**MULAI DARI 0 YA BODOHHH**

**PENGEMBANGAN WEBSITE E-COMMERCE CLOTHING DENGAN SISTEM REKOMENDASI PRODUK BERBASIS CONTENT-BASED FILTERING DAN METODE PROTOTYPING**



**Disusun oleh:**

**Joyva Alians Tamam**

**4.33.21.2.14**

**PROGRAM STUDI S.Tr TEKNOLOGI REKAYASA KOMPUTER**

**JURUSAN TEKNIK ELEKTRO**

**POLITEKNIK NEGERI SEMARANG**

**2024**

**PENGEMBANGAN WEBSITE E-COMMERCE CLOTHING DENGAN SISTEM REKOMENDASI PRODUK BERBASIS CONTENT-BASED FILTERING DAN METODE PROTOTYPING**



Skripsi ini disusun untuk melengkapi sebagian persyaratan menjadi Sarjana Terapan

**Disusun oleh:**

**Joyva Alians Tamam**

**4.33.21.2.14**

**PROGRAM STUDI S.Tr TEKNOLOGI REKAYASA KOMPUTER**

**JURUSAN TEKNIK ELEKTRO**

**POLITEKNIK NEGERI SEMARANG**

**2024**

# PERNYATAAN KEASLIAN SKRIPSI

Saya menyatakan dengan sesungguhnya bahwa skripsi dengan judul **Pengembangan Website E-Commerce Clothing Dengan Sistem Rekomendasi Produk Berbasis Content-Based Filtering Dan Metode Protyping** yang dibuat untuk melengkapi sebagian persyaratan menjadi Sarjana Terapan pada Program Studi S.Tr Teknologi Rekayasa Komputer Jurusan Teknik Elektro Politeknik Negeri Semarang, sejauh yang saya ketahui bukan merupakan tiruan atau duplikasi dari skripsi yang sudah dipublikasikan dan atau pernah dipakai untuk mendapatkan gelar Sarjana terapan di lingkungan Politeknik negeri Semarang maupun di perguruan tinggi atau instansi manapun, kecuali bagian yang sumber informasinya dicantumkan sebagaimana mestinya.

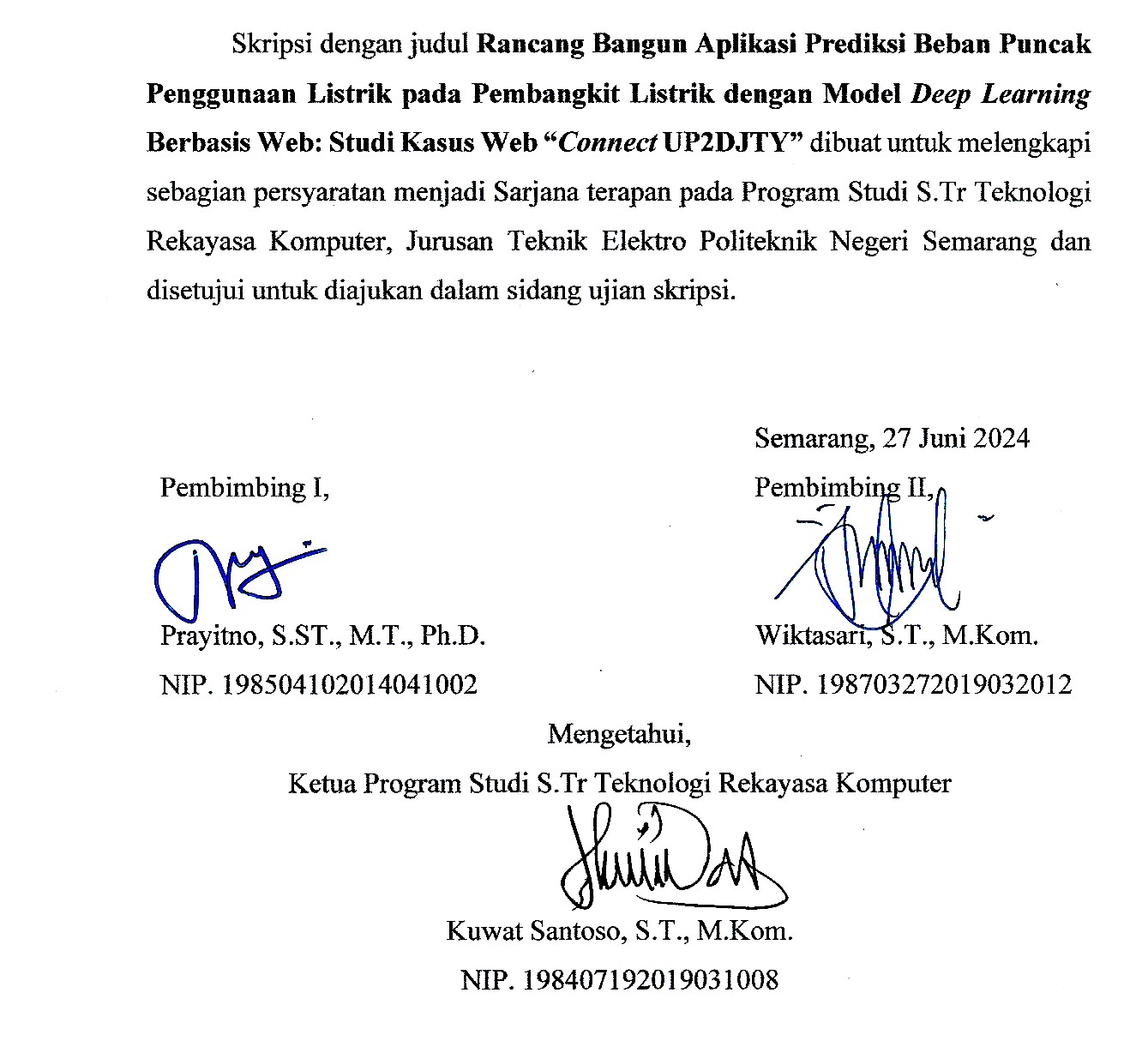
Semarang, 27 Juni 2024

Penulis

Joyva Alians Tamam

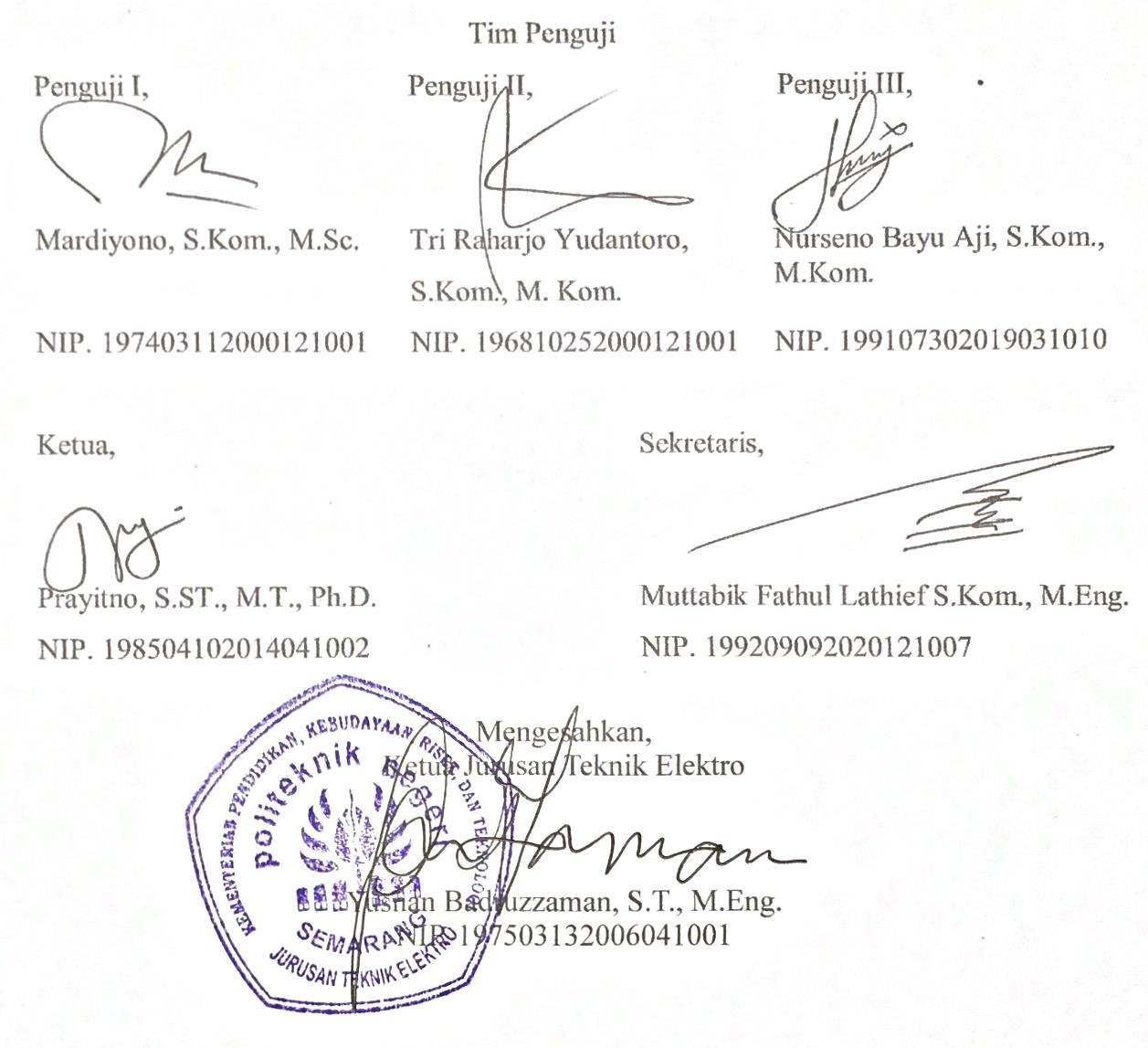
NIM 4.33.21.2.14

# HALAMAN PERSETUJUAN



# HALAMAN PENGESAHAN

Skripsi dengan judul **Pengembangan Website E-Commerce Clothing Dengan Sistem Rekomendasi Produk Berbasis Content-Based Filtering Dan Metode Protyping** telah dipertahankan dalam ujian wawancara dan diterima sebagai syarat untuk menjadi Sarjana Terapan pada Program Studi S.Tr Teknologi Rekayasa Komputer Jurusan Teknik Elektro Politeknik Negeri Semarang pada tanggal 5 Juli 2024.



# KATA PENGANTAR

Puji syukur atas segala nikmat dan karunia yang telah diberikan oleh Tuhan Yang Maha Esa, sehingga penulis dapat menyelesaikan skripsi sebagai salah satu rangkaian akademis dari Program Studi S.Tr Teknologi Rekayasa Komputer Jurusan Teknik Elektro Politeknik Negeri Semarang.

Adapun tujuan dari penulisan skripsi ini adalah untuk memenuhi persyaratan kelulusan mahasiswa yang dibuat oleh Program Studi S.Tr Teknologi Rekayasa Komputer. Serta sebagai sarana belajar bagi para pembaca yang ingin mengetahui pembuatan dan hasil dari Pengembangan Website E-Commerce Clothing Dengan Sistem Rekomendasi Produk Berbasis Content-Based Filtering Dan Metode Protyping.

Penulis ucapkan terima kasih kepada seluruh pihak yang telah memberikan bantuan, dukungan, serta doanya kepada penulis untuk menyelesaikan program ini secara baik. Pihak – pihak yang terkait itu diantaranya sebagai berikut:

1. Bapak Yusnan Badruzzaman, S.T.,M.Eng, Ketua Jurusan Teknik Elektro Politeknik Negeri Semarang
2. Kuwat Santoso, S.T., M.Kom., Ketua Program Studi S.Tr Teknologi Rekayasa Komputer Politeknik Negeri Semarang.
3. Prayitno, S.ST., M.T., Ph.D., Dosen Pembimbing 1 (BELUM)
4. Wiktasari, S.T., M.Kom., Dosen Pembimbing 2 (BELUM)
5. Keluarga dari penulis
6. Rizki Shaumi Sabiq, Ashabul Kahfi, dan Afif Ramzy Badrani, Bukhary Azriellorezqa Yufar sahabat dari penulis.

Semarang, 4 Mei 2024

Penulis

Joyva Alians Tamam NIM 4.33.20.0.09

# ABSTRAK

*Joyva Alians Tamam, "Pengembangan Website E-Commerce Clothing Dengan Sistem Rekomendasi Produk Berbasis Content-Based Filtering Dan Metode Prototyping", Skripsi DIV Jurusan Teknik Elektro Politeknik Negeri Semarang, di bawah bimbingan Prayitno, S.ST., M.T., Ph.D. dan Wiktasari, S.T., M.Kom., Juli 2025, 53.*

*Industri e-commerce fashion di Indonesia mengalami pertumbuhan pesat, namun masih menghadapi tantangan dalam memberikan pengalaman berbelanja yang personal dan relevan bagi konsumen. Banyak platform e-commerce yang belum mampu memberikan rekomendasi produk yang akurat sesuai preferensi pengguna, sehingga menyebabkan rendahnya tingkat konversi dan kepuasan pelanggan. Penelitian ini bertujuan untuk mengembangkan website e-commerce clothing dengan sistem rekomendasi produk cerdas berbasis content-based filtering menggunakan metode prototyping.*

*Penelitian ini menggunakan metode prototyping untuk pengembangan sistem secara iteratif dan algoritma content-based filtering dengan pendekatan Term Frequency-Inverse Document Frequency (TF-IDF) dan Cosine Similarity untuk menganalisis kesamaan produk berdasarkan fitur konten seperti nama, kategori, dan deskripsi produk. Sistem dikembangkan menggunakan teknologi React.js untuk frontend, Node.js dengan Express.js untuk backend, MongoDB sebagai database, dan Python Flask untuk service machine learning. Data produk clothing diproses melalui tahap preprocessing, feature extraction menggunakan TF-IDF vectorization, dan perhitungan similarity score untuk menghasilkan rekomendasi yang relevan.*

*Hasil penelitian menunjukkan bahwa sistem rekomendasi content-based filtering berhasil diimplementasikan dengan tingkat akurasi yang baik dalam memberikan rekomendasi produk serupa berdasarkan preferensi pengguna. Sistem mampu melakukan auto-sync data produk, memberikan rekomendasi real-time melalui REST API, dan mengintegrasikan fallback system untuk memastikan stabilitas aplikasi. Testing workflow menunjukkan bahwa aplikasi dapat menangani operasi CRUD produk, sync otomatis ke recommendation service, dan menampilkan similarity score dengan akurat.*

*Implementasi sistem ini berupa aplikasi web full-stack dengan arsitektur microservice yang memisahkan service rekomendasi dari aplikasi utama. Aplikasi ini memungkinkan pemilik toko online clothing untuk mengelola produk dan memberikan pengalaman berbelanja yang lebih personal kepada pelanggan melalui sistem rekomendasi yang cerdas. Penelitian ini memberikan kontribusi praktis bagi industri e-commerce dalam meningkatkan engagement pelanggan dan optimalisasi penjualan melalui personalisasi produk.*

***Kata kunci:****E-Commerce, Content-Based Filtering, TF-IDF, Cosine Similarity, Sistem Rekomendasi, Prototyping, React.js, Machine Learning.*

# *ABSTRACT*

*Joyva Alians Tamam, "Development of E-Commerce Clothing Website with Product Recommendation System Based on Content-Based Filtering and Prototyping Method", Thesis of Applied Bachelor Program, Department of Electrical Engineering, State Polytechnic of Semarang, under the guidance of Prayitno, S.ST., M.T., Ph.D. and Wiktasari, S.T., M.Kom., July 2025, 53.*

*The fashion e-commerce industry in Indonesia is experiencing rapid growth, but still faces challenges in providing personalized and relevant shopping experiences for consumers. Many e-commerce platforms have not been able to provide accurate product recommendations according to user preferences, resulting in low conversion rates and customer satisfaction. This research aims to develop an e-commerce clothing website with an intelligent product recommendation system based on content-based filtering using prototyping methodology.*

*This research employs prototyping method for iterative system development and content-based filtering algorithm with Term Frequency-Inverse Document Frequency (TF-IDF) and Cosine Similarity approaches to analyze product similarity based on content features such as product name, category, and description. The system is developed using React.js for frontend, Node.js with Express.js for backend, MongoDB as database, and Python Flask for machine learning service. Clothing product data is processed through preprocessing stages, feature extraction using TF-IDF vectorization, and similarity score calculation to generate relevant recommendations.*

*The research results show that the content-based filtering recommendation system has been successfully implemented with good accuracy in providing similar product recommendations based on user preferences. The system is capable of performing automatic product data synchronization, providing real-time recommendations through REST API, and integrating fallback system to ensure application stability. Testing workflow demonstrates that the application can handle product CRUD operations, automatic sync to recommendation service, and display similarity scores accurately.*

*The system implementation is a full-stack web application with microservice architecture that separates recommendation service from the main application. This application enables clothing online store owners to manage products and provide more personalized shopping experiences to customers through intelligent recommendation system. This research provides practical contributions to the e-commerce industry in improving customer engagement and sales optimization through product personalization.*

***Keywords:****E-Commerce, Content-Based Filtering, TF-IDF, Cosine Similarity, Recommendation System, Prototyping, React.js, Machine Learning.*

# DAFTAR ISI

**HALAMAN JUDUL** ............................................................................................. ii

[PERNYATAAN KEASLIAN SKRIPSI iii](#_Toc205081652)

[HALAMAN PERSETUJUAN iv](#_Toc205081653)

[HALAMAN PENGESAHAN iv](#_Toc205081654)

[KATA PENGANTAR v](#_Toc205081655)

[ABSTRAK vii](#_Toc205081656)

[*ABSTRACT* ix](#_Toc205081657)

[DAFTAR ISI xi](#_Toc205081658)

[DAFTAR TABEL xii](#_Toc205081659)

[DAFTAR GAMBAR xiii](#_Toc205081660)

[BAB I 1](#_Toc205081661)

[PENDAHULUAN 1](#_Toc205081662)

[1.1. Latar Belakang 1](#_Toc205081663)

[1.2. Rumusan Masalah 3](#_Toc205081664)

[1.3. Batasan Masalah 3](#_Toc205081665)

[1.4. Tujuan 3](#_Toc205081666)

[1.5. Manfaat 4](#_Toc205081667)

[1.6. Sistematika Penulisan 4](#_Toc205081668)

[BAB II 7](#_Toc205081669)

[TINJAUAN PUSTAKA 7](#_Toc205081670)

[2.1. Dasar Teori 7](#_Toc205081671)

[2.2. Penelitian Terkait 10](#_Toc205081672)

[2.2. Alat dan Teknologi yang Digunakan 18](#_Toc205081673)

[BAB III 20](#_Toc205081674)

[KEGIATAN PELAKSANAAN 20](#_Toc205081675)

[3.1. Metode Pelaksanaan 20](#_Toc205081676)

[3.2. Gambaran Umum Sistem 22](#_Toc205081677)

[3.3. Analisis Kebutuhan 24](#_Toc205081678)

[3.4. Perancangan Sistem 25](#_Toc205081679)

[BAB IV 30](#_Toc205081680)

[ANALISIS DAN PEMBAHASAN 30](#_Toc205081681)

[4.1. Analisis Algoritma 30](#_Toc205081682)

[BAB V 46](#_Toc205081683)

[KESIMPULAN 46](#_Toc205081684)

[5.1. Kesimpulan 46](#_Toc205081685)

[5.2. Saran 46](#_Toc205081686)

[DAFTAR PUSTAKA 48](#_Toc205081687)

[LAMPIRAN 53](#_Toc205081688)

[A. Lembar Kontrol Bimbingan 53](#_Toc205081689)

[B. Surat Keterangan Selesai bimbingan Skripsi 55](#_Toc205081690)

[C. Surat Keterangan Siap Ujian Skripsi 56](#_Toc205081691)

[D. Surat Tugas 57](#_Toc205081692)

[E. Lembar Revisi Skripsi 58](#_Toc205081693)

[F. Lembar Pernyataan Telah Melakukan Revisi 61](#_Toc205081694)

[G. Lembar Responden 62](#_Toc205081695)

[H. Dokumentasi 64](#_Toc205081696)

[I. Kode Program *Deep Learning* 65](#_Toc205081697)

[J. Kode Program *Flask* 66](#_Toc205081698)

[K. Kode Program Larvel *Controller* 69](#_Toc205081699)

[L. Kode Program Larvel *View* 69](#_Toc205081700)

# DAFTAR TABEL

Tabel 2. 1 Perbandingan Model Antar Jurnal pendahulu ...................................... 20 Tabel 4. 1 Contoh Dataset Simulasi Beban Puncak .............................................. 40 Tabel 4. 2 Skenario Ukuran Data .......................................................................... 41 Tabel 4. 3 Hasil Percobaan GRU Skenario 1 ........................................................ 42

Tabel 4. 4 Hasil Percobaan GRU Skenario 2 ........................................................ 42

Tabel 4. 5 Hasil Percobaan GRU Skenario 3 ........................................................ 42 Tabel 4. 6 Hasil Percobaan LSTM Skenario 1 ...................................................... 43

Tabel 4. 7 Hasil Percobaan LSTM Skenario 2 ...................................................... 43

Tabel 4. 8 Hasil Percobaan LSTM Skenario 3 ...................................................... 43 Tabel 4. 9 Hasil Percobaan RNN Skenario 1 ........................................................ 44 Tabel 4. 10 Hasil Percobaan RNN Skenario 2 ...................................................... 44

Tabel 4. 11 Hasil Percobaan RNN Skenario 3 ...................................................... 44

Tabel 4. 12 Hasil Percobaan CNN Skenario 1 ...................................................... 45

Tabel 4. 13 Hasil Percobaan CNN Skenario 2 ...................................................... 45

Tabel 4. 14 Hasil Percobaan CNN Skenario 3 ...................................................... 45 Tabel 4. 15 Hasil Percobaan Hybrid LSTM-GRU Skenario 1.............................. 46

Tabel 4. 16 Hasil Percobaan Hybrid LSTM-GRU Skenario 2.............................. 46

Tabel 4. 17 Hasil Percobaan Hybrid LSTM-GRU Skenario 3.............................. 47 Tabel 4. 18 Hasil Terbaik dari Seluruh Model ...................................................... 47

Tabel 4. 19 Hasil Prediksi Pembangkit Tahun 2024 ............................................. 52

# DAFTAR GAMBAR

Gambar 2. 1 Arsitektur RNN ................................................................................ 24 Gambar 2. 2 Arsitektur LSTM .............................................................................. 25 Gambar 2. 3 Arsitektur GRU ................................................................................ 26

Gambar 2. 4 Arsitektur CNN ................................................................................ 26 Gambar 3. 1 Metode *Waterfall* ............................................................................. 29 Gambar 3. 2 Gambaran Umum Sistem ................................................................. 31 Gambar 3. 3 *Use Case Diagram* ........................................................................... 35 Gambar 3. 4 Rincian *Use Case Diagram* .............................................................. 35 Gambar 3. 5 *Activity Diagram* .............................................................................. 36 Gambar 3. 6 *Flowchart* Model Prediksi ................................................................ 37 Gambar 3. 7 *Mockup* Halaman Login ................................................................... 38 Gambar 3. 8 *Mockup* Halaman Dashboard ........................................................... 39 Gambar 4. 1 Grafik Hasil Terbaik Model Hybrid LSTM-GRU ........................... 48 Gambar 4. 2 Grafik Hasil Terbaik Model CNN .................................................... 48

Gambar 4. 3 Grafik Hasil Terbaik Model RNN .................................................... 49 Gambar 4. 4 Grafik Hasil Terbaik Model LSTM ................................................. 49 Gambar 4. 5 Grafik Hasil Terbaik Model GRU .................................................... 50 Gambar 4. 6 Grafik Hasil Prediksi Beban Puncak Pembangkit Listrik ................ 51 Gambar 4. 7 Response API *Framework* Flask ...................................................... 52 Gambar 4. 8 Tampilan Grafik Prediksi Januari .................................................... 53 Gambar 4. 9 Tampilan Grafik Prediksi Februari .................................................. 53

Gambar 4. 10 Tampilan Grafik Prediksi Maret .................................................... 54 Gambar 4. 11 Tampilan Grafik Prediksi April...................................................... 54 Gambar 4. 12 Tampilan Grafik Prediksi Hanya 1 Pembangkit ............................ 55

# BAB I

## PENDAHULUAN

#### 1.1. Latar Belakang

Perkembangan pesat teknologi internet telah mengubah secara signifikan cara konsumen berinteraksi dengan produk, khususnya pada sektor *fashion* yang sangat dinamis. Platform *e-commerce* menawarkan pilihan produk yang hampir tak terbatas, namun hal ini justru menimbulkan tantangan baru yang signifikan: kelebihan informasi (*information overload*). Konsumen sering kali kesulitan menemukan produk yang benar-benar sesuai dengan selera mereka di tengah katalog yang sangat besar (Malisi, 2023). Fenomena ini tidak hanya terjadi pada produk ritel, tetapi juga dalam konteks lain seperti wisata kuliner, di mana banyaknya pilihan justru membuat pengguna bingung dalam mengambil keputusan (Mondi, Wijayanto, & Winarno, 2019). Untuk mengatasi masalah ini dan meningkatkan pengalaman pengguna, sistem rekomendasi menjadi komponen teknologi yang krusial, bukan lagi sekadar fitur tambahan (Syaifuddin & Ningsih, 2023).

Untuk mengatasi tantangan spesifik pada *e-commerce clothing*, pendekatan *Content-Based Filtering* (CBF) menawarkan solusi yang sangat efektif. Metode ini bekerja dengan menganalisis atribut atau "konten" dari produk—seperti kategori, bahan, merek, dan deskripsi—lalu merekomendasikannya kepada pengguna berdasarkan kemiripan dengan produk yang pernah disukai sebelumnya (Salim, Pragantha, & Lauro, n.d.). Keunggulan utama CBF adalah kemampuannya untuk mengatasi masalah *cold start*, di mana produk baru dapat segera direkomendasikan tanpa perlu menunggu data interaksi dari pengguna lain, sebuah masalah yang sering dihadapi sistem rekomendasi (Nurcahya & Supriyanto, 2020). Hal ini sangat vital untuk bisnis *fashion* yang bergantung pada kecepatan peluncuran koleksi baru ke pasar.

Secara teknis, implementasi CBF pada data tekstual seperti deskripsi produk memerlukan mekanisme untuk mengubah teks menjadi representasi numerik yang dapat diukur. Kombinasi algoritma *Term Frequency-Inverse Document Frequency* (TF-IDF) dan *Cosine Similarity* telah terbukti menjadi standar industri yang andal untuk tugas ini. TF-IDF berfungsi untuk mengekstraksi dan membobot kata kunci yang paling signifikan dari deskripsi produk, sementara *Cosine Similarity* mengukur kedekatan atau relevansi antar produk berdasarkan vektor fitur yang dihasilkan (Huda, Fajarudin, & Hadinegoro, 2022). Efektivitas kombinasi ini telah divalidasi dalam berbagai penelitian, mulai dari sistem rekomendasi resep masakan (Sari, Priskila, & Putra, 2024) hingga rekomendasi pekerjaan (Aziz, 2023), yang menunjukkan kemampuannya dalam memberikan hasil yang akurat dan relevan.

Selain keunggulan teknis, pemilihan CBF juga didasari oleh kemampuannya memberikan rekomendasi yang transparan dan dapat dijelaskan (*explainable*) kepada pengguna (Rolanda, Gunawan, & Wanayumini, 2023). Sistem dapat memberikan justifikasi sederhana seperti "Direkomendasikan karena Anda menyukai produk dari merek yang sama," yang dapat meningkatkan kepercayaan dan pengalaman pengguna. Untuk memastikan sistem yang dikembangkan tidak hanya akurat secara algoritmis tetapi juga efektif dari sisi pengguna, penelitian ini mengadopsi metode pengembangan *prototyping*. Metode ini memungkinkan validasi desain dan fungsionalitas secara iteratif melalui umpan balik pengguna, memastikan produk akhir benar-benar menjawab kebutuhan (Warisman, Putra, & Rita, 2021). Pendekatan ini sejalan dengan prinsip *User Centered Design* (UCD), yang memfokuskan proses pengembangan pada kebutuhan dan pengalaman pengguna untuk menciptakan produk yang intuitif dan fungsional (Sanjaya, Sularsih, & Setiani, 2022).

Berdasarkan justifikasi tersebut, penelitian ini bertujuan untuk merancang dan membangun sebuah *website e-commerce clothing* yang fungsional, dilengkapi dengan sistem rekomendasi produk yang mengimplementasikan *content-based filtering* menggunakan algoritma TF-IDF dan *Cosine Similarity*. Pengembangan sistem akan mengikuti metodologi *prototyping* untuk memastikan pendekatan yang berpusat pada pengguna. Arsitektur sistem akan dibangun menggunakan tumpukan teknologi modern yang terpisah (*decoupled*), yaitu React.js dan Vite.js untuk *frontend*, Node.js untuk *backend*, MongoDB sebagai *database*, dan Python Flask sebagai layanan mikro (*microservice*) khusus untuk model *machine learning*.

#### 1.2. Rumusan Masalah

Berdasarkan uraian di atas, penulis telah berhasil mengidentifikasi permasalahan yang muncul. Permasalahan tersebut menjadi titik sentral dalam analisis yang dilakukan, diantaranya yaitu:

1. Bagaimana merancang dan mengimplementasikan sistem rekomendasi produk *clothing* menggunakan metode *content-based filtering* dengan algoritma *TF-IDF* dan *Cosine Similarity*?
2. Bagaimana efektivitas sistem rekomendasi yang dibangun dalam memberikan rekomendasi produk yang relevan berdasarkan kemiripan konten?
3. Bagaimana cara implementasi sistem rekomendasi berbasis *TF-IDF* dan *Cosine Similarity* dalam platform *e-commerce* dengan arsitektur *microservice*?

#### 1.3. Batasan Masalah

Pada pembuatan skripsi, agar pembahasan tetap terfokus, diberikan batasan masalah sebagai berikut:

1. Penelitian berfokus pada perancangan dan pengembangan *website e-commerce clothing* dengan fitur utama sistem rekomendasi produk.
2. Sistem rekomendasi yang dikembangkan menggunakan pendekatan content-based filtering dengan algoritma TF-IDF dan Cosine Similarity.
3. Data produk yang digunakan adalah data clothing/fashion dengan atribut nama, kategori, subcategory sebagai type.
4. Platform dikembangkan menggunakan teknologi React.js dan Vite.js untuk frontend dan admin, Node.js dengan Express.js untuk backend, MongoDB sebagai database, dan Python Flask untuk machine learning service.
5. Sistem hanya memberikan rekomendasi produk berdasarkan similarity score.
6. Aplikasi berfokus pada fitur core e-commerce dan sistem rekomendasi tanpa implementasi advanced features seperti chatbot atau augmented reality.

#### 1.4. Tujuan

Dalam pembuatan skripsi ini, ada beberapa tujuan utama yang ingin dicapai diantaranya yakni:

1. Mengimplementasikan sistem rekomendasi produk berbasis *content-based filtering* dengan algoritma *TF-IDF* dan *Cosine Similarity* pada sebuah *website e-commerce clothing*.
2. Membangun aplikasi *full-stack* dengan arsitektur *microservice* yang mengintegrasikan frontend *React.js, backend Node.js,* dan layanan *machine learning* Python *Flask*.

#### 1.5. Manfaat

Ada beberapa manfaat dari pengerjaan skripsi ini yang dapat berdampak pada sekitar, antara lain:

1. Bagi Penulis: Memberikan pengalaman praktis dalam merancang, mengimplementasikan, dan mengevaluasi sistem *machine learning* dengan arsitektur *microservice* menggunakan tumpukan teknologi modern, serta memperdalam pemahaman teoretis mengenai sistem rekomendasi.
2. Bagi Pelaku Usaha: Menyediakan model aplikasi yang dapat membantu pemilik usaha *clothing* untuk meningkatkan penjualan dan keterlibatan pelanggan (*customer engagement*) melalui sistem rekomendasi yang personal dan relevan.
3. Bagi Konsumen: Memberikan pengalaman berbelanja yang lebih baik dengan menyajikan rekomendasi produk yang relevan sesuai preferensi, sehingga mengurangi waktu pencarian.
4. Bagi Komunitas Pengembang: Menyediakan studi kasus yang dapat digunakan sebagai referensi oleh pengembang lain untuk membangun sistem *e-commerce* dengan fitur rekomendasi berbasis AI menggunakan tumpukan teknologi modern.

#### 1.6. Sistematika Penulisan

Sistematika penulisan dari penyusunan skripsi ini akan dijelaskan dalam setiap bab dengan penjelasan sebagai berikut:

###### BAB I : PENDAHULUAN

Pada bab pendahuluan memberikan gambaran umum mengenai permasalahan yang akan dibahas. Bab ini terdiri dari beberapa sub bab seperti latar belakang, rumusan masalah, batasan masalah, tujuan, dan manfaat dari skripsi yang telah dibuat, serta sistematika penulisan.

###### BAB II : TINJAUAN PUSTAKA

Aktivitas ini melibatkan peninjauan literatur yang telah dipublikasikan oleh akademisi atau peneliti lain. Tujuan utamanya adalah untuk mengkaji kembali hasil-hasil penelitian sebelumnya. Kegiatan ini membantu memahami perkembangan terkini dalam topik yang kita teliti.

###### BAB III : KEGIATAN PELAKSANAAN

Pada bab pelaksanaan kegiatan ini, dijelaskan langkah-langkah metode pelaksanaan serta tahapan perancangan fitur prediksi. Bab ini membahas mengenai teknik implementasi dan pengujian fitur yang dikembangkan. Akhirnya, hasil dari setiap tahapan dijelaskan untuk memberikan gambaran yang menyeluruh.

###### BAB IV : ANALISA DAN PEMBAHASAN

Pada bab analisis dan pembahasan, dijelaskan hasil analisis dari pembuatan fitur serta jawaban terhadap pertanyaan dari rumusan masalah. Bab ini menguraikan temuan yang diperoleh selama proses penelitian. Selain itu, bab ini juga mengaitkan hasil yang ditemukan dengan tujuan penelitian untuk memberikan pemahaman yang menyeluruh.

###### BAB V : KESIMPULAN

Dalam bab kesimpulan ini, terdapat rangkuman dari keseluruhan laporan skripsi serta penarikan kesimpulan yang dibuat berdasarkan hasil penelitian yang telah dilakukan. Bab ini menyajikan temuan utama dan dampaknya, serta menyertakan rekomendasi untuk penelitian lebih lanjut.

###### DAFTAR PUSTAKA

Dalam daftar pustaka, ditampilkan semua sumber yang digunakan sebagai referensi dalam laporan skripsi ini. Sumber-sumber ini berfungsi sebagai penunjang dalam pengerjaan laporan skripsi, memastikan akurasi dan kredibilitas penelitian. Daftar pustaka mencakup buku, artikel jurnal, serta sumber-sumber lain yang relevan dengan topik penelitian.

###### LAMPIRAN

Dalam lampiran terdapat dokumentasi dan dokumen tambahan yang mendukung isi dari laporan skripsi. Lampiran ini mencakup berbagai materi pelengkap seperti kode program dan formulir yang digunakan dalam penelitian. Tujuannya adalah untuk memberikan informasi tambahan yang dapat membantu pembaca memahami lebih dalam mengenai penelitian yang dilakukan.

# BAB II

## TINJAUAN PUSTAKA

Pada bab tinjauan pustaka, diuraikan dasar-dasar teori yang mendukung pengembangan website e-commerce dan sistem rekomendasi, serta ulasan penelitian terdahulu yang relevan dengan skripsi ini. Bab ini juga menyajikan perbandingan antar metode berdasarkan jurnal-jurnal yang telah diulas untuk memposisikan penelitian ini dalam lanskap akademis yang ada. Selain itu, dijelaskan pula perangkat dan teknologi yang digunakan dalam pengembangan sistem.

#### 2.1. Dasar Teori

Pada bagian ini dijelaskan mengenai konsep dan teori fundamental yang diambil dari berbagai literatur ilmiah seperti jurnal, buku, dan konferensi yang menjadi landasan dalam penyusunan skripsi ini.

###### 2.1.1. Sistem Rekomendasi

Sistem rekomendasi adalah perangkat lunak yang bertujuan untuk menyarankan konten atau produk yang relevan kepada pengguna. Dalam konteks *e-commerce*, sistem ini menjadi komponen krusial untuk mengatasi masalah kelebihan informasi (*information overload*) dan meningkatkan pengalaman pengguna dengan menyajikan produk yang kemungkinan besar akan mereka sukai atau beli (Syaifuddin & Ningsih, 2023). Sistem ini memprediksi item yang menarik bagi pengguna berdasarkan data interaksi atau atribut item itu sendiri, sehingga dapat membantu pengguna menemukan produk yang sesuai dengan ketertarikan mereka secara efisien (Badriyah, Fernando, & Syarif, 2018).

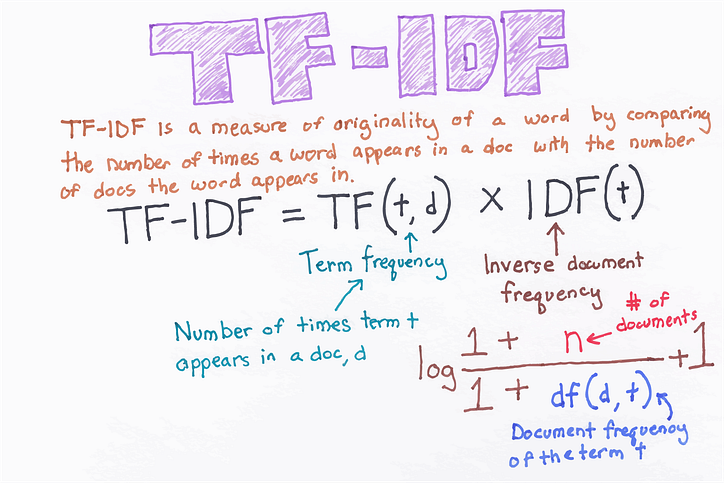
###### 2.1.2. Content-Based Filtering (CBF)

*Content-Based Filtering (CBF*) adalah paradigma sistem rekomendasi yang beroperasi berdasarkan karakteristik intrinsik dari item dan profil minat pengguna. Sistem ini merekomendasikan item dengan mencocokkan fitur-fiturnya (misalnya, deskripsi produk, kategori, merek) dengan fitur dari item yang sebelumnya telah disukai oleh pengguna (Salim, Pragantha, & Lauro, n.d.). Keunggulan utama *CBF* adalah sifatnya yang user *independence*,artinya tidak memerlukan data dari pengguna lain, sehingga mampu mengatasi masalah *cold-start* untuk item baru (Mondi, Wijayanto, & Winarno, 2019). Dalam e-commerce fashion, di mana produk baru terus-menerus diperkenalkan, *CBF* sangat efektif karena sebuah pakaian baru dapat direkomendasikan segera setelah fiturnya ditambahkan ke sistem (Nurcahya & Supriyanto, 2020).

###### 2.1.3. Representasi Fitur Teks: TF-IDF dan *Cosine Similarity*

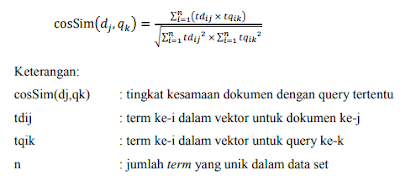
Untuk memungkinkan mesin membandingkan produk berdasarkan deskripsi tekstualnya, teks perlu diubah menjadi representasi numerik. Kombinasi *Term Frequency-Inverse Document Frequency* *(TF-IDF)* dan *Cosine Similarity* adalah metode yang mapan untuk tujuan ini.

1. *Term Frequency-Inverse Document Frequency (TF-IDF):* *TF-IDF* adalah teknik statistik yang digunakan untuk mengukur pentingnya sebuah kata dalam sebuah dokumen relatif terhadap seluruh koleksi dokumen (korpus). *Term Frequency* (TF) mengukur frekuensi kemunculan kata dalam satu dokumen, sementara *Inverse Document Frequency* (IDF) mengukur kelangkaan kata di seluruh korpus. Dengan mengalikan kedua nilai ini, *TF-IDF* memberikan bobot yang merepresentasikan signifikansi sebuah kata, sehingga efektif untuk ekstraksi fitur dari deskripsi produk (Huda, Fajarudin, & Hadinegoro, 2022).



*Gambar 2.1: Rumus Perhitungan TF-IDF*

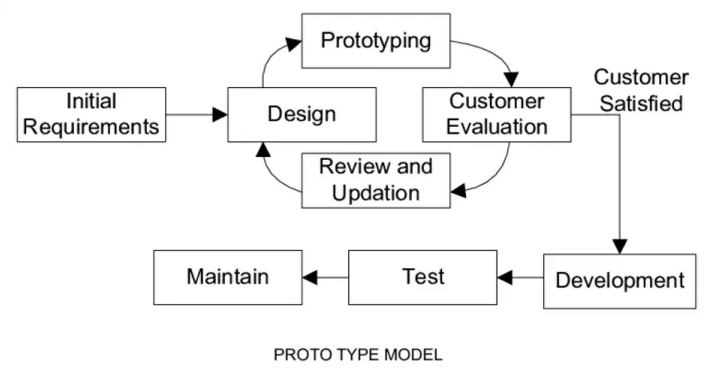
1. *Cosine Similarity:* Setelah deskripsi produk diubah menjadi vektor *TF-IDF*, *Cosine Similarity* digunakan untuk mengukur kemiripan antara dua vektor. Metrik ini menghitung kosinus sudut antara dua vektor, di mana nilai yang mendekati 1 menandakan kemiripan yang tinggi. Metode ini sangat efektif untuk membandingkan deskripsi produk karena tidak terpengaruh oleh panjang dokumen (Aziz, 2023). Efektivitasnya telah terbukti dalam berbagai kasus, seperti rekomendasi resep masakan, yang menunjukkan kemampuannya menghasilkan rekomendasi akurat (Sari, Priskila, & Putra, 2024).



*Gambar 2.2: Rumus Perhitungan Cosine Similarity*

###### 2.1.4. Metodologi Pengembangan Perangkat Lunak: *Prototyping*

Model *Prototyping* adalah metodologi pengembangan perangkat lunak yang bersifat evolusioner dan iteratif. Tujuan utamanya adalah untuk membangun, menguji, dan memvalidasi ide dengan pengguna sebelum pengembangan skala penuh dilakukan, sehingga dapat mengurangi risiko proyek (Warisman, Putra, & Rita, 2021). Proses ini sangat berorientasi pada pengguna, di mana umpan balik dari pengguna digunakan untuk menyempurnakan prototipe secara berulang hingga produk akhir selaras dengan kebutuhan dan ekspektasi mereka (Riyanti et al., 2024). Pendekatan ini sejalan dengan *User Centered Design* (UCD), yang memastikan bahwa desain antarmuka yang dihasilkan berfokus pada kebutuhan dan kemudahan pengguna (Sanjaya, Sularsih, & Setiani, 2022).



Gambar 2.3: Siklus Metode Prototyping

#### 2.2. Penelitian Terkait

Pada bagian penelitian terkait, akan dijabarkan hasil pembacaan referensi dari beberapa jurnal terdahulu yang digunakan sebagai landasan dalam penyusunan laporan skripsi ini. Informasi ini bertujuan untuk mendukung pemahaman dan analisis lebih lanjut terhadap metode yang digunakan dalam penelitian ini. Pada bagian ini, penting untuk memastikan bahwa penelitian berlangsung dengan dasar yang kuat.

Sebuah penelitian berjudul "Sistem Rekomendasi Product Emina Cosmetics Dengan Menggunakan Metode *Content-Based Filtering*" berfokus pada produk kosmetik yang memiliki banyak varian. Dalam studi ini, deskripsi produk digunakan sebagai konten utama untuk dihitung kemiripannya menggunakan *TF-IDF* dan *Cosine Similarity*. Hasilnya menunjukkan bahwa metode ini berhasil memberikan rekomendasi produk dengan nilai similaritas tertinggi mencapai 0,7195, membuktikan relevansinya untuk menangani tantangan pada produk ritel seperti *fashion* (Larasati & Februariyanti, 2021).

Penelitian lain dengan judul "Implementasi *Content-based Filtering* Menggunakan *TF-IDF* and *Cosine Similarity* untuk Sistem Rekomendasi Resep Masakan" menerapkan pendekatan serupa pada domain yang berbeda. Sistem ini dibangun untuk merekomendasikan resep masakan berdasarkan bahan-bahan yang dimiliki pengguna. Hasil pengujiannya menunjukkan bahwa sistem mampu menghasilkan rekomendasi yang cukup akurat, yang semakin menguatkan bahwa kombinasi *TF-IDF* dan *Cosine Similarity* adalah pendekatan yang andal untuk mencocokkan item berdasarkan konten tekstual (Sari, Priskila, & Putra, 2024).

Dalam penelitian berjudul "Penerapan Metode Content-Based Filtering dalam Strategi Komunikasi Pemasaran pada Marketplace Tokopedia", metode ini diterapkan dalam skala yang lebih besar. Dengan menggunakan nama dan deskripsi produk dari dataset yang diambil melalui *web scraping*, sistem rekomendasi dibangun untuk mendukung strategi komunikasi pemasaran. Hasilnya menunjukkan tingkat kepercayaan rekomendasi yang sangat tinggi, yaitu 97,39%, yang membuktikan efektivitas metode ini pada platform *e-commerce* nyata dan perannya dalam meningkatkan *brand awareness* (Syaifuddin & Ningsih, 2023).

Dari sisi pengembangan perangkat lunak, penelitian berjudul "Implementasi Metode Prototipe pada Website E-Commerce Toko Buku Penuntun" menyoroti pentingnya metodologi yang tepat. Studi ini menunjukkan bagaimana pendekatan *prototyping* memungkinkan pengembangan sistem yang cepat dan bertahap. Setiap versi prototipe dievaluasi langsung oleh pengguna, sehingga memastikan produk akhir, mulai dari desain antarmuka hingga fungsionalitasnya, benar-benar sesuai dengan kebutuhan pengguna dan pemilik toko (Warisman, Putra, & Rita, 2021).

Studi lain berjudul "Development of Rental Application using Prototyping Method" juga mengkonfirmasi keunggulan metode ini. *Prototyping* dipilih karena kemampuannya untuk mengevaluasi dan menyempurnakan fitur berdasarkan umpan balik pengguna sejak awal proses desain. Hasil pengujian *blackbox* pada aplikasi yang dikembangkan menunjukkan bahwa semua fitur yang dibuat melalui proses iteratif ini valid dan fungsional, yang sekali lagi membuktikan efektivitas *prototyping* dalam menghasilkan solusi teknologi yang berpusat pada pengguna (Riyanti et al., 2024).

Dengan melihat penelitian-penelitian tersebut mendapati beberapa perbandingan metode beserta dataset yang dijelaskan dalam tabel 2.1 sebagai berikut.

| **No** | **Penelitian** | **Dataset** | **Metode** | **Hasil Artikel** |
| --- | --- | --- | --- | --- |
| 1 | *Sistem Rekomendasi Product Emina Cosmetics Dengan Menggunakan Metode Content-Based Filtering (Larasati & Februariyanti, 2021)* | 87 varian produk kosmetik Emina. | CBF, TF-IDF, Cosine Similarity. | Berhasil menghasilkan 10 produk rekomendasi dengan nilai similaritas tertinggi 0,7195. |
| 2 | Implementasi Content-based Filtering Menggunakan TF-IDF and Cosine Similarity untuk Sistem Rekomendasi Resep Masakan (Sari, Priskila, & Putra, 2024) | 30 data resep masakan dari website. | CBF, TF-IDF, Cosine Similarity. | Sistem menghasilkan rekomendasi yang cukup akurat dengan nilai RMSE mendekati nol (0.356). |
| 3 | Penerapan Metode Content-Based Filtering dalam Strategi Komunikasi Pemasaran pada Marketplace Tokopedia (Syaifuddin & Ningsih, 2023) | 3.233 data produk dari Tokopedia (via *web scraping*). | CBF, TF-IDF, Cosine Similarity, IMC. | Memberikan rekomendasi produk dengan tingkat kepercayaan 97,39% dan mendukung *brand awareness*. |
| 4 | Implementasi Metode Prototipe pada Website E-Commerce Toko Buku Penuntun (Warisman, Putra, & Rita, 2021) | Studi kasus Toko Buku Penuntun. | Prototyping, UML. | Berhasil membangun website e-commerce yang membantu memasarkan produk dan memudahkan transaksi. |
| 5 | Development of Rental Application using Prototyping Method (Riyanti et al., 2024) | Studi kasus aplikasi rental HP dan kamera. | Prototyping. | Semua fitur sistem berfungsi sesuai tujuan dan valid berdasarkan 7 skenario pengujian *blackbox*. |
| 6 | Sistem Rekomendasi Peminjaman Buku Menggunakan Metode Vector Space Model Berbasis Pembobotan TF-IDF dan FastText (Perdana, Hartanti, & Hasanah, 2025) | 2.079 data buku dari perpustakaan sekolah. | CBF, VSM, TF-IDF, FastText, Cosine Similarity. | Sistem menunjukkan kinerja sangat baik dengan rata-rata Precision 10 mencapai 93%. |
| 7 | A Recommendation System for Prewedding Location Selection using Count Vectorization and Cosine Similarity (Kurniadi & Maulana, 2025) | 75 data lokasi prewedding. | CBF, Count Vectorization, Cosine Similarity, Rapid Throwaway Prototyping. | Sistem efektif memberikan rekomendasi relevan dengan nilai *Mean Reciprocal Rank* (MRR) sebesar 0.88. |
| 8 | Development of Mobile Application by Applying Content-Based Filtering (Hermanto et al., 2025) | Studi kasus produk ritel Ampu Mart. | CBF, String Matching, Cosine Similarity. | Aplikasi berhasil meningkatkan pengalaman belanja dan penjualan melalui rekomendasi yang akurat. |
| 9 | Recommendation System with Content-Based Filtering Method for Culinary Tourism in Mangan Application (Mondi, Wijayanto, & Winarno, 2019) | 353 data restoran dari aplikasi MANGAN. | CBF CBF, Cosine Similarity. | Sistem mampu membantu pengguna memilih restoran dengan rata-rata presisi 0,8915, 0,4820 dan MAE sebesar  0,3292. |
| 10 | Perancangan Sistem Rekomendasi Film menggunakan metode Content-based Filtering (Salim, Pragantha, & Lauro, n.d.) | Dataset film dari The Movies Database (TMDb). | CBF, TF-IDF, Cosine Similarity. | Sistem berhasil merekomendasikan film berdasarkan kemiripan sutradara, aktor, dan genre. |
| 11 | Content-based recommender system architecture for similar e-commerce products (Nurcahya & Supriyanto, 2020) | 868 data produk agrikultur dari e-commerce. | CBF, TF-IDF, VSM. | Arsitektur sistem yang dibangun menghasilkan nilai rata-rata Recall 0.84 dan Precision 0.78. |
| 12 | Metode User Centered Design dalam Merancang Tampilan Antarmuka Ecommerce Penjualan Produk Makanan Sweetbites By Caca Berbasis Website Menggunakan Aplikasi Balsamiq Mockups (Sanjaya, Sularsih, & Setiani, 2022) | Studi kasus *home industry* makanan | User Centered Design (UCD). | Berhasil merancang antarmuka e-commerce yang berfokus pada kebutuhan pengguna. |
| 13 | Sistem Rekomendasi Pekerjaan Menggunakan Algoritma Content Based Filtering (TF-IDF dan Cosine Similarity) (Aziz, 2023) | Data lowongan pekerjaan dari BKK SMK. | CBF, TF-IDF, Cosine Similarity. | Sistem rekomendasi berhasil dengan rata-rata *precision* 85,25% dan *recall* 100%. |
| 14 | Content-Based Filtering Recommendation System Using Categories Search Engine (Rolanda, Gunawan, & Wanayumini, 2023) | Dataset MovieLens. | CBF. | Berhasil mengembangkan sistem rekomendasi berdasarkan kategori untuk meningkatkan akurasi pencarian. |
| 15 | Sistem Rekomendasi Content Based Filtering Menggunakan Algoritma Apriori (Badriyah, Fernando, & Syarif, 2018) | Data transaksi e-commerce. | CBF, Algoritma Apriori. | Sistem dapat memberikan rekomendasi produk berdasarkan histori transaksi belanja pembeli. |
| 16 | Sistem Rekomendasi Content-based Filtering Menggunakan TF-IDF Vector Similarity Untuk Rekomendasi Artikel Berita (Huda, Fajarudin, & Hadinegoro, 2022) | 22 artikel berita dari portal universitas. | CBF, TF-IDF, Cosine Similarity. | Sistem mampu memberikan skor Recall@5 sekitar 73% dan Recall@10 sekitar 80%. |
| 17 | Sistem Informasi Penjualan Berbasis Website dengan Pendekatan Metode Prototype (Ichwani, Anwar, Karsono, & Alrifqi, 2021) | Studi kasus penjualan seprai Kinov Seprai. | Prototyping, PIECES. | Sistem informasi berbasis web berhasil menyediakan media promosi yang efektif dan memperluas target pasar. |
| 18 | Rancang Bangun Pada Aplikasi E-Commerce Dengan Fitur Rekomendasi Produk Menggunakan Metode Apriori Dan Content Based Filtering Berbasis Android (Malisi, 2023) | Studi kasus Toko Elis Collection (keperluan rumah tangga). | Apriori, CBF. | Merancang aplikasi e-commerce berbasis Android dengan fitur rekomendasi untuk memperluas area penjualan. |
| 19 | Prototype of Web System for Organizations Dedicated to e-Commerce under the SCRUM Methodology (Gomero-Fanny, Bengy, & Andrade-Arenas, 2021) | Studi kasus organisasi e-commerce. | SCRUM. | Berhasil merancang prototipe sistem web e-commerce yang memenuhi kebutuhan organisasi dengan 4 Sprint. |
| 20 | Prototype of Web System for Organizations Dedicated to e-Commerce under the SCRUM Methodology (Gomero-Fanny, Bengy, & Andrade-Arenas, 2021) | Studi kasus organisasi e-commerce. | Studi kasus organisasi e-commerce. | Berhasil merancang prototipe sistem web e-commerce yang memenuhi kebutuhan organisasi dengan 4 Sprint. |

*Tabel 2. 1 Perbandingan Model Antar Jurnal pendahulu*

#### 2.2. Alat dan Teknologi yang Digunakan

Pada bagian ini akan dibahas mengenai berbagai alat dan teknologi yang digunakan untuk membangun website *e-commerce clothing* beserta sistem rekomendasinya, dengan rincian sebagai berikut:

###### 2.2.1. Alat Pengembangan

Dalam perancangan fitur prediksi pada website *e-commerce clothing* menggunakan beberapa Alat diantaranya yaitu:

a) Code Editor

Dalam penelitian ini, digunakan beberapa editor modern yang mendukung ekosistem *JavaScript* secara luas, seperti *Visual Studio Code (VSCode), IDE dari JetBrains (*contohnya *WebStorm), dan Cursor.*

b) Terminal

Untuk interaksi dengan sistem melalui baris perintah *(command line),* digunakan beberapa terminal seperti *GitBash, PowerShell, dan Warp.*

###### 2.2.2. Teknologi yang Digunakan

Teknologi dalam pengembangan website *e-commerce clothing* menggunakan *framework, database dan third party* diantaranya yaitu:

1. Frontend

*ReactJS & Vite*: Untuk membangun antarmuka pengguna (*User Interface*) yang modern dan interaktif, penelitian ini menggunakan *ReactJS*, sebuah pustaka *JavaScript* berbasis komponen. Proses pengembangan dioptimalkan menggunakan *Vite* sebagai *build tool* yang menawarkan pengalaman pengembangan yang sangat cepat.

1. Backend

*Node.js & Express.js*: Logika sisi server, manajemen data, dan API dibangun menggunakan *Node.js* sebagai *runtime environment* dan *Express.js* sebagai kerangka kerja web yang minimalis dan fleksibel.

1. Database

*MongoDB*: Untuk menyimpan dan mengelola data aplikasi seperti informasi produk dan pengguna, penelitian ini menggunakan *MongoDB*, sebuah sistem basis data *NoSQL* yang berorientasi pada dokumen dengan skema yang fleksibel.

1. Layanan Pihak Ketiga

*Cloudinary*: Untuk menangani manajemen aset media seperti gambar produk, penelitian ini memanfaatkan *Cloudinary*, sebuah platform berbasis *cloud* untuk mengunggah, menyimpan, mengoptimalkan, dan mengirimkan gambar secara efisien.

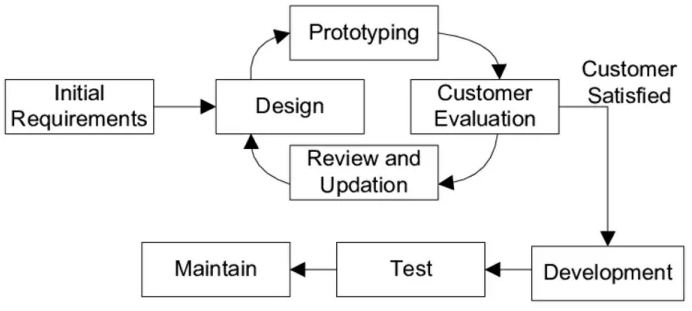
# BAB III

## KEGIATAN PELAKSANAAN

Dalam pembuatan laporan skripsi membutuhkan beberapa tahapan yang dibutuhkan seperti metode pelaksanaan, gambaran umum sistem, analisa kebutuhan, perancangan sistem, dan perancangan pengujian sistem yang akan dibahas sebagai berikut.

#### 3.1. Metode Pelaksanaan

Pengembangan sistem dalam penelitian ini mengadopsi Metode *Prototyping*. Metode ini dipilih karena sifatnya yang iteratif dan berpusat pada pengguna, sangat sesuai untuk proyek yang memerlukan validasi desain dan fungsionalitas secara berkala. Pendekatan ini memungkinkan pengembang dan pengguna untuk berkolaborasi secara erat, memastikan bahwa produk akhir benar-benar memenuhi kebutuhan yang telah ditetapkan.



*Gambar 3. 1 Metode Prototyping*

###### a) Initial Requirements (Pengumpulan Kebutuhan)

Tahap awal melibatkan diskusi mendalam dengan *stakeholder* untuk mengidentifikasi tujuan utama, ruang lingkup, dan kebutuhan dasar dari website e-commerce dan sistem rekomendasinya.

###### b) Design (Perancangan Cepat)

Berdasarkan kebutuhan awal, dibuatlah perancangan dasar yang mencakup sketsa alur pengguna dan desain antarmuka *(mockup)* tingkat rendah. Fokus pada tahap ini adalah memvisualisasikan ide utama dari sistem.

###### c) Prototyping (Pembangunan Prototipe)

Desain yang telah dibuat kemudian diimplementasikan menjadi sebuah prototipe interaktif. Prototipe ini mungkin belum memiliki semua fungsionalitas, namun sudah dapat memberikan gambaran nyata tentang bagaimana sistem akan bekerja.

d) Customer Evaluation (Evaluasi dan Umpan Balik Pengguna)

Prototipe diserahkan kepada pengguna (calon konsumen dan admin) untuk diuji coba. Umpan balik mengenai kemudahan penggunaan, alur, dan fungsionalitas dikumpulkan untuk iterasi selanjutnya*.*

e) Review and Updation (Perbaikan dan Iterasi)

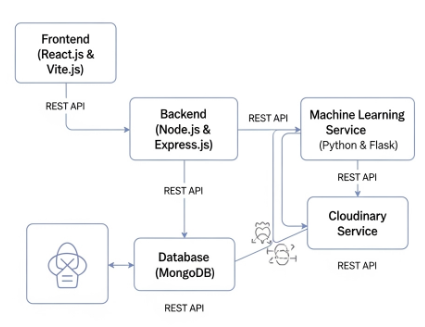
Berdasarkan umpan balik, prototipe diperbaiki dan disempurnakan. Siklus antara *Design*, *Prototyping*, dan *Customer Evaluation* ini diulang beberapa kali hingga prototipe dianggap matang dan telah memenuhi semua kebutuhan yang ditetapkan (*Customer Satisfied*).

f) Development, Test, Maintain (Implementasi, Pengujian, dan Pemeliharaan)

Setelah *prototipe* final disetujui, pengembangan skala penuh (*Development*) dilakukan dengan mengimplementasikan logika *backend*, koneksi *database*, dan algoritma *machine learning*. Sistem yang sudah jadi kemudian diuji secara menyeluruh (*Test*) untuk memastikan tidak ada *bug* dan semua fungsi berjalan sesuai harapan, sebelum akhirnya masuk ke tahap pemeliharaan (*Maintain*).

#### 3.2. Gambaran Umum Sistem

Sistem yang dikembangkan merupakan aplikasi web *full-stack* dengan arsitektur *microservice*. Arsitektur ini memisahkan antara aplikasi *e-commerce* utama dengan layanan sistem rekomendasi, sehingga pengembangan dan pemeliharaan dapat dilakukan secara independen dan lebih fleksibel. Diagram arsitektur sistem secara umum ditunjukkan pada Gambar 3.2.



*Gambar 3.2: Gambaran Umum Sistem.*

Alur kerja dan komponen dalam arsitektur tersebut dapat dijelaskan sebagai berikut:

1. Frontend (React.js & Vite.js)

Merupakan antarmuka pengguna yang diakses oleh admin dan konsumen. Bagian ini bertanggung jawab untuk menampilkan data dan mengirimkan permintaan pengguna ke *backend* melalui REST API.

1. Backend (Node.js & Express.js)

Berperan sebagai otak utama aplikasi yang menangani semua logika bisnis, seperti manajemen produk, autentikasi pengguna, dan pemrosesan transaksi. *Backend* berkomunikasi dengan semua komponen lain dalam sistem.

1. Database (MongoDB)

Menyimpan semua data persisten aplikasi, termasuk data pengguna, katalog produk, dan riwayat pesanan. *Backend* melakukan operasi *Create, Read, Update, Delete* (CRUD) ke *database* ini.

1. Machine Learning Service (Python & Flask)

Sebuah layanan mikro terpisah yang didedikasikan untuk fungsi rekomendasi. *Backend* akan melakukan sinkronisasi data produk ke layanan ini. Ketika ada permintaan rekomendasi, *backend* akan memintanya dari layanan ini melalui *REST API*.

1. Cloudinary Service

Layanan pihak ketiga yang digunakan khusus untuk menyimpan, mengelola, dan menyajikan aset gambar produk. *Backend* akan mengunggah gambar ke *Cloudinary*, lalu menyimpan *URL-nya* di *MongoDB* untuk kemudian dikirim ke *frontend*.

#### 3.3. Analisis Kebutuhan

Analisis kebutuhan mencakup analisis kebutuhan fungsional dan nonfungsional, yang merupakan langkah penting dalam pengembangan sistem. Proses ini melibatkan identifikasi dan penentuan persyaratan yang harus dipenuhi oleh sistem, baik dalam hal fungsi yang harus dilakukan (fungsional) maupun kualitas atau karakteristik tertentu yang harus dimiliki (nonfungsional). Kedua jenis kebutuhan ini kemudian digunakan sebagai dasar untuk merancang dan mengembangkan solusi yang sesuai dengan tujuan dan kebutuhan pengguna dan admin.

###### 3.3.1. Analisis Kebutuhan Fungsional

Analisis kebutuhan dibagi menjadi dua kategori utama: kebutuhan fungsional yang mendefinisikan fitur-fitur sistem, dan kebutuhan non-fungsional yang mendefinisikan kualitas dan batasan sistem. Dalam konteks ini, perlu dipertimbangkan bahwa sistem harus memiliki kemampuan untuk:

1. Kebutuhan Pengguna (Konsumen):
2. Pengguna dapat melakukan registrasi dan login ke dalam sistem.
3. Pengguna dapat melihat daftar semua produk beserta detailnya (nama, harga, deskripsi, gambar).
4. Pengguna dapat melakukan pencarian produk berdasarkan nama.
5. Pengguna dapat melihat daftar produk rekomendasi di halaman detail produk.
6. Pengguna dapat menambahkan produk ke keranjang belanja dan mengelola isinya.
7. Pengguna dapat melakukan proses *checkout* dan simulasi transaksi.
8. Kebutuhan Admin
9. Admin dapat login ke dasbor khusus admin.
10. Admin dapat mengelola data produk (Menambah, Melihat, Mengubah, Menghapus - CRUD).
11. Admin dapat melihat daftar pengguna yang terdaftar.
12. Admin dapat melihat riwayat transaksi yang masuk.

###### 3.3.2. Analisis Kebutuhan Non Fungsional

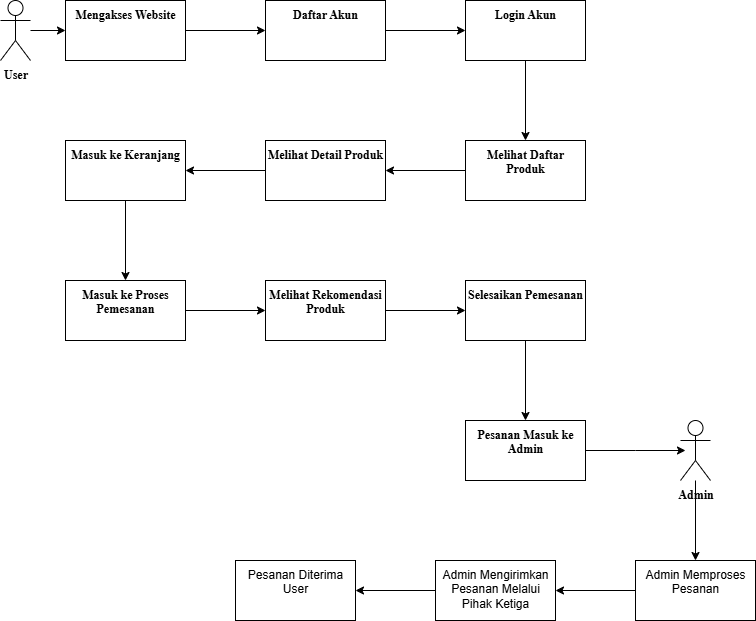
1. Kinerja**:** Waktu muat halaman tidak boleh lebih dari 3 detik. Sistem rekomendasi harus memberikan respons dalam waktu kurang dari 1 detik.
2. Keamanan**:** Kata sandi pengguna harus di-*hash* sebelum disimpan di *database*. Komunikasi antara klien dan server harus aman.
3. Usabilitas**:** Antarmuka harus intuitif, mudah dipahami, dan responsif di berbagai perangkat (desktop dan mobile).
4. Skalabilitas**:** Arsitektur *microservice* harus memungkinkan penambahan fitur atau peningkatan kapasitas di masa depan tanpa mengganggu sistem utama.

#### 3.4. Perancangan Sistem

Pada tahap ini, hasil analisis kebutuhan diterjemahkan ke dalam bentuk perancangan visual dan struktural menggunakan diagram UML (*Unified Modeling Language*), *flowchart* dan *mockup* antarmuka.

###### 3.4.1. UML

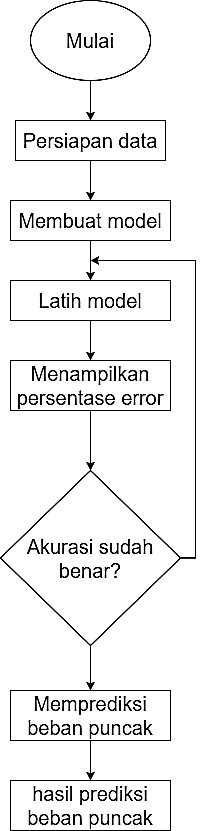
Pada laporan skripsi ini UML fitur prediksi yang digunakan sebagai salah satu rancangan sistem menggunakan *use case diagram* dan *activity diagram.*



*Gambar 3. 3 Use Case Diagram*

###### *3.4.2. Activity Diagram*

*Activity diagram* digunakan untuk memodelkan alur kerja (*workflow*) dari sebuah proses dalam sistem. Diagram ini akan merinci langkah-langkah yang terjadi pada proses-proses penting.



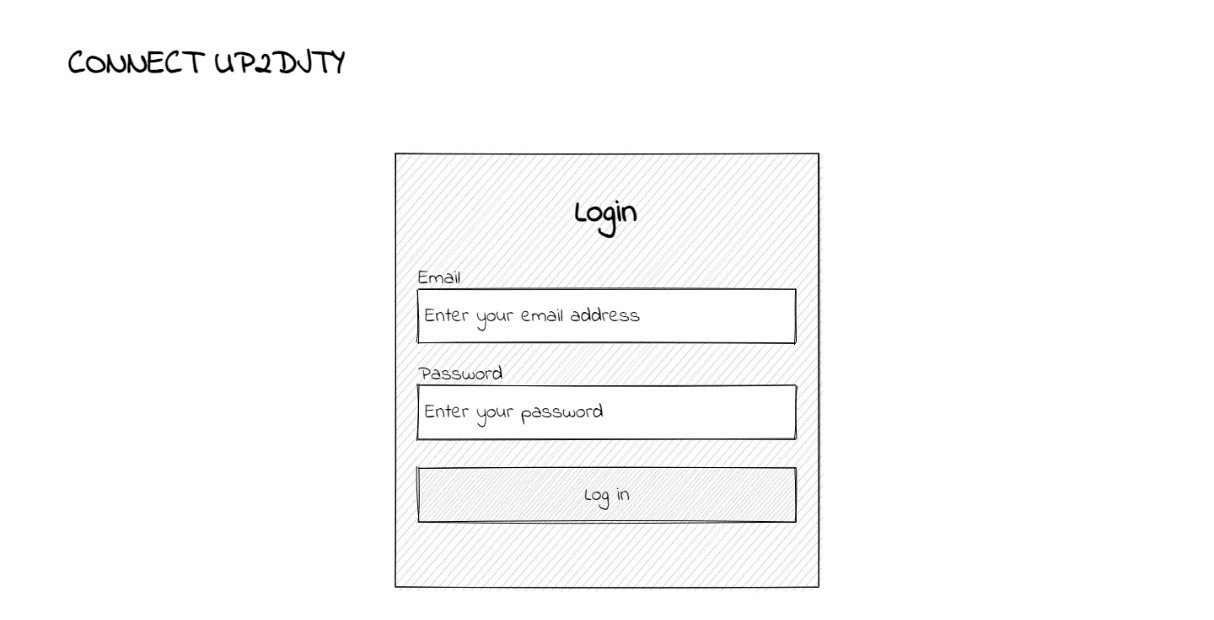
*Gambar 3. 6 Flowchart Model Prediksi*

Tahap pertama adalah melakukan persiapan data dengan membaca data, memproses kolom, dan mengatur indeks. Selanjutnya, data ditransformasi dengan menetapkan panjang dan urutan jumlah fitur. Setelah itu, membuat urutan dan label dari data yang sudah dinormalisasi, kemudian mengonversi data ke format yang sesuai. Setelah itu membuat model dengan menggunakan arsitektur yang sesuai untuk masing masing model. Selanjutnya masuk ke pelatihan model jika akurasi dirasa sudah cukup maka masuk ke bagian prediksi beban puncak setelah itu menampilkan hasil prediksi beban puncak pembangkit listrik.

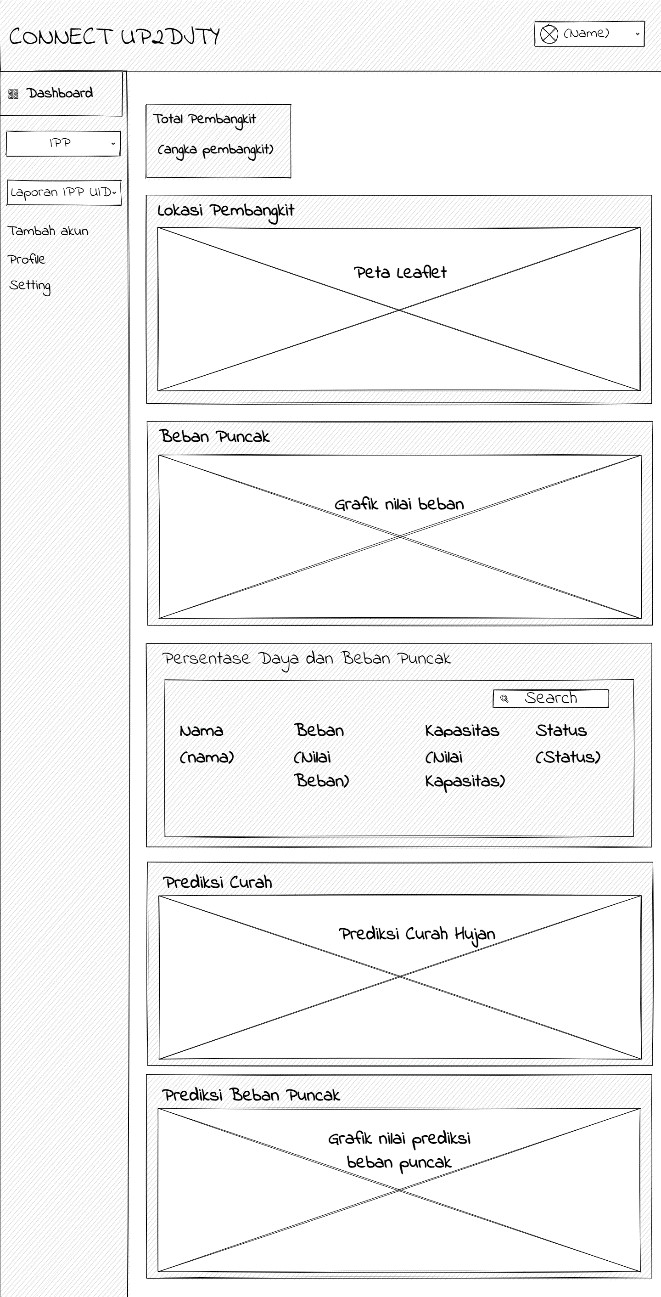
###### 3.4.3. Rancangan Antar Muka dan Interaksi Aplikasi

Tujuan penggunaan website ini adalah untuk memprediksi beban puncak pembangkit listrik. Fitur ini menggunakan API flask untuk menampilkan data hasil prediksi beban puncak pembangkit listrik pada *interface*. *Input* yang diberikan adalah data dari pembangkit listrik ke dalam algoritma, sehingga memiliki *output* hasil prediksi selama 4 bulan. Dengan demikian, pengguna dapat melihat pergerakan dari nilai beban puncak pembangkit listrik melalui grafik. Dalam interaksinya seperti yang digambarkan dalam UML gambar 3.2 hingga gambar 3.4

Dalam laporan skripsi ini, *mockup* digunakan sebagai alat visual yang efektif untuk menjelaskan rancangan antarmuka dan interaksi sistem dengan lebih jelas. *Mockup* ini membantu mengubah konsep dari pemikiran menjadi gambar yang dapat divisualisasikan. Dengan demikian, *mockup* memudahkan pembaca untuk memahami tampilan dan struktur web. Selain itu, *mockup* juga berfungsi untuk menjelaskan komponen-komponen yang ada dalam web. Adapun *mockup* dalam skripsi ini digambarkan sebagai berikut:



*Gambar 3. 7 Mockup Halaman Login*



*Gambar 3. 8 Mockup Halaman Dashboard*

# BAB IV

## ANALISIS DAN PEMBAHASAN

Pada bab ini akan menjelaskan mengenai analisis dan pembahasan sistem dari hasil percobaan dengan rangakian yang telah dibuat sebelumnya. Analisa algoritma dan pengimplementasian model ke dalam web merupakan dua hal yang akan dibahas dan dianalisis pada bab ini.

#### 4.1. Analisis Algoritma

Analisis algoritma akan menjelaskan hasil dari percobaan algoritma yang telah dilakukan. Dalam analisis ini, akan dijelaskan mulai dari data yang digunakan, arsitektur algoritma yang diterapkan, hingga nilai evaluasi yang diperoleh. Tujuan dari analisis ini adalah untuk memberikan gambaran yang jelas tentang kinerja algoritma dan bagaimana setiap komponen berkontribusi terhadap hasil akhir.

###### 4.1.1. Data

Dalam skripsi ini, digunakan data simulasi sumber terbuka dari situs web RTE (https://www.services-rte.com/), yaitu data beban pembangkit listrik per jam. Dari semua data ini, beban puncak bulanan diambil dari salah satu unit pembangkit yang berkaitan dengan tenaga air. Data tersebut terdiri dari 145 beban puncak bulanan yang dikumpulkan dari Desember 2011 hingga Desember 2023, dengan beberapa data seperti yang ditampilkan dalam tabel 4.1.

|  |  |
| --- | --- |
| Tanggal | Beban |
| 1-Dec-11 | 144 |
| 1-Jan-12 | 142 |
| 1-Feb-12 | 135 |
| 1-Mar-12 | 133 |
| 1-Apr-12 | 129 |
| 1-May-12 | 132 |
| 1-Jun-12 | 130 |

*Tabel 4. 1 Contoh Dataset Simulasi Beban Puncak*

Untuk prediksi kedepannya menggunakan data dari 13 pembangkit listrik yang diberikan oleh perusahaan, masing-masing pembangkit memiliki 5 data dari Agustus hingga Desember 2023. Dalam proses pengujian dan pelatihan, terdapat tiga skenario ukuran data pengujian dan pelatihan yang ditampilkan pada tabel 4.2. Skenario pertama adalah pelatihan sebesar 70% dan pengujian sebesar 30%, skenario kedua adalah pelatihan sebesar 80% dan pengujian sebesar 20%, serta skenario ketiga adalah pelatihan sebesar 90% dan pengujian sebesar 10%.

|  |  |  |
| --- | --- | --- |
| **Skenario** | **Ukuran Data** | |
| **Pelatihan** | **Pengujian** |
| 1 | 70% | 30% |
| 2 | 80% | 20% |
| 3 | 90% | 10% |

*Tabel 4. 2 Skenario Ukuran Data*

Pada tahap *preprocessing* data, beberapa langkah dilakukan. Pertama, data dibaca dan diproses kolom tanggal serta diatur indeksnya. Selanjutnya, data dinormalisasi menggunakan *MinMax Scaler*. Setelah itu, data ditransformasi sesuai kebutuhan. Terakhir, data dikonversi ke format yang sesuai untuk pemrosesan lebih lanjut.

###### 4.1.2. Evaluasi Model

Dalam penelitian ini, algoritma GRU digunakan dan dibandingkan dengan model LSTM, RNN, CNN, dan *hybrid* LSTM-GRU. Percobaan dilakukan untuk setiap model dengan berbagai skenario pembagian data, yaitu 70% untuk pelatihan dan 30% untuk pengujian, 80% untuk pelatihan dan 20% untuk pengujian, serta 90% untuk pelatihan dan 10% untuk pengujian. Hasil percobaan menunjukkan variasi performa yang signifikan antara model-model tersebut, yang dirangkum dalam tabel di bawah ini.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| No | Unit | Hyperparameter | | | MAE | MSE | RMSE | Avg  RMSE  Kfold  10 |
| Epochs | Batch\_size | Validation\_split |
| 1 | 64 | 26/300 | 29/300 | 0,4 | 15,37% | 6,32% | 25,14% | 0,120 |
| 2 | 128 | 19/300 | 18/300 | 0,4 | 15,27% | 6,83% | 26,13% | 0,123 |
| 3 | 64 | 24/300 | 26/300 | 0,4 | 15,05% | 6,18% | 24,87% | 0,120 |
| 4 | 64 | 32/300 | 20/300 | 0,3 | 14,50% | 6,64% | 25,77% |
| 5 | 128 | 17/300 | 19/300 | 0,4 | 15,53% | 6,76% | 26,00% | 0,120 |

*Tabel 4. 3 Hasil Percobaan GRU Skenario 1*

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| No | Unit | Hyperparameter | | | MAE | MSE | RMSE | Avg  RMSE  Kfold  10 |
| Epochs | Batch\_size | Validation\_split |
| 1 | 64 | 22/300 | 90 | 0,4 | 19,28% | 9,82% | 31,34% | 0,113 |
| 2 | 128 | 16/300 | 90 | 0,4 | 19,34% | 10,26% | 32,03% | 0,121 |
| 3 | 64 | 39/300 | 70 | 0,4 | 18,81% | 9,24% | 30,40% | 0,114 |
| 4 | 64 | 29/300 | 70 | 0,3 | 20,04% | 9,92% | 31,50% |
| 5 | 128 | 17/300 | 70 | 0,4 | 20,12% | 10,13% | 31,83% | 0,113 |

*Tabel 4. 4 Hasil Percobaan GRU Skenario 2*

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| No | Unit | Hyperparameter | | | MAE | MSE | RMSE | Avg  RMSE  Kfold  10 |
| Epochs | Batch\_size | Validation\_split |
| 1 | 64 | 30/300 | 90 | 0,4 | 26,50% | 13,12% | 36,22% | 0,126 |
| 2 | 128 | 20/300 | 90 | 0,4 | 27,79% | 14,63% | 38,24% | 0,136 |
| 3 | 64 | 15/300 | 70 | 0,4 | 25,43% | 12,73% | 35,69% | 0,127 |
| 4 | 64 | 14/300 | 70 | 0,3 | 26,82% | 13,70% | 37,01% |
| 5 | 128 | 12/300 | 70 | 0,4 | 27,17% | 14,32% | 37,84% | 0,126 |

*Tabel 4. 5 Hasil Percobaan GRU Skenario 3*

Tabel 4.3 menjelaskan performa algoritma GRU pada pembagian data 70% untuk pelatihan dan 30% untuk pengujian. Hasil terbaik diperoleh pada percobaan ketiga dengan nilai MAE sebesar 15,05%, MSE sebesar 6,18%, dan RMSE sebesar 24,87%. Tabel 4.4 menjelaskan hasil algoritma GRU dengan ukuran data pelatihan sebesar 80% dan data pengujian sebesar 20%. Pada percobaan ini, hasil *error* terendah dicapai pada percobaan ketiga, dengan nilai MAE sebesar 18,81%, MSE sebesar 9,24%, dan RMSE sebesar 30,40%. Pada tabel 4.5 memberikan hasil performa algoritma GRU dengan pembagian data 90% *training* dan 10% *testing*. 5 percobaan telah dilakukan dan hasil yang paling terbaik adalah pada percobaan ketiga dengan MSE sebesar 25,43%, MSE sebesar 12,73%, dan RMSE sebesar 35,69%.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| No | Unit | Hyperparameter | | | MAE | MSE | RMSE | Avg  RMSE  Kfold  10 |
| Epochs | Batch\_size | Validation\_split |
| 1 | 64 | 41/300 | 90 | 0,4 | 16,19% | 7,02% | 26,49% | 0,123 |
| 2 | 128 | 25/300 | 90 | 0,4 | 16,63% | 7,29% | 26,99% | 0,128 |
| 3 | 64 | 37/300 | 70 | 0,4 | 16,89% | 7,09% | 26,63% | 0,123 |
| 4 | 64 | 41/300 | 70 | 0,3 | 15,02% | 7,05% | 26,56% |
| 5 | 128 | 29/300 | 70 | 0,4 | 16,37% | 7,44% | 27,29% | 0,123 |

*Tabel 4. 6 Hasil Percobaan LSTM Skenario 1*

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| No | Unit | Hyperparameter | | | MAE | MSE | RMSE | Avg  RMSE  Kfold 10 |
| Epochs | Batch\_size | Validation\_split |
| 1 | 64 | 48/300 | 90 | 0,4 | 20,68% | 10,60% | 32,56% | 0,117 |
| 2 | 128 | 27/300 | 90 | 0,4 | 20,50% | 11,22% | 33,50% | 0,127 |
| 3 | 64 | 36/300 | 70 | 0,4 | 20,51% | 10,69% | 32,69% | 0,116 |
| 4 | 64 | 21/300 | 70 | 0,3 | 21,22% | 10,49% | 32,39% |
| 5 | 128 | 29/300 | 70 | 0,4 | 20,32% | 11,04% | 33,23% | 0,118 |

*Tabel 4. 7 Hasil Percobaan LSTM Skenario 2*

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| No | Unit | Hyperparameter | | | MAE | MSE | RMSE | Avg  RMSE  Kfold 10 |
| Epochs | Batch\_size | Validation\_split |
| 1 | 64 | 37/300 | 90 | 0,4 | 28,64% | 14,84% | 38,52% | 0,130 |
| 2 | 128 | 29/300 | 90 | 0,4 | 29,69% | 15,88% | 39,85% | 0,137 |
| 3 | 64 | 26/300 | 70 | 0,4 | 29,10% | 15,34% | 39,17% | 0,130 |
| 4 | 64 | 24/300 | 70 | 0,3 | 28,85% | 14,66% | 38,28% |
| 5 | 128 | 16/300 | 70 | 0,4 | 29,55% | 16,15% | 40,18% | 0,130 |

*Tabel 4. 8 Hasil Percobaan LSTM Skenario 3*

Berdasarkan hasil percobaan algoritma LSTM dengan tiga skenario yang disajikan dalam tabel 4.6 hingga tabel 4.8, didapatkan beberapa temuan penting. Pada skenario pertama, dengan pembagian data 70% untuk pelatihan dan 30% untuk pengujian, percobaan pertama memberikan hasil terbaik dengan nilai MAE sebesar 16,19%, MSE sebesar 7,02%, dan RMSE sebesar 26,49%. Pada skenario kedua, dengan pembagian data 80% untuk pelatihan dan 20% untuk pengujian, hasil terbaik juga dicapai pada percobaan keempat dengan nilai MAE sebesar 21,22%, MSE sebesar 10,49%, dan RMSE sebesar 32,39%. Terakhir, pada skenario ketiga dengan pembagian data 90% untuk pelatihan dan 10% untuk pengujian, percobaan keempat memberikan hasil terbaik dengan nilai MAE sebesar 28,84%, MSE sebesar 14,66%%, dan RMSE sebesar 38,28%.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| No | Unit | Hyperparaeter | | | MAE | MSE | RMSE | Avg  RMSE |
|  |  |  |
|  |  | Epochs | Batch\_size | Validation\_split |  |  |  | Kfold 10 |
| 1 | 64 | 11/300 | 90 | 0,4 | 19,83% | 12,66% | 35,58% | 0,144 |
| 2 | 128 | 18/300 | 90 | 0,4 | 20,15% | 12,12% | 34,81% | 0,172 |
| 3 | 64 | 15/300 | 70 | 0,4 | 24,97% | 14,46% | 38,02% | 0,160 |
| 4 | 64 | 14/300 | 70 | 0,3 | 22,51% | 16,27% | 40,34% |
| 5 | 128 | 7/300 | 70 | 0,4 | 17,62% | 10,93% | 33,06% | 0,144 |

*Tabel 4. 9 Hasil Percobaan RNN Skenario 1*

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| No | Unit | Hyperparameter | | | MAE | MSE | RMSE | Avg  RMSE |
|  |  |  |
|  |  | Epochs | Batch\_size | Validation\_split |  |  |  | Kfold 10 |
| 1 | 64 | 12/300 | 90 | 0,4 | 27,75% | 20,05% | 44,77% | 0,140 |
| 2 | 128 | 8/300 | 90 | 0,4 | 27,07% | 16,13% | 40,16% | 0,145 |
| 3 | 64 | 12/300 | 70 | 0,4 | 20,21% | 10,95% | 33,09% | 0,140 |
| 4 | 64 | 12/300 | 70 | 0,3 | 19,03% | 9,45% | 30,74% |
| 5 | 128 | 8/300 | 70 | 0,4 | 27,48% | 13,40% | 36,61% | 0,143 |

*Tabel 4. 10 Hasil Percobaan RNN Skenario 2*

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| No | Unit | Hyperparameter | | | MAE | MSE | RMSE | Avg  RMSE |
|  |  |  |
|  |  | Epochs | Batch\_size | Validation\_split |  |  |  | Kfold 10 |
| 1 | 64 | 12/300 | 90 | 0,4 | 30,42% | 17,47% | 41,79% | 0,161 |
| 2 | 128 | 8/300 | 90 | 0,4 | 35,32% | 20,98% | 45,80% | 0,154 |
| 3 | 64 | 12/300 | 70 | 0,4 | 37,61% | 23,82% | 48,80% | 0,153 |
| 4 | 64 | 32/300 | 70 | 0,3 | 30,03% | 16,65% | 40,81% |
| 5 | 128 | 10/300 | 70 | 0,4 | 32,31% | 19,46% | 44,11% | 0,162 |

*Tabel 4. 11 Hasil Percobaan RNN Skenario 3*

Percobaan menggunakan metode RNN dilakukan dengan tiga skenario data yang ditampilkan pada tabel 4.9 hingga tabel 4.11. Pada skenario pertama, dengan pembagian data 70% untuk pelatihan dan 30% untuk pengujian, hasil terbaik diperoleh pada percobaan kelima dengan nilai MAE sebesar 17,62%, MSE sebesar 10,93%, dan RMSE sebesar 33,06%. Pada skenario kedua, dengan pembagian data 80% untuk pelatihan dan 20% untuk pengujian, hasil terbaik dicapai pada percobaan keempat dengan nilai MAE sebesar 19,03%, MSE sebesar 9,45%, dan RMSE sebesar 30,74%. Pada skenario ketiga, dengan pembagian data 90% untuk pelatihan dan 10% untuk pengujian, hasil terbaik diperoleh pada percobaan kelima dengan nilai MAE sebesar 30,03%, MSE sebesar 16,65%, dan RMSE sebesar 40,81%.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| No | Filter | Hyperparameter | | | MAE | MSE | RMSE | Avg  RMSE  Kfold  10 |
| Epochs | Batch\_size | Validation\_split |
| 1 | 64 | 38/300 | 90 | 0,4 | 15,93% | 6,85% | 26,17% | 0,124 |
| 2 | 128 | 22/300 | 90 | 0,4 | 15,71% | 6,75% | 25,98% | 0,124 |
| 3 | 64 | 43/300 | 70 | 0,4 | 15,75% | 6,68% | 25,85% | 0,129 |
| 4 | 64 | 40/300 | 70 | 0,3 | 15,51% | 7,65% | 27,66% |
| 5 | 128 | 30/300 | 70 | 0,4 | 15,47% | 6,60% | 25,70% | 0,123 |

*Tabel 4. 12 Hasil Percobaan CNN Skenario 1*

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| No | Filter | Hyperparameter | | | MAE | MSE | RMSE | Avg  RMSE  Kfold  10 |
| Epochs | Batch\_size | Validation\_split |
| 1 | 64 | 27/300 | 90 | 0,4 | 18,15% | 8,83% | 29,72% | 0,116 |
| 2 | 128 | 27/300 | 90 | 0,4 | 19,91% | 10,31% | 32,11% | 0,128 |
| 3 | 64 | 44/300 | 70 | 0,4 | 19,11% | 9,61% | 31,01% | 0,116 |
| 4 | 64 | 23/300 | 70 | 0,3 | 18,90% | 9,07% | 30,11% |
| 5 | 128 | 28/300 | 70 | 0,4 | 19,79% | 10,08% | 31,75% | 0,119 |

*Tabel 4. 13 Hasil Percobaan CNN Skenario 2*

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| No | Filter | Hyperparameter | | | MAE | MSE | RMSE | Avg  RMSE  Kfold  10 |
| Epochs | Batch\_size | Validation\_split |
| 1 | 64 | 32/300 | 90 | 0,4 | 26,87% | 14,12% | 37,57% | 0,130 |
| 2 | 128 | 30/300 | 90 | 0,4 | 24,32% | 12,15% | 34,86% | 0,128 |
| 3 | 64 | 30/300 | 70 | 0,4 | 25,50% | 12,60% | 35,50% | 0,128 |
| 4 | 64 | 19/300 | 70 | 0,3 | 31,68% | 17,94% | 42,36% |
| 5 | 128 | 17/300 | 70 | 0,4 | 27,24% | 14,12% | 37,58% | 0,130 |

*Tabel 4. 14 Hasil Percobaan CNN Skenario 3*

Dari percobaan CNN yang telah dilakukan dengan menggunakan 3

skenario, pada skenario pertama dengan pembagian data *size* yakni 70% data *train* dan 30% data *test* mendapatkan hasil terbaik pada percobaan kelima dengan RMSE 25,70%. Pada skenario kedua dengan pembagian data *size* yakni 80% data *train* dan 20% data *test* terdapat pada percobaan pertama dengan nilai RMSE sebesar 29,72%. Terakhir pada skenario ketiga dengan pembagian data *size* yakni 90% data *train* dan 10% data *test* mendapatkan hasil terbaik pada percobaan kedua dengan RMSE sebesar 34,86%. Dari percobaan CNN tersebut didapatkan hasil terbaik yakni pada skenario pertama dengan percobaan kelima yakni 25,70%.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| No | Unit | Hyperparameter | | | MAE | MSE | RMSE | Avg  RMSE  Kfold  10 |
| Epochs | Batch\_size | Validation\_split |
| 1 | 64 | 22/300 | 90 | 0,4 | 15,74% | 6,65% | 25.79% | 0,122 |
| 2 | 128 | 14/300 | 90 | 0,4 | 17,47% | 7,08% | 26,61% | 0,126 |
| 3 | 64 | 21/300 | 70 | 0,4 | 15,31% | 6,78% | 26,03% | 0,127 |
| 4 | 64 | 22/300 | 70 | 0,3 | 1423% | 7,00% | 26,45% |
| 5 | 128 | 14/300 | 70 | 0,4 | 16,16% | 7,05% | 26,55% | 0,122 |

*Tabel 4. 15 Hasil Percobaan Hybrid LSTM-GRU Skenario 1*

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| No | Unit | Hyperparameter | | | MAE | MSE | RMSE | Avg  RMSE  Kfold  10 |
| Epochs | Batch\_size | Validation\_split |
| 1 | 64 | 18/300 | 90 | 0,4 | 20,68% | 10,14% | 31,85% | 0,121 |
| 2 | 128 | 15/300 | 90 | 0,4 | 20,18% | 10,48% | 32,37% | 0,120 |
| 3 | 64 | 21/300 | 70 | 0,4 | 19,29% | 10,22% | 31,97% | 0,120 |
| 4 | 64 | 13/300 | 70 | 0,3 | 22,02% | 9,83% | 31,35% |
| 5 | 128 | 14/300 | 70 | 0,4 | 19,72% | 10,87% | 32,98% | 0,121 |

*Tabel 4. 16 Hasil Percobaan Hybrid LSTM-GRU Skenario 2*

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| No | Unit | Hyperparameter | | | MAE | MSE | RMSE | Avg  RMSE  Kfold  10 |
| Epochs | Batch\_size | Validation\_split |
| 1 | 64 | 22/300 | 90 | 0,4 | 17,82% | 6,87% | 37,56% | 0,133 |
| 2 | 128 | 14/300 | 90 | 0,4 | 28,70% | 14,75% | 38,40% | 0,130 |
| 3 | 64 | 12/300 | 70 | 0,4 | 28,96% | 14,11% | 37,56% | 0,131 |
| 4 | 64 | 13/300 | 70 | 0,3 | 27,23% | 14,53% | 38,12% |
| 5 | 128 | 15/300 | 70 | 0,4 | 26,85% | 14,03% | 37,45% | 0,133 |

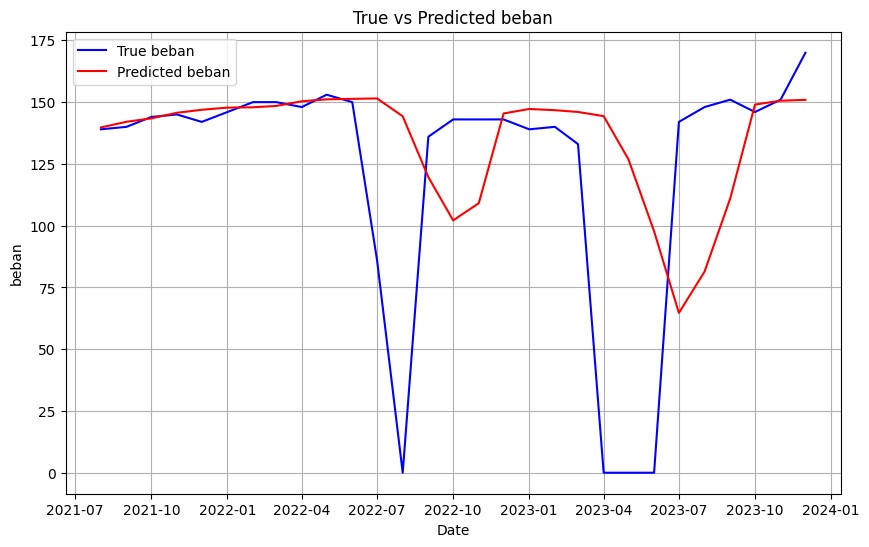
*Tabel 4. 17 Hasil Percobaan Hybrid LSTM-GRU Skenario 3*

Pada percobaan selanjutnya menggunakan metode *hybrid* dari LSTM dan GRU dengan menggunakan ketiga skenario tersebut. Pada skenario pertama mendapatkan hasil terbaik pada percobaan pertama dengan hasil RMSE 25,79%. Setelah itu dengan skenario kedua mendapatkan hasil terbaik dengan RMSE 31,35% pada percobaan keempat. Skenario terakhir yakni skenario ketiga memiliki nilai RMSE terendah yakni sebesar 37,45% yang diraih oleh percobaan kelima. Dari percobaan penelitian menggunakan model *hybrid* model terbaik didapatkan oleh skenario pertama pada percobaan pertama dengan RMSE sebesar 25,79%.

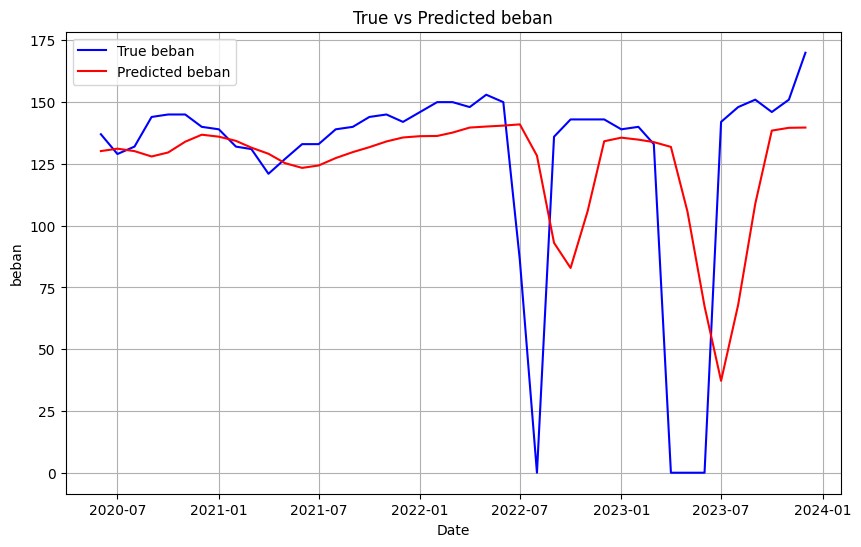
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| No | Model | Skenario | Percobaan | MAE | MSE | RMSE |
| 1 | GRU | 1 | 3 | 15,05% | 6,18% | 24,87% |
| 2 | LSTM | 1 | 1 | 16,19% | 7,02% | 26,49% |
| 3 | RNN | 2 | 4 | 19,03% | 9,45% | 30,74% |
| 4 | CNN | 1 | 5 | 15,47% | 6,60% | 25,70% |
| 5 | *Hybrid LSTM-GRU* | 2 | 1 | 15,74% | 6,65% | 25,79% |

*Tabel 4. 18 Hasil Terbaik dari Seluruh Model*

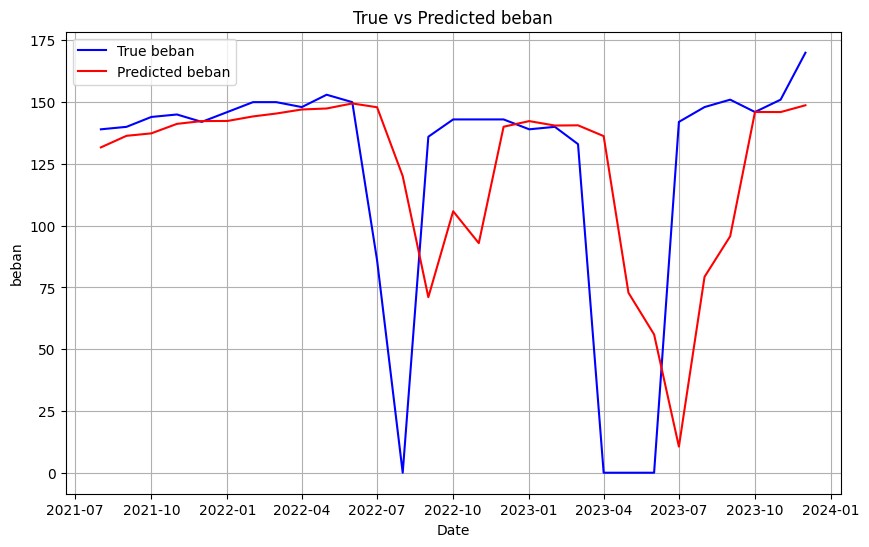
Setelah metode GRU, LSTM, RNN, CNN, dan *hybrid* LSTM-GRU diuji, terlihat bahwa setiap algoritma memiliki performa yang berbeda-beda tergantung pada skenario pembagian data dan arsitektur yang digunakan. Pada penelitian ini metode GRU lebih unggul dengan RMSE sebesar 24,87%, MSE sebesar 6,18%, dan MAE sebesar 15,05% serta grafik perbandingan data prediksi dan *true* yang ada pada gambar 4.1. Setelah model didapatkan maka mencoba memasukkan data prediksi dari beberapa perusahaan untuk memprediksi nilai beban puncak per pembangkit kedepannya, hasil tersebut ditunjukkan pada gambar 4.1 dengan nilai yang ditampilkan oleh tabel 4.19 dibawah ini.



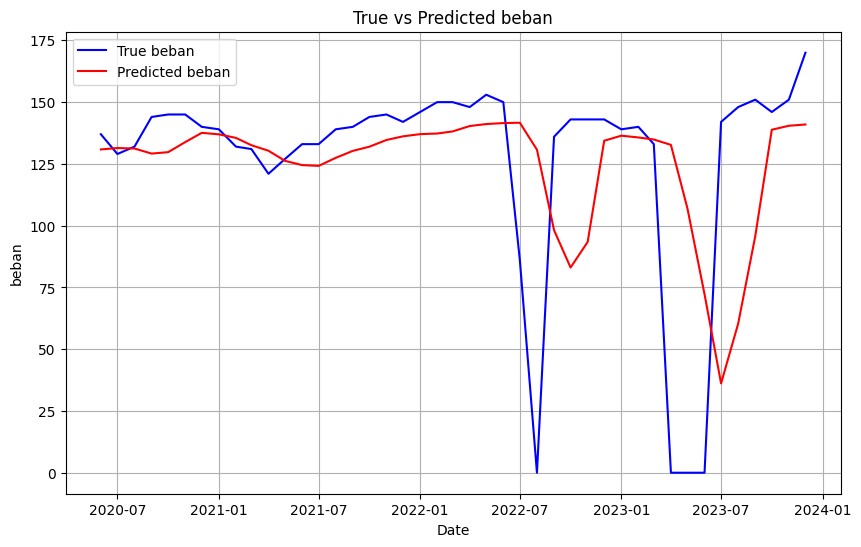
*Gambar 4. 1 Grafik Hasil Terbaik Model Hybrid LSTM-GRU*



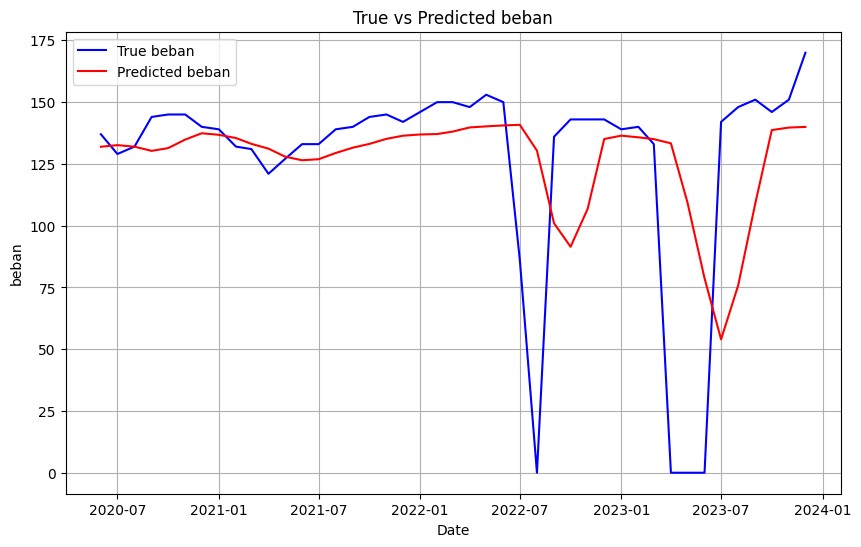
*Gambar 4. 2 Grafik Hasil Terbaik Model CNN*



*Gambar 4. 3 Grafik Hasil Terbaik Model RNN*

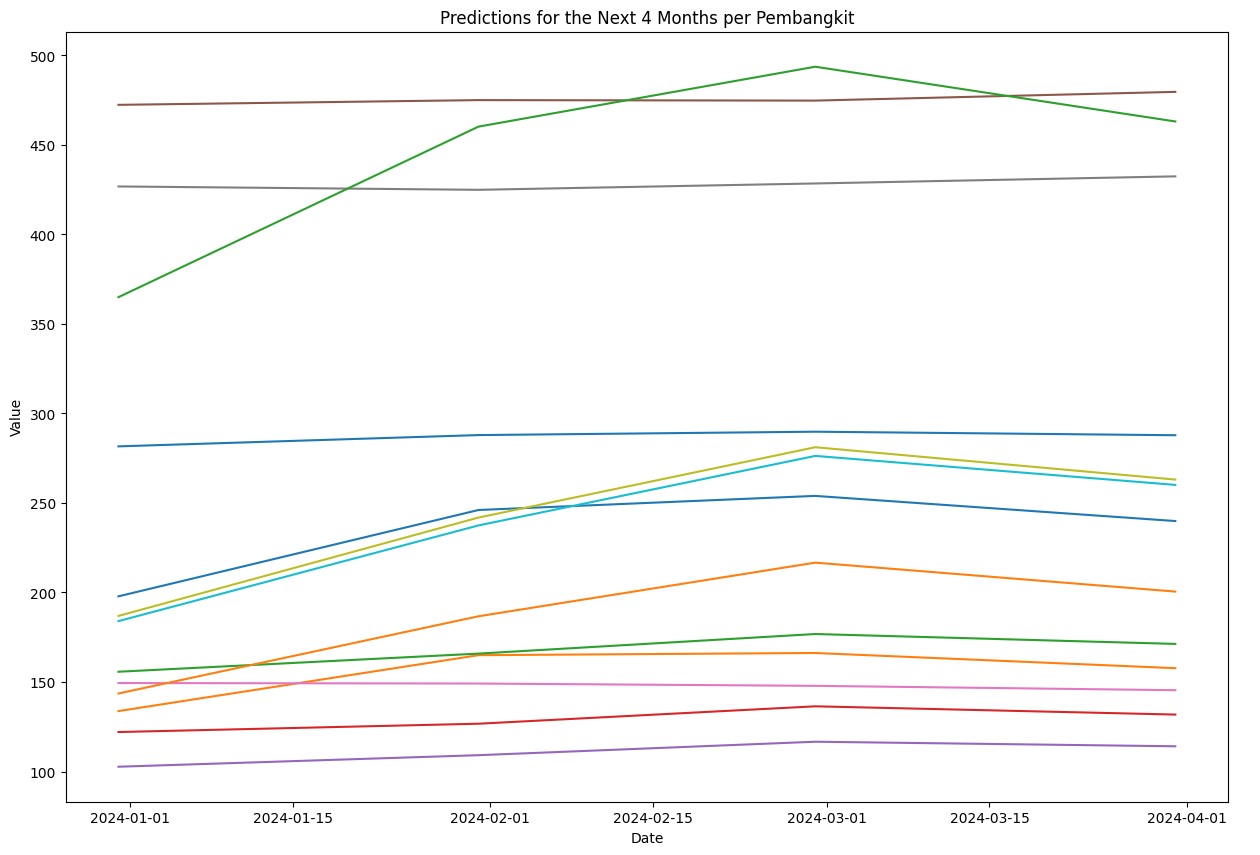


*Gambar 4. 4 Grafik Hasil Terbaik Model LSTM*



*Gambar 4. 5 Grafik Hasil Terbaik Model GRU*

Dari keseluruhan grafik yang ditampilkan pada gambar 4.1 hingga 4.5, terdapat informasi mengenai prediksi pada periode dari Juli 2022 hingga Juli 2023. Dalam periode tersebut, error terbesar terjadi. Hal ini disebabkan oleh grafik yang curam atau grafik yang memiliki nilai 0. Faktor-faktor ini mempengaruhi akurasi prediksi secara signifikan. Sehingga, tantangan dalam memprediksi data tersebut menjadi lebih besar.



*Gambar 4. 6 Grafik Hasil Prediksi Beban Puncak Pembangkit Listrik*

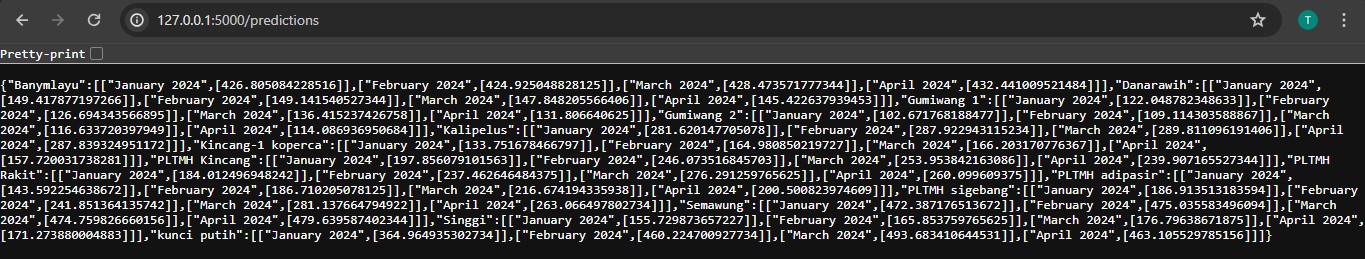
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nama Pembangkit** | **Januari** | **Februari** | **Maret** | **April** |
| Kincang | 197,85608 | 246,07352 | 253,95384 | 239,90717 |
| Kincang-1 koperca | 133,75168 | 164,98085 | 166,20317 | 157,72003 |
| Singgi | 155,72987 | 165,85376 | 176,79639 | 171,27388 |
| Gumiwang 1 | 122,04878 | 126,69434 | 136,41524 | 131,80664 |
| Gumiwang 2 | 102,67177 | 109,1143 | 116,63372 | 114,08694 |
| Semawung | 472,38718 | 475,03558 | 474,75983 | 479,6396 |
| Danarawih | 149,41788 | 149,14154 | 147,8482 | 145,42264 |
| Banyumlayu | 426,80508 | 424,92505 | 428,47357 | 432,441 |
| Sigebang | 186,91351 | 241,85136 | 281,13766 | 263,0665 |
| Rakit | 184,0125 | 237,46265 | 276,29126 | 260,0996 |
| Kalipelus | 281,62015 | 287,92294 | 289,8111 | 287,83932 |
| Adipasir | 143,59225 | 186,7102 | 216,6742 | 200,50082 |
| Kunci putih | 364,96494 | 460,2247 | 493,6834 | 463,10553 |

*Tabel 4. 19 Hasil Prediksi Pembangkit Tahun 2024*

Pada penelitian ini, dengan menggunakan data yang disediakan oleh perusahaan dan model yang telah dibuat yaitu GRU, diperoleh hasil prediksi yang ditampilkan pada tabel diatas. Hasil prediksi ini mencakup periode 4 bulan di tahun 2024 setelah data yang diberikan. Prediksi tersebut menunjukkan bahwa model GRU mampu menangkap pola dan tren dalam data historis. Hasil ini dapat digunakan oleh perusahaan untuk perencanaan strategis dan pengambilan keputusan yang lebih baik.

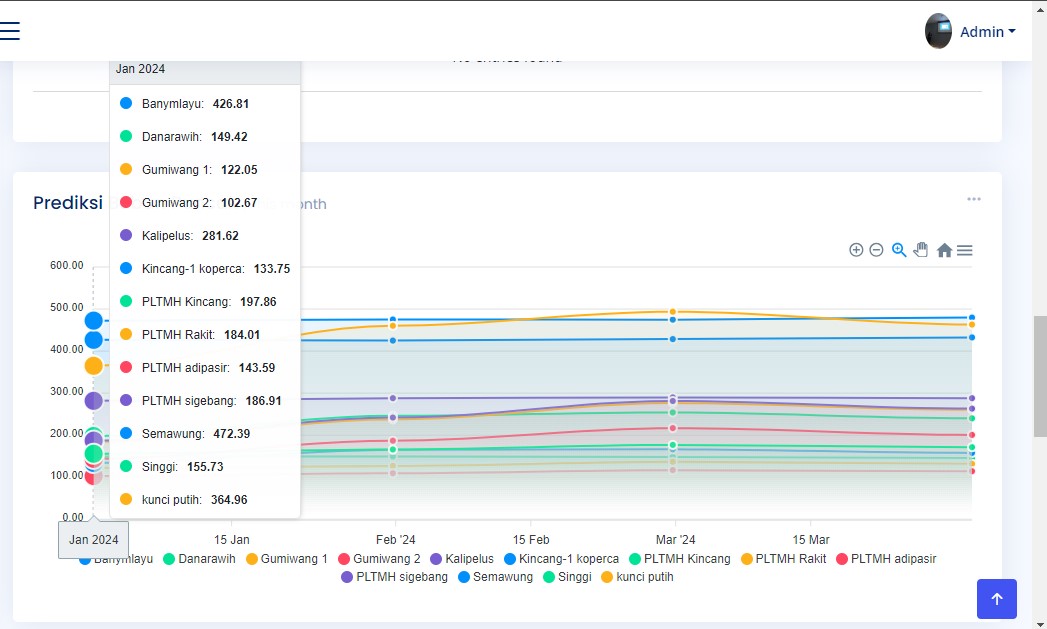
###### 4.1.3. Pengimplementasian Model pada Website

Setelah melakukan percobaan dan mengevaluasi lima model yang terdiri dari RNN, GRU, LSTM, CNN, dan *hybrid* LSTM-GRU, penelitian telah berhasil menemukan model terbaik. Model terbaik kemudian diubah menjadi *file* H5 dan diintegrasikan ke dalam *framework* Flask. Selanjutnya, membuat API dari *framework* Flask yang akan memberikan *response* sesuai dengan yang ditampilkan pada gambar 4.7, yaitu hasil prediksi model untuk 4 bulan ke depan. Data tersebut kemudian diakses oleh *framework* Laravel melalui metode API "GET".



*Gambar 4. 7 Response API Framework Flask*

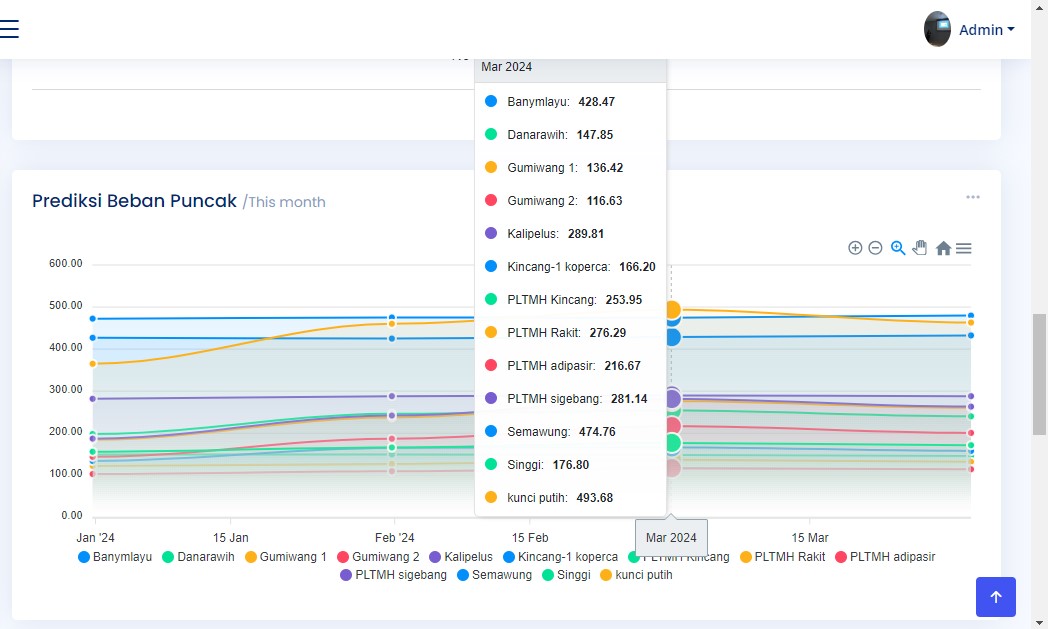
Setelah menerima respons dari API, data tersebut akan ditampilkan dalam bentuk grafik pada halaman *dashboard* web seperti yang terlihat pada gambar dibawah ini. Masing-masing gambar pada gambar 4.8 hingga gambar 4.11 menampilkan nilai-nilai hasil prediksi beban puncak secara berturut dari Januari hingga April. Dalam grafik tersebut dapat dipilah sesuai grafik yang ingin ditampilkan seperti gambar 4.12 yang menampilkan 1 grafik pembangkit.



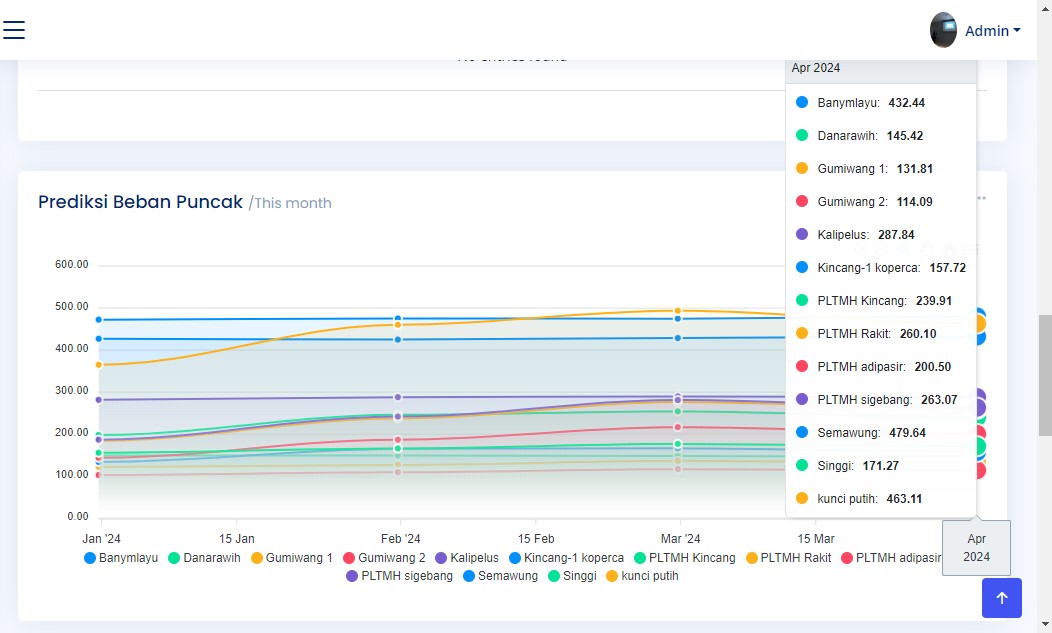
*Gambar 4. 8 Tampilan Grafik Prediksi Januari*



*Gambar 4. 9 Tampilan Grafik Prediksi Februari*



*Gambar 4. 10 Tampilan Grafik Prediksi Maret*



*Gambar 4. 11 Tampilan Grafik Prediksi April*



*Gambar 4. 12 Tampilan Grafik Prediksi Hanya 1 Pembangkit*

# BAB V

## KESIMPULAN

#### 5.1. Kesimpulan

Setelah melakukan penelitian dari skripsi Rancang Bangun Aplikasi Prediksi Beban Puncak Penggunaan Listrik pada Pembangkit Listrik dengan Model *Deep Learning* Berbasis Web: Studi Kasus Web “*Connect* UP2DJTY” maka dari itu memiliki beberapa hasil sebagai berikut:

1. Prediksi adalah proses menggunakan algoritma untuk menganalisis data masa lalu dan menemukan pola atau tren yang bisa digunakan untuk memperkirakan nilai atau kejadian di masa depan. Proses ini melibatkan penggunaan data untuk membuat estimasi berdasarkan informasi yang sudah ada. Dengan menggunakan model yang telah terlatih maka selanjutnya digunakan kembali dengan data yang akan diprediksi.
2. Di antara kelima metode tersebut, yaitu RNN, GRU, LSTM, CNN, dan hybrid LSTM-GRU, arsitektur model GRU menggunakan 64-unit, batch size sebesar 70, dan validation split 0,4, dengan hasil error GRU menunjukkan RMSE sebesar 24,87%, MSE sebesar 6,18%, dan MAE sebesar 15,05%, sehingga dari hasil error tersebut GRU yang dibandingkan dengan model lainnya didapatkan bahwa GRU dengan skenario ketiga adalah model paling efektif; untuk mengimplementasikan fitur ke dalam website adalah dengan menggunakan API yang telah dibuat oleh framework Flask, yang secara teknis menggunakan file dengan format H5 berisikan algoritma yang telah dilatih sebelumnya, dan setelah itu program website akan mengambil data tersebut dengan metode “GET” dan menampilkannya ke dalam dashboard interface.

#### 5.2. Saran

Berdasarkan dari hasil yang telah dibuat dalam skripsi ini, penulis memiliki beberapa saran yang bisa digunakan untuk pengembangan selanjutnya, saran sebagai berikut:

1. Penelitian ini masih menggunakan data *open source* dan belum menggunakan data perusahaan untuk *train* dan *testing*. Dalam pengembangan selanjutnya, dapat menggunakan data perusahaan yang lebih banyak untuk masing-masing pembangkit sehingga pola-pola prediksi akan lebih akurat.
2. Dalam pengembangan selanjutnya, dapat dicoba menggunakan susunan *layer* yang berbeda dan juga menggunakan metode lainnya untuk mendapatkan hasil yang lebih baik.

## DAFTAR PUSTAKA

ArunKumar, K. E., Kalaga, D. V., Mohan Sai Kumar, Ch., Kawaji, M., & Brenza, T. M. (2022). Comparative analysis of Gated Recurrent Units (GRU), long Short-Term memory (LSTM) cells, autoregressive Integrated moving average (ARIMA), seasonal autoregressive Integrated moving average (SARIMA) for forecasting COVID-19 trends. *Alexandria Engineering Journal*, *61*(10), 7585–7603. https://doi.org/10.1016/j.aej.2022.01.011

Attiwal, P., & Indora, S. (2024). A Comprehensive Review of Modern Methods for Load Prediction in the Smart Grid. *Recent Patents on Engineering*, *18*. https://doi.org/10.2174/1872212118666230423143331

Bai, S., Kolter, J., & Koltun, V. (2018). *An Empirical Evaluation of Generic Convolutional and Recurrent Networks for Sequence Modeling*.

Barman, P. P., & Boruah, A. (2018). A RNN based Approach for next word prediction in Assamese Phonetic Transcription. *Procedia Computer Science*, *143*, 117–123. https://doi.org/10.1016/j.procs.2018.10.359

Bi̇Lgi̇Li̇, M., Arslan, N., Şekerteki̇N, A., & Yaşar, A. (2022). Application of long short-term memory (LSTM) neural network based on deep learning for electricity energy consumption forecasting. *Turkish Journal of Electrical*

*Engineering and Computer Sciences*, *30*(1), 140–157. https://doi.org/10.3906/elk-2011-14

Chen, Y., Liao, S., & Xu, J. (2022). Emergency load-shedding optimization control method based on reinforcement learning assistance. *Energy Reports*, *8*, 1051–1061. https://doi.org/10.1016/j.egyr.2022.02.140

Chicco, D., Warrens, M. J., & Jurman, G. (2021). The coefficient of determination R-squared is more informative than SMAPE, MAE, MAPE, MSE and RMSE in regression analysis evaluation. *PeerJ. Computer Science*, *7*, e623. https://doi.org/10.7717/peerj-cs.623

Chrismawan, P. E. E., Winardi, S., & Tambunan, R. (2023). *Robust Stock Price*

*Prediction using Gated Recurrent Unit (GRU)*. *5*(1). https://doi.org/10.35842/ijicom

Dewandra, A. R. F., Wibawa, A. P., Pujianto, U., Utama, A. B. P., & Nafalski, A. (2022). Journal Unique Visitors Forecasting Based on Multivariate Attributes Using CNN. *International Journal of Artificial Intelligence Research*, *6*(2). https://doi.org/10.29099/ijair.v6i1.274

Dhaked, D. K., Dadhich, S., & Birla, D. (2023). Power Output Forecasting of Solar Photovoltaic Plant Using LSTM. *Green Energy and Intelligent*

*Transportation*, *2*(5), 100113. https://doi.org/10.1016/j.geits.2023.100113

Enterprise, J. (2018). *HTML, PHP, dan MySQL untuk Pemula*. PT. Elex Media Komputindo.

Grinberg, M. (2018). *Flask web development: Developing web applications with Python* (2nd edition). O’Reilly.

Hasanuddin, Asgar, H., & Hartono, B. (2022). Rancang Bangun REST API Aplikasi Weshare sebagai Upaya Mempermudah Pelayanan Donasi Kemanusiaan. *Jurnal Informatika Teknologi dan Sains*, *4*(1), 8–14. https://doi.org/10.51401/jinteks.v4i1.1474

Huang, B., Tong, L., & Zuo, Y. (2022a). Short-term power generation load forecasting based on LSTM neural network. *Journal of Physics: Conference*

*Series*, *2247*(1), 012033. https://doi.org/10.1088/1742-6596/2247/1/012033

Huang, B., Tong, L., & Zuo, Y. (2022b). Short-term power generation load forecasting based on LSTM neural network. *Journal of Physics: Conference*

*Series*, *2247*(1), 012033. https://doi.org/10.1088/1742-6596/2247/1/012033

Islam, M. S., & Hossain, E. (2021). Foreign exchange currency rate prediction using a GRU-LSTM hybrid network. *Soft Computing Letters*, *3*, 100009. https://doi.org/10.1016/j.socl.2020.100009

Karim, Md. E., Foysal, Md., & Das, S. (2023). Stock Price Prediction Using BiLSTM and GRU-Based Hybrid Deep Learning Approach. In A. Khanna, D. Gupta, V. Kansal, G. Fortino, & A. E. Hassanien (Eds.), *Proceedings of Third Doctoral Symposium on Computational Intelligence* (pp. 701–711). Springer Nature Singapore.

Karno, A. S. B. (2020). Prediksi Data Time Series Saham Bank BRI Dengan Mesin Belajar LSTM (Long ShortTerm Memory). *Journal of Informatic and Information Security*, *1*(1), 1–8. https://doi.org/10.31599/jiforty.v1i1.133

Li, W., Wu, H., Zhu, N., Jiang, Y., Tan, J., & Guo, Y. (2021). Prediction of dissolved oxygen in a fishery pond based on gated recurrent unit (GRU).

*Information Processing in Agriculture*, *8*(1), 185–193.

https://doi.org/10.1016/j.inpa.2020.02.002

Malek, S., Melgani, F., & Bazi, Y. (2017). One-dimensional convolutional neural networks for spectroscopic signal regression. *Journal of Chemometrics*, *32*. https://doi.org/10.1002/cem.2977

Martelli, A., Holden, S., McGuire, P., & Ravenscroft Martelli, A. (2023). *Python in a Nutshell: A Desktop Quick Reference* (Fourth Edition). O’Reilly Media.

Pandowo, H., Kusumaningrum, D., & Amir, V. (2022). Implementasi Deep Learning Untuk Optimasi Slump Menggunakan Convolutional Neural Network Pada PT. Handaru Wijaya Mulya. *JURNAL INFORMATIKA*, *9*(1).

Peng, F.-L., Qiao, Y.-K., & Yang, C. (2023). A LSTM-RNN based intelligent control approach for temperature and humidity environment of urban utility tunnels. *Heliyon*, *9*(2), e13182. https://doi.org/10.1016/j.heliyon.2023.e13182

Permana, A. Y., & Romadlon, P. (2019). *PERANCANGAN SISTEM INFORMASI PENJUALAN PERUMAHAN MENGUNAKAN METODE SDLC PADA PT. MANDIRI LAND PROSPEROUS BERBASIS MOBILE*. *10*.

Pratiwi, D., Pratama, D. A., & Rochman, A. (2024). Application Programming

Interface (API) Development for Student Attendance System Using QR Code. *Qeios*. https://doi.org/10.32388/9OJFXV

Pumsirirat, A., & Yan, L. (2018). Credit Card Fraud Detection using Deep Learning based on Auto-Encoder and Restricted Boltzmann Machine. *International Journal of Advanced Computer Science and Applications*, *9*(1). https://doi.org/10.14569/IJACSA.2018.090103

Raup, A., Ridwan, W., Khoeriyah, Y., Supiana, S., & Zaqiah, Q. Y. (2022). Deep Learning dan Penerapannya dalam Pembelajaran. *JIIP - Jurnal Ilmiah Ilmu Pendidikan*, *5*(9), 3258–3267. https://doi.org/10.54371/jiip.v5i9.805

Riziq sirfatullah Alfarizi, M., Zidan Al-farish, M., Taufiqurrahman, M., Ardiansah, G., & Elgar, M. (2022). Penggunaan Python Sebagai Bahasa Pemrograman untuk Machine Learning dan Deep Learning. *Karimah Tauhid*, *2*(1).

Rizvi, D. R., & Khalid, M. (2024). Performance Analysis of Stocks using Deep

Learning Models. *5th International Conference on Innovative Data Communication Technologies and Application (ICIDCA 2024)*, *233*, 753–

762. https://doi.org/10.1016/j.procs.2024.03.264

Sabar Sautomo & Hilman Ferdinandus Pardede. (2021). Prediksi Belanja Pemerintah Indonesia Menggunakan Long Short-Term Memory (LSTM). *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, *5*(1), 99–106. https://doi.org/10.29207/resti.v5i1.2815

Sadli, M., Fajriana, Fuadi, W., Ermatita, & Pahendra, I. (2020). Electrical peak load forecasting using long short term memory and support vector machine. *IOP Conference Series: Materials Science and Engineering*, *725*(1), 012060. https://doi.org/10.1088/1757-899X/725/1/012060

Sari, Y., Arifin, Y. F., Novitasari, N., & Faisal, M. R. (n.d.). Deep Learning Approach Using the GRU-LSTM Hybrid Model for Air Temperature

Prediction on Daily Basis. *International Journal of Intelligent Systems and Applications in Engineering*.

Selle, N., Yudistira, N., & Dewi, C. (2022). Perbandingan Prediksi Penggunaan Listrik dengan Menggunakan Metode Long Short Term Memory (LSTM) dan Recurrent Neural Network (RNN). *Jurnal Teknologi Informasi dan Ilmu Komputer*, *9*(1), 155–162. https://doi.org/10.25126/jtiik.2022915585

Shen, G., Tan, Q., Zhang, H., Zeng, P., & Xu, J. (2018). Deep Learning with Gated Recurrent Unit Networks for Financial Sequence Predictions. *Recent*

*Advancement in Information and Communication Technology:*, *131*, 895–

903. https://doi.org/10.1016/j.procs.2018.04.298

Shenfield, A., & Howarth, M. (2020). A Novel Deep Learning Model for the

Detection and Identification of Rolling Element-Bearing Faults. *Sensors*, *20*(18), 5112. https://doi.org/10.3390/s20185112

Sontana, I., Rahmatulloh, A., & Rachman, A. N. (2019). Application Programming Interface Google Picker Sebagai Penyimpanan Data Sistem Informasi Arsip

Berbasis Cloud. *Jurnal Nasional Teknologi dan Sistem Informasi*, *5*(1), 25–

32. https://doi.org/10.25077/TEKNOSI.v5i1.2019.25-32

Stauffer, M. (2019). *Laravel: Up & Running: A Framework for Building Modern PHP Apps* (Second edition). O’Reilly Media.

Sulistyo, M. (2018). *DATABASE*. https://doi.org/10.31219/osf.io/pbfzm

Surya, L., Teja Yarlagadda, R., & Patel, M. (2021). *PHP For Beginners* (First edition). REDSHINE Publication. https://redshine.co.in/product/978-93-

90937-52-3/

Talwariya, A., Singh, P., Jobanputra, J. H., & Kolhe, M. L. (2023). Machine learning based renewable energy generation and energy consumption forecasting. *Energy Sources, Part A: Recovery, Utilization, and*

*Environmental Effects*, *45*(2), 3266–3278. https://doi.org/10.1080/15567036.2023.2194257

Wahid, A. A. (2020). *Analisis Metode Waterfall Untuk Pengembangan Sistem Informasi*.

Wakhid, M. A., Raharjo, A. B., & Indralaksono, R. (2022). Peramalan Beban pada Rencana Operasi Harian dengan Menggunakan LSTM Studi Kasus: Sub

Sistem Sulawesi Selatan. *JUTI: Jurnal Ilmiah Teknologi Informasi*, *20*(2), 99–108. https://doi.org/10.12962/J24068535.V20I2.A1138

Yani, A., & Saputra, B. (2018). *RANCANG BANGUN SISTEM INFORMASI EVALUASI SISWA DAN KEHADIRAN GURU BERBASIS WEB (Studi*

*Kasus di SMK Nusa Putra Kota Tangerang)*. *11*(2).

Z. Ye, A. Gilman, Q. Peng, K. Levick, P. Cosman, & L. Milstein. (2019). Comparison of Neural Network Architectures for Spectrum Sensing. *2019*

*IEEE Globecom Workshops (GC Wkshps)*, 1–6. https://doi.org/10.1109/GCWkshps45667.2019.9024482

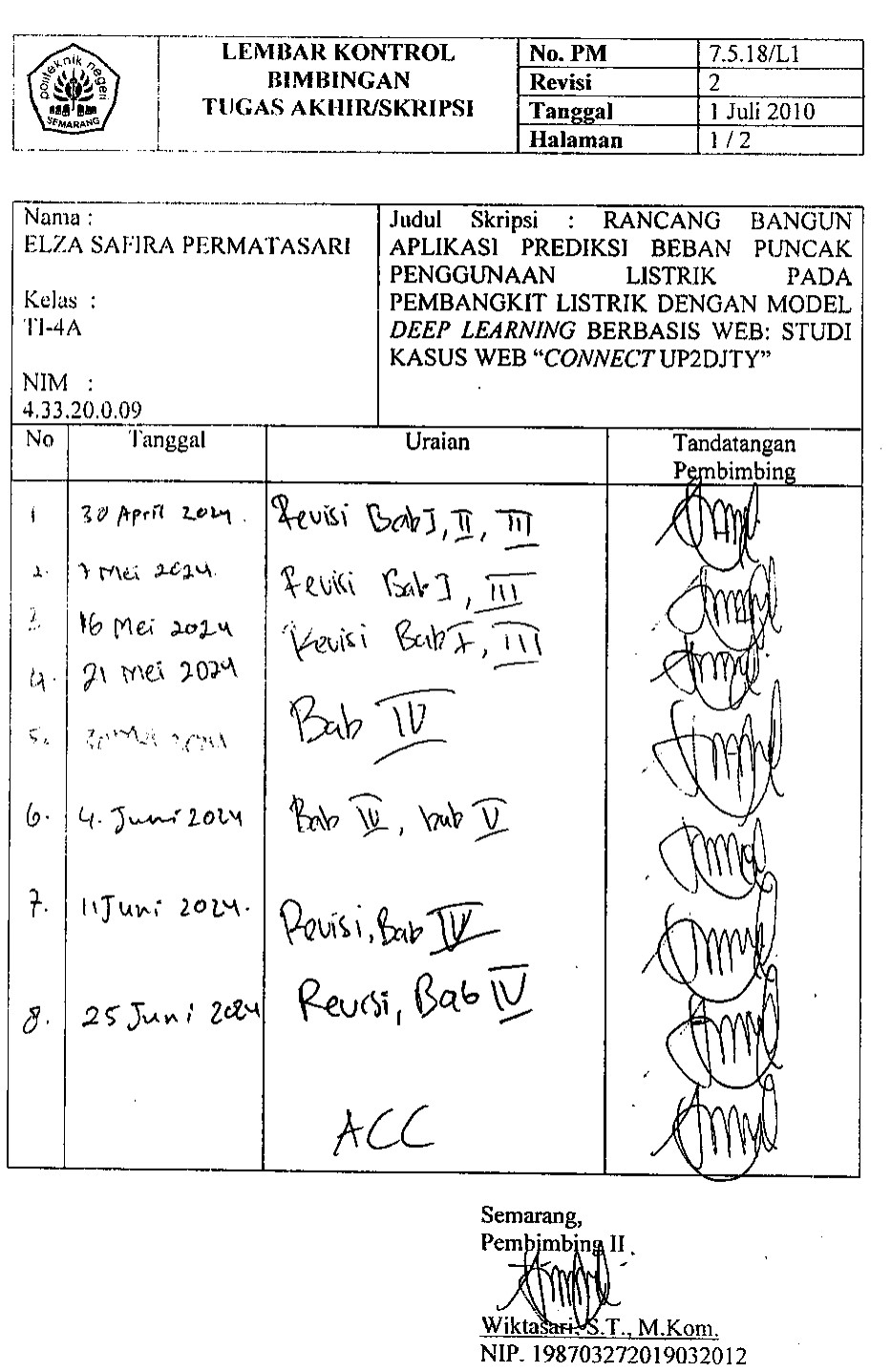
Zhu, Y. (2020). Stock price prediction using the RNN model. *Journal of Physics:*

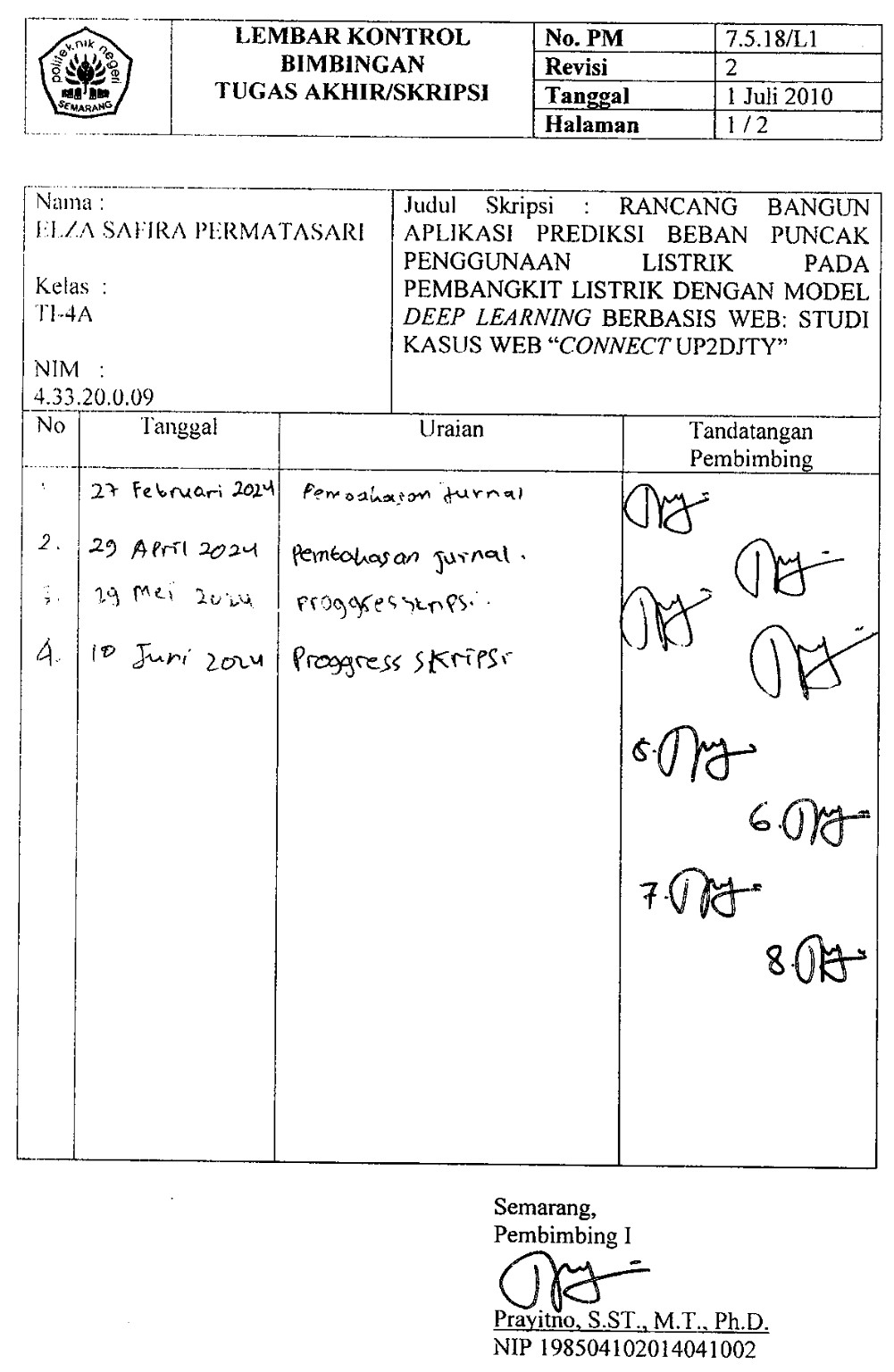
*Conference Series*, *1650*, 032103. https://doi.org/10.1088/1742-

6596/1650/3/032103

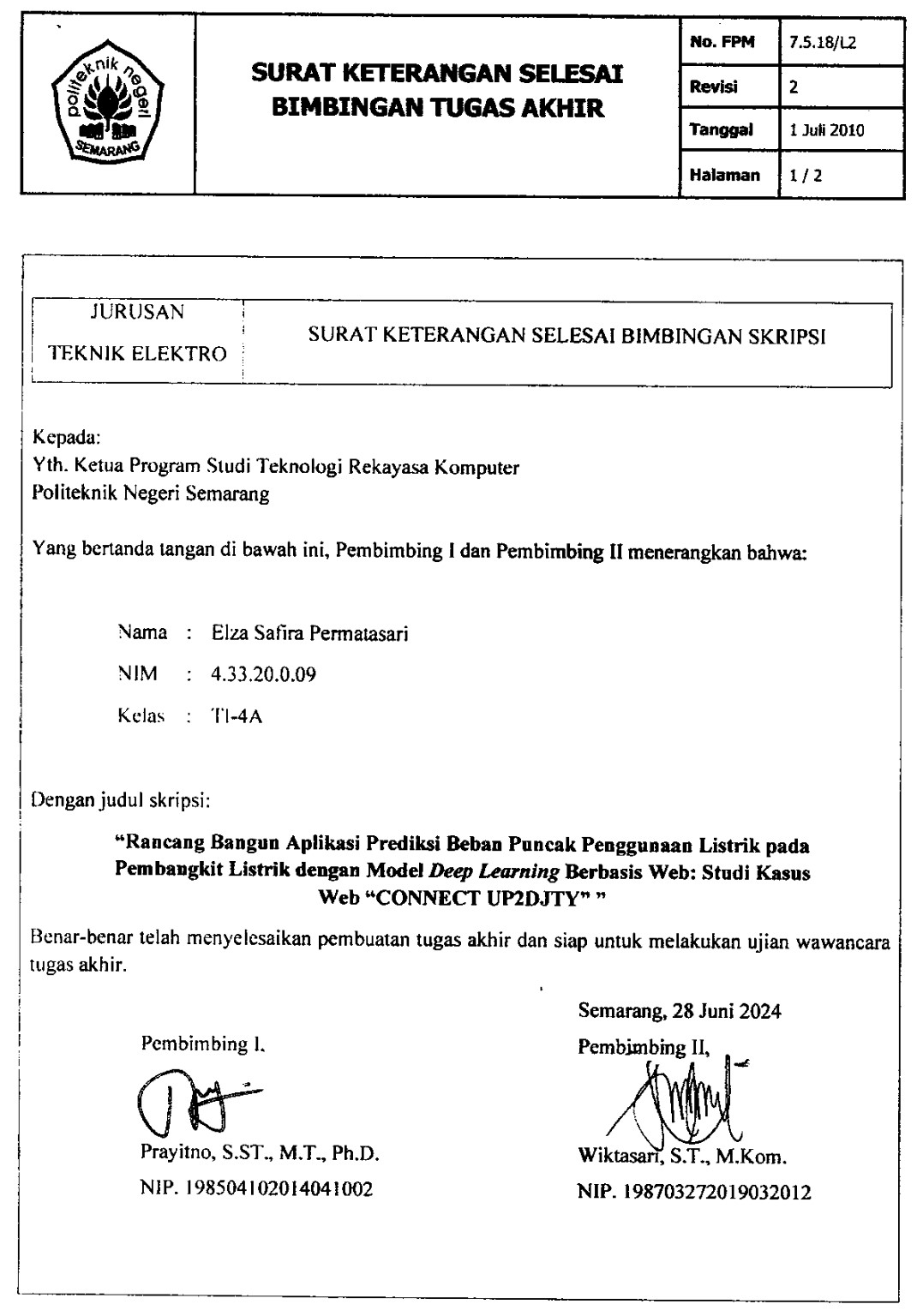
## LAMPIRAN

### A. Lembar Kontrol Bimbingan

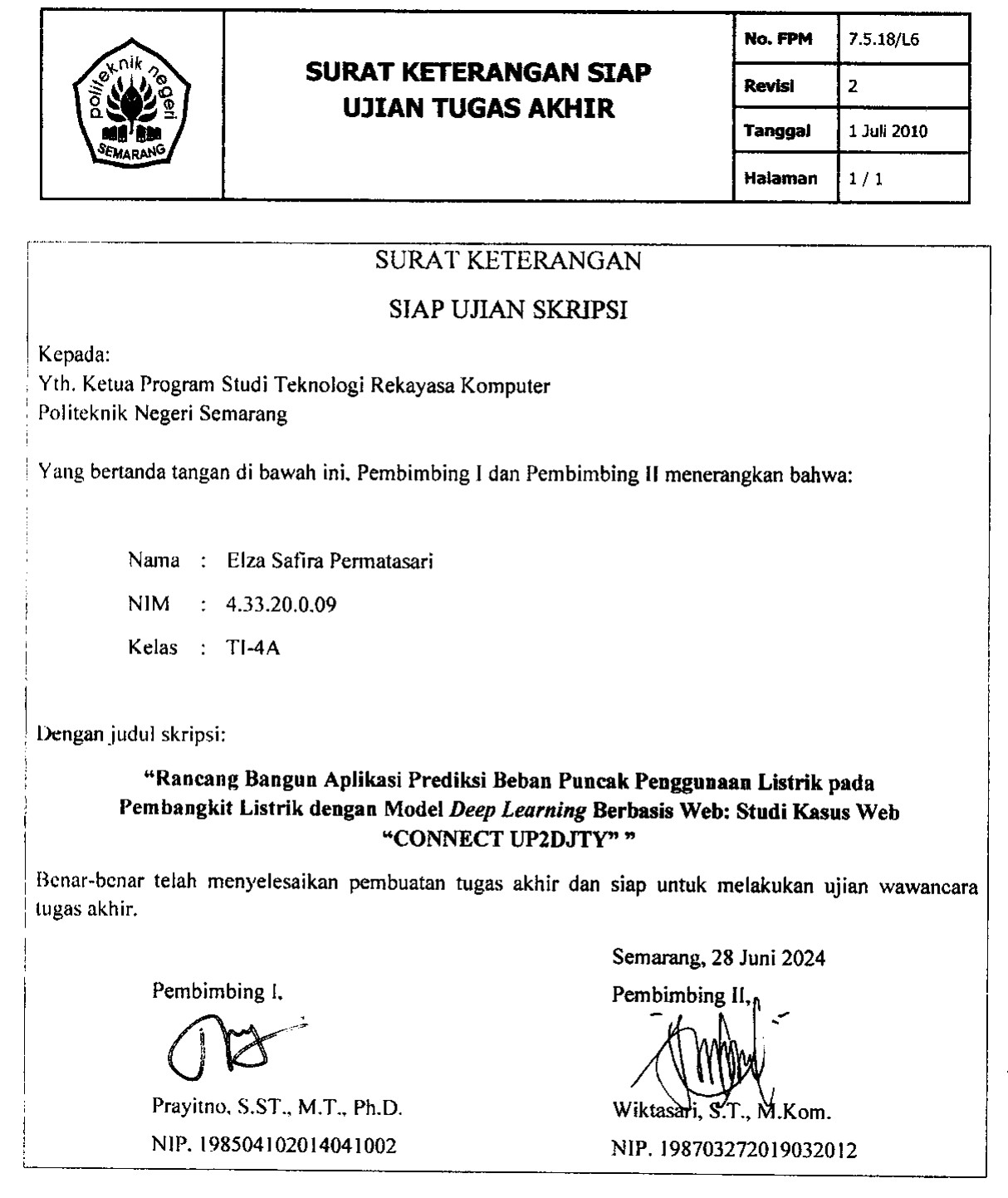




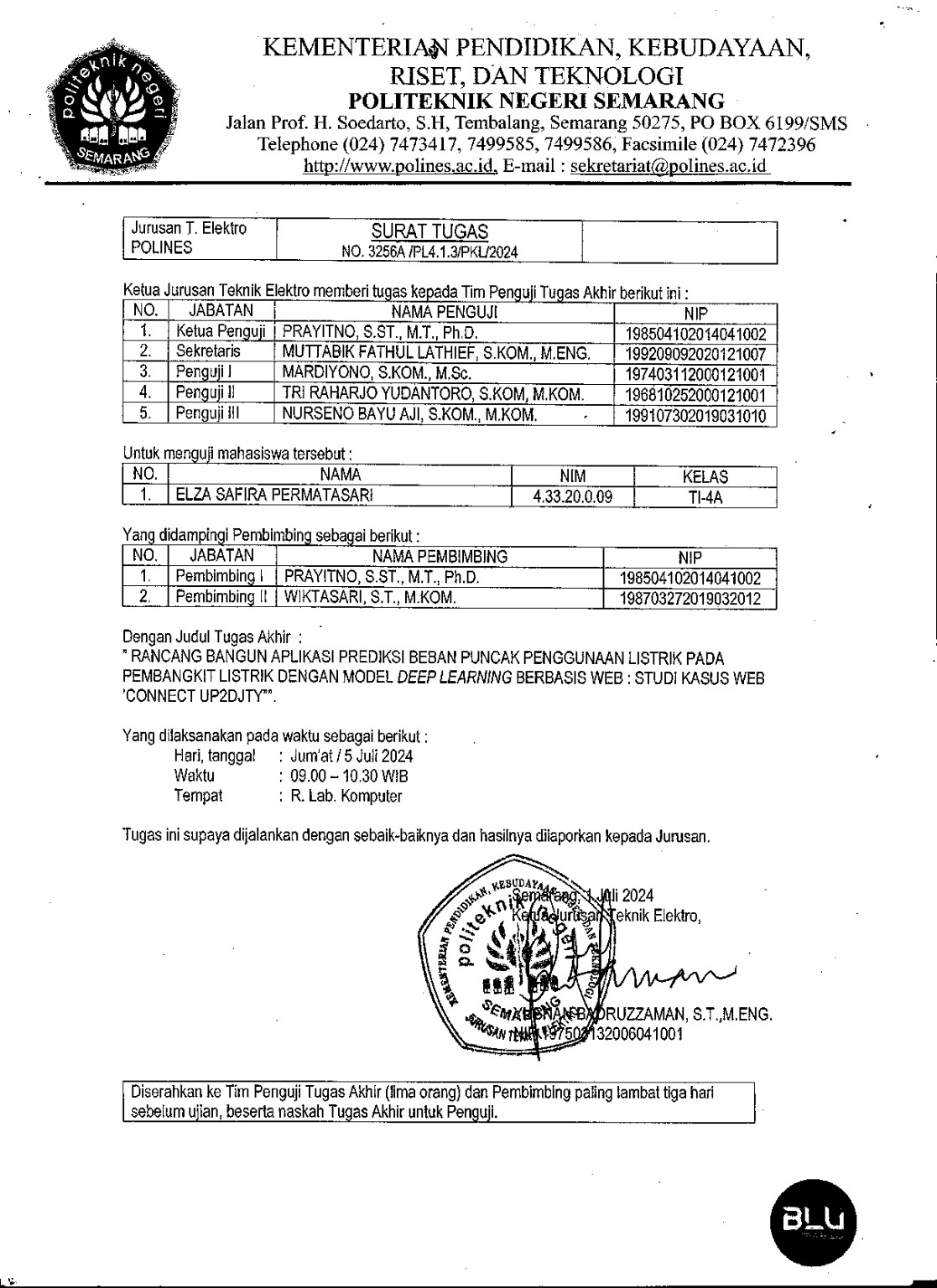
### B. Surat Keterangan Selesai bimbingan Skripsi



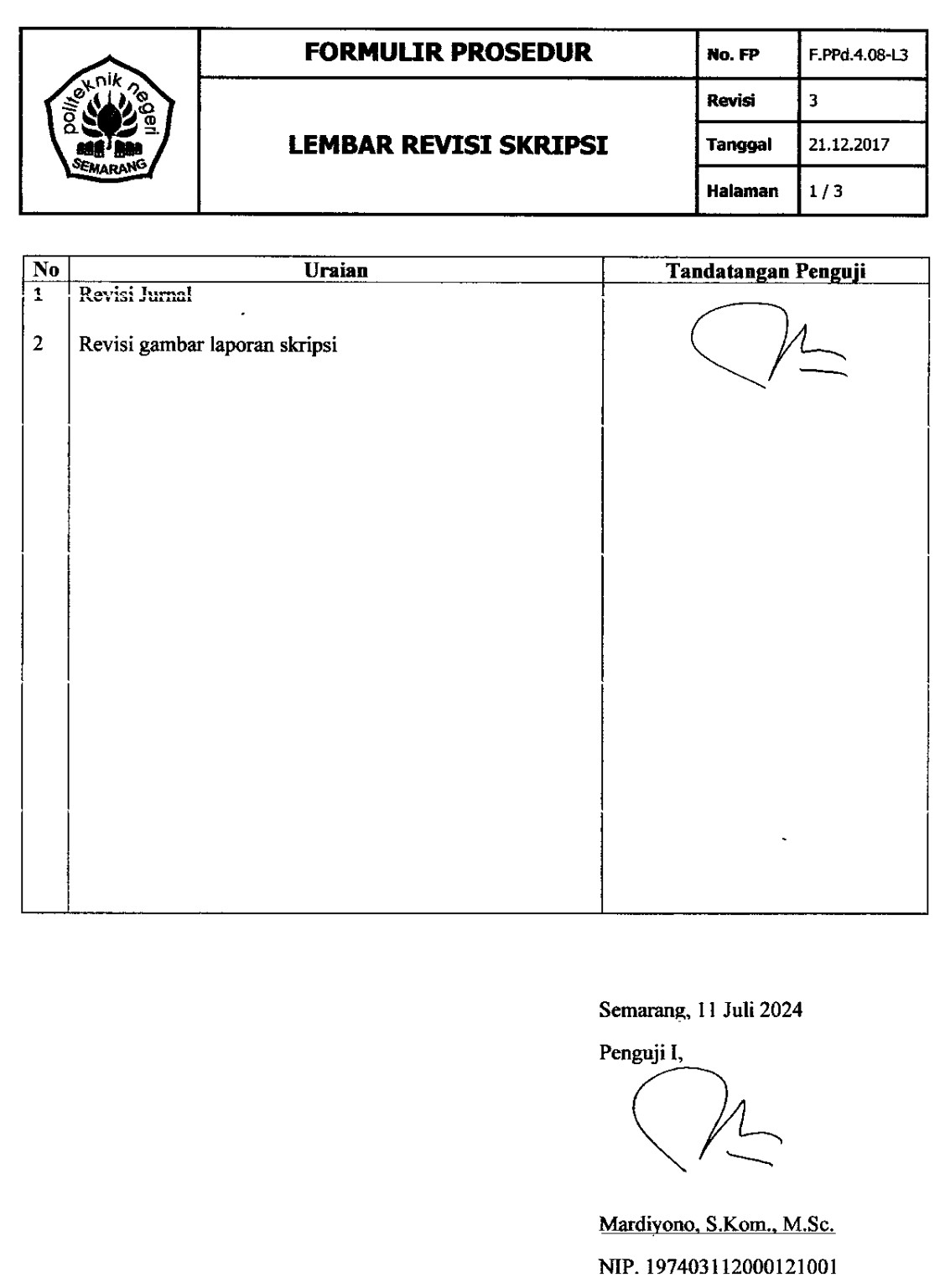
### C. Surat Keterangan Siap Ujian Skripsi

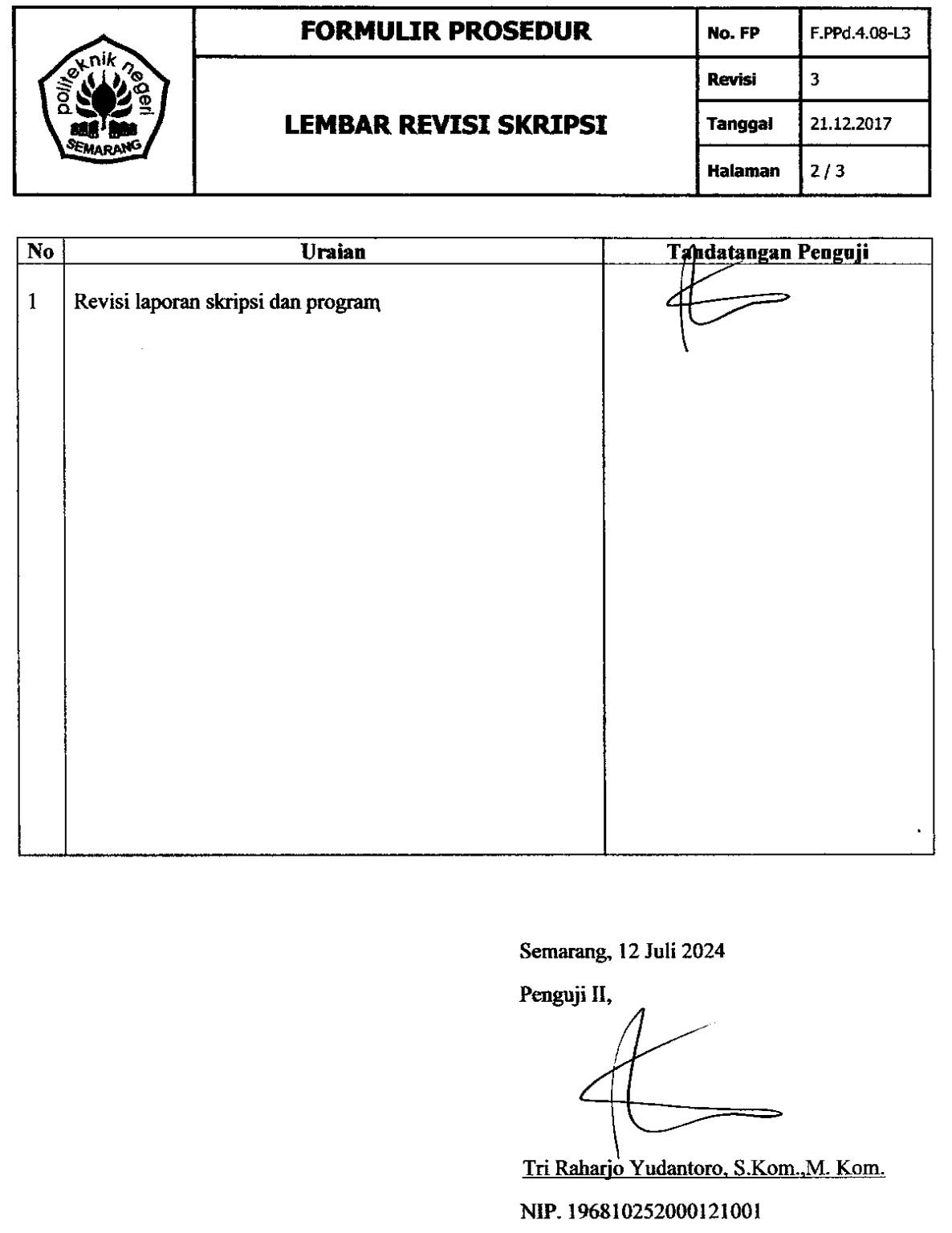


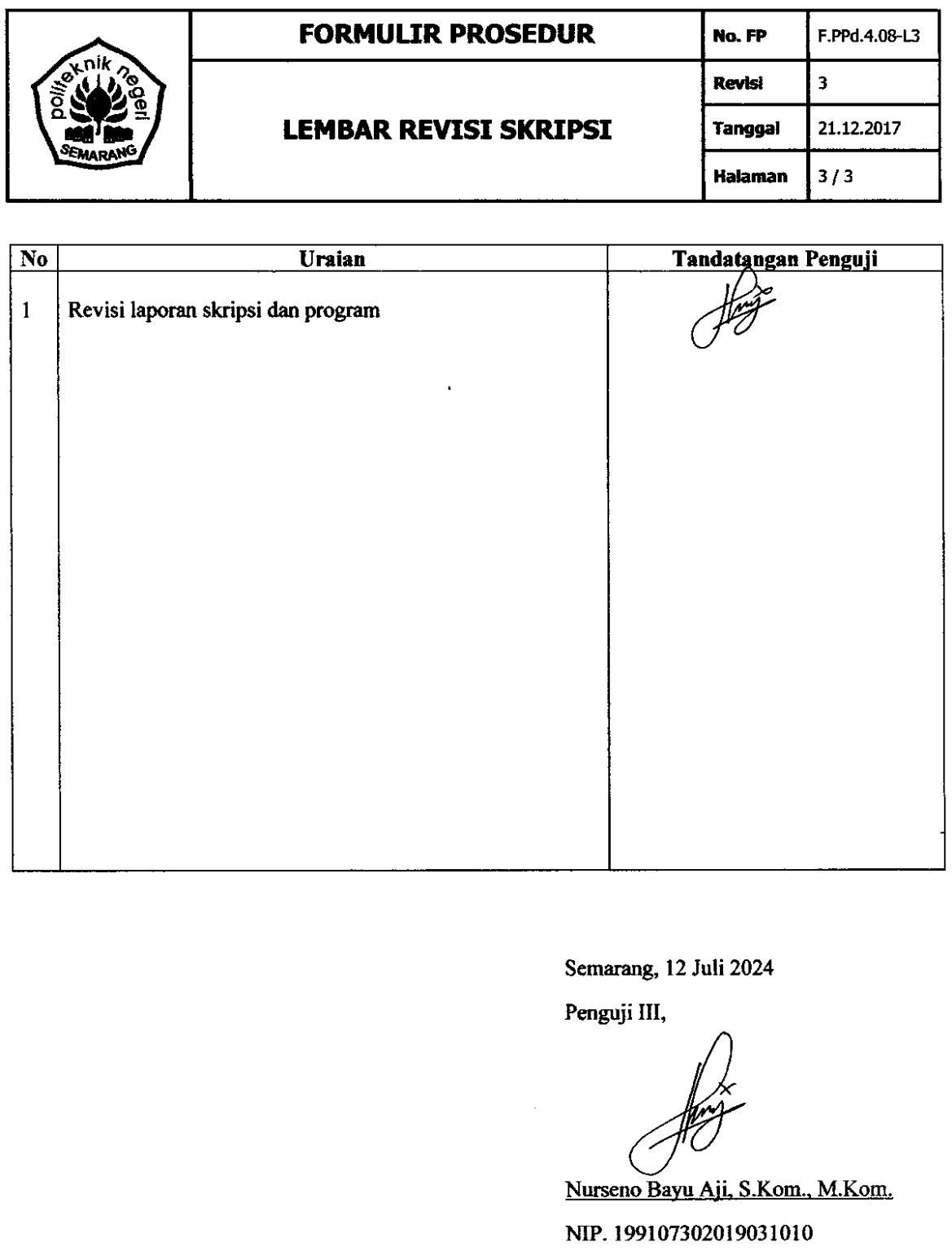
### D. Surat Tugas



### E. Lembar Revisi Skripsi



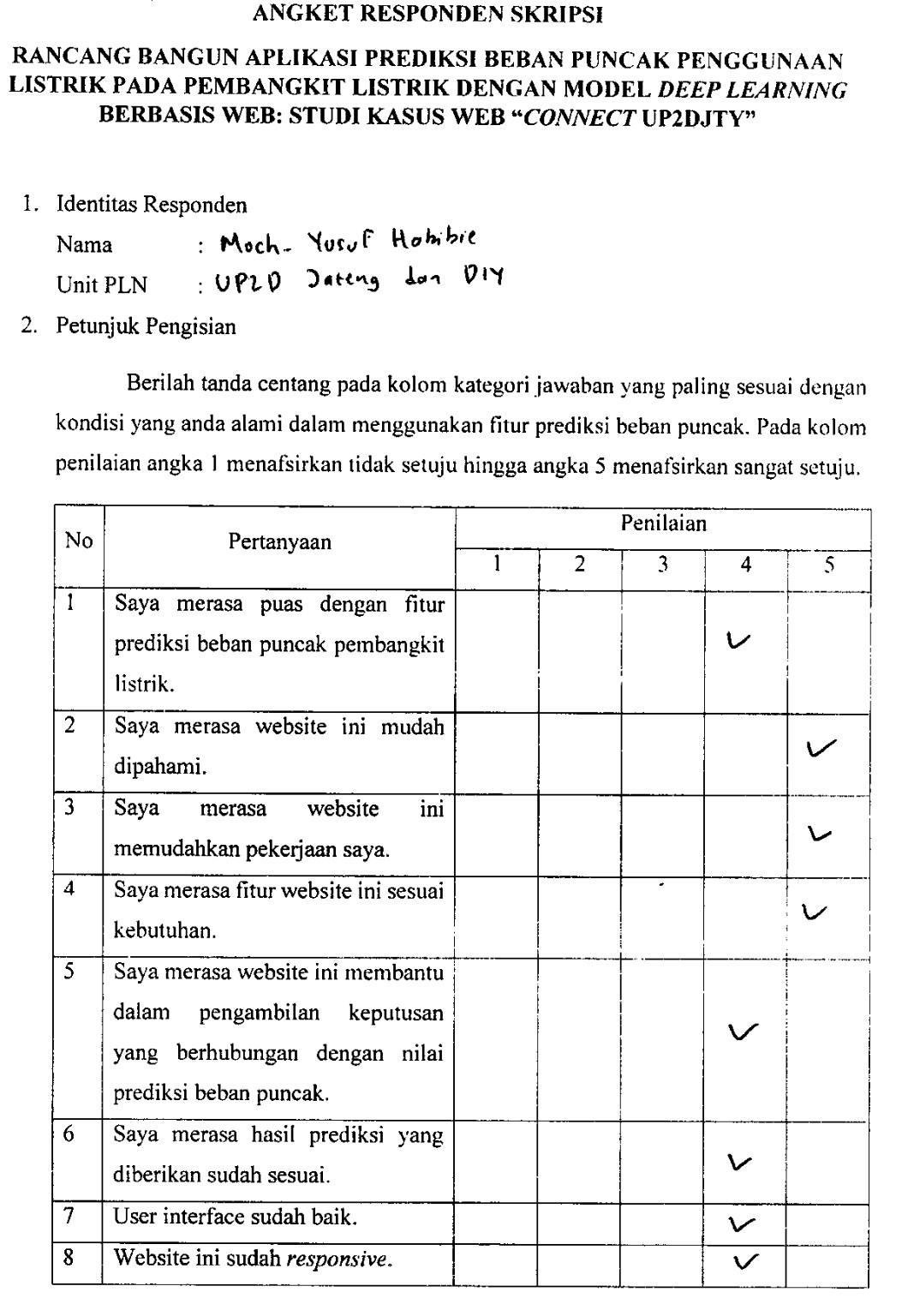




### F. Lembar Pernyataan Telah Melakukan Revisi

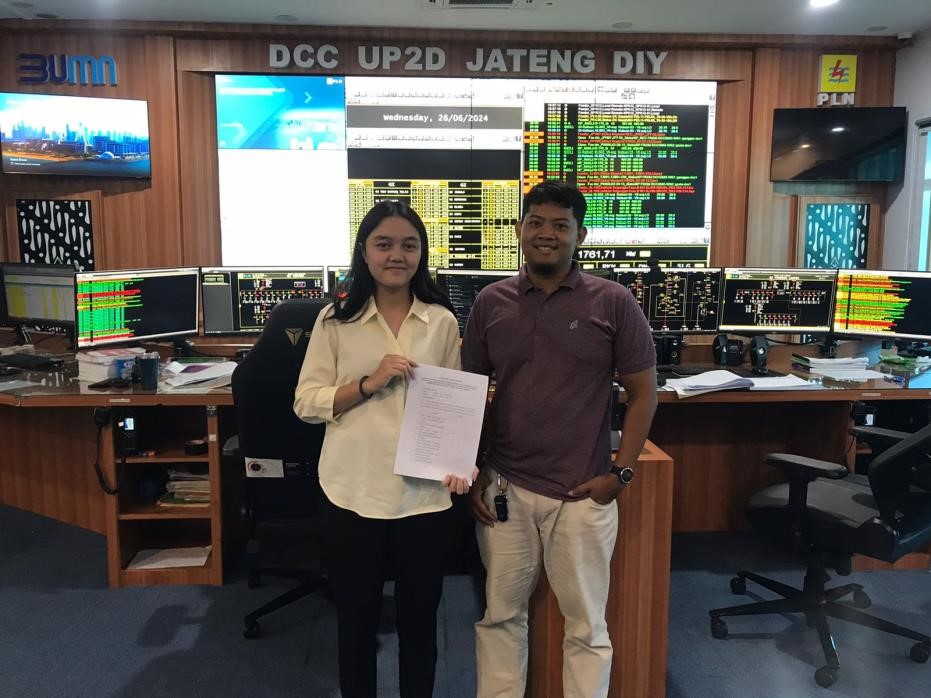


### G. Lembar Responden





### H. Dokumentasi



### I. Kode Program *Deep Learning*

|  |
| --- |
| import pandas as pd import numpy as np  import matplotlib.pyplot as plt import seaborn as sns  from sklearn.preprocessing import MinMaxScaler from tensorflow.keras.models import Sequential from tensorflow.keras.layers import GRU, Dense, Dropout from tensorflow.keras.callbacks import EarlyStopping, ModelCheckpoint  from sklearn.metrics import mean\_absolute\_error, mean\_squared\_error import pandas as pd    df = pd.read\_csv(r'Your\_path)  df['date'] = pd.to\_datetime(df['date'], format='%d-%b-%y') df.set\_index('date', inplace=True) df = df.asfreq('MS')    scaler = MinMaxScaler()  scaled\_data = scaler.fit\_transform(df)    sequence\_length = 3 num\_features = len(df.columns) sequences = [] labels = []  for i in range(len(scaled\_data) - sequence\_length):  seq = scaled\_data[i:i+sequence\_length] label = scaled\_data[i+sequence\_length, 0] sequences.append(seq) labels.append(label)  sequences = np.array(sequences) labels = np.array(labels)  train\_size = int(0.7 \* len(sequences)) train\_x, test\_x = sequences[:train\_size], sequences[train\_size:]  train\_y, test\_y = labels[:train\_size], labels[train\_size:]    model = Sequential()  model.add(GRU(units=64, input\_shape=(train\_x.shape[1], train\_x.shape[2]), return\_sequences=False)) model.add(Dense(units=1))  model.compile(optimizer='adam', loss='mean\_squared\_error')  early\_stopping = EarlyStopping(monitor='val\_loss', patience=5, restore\_best\_weights=True)  model\_checkpoint = ModelCheckpoint('GRUPKL.h5', monitor='val\_loss', save\_best\_only=True)    history = model.fit( train\_x, train\_y, epochs=300, batch\_size=70, validation\_split=0.4, |
| callbacks=[early\_stopping, model\_checkpoint]  )  predictions = best\_model.predict(test\_x)  mae = mean\_absolute\_error(test\_y, predictions) mse = mean\_squared\_error(test\_y, predictions) rmse = np.sqrt(mse)    true\_beban = scaler.inverse\_transform(test\_y.reshape(-1, 1))[:,  0]  predicted\_beban = scaler.inverse\_transform(predictions)[:, 0] test\_dates = df.index[train\_size+sequence\_length:]    plt.figure(figsize=(10, 6))  plt.plot(test\_dates, true\_beban, label='True beban', color='blue')  plt.plot(test\_dates, predicted\_beban, label='Predicted beban', color='red')  plt.title('True vs Predicted beban') plt.xlabel('Date') plt.ylabel('beban') plt.legend() plt.grid(True) plt.show() |

### J. Kode Program *Flask*

|  |
| --- |
| from flask import Flask, jsonify import pandas as pd import numpy as np  from sklearn.preprocessing import MinMaxScaler import tensorflow as tf import pymysql    app = Flask(\_\_name\_\_)    # Load the model  best\_model = tf.keras.models.load\_model('fix\_GRU\_7030.h5')  def get\_data\_from\_database():  connection = pymysql.connect(host='YOUR\_HOST', user='YOUR\_USER', password='YOUR\_PASS', database='YOUR\_DB', cursorclass=pymysql.curso rs.DictCursor)  try:  with connection.cursor() as cursor:    sql = "SELECT tanggal, nama\_pembangkit, REPLACE(beban\_puncak, ',', '.') AS beban\_puncak FROM laporanbebanpuncak"  cursor.execute(sql) result = cursor.fetchall() |

|  |
| --- |
| df = pd.DataFrame(result)  df['tanggal'] = pd.to\_datetime(df['tanggal']) df['beban\_puncak'] = df['beban\_puncak'].astype( float)  df.set\_index('tanggal', inplace=True) finally: connection.close() return df  def generate\_predictions(df, pembangkit\_name): df\_pembangkit = df[df['nama\_pembangkit'] == pembangkit\_name].drop(columns='nama\_pembangkit') scaler = MinMaxScaler()  scaled\_data = scaler.fit\_transform(df\_pembangkit) sequence\_length = 3  sequences = np.array([scaled\_data[i:i + sequence\_length] for i in range(len(scaled\_data) - sequence\_length + 1)]) predictions = []  last\_date = df\_pembangkit.index.max() if len(sequences) > 0:  current\_sequence = sequences[-1] for i in range(4):  current\_sequence\_reshaped = current\_sequence.reshape(1, sequence\_length, scaled\_data.shape[1]) next\_value = best\_model.predict(current\_sequenc e\_reshaped)  predictions.append(next\_value) current\_sequence = np.vstack([current\_sequence [1:], next\_value])  predicted\_values = scaler.inverse\_transform(np.con catenate(predictions).reshape(1, df\_pembangkit.shape[1]))  dates = [last\_date + pd.DateOffset(months=i+1) for i in range(4)]  formatted\_dates = [date.strftime("%B %Y") for date in dates]  return list(zip(formatted\_dates, predicted\_values.tolist())) else:  return []    @app.route('/predictions', methods=['GET']) def predictions():  new\_df = get\_data\_from\_database()  pembangkit\_names = new\_df['nama\_pembangkit'].unique() all\_predictions = {} for pembangkit in pembangkit\_names: |
| pred\_values = generate\_predictions(new\_df, pembangkit)  all\_predictions[pembangkit] = pred\_values return jsonify(all\_predictions)  if \_\_name\_\_ == '\_\_main\_\_': app.run(debug=True) |

### K. Kode Program Larvel *Controller*

public function AdminDashboard()

{

$response = Http::get('http://127.0.0.1:5000/predictions');

if ($response->successful()) { $predictions = $response->json();

foreach ($predictions as $namaPembangkit => $prediksi) {

foreach ($prediksi as $key => $value) { $prediksi[$key][1][0] = round($value[1][0], 2);

}

$predictions[$namaPembangkit] = $prediksi;

}

} else {

$predictions = [];

}

return view('admin.admin\_dashboard', compact(

'predictions'

));

}

### L. Kode Program Larvel *View*

<!-- Prediksi Beban Puncak -->

<div class="col-12">

<div class="card">

<div class="filter">

<a class="icon" href="#" data-bstoggle="dropdown"><i class="bi bi-threedots"></i></a>

<ul class="dropdown-menu dropdownmenu-end dropdown-menu-arrow">

<li class="dropdown-header textstart">

<h6>Filter</h6>

</li>

<li><a class="dropdown-item" href="#">Today</a></li>

|  |
| --- |
| <li><a class="dropdown-item" href="#">This Month</a></li>  <li><a class="dropdown-item" href="#">This Year</a></li>  </ul>  </div>  <div class="card-body">  <h5 class="card-title">Prediksi Beban  Puncak <span>/This month</span></h5>  <!-- Line Chart -->  <div id="prediksiChart"></div>  <script>  document.addEventListener("DOMConte ntLoaded", () => { let predictions = {!!  json\_encode($predictions) !!};    let seriesData = []; let categories = []; for (let namaPembangkit in predictions) {  let prediksi = predictions[namaPembangkit];  let data = prediksi.map(item =>  ({  x: new Date(item[0]), y: item[1][0]  }));  seriesData.push({ name: namaPembangkit, data: data  });  categories = prediksi.map(item  => item[0]);  } new  ApexCharts(document.querySelector("#prediksiChart"),  {  series: seriesData, chart: { height: 350, type: 'area', toolbar: { show: true  }, }, markers: { size: 4 |
| }, fill: {  type: "gradient", gradient: { shadeIntensity: 1, opacityFrom: 0.3, opacityTo: 0.4, stops: [0, 90, 100]  } },  dataLabels: { enabled: false  }, stroke: { curve: 'smooth', width: 2  }, xaxis: {  type: 'datetime', categories: categories  }, tooltip: { x: {  format: 'MM/yy',  formatter: function(value)  {  return new  Date(value).toLocaleDateString('en-US', { month:  'short', year: 'numeric' });  }  }  }  }).render();  });  </script>  <!-- End Line Chart -->  </div>  </div>  </div><!-- End Prediksi Beban Puncak --> |

**BIOGRAPHIES OF AUTHORS**

|  |  |
| --- | --- |
|  | **Elza Safira Permatasari** is a final semester student of D4 Computer Science study program of Politeknik Negeri Semarang. During her lectures, she has gained a deep understanding of programming, networking, and in her final semester, she focused on web and artificial intelligence. Based on her interest in this field, she decided to take the final project on the topic of deep learning, specifically design and development of a web-based peak load prediction application for electricity usage at power plants using deep learning models: a case study of the "connect UP2DJTY" website. She can be contacted via email: elza.safira@polines.ac.id |
|  |  |
|  | **Prayitno, S.ST., M.T., Ph.D.** received the B.Sc. degree in computer science from the Politeknik Elektronika Negeri Surabaya, the M.Sc. degree in electrical engineering from the Institute of Technology Bandung, and the Ph.D. degree in computer science and information engineering from the Asia University, Taiwan. He is currently an assistant professor with the department of electrical engineering Politeknik Negeri Semarang (Polines). His research interests include deep learning, data mining, and artificial intelligence for education. He can be contacted at email: prayitno@polines.ac.id |
|  |  |
|  | **Wiktasari, S.T., M.Kom.** received the B.E. degree Universitas Islam Sultan Agung, the M.Sc. degree in Diponegoro University. she is currently an expert assistant with the department of electrical engineering Politeknik Negeri Semarang (Polines). Her interest is web development. she can be contacted at email:  wiktasari@polines.ac.id |