

Penerapan *Natural Language Processing* (NLP) dengan Metode *Cosine Similarity* pada Sistem E-Monev untuk Pencarian Program Pembangunan Daerah

Application of Natural Language Processing (NLP) with the Cosine Similarity Method in an E-Monev System for Searching Regional Development Programs

Moch Faizal Ansyori, Ahmad Heru Mujiyanto

Fakultas Teknologi Informasi, Universitas Hasyim Asy'ari, Jombang, Indonesia

mochfaizal@mhsunhasy.ac.id, ahmadmujiyanto@unhasy.ac.id

Notes

Submitted: 09-05-2025

Revised: 15-06-2025

Accepted: 17-06-2025

Published: 30-06-2025

How to cite: M. F. Ansyori and A. H. Mujiyanto, "Penerapan Natural Language Processing (NLP) dengan Metode Cosine Similarity pada Sistem E-Monev untuk Pencarian Program Pembangunan Daerah," *Journal Software, Hardware and Information Technology (SHiFT)*, vol. 5, no. 2, pp. 85–105, Jun. 2025, doi: 10.24252/shift.v5i2.183.

DOI: 10.24252/shift.v5i2.183

Copyright 2025© the Author(s)

This work is licensed under a [Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License](#).



ABSTRACT

The Evaluation and Monitoring System (E-Monev) is a digital tool for periodically monitoring performance achievement and budget absorption in Regional Government Organizations (OPD). A key challenge in its implementation is accurately identifying relevant regional development programs from user text input. Conventional keyword-based search cannot fully capture the semantic meaning of text and often returns inaccurate or contextually irrelevant results. This study designs and develops a semantic search feature for E-Monev using a Natural Language Processing (NLP) approach that combines BERT (Bidirectional Encoder Representations from Transformers) text representations with the Cosine Similarity method. The data were drawn from the Work Plan (Renja) documents of all OPDs in Bojonegoro Regency, prepared in accordance with Minister of Home Affairs Decree Number 900.1.15.5-3406 of 2024, the second amendment to Decree Number 050-5889 of 2021 on the verification, validation, and inventory of updates to the classification, codification, and nomenclature of regional development and financial planning. Both user queries and Renja data were converted into embedding vectors, and their semantic similarity was then calculated. Test results show that the system significantly improves search accuracy, achieving a precision of 0.884. This approach is expected to make the retrieval of regional development program information more efficient and relevant, thereby supporting more precise and contextual performance evaluation by the local government.

Keywords: BERT embedding, cosine similarity, E-Monev, information retrieval, local government.

ABSTRAK

Sistem Evaluasi dan Monitoring (E-Monev) merupakan perangkat digital untuk memantau capaian kinerja dan

penyerapan anggaran Organisasi Perangkat Daerah (OPD) secara berkala. Salah satu tantangan utama dalam penerapannya adalah mengidentifikasi program pembangunan daerah yang relevan secara akurat berdasarkan masukan teks pengguna. Pencarian konvensional berbasis kata kunci belum mampu menangkap makna semantik teks secara utuh sehingga kerap menghasilkan temuan yang tidak akurat atau tidak relevan secara kontekstual. Penelitian ini merancang dan mengembangkan fitur pencarian semantik pada E-Monev menggunakan pendekatan Natural Language Processing (NLP) yang memadukan representasi teks berbasis BERT (Bidirectional Encoder Representations from Transformers) dengan metode Cosine Similarity. Data penelitian bersumber dari dokumen Rencana Kerja (Renja) seluruh OPD di Kabupaten Bojonegoro yang disusun berdasarkan Keputusan Menteri Dalam Negeri Nomor 900.1.15.5-3406 Tahun 2024 tentang Perubahan Kedua atas Keputusan Menteri Dalam Negeri Nomor 050-5889 Tahun 2021 tentang Hasil Verifikasi, Validasi, dan Inventarisasi Pemutakhiran Klasifikasi, Kodefikasi, dan Nomenklatur Perencanaan Pembangunan dan Keuangan Daerah. Teks masukan pengguna dan data Renja dikonversi menjadi vektor embedding, kemudian dihitung tingkat kemiripan semantiknya. Hasil pengujian menunjukkan bahwa sistem yang dikembangkan meningkatkan akurasi pencarian secara signifikan dengan nilai precision sebesar 0,884. Pendekatan ini diharapkan dapat meningkatkan efisiensi dan relevansi penelusuran informasi program pembangunan daerah sehingga mendukung proses evaluasi kinerja pemerintah daerah secara lebih tepat dan kontekstual.

Kata kunci: BERT embedding, cosine similarity, E-Monev, pemerintahan daerah, pencarian informasi.

1. PENDAHULUAN

Transformasi digital telah mendorong peningkatan tuntutan terhadap transparansi dan akuntabilitas dalam pelaksanaan program pembangunan serta pengelolaan anggaran. Di Kabupaten Bojonegoro, proses monitoring dan evaluasi (monev) kinerja dan serapan anggaran masih banyak dilakukan secara manual atau semi-digital menggunakan lembar kerja daring seperti *Google Spreadsheet*. Praktik ini tidak hanya menyulitkan dalam menjaga konsistensi data, tetapi juga menyulitkan pencarian program pembangunan yang akurat dan relevan, terutama karena sistem pencarian program pembangunan yang digunakan masih berbasis pencocokan teks secara eksak (*exact match*) [1].

Kepmendagri No. 900.1.15.5-3406 Tahun 2024 [2] mengatur struktur data monev dalam bentuk hierarki yang kompleks, mulai dari urusan, bidang urusan, tujuan, sasaran, program, kegiatan, hingga sub-kegiatan, dengan indikator kinerja di setiap tingkatan. Kompleksitas ini menuntut adanya sistem yang mampu mengelola, menghitung, dan menampilkan data secara

sistematis dan otomatis. Oleh karena itu, dibutuhkan pengembangan sistem E-Monev (*Electronic Monitoring and Evaluation*) yang tidak hanya mendukung otomatisasi perhitungan kinerja dan anggaran, tetapi juga mampu menyediakan fitur pencarian program pembangunan daerah berbasis semantik [3].

Natural Language Processing (NLP) merupakan salah satu pendekatan yang relevan dalam memahami makna secara semantik [4] dalam pencarian program pembangunan daerah. Salah satu metode dalam NLP yang umum digunakan untuk mengukur kemiripan semantik antar teks adalah *cosine similarity*. Berdasarkan penelitian [5] *cosine similarity* menunjukkan performa yang lebih tinggi dalam mengukur kesamaan antar dokumen dibandingkan metode *jaccard similarity*. Selain itu, [6] menyatakan bahwa *cosine similarity* efektif dalam merepresentasikan hubungan antar teks dalam ruang vektor multidimensi.

Sejauh penelusuran penulis, belum ditemukan sistem monitoring dan evaluasi pemerintahan daerah yang menerapkan metode *cosine similarity* dalam proses pencarian program pembangunan daerah secara semantik. Sebagian besar sistem monev yang ada umumnya hanya memfokuskan pada pengumpulan dan visualisasi data serapan anggaran atau capaian kinerja, tanpa memfasilitasi pencarian informasi yang kontekstual.

Fokus utama dari penelitian ini adalah menerapkan *cosine similarity* dalam sistem e-monev untuk meningkatkan relevansi, fleksibilitas, dan efisiensi proses pencarian program pembangunan secara semantik, tanpa bergantung pada pencocokan literal, dan sistem yang dibangun dapat membantu pengguna menemukan program pembangunan yang sesuai secara lebih cepat dan kontekstual, tanpa harus mencocokkan kata kunci secara persis. Implementasi ini bertujuan untuk mendukung perbaikan tata kelola pembangunan daerah melalui proses monev yang lebih efisien, transparan, dan berbasis data [7].

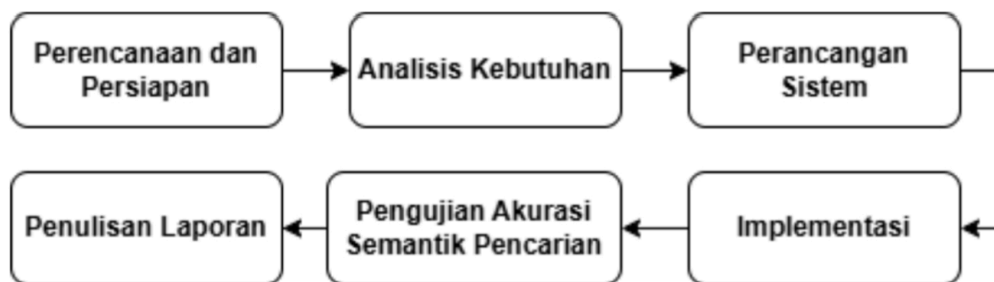
2. METODE PENELITIAN

Cosine similarity adalah metode untuk mengukur tingkat kemiripan antara dua vektor berdasarkan sudut di antara keduanya. Nilainya berkisar antara -1 hingga 1, di mana 1 berarti kedua vektor searah, 0 berarti tegak lurus (tidak berkorelasi), dan -1 berarti berlawanan arah [8]. Dalam pengolahan teks, metode ini digunakan untuk mengukur kesamaan antara *query* pencarian dengan dokumen berdasarkan representasi vektor dari teks tersebut [9]. Adapun rumus *cosine similarity* dapat dilihat pada persamaan 1 berikut.

$$\text{cosine_similarity} = \frac{A \cdot B}{\|A\| \|B\|} \quad (1)$$

2.1. Prosedur Penelitian

Prosedur penelitian ini terdiri dari enam tahapan utama yang saling berkaitan, yaitu Perencanaan dan Persiapan, Analisis Kebutuhan, Perancangan Sistem, Implementasi, Pengujian Akurasi Semantik Pencarian, dan Penulisan Laporan. Pada tahap awal dilakukan identifikasi masalah, studi pustaka, dan pengumpulan data. Kemudian ditentukan kebutuhan sistem baik fungsional maupun non-fungsional, dilanjutkan dengan perancangan sistem yang terdiri dari perancangan database, dan antarmuka pengguna, serta metode pencarian semantik. Setelah sistem dikembangkan, dilakukan pengujian akurasi dengan *precision*, lalu ditutup dengan penyusunan laporan akhir. Alur lengkap tahapan tersebut ditampilkan pada Gambar 1.



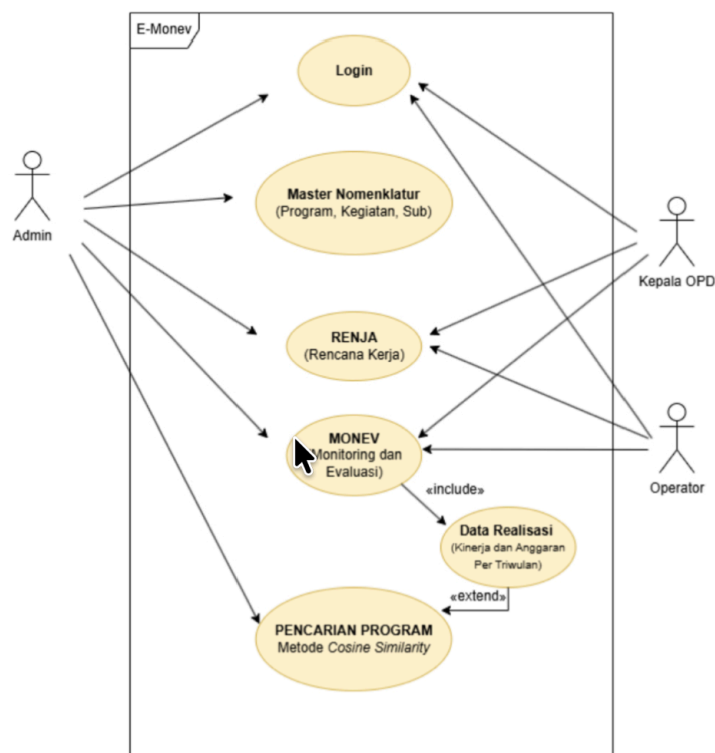
Gambar 1. Prosedur Penelitian

2.2. Perancangan Sistem

Dalam analisis ini penulis menggunakan UML (*Unified Modeling Language*) untuk lebih memudahkan dalam perancangan dari sistem informasi yang akan dibangun, dikarenakan UML termasuk metode untuk merancang program yang berorientasi pada objek [10] dan sesuai dengan sistem informasi yang akan dibangun.

Use Case Diagram

Fungsi dari *use case diagram* memberikan gambaran kegiatan yang dilakukan oleh user di dalam sistem yang akan dibangun [11]. Adapun *use case diagram* e-monev ditampilkan pada Gambar 2.

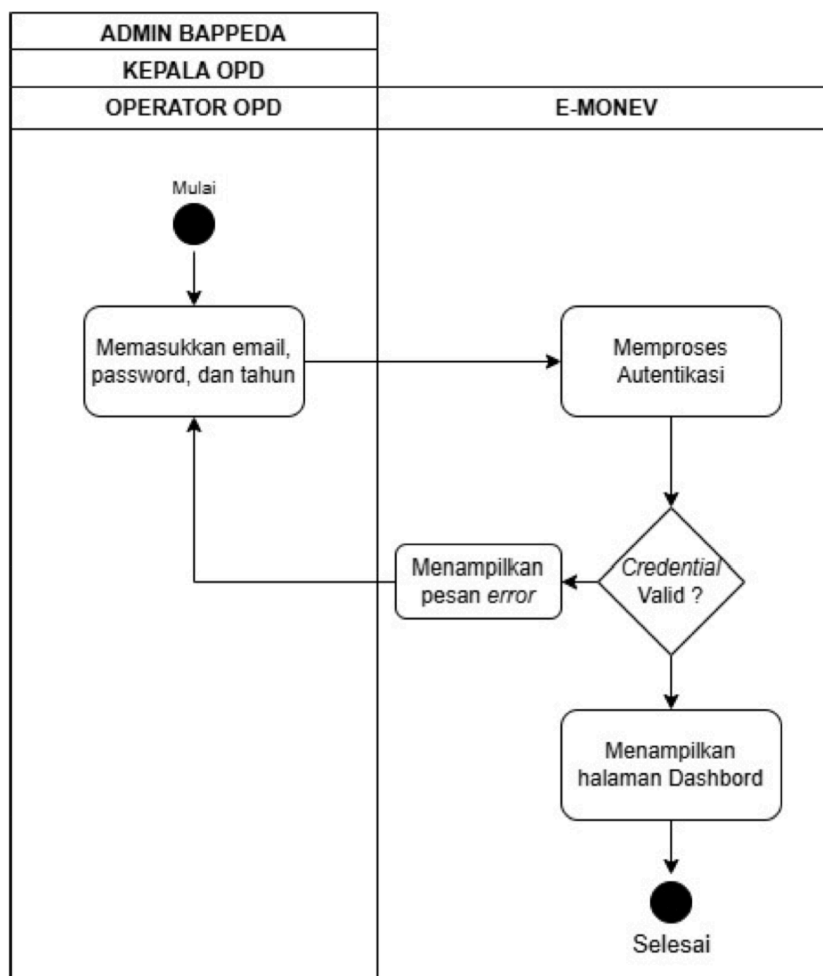


Gambar 2. Use Case Diagram E-Monev

Gambar 2 memperlihatkan *use case diagram* system e-monev. Diagram tersebut menunjukkan bahwa admin memiliki akses penuh terhadap seluruh fitur sistem e-monev, Kepala OPD memiliki hak akses untuk dapat melihat data Renja serta hasil monitoring dan evaluasi, sedangkan Operator berfokus pada input data realisasi per triwulan.

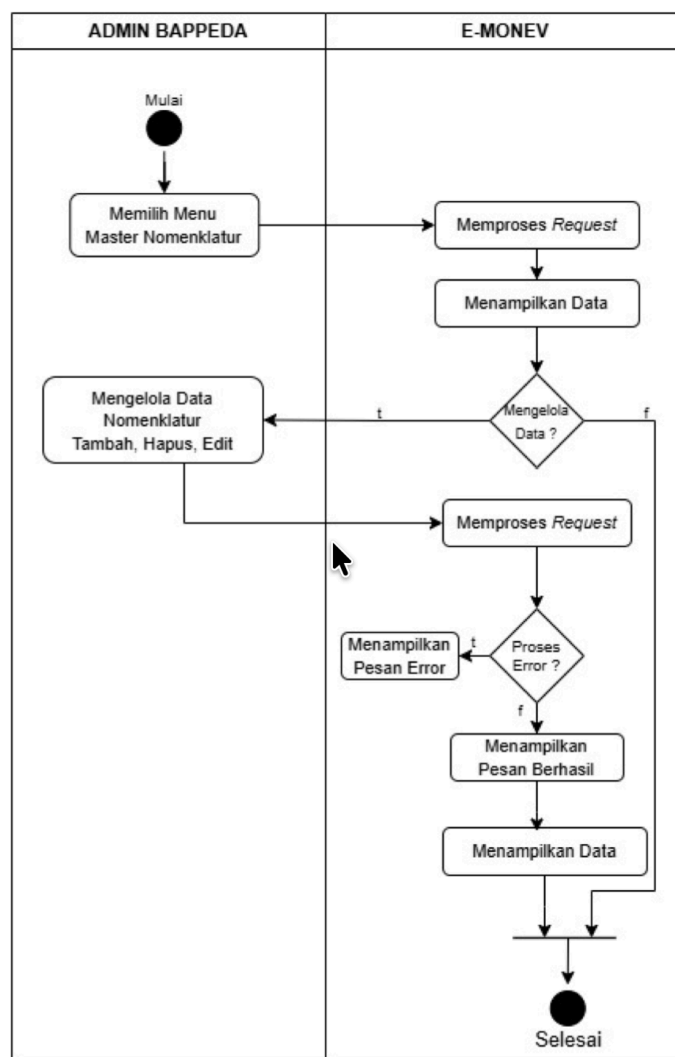
Activity Diagram

Activity diagram menggambarkan alur aktivitas atau proses bisnis yang berlangsung dalam sistem e-monev, mulai dari interaksi pengguna hingga logika sistem. Untuk dapat mengakses sistem e-monev, pengguna memerlukan kredensial berupa *email*, *password*, dan *ta-hun*. Proses login dalam bentuk *activity diagram* ditampilkan pada Gambar 3.



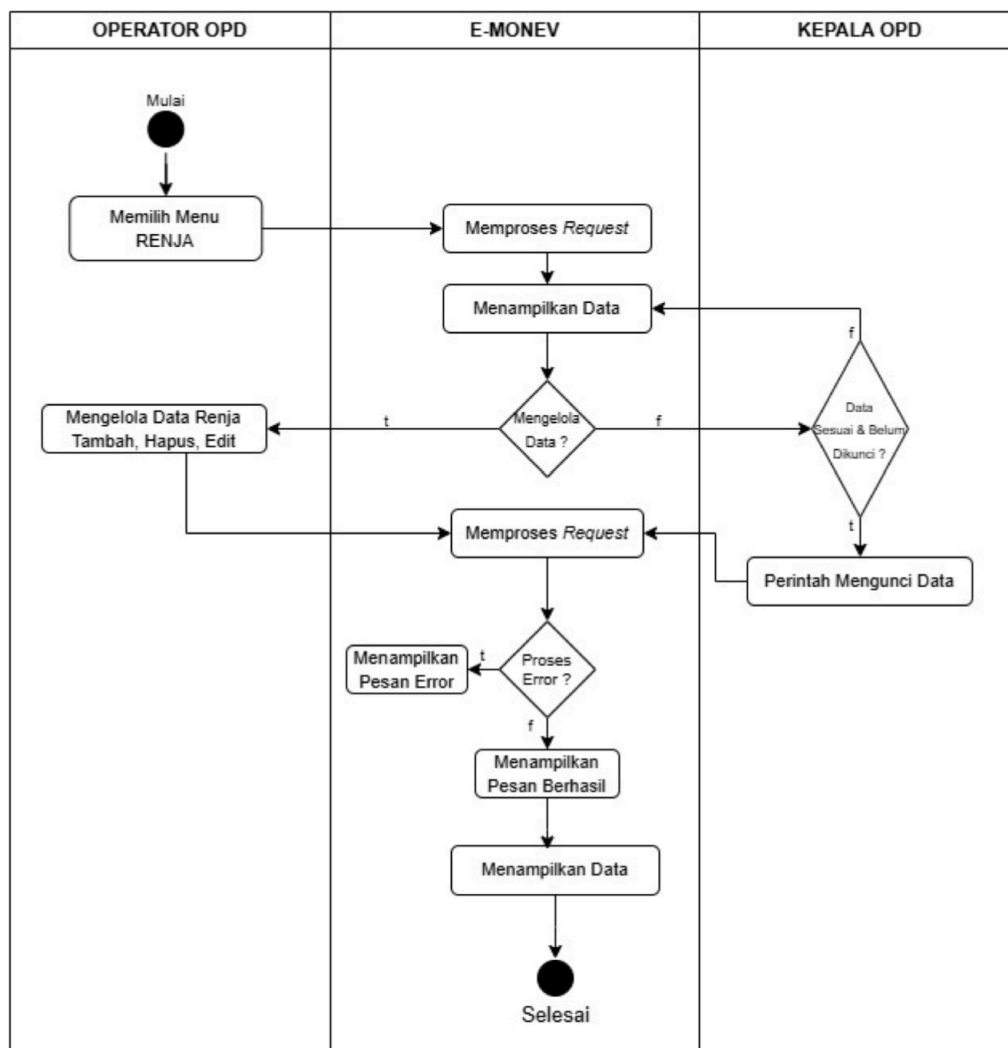
Gambar 3. Activity Diagram Login

Data master yang wajib tersedia dalam sistem e-monev adalah nomenklatur pembangunan daerah yang memuat data berdasarkan Keputusan Menteri Dalam Negeri (Kepmendagri) [2]. Fitur master nomenklatur hanya dapat dikelola oleh Admin yang memiliki kewenangan untuk menambah, mengubah, dan menghapus data sesuai kebutuhan. Setiap aksi akan direspons oleh sistem; apabila terjadi kesalahan, sistem akan menampilkan pesan *error*, sedangkan apabila berhasil, sistem akan memberikan notifikasi bahwa data telah berhasil diproses dan disimpan. Activity diagram proses pengelolaan nomenklatur ditampilkan pada Gambar 4.



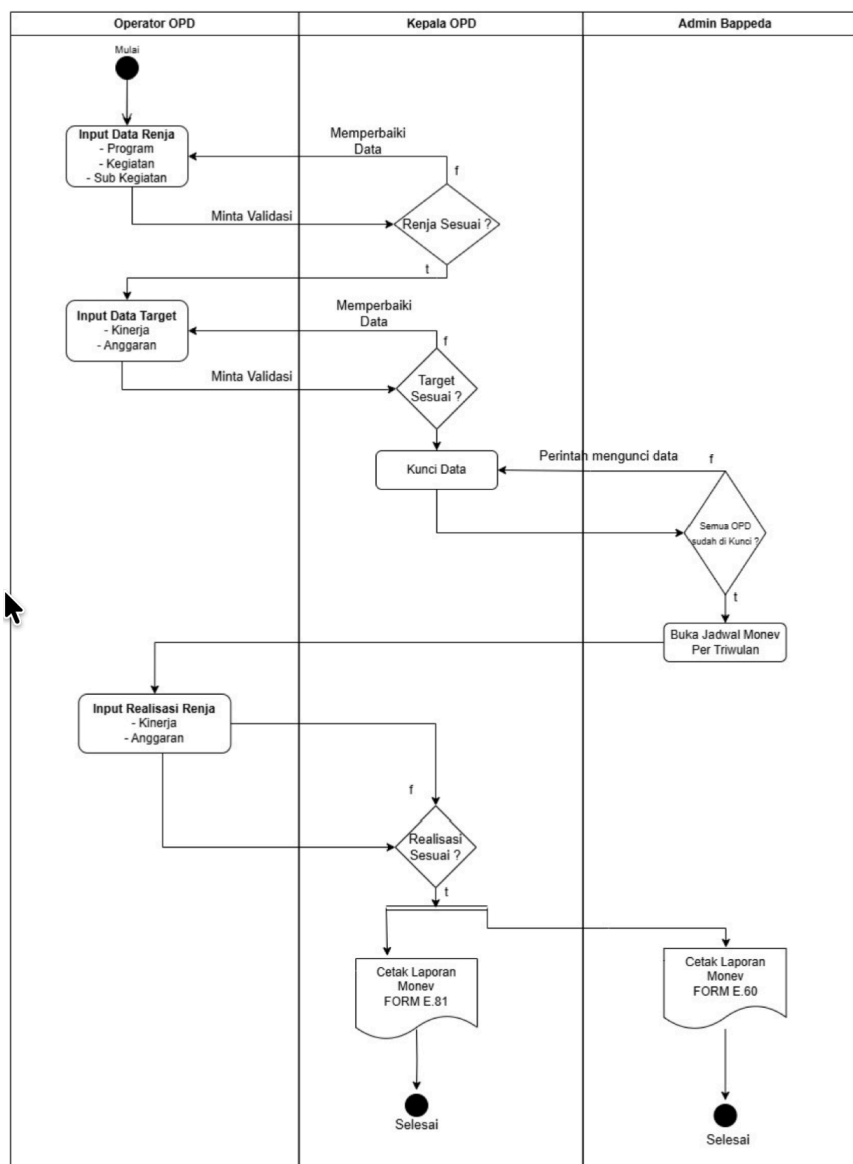
Gambar 4. Activity Diagram Master Nomenklatur Pembangunan

Setelah nomenklatur pembangunan daerah dibuat dan dikelola, proses selanjutnya adalah penyusunan Rencana Kerja (Renja). Proses ini dimulai ketika Operator memilih menu Renja pada sistem, kemudian sistem akan memproses permintaan dan menampilkan data yang relevan. Apabila pengguna melakukan pengelolaan data, seperti penambahan, perubahan, atau penghapusan, sistem akan memproses kembali tindakan tersebut dan memberikan respons. Jika terjadi kesalahan, sistem akan menampilkan pesan *error*; sedangkan jika berhasil, sistem akan menampilkan notifikasi sukses disertai dengan data yang telah diperbarui. Activity diagram proses penyusunan Renja ditampilkan pada Gambar 5.



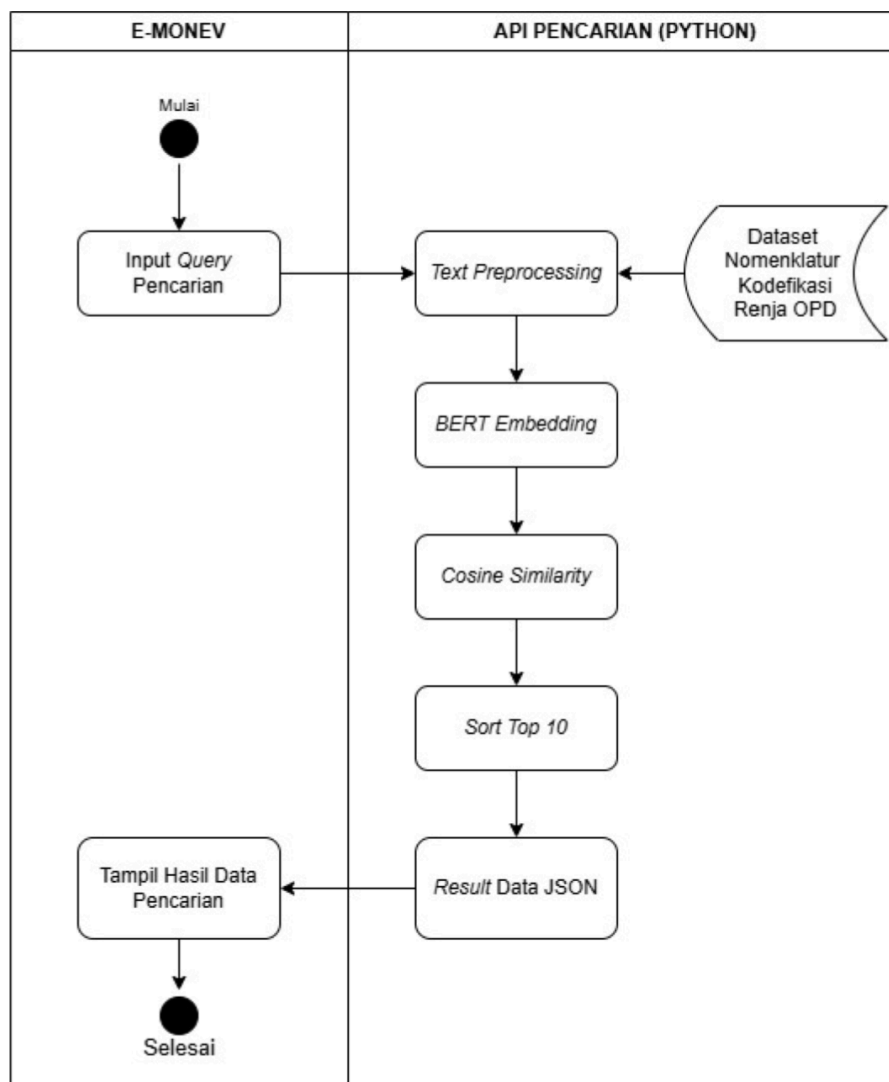
Gambar 5. Activity Diagram Renja

Setelah penyusunan Renja selesai dan sesuai, langkah selanjutnya adalah masuk ke menu Monitoring dan Evaluasi (Monev). Pada tahap ini, dipastikan bahwa data Renja meliputi program, kegiatan, dan sub kegiatan beserta target kinerja masing-masing telah diinputkan secara lengkap. Operator berperan aktif dalam pengisian data Renja, target kinerja dan anggaran, serta realisasi kinerja ketika jadwal monev dibuka oleh Admin. Sementara itu, Kepala OPD memiliki akses untuk memantau capaian realisasi di unit kerjanya. Proses ini diakhiri dengan pencetakan dokumen E-60 dan E-81 yang digunakan sebagai bahan evaluasi untuk dikirimkan ke tingkat Provinsi. Activity diagram proses monitoring dan evaluasi ditampilkan pada Gambar 6.



Gambar 6. Activity Diagram Monitoring dan Evaluasi

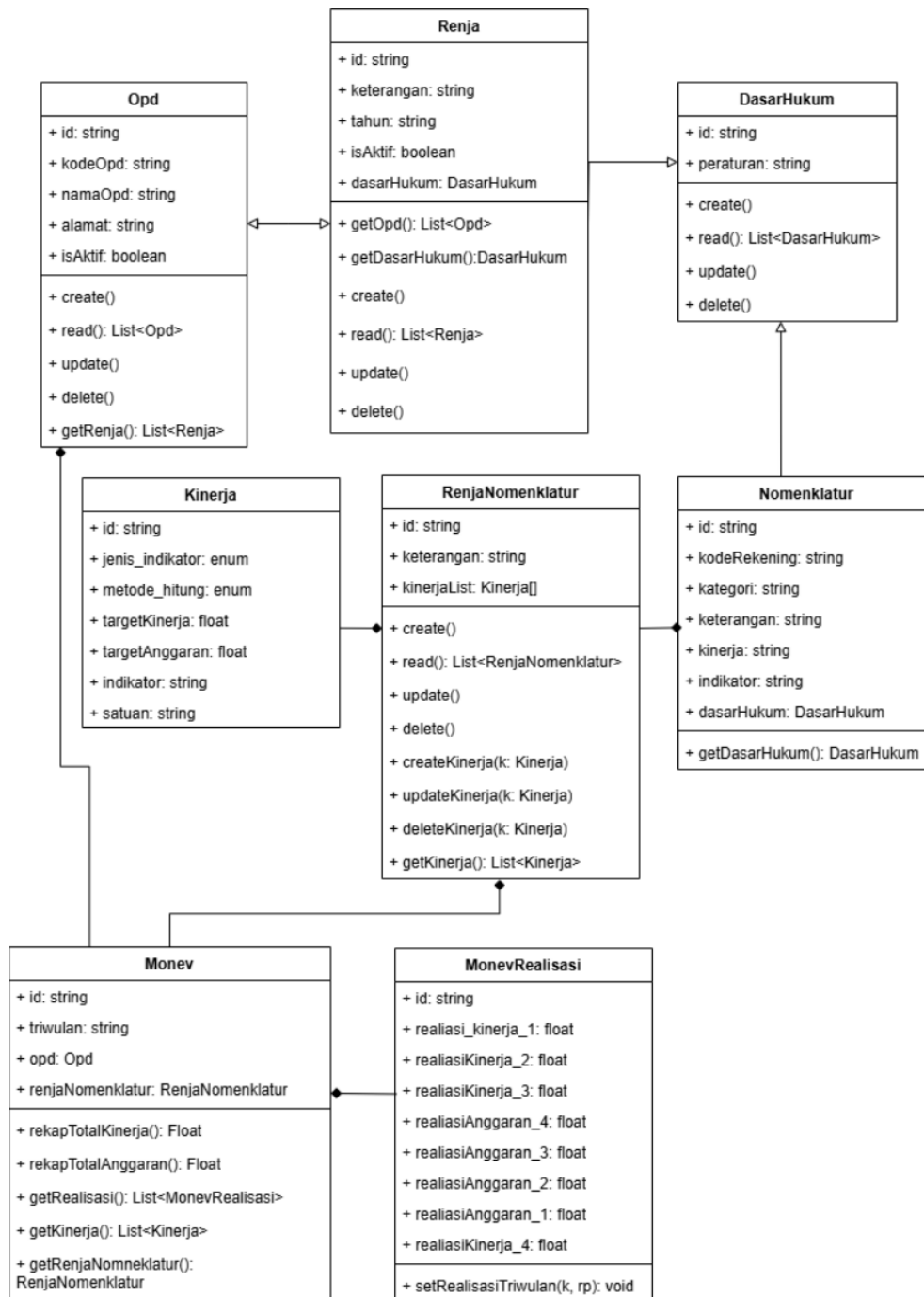
Setelah seluruh proses perencanaan renja, input realisasi kinerja dan anggaran dilalui, fitur pencarian semantik dapat diterapkan untuk mendukung proses monitoring dan evaluasi yang lebih informatif dan kontekstual. Proses diawali saat pengguna memasukkan kata kunci pencarian, yang kemudian melalui tahapan *text preprocessing* [12]. Selanjutnya, sistem membentuk representasi vektor menggunakan model BERT (*Bidirectional Encoder Representations from Transformers*), yang dipilih karena kemampuannya dalam memahami konteks kata secara dua arah [13]. Dalam implementasinya, digunakan pustaka IndoBERT [3] sebagai model bahasa berbasis Bahasa Indonesia. Setelah kata kunci dan *dataset* dikonversi menjadi vektor, seluruh representasi disimpan dalam bentuk *embedding matrix* untuk keperluan pencocokan semantik menggunakan metode *cosine similarity*. Setelah proses perhitungan skor selesai, sistem akan mengurutkan hasil berdasarkan skor tertinggi dan menyajikannya dalam format *JSON*. Activity diagram proses pencarian data pembangunan daerah secara semantik ini ditampilkan pada gambar 7.



Gambar 7. Activity Diagram Pencarian Program Pembangunan Daerah

Class Diagram

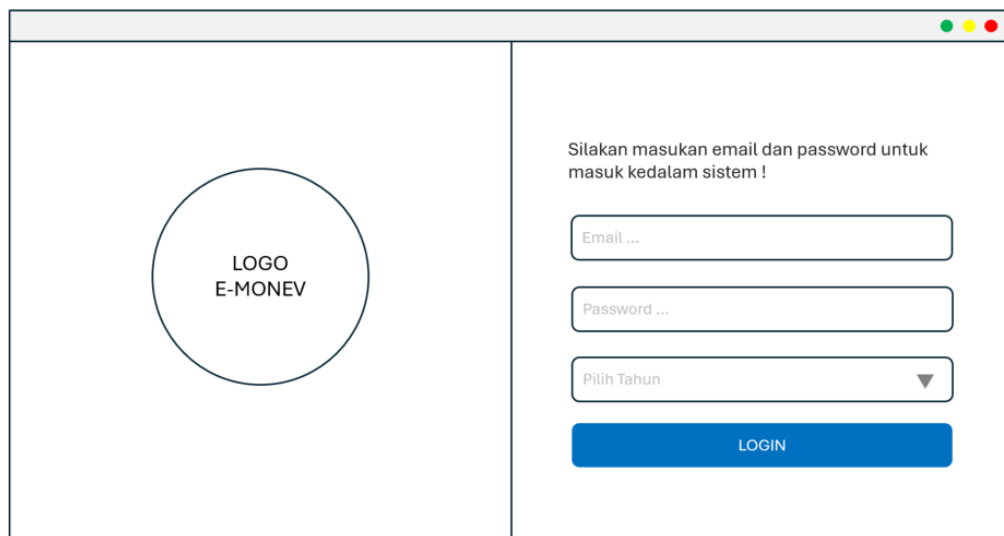
Class diagram adalah diagram struktur sistem yang menunjukkan kelas beserta atribut dan relasi antar kelas [14]. Sistem e-monev terdiri dari beberapa kelas utama yang saling terhubung. Kelas Opd berfungsi mengelola data Organisasi Perangkat Daerah dan memiliki relasi *One-to-Many* dengan Renja, yang merepresentasikan rencana kerja tahunan serta terhubung ke Opd dan DasarHukum melalui relasi *Many-to-One*. Kelas DasarHukum menyimpan regulasi perencanaan dan memiliki relasi *One-to-Many* dengan Renja dan Nomenklatur. Nomenklatur menyimpan kode dan uraian nomenklatur berdasarkan DasarHukum *Many-to-One*. Relasi antara Renja dan indikator kinerja dikelola oleh kelas RenjaNomenklatur, yang memiliki relasi *One-to-Many* dengan Kinerja. Kelas Kinerja menyimpan indikator dan target yang akan dievaluasi, dengan relasi *Many-to-One* ke RenjaNomenklatur. Evaluasi dilakukan melalui kelas Monev, yang merekam data berdasarkan Opd dan RenjaNomenklatur, serta terhubung ke MonevRealisasi *One-to-Many*. Kelas terakhir, MonevRealisasi, menyimpan data realisasi kinerja dan anggaran triwulanan dengan relasi *Many-to-One* ke Monev. Tampilan *class diagram* pada sistem e-monev ditunjukkan pada gambar 8.



Gambar 8. Class Diagram E-Monev

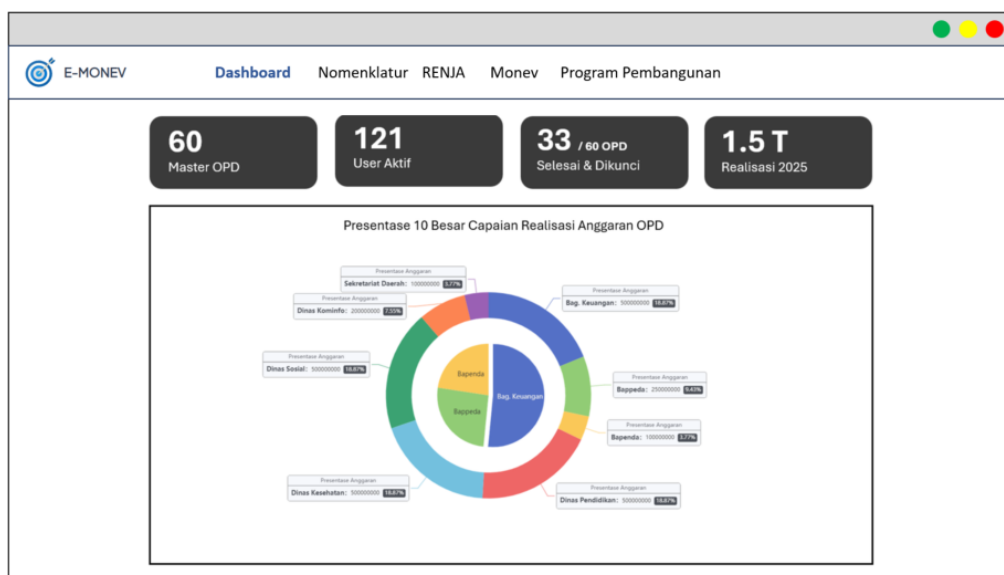
Desain User Interface

User Interface (UI) Design adalah seni dan ilmu menciptakan antarmuka digital yang menarik, intuitif, dan mudah digunakan. UI Design mencakup berbagai elemen visual yang berinteraksi dengan user, seperti tombol, ikon, tipografi, palet warna, dan tata letak [15]. Pada desain UI halaman login pengguna diminta untuk melakukan autentikasi dengan mengisi *email*, *password*, dan tahun evaluasi agar bisa masuk kedalam sistem. Halaman login ditampilkan pada Gambar 9.



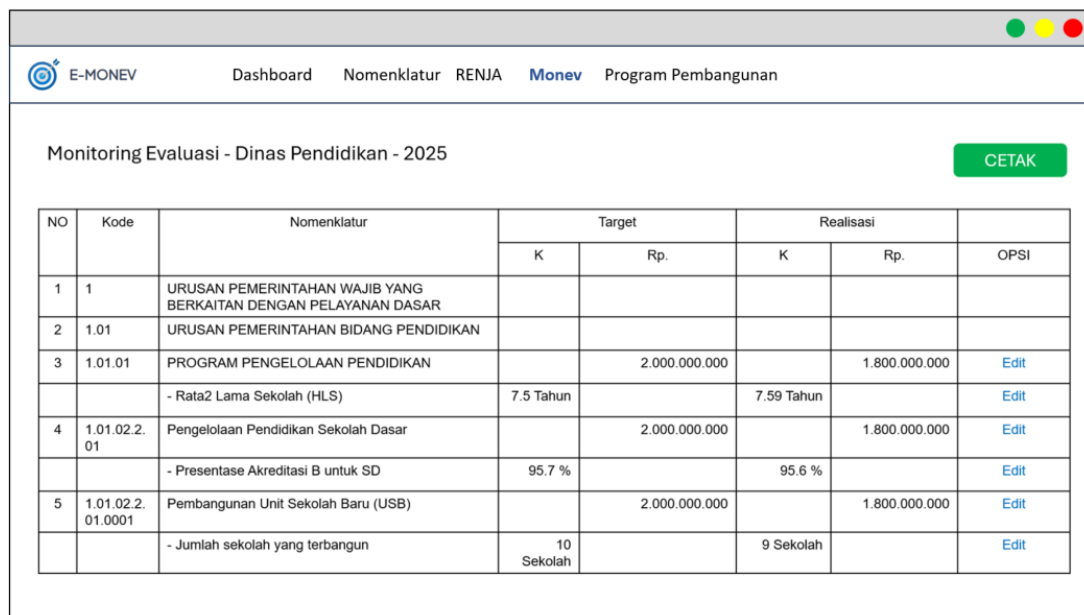
Gambar 9. Desain User Interface Halaman Login

Pada desain UI halaman *dashboard* ditampilkan jumlah realisasi keseluruhan dari total OPD di Kabupaten Bojonegoro serta terdapat grafik Presentase 10 Besar Capaian Realisasi Anggaran OPD, serta terdapat beberapa informasi lain terkait user yang aktif menggunakan e-monev. Halaman *dashboard* ditampilkan pada Gambar 10.



Gambar 10. Desain User Interface Halaman Dashboard

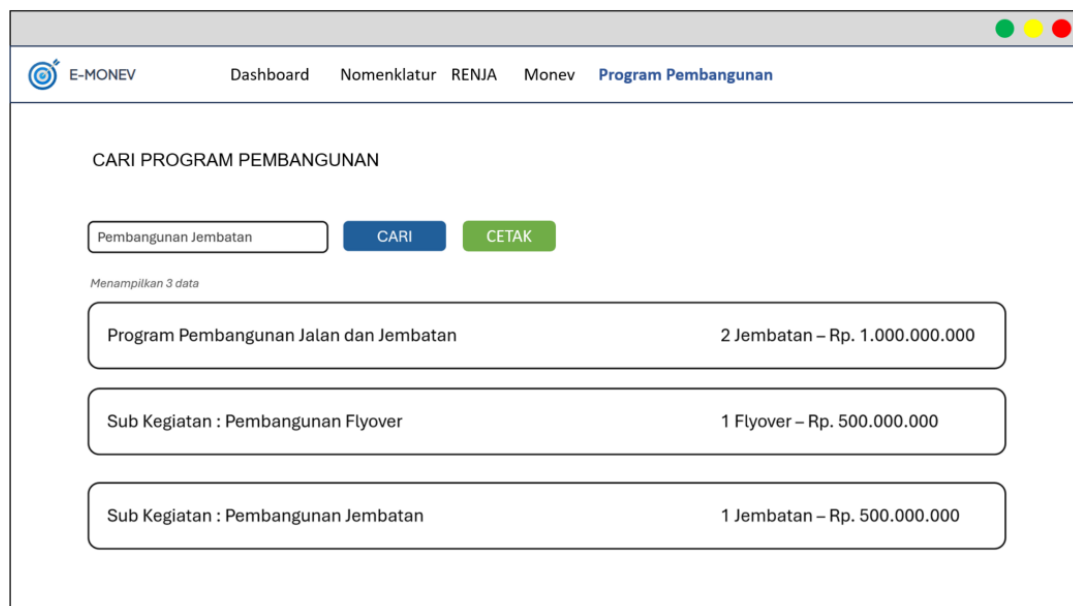
Pada desain UI halaman monitoring dan evaluasi, terdapat tabel yang berisi data RENJA dari masing-masing OPD, dengan kolom No, Kode, Nomenklatur, Target (Kinerja dan Anggaran), serta Realisasi (Kinerja dan Anggaran). Operator bertugas untuk melengkapi data target dan realisasi dari setiap nomenklatur yang tersedia. Sementara itu, Kepala OPD dan Admin dapat secara langsung memantau realisasi program, kegiatan, dan subkegiatan per triwulan dalam tahun berjalan. Tampilan halaman monitoring dan evaluasi dapat dilihat pada gambar 11.



NO	Kode	Nomenklatur	Target		Realisasi		OPSI
			K	Rp.	K	Rp.	
1	1	URUSAN PEMERINTAHAN WAJIB YANG BERKAITAN DENGAN PELAYANAN DASAR					
2	1.01	URUSAN PEMERINTAHAN BIDANG PENDIDIKAN					
3	1.01.01	PROGRAM PENGELOLAAN PENDIDIKAN		2.000.000.000		1.800.000.000	Edit
		- Rata2 Lama Sekolah (HLS)	7.5 Tahun		7.59 Tahun		Edit
4	1.01.02.2.01	Pengelolaan Pendidikan Sekolah Dasar		2.000.000.000		1.800.000.000	Edit
		- Presentase Akreditasi B untuk SD	95.7 %		95.6 %		Edit
5	1.01.02.2.01.0001	Pembangunan Unit Sekolah Baru (USB)		2.000.000.000		1.800.000.000	Edit
		- Jumlah sekolah yang terbangun	10 Sekolah		9 Sekolah		Edit

Gambar 11. Desain *User Interface* Monitoring dan Evaluasi

Pada desain UI halaman pencarian program pembangunan daerah dimunculkan input teks yang dapat digunakan untuk memasukkan kata kunci pencarian. Data hasil pencarian ditampilkan menurut similaritas dari nomenklatur yang memuat program, kegiatan, dan subkegiatan. Halaman pencarian program pembangunan ditampilkan pada Gambar 12.



CARI PROGRAM PEMBANGUNAN

Pembangunan Jembatan

Menampilkan 3 data

Program Pembangunan Jalan dan Jembatan	2 Jembatan – Rp. 1.000.000.000
Sub Kegiatan : Pembangunan Flyover	1 Flyover – Rp. 500.000.000
Sub Kegiatan : Pembangunan Jembatan	1 Jembatan – Rp. 500.000.000

Gambar 12. Desain *User Interface* Pencarian Program Pembangunan Daerah

3. HASIL PENELITIAN DAN PEMBAHASAN

Penelitian ini difokuskan pada pengembangan fitur pencarian semantik yang memanfaatkan pendekatan NLP dan pengukuran kesamaan teks menggunakan metode *cosine similarity*. Hasil implementasi sistem tersebut dijelaskan pada bagian berikut.

3.1. Arsitektur Sistem

Arsitektur sistem menggambarkan struktur umum dari komponen-komponen penyusun sistem e-monev berbasis NLP yang dikembangkan. Secara garis besar, sistem ini dibagi menjadi tiga lapisan utama, yaitu:

1. Lapisan Layar Pengguna (*Frontend*)

Pada sistem e-monev, teknologi yang digunakan untuk membangun antarmuka pengguna adalah Nuxt JS [16], yang merupakan *framework* dari Vue.js.

2. Lapisan Logika Aplikasi (*Backend*)

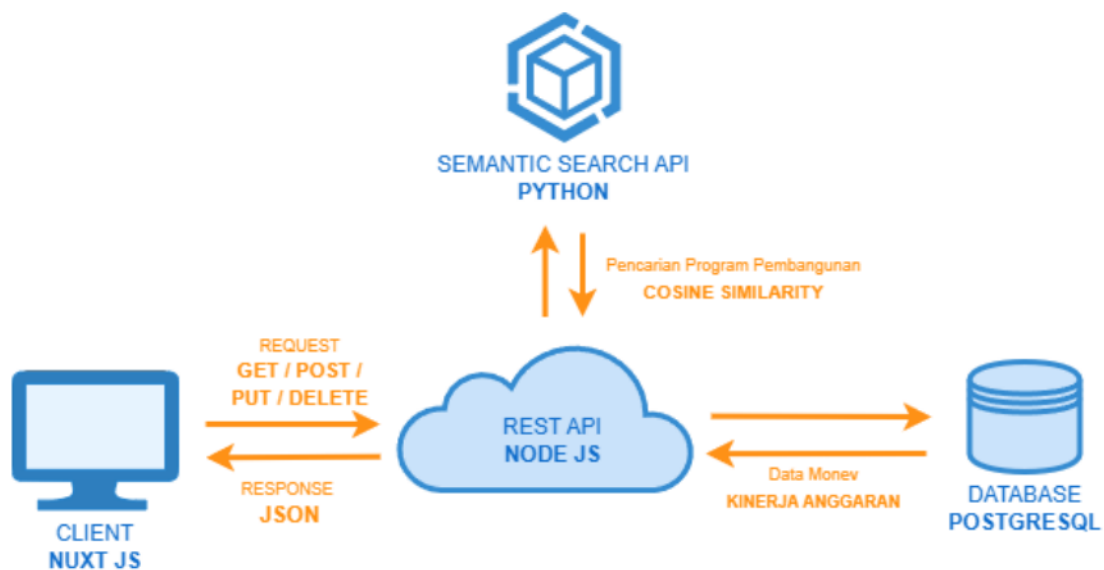
Backend sistem terdiri dari dua *REST API* utama yang dikembangkan menggunakan Node.js [17] dan Python [18].

- Node.js digunakan untuk menangani fungsi inti sistem e-monev, seperti manajemen data evaluasi, pengguna, dan OPD.
- Python digunakan khusus untuk fitur pencarian semantik.

3. Lapisan Penyimpanan Data (*Database*)

Sistem menggunakan RDBMS dari PostgreSQL [19]

Untuk lebih memperjelas hubungan antar komponen, berikut ditampilkan diagram arsitektur sistem pada Gambar 13.

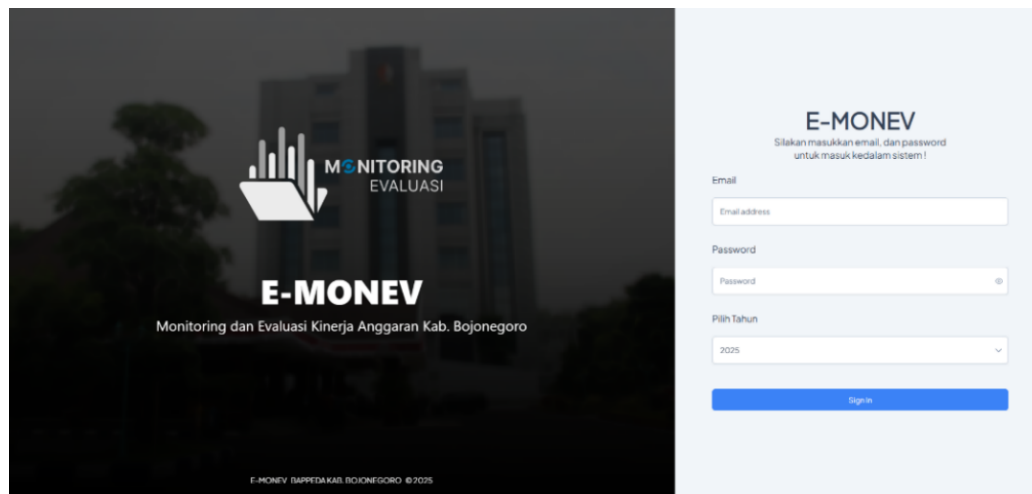


Gambar 13. Arsitektur Sistem E-Monev

3.2. Antar Muka Pengguna Sistem E-Monev

Halaman Login

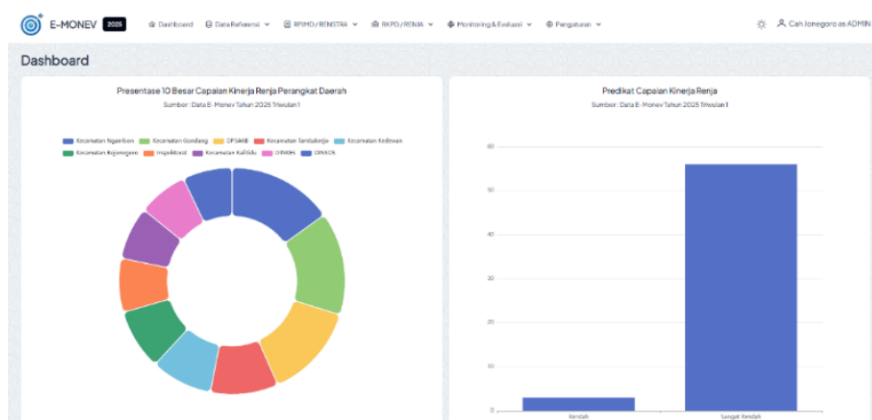
Pada halaman ini pengguna harus mengisi email, password, dan tahun evaluasi untuk masuk ke dalam sistem. Tampilan halaman login dapat dilihat pada gambar 14.



Gambar 14. Halaman Login E-Monev

Halaman Dashboard

Halaman ini menyajikan infografis capaian hasil evaluasi kinerja dan anggaran seluruh Organisasi Perangkat Daerah (OPD), yang meliputi: 10 besar capaian kinerja tertinggi (mendekati 100%), sebaran predikat capaian kinerja, 10 besar capaian serapan anggaran tertinggi (mendekati 100%), serta sebaran predikat capaian anggaran. Halaman *Dashboard* ditampilkan pada gambar 15.



Gambar 15. Halaman Dashboard E-Monev

Halaman Monitoring dan Evaluasi

Halaman ini memuat lembar kerja yang menampilkan data Urusan, Bidang Urusan, Tujuan, Sasaran, Program, Kegiatan, dan Sub Kegiatan, yang mencakup indikator kinerja dan anggaran pada tahun evaluasi serta realisasi capaian kinerja dan anggarannya per triwulan. Tampilan halaman tersebut ditunjukkan pada Gambar 16.

E-MONEV

2024

Dashboard

Data Referensi

RPMD / RENDITA

RPMD / RENDITA

Monitoring & Evaluasi

Pengukuran

Cat. Ikonografi as IDN

Konfigurasi

Download Laporan

Kode Rekening	Uraian / Bidang Uraian / Tujuan / Program / Kegiatan / Sub Kegiatan	Indikator Kinerja	Satuan Indikator	Target RPMD sampai pada Tahun 2024 (skala Persentase RPMD)		Realisasi Capaian Kinerja RPMD s.d RPMD Tahun Lalu (2023)		Target Kinerja & Anggaran RPMD Tahun 2024 yang Dievaluasi		Realisasi Kinerja Pada Tahunan					
				K	Rp.	K	Rp.	K	Rp.	I		II		III	
										K	Rp.	K	Rp.	K	Rp.
1	URAIAN PEMERINTAHAN KAWILAYATAN DENGAN PELAKSIAN DASAR			849.204.302.731				401.034.371.312		4.834.670.520		38.088.111.524		110.016.924.15	
1.01	URAIAN PEMERINTAHAN BIDANG PENDIDIKAN			849.204.302.731				401.034.371.312		4.834.670.520		38.088.111.524		110.016.924.15	
	Tujuan: Meningkatkan kualitas pendidikan														
Sub Kegiatan	Over - Angkutan Lintas Substansi (ALKS)	%		13,16				13,16							
Sub Kegiatan	Over - Rata-rata Lama Sekolah (RLS)	tahun		7,50				7,50							
	Sasaran: Meningkatkan Mutu dan Akses Pendidikan														
Sub Kegiatan	Over - APK PALU	%		97,99				97,94							
Sub Kegiatan	Over - APK SD	%		107,46				107,44							
Sub Kegiatan	Over - APK SMP	%		99,42				99,42							
Sub Kegiatan	Over - APK SMP	%		99				98,95							
Sub Kegiatan	Over - APK SD	%		99,89				99,87							
Sub Kegiatan	Over - Rapor pendidikan SD	100		99				98,90							

Gambar 16. Halaman Monitoring dan Evaluasi

Halaman Pencarian Program Pembangunan

Halaman pencarian program pembangunan menyediakan input teks untuk memasukkan kata kunci pencarian. Hasil pencarian ditampilkan berdasarkan tingkat similaritas terhadap nomenklatur program, kegiatan, dan subkegiatan. Selain nomenklatur, hasil juga mencakup target dan realisasi kinerja serta anggaran, beserta skor *cosine similarity*-nya. Halaman tersebut ditampilkan pada Gambar 17.

PENCARIAN PROGRAM PEMBANGUNAN			
Pembangunan Jembatan Cari Data			
Menampilkan 10 data			
1.03.10.2.01.0040 Pembangunan Jembatan Sub Kegiatan	Jumlah Jembatan yang Dibangun Target : 4 Jembatan Realisasi : 1 Jembatan Capaian : 25 %	Anggaran Target : 15.208.381.315 Realisasi : 7.790.714.367 Capaian : 51,23 %	Cosine Similarity : 1
1.03.10.2.01.0032 Pembangunan Jalan Sub Kegiatan	Panjang Jalan yang Dibangun Target : 70 Km Realisasi : 70 Km Capaian : 100 %	Anggaran Target : 140.691.611.410 Realisasi : 133.067.049.686 Capaian : 94,58 %	Cosine Similarity : 0,97
1.03.10.2.01.0039 Rehabilitasi Jembatan Sub Kegiatan	Jumlah Jembatan yang Direhabilitasi Target : 79 Jembatan Realisasi : 79 Jembatan Capaian : 100 %	Anggaran Target : 47.521.486.825 Realisasi : 42.406.858.153 Capaian : 89,24 %	Cosine Similarity : 0,96
1.03.10.2.01.0035 Pembangunan Flyover	Jumlah Jembatan yang Direhabilitasi	Anggaran	

Gambar 17. Hasil Pencarian Program Pembangunan

3.3. Penerapan Cosine Similarity dalam Proses Pencarian Program Pembangunan Daerah

Penelitian ini menggunakan metode *cosine similarity* untuk menghitung tingkat kemiripan antara dua vektor teks, yaitu vektor dari *query* pencarian dan vektor dari data program pembangunan. Adapun rumus dari *cosine similarity* adalah sebagai berikut [20]:

$$\text{cosine_similarity} = \frac{A \cdot B}{\|A\| \|B\|} \quad (1)$$

Keterangan :

- A dan B adalah dua vektor.
- $A \cdot B$ adalah dot product (hasil kali titik) dari vektor A dan B.
- $\|A\|$ dan $\|B\|$ adalah norm (panjang vektor) dari A dan B.

Berikut ini adalah penjabaran rumus dari dot product $A.B$:

$$A.B = \sqrt{\sum_{i=1}^n A_i^2 \cdot B_i^2} \quad (2)$$

Keterangan :

- A_i : Elemen ke-i dari vektor A.
- B_i : Elemen ke-i dari vektor B.
- n : Jumlah dimensi vektor (jumlah elemen dalam vektor)

Berikut ini adalah penjabaran *norm* vektor

$$\|A\| = \sqrt{\sum_{i=1}^n A_i^2} \quad (3)$$

Keterangan :

- A_i : Elemen ke-i dari vektor A.
- n : Jumlah dimensi vektor.

Pengujian Akurasi Semantik Pencarian

Pencarian program secara semantik dilakukan dengan memanfaatkan BERT sebagai model representasi teks yang menghasilkan vektor *embedding* untuk setiap entri dalam *dataset*. Berikut ini ditampilkan hasil pencarian dengan query “pembangunan jembatan”, yang direpresentasikan dalam bentuk vektor *embedding* [0.2235, 1.1393, 0.5842, 0.6146, 1.7925]. Representasi vektor tersebut ditampilkan pada Tabel 1.

Tabel 1. Hasil Pencarian Program Pembangunan

No	Program / Kegiatan / Sub Kegiatan	Vektor Embedding	Cosine Similarity
1	Pembangunan Jembatan	[0.2235, 1.1393, 0.5842, 0.6146, 1.7925]	1
2	Pembangunan Bendungan	[0.1228, 1.4497, 0.5963, 0.3174, 1.6575]	0.9726
3	Pembangunan Jalan	[0.4507, 1.3421, 0.6673, 0.3799, 1.7254]	0.9711
4	Pelebaran Jembatan	[0.0501, 0.8853, 0.4110, 0.6196, 1.7130]	0.9706
5	Penggantian Jembatan	[0.4510, 1.0810, 0.8648, 0.5983, 2.0137]	0.9596
6	Rehabilitasi Jembatan	[0.1211, 1.2551, 0.5611, 0.4920, 1.8231]	0.9594
7	Pembangunan Flyover	[-0.3715, 1.2560, 0.5842, 0.6146, 1.7925]	0.9585
8	Pemeliharaan Rutin Jembatan	[0.4475, 1.4604, 0.7352, 0.6935, 1.8654]	0.9447
9	Pembangunan Prasarana Pertanian	[0.0764, 1.8102, 0.6215, 0.5512, 1.6988]	0.9384
10	Pembangunan Puskesmas	[0.3815, 1.4053, 1.18092, 0.8388, 1.1047]	0.9164

Setelah mendapatkan hasil vektor *embedding* pada *dataset*, penelitian ini dilakukan untuk uji akurasi semantik pencarian untuk kata kunci Pembangunan Jembatan dan diuji dengan hasil pencarian pada Tabel 1 dengan data nomor 7 Pembangunan *Flyover*. Untuk menghitung kesamaan semantik antara kata kunci dan *dataset*, digunakan metode *cosine similarity*. Misalnya, vektor representasi untuk kata kunci “Pembangunan Jembatan” (A) adalah [0.2235, 1.1393, 0.5842, 0.6146, 1.7925], sedangkan vektor representasi untuk entri *dataset* “Pembangunan *Flyover*” (B) adalah [-0.3715, 1.2560, 0.5842, 0.6146, 1.7925]. Perhitungan dimulai dengan mencari dot product antara vektor A dan B, yaitu:

$$A \cdot B = (0.2235 \times -0.3715) + (1.1393 \times 1.2560) + (0.5842 \times 0.5842) + (0.6146 \times 0.6146) + (1.7925 \times 1.7925) = -0.0830 + 1.4309 + 0.3412 + 0.3777 + 3.2130 = 5.2798$$

Selanjutnya, dihitung nilai norm dari masing-masing vektor. Norm vektor A diperoleh dari:

$$\|A\| = \sqrt{0.2235^2 + 1.1393^2 + 0.5842^2 + 0.6146^2 + 1.7925^2} = \sqrt{5.2800} = 2.29$$

Sedangkan norm vektor B adalah:

$$\|B\| = \sqrt{-0.3715^2 + 1.2560^2 + 0.5842^2 + 0.6146^2 + 1.7925^2} = \sqrt{5.6477} = 2.37$$

Dengan demikian, nilai *cosine similarity* antara kata kunci dan *dataset* dapat dihitung sebagai berikut:

$$\text{cosine_similarity} = \frac{5.2023}{2.29 \times 2.37} = \frac{5.2023}{5.4273} = 0.958$$

Nilai *cosine similarity* sebesar 0.958 menunjukkan bahwa "Pembangunan Jembatan" dan "Pembangunan Flyover" memiliki kemiripan makna yang sangat tinggi. Artinya, sistem berhasil mengenali bahwa kedua istilah tersebut saling relevan dalam konteks pencarian program pembangunan dan layak ditampilkan dalam hasil pencarian.

Hasil Pengujian Precision

Pengujian *precision* dilakukan untuk mengukur tingkat relevansi hasil pencarian dari sistem e-monev berbasis *cosine similarity*. Lima *query* uji digunakan sebagai representasi variasi topik pembangunan daerah, dan masing-masing dievaluasi oleh lima evaluator (Eval 1 hingga Eval 5). Evaluator memberikan penilaian terhadap 10 hasil teratas (Top-10) yang dihasilkan sistem, dengan mencatat jumlah hasil yang dianggap relevan. Nilai *precision* dihitung dari rata-rata relevansi tiap *query* terhadap jumlah total hasil (10) [21]. Berdasarkan hasil evaluasi, *precision* rata-rata yang dicapai adalah sebesar 0,884. Hasil evaluasi ditampilkan pada tabel 2.

Tabel 2. Evaluasi Ujicoba Pencarian Program Pembangunan

Query	Eval 1	Eval 2	Eval 3	Eval 4	Eval 5	Precision
Pembangunan Jalan	9/10	8/10	8/10	8/10	8/10	0.82
Pembangunan Jembatan	8/10	8/10	8/10	8/10	8/10	0.80
Makan Bergizi untuk Anak Sekolah	9/10	9/10	9/10	9/10	9/10	0.90
Ketahanan Pangan	10/10	10/10	10/10	10/10	10/10	1.00
Gedung Sekolah Dasar	9/10	9/10	9/10	9/10	9/10	0.90
Precision						0.884

Pada Tabel 2 di atas menyajikan hasil evaluasi uji coba pencarian program pembangunan daerah dengan menggunakan lima *query*, yaitu Pembangunan Jalan, Pembangunan Jembatan, Makan Bergizi untuk Anak Sekolah, Ketahanan Pangan, dan Gedung Sekolah Dasar. Dari tabel tersebut dapat didapatkan hasil pada setiap kolom Eval berisikan n/10 dimana n adalah jumlah pencarian yang dianggap relevan oleh evaluator. Dilihat bahwa *precision* tertinggi sebesar 1.00 diperoleh dari kata kunci Ketahanan Pangan dan Gedung Sekolah Dasar, sementara *precision* terendah adalah 0.80 yang diperoleh dari kata kunci Pembangunan Jembatan. Dari lima kata kunci yang diuji oleh evaluator menghasilkan rata-rata *precision* sebesar 0,884. Hal ini menunjukkan bahwa secara umum, sistem mampu menghasilkan pencarian dengan tingkat relevansi yang sangat baik.

3.4. Penulisan Kode Sumber Pencarian Semantik Menggunakan Cosine Similarity

Berikut ini adalah *source code* pencarian program pembangunan menggunakan metode *cosine similarity* dalam bahasa Python [18]. *Source code* ini memiliki fungsi *search* dengan parameter *query* sebagai kata kunci pencarian, dan menggunakan *dataset* Kodefikasi dan Nomenklatur Perencanaan Pembangunan Keuangan Daerah [2]. Contoh penulisan kode sumber ditunjukkan pada kode sumber 1 berikut :

```
def search(self, query):
    clean_query = TextPreprocessor.clean_text(query)

    query_embedding = self.get_bert_embedding(clean_query)[0]

    results = []
    for idx in range(len(self.df)):
        doc_embedding =
self.get_bert_embedding(self.df.iloc[idx]['clean_text'])[0]

        dot_product = np.dot(query_embedding, doc_embedding)

        magnitude_query = np.linalg.norm(query_embedding)
        magnitude_doc = np.linalg.norm(doc_embedding)

        similarity = dot_product / (magnitude_query * magnitude_doc)

        if similarity > Config.SIMILARITY_THRESHOLD:
            results.append({
                'kode': self.df.iloc[idx]['kode_rekening'],
                'nomenklatur': self.df.iloc[idx]['nomenklatur'],
                'similarity_score': float(similarity),
                'query_embedding':
query_embedding.tolist()[0:Config.VECTOR_LIMIT[1]],
                'doc_embedding':
doc_embedding.tolist()[0:Config.VECTOR_LIMIT[1]]
            })
        results.sort(key=lambda x: x['similarity_score'], reverse=True)
    return {
        'cosine_similarity': results[0:Config.TOP_K_RESULTS],
        'result_all': results
    }
```

Kode Sumber 1. Pencarian Program Pembangunan *Cosine Similarity*

Dari kode sumber 1 diatas dapat dijelaskan bahwa fungsi *search (self, query)* merupakan inti dari pencarian semantik pada sistem e-monev. Fungsi ini mencocokkan masukan pengguna dengan data nomenklatur program pembangunan menggunakan BERT *embedding* dan *cosine similarity*. Prosesnya terbagi dalam beberapa tahap.

Pertama, sistem melakukan pembersihan terhadap input *query* menggunakan fungsi *clean_text*. Pada tahap ini, karakter atau tanda baca yang tidak relevan dihapus agar teks dalam kondisi bersih dan sesuai untuk diproses lebih lanjut oleh model. Setelah teks dibersihkan, *query* kemudian dikonversi menjadi vektor *embedding* melalui model BERT. Proses ini memungkinkan sistem memahami konteks semantik dari kata atau frasa yang dimasukkan oleh pengguna. Selanjutnya, vektor *query* dibandingkan dengan vektor *embedding* dari seluruh entri nomenklatur yang telah diproses sebelumnya. Sistem menghitung tingkat kemiripan antara vektor *query* dan masing-masing vektor dokumen menggunakan metode *cosine similarity*. Nilai kemiripan ini menjadi dasar dalam menentukan relevansi antara *query* dan dokumen dalam database.

Dari hasil perhitungan tersebut, sistem hanya menyimpan dokumen yang memiliki skor similarity melebihi ambang batas yang telah ditetapkan pada konfigurasi sistem (Config.SIMILARITY_THRESHOLD). Setelah proses seleksi selesai, hasil pencarian diurutkan berdasarkan skor similarity tertinggi. Sepuluh hasil teratas kemudian dikembalikan kepada pengguna, sesuai dengan parameter konfigurasi Config.TOP_K_RESULTS. Fungsi ini memungkinkan pencarian yang lebih relevan secara semantik, tidak lagi terbatas pada pencocokan literal kata kunci. Implementasi ini meningkatkan akurasi dan pengalaman pengguna dalam menemukan program atau kegiatan pembangunan yang sesuai.

4. KESIMPULAN DAN SARAN

Berdasarkan penelitian yang telah dilakukan, dapat disimpulkan bahwa penerapan *Natural Language Processing* (NLP) dengan metode *cosine similarity* berhasil meningkatkan relevansi hasil pencarian program pembangunan pada sistem e-monev secara signifikan. Sistem mampu mengenali kemiripan makna antar kalimat, sehingga pencarian tidak lagi terbatas pada kecocokan kata kunci secara literal. Dengan demikian, pengguna dapat menemukan program, kegiatan, atau sub kegiatan melalui pencarian berbasis semantik yang lebih kontekstual. Hasil pengujian menunjukkan nilai *precision* sebesar 0,884, yang menandakan bahwa sistem memiliki tingkat akurasi yang tinggi dalam menampilkan hasil pencarian yang relevan terhadap input pengguna. Temuan ini menjadi keunggulan utama dibandingkan pendekatan berbasis kata kunci konvensional yang hanya menampilkan data sesuai dengan kata kunci yang dimasukkan, sehingga proses monitoring dan evaluasi pada sistem e-monev dapat memberikan pencarian informasi yang lebih baik terhadap pengguna.

Sebagai tindak lanjut, terdapat beberapa saran pengembangan. Pertama, sistem disarankan untuk dilengkapi atribut tambahan berupa definisi operasional dalam *dataset* guna memperluas cakupan makna nomenklatur program pembangunan. Hal ini akan meningkatkan konteks pemahaman dalam pencarian semantik. Kedua, sistem perlu diuji lebih lanjut dalam skala yang lebih luas dengan melibatkan lebih banyak Organisasi Perangkat Daerah (OPD), agar kemampuannya dapat beradaptasi dengan kebutuhan nyata di lapangan secara lebih menyeluruh dan representatif.

PERNYATAAN PENGGUNAAN ARTIFICIAL INTELLIGENCE

Penulis menggunakan ChatGPT dan Gemini untuk membantu penyuntingan bahasa, penyempurnaan struktur penulisan, dan peningkatan keterbacaan artikel ilmiah. Setiap saran yang dihasilkan oleh kedua AI tersebut telah ditinjau dan disesuaikan dengan konteks penelitian. Penulis bertanggung jawab penuh atas keakuratan informasi dan seluruh isi artikel.

KONTRIBUSI PENULIS

Moch Faizal Ansyori: Konsep, Metode, Pengembangan Sistem, Penulisan Artikel, Pengumpulan Data.

Ahmad Heru Mujianto: Konsep, Metode, Pengembangan Sistem, Penulisan Artikel, Supervisi, Review, Persetujuan Akhir Artikel.

PENDANAAN

Penelitian ini tidak menerima hibah khusus dari lembaga pendanaan mana pun, baik sektor publik, komersial, maupun nirlaba.

KONFLIK KEPENTINGAN

Penulis menyatakan tidak terdapat konflik kepentingan terkait publikasi artikel ini.

DAFTAR PUSTAKA

- [1] M. D. Maulidin, Efektivitas Pelaksanaan Monitoring dan Evaluasi Berbasis Aplikasi e-Monev pada Rencana Kerja Organisasi Perangkat Daerah Periode 2017 (Studi Kasus Pada Badan Perencanaan Pembangunan Daerah Kabupaten Bojonegoro). Universitas Brawijaya, 2017. [Online]. Available: <https://repository.ub.ac.id/id/eprint/162822/>
- [2] Kementerian Dalam Negeri Republik Indonesia, "Keputusan Menteri Dalam Negeri No. 900.1.15.5-3406 Tahun 2024." Kementerian Dalam Negeri, 2024. [Online]. Available: <https://keuda.kemendagri.go.id/produkhukum/download/796/kepmendagri-no-9001155-3406-tahun-2024>
- [3] E. Yulianti and N. K. Nissa, "ABSA of Indonesian customer reviews using IndoBERT: single-sentence and sentence-pair classification approaches," *Bull. Electr. Eng. Inform.*, vol. 13, no. 5, pp. 3579–3589, Oct. 2024, doi: 10.11591/eei.v13i5.8032.
- [4] M. Amien, "Sejarah dan Perkembangan Teknik Natural Language Processing (NLP) Bahasa Indonesia: Tinjauan tentang sejarah, perkembangan teknologi, dan aplikasi NLP dalam bahasa Indonesia," 2023, arXiv. doi: 10.48550/ARXIV.2304.02746.
- [5] L. Zahrotun, "Comparison Jaccard similarity, Cosine Similarity and Combined Both of the Data Clustering With Shared Nearest Neighbor Method," *Comput. Eng. Appl. J.*, vol. 5, no. 1, pp. 11–18, Feb. 2016, doi: 10.18495/comengapp.v5i1.160.
- [6] C. D. Manning, P. Raghavan, and H. Schütze, *Introduction to information retrieval*. New York: Cambridge University Press, 2008.
- [7] Y. E. Rachmad, R. Ilham, and N. Indrayani, *Layanan dan Tata Kelola E-Government*. Green Pustaka Indonesia, 2024.
- [8] R. R. Putra, N. A. Putri, and A. D. Putra, *Teknik Cosine Similarity Dan TF-IDF Dalam Analisis Data*. Serasi Media Teknologi, 2024.
- [9] D. Gunawan, C. A. Sembiring, and M. A. Budiman, "The Implementation of Cosine Similarity to Calculate Text Relevance between Two Documents," *J. Phys. Conf. Ser.*, vol. 978, p. 012120, Mar. 2018, doi: 10.1088/1742-6596/978/1/012120.
- [10] J. T. Santoso, *Desain & Analisis Sistem Berorientasi Obyek dengan UML*. in Migunani. Universitas STEKOM, 2021.
- [11] L. P. Sumirat, D. Cahyono, and Y. Kristyawan, *Dasar Rekayasa Perangkat Lunak*. Madza Media, 2023.
- [12] A. S and P. Gnanasekaran, "Juncture of Text Preprocessing Techniques & Extracting Sentiment Analyzing of Micro-Blog Based on Machine Learning Algorithms," in *2023 International Conference on Innovative Computing, Intelligent Communication and Smart Electrical Systems (ICSES)*, Chennai, India: IEEE, Dec. 2023, pp. 1–9. doi: 10.1109/ICSES60034.2023.10465450.
- [13] M. V. Koroteev, "BERT: A Review of Applications in Natural Language Processing and Understanding," 2021, arXiv. doi: 10.48550/ARXIV.2103.11943.
- [14] S. Ramdany, "Penerapan UML Class Diagram dalam Perancangan Sistem Informasi Perpustakaan Berbasis Web," *J. Ind. Eng. Syst.*, vol. 5, no. 1, Jul. 2024, doi: 10.31599/2e9afp31.
- [15] Suhendri and D. Susanti, *Desain User Interface (UI)/User Experience (UX) Menggunakan Figma*. PT. Adab Indonesia, 2024.
- [16] H. Yang, *Vue. JS Framework: Design and Implementation*. Singapore: Springer Nature Singapore, 2023. doi: 10.1007/978-981-99-4947-2.
- [17] G. Sidelnikov, *Node.js API*, 1st ed. Learning Curve Books, 2020.
- [18] Sholihun and Z. S. Fatomi, *Pemrograman dan komputasi numerik: menggunakan Python*, Cetakan pertama. Sleman, Depok, D. I. Yogyakarta: Gadjah Mada University Press, 2021.

- [19] L. Ferrari and E. Pirozzi, *Learn PostgreSQL: build and manage high-performance database solutions using PostgreSQL 12 and 13*. Birmingham Mumbai: Packt, 2020.
- [20] R. Al Rasyid and D. H. U. Ningsih, "Penerapan Algoritma TF-IDF dan Cosine Similarity untuk Query Pencarian Pada Dataset Destinasi Wisata," *J. JTIK J. Teknol. Inf. Dan Komun.*, vol. 8, no. 1, pp. 170–178, Jan. 2024, doi: 10.35870/jtik.v8i1.1416.
- [21] F. A. Nugroho, F. Septian, D. A. Pungkastyo, and J. Riyanto, "Penerapan Algoritma Cosine Similarity untuk Deteksi Kesamaan Konten pada Sistem Informasi Penelitian dan Pengabdian Kepada Masyarakat," *J. Inform. Univ. Pamulang*, vol. 5, no. 4, p. 529, Dec. 2021, doi: 10.32493/informatika.v5i4.7126.