

Sistem Rekomendasi Pada Aplikasi Wisata Mangunan Menggunakan Metode *Content Based Filtering*

Fahmi Majid¹

¹ Program Studi Informatika, STMIK El Rahma Yogyakarta

e-mail :¹ fahmi.majid26@gmail.com

Abstrak

Desa Mangunan adalah sebuah desa yang terletak di Kecamatan Dlingo, Kabupaten Bantul, D.I. Yogyakarta. Desa ini merupakan salah satu desa wisata terkenal di Yogyakarta dengan menyajikan destinasi wisata alam yang asri di ketinggian 150-200 mdpl. Kebutuhan akan teknologi seperti smartphone dan internet mengharuskan penyebaran informasi di sektor pariwisata untuk memanfaatkan perkembangan teknologi supaya informasi wisata dapat diakses oleh wisatawan dengan mudah, cepat dan efisien. Namun belum ada sebuah aplikasi untuk smartphone yang secara khusus memberikan informasi destinasi wisata di daerah Mangunan.

Berdasarkan permasalahan tersebut dibuatlah sebuah aplikasi wisata Mangunan yang dibangun menggunakan bahasa pemrograman Dart dan framework Flutter dengan memanfaatkan layanan dari Mapbox untuk menampilkan peta dan navigasinya. Pada aplikasi ini juga akan dibangun sebuah sistem rekomendasi menggunakan metode content based filtering dengan menerapkan algoritma TF-IDF dan cosine similarity.

Hasil dari penelitian ini adalah sebuah aplikasi android yang membantu pengguna untuk mengetahui informasi wisata di Mangunan. Selain itu aplikasi ini akan memberikan 5 rekomendasi wisata berdasarkan wisata yang disukai oleh pengguna. Pada pengujian sistem rekomendasi wisata dengan metode perhitungan precision, nilai akurasi dari sistem rekomendasi wisata dengan menggunakan 10 sampel data mampu mencapai angka 96% yang menunjukkan bahwa tingkat akurasi sistem rekomendasi sangat tinggi.

Kata kunci - Aplikasi Wisata Mangunan, Sistem Rekomendasi, Content Based Filtering

Abstract

Mangunan is a village located in Dlingo, Bantul, Yogyakarta. This village become one of the popular tourist village in Yogyakarta with the beautiful natural tourist destination at an altitude of 150-200 masl. The need for technology such as smartphone and internet are force the spread of tourism information to utilize the growth of technology to make the information can accessed by tourist easily, fast, and efficient. However, there is no application for smartphone that specifically provides the information about tourism destination in Mangunan.

Based on those problems, a Mangunan tourism application was made using Dart as the programming language and Flutter as the framework. The services from Mapbox also deployed in this application to show a map and navigation for tourist. Moreover a reccomendation system will be built in this app using content based filtering as the methods and implementing TF-IDF and cosine similarity algorithms.

The result of this research is an android application that used to provide information about tourism destination in Mangunan. This application also will give 5 reccomendation of tourism destination based on user preferences. After doing a test to the reccomendation system using precision method, the accuracy of reccomendation system using 10 data sample reach 96% which mean if the accuracy level of reccomendation system is high.

Keywords - Mangunan Tourism App, Reccomendation System, Content Based Filtering

1. PENDAHULUAN

Desa Mangunan adalah sebuah desa yang terletak di Kecamatan Dlingo, Kabupaten Bantul, D.I. Yogyakarta. Desa ini merupakan salah satu desa wisata terkenal di Yogyakarta dengan menyajikan destinasi wisata alam yang asri di ketinggian 150-200 mdpl. Titik ketinggian tersebut membuat kawasan ini memiliki alam yang indah dan udara yang sejuk serta pemandangan pegunungan seribu yang menyajikan berbagai keindahan alam. Desa Mangunan ini memiliki beberapa keunggulan potensi dan keunikan daya tarik pariwisata terutama pada daya tarik wisata alam [1].

Destinasi wisata yang terdapat di Mangunan berjumlah 16 yang terdiri dari hamparan hutan pinus, perbukitan dan juga air terjun. Destinasi wisata tersebut antara lain Jurang Tembelan, Seribu Batu Songgo Langit, Pinus Pengger, Watu Goyang, Watu Ngadek, Rumah Hobbit Mangunan, Gardu Pandang Pinus Asri, Hutan Pinus Mangunan, Kebun Buah Mangunan, Bukit Lintang Sewu, Bukit Panguk Kediwung, Bukit Mojo Gumelem, Tebing Watu Mabur, Puncak Becici, Air Terjun Randusari dan Grojogan Lepo Dlingo [2].

Saat ini hampir semua orang mempunyai *smartphone*, bahkan bisa dibilang *smartphone* merupakan sebuah kebutuhan dikarenakan kemudahan dalam penggunaan dan mengakses informasi yang tersedia di internet. Kementerian Komunikasi dan Informatika menyatakan bahwa pengguna *smartphone* di Indonesia mencapai 167 juta orang atau dengan persentase 89% dari total penduduk Indonesia. Hal tersebut menunjukkan bahwa kebutuhan *smartphone* dan internet di Indonesia sangatlah tinggi [3].

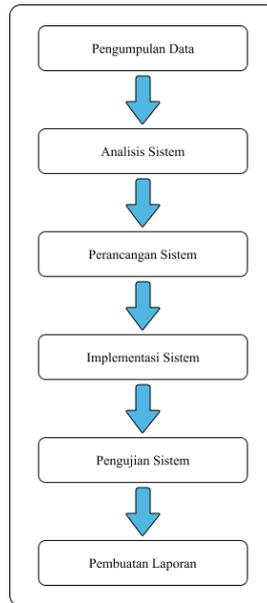
Penyebaran informasi di sektor pariwisata juga harus memanfaatkan perkembangan teknologi supaya informasi wisata dapat diakses oleh wisatawan dengan mudah, cepat dan efisien. Salah satu pemanfaatan teknologi di bidang pariwisata adalah aplikasi wisata berbasis *mobile* yang di dalamnya terdapat informasi mengenai destinasi wisata yang terdapat di suatu daerah. Dalam penelitian [4] menghasilkan sebuah Aplikasi Objek Wisata Berbasis Android di Kota Pekanbaru yang dapat membantu masyarakat untuk mencari objek wisata yang terdapat di kota Pekanbaru secara terarah karena memiliki sumber acuan. Dalam aplikasi tersebut juga terdapat fitur yang sangat berguna untuk wisatawan yaitu layanan navigasi yang dapat menampilkan rute yang dapat ditempuh wisatawan untuk mencapai objek wisata tersebut.

Pemanfaatan teknologi dalam bidang pariwisata lainnya yang dapat membantu wisatawan dalam merancang dan melaksanakan kegiatan wisatanya adalah sistem rekomendasi destinasi wisata. Dengan adanya sistem rekomendasi tersebut maka wisatawan dapat mengetahui rekomendasi destinasi wisata berdasarkan destinasi wisata lain yang disukai oleh pengguna. Selain itu dalam penelitian [5] menyatakan bahwa salah satu cara untuk membuat pengunjung mengakses sistem website maupun aplikasi lebih lama adalah dengan membuat sistem rekomendasi.

Dengan adanya permasalahan dan uraian data di atas maka dibuatlah sebuah aplikasi wisata Mangunan berbasis android untuk memudahkan wisatawan dalam mencari lokasi wisata di Mangunan beserta layanan navigasinya. Di dalam aplikasi ini wisatawan akan diberikan lokasi-lokasi wisata yang terdapat di Mangunan. Aplikasi ini juga akan menampilkan rekomendasi wisata berdasarkan lokasi wisata yang disukai oleh pengguna dengan menerapkan metode *content-based filtering*.

2. METODE PENELITIAN

Metode pengembangan sistem yang digunakan dalam pembuatan Aplikasi Wisata Mangunan ini adalah SDLC(*System Development Life Cycle*) dengan metode *Waterfall*. Adapun metode pengembangan sistem terdapat pada Gambar 1.



Gambar 1 Metode pengembangan sistem

Adapun penjelasan dari metode pengembangan sistem pada Gambar 1 dapat dilihat seperti berikut.

a. Pengumpulan Data

Pengumpulan data dilakukan dengan mencari informasi lokasi wisata yang ada di Mangunan melalui website. Koordinat dari setiap lokasi wisata didapatkan melalui layanan Google Maps dan OpenStreetMap.

b. Analisis Sistem

Analisis sistem dilakukan dengan mencari aplikasi wisata yang terdapat di Google Play Store untuk mengetahui fitur apa saja yang dibutuhkan pada aplikasi wisata dan fitur apa yang belum ada namun dapat membantu wisatawan.

c. Perancangan Sistem

Dalam perancangan sistem ini dilakukan beberapa tahapan yaitu perancangan UML untuk memberikan gambaran interaksi antara wisatawan dan sistem, perancangan database, perancangan sistem rekomendasi dan perancangan desain UI sebagai acuan untuk tampilan antarmuka aplikasi.

d. Implementasi Sistem

Rancangan yang telah dibuat sebelumnya diimplementasikan menjadi aplikasi Android menggunakan bahasa pemrograman Dart dan *framework* Flutter. Database diimplementasikan di Firebase, lalu layanan peta dan navigasi menggunakan Mapbox. Sistem rekomendasi dibangun menggunakan metode *content based filtering* dengan menerapkan algoritma *cosine similarity* dan TF-IDF sedangkan untuk bahasa pemrograman yang digunakan adalah Python dan *framework* Flask.

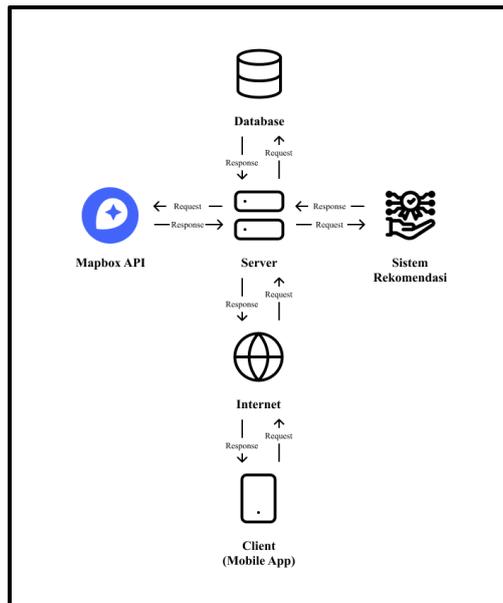
e. Pengujian Sistem

Pengujian sistem dilakukan menggunakan metode *black box testing* dengan cara mengamati hasil *input* dan *output*. Untuk pengujian sistem rekomendasi menggunakan perhitungan *precision* dengan menggunakan nilai *true positives* dan *false positives*.

f. Pembuatan Laporan

Pembuatan laporan digunakan sebagai dokumentasi dalam proses pembuatan aplikasi wisata Mangunan dan dapat dijadikan acuan dalam proses pengujian dan penelitian lebih lanjut.

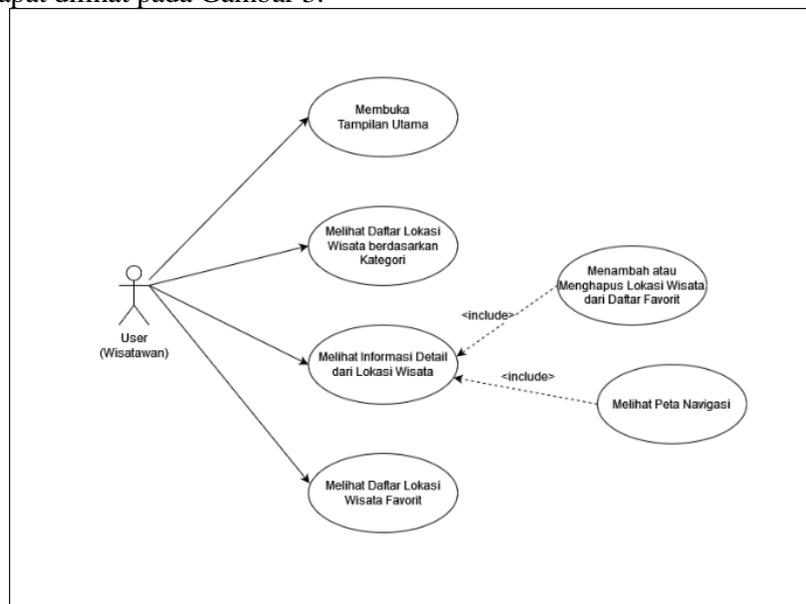
Pada tahap perancangan, sistem akan diuraikan berdasarkan analisis kebutuhan yang terdiri dari arsitektur sistem, pemodelan menggunakan UML dan perancangan desain atarmuka. Arsitektur aplikasi wisata Mangunan dapat dilihat pada Gambar 2.



Gambar 2 Arsitektur sistem

Pada Gambar 2 dapat dilihat bahwa aplikasi diakses menggunakan perangkat android yang terhubung ke koneksi internet. Setelah *client* terhubung ke server maka aplikasi dapat membuka layanan yang disiapkan oleh server seperti *database*, sistem rekomendasi, dan layanan peta menggunakan Mapbox API.

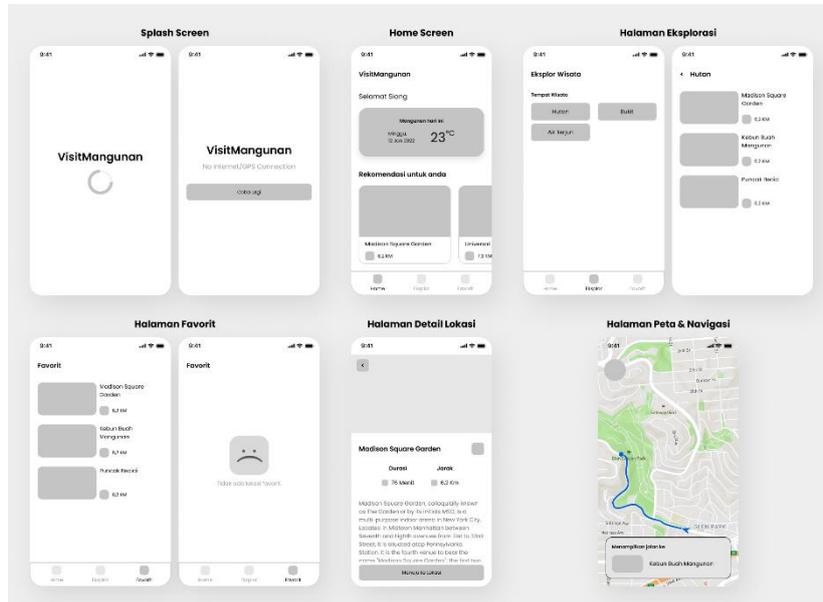
Perancangan berikutnya adalah *use case diagram* yang menggambarkan hubungan interaksi antara sistem dan aktor. *Use case diagram* dapat mendeskripsikan tipe interaksi antara si pengguna sistem dengan sistemnya. Tampilan *use case diagram* pada *user* aplikasi wisata Mangunan dapat dilihat pada Gambar 3.



Gambar 3 Use case diagram wisatawan

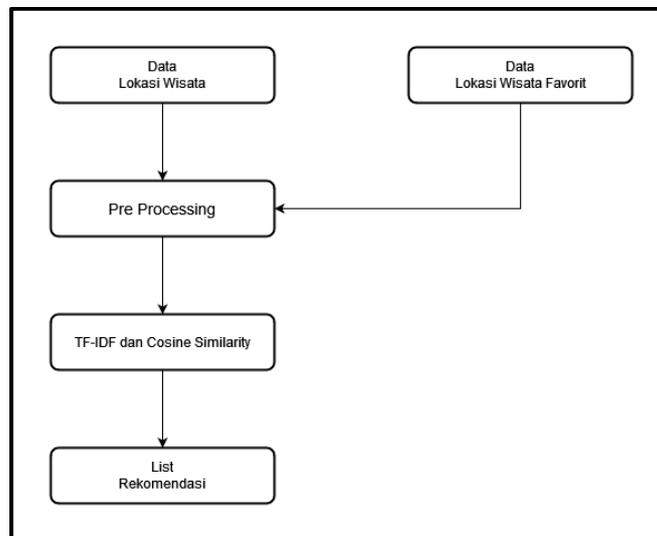
Pada Gambar 3 dapat dilihat bahwa dalam aplikasi wisata Mangunan terdapat 1 aktor utama dengan 6 fungsionalitas yaitu membuka tampilan utama, melihat daftar lokasi wisata berdasarkan kategori, melihat daftar lokasi wisata favorit, melihat halaman informasi detail lokasi wisata, melihat peta navigasi dan juga menambah atau menghapus lokasi wisata dari daftar favorit. Untuk melihat peta navigasi dan menambah atau menghapus lokasi wisata dari daftar favorit pengguna harus membuka halaman detail lokasi wisata terlebih dahulu.

Berikutnya perancangan desain antarmuka digunakan untuk memberikan gambaran dasar mengenai struktur menu, navigasi, tampilan, serta apa saja data dan informasi yang akan ditampilkan dalam aplikasi. Rancangan desain antarmuka dari aplikasi wisata Mangunan dapat dilihat pada Gambar 4.



Gambar 4 Rancangan desain antarmuka aplikasi wisata Mangunan

Dalam pengembangan sistem rekomendasi ini diperlukan rancangan yang dapat menjelaskan proses apa saja yang berlangsung. Gambaran dari tahapan pada sistem rekomendasi dapat dilihat pada Gambar 5.



Gambar 5 Tahapan sistem rekomendasi

Pada Gambar 5 tahapan dari sistem rekomendasi ini dimulai dari data lokasi wisata dari *database* dan data lokasi wisata favorit pengguna akan digabungkan untuk dilakukan tahap *preprocessing* yang terdiri dari *case folding*, *tokenizing*, *filtering* dan *stemming*. Setelah itu akan dilakukan proses penghitungan kemiripan menggunakan 2 algoritma yaitu pembobotan TF-IDF dan *cosine similarity*.

Pre processing

Pre processing adalah tahapan awal dalam *data mining* untuk mengubah data mentah menjadi informasi yang lebih bersih dan bisa digunakan untuk pengolahan data selanjutnya. Pada tahapan *preprocessing* kata-kata atau teks yang tidak perlu atau tidak mempunyai arti akan dihilangkan,

dengan berkurangnya jumlah teks diharapkan dapat meringankan proses selanjutnya dalam rangka menambang informasi yang berada dalam dokumen atau teks untuk dapat menghasilkan informasi yang berguna tanpa mengurangi arti ataupun makna serta informasi yang dikandung dalam dokumen tersebut [6].

Pada penelitian ini tahapan *preprocessing* dilakukan dengan memanfaatkan NLP (*Natural Language Processing*) menggunakan tools NLTK (*Natural Language ToolKit*) yang digunakan pada bahasa pemrograman Python. Adapun tahapan dalam *preprocessing* terdiri dari *case folding*, *tokenizing*, *filtering*, dan *stemming*.

a. *Case folding*

Case folding merupakan tahapan untuk mengkonversi seluruh teks menjadi huruf kecil (*lowercase*). Fungsi dari *case folding* adalah menyamaratakan penggunaan huruf di setiap dokumen agar lebih terstruktur dan konsisten. Contoh dari proses *case folding* dapat dilihat pada Tabel 1.

Tabel 1 Contoh proses *case folding*

Sebelum	Sesudah
Perjalanan menuju Bukit Panguk Kediwung akan melalui bukit yang naik turun sehingga diperlukan kondisi tubuh dan kendaraan yang fit. Dari bukit ini kita akan menikmati pemandangan alam berupa hamparan perbukitan yang hijau dan subur. Di pagi buta, gumpalan awan dan kabut tipis akan menambah indahny pemandangan dan akan sangat bagus sebagai latar belakang untuk berfoto.	perjalanan menuju bukit panguk kediwung akan melalui bukit yang naik turun sehingga diperlukan kondisi tubuh dan kendaraan yang fit. dari bukit ini kita akan menikmati pemandangan alam berupa hamparan perbukitan yang hijau dan subur. di pagi buta, gumpalan awan dan kabut tipis akan menambah indahny pemandangan dan akan sangat bagus sebagai latar belakang untuk berfoto.
Grojogan Lepo Dlingo menjadi salah satu air terjun yang cukup terkenal di kabupaten Bantul karena kondisi airnya yang masih sangat alami. Grojogan ini memiliki beberapa tingkatan air terjun yang berada dalam satu aliran sungai.	grojogan lepo dlingo menjadi salah satu air terjun yang cukup terkenal di kabupaten bantul karena kondisi airnya yang masih sangat alami. grojogan ini memiliki beberapa tingkatan air terjun yang berada dalam satu aliran sungai.

b. *Tokenizing*

Tokenizing atau dikenal juga dengan tahap *Lexical Analysis* adalah proses pemotongan teks menjadi bagian-bagian yang lebih kecil yang disebut token. Selain itu pada tahap *tokenizing* juga dilakukan proses penghilangan angka, tanda baca dan karakter lain yang dianggap tidak memiliki pengaruh terhadap pemrosesan teks. Contoh dari proses *tokenizing* dapat dilihat pada Tabel 2.

Tabel 2 Contoh proses *tokenizing*

Sebelum	Sesudah
perjalanan menuju bukit panguk kediwung akan melalui bukit yang naik turun sehingga diperlukan kondisi tubuh dan kendaraan yang fit. dari bukit ini kita akan menikmati pemandangan alam berupa hamparan perbukitan yang hijau dan subur. di pagi buta, gumpalan awan dan kabut tipis akan menambah indahny pemandangan dan akan sangat bagus sebagai latar belakang untuk berfoto.	perjalanan menuju bukit panguk kediwung akan melalui bukit yang naik turun sehingga diperlukan kondisi tubuh dan kendaraan yang fit dari bukit ini kita akan menikmati pemandangan alam berupa hamparan perbukitan yang hijau dan subur di pagi buta gumpalan awan dan kabut tipis akan menambah indahny pemandangan dan akan sangat bagus sebagai latar belakang untuk berfoto
grojogan lepo dlingo menjadi salah satu air terjun yang cukup terkenal di kabupaten bantul karena kondisi airnya yang masih sangat alami. grojogan ini memiliki beberapa tingkatan air terjun yang berada dalam satu aliran sungai.	grojogan lepo dlingo menjadi salah satu air terjun yang cukup terkenal di kabupaten bantul karena kondisi airnya yang masih sangat alami grojogan ini memiliki beberapa tingkatan air terjun yang berada dalam satu aliran sungai

c. *Filtering*

Filtering adalah tahap yang digunakan untuk mengambil kata-kata yang dianggap penting. Tahap *filtering* juga biasa disebut dengan *stopword removal* dimana *stopword* adalah kata umum yang biasanya muncul dan tidak memiliki makna, contoh kata penghubung seperti “dan”, “yang”, “serta”, “setelah”, dan lainnya. Kata yang termasuk ke dalam daftar *stopword* akan dibuang dan tidak digunakan pada proses selanjutnya. Contoh dari proses *filtering* dapat dilihat pada Tabel 3.

Tabel 3 Contoh proses *filtering*

Sebelum	Sesudah
perjalanan menuju bukit panguk kediwung akan melalui bukit yang naik turun sehingga diperlukan kondisi tubuh dan kendaraan yang fit dari bukit ini kita akan menikmati pemandangan alam berupa hamparan perbukitan yang hijau dan subur di pagi buta gumpalan awan dan kabut tipis akan menambah indahny pemandangan dan akan sangat bagus sebagai latar belakang untuk berfoto	perjalanan bukit panguk kediwung bukit turun kondisi tubuh kendaraan fit bukit menikmati pemandangan alam hamparan perbukitan hijau subur pagi buta gumpalan awan kabut tipis menambah indahny pemandangan bagus latar berfoto
grojogan lepo dlingo menjadi salah satu air terjun yang cukup terkenal di kabupaten bantul karena kondisi airnya yang masih sangat alami grojogan ini memiliki beberapa tingkatan air terjun yang berada dalam satu aliran sungai	grojogan lepo dlingo salah air terjun terkenal kabupaten bantul kondisi airnya alami grojogan memiliki tingkatan air terjun aliran sungai

d. *Stemming*

Stemming adalah proses untuk mengubah bentuk kata menjadi kata dasar. Imbuan yang terdapat di awal kata(*prefix*) dan akhir kata(*suffix*) akan dihapus. Contoh dari proses *stemming* dapat dilihat pada Tabel 4.

Tabel 4 Contoh proses *stemming*

Sebelum	Sesudah
perjalanan bukit panguk kediwung bukit turun kondisi tubuh kendaraan fit bukit menikmati pemandangan alam hamparan perbukitan hijau subur pagi buta gumpalan awan kabut tipis menambah indahny pemandangan bagus latar berfoto	jalan bukit panguk kediwung bukit turun kondisi tubuh kendar fit bukit nikmat pandang alam hampar bukit hijau subur pagi buta gumpal awan kabut tipis tambah indah pandang bagus latar foto
grojogan lepo dlingo salah air terjun terkenal kabupaten bantul kondisi airnya alami grojogan memiliki tingkatan air terjun aliran sungai	grojogan lepo dlingo salah air terjun kenal kabupaten bantul kondisi air alami grojogan milik tingkat air terjun alir sungai

Pembobotan TF-IDF

Pembobotan kata TF-IDF digunakan untuk memberikan bobot pada setiap kata dengan cara menghitung *term frequency* yaitu frekuensi kemunculan sebuah kata di dalam sebuah dokumen dan *inverse document frequency* yaitu menghitung frekuensi dokumen yang mengandung kata tersebut. Frekuensi kemunculan kata di dalam dokumen yang diberikan menunjukkan seberapa penting kata itu di dalam dokumen. Bobot hubungan antara sebuah kata dan sebuah dokumen akan tinggi apabila frekuensi kata tersebut tinggi dan frekuensi keseluruhan dokumen yang mengandung kata tersebut rendah [7].

Untuk melakukan penghitungan *term frequency* dapat menggunakan Persamaan (1) [8].

$$TF_{i,j} = f_{i,j} \quad (1)$$

Dengan $TF_{i,j}$ adalah *term frequency*, $f_{i,j}$ adalah frekuensi kemunculan dari kata k_i dalam dokumen d_j . Lalu untuk menghitung *inverse document frequency* dapat dilakukan dengan menggunakan Persamaan (2).

$$IDF_i = \log \frac{N}{n_i} \quad (2)$$

Dengan IDF_i adalah *inverse document frequency*, N adalah jumlah dokumen yang digunakan oleh sistem dan n_i adalah jumlah dokumen yang didalamnya terdapat kata k_i . Setelah itu bobot TF-IDF dari kata k_i di dalam dokumen d_j dapat dihitung dengan Persamaan (3).

$$w_{i,j} = TF_{i,j} \times IDF_i \quad (3)$$

Sebagai contoh, terdapat 3 dokumen berisikan kalimat yang sudah melewati tahap *preprocessing* seperti berikut.

- Dokumen 1: “air terjun randusari letak dekat puncak becici”
- Dokumen 2: “grojogan lepo dlingo salah air terjun kenal bantul”

- Dokumen 3: “puncak beciici suasana hutan pinus”

Perhitungan bobot kata menggunakan algoritma TF-IDF dapat dilakukan dengan menggunakan Persamaan (3). Proses dan hasil pembobotan kata dari ketiga dokumen diatas dapat dilihat pada Tabel 5.

Tabel 5 Contoh pembobotan TF-IDF

No	Kata	TF			n	Idf	TF x IDF		
		Doc 1	Doc 2	Doc 3			Doc 1	Doc 2	Doc 3
1.	air	1	1	0	2	Log(3/2) = 0.1760	0.1760	0.1760	0
2.	terjun	1	1	0	2	Log(3/2) = 0.1760	0.1760	0.1760	0
3.	randusari	1	0	0	1	Log(3/1) = 0.4771	0.4771	0	0
4.	letak	1	0	0	1	Log(3/1) = 0.4771	0.4771	0	0
5.	dekat	1	0	0	1	Log(3/1) = 0.4771	0.4771	0	0
6.	puncak	1	0	1	2	Log(3/2) = 0.1760	0.1760	0	0.1760
7.	becici	1	0	1	2	Log(3/2) = 0.1760	0.1760	0	0.1760
8.	grojogan	0	1	0	1	Log(3/1) = 0.4771	0	0.4771	0
9.	lepo	0	1	0	1	Log(3/1) = 0.4771	0	0.4771	0
10.	dlingo	0	1	0	1	Log(3/1) = 0.4771	0	0.4771	0
11.	salah	0	1	0	1	Log(3/1) = 0.4771	0	0.4771	0
12.	kenal	0	1	0	1	Log(3/1) = 0.4771	0	0.4771	0
13.	bantul	0	1	0	1	Log(3/1) = 0.4771	0	0.4771	0
14.	suasana	0	0	1	1	Log(3/1) = 0.4771	0	0	0.4771
15.	hutan	0	0	1	1	Log(3/1) = 0.4771	0	0	0.4771
16.	pinus	0	0	1	1	Log(3/1) = 0.4771	0	0	0.4771

Cosine Similarity

Cosine similarity adalah algoritma yang digunakan untuk menghitung tingkat kesamaan antar 2 buah atau lebih objek yang dinyatakan dalam vektor. *Cosine similarity* dapat digunakan untuk menemukan kemiripan antar dokumen dalam *dataset* dengan jumlah yang besar [9]. Hasil dari pembobotan TF-IDF di tahap sebelumnya dapat diterapkan dalam algoritma *cosine similarity* ini.

Untuk menghitung *cosine similarity* dapat dilakukan dengan menggunakan Persamaan (4) [8].

$$sim(\vec{w}_c, \vec{w}_s) = \frac{\vec{w}_c \cdot \vec{w}_s}{\|\vec{w}_c\|_2 \times \|\vec{w}_s\|_2} = \frac{\sum_{i=1}^K w_{i,c} \cdot w_{i,s}}{\sqrt{\sum_{i=1}^K w_{i,c}^2} \sqrt{\sum_{i=1}^K w_{i,s}^2}} \quad (4)$$

Dengan *sim* adalah *cosine similarity*, \vec{w}_c adalah bobot query dan \vec{w}_s adalah bobot dokumen. Lalu $\|\vec{w}_c\|_2$ adalah jarak query dan $\|\vec{w}_s\|_2$ adalah jarak dokumen. $w_{i,c}$ adalah bobot *query* \vec{w}_c dokumen ke-*i* dan $w_{i,s}$ adalah bobot dokumen \vec{w}_s ke-*i*.

Sebagai contoh, dari Tabel 4 akan dihitung kemiripan antara dokumen 1 dengan dokumen 3. Adapun data yang digunakan adalah hasil perhitungan TF-IDF dari masing-masing kata pada dokumen 1 dan dokumen 3. Data yang telah dipilah dari Tabel 5 dapat dilihat pada Tabel 6.

Tabel 6 Data untuk perhitungan *cosine similarity*

No	Kata	TF-IDF	
		Doc 1	Doc 3
1.	air	0.1760	0
2.	terjun	0.1760	0
3.	randusari	0.4771	0
4.	letak	0.4771	0
5.	dekat	0.4771	0
6.	puncak	0.1760	0.1760
7.	becici	0.1760	0.1760
8.	suasana	0	0.4771
9.	hutan	0	0.4771

10.	pinus	0	0.4771
-----	-------	---	--------

Dengan menggunakan Persamaan (4), perhitungan kemiripan antara dokumen 1 dan dokumen 3 pada Tabel 6 menggunakan algoritma *cosine similarity* dapat dilakukan sebagai berikut.

$$\begin{aligned}\overrightarrow{doc1} \cdot \overrightarrow{doc3} &= (0.1760 \times 0) + (0.1760 \times 0) + (0.4771 \times 0) + (0.4771 \times 0) \\ &\quad + (0.4771 \times 0) + (0.1760 \times 0.1760) + (0.1760 \times 0.1760) + (0 \times 0.4771) \\ &\quad + (0 \times 0.4771) + (0 \times 0.4771) \\ &= 0.0309 + 0.0309 \\ &= 0.0618\end{aligned}$$

$$\begin{aligned}\|\overrightarrow{doc1}\| &= \sqrt{\frac{0.1760^2 + 0.1760^2 + 0.4771^2 + 0.4771^2 + 0.4771^2}{+0.1760^2 + 0.1760^2 + 0^2 + 0^2 + 0^2}} \\ &= \sqrt{0.0309 + 0.0309 + 0.2276 + 0.2276 + 0.2776 + 0.0309 + 0.0309} \\ &= \sqrt{0.8064} \\ &= 0.8979\end{aligned}$$

$$\begin{aligned}\|\overrightarrow{doc3}\| &= \sqrt{\frac{0^2 + 0^2 + 0^2 + 0^2 + 0^2 + 0.1760^2 + 0.1760^2 + 0.4771^2}{+0.4771^2 + 0.4771^2}} \\ &= \sqrt{0.0309 + 0.0309 + 0.2276 + 0.2276 + 0.2276} \\ &= \sqrt{0.7446} \\ &= 0.8629\end{aligned}$$

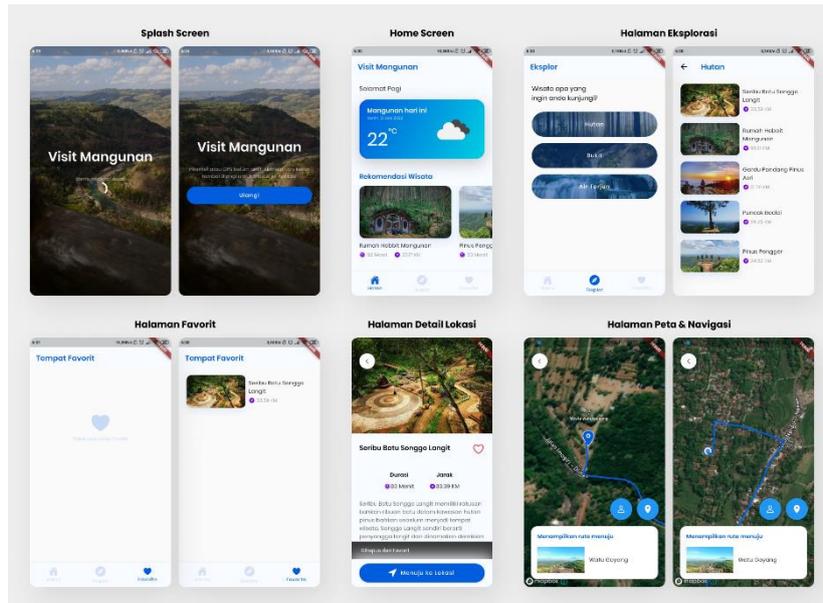
$$sim(\overrightarrow{doc1}, \overrightarrow{doc3}) = \frac{\overrightarrow{doc1} \cdot \overrightarrow{doc3}}{\|\overrightarrow{doc1}\| \times \|\overrightarrow{doc3}\|} = \frac{0.0618}{0.8979 \times 0.8629} = 0.0797$$

Setelah melakukan perhitungan *cosine similarity* menggunakan data hasil perhitungan TF-IDF, maka diperoleh hasil dari tingkat kemiripan antara dokumen 1 dengan dokumen 3 yaitu sebesar 0.0797.

3. HASIL DAN PEMBAHASAN

Implementasi Aplikasi

Aplikasi wisata Mangunan telah diimplementasikan pada program aplikasi *android* dengan menggunakan *framework* Flutter dan bahasa pemrograman Dart berdasarkan perancangan sistem yang telah dibuat sebelumnya. Implementasi dari aplikasi wisata Mangunan dapat dilihat pada Gambar 6.



Gambar 6 Implementasi aplikasi wisata Mangunan

Pengujian Black Box

Black box testing merupakan salah satu metode yang digunakan untuk pengujian suatu fungsionalitas sebuah aplikasi atau perangkat lunak dengan menyampingkan pengetahuan mengenai baris kode yang ada didalamnya. Fokus yang diuji dalam pengujian *black box* adalah *input* dan *output* pada aplikasi yang dikembangkan. Penguji memberikan masukan dan mengamati keluaran yang dihasilkan oleh sistem. Hal ini memungkinkan untuk mengidentifikasi bagaimana sistem merespons tindakan pengguna apakah sesuai dengan apa yang diharapkan atau tidak [10]. Hasil pengujian *black box* dapat dilihat pada Tabel 7.

Tabel 7 Hasil pengujian *black box*

Lokasi Pengujian	Aktivitas Pengujian	Keluaran yang diharapkan	Hasil Pengujian	Kesimpulan
<i>Splash screen</i>	Menjalankan aplikasi tanpa terhubung ke internet atau tanpa mengaktifkan GPS	Menampilkan dialog permintaan mengaktifkan internet dan GPS	Menampilkan dialog permintaan mengaktifkan internet dan GPS	Berhasil
	Menolak permintaan mengaktifkan internet dan GPS setelah menjalankan aplikasi tanpa terhubung ke internet atau tanpa mengaktifkan GPS	Menampilkan pesan kesalahan dan tombol "Ulangi"	Menampilkan pesan kesalahan dan tombol "Ulangi"	Berhasil
	Menekan tombol "Ulangi" setelah menolak permintaan mengaktifkan internet dan GPS	Menampilkan dialog permintaan mengaktifkan internet dan GPS	Menampilkan dialog permintaan mengaktifkan internet dan GPS	Berhasil
	Menerima permintaan mengaktifkan internet dan GPS setelah menjalankan aplikasi tanpa terhubung ke internet atau tanpa mengaktifkan GPS	Masuk ke halaman utama(<i>home screen</i>)	Masuk ke halaman utama(<i>home screen</i>)	Berhasil
<i>Home screen</i>	Memilih dan menekan salah satu lokasi wisata yang ditampilkan	Menampilkan halaman detail lokasi wisata yang dipilih	Menampilkan halaman detail lokasi wisata yang dipilih	Berhasil

Lokasi Pengujian	Aktivitas Pengujian	Keluaran yang diharapkan	Hasil Pengujian	Kesimpulan
	Buka detail lokasi wisata, tandai lokasi wisata sebagai favorit, kembali ke <i>home screen</i>	Menampilkan <i>section</i> “Rekomendasi Wisata” dan animasi <i>loading</i> , lalu menampilkan rekomendasi wisata	Menampilkan nama <i>section</i> “Rekomendasi Wisata” dan animasi <i>loading</i> , lalu menampilkan rekomendasi wisata	Berhasil
	Menghapus semua lokasi dari wisata favorit	Menghapus <i>section</i> “Rekomendasi Wisata”	Menghapus <i>section</i> “Rekomendasi Wisata”	Berhasil
Halaman Eksplorasi	Menekan salah satu tombol yang bertuliskan masing-masing kategori wisata	Menampilkan lokasi wisata berdasarkan kategori yang dipilih	Menampilkan lokasi wisata berdasarkan kategori yang dipilih	Berhasil
Halaman Favorit	Buka detail lokasi wisata, tandai lokasi wisata sebagai favorit, kembali ke halaman favorit	Menampilkan lokasi wisata yang ditandai sebagai favorit	Menampilkan lokasi wisata yang ditandai sebagai favorit	Berhasil
	Menghapus semua lokasi dari wisata favorit	Menampilkan pesan bahwa tidak ada wisata yang ditandai sebagai favorit	Menampilkan pesan bahwa tidak ada wisata yang ditandai sebagai favorit	Berhasil
Halaman Detail Lokasi	Menekan ikon hati bergaris tepi merah	Merubah ikon menjadi ikon hati dengan warna merah dan menampilkan pesan “Ditambahkan ke Favorit”	Merubah ikon menjadi ikon hati dengan warna merah dan menampilkan pesan “Ditambahkan ke Favorit”	Berhasil
	Menekan ikon hati berwarna merah	Merubah ikon menjadi ikon hati dengan garis tepi berwarna merah dan menampilkan pesan “Dihapus dari Favorit”	Merubah ikon menjadi ikon hati dengan garis tepi berwarna merah dan menampilkan pesan “Dihapus dari Favorit”	Berhasil
	Menekan tombol “Menuju ke Lokasi”	Menampilkan halaman peta dan navigasi lokasi tersebut	Menampilkan halaman peta dan navigasi lokasi tersebut	Berhasil
Halaman Peta dan Navigasi	Menekan tombol dengan ikon manusia	Mengubah fokus peta ke lokasi pengguna	Mengubah fokus peta ke lokasi pengguna	Berhasil
	Menekan tombol dengan ikon pin lokasi	Mengubah fokus peta ke lokasi wisata	Mengubah fokus peta ke lokasi wisata	Berhasil

Implementasi Sistem Rekomendasi

Sistem rekomendasi dibangun menggunakan *framework* Flask, oleh karena itu cara agar aplikasi wisata Mangunan dapat terhubung dengan sistem rekomendasi adalah menggunakan API. Setelah menerima *request* API dari aplikasi maka sistem rekomendasi akan melakukan semua proses yang sudah dijelaskan sebelumnya dari persiapan data hingga perhitungan *cosine similarity*, lalu hasilnya akan diberikan ke pengguna atau aplikasi dalam bentuk JSON.

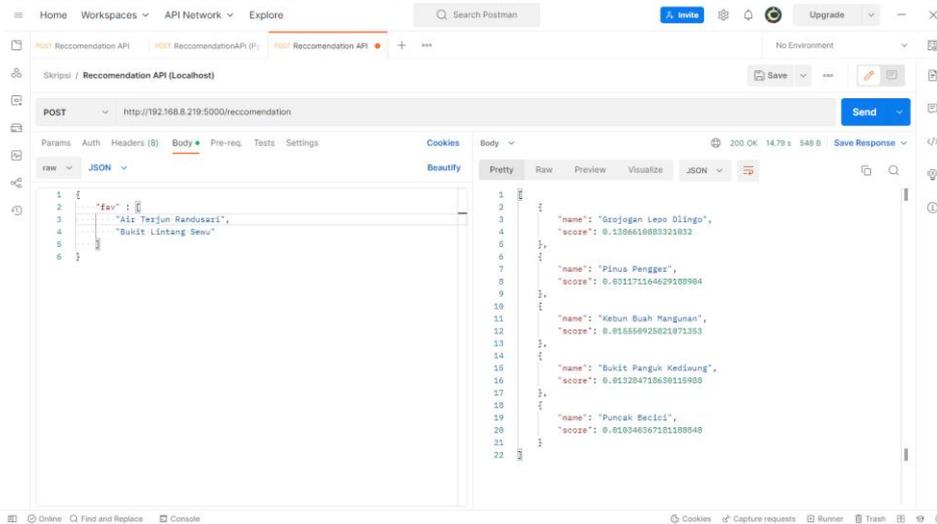
Untuk dapat melakukan *request* API pada sistem rekomendasi wisata Mangunan diperlukan URL, *method*, dan parameter yang telah ditentukan. Penjelasan dari API sistem rekomendasi wisata Mangunan dapat dilihat pada Tabel 8.

Tabel 8 *Request* API sistem rekomendasi wisata Mangunan

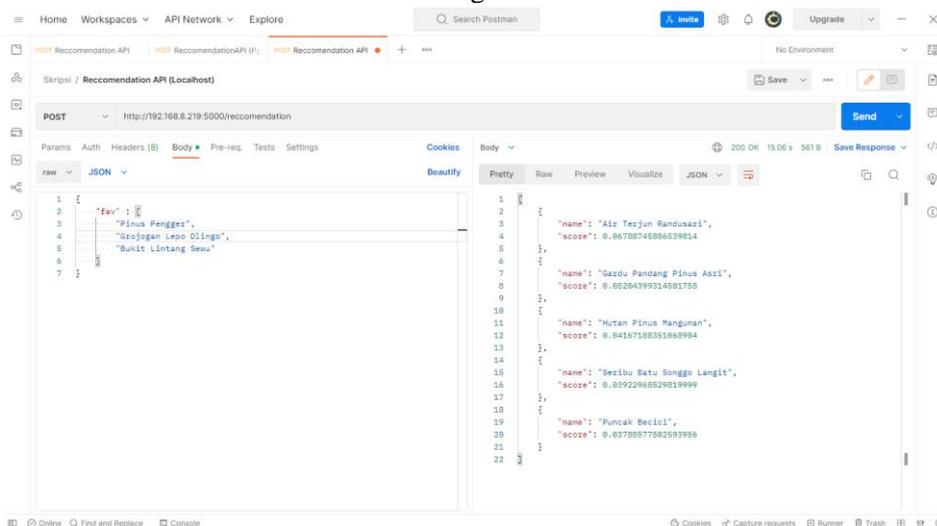
Method	: POST
URL	: <a href="http://<domain-name>/reccomendation">http://<domain-name>/reccomendation
	Keterangan: Karna sistem rekomendasi berjalan di <i>localhost</i> maka <domain-name> diisi dengan IP sistem. Dalam kasus ini menjadi http://192.168.43.83/reccomendation
	{

Body :	<pre>"fav" : [wisata_favorit] }</pre>
	<p>Contoh:</p> <pre>{ "fav" : ["Air Terjun Randusari", "Watu Goyang"] }</pre>

Setelah melakukan *request* API, sistem rekomendasi wisata Mangunan akan memberikan 5 rekomendasi wisata beserta nilai *cosine similarity* dalam bentuk JSON. Hasil dari *request* API sistem rekomendasi wisata Mangunan dapat dilihat pada Gambar 7 dan Gambar 8.



Gambar 7 Hasil rekomendasi dengan data wisata favorit berisi Air Terjun Randusari dan Bukit Lintang Sewu



Gambar 8 Hasil rekomendasi dengan data wisata favorit berisi Pinus Pengger, Grojogan Lepo Dlingo, dan Bukit Lintang Sewu

Penghitungan Precision

Pengujian akurasi dari sistem rekomendasi dilakukan menggunakan metode *precision* dengan menghitung perbandingan antara jumlah hasil yang relevan dengan jumlah hasil keseluruhan. *Precision* biasanya digunakan untuk mengukur kinerja sistem. *Precision* juga dapat dikatakan sebagai perbandingan tingkat ketepatan antara informasi yang diminta oleh pengguna dengan jawaban yang diberikan oleh sistem. Untuk menghitung nilai *precision* dapat dilakukan dengan Persamaan (5).

$$Precision = \frac{TP}{FP + TP} \cdot 100\% \quad (5)$$

Dimana TP adalah *True Positive* (data relevan yang diambil dalam sistem rekomendasi), FP adalah *False Positive* (data tidak relevan yang diambil dalam sistem rekomendasi). Melalui perhitungan ini, sistem rekomendasi dapat diukur performanya berdasarkan *precision* [11].

Dalam pengujian ini akan dilakukan perhitungan dengan menggunakan 10 sampel data wisata favorit. Pengujian dilakukan secara manual dengan menganalisis rekomendasi yang diberikan dengan data sampel. Semakin banyak rekomendasi wisata yang memiliki kemiripan dengan data wisata favorit maka skor *precision* semakin tinggi. Hasil pengujian *precision* sistem rekomendasi wisata Mangunan dapat dilihat pada Tabel 9.

Tabel 9 Pengujian *precision*

No	Data uji (data wisata favorit)	Jumlah data relevan	<i>Precision</i>
1.	• Air Terjun Randusari	3/5	60%
2.	• Air Terjun Randusari • Bukit Lintang Sewu	5/5	100%
3.	• Air Terjun Randusari • Bukit Lintang Sewu • Watu Goyang	5/5	100%
4.	• Bukit Lintang Sewu • Watu Goyang	5/5	100%
5.	• Watu Goyang	5/5	100%
6.	• Hutan Pinus Mangunan	5/5	100%
7.	• Tebing Watu Mabur • Hutan Pinus Mangunan • Grojogan Lepo Dlingo • Watu Goyang	5/5	100%
8.	• Tebing Watu Mabur • Grojogan Lepo Dlingo • Watu Goyang	5/5	100%
9.	• Tebing Watu Mabur • Watu Goyang	5/5	100%
10.	• Bukit Mojo Gumelem • Seribu Batu Songgo Langit	5/5	100%
Rata-rata <i>precision</i>			96%

Hasil dari pengujian sistem rekomendasi wisata Mangunan menggunakan algoritma TF-IDF dan *cosine similarity* menghasilkan rata-rata nilai *precision* sebesar 96%. Hasil tersebut menunjukkan bahwa kemampuan sistem dalam memberikan rekomendasi berdasarkan deskripsi dari tiap lokasi wisata sangat tinggi.

4. KESIMPULAN

Penelitian ini menghasilkan aplikasi wisata Mangunan berbasis android menggunakan Dart dan sistem rekomendasi wisata Mangunan menggunakan algoritma TF-IDF dan *cosine similarity*. Aplikasi wisata Mangunan dapat menampilkan wisata yang ada di daerah Mangunan dan juga menampilkan peta dan navigasi menggunakan Mapbox API. Lalu sistem rekomendasi dapat diimplementasikan ke aplikasi wisata Mangunan dan dapat memberikan 5 rekomendasi wisata berdasarkan data wisata favorit.

Berdasarkan hasil pengujian *precision* dengan menggunakan 10 data sampel menunjukkan bahwa sistem rekomendasi wisata Mangunan memiliki akurasi yang sangat tinggi dengan nilai *precision* mencapai 96%.

5. SARAN

Melalui penelitian yang telah dilakukan ini terdapat beberapa kekurangan yang dapat dijadikan pertimbangan untuk membuat pengembangan yang lebih baik. Saran yang dapat diberikan oleh penulis adalah aplikasi yang menggunakan framework Flutter dapat diimplementasikan untuk iOS dan web. Berikutnya implementasi sistem rekomendasi menggunakan library TensorFlow Lite yang dapat mempercepat proses pada sistem rekomendasi karena berjalan secara backend di aplikasi tersebut.

DAFTAR PUSTAKA

- [1] “JAGAD MANGUNAN: Menjangkau Mangunan,” Nov. 24, 2022. <https://visitingjogja.jogjaprov.go.id/37732/jagad-mangunan-menjangkau-mangunan/> (accessed Jan. 17, 2023).
- [2] Nofendy, “16 Tempat Wisata di Dlingo 2022, Mangunan Bantul yang Bagus Dan Wajib Dikunjungi,” Mar. 09, 2022. <https://eksotisjogja.com/tempat-wisata-dlingo-mangunan-bantul/> (accessed Jan. 17, 2023).
- [3] N. Adisty, “Mengulik Perkembangan Penggunaan Smartphone di Indonesia,” Nov. 05, 2022. <https://goodstats.id/article/mengulik-perkembangan-penggunaan-smartphone-di-indonesia-sT2LA> (accessed Jan. 17, 2023).
- [4] M. R. Munzir and D. Iriadi, “Aplikasi Objek Wisata Berbasis Android di Kota Pekanbaru,” *Jurnal Ilmiah Rekayasa dan Manajemen Sistem Informasi*, vol. 7, pp. 1–10, 2021.
- [5] S. Abraham and Y. D. Rahayu, “Sistem Rekomendasi Artikel Berita Menggunakan Metode K-Nearest Neighbor Berbasis Website,” Universitas Muhammadiyah Jember, 2017.
- [6] W. W. Song and S. Barakat, “Preprocessing Method Comparison and Model Tuning for Natural Language Data,” Dalarna University, 2020.
- [7] M. Nurjannah, Hamdani, and I. F. Astuti, “Penerapan Algoritma Term Frequency-Inverse Document Frequency (TF-IDF) untuk Text Mining,” *Jurnal Informatika Mulawarman*, vol. 8, No. 3, pp. 110–113, Sep. 2013.
- [8] G. Adomavicius and A. Tuzhilin, “Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions,” *IEEE Trans Knowl Data Eng*, vol. 17, no. 6, pp. 734–749, Jun. 2005, doi: 10.1109/TKDE.2005.99.
- [9] D. Kurniadi, S. F. C. Haviana, and A. Novianto, “Implementasi Algoritma Cosine Similarity pada sistem arsip dokumen di Universitas Islam Sultan Agung,” *Jurnal Transformatika*, vol. 17, no. 2, p. 124, Jan. 2020, doi: 10.26623/transformatika.v17i2.1613.
- [10] A. R. F. Rabbani and A. R. Pratama, “Aplikasi Sistem Jemput Sampah Berbasis Android untuk Rumah Kos dan Area Sekitar Kampus,” *Jurnal Sains dan Informatika*, vol. 7, no. 1, 2021, doi: 10.34128/jsi.v7i1.299.
- [11] M. R. A. Zayyad, “Sistem Rekomendasi Buku menggunakan Metode Content Based Filtering,” Universitas Islam Indonesia, Yogyakarta, 2021.