends untuk *monitoring / surveillance*. Adapun cara mengukur skor usabilitas adalah dengan menggunakan persamaan (4.3), sehingga skor usabilitas yang didapat adalah  $\frac{42,8}{10} = 4,28$ .

$$\text{Skor Usabilitas} = \frac{\sum \text{nilai pertanyaan}}{\text{Jumlah Pertanyaan}} \quad (4.3)$$

Setelah melakukan perhitungan skor usabilitas maka penulis menarik kesimpulan kategori kelayakan untuk aplikasi alat bantu analisis korelasi data faktual dan Google Trends berdasarkan klasifikasi pada Tabel 3.4 adalah layak untuk digunakan.

## **BAB V**

### **KESIMPULAN DAN SARAN**

#### **5.1 Kesimpulan**

Setelah semua tahapan penelitian selesai dilakukan, penulis menyimpulkan bahwa aplikasi alat bantu analisis korelasi data faktual dan Google Trends untuk *monitoring/surveillance* berhasil dikembangkan dengan bahasa pemrograman PHP menggunakan *framework* Laravel dibantu dengan beberapa fitur pendukung yaitu sistem antrian dan proxy rotator. Hal tersebut diperkuat dengan hasil pengujian yang dilakukan yaitu pengujian fungsionalitas menggunakan metode *black box* di mana didapatkan fakta bahwa aplikasi ini minim kesalahan, dan juga dari pengujian usabilitas didapatkan hasil bahwa aplikasi ini layak untuk digunakan.

#### **5.2 Saran**

Adapun saran dari penulis yang mungkin dapat dijadikan pertimbangan dalam pengembangan aplikasi ini kedepannya adalah sebagai berikut:

- a. Waktu yang digunakan pada saat proses *scraping* masih dapat dipersingkat dengan cara menggunakan *private proxy*.
- b. Penggunaan aplikasi ini bersifat umum, sehingga pengguna tidak dapat melihat aktifitas *scraping* yang pernah dilakukan sebelumnya. Dengan menerapkan sistem akun pada aplikasi ini diharapkan pengguna akan lebih terbantu dalam proses analisis korelasi.
- c. Penggunaan sistem antrian pada aplikasi ini dapat dihilangkan karena membuat waktu *scraping* menjadi lebih lama jika nantinya banyak pengguna yang menggunakan aplikasi ini dalam waktu yang bersamaan.



## DAFTAR PUSTAKA

- Anisah, A., & Kuswaya, K. (2017). Analisis Dan Perancangan Sistem Informasi Pengolahan Data Pengeluaran, Penggunaan Bahan Dan Hutang Dalam Pelaksanaan Proyek Pada Pt Banamba Putratama. *Simetris : Jurnal Teknik Mesin, Elektro Dan Ilmu Komputer*, 8(2), 507. <https://doi.org/10.24176/simet.v8i2.1352>
- Ashari. (2015). *PENERAPAN METODE TIMES SERIES DALAM SIMULASI FORECASTING PERKEMBANGAN AKADEMIK MAHASISWA Ashari Program Studi Teknik Informatika, STMIK AKBA Abstrak*.
- Biro Humas Kementerian Komunikasi dan Informatika. (2018). Pengguna Internet Indonesia Nomor Enam Dunia. Retrieved February 28, 2020, from [https://kominfo.go.id/content/detail/4286/pengguna-internet-indonesia-nomor-enam-dunia/0/sorotan\\_media](https://kominfo.go.id/content/detail/4286/pengguna-internet-indonesia-nomor-enam-dunia/0/sorotan_media)
- Budiarti, R. P. N., Widyatmoko, N., Hariadi, M., & Purnomo, M. H. (2017). Web scraping for automated water quality monitoring system: A case study of PDAM Surabaya. *Proceeding - 2016 International Seminar on Intelligent Technology and Its Application, ISITIA 2016: Recent Trends in Intelligent Computational Technologies for Sustainable Energy*, (July), 641–648. <https://doi.org/10.1109/ISITIA.2016.7828735>
- Bunafit, N. (2016). *Dasar Pemograman Web PHP-MySQL dengan Dreamweaver*. Yogyakarta: Gava Media.
- Dan, K., Pencarian, P., Taruna, I., Studi, P., Perpustakaan, I., Budaya, F. I., & Diponegoro, U. (2013). *Winarsih, Sri Ati \* Keyword : information need, information seeking behavior, cadets, Politeknik Ilmu Pelayaran Semarang. Abstrak Kata kunci : kebutuhan informasi, perilaku pencarian informasi, taruna, Politeknik Ilmu Pelayaran Semarang \* Penulis. 2.*
- Draxl, V. (2018). *Web Scraping Data Extraction from websites*.
- Google. (2020). FAQ about Google Trends data. Retrieved February 28, 2020, from <https://support.google.com/trends/answer/4365533?hl=en>
- Haddaway, N. R. (2016). The use of web-scraping software in searching for grey literature. *Grey Journal*, 11(February), 186–190.
- Husnayain, A., Fuad, A., & Lazuardi, L. (2019). Correlation between Google Trends on dengue fever and national surveillance report in Indonesia Correlation between Google Trends on dengue fever and national surveillance report in Indonesia. *Global Health Action*, 12(1).

<https://doi.org/10.1080/16549716.2018.1552652>

- Josi, A., Abdillah, L. A., & Suryayusra. (2014). *Penerapan teknik web scraping pada mesin pencari artikel ilmiah*. (October). Retrieved from <http://arxiv.org/abs/1410.5777>
- Kang, M., Zhong, H., He, J., Rutherford, S., & Yang, F. (2013). Using Google Trends for Influenza Surveillance in South China. *PLoS ONE*, 8(1), 2009–2014. <https://doi.org/10.1371/journal.pone.0055205>
- Larman, C. (2004). *Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development, Third Edition*.
- Mooney, S. J., Westreich, D. J., & El-Sayed, A. M. (2015). Epidemiology in the era of big data. *Epidemiology*, 26(3), 390–394. <https://doi.org/10.1097/eDe.0000000000000274>
- Tkachenko, N., Chotvijit, S., Gupta, N., Bradley, E., Gilks, C., Guo, W., ... Jarvis, S. (2017). Google Trends can improve surveillance of Type 2 diabetes. *Scientific Reports*, 7(1), 1–10. <https://doi.org/10.1038/s41598-017-05091-9>
- Turland, M. (2010). *Guide to Web Scraping with PHP*.

## LAMPIRAN

### Lampiran A data faktual covid-19

<b>Tanggal Mulai</b>	<b>Tanggal Selesai</b>	<b>Nilai</b>
3/2/2020	3/2/2020	2
3/3/2020	3/3/2020	0
3/4/2020	3/4/2020	0
3/5/2020	3/5/2020	0
3/6/2020	3/6/2020	2
3/7/2020	3/7/2020	0
3/8/2020	3/8/2020	2
3/9/2020	3/9/2020	13
3/10/2020	3/10/2020	8
3/11/2020	3/11/2020	7
3/12/2020	3/12/2020	0
3/13/2020	3/13/2020	35
3/14/2020	3/14/2020	27
3/15/2020	3/15/2020	21
3/16/2020	3/16/2020	17
3/17/2020	3/17/2020	38
3/18/2020	3/18/2020	55
3/19/2020	3/19/2020	82
3/20/2020	3/20/2020	60
3/21/2020	3/21/2020	81
3/22/2020	3/22/2020	64
3/23/2020	3/23/2020	65
3/24/2020	3/24/2020	106
3/25/2020	3/25/2020	105
3/26/2020	3/26/2020	103
3/27/2020	3/27/2020	153
3/28/2020	3/28/2020	109
3/29/2020	3/29/2020	130
3/30/2020	3/30/2020	129
3/31/2020	3/31/2020	114
4/1/2020	4/1/2020	149
4/2/2020	4/2/2020	113
4/3/2020	4/3/2020	196
4/4/2020	4/4/2020	106
4/5/2020	4/5/2020	181
4/6/2020	4/6/2020	218
4/7/2020	4/7/2020	247
4/8/2020	4/8/2020	218
4/9/2020	4/9/2020	337

4/10/2020	4/10/2020	219
4/11/2020	4/11/2020	330
4/12/2020	4/12/2020	399
4/13/2020	4/13/2020	316
4/14/2020	4/14/2020	282
4/15/2020	4/15/2020	297
4/16/2020	4/16/2020	380
4/17/2020	4/17/2020	407
4/18/2020	4/18/2020	325
4/19/2020	4/19/2020	327
4/20/2020	4/20/2020	185
4/21/2020	4/21/2020	375
4/22/2020	4/22/2020	283
4/23/2020	4/23/2020	357
4/24/2020	4/24/2020	436
4/25/2020	4/25/2020	396
4/26/2020	4/26/2020	275
4/27/2020	4/27/2020	214
4/28/2020	4/28/2020	415
4/29/2020	4/29/2020	260
4/30/2020	4/30/2020	347
5/1/2020	5/1/2020	433
5/2/2020	5/2/2020	292
5/3/2020	5/3/2020	349
5/4/2020	5/4/2020	395
5/5/2020	5/5/2020	484
5/6/2020	5/6/2020	367
5/7/2020	5/7/2020	338
5/8/2020	5/8/2020	336
5/9/2020	5/9/2020	533
5/10/2020	5/10/2020	387
5/11/2020	5/11/2020	233
5/12/2020	5/12/2020	484
5/13/2020	5/13/2020	689
5/14/2020	5/14/2020	568
5/15/2020	5/15/2020	490
5/16/2020	5/16/2020	529
5/17/2020	5/17/2020	489
5/18/2020	5/18/2020	496
5/19/2020	5/19/2020	486
5/20/2020	5/20/2020	693
5/21/2020	5/21/2020	973
5/22/2020	5/22/2020	634
5/23/2020	5/23/2020	949

5/24/2020	5/24/2020	526
5/25/2020	5/25/2020	479
5/26/2020	5/26/2020	415
5/27/2020	5/27/2020	686
5/28/2020	5/28/2020	687
5/29/2020	5/29/2020	678
5/30/2020	5/30/2020	557
5/31/2020	5/31/2020	700
6/1/2020	6/1/2020	467
6/2/2020	6/2/2020	609
6/3/2020	6/3/2020	684
6/4/2020	6/4/2020	585
6/5/2020	6/5/2020	703
6/6/2020	6/6/2020	993
6/7/2020	6/7/2020	672
6/8/2020	6/8/2020	847
6/9/2020	6/9/2020	1042
6/10/2020	6/10/2020	1241
6/11/2020	6/11/2020	979
6/12/2020	6/12/2020	1111
6/13/2020	6/13/2020	1014
6/14/2020	6/14/2020	857
6/15/2020	6/15/2020	1017
6/16/2020	6/16/2020	1106
6/17/2020	6/17/2020	1031
6/18/2020	6/18/2020	1331
6/19/2020	6/19/2020	1041
6/20/2020	6/20/2020	1226
6/21/2020	6/21/2020	862
6/22/2020	6/22/2020	954
6/23/2020	6/23/2020	1051
6/24/2020	6/24/2020	1113
6/25/2020	6/25/2020	1178
6/26/2020	6/26/2020	1240
6/27/2020	6/27/2020	1385
6/28/2020	6/28/2020	1198
6/29/2020	6/29/2020	1082
6/30/2020	6/30/2020	1293
7/1/2020	7/1/2020	1385
7/2/2020	7/2/2020	1624
7/3/2020	7/3/2020	1301
7/4/2020	7/4/2020	1447
7/5/2020	7/5/2020	1607
7/6/2020	7/6/2020	1209

7/7/2020	7/7/2020	1268
7/8/2020	7/8/2020	1853
7/9/2020	7/9/2020	2657
7/10/2020	7/10/2020	1611
7/11/2020	7/11/2020	1671
7/12/2020	7/12/2020	1681
7/13/2020	7/13/2020	1282
7/14/2020	7/14/2020	1591
7/15/2020	7/15/2020	1522
7/16/2020	7/16/2020	1574
7/17/2020	7/17/2020	1462
7/18/2020	7/18/2020	1752
7/19/2020	7/19/2020	1639
7/20/2020	7/20/2020	1693
7/21/2020	7/21/2020	1655
7/22/2020	7/22/2020	1882
7/23/2020	7/23/2020	1906
7/24/2020	7/24/2020	1761
7/25/2020	7/25/2020	1868
7/26/2020	7/26/2020	1492
7/27/2020	7/27/2020	1525
7/28/2020	7/28/2020	1748
7/29/2020	7/29/2020	2381
7/30/2020	7/30/2020	1904
7/31/2020	7/31/2020	2040

Lampiran B data *time series* Google Trends untuk kata kunci covid-19

<b>Tanggal Mulai</b>	<b>Tanggal Selesai</b>	<b>Nilai</b>
3/2/2020	3/2/2020	1
3/3/2020	3/3/2020	4
3/4/2020	3/4/2020	3
3/5/2020	3/5/2020	2
3/6/2020	3/6/2020	1
3/7/2020	3/7/2020	2
3/8/2020	3/8/2020	1
3/9/2020	3/9/2020	1
3/10/2020	3/10/2020	3
3/11/2020	3/11/2020	3
3/12/2020	3/12/2020	4
3/13/2020	3/13/2020	5
3/14/2020	3/14/2020	8
3/15/2020	3/15/2020	13

3/16/2020	3/16/2020	19
3/17/2020	3/17/2020	28
3/18/2020	3/18/2020	33
3/19/2020	3/19/2020	39
3/20/2020	3/20/2020	39
3/21/2020	3/21/2020	42
3/22/2020	3/22/2020	37
3/23/2020	3/23/2020	41
3/24/2020	3/24/2020	50
3/25/2020	3/25/2020	58
3/26/2020	3/26/2020	56
3/27/2020	3/27/2020	76
3/28/2020	3/28/2020	100
3/29/2020	3/29/2020	55
3/30/2020	3/30/2020	48
3/31/2020	3/31/2020	83
4/1/2020	4/1/2020	85
4/2/2020	4/2/2020	78
4/3/2020	4/3/2020	74
4/4/2020	4/4/2020	72
4/5/2020	4/5/2020	51
4/6/2020	4/6/2020	41
4/7/2020	4/7/2020	68
4/8/2020	4/8/2020	67
4/9/2020	4/9/2020	60
4/10/2020	4/10/2020	57
4/11/2020	4/11/2020	42
4/12/2020	4/12/2020	46
4/13/2020	4/13/2020	36
4/14/2020	4/14/2020	47
4/15/2020	4/15/2020	48
4/16/2020	4/16/2020	45
4/17/2020	4/17/2020	46
4/18/2020	4/18/2020	43
4/19/2020	4/19/2020	34
4/20/2020	4/20/2020	32
4/21/2020	4/21/2020	42
4/22/2020	4/22/2020	39
4/23/2020	4/23/2020	38
4/24/2020	4/24/2020	27
4/25/2020	4/25/2020	27

4/26/2020	4/26/2020	25
4/27/2020	4/27/2020	24
4/28/2020	4/28/2020	32
4/29/2020	4/29/2020	32
4/30/2020	4/30/2020	32
5/1/2020	5/1/2020	34
5/2/2020	5/2/2020	25
5/3/2020	5/3/2020	25
5/4/2020	5/4/2020	20
5/5/2020	5/5/2020	31
5/6/2020	5/6/2020	25
5/7/2020	5/7/2020	28
5/8/2020	5/8/2020	23
5/9/2020	5/9/2020	23
5/10/2020	5/10/2020	22
5/11/2020	5/11/2020	20
5/12/2020	5/12/2020	27
5/13/2020	5/13/2020	24
5/14/2020	5/14/2020	26
5/15/2020	5/15/2020	25
5/16/2020	5/16/2020	22
5/17/2020	5/17/2020	21
5/18/2020	5/18/2020	18
5/19/2020	5/19/2020	21
5/20/2020	5/20/2020	24
5/21/2020	5/21/2020	21
5/22/2020	5/22/2020	19
5/23/2020	5/23/2020	17
5/24/2020	5/24/2020	13
5/25/2020	5/25/2020	13
5/26/2020	5/26/2020	16
5/27/2020	5/27/2020	17
5/28/2020	5/28/2020	16
5/29/2020	5/29/2020	17
5/30/2020	5/30/2020	17
5/31/2020	5/31/2020	17
6/1/2020	6/1/2020	15
6/2/2020	6/2/2020	16
6/3/2020	6/3/2020	26
6/4/2020	6/4/2020	23
6/5/2020	6/5/2020	27



6/6/2020	6/6/2020	25
6/7/2020	6/7/2020	21
6/8/2020	6/8/2020	19
6/9/2020	6/9/2020	23
6/10/2020	6/10/2020	28
6/11/2020	6/11/2020	24
6/12/2020	6/12/2020	24
6/13/2020	6/13/2020	23
6/14/2020	6/14/2020	15
6/15/2020	6/15/2020	13
6/16/2020	6/16/2020	18
6/17/2020	6/17/2020	18
6/18/2020	6/18/2020	16
6/19/2020	6/19/2020	15
6/20/2020	6/20/2020	13
6/21/2020	6/21/2020	11
6/22/2020	6/22/2020	10
6/23/2020	6/23/2020	11
6/24/2020	6/24/2020	10
6/25/2020	6/25/2020	12
6/26/2020	6/26/2020	11
6/27/2020	6/27/2020	11
6/28/2020	6/28/2020	6
6/29/2020	6/29/2020	8
6/30/2020	6/30/2020	8
7/1/2020	7/1/2020	8
7/2/2020	7/2/2020	9
7/3/2020	7/3/2020	9
7/4/2020	7/4/2020	8
7/5/2020	7/5/2020	7
7/6/2020	7/6/2020	6
7/7/2020	7/7/2020	9
7/8/2020	7/8/2020	7
7/9/2020	7/9/2020	9
7/10/2020	7/10/2020	11
7/11/2020	7/11/2020	11
7/12/2020	7/12/2020	7
7/13/2020	7/13/2020	9
7/14/2020	7/14/2020	14
7/15/2020	7/15/2020	19
7/16/2020	7/16/2020	17

7/17/2020	7/17/2020	17
7/18/2020	7/18/2020	14
7/19/2020	7/19/2020	12
7/20/2020	7/20/2020	10
7/21/2020	7/21/2020	16
7/22/2020	7/22/2020	17
7/23/2020	7/23/2020	18
7/24/2020	7/24/2020	16
7/25/2020	7/25/2020	15
7/26/2020	7/26/2020	12
7/27/2020	7/27/2020	9
7/28/2020	7/28/2020	16
7/29/2020	7/29/2020	21
7/30/2020	7/30/2020	13
7/31/2020	7/31/2020	13