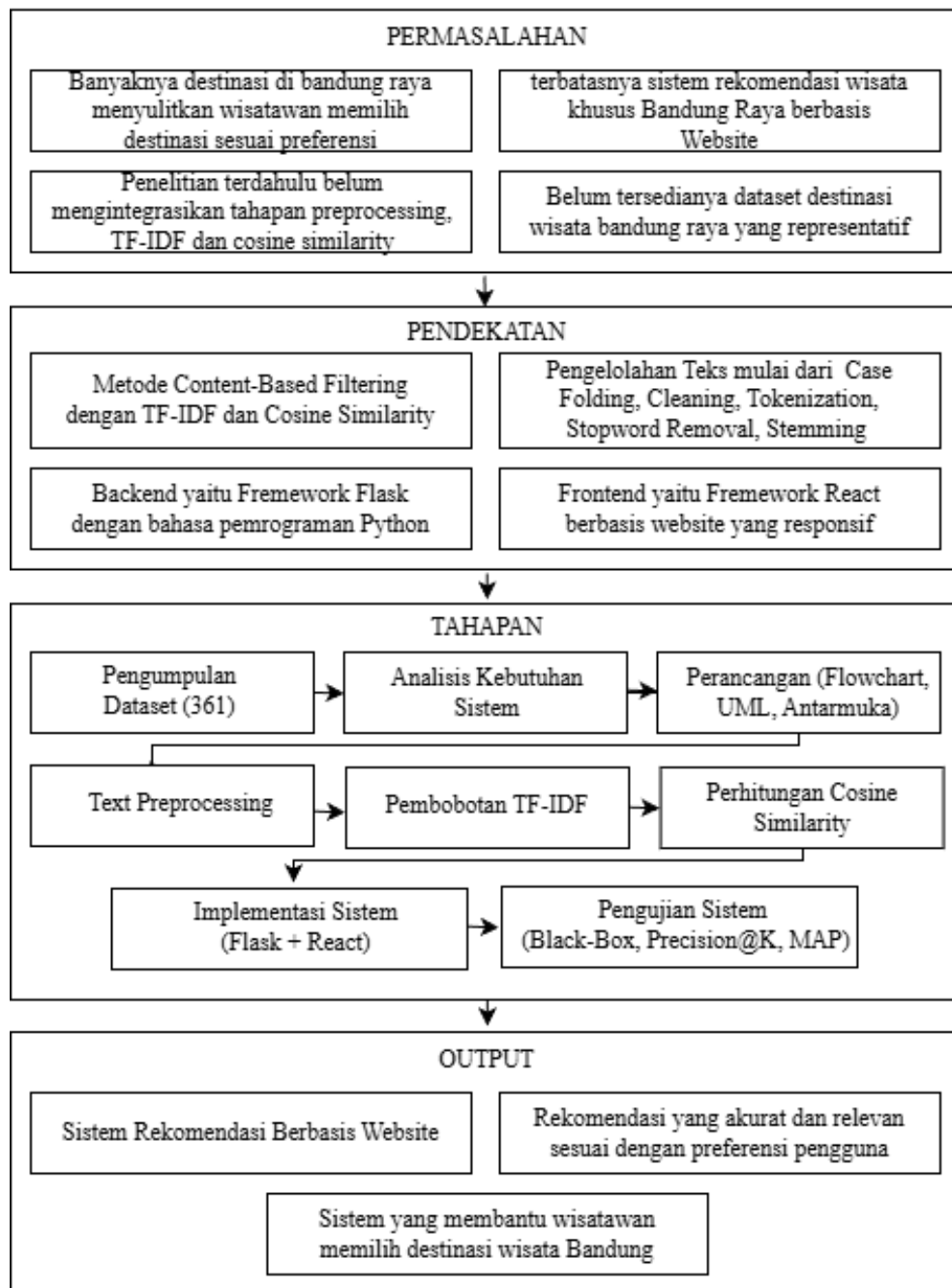


BAB III

METODELOGI PENELITIAN

3.1 Kerangka Berpikir

Kerangka berpikir adalah representasi konseptual mengenai cara suatu penelitian dilakukan, mulai dari permasalahan yang melatarbelakangi penelitian hingga hasil yang ingin dicapai. Kerangka berpikir disusun secara sistematis sehingga setiap tahapan saling berhubungan antara satu dengan yang lainnya. Dalam penelitian ini, kerangka berpikir digunakan sebagai landasan dalam menentukan alur penelitian yang akan dilakukan. Kerangka berpikir penelitian ini terdiri atas empat bagian utama, yaitu permasalahan, pendekatan, tahapan, dan output, yang dapat dilihat pada Gambar 3.1 berikut.



Gambar 3. 1 Kerangka Berpikir Penelitian

Berdasarkan Gambar 3.1, penelitian ini berangkat dari empat permasalahan utama yang saling berkaitan. Banyaknya destinasi wisata di Bandung Raya yang tersebar di Kota Bandung, Kabupaten Bandung, dan Kabupaten Bandung Barat menyulitkan wisatawan dalam memilih destinasi yang sesuai dengan preferensi mereka. Kondisi tersebut diperparah oleh masih terbatasnya sistem rekomendasi

wisata khusus wilayah Bandung Raya yang berbasis *website* dan memanfaatkan pendekatan *Machine Learning*, padahal wilayah ini memiliki jumlah destinasi yang banyak dengan karakteristik yang beragam. Di sisi lain, penelitian terdahulu umumnya masih menekankan pada satu tahapan tertentu dan belum mengintegrasikan seluruh tahapan *text Preprocessing*, TF-IDF, dan *Cosine Similarity* ke dalam satu sistem rekomendasi wisata berbasis *website* yang utuh. Selain itu, dataset destinasi wisata Bandung Raya yang representatif dan siap olah untuk keperluan pengembangan sistem rekomendasi juga belum tersedia. Keempat permasalahan tersebut menjadi dasar dilakukannya penelitian ini.

Untuk menjawab permasalahan tersebut, penelitian ini menggunakan empat pendekatan yang saling melengkapi. Metode *Content-Based Filtering* dengan kombinasi teknik TF-IDF dan *Cosine Similarity* digunakan sebagai inti mesin rekomendasi karena metode ini memberikan rekomendasi berdasarkan kemiripan karakteristik antar destinasi, sehingga tetap dapat berjalan meskipun data interaksi pengguna terbatas. Pengolahan teks Bahasa Indonesia dilakukan melalui lima tahap *Preprocessing*, yaitu *Case Folding*, *cleaning*, *tokenization*, *stopword removal*, dan *stemming* menggunakan *library* Sastrawi untuk membersihkan dan menyederhanakan teks deskripsi destinasi. Pada sisi *backend*, sistem dibangun menggunakan *framework Flask* berbasis bahasa pemrograman *Python* untuk mengolah data dan menyediakan *endpoint API*. Sementara itu, pada sisi *Frontend*, sistem dibangun menggunakan *framework React* sehingga sistem dapat diakses melalui *website* yang responsif dan lintas perangkat.

Tahapan penelitian menggambarkan proses pembangunan sistem yang dilakukan secara sistematis dengan mengintegrasikan dua metode, yaitu metode *Waterfall* sebagai metode pengembangan perangkat lunak dan metode *Content-Based Filtering* sebagai inti mesin rekomendasi. Penelitian dimulai dengan pengumpulan dataset destinasi wisata Bandung Raya sebanyak 331 destinasi yang diperoleh dari berbagai sumber publik dan Dinas Pariwisata dan Kebudayaan Provinsi Jawa Barat. Tahap berikutnya adalah analisis kebutuhan sistem dan perancangan sistem sebagai bagian dari metode *Waterfall*, yang mencakup

penyusunan *flowchart*, diagram UML, dan rancangan antarmuka pengguna. Setelah rancangan sistem selesai, dilanjutkan dengan penerapan metode *Content-Based Filtering* yang terdiri atas *Text Preprocessing* untuk membersihkan teks deskripsi destinasi, pembobotan kata menggunakan TF-IDF untuk merepresentasikan deskripsi destinasi sebagai vektor numerik, dan perhitungan *Cosine Similarity* untuk mengukur tingkat kemiripan antar destinasi. Hasil dari metode *Content-Based Filtering* kemudian diintegrasikan pada tahap implementasi sistem berbasis *website* menggunakan *Flask* dan *React*. Tahap terakhir adalah pengujian sistem yang dilakukan menggunakan *Black-Box Testing* untuk menguji fungsionalitas serta metrik *Precision@K* dan *Mean Average Precision* (MAP) untuk mengukur akurasi rekomendasi.

Bagian akhir kerangka berpikir menggambarkan hasil yang diharapkan dari penelitian ini, yaitu terbentuknya sistem rekomendasi destinasi wisata Bandung Raya berbasis *website* yang dapat diakses secara luas oleh wisatawan. Sistem yang dibangun diharapkan mampu memberikan rekomendasi destinasi wisata yang akurat dan relevan sesuai dengan preferensi pengguna berdasarkan kemiripan karakteristik antar destinasi, sehingga dapat membantu wisatawan dalam memilih destinasi wisata Bandung Raya secara lebih efektif dan efisien. Pada akhirnya, penelitian ini diharapkan memberikan kontribusi bagi pengembangan sistem rekomendasi di bidang pariwisata, khususnya untuk wilayah Bandung Raya.

3.2 Metode Penelitian

Bagian ini menjelaskan pendekatan dan metode yang digunakan dalam pelaksanaan penelitian. Penjelasan meliputi jenis penelitian, metode pengembangan sistem, serta metode pengumpulan data untuk mendukung jalannya penelitian.

3.2.1 Tempat Penelitian

Dilaksanakan di Dinas Pariwisata dan Kebudayaan Provinsi Jawa Barat yang beralamat di Jl. R.E. Martadinata No. 209, Cihapit, Bandung Wetan, Kota Bandung, Jawa Barat 40114. Dinas Pariwisata dan Kebudayaan Provinsi Jawa Barat

merupakan perangkat daerah di Provinsi Jawa Barat yang mengurus urusan pemerintah di bidang pariwisata dan kebudayaan, termasuk pengelolaan data dan informasi destinasi wisata di seluruh wilayah Jawa Barat.

Pemilihan lokasi ini didasarkan pada relevansinya sebagai instansi pemerintah yang membidangi pengelolaan dan pengembangan pariwisata di Provinsi Jawa Barat, khususnya wilayah Bandung Raya yang menjadi fokus penelitian ini. Instansi ini mendukung ketersediaan data destinasi wisata yang dibutuhkan sebagai dataset dalam pembangunan sistem rekomendasi. Surat balasan izin penelitian dari Dinas Pariwisata dan Kebudayaan Provinsi Jawa Barat dapat dilihat pada Lampiran 2.

3.2.2 Jenis Penelitian

Penelitian terapan bertujuan untuk menerapkan teori atau metode yang sudah ada ke dalam suatu permasalahan nyata, dengan harapan dapat menghasilkan produk atau solusi yang bermanfaat secara langsung (Prasetyo *et al.*, 2021). Pada penelitian ini, metode *Content-Based Filtering* dengan kombinasi TF-IDF dan *Cosine Similarity* diterapkan untuk membangun sistem rekomendasi destinasi wisata Bandung Raya berbasis *website*.

Kuantitatif dipilih karena penelitian ini menggunakan data berupa angka, terutama pada proses perhitungan bobot TF-IDF dan nilai *Cosine Similarity*, serta pengukuran akurasi sistem yang menggunakan metrik *Precision@K* dan *Mean Average Precision* (MAP). Selain itu, hasil penelitian juga dievaluasi secara objektif melalui pengujian fungsional dengan menerapkan metode *Black-Box Testing*.

3.2.3 Metode *Content-Based Filtering*

Pendekatan utama adalah *Content-Based Filtering*, yaitu yang merekomendasikan item ke pengguna berdasarkan kemiripan karakteristik antar item (Christyawan *et al.*, 2024). Pada penelitian ini, konten yang digunakan berupa atribut teks dari setiap destinasi wisata, terutama deskripsi, kategori, dan wilayah, sehingga destinasi yang memiliki kemiripan konten dengan kata kunci atau preferensi yang dimasukkan pengguna akan direkomendasikan oleh sistem.

Content-Based Filtering dipilih karena relevan dengan karakteristik data pada penelitian ini yang berupa informasi konten destinasi wisata, serta tidak bergantung pada data interaksi atau penilaian dari pengguna lain sebagaimana pada metode *Collaborative Filtering* (Faurina & Sitanggang, 2023). Dengan demikian, sistem tetap dapat memberikan rekomendasi meskipun belum memiliki data riwayat pengguna, sehingga cocok diterapkan pada sistem rekomendasi destinasi wisata Bandung Raya yang dibangun.

Penerapan metode *Content-Based Filtering* pada penelitian ini dilakukan melalui beberapa tahapan. Pertama, data teks destinasi melalui tahap *Preprocessing* yang meliputi *Case Folding* buat menyamakan huruf jadi kecil, *tokenizing* untuk membagi teks jadi kata, penghapusan *stopword* membuang kata yang tidak bermakna, serta *stemming* menggunakan *library* Sastrawi untuk mengonversi kata berimbuhan menjadi kata dasarnya. Kedua, hasil *Preprocessing* direpresentasikan ke dalam bentuk vektor pembobotan kata menggunakan teknik TF-IDF. Ketiga, tingkat kemiripan antara kata kunci yang dimasukkan pengguna dengan setiap destinasi dihitung menggunakan *Cosine Similarity*. Keempat, sistem mengurutkan destinasi berdasarkan nilai kemiripan tertinggi dan menampilkannya sebagai daftar rekomendasi teratas (Top-K) kepada pengguna.

3.2.4 Metode Pengembangan Sistem

Pengembangan sistem rekomendasi destinasi wisata pada penelitian ini menggunakan metode *Waterfall*. Metode ini dipilih karena memiliki alur yang terstruktur dan sistematis, di mana setiap tahapan wajib diselesaikan sebelum melanjutkan ke tahapan selanjutnya (Faurina & Sitanggang, 2023).

Adapun lima tahapan metode *Waterfall* yang digunakan pada penelitian ini adalah sebagai berikut:

a. Analisis Kebutuhan (*Requirement Analysis*)

Dilakukan identifikasi permasalahan, pengumpulan kebutuhan sistem, baik kebutuhan fungsional maupun non-fungsional, serta studi terhadap penelitian-penelitian terdahulu yang relevan. Hasil dari tahap ini menjadi acuan untuk tahap perancangan.

b. Desain Sistem (*System Design*)

Fase desain mencakup perancangan struktur sistem, perancangan diagram UML yang terdiri atas *Use Case Diagram*, *Activity Diagram*, *Class Diagram*, dan *Sequence Diagram*, perancangan *flowchart* sistem, serta perancangan antarmuka pengguna. Pada tahap desain ini, alur kerja metode *Content-Based Filtering* dirancang dan dituangkan ke dalam *flowchart* sistem rekomendasi. Rancangan tersebut menggambarkan tahapan proses metode *Content-Based Filtering* secara berurutan, dimulai dari pembacaan dataset, tahap *Text Preprocessing* terhadap deskripsi destinasi, pembobotan kata menggunakan TF-IDF, perhitungan kemiripan antar destinasi menggunakan *Cosine Similarity*, hingga penyusunan hasil rekomendasi berdasarkan nilai kemiripan tertinggi. Dengan demikian, *flowchart* yang dirancang pada tahap desain menjadi representasi langkah-langkah penerapan metode *Content-Based Filtering* yang kemudian diwujudkan pada tahap implementasi. Hasil rancangan pada tahap ini menjadi dasar implementasi sistem pada tahap berikutnya.

c. Implementasi Sistem (*Implementation*)

Sisi *backend* dibangun menggunakan *framework* Flask dengan bahasa pemrograman Python, sedangkan sisi *Frontend* dibangun menggunakan *framework* React. Pada tahap ini juga dilakukan integrasi seluruh komponen sistem, mulai dari proses *Preprocessing*, perhitungan TF-IDF, *Cosine Similarity*, hingga tampilan rekomendasi pada antarmuka *website*.

d. Pengujian (*Testing*)

Sistem yang telah dibangun kemudian diuji untuk memastikan bahwa seluruh fitur berjalan sesuai dengan kebutuhan. Pengujian fungsional dilakukan dengan metode *Black-Box Testing*, sedangkan pengujian akurasi rekomendasi dilakukan menggunakan metrik *Precision@K* dan *Mean Average Precision* (MAP).

e. Pemeliharaan (*Maintenance*)

Pada penelitian ini, tahap pemeliharaan dilakukan secara terbatas dengan melakukan perbaikan *bug* dan penyesuaian sistem apabila ditemukan

kekurangan saat pengujian. Tahap pemeliharaan jangka panjang berada di luar ruang lingkup penelitian.

3.2.5 Metode Pengumpulan Data

Pengumpulan data dilaksanakan dalam dua cara, yakni:

a. Studi Literatur

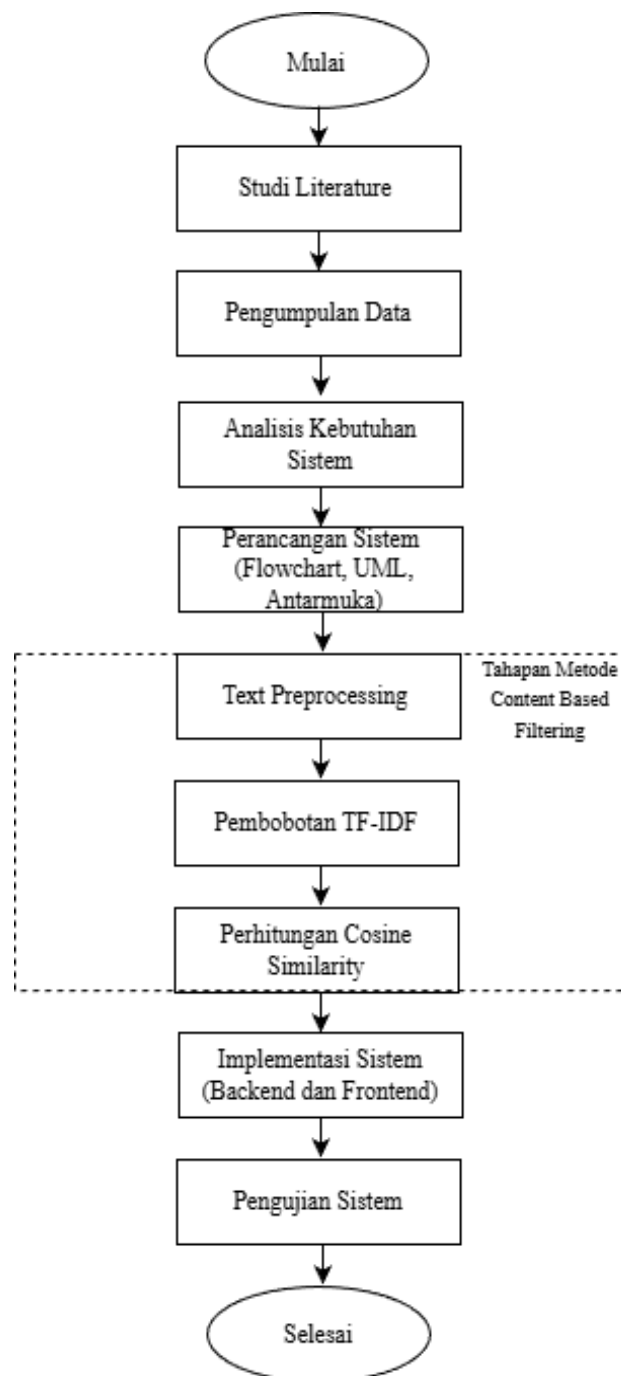
dilaksanakan dengan mempelajari teori yang berhubungan dengan penelitian, Literatur diperoleh dari jurnal, buku, dan penelitian terdahulu yang relevan sebagai dasar dalam membangun sistem.

b. Pengumpulan *Dataset*

Pengumpulan data dalam studi ini dilaksanakan dengan memanfaatkan data sekunder berupa dataset destinasi wisata di wilayah Bandung Raya yang meliputi Kota Bandung, Kabupaten Bandung, dan Kabupaten Bandung Barat. Dataset diperoleh dari berbagai sumber, seperti situs web pariwisata, artikel terkait, serta dataset resmi dari Dinas Pariwisata dan Kebudayaan Provinsi Jawa Barat (Disparbud Jabar). Data tersebut digunakan sebagai input utama dalam proses *text Preprocessing*, pembobotan TF-IDF, hingga perhitungan *Cosine Similarity* pada sistem rekomendasi.

3.3 Prosedur Penelitian

Prosedur penelitian merupakan rangkaian tahapan yang dilakukan secara sistematis untuk mencapai tujuan penelitian. Pada penelitian ini, prosedur penelitian disusun dalam sembilan tahapan yang saling berurutan, mulai dari studi literatur hingga pengujian sistem. Adapun prosedur penelitian dapat dilihat pada Gambar 3.2 berikut:



Gambar 3. 2 Prosedur Penelitian

Berdasarkan diagram pada Gambar 3.2, berikut penjelasan dari setiap tahapan prosedur penelitian:

3.3.1 Studi Literatur

Studi literatur dilakukan sebagai langkah awal penelitian untuk memperoleh pemahaman mendalam mengenai topik yang dikaji. Pada tahap ini, dikumpulkan berbagai referensi yang relevan dengan sistem rekomendasi, metode *Content-Based Filtering*, teknik TF-IDF dan *Cosine Similarity*, pengolahan teks Bahasa Indonesia, serta teknologi pendukung seperti *framework* Flask dan React. Hasil studi literatur menjadi landasan teori yang digunakan pada Bab II.

3.3.2 Pengumpulan Data

Dilaksanakan dengan menghimpun informasi destinasi wisata di wilayah Bandung Raya yang meliputi Kota Bandung, Kabupaten Bandung, dan Kabupaten Bandung Barat. Data yang digunakan dalam penelitian ini merupakan data sekunder yang diperoleh dari berbagai sumber publik, seperti situs web informasi pariwisata, artikel wisata, serta dataset resmi dari Dinas Pariwisata dan Kebudayaan Provinsi Jawa Barat (Disparbud Jabar). Seluruh data tersebut digunakan sebagai dasar dalam penyusunan dataset destinasi wisata yang akan diolah pada sistem rekomendasi.

Data yang dikumpulkan kemudian diseleksi berdasarkan kesesuaian dengan cakupan penelitian. Destinasi yang berada di luar wilayah Kota Bandung, Kabupaten Bandung, dan Kabupaten Bandung Barat tidak dimasukkan ke dalam *dataset*. Selain itu, data yang memiliki informasi ganda atau tidak memiliki deskripsi yang cukup jelas juga disaring agar *dataset* yang digunakan lebih konsisten dan layak untuk diproses.

Secara lebih rinci, data awal yang berhasil dihimpun berjumlah 377 destinasi. Setelah melalui tahap seleksi, sebanyak 46 data dihapus karena tidak memenuhi kriteria, yaitu berupa data duplikat, yakni destinasi yang tercatat lebih dari satu kali dengan penulisan nama berbeda seperti "Situ Patenggang" dan "Situ Patengan", destinasi yang berada di luar wilayah Bandung Raya, serta destinasi yang tidak memiliki deskripsi yang memadai. Setelah proses penyaringan tersebut, diperoleh dataset akhir sebanyak 331 destinasi wisata yang siap untuk diolah.

Dataset akhir yang digunakan dalam penelitian ini berjumlah 331 destinasi wisata. Setiap data destinasi memiliki empat atribut utama, yaitu nama destinasi, deskripsi, kategori, dan wilayah. Atribut deskripsi digunakan sebagai data utama dalam proses rekomendasi karena berisi informasi tekstual mengenai karakteristik destinasi wisata. Atribut kategori dan wilayah digunakan sebagai informasi pendukung untuk menampilkan detail destinasi serta membantu pengguna dalam memahami jenis dan lokasi destinasi yang direkomendasikan.

Penentuan kategori destinasi wisata pada penelitian ini mengacu pada klasifikasi jenis daya tarik wisata yang ditetapkan dalam Undang-Undang Republik Indonesia Nomor 10 Tahun 2009 tentang Kepariwisata, yang membagi daya tarik wisata menjadi wisata alam, wisata budaya, dan wisata buatan/hasil karya manusia. Klasifikasi tersebut kemudian disesuaikan dengan karakteristik destinasi yang terdapat di wilayah Bandung Raya serta merujuk pada pengelompokan kategori yang digunakan pada dataset pariwisata dari Dinas Pariwisata dan Kebudayaan Provinsi Jawa Barat (Disparbud Jabar). Berdasarkan acuan tersebut, destinasi pada penelitian ini dikelompokkan ke dalam kategori-kategori seperti Wisata Alam, Budaya, Taman Hiburan, Cagar Alam, Pusat Perbelanjaan, dan kategori lainnya yang sesuai dengan kondisi destinasi di Bandung Raya.

Setelah data terkumpul dan diseleksi, dataset disimpan dalam bentuk file dengan format CSV (*Comma-Separated Values*) yang diberi nama `dataset_wisata_clean.csv`. Format CSV dipilih karena bersifat sederhana, ringan, dan mudah diolah menggunakan bahasa pemrograman Python melalui pustaka Pandas. Di dalam file tersebut, setiap baris mewakili satu destinasi wisata, sedangkan setiap kolom mewakili satu atribut. Struktur file terdiri dari lima kolom, yaitu `Place_Id` sebagai nomor identitas unik destinasi, `Place_Name` sebagai nama destinasi, `Description` sebagai deskripsi destinasi, `Category` sebagai kategori wisata, dan `City` sebagai wilayah. Dataset dalam bentuk file inilah yang kemudian dibaca oleh sistem dan diproses pada tahap *text Preprocessing*.

Tabel 3. 1 Ringkasan *Dataset* Penelitian

No	Komponen	Keterangan
1	Jenis data	Data sekunder
2	Objek data	Destinasi wisata Bandung Raya
3	Jumlah data	331 destinasi wisata
4	Cakupan wilayah	Kota Bandung, Kabupaten Bandung, dan Kabupaten Bandung Barat
5	Format <i>dataset</i>	CSV
6	Atribut <i>dataset</i>	Nama destinasi, deskripsi, kategori, dan wilayah
7	Atribut utama untuk rekomendasi	Deskripsi destinasi
8	Proses seleksi data	Penyaringan data berdasarkan wilayah, kelengkapan deskripsi, dan penghapusan data ganda
9	Penggunaan data	Sebagai input untuk <i>text Preprocessing</i> , TF-IDF, dan <i>Cosine Similarity</i>

3.3.3 Analisis Kebutuhan Sistem

Tahap analisis kebutuhan merupakan tahap pertama dari metode *Waterfall* yang bertujuan mengidentifikasi kebutuhan-kebutuhan sistem agar dapat berjalan sesuai tujuan penelitian. Kebutuhan sistem dibagi menjadi dua, yaitu kebutuhan fungsional dan kebutuhan non-fungsional. Kebutuhan fungsional mencakup fitur-fitur wajib yang harus tersedia pada sistem, seperti menampilkan daftar destinasi, mencari destinasi berdasarkan kata kunci, memfilter destinasi, menampilkan detail destinasi, dan memberikan rekomendasi destinasi serupa. Kebutuhan non-fungsional mencakup spesifikasi perangkat keras dan perangkat lunak yang digunakan dalam pengembangan sistem. Hasil analisis kebutuhan ini menjadi acuan untuk tahap perancangan sistem.

3.3.4 Perancangan Sistem

Tahap perancangan merupakan tahap kedua dari metode *Waterfall* yang bertujuan menghasilkan rancangan sistem sebelum diimplementasikan ke dalam kode program. Perancangan sistem pada penelitian ini terdiri atas tiga bagian, yaitu perancangan alur kerja sistem menggunakan *flowchart*, perancangan struktur dan perilaku sistem menggunakan diagram UML (*Use Case Diagram*, *Activity Diagram*, *Class Diagram*, dan *Sequence Diagram*), serta perancangan antarmuka pengguna (*user interface*). Flowchart pada tahap ini menggambarkan alur kerja

metode *Content-Based Filtering* secara berurutan, dimulai dari pembacaan dataset, *text Preprocessing*, pembobotan TF-IDF, perhitungan *Cosine Similarity*, hingga penyusunan hasil rekomendasi. Rancangan yang dihasilkan pada tahap ini menjadi dasar bagi tahap implementasi sistem.

3.3.5 *Text Preprocessing*

Setelah data terkumpul, tahap berikutnya adalah pengolahan teks atau *text Preprocessing*. Tahap ini bertujuan untuk menyederhanakan dan memurnikan teks deskripsi destinasi wisata agar dapat diolah lebih lanjut oleh algoritma. Proses *Preprocessing* yang dilakukan pada penelitian ini meliputi lima tahap, yaitu:

a. *Case Folding*

Yaitu mengonversi seluruh huruf pada teks menjadi huruf kecil (*lowercase*) sehingga sistem tidak membedakan kata yang memiliki perbedaan penggunaan huruf.

b. *Cleaning*

Yaitu menghapus karakter-karakter yang tidak diperlukan seperti tanda baca, angka, simbol, *emoji*, dan spasi berlebih.

c. *Tokenization*

Yaitu membagi teks menjadi bagian-bagian kecil berupa kata atau token.

d. *Stopword Removal*

Yaitu menghapus kata umum yang tidak ada makna penting dalam analisis.

e. *Stemming*

Yaitu mengubah kata berimbuhan menjadi bentuk dasarnya menggunakan *library* Sastrawi yang dirancang khusus untuk Bahasa Indonesia.

3.3.6 **Pembobotan TF-IDF**

Pada tahap ini, teks hasil *Preprocessing* diolah menggunakan TF-IDF yang memberikan bobot setiap kata dalam dokumen. Nilai TF menghitung frekuensi kemunculan suatu kata, sementara IDF mengukur tingkat kelangkaan kata pada keseluruhan. Bobot akhir setiap kata diperoleh dengan mengalikan nilai TF dan IDF.

Hasil dari tahap ini adalah representasi vektor numerik dari setiap deskripsi destinasi wisata yang siap digunakan untuk proses perhitungan kemiripan. Pada studi ini, pembobotan TF-IDF diterapkan dengan memanfaatkan library Scikit-learn melalui modul `TfidfVectorizer`.

3.3.7 Perhitungan *Cosine Similarity*

Setelah vektor TF-IDF terbentuk, tahap berikutnya adalah menghitung tingkat kemiripan antar destinasi menggunakan *Cosine Similarity*. Hasil perhitungan ini menghasilkan sebuah *matrix similarity* berukuran 331×331 yang berisi nilai kemiripan antar setiap pasangan destinasi. Matriks tersebut menjadi dasar dalam penyusunan rekomendasi pada sistem.

3.3.8 Implementasi Sistem

Tahap implementasi merupakan tahap penerjemahan rancangan ke dalam kode program. Sistem rekomendasi dibangun dalam bentuk *website* dengan arsitektur *client-server*. Sisi *backend* dibangun menggunakan *framework* Flask berbasis bahasa pemrograman Python, yang berfungsi untuk mengolah data, menjalankan perhitungan TF-IDF dan *Cosine Similarity*, serta menyediakan *endpoint* API. Sisi *Frontend* dibangun menggunakan *framework* React yang berfungsi untuk menampilkan antarmuka pengguna dan berkomunikasi dengan *backend* melalui REST API.

Komunikasi antara *Frontend* dan *backend* menggunakan format JSON. Ketika pengguna melakukan pencarian atau memilih destinasi, *Frontend* mengirim permintaan ke *backend*, kemudian *backend* memproses permintaan dan mengembalikan hasil rekomendasi dalam bentuk JSON yang ditampilkan kembali pada antarmuka pengguna.

3.3.9 Pengujian Sistem

Pengujian untuk menjamin sistem yang dibuat berjalan sesuai dan memberikan hasil rekomendasi yang akurat. Pengujian dilakukan menggunakan dua pendekatan, yaitu:

a. Pengujian Fungsional dengan *Black-Box Testing*

Dilaksanakan dengan mengevaluasi setiap fitur sistem berdasarkan masukan dan keluaran yang dihasilkan tanpa memperhatikan struktur kode internal. Setiap skenario pengujian dicocokkan dengan hasil yang diharapkan, lalu dikategorikan sebagai berhasil atau gagal.

b. Pengujian Akurasi dengan *Precision@K* dan MAP

Untuk mengukur kualitas rekomendasi yang dihasilkan oleh sistem. *Precision@K* menghitung proporsi rekomendasi relevan pada *Top-K* hasil, sedangkan *Mean Average Precision* (MAP) menghitung rata-rata *Average Precision* dari beberapa *query* uji.

Hasil pengujian kemudian dianalisis untuk menguraikan gambaran mengenai kinerja sistem secara keseluruhan, dari sisi fungsionalitas ataupun akurasi.

3.4 Rancangan Sistem

Bagian ini menjelaskan rancangan sistem rekomendasi destinasi wisata Bandung Raya yang akan dibangun. Penjelasan meliputi analisis kebutuhan, *flowchart*, perancangan UML, serta perancangan antarmuka. Rancangan ini menjadi acuan dalam tahap implementasi sistem.

3.4.1 Analisis Kebutuhan Sistem

Yaitu bertujuan menentukan kebutuhan-kebutuhan yang perlu dipenuhi oleh sistem agar dapat berjalan sesuai dengan tujuan penelitian. Kebutuhan sistem dibagi menjadi dua, yaitu kebutuhan fungsional dan kebutuhan non-fungsional.

a. Kebutuhan Fungsional

Menjabarkan fungsi-fungsi atau fitur wajib tersedia dalam sistem. Adapun kebutuhan fungsional dari sistem rekomendasi destinasi wisata Bandung Raya.

Tabel 3. 2 Kebutuhan Fungsional Sistem

No	Kode	Nama Fungsi	Deskripsi	Aktor
1	KF-01	Menampilkan Halaman Beranda	Sistem dapat menampilkan halaman beranda yang berisi informasi singkat tentang sistem,	Pengguna

No	Kode	Nama Fungsi	Deskripsi	Aktor
			fitur pencarian, dan daftar kategori destinasi wisata Bandung Raya	
2	KF-02	Menampilkan Daftar Destinasi	Sistem dapat menampilkan daftar seluruh destinasi wisata Bandung Raya yang tersedia pada <i>dataset</i> , sebanyak 331 destinasi, beserta informasi nama, foto, kategori, dan wilayah	Pengguna
3	KF-03	Mencari Destinasi Berdasarkan Kata Kunci	Sistem dapat menerima <i>input</i> berupa kata kunci pencarian dari pengguna dan menampilkan destinasi yang paling relevan berdasarkan perhitungan TF-IDF dan <i>Cosine Similarity</i>	Pengguna
4	KF-04	Memfilter Destinasi	Sistem dapat menyaring daftar destinasi berdasarkan kategori wisata (misal: Wisata Alam, Wisata Edukasi, Wisata Kuliner) dan wilayah bandung raya.	Pengguna
5	KF-05	Menampilkan Detail Destinasi	Sistem dapat menampilkan halaman detail destinasi yang berisi informasi lengkap mengenai destinasi terpilih, meliputi nama, deskripsi, foto, kategori, wilayah, dan informasi pendukung lainnya	Pengguna
6	KF-06	Memberikan Rekomendasi Destinasi Serupa	Sistem dapat menampilkan rekomendasi destinasi wisata serupa berdasarkan destinasi yang sedang dilihat pengguna, menggunakan metode <i>Content-Based Filtering</i> dengan perhitungan kemiripan vektor TF-IDF melalui <i>Cosine Similarity</i>	Sistem
7	KF-07	Menampilkan Hasil Rekomendasi Berdasarkan <i>Query</i>	Sistem dapat menampilkan daftar destinasi paling relevan berdasarkan yang telah dimasukkan, dengan urutan dari nilai kemiripan tertinggi ke terendah	Sistem
8	KF-08	Menampilkan Halaman Tentang Sistem	Sistem dapat menampilkan halaman tentang yang berisi penjelasan mengenai sistem, metode yang digunakan, serta informasi pengembang	Pengguna

b. Kebutuhan Non-Fungsional

Memaparkan bagian pendukung yang harus dipenuhi oleh sistem, seperti kebutuhan perangkat keras dan perangkat lunak. Kebutuhan ini ada dua yaitu *hardware* dan *software*.

1. Kebutuhan Perangkat Keras

Kebutuhan yang dimanfaatkan di pengembangan sistem sebagai berikut:

Tabel 3. 3 Kebutuhan Perangkat Keras

No	Perangkat Keras	Spesifikasi
1	Laptop	<i>Processor</i> Intel atau setara, RAM minimal 8 GB
2	<i>Storage</i>	Minimal 12 GB
3	<i>Display</i>	Resolusi minimal 1366 × 768
4	Koneksi Internet	Stabil, minimal 10 Mbps

2. Kebutuhan Perangkat Lunak

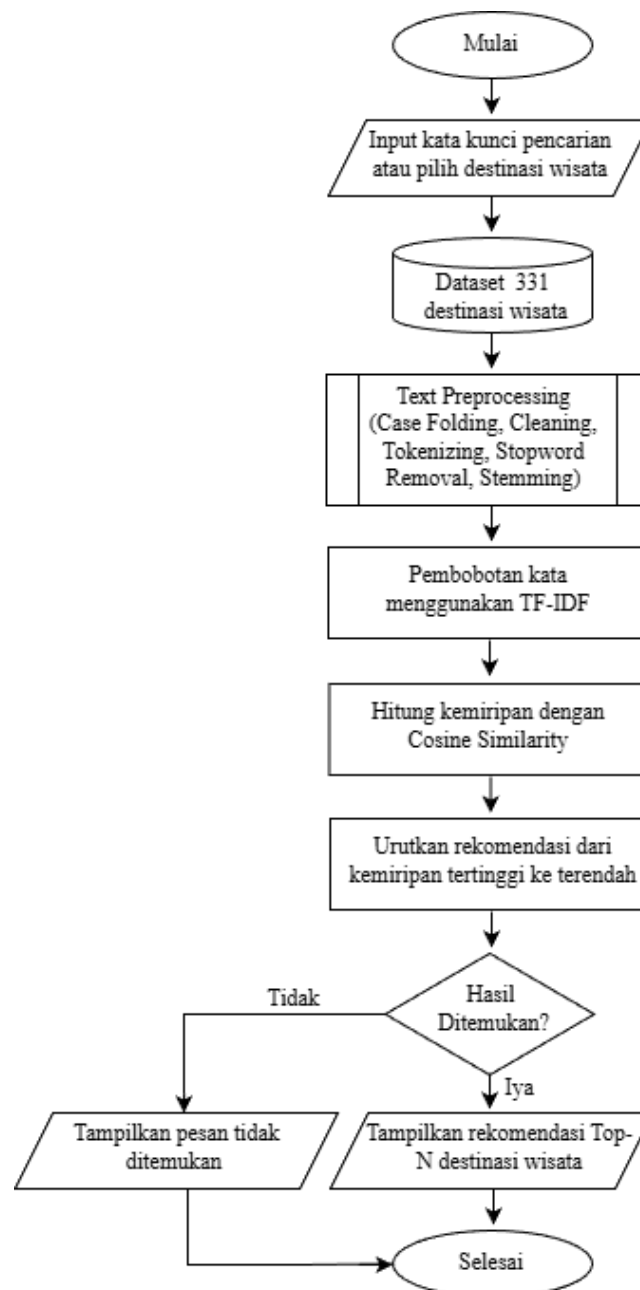
Yang digunakan pada pengembangan sistem dapat dilihat pada Tabel berikut:

Tabel 3. 4 Kebutuhan Perangkat Lunak

No	Perangkat Lunak	Keterangan
1	Sistem Operasi	Windows 10 / Windows 11
2	Python	Versi 3.12.0
3	Node.js	Versi 22.21.0
4	<i>Code Editor</i>	Visual Studio Code
5	<i>Framework Backend</i>	Flask
6	<i>Framework Frontend</i>	React
7	<i>Library Python</i>	Scikit-learn, Pandas, NumPy, Sastrawi, Flask-CORS
8	<i>Web Browser</i>	Google Chrome / Mozilla Firefox
9	<i>Tools Pendukung</i>	Jupyter Notebook, Postman, draw.io

3.4.2 Flowchart Sistem Rekomendasi

Flowchart sistem rekomendasi menggambarkan alur kerja sistem secara keseluruhan, mulai dari pengguna memasukkan kata kunci pencarian hingga sistem menampilkan rekomendasi destinasi wisata. *Flowchart* ini menjadi acuan dalam tahap implementasi agar alur logika sistem sesuai dengan rancangan. Alur sistem rekomendasi destinasi wisata Bandung Raya dapat dilihat pada Gambar 3.3 berikut:



Gambar 3. 3 Flowchart Sistem Rekomendasi

Berdasarkan Gambar tersebut alur kerja sistem rekomendasi destinasi wisata dijelaskan sebagai berikut:

a. Mulai

Sistem dijalankan ketika pengguna mengakses *website* sistem rekomendasi destinasi wisata.

b. Input Pencarian

Pengguna memasukkan kata kunci pencarian pada kolom *search*, atau memilih salah satu destinasi yang tersedia untuk mendapatkan rekomendasi destinasi serupa.

c. Memuat *Dataset*

Sistem memuat *dataset* yang berisi 331 destinasi wisata Bandung Raya beserta atribut nama, deskripsi, kategori, dan wilayah.

d. *Text Preprocessing*

Sistem melakukan pemrosesan teks terhadap *input* pengguna dan deskripsi destinasi, meliputi *Case Folding*, *cleaning*, *tokenization*, *stopword removal*, dan *stemming* menggunakan *library* Sastrawi.

e. Pembobotan TF-IDF

Teks hasil *Preprocessing* diberi bobot menggunakan TF-IDF untuk menentukan tingkat kepentingan setiap kata.

f. Perhitungan *Cosine Similarity*

Sistem menghitung tingkat kemiripan antara *input* pengguna dengan seluruh destinasi pada *dataset* menggunakan *Cosine Similarity*.

g. Pengurutan Hasil

Sistem mengurutkan destinasi berdasarkan nilai *Cosine Similarity* dari yang tertinggi ke terendah.

h. Pengecekan Hasil

Sistem memeriksa apakah hasil rekomendasi ditemukan. Jika ditemukan, sistem menampilkan rekomendasi. Jika tidak, sistem menampilkan pesan bahwa destinasi tidak ditemukan.

i. Menampilkan Rekomendasi

Sistem menampilkan Top-N destinasi wisata yang paling relevan kepada pengguna melalui antarmuka *website*.

j. Selesai

Proses sistem rekomendasi berakhir setelah hasil ditampilkan kepada pengguna.

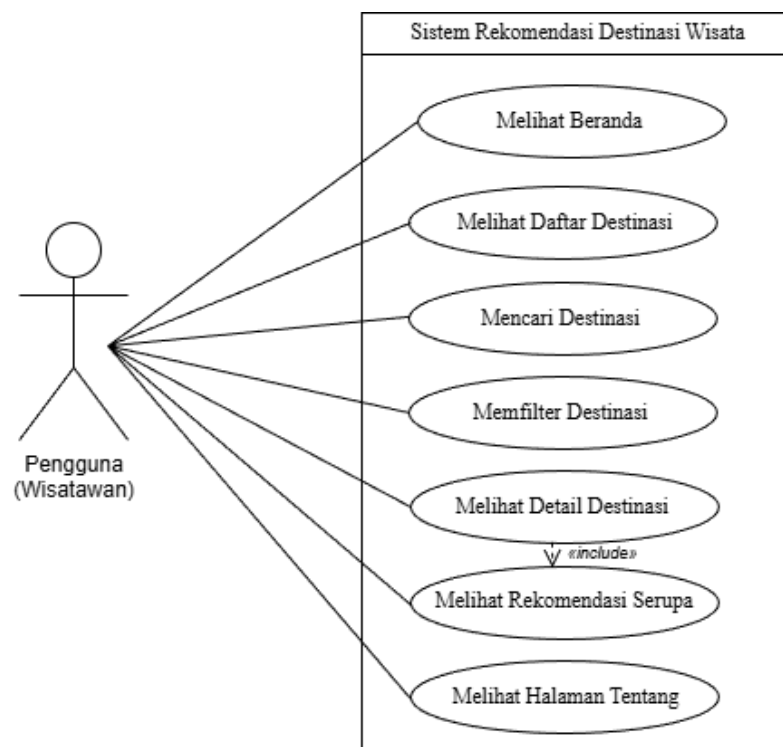
3.4.3 Perancangan UML

Perancangan sistem pada penelitian ini menggunakan *Unified Modeling Language* (UML) untuk menggambarkan rancangan sistem dari berbagai sudut pandang.

Pada sistem ini hanya terdapat satu aktor utama yaitu pengguna (wisatawan) yang berinteraksi dengan sistem untuk mendapatkan rekomendasi destinasi wisata. Sistem ini dirancang tanpa peran admin karena fokus penelitian adalah pada penerapan algoritma *Content-Based Filtering*, bukan pada manajemen konten. *Dataset* destinasi bersifat statis yang telah ada di tahap analisis kebutuhan, sehingga tidak diperlukan fitur *content management* atau *user management* dalam sistem.

a. Use Case Diagram

Use Case Diagram mengilustrasikan interaksi antara pengguna (*actor*) dengan sistem rekomendasi destinasi wisata. Pada sistem, hanya ada satu aktor yaitu wisatawan, yang dapat melakukan beberapa hal tanpa perlu melakukan *login*. *Use Case Diagram* sistem rekomendasi destinasi wisata Bandung Raya pada Gambar 3.4 berikut:



Gambar 3. 4 Use Case Diagram

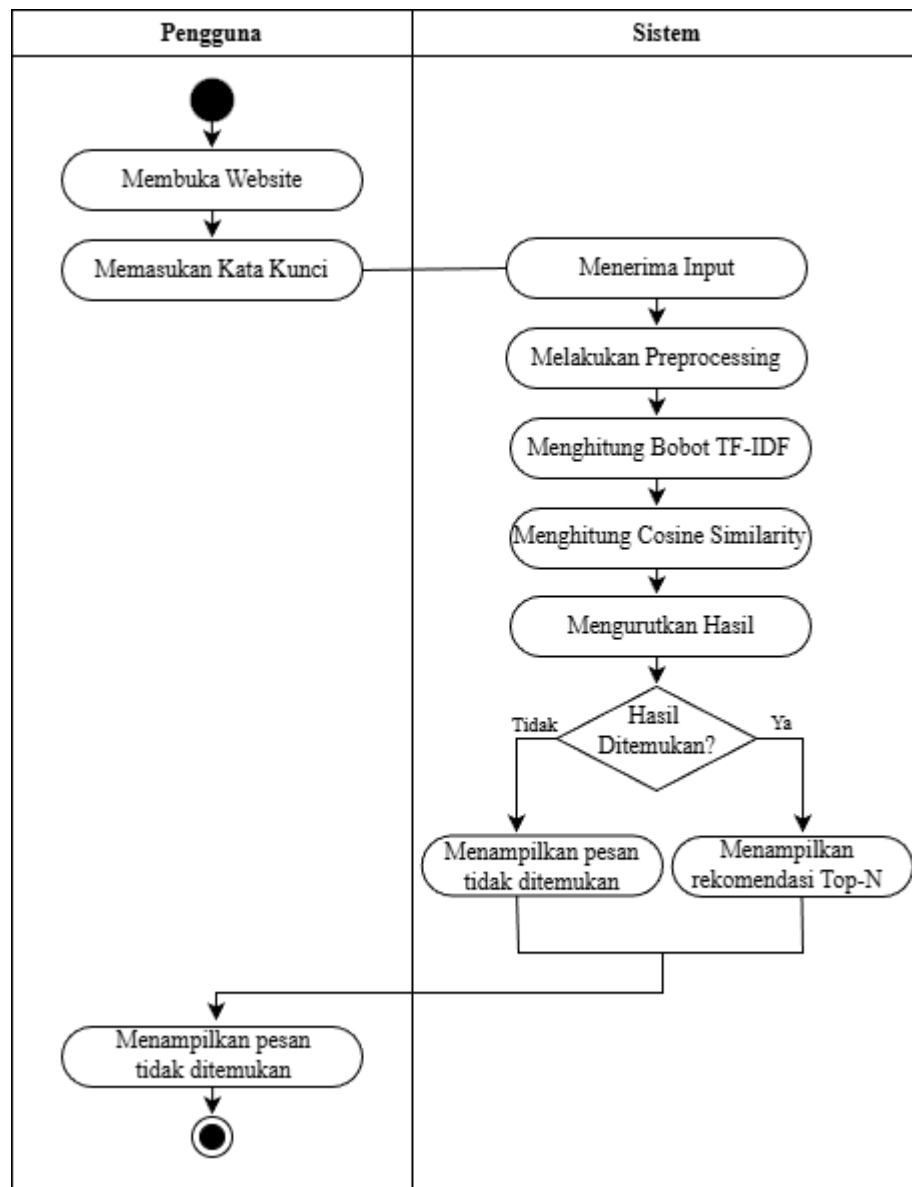
Berdasarkan Gambar 3.4, sistem rekomendasi destinasi wisata memiliki satu aktor utama, yaitu Pengguna (wisatawan). Pengguna dapat melakukan tujuh aktivitas utama dalam sistem.

Tabel 3. 5 Deskripsi *Use Case*

Kode	<i>Use Case</i>	Deskripsi
UC-01	Melihat Beranda	Pengguna dapat mengakses halaman beranda yang menampilkan informasi singkat sistem dan fitur pencarian
UC-02	Melihat Daftar Destinasi	Pengguna dapat menampilkan keseluruhan daftar destinasi wisata yang tersedia
UC-03	Mencari Destinasi	Pengguna dapat mencari destinasi dengan memasukkan kata kunci pencarian
UC-04	Memfilter Destinasi	Pengguna dapat menyaring destinasi berdasarkan kategori dan wilayah
UC-05	Melihat Detail Destinasi	Pengguna dapat melihat informasi lengkap dari destinasi yang dipilih
UC-06	Melihat Rekomendasi Serupa	Sistem menampilkan rekomendasi destinasi serupa ketika pengguna melihat detail destinasi (<i>include</i> dari UC-05)
UC-07	Melihat Halaman Tentang	Pengguna dapat melihat informasi mengenai sistem dan metode yang digunakan

b. *Activity Diagram*

Merupakan alur kegiatan yang terjadi antara pengguna dan sistem ketika tahapan rekomendasi berlangsung. Diagram dibawah menggambarkan kegiatan dari awal hingga tampil hasil rekomendasi destinasi. *Activity Diagram* sistem rekomendasi destinasi wisata dapat dilihat berikut ini:



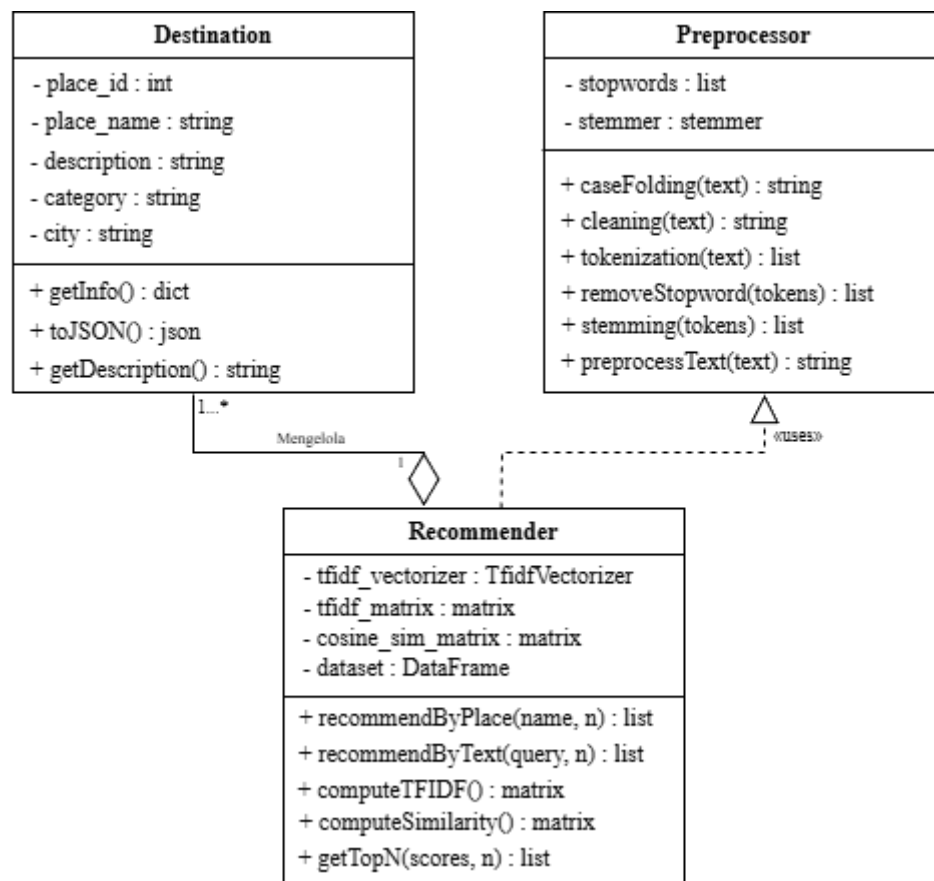
Gambar 3. 5 Activity Diagram Proses Rekomendasi

Berdasarkan Gambar *Activity Diagram* terbagi menjadi dua *swimlane*, yaitu *swimlane* Pengguna dan *swimlane* Sistem. Alur aktivitas dimulai ketika pengguna membuka *website* dan memasukkan kata kunci pencarian. Sistem kemudian menerima *input* tersebut dan melakukan serangkaian proses, dimulai dari *Preprocessing* teks, perhitungan bobot TF-IDF, perhitungan *Cosine Similarity*, hingga pengurutan hasil.

Setelah hasil diurutkan, sistem memeriksa apakah rekomendasi ditemukan. Jika ditemukan, sistem akan menampilkan Top-N destinasi wisata yang relevan, namun jika tidak ada, sistem akan menampilkan pesan bahwa destinasi tidak ditemukan. Hasil yang ditampilkan kemudian dilihat oleh pengguna, dan proses berakhir.

c. *Class Diagram*

Class Diagram yaitu struktur kelas-kelas yang menyusun sistem rekomendasi destinasi wisata disertai atribut, metode, dan relasi antar kelas. Pada sistem ini, terdapat tiga kelas utama, yaitu kelas *Destination* yang merepresentasikan data destinasi wisata, kelas *Preprocessor* yang menangani pengolahan teks, dan kelas *Recommender* yang menjalankan proses rekomendasi.



Gambar 3. 6 Class Diagram

Jadi sistem rekomendasi destinasi wisata terdiri dari tiga kelas utama yang saling berkaitan, yaitu:

1. Kelas *Destination*

Merepresentasikan data destinasi wisata dengan atribut `place_id`, `place_name`, `description`, `category`, dan `city`. Kelas ini memiliki metode untuk mengambil informasi destinasi seperti `getInfo()`, `toJSON()`, dan `getDescription()`.

2. Kelas *Preprocessor*

Menangani proses pengolahan teks. Kelas ini memiliki atribut `stopwords` dan `stemmer`, serta metode `caseFolding()`, `cleaning()`, `tokenization()`, `removeStopword()`, `stemming()`, dan `preprocessText()` yang menggabungkan seluruh tahapan *Preprocessing*.

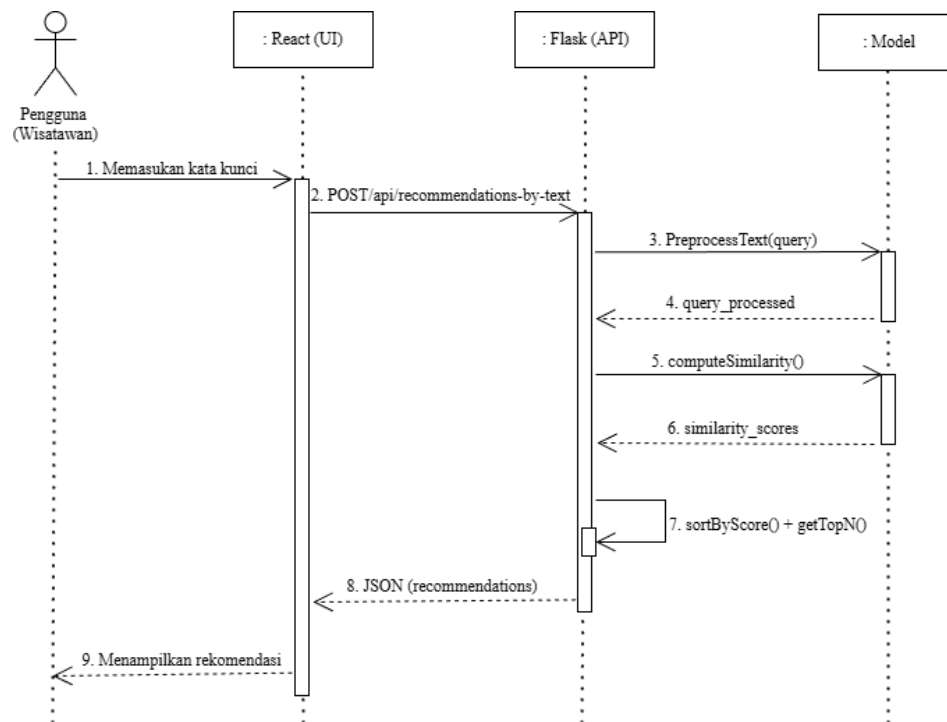
3. Kelas *Recommender*

Merupakan kelas inti yang menjalankan proses rekomendasi. Kelas ini memiliki atribut `tfidf_vectorizer`, `tfidf_matrix`, `cosine_sim_matrix`, dan `dataset`, serta metode `recommendByPlace()`, `recommendByText()`, `computeTFIDF()`, `computeCosineSimilarity()`, dan `getTopN()`.

Relasi antar kelas terdiri dari hubungan agregasi antara kelas *Recommender* dan *Destination*, di mana satu *recommender* mengelola banyak destinasi (1 ke 1..*). Selain itu, kelas *Recommender* memiliki ketergantungan (*dependency*) terhadap kelas *Preprocessor*, karena proses rekomendasi memerlukan tahapan *Preprocessing* sebelum perhitungan dilakukan.

d. *Sequence Diagram*

Digunakan untuk menunjukkan urutan komunikasi antarobjek dalam sistem berdasarkan waktu ketika rekomendasi berlangsung. Diagram ini menunjukkan bagaimana pesan dikirim antara pengguna, antarmuka *Frontend* (React), *backend* (Flask), dan model rekomendasi. *Sequence Diagram* sistem dapat dilihat pada Gambar 3.7 berikut:



Gambar 3. 7 Sequence Diagram Proses Rekomendasi

Gambar di atas menggambarkan interaksi antar empat objek, yaitu Pengguna, React (antarmuka *Frontend*), Flask (*backend* API), dan Model rekomendasi. Urutan interaksi dijelaskan sebagai berikut:

1. Pengguna memasukkan kata kunci pencarian melalui antarmuka React.
2. React mengirimkan permintaan ke *backend* Flask melalui *endpoint* POST /api/recommendations-by-text.
3. Flask memanggil fungsi preprocessText() pada Model untuk memproses *query* pengguna melalui tahapan *Preprocessing*.
4. Model mengembalikan teks yang sudah diproses (*query_processed*) ke Flask.
5. Flask meminta Model untuk menghitung kemiripan (computeSimilarity()) menggunakan TF-IDF dan *Cosine Similarity*.
6. Model mengembalikan nilai *similarity scores* ke Flask.
7. Flask melakukan pengurutan hasil dan pengambilan *Top-N* destinasi (sortByScore() dan getTopN()).

8. Flask mengirimkan hasil rekomendasi dalam format JSON ke React.
9. React menampilkan hasil rekomendasi kepada pengguna melalui antarmuka *website*.

3.4.4 Perancangan Antarmuka

Perancangan antarmuka yaitu untuk mengimplementasikan rancangan tata letak tampilan *website* sebelum tahap implementasi dilakukan. Perancangan antarmuka pada penelitian ini dibuat dalam bentuk *wireframe*, yaitu kerangka dasar yang menggambarkan struktur dan tata letak setiap halaman tanpa memperhatikan detail warna, gambar, maupun tampilan akhir. *Wireframe* digunakan agar struktur antarmuka dapat dirancang terlebih dahulu sebelum diterjemahkan ke dalam tampilan final pada tahap implementasi.

Sistem rekomendasi destinasi wisata Bandung Raya ada empat halaman utama, yaitu Halaman Beranda, Halaman Destinasi, Halaman Detail Destinasi, dan Halaman Tentang. Berikut penjelasan rancangan *wireframe* dari masing-masing halaman.

a. Rancangan Halaman Beranda

Halaman Beranda adalah halaman yang pertama kali dilihat saat pengguna membuka sistem. Dimana terdiri dari beberapa bagian utama, yaitu *header* navigasi yang memuat logo "Parahyangan" dan menu (Beranda, Destinasi, Tentang), bagian *hero* yang berisi judul utama, sub-judul, deskripsi singkat, serta kolom pencarian. Di bawah bagian *hero* terdapat bar statistik yang menampilkan jumlah destinasi, kategori, dan wilayah. Halaman ini juga menampilkan daftar destinasi populer dalam bentuk kartu, daftar kategori destinasi, serta penjelasan singkat mengenai cara kerja sistem rekomendasi. Rancangan *wireframe* Halaman Beranda dapat dilihat pada Gambar 3.8 berikut:

The wireframe illustrates the layout of a homepage titled "Parahyangan". At the top right, there are three navigation buttons: "Beranda", "Destinasi", and "Tentang". The main content area is divided into several sections. The top section features a large "Judul Utama (Headline)" box, followed by "Sub-judul" and "Deskripsi singkat" boxes, and a wide "Kolom Pencarian + Tombol Cari" box. Below this is a statistics bar showing "331 Destinasi", "13 Kategori", and "3 Wilayah". The next section, titled "Judul: Destinasi Populer", contains three cards, each with a placeholder for a photo, a "Nama Destinasi" input field, and a "Wilayah" label. This is followed by a section titled "Judul: Berdasarkan Kategori" with three category cards labeled "Kategori 1", "Kategori 2", and "Kategori 3". Below these is a "Judul: Cara Kerja Sistem" section with three steps: "Langkah 1 Preprocessing", "Langkah 2 TF-IDF", and "Langkah 3 Cosine Similarity". The footer contains the text "Footer (Info Sistem dan Penelitian)".

Gambar 3. 8 Rancangan Halaman Beranda

Pada Gambar 3.8, kolom pencarian ditempatkan di bagian *hero* sehingga pengguna dapat langsung memasukkan kata kunci destinasi yang diinginkan tanpa harus berpindah halaman. Bar statistik memberikan gambaran ringkas mengenai cakupan sistem, sedangkan bagian destinasi

populer dan kategori berfungsi sebagai akses cepat bagi pengguna untuk menjelajahi destinasi tanpa perlu melakukan pencarian secara manual.

b. Rancangan Halaman Destinasi

Halaman Destinasi dirancang untuk menampilkan seluruh daftar destinasi wisata yang tersedia pada sistem. Halaman ini memuat *header* navigasi yang sama dengan Halaman Beranda, bagian judul halaman, kolom pencarian, serta fitur filter berdasarkan wilayah dan kategori. Daftar destinasi ditampilkan dalam bentuk *grid* kartu, di mana setiap kartu memuat foto, nama destinasi, dan wilayah. Rancangan *wireframe* Halaman Destinasi.

Parahyangan [Beranda](#) [Destinasi](#) [Tentang](#)

Judul: Semua Destinasi Bandung Raya

Deskripsi singkat

Kolom Cari Destinasi [Filter Wilayah](#) [Filter Kategori](#)

Info: 331 Destinasi ditemukan

 Foto Nama Destinasi Wilayah	 Foto Nama Destinasi Wilayah	 Foto Nama Destinasi Wilayah
 Foto Nama Destinasi Wilayah	 Foto Nama Destinasi Wilayah	 Foto Nama Destinasi Wilayah

Grid Kartu Destinasi (lanjut hingga 331 destinasi)

Footer (Info Sistem dan Penelitian)

Gambar 3. 9 Rancangan Halaman Destinasi

Pada tampilan dirancang dalam bentuk *grid* tiga kolom agar pengguna dapat melihat banyak destinasi sekaligus dengan tata letak yang terstruktur. Fitur filter wilayah dan kategori memudahkan pengguna untuk mempersempit pilihan destinasi sesuai preferensinya. Informasi jumlah destinasi yang ditemukan ditampilkan di bawah filter sebagai konfirmasi hasil pencarian.

c. Rancangan Halaman Detail Destinasi

Halaman Detail Destinasi dirancang untuk menampilkan informasi dari destinasi yang dipilih. Halaman ini memuat tombol kembali, label kategori, wilayah, nama destinasi sebagai judul utama, serta kutipan singkat sebagai pengantar deskripsi. Konten utama terbagi menjadi dua kolom, yaitu kolom kiri yang berisi daftar aktivitas yang dapat dilakukan dan tips berkunjung, serta kolom kanan (*sidebar*) yang berisi informasi kunjungan, tombol menuju Google Maps, dan opsi bagikan. Di bagian bawah halaman ditampilkan rekomendasi destinasi serupa yang didapatkan sistem. Rancangan *wireframe* Halaman Detail Destinasi dapat dilihat berikut:

Parahyangan
Beranda
Destinasi
Tentang

Tombol Kembali

Label Kategori

Wilayah

Nama Destinasi (Judul Besar)

Kutipan / lead deskripsi (italic)

Judul: Aktivitas yang bisa dilakukan

☐
Aktivitas 1 (judul)

☐
Aktivitas 2 (judul)

☐
Aktivitas 3 (judul)

Judul: Tips Berkunjungan

01 Tips 1 (judul + deskripsi)

02 Tips 2 (judul + deskripsi)

03 Tips 3 (judul + deskripsi)

Judul: Destinasi Serupa (Rekomendasi CBF)

Rekomendasi 1

Rekomendasi 2

Rekomendasi 3

Rekomendasi 4

Info Kunjungan

Durasi

Waktu terbaik

Pemarahalan

Tombol Google Maps

Bagian

☐
☐
☐

Gambar 3. 10 Rancangan Halaman Detail Destinasi

Tata letak dua kolom dipilih agar informasi utama destinasi dapat dibaca secara mengalir di kolom kiri, sementara informasi pendukung seperti durasi kunjungan dan akses peta tetap mudah dijangkau di *sidebar*. Bagian rekomendasi destinasi serupa diletakkan di bagian bawah halaman, sebagai penerapan langsung dari metode *Content-Based Filtering* pada

antarmuka, sehingga pengguna dapat menemukan alternatif destinasi lain yang memiliki kemiripan dengan destinasi yang sedang dilihat.

d. Rancangan Halaman Tentang

Halaman Tentang dirancang untuk menampilkan informasi mengenai sistem rekomendasi yang dibangun. Halaman ini memuat judul halaman, deskripsi singkat sistem, serta penjelasan tahapan metode yang digunakan, yaitu *Text Preprocessing*, *TF-IDF Vectorization*, dan *Cosine Similarity*. Di bagian bawah halaman ditampilkan tiga kotak informasi tambahan, yaitu informasi mengenai *dataset*, teknologi yang digunakan, dan penelitian.

Parahyangan Beranda Destinasi Tentang

Judul: Tentang Sistem

Deskripsi sistem

Section: Tahapan Metode

01 Text Preprocessing

02 TF-IDF Vectorization

03 Cosine Similarity

Dataset

Teknologi

Penelitian

Footer (Info Sistem dan Penelitian)

Gambar 3. 11 Rancangan Halaman Tentang

Pada Gambar 3.11, penjelasan tahapan metode ditampilkan secara berurutan dari 01 hingga 03 agar pengguna dapat memahami alur kerja sistem rekomendasi secara sistematis. Halaman ini juga berfungsi sebagai bentuk transparansi mengenai metode yang digunakan dalam sistem, sehingga pengguna mengetahui dasar logika yang digunakan oleh sistem dalam memberikan rekomendasi destinasi.