

LAPORAN TUGAS AKHIR
IMPLEMENTASI METODE *AUTOREGRESSIVE INTEGRATED MOVING*
***AVERAGE* (ARIMA) UNTUK PREDIKSI JUMLAH WISATAWAN OBJEK**
WISATA GUCI TEGAL DENGAN VISUALISASI *WEBSITE*
MENGGUNAKAN *FRAMEWORK STREAMLIT*

SKRIPSI

Diajukan sebagai Pedoman Pelaksanaan Penelitian Tugas Akhir pada Jurusan
Informatika

Fakultas Teknik Universitas Jenderal Soedirman



Disusun oleh:

YOSI JULIA UTAMI

H1D021022

KEMENTERIAN PENDIDIKAN TINGGI, SAINS, DAN TEKNOLOGI
UNIVERSITAS JENDERAL SOEDIRMAN
FAKULTAS TEKNIK
JURUSAN INFORMATIKA
PURWOKERTO

2025

LEMBAR PENGESAHAN

Laporan Tugas Akhir dengan Judul:
IMPLEMENTASI METODE *AUTOREGRESSIVE INTEGRATED MOVING
AVERAGE* (ARIMA) UNTUK PREDIKSI JUMLAH WISATAWAN OBJEK
WISATA GUCI TEGAL DENGAN VISUALISASI *WEBSITE*
MENGUNAKAN *FRAMEWORK STREAMLIT*

Disusun oleh:

Yosi Julia Utami

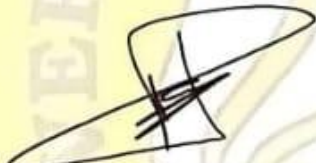
H1D021022

Diajukan untuk memenuhi salah satu persyaratan memperoleh gelar
Sarjana Komputer pada Jurusan Informatika Fakultas Teknik Universitas
Jenderal Soedirman

Diterima dan disetujui

Pada tanggal

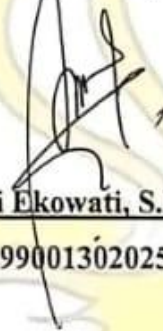
Pembimbing I



Ir. Ipung Permadi, S.Si., M.Cs.

NIP. 198311162008121005

Pembimbing II



Ir. Nur Alfi Ekowati, S.Kom., M.Sc.

NIP. 199001302025062006

Mengetahui,

Dekan Fakultas Teknik



Prof. Dr. Eng. Ir. Agus Marvoto, S.T., M.T., IPU., ASEAN Eng.

NIP. 197109202006041001

LEMBAR PERNYATAAN KEASLIAN SKRIPSI

Saya yang bertanda tangan di bawah ini:

Nama : Yosi Julia Utami

NIM : H1D021022

Program Studi : S1 Informatika

Dengan ini menyatakan bahwa Skripsi dengan judul "*IMPLEMENTASI METODE AUTOREGRESSIVE INTEGRATED MOVING AVERAGE (ARIMA) UNTUK PREDIKSI JUMLAH WISATAWAN OBJEK WISATA TEGAL DENGAN VISUALISASI WEBSITE MENGGUNAKAN FRAMEWORK STREAMLIT*" adalah benar-benar hasil karya saya sendiri dan bukan merupakan plagiat, saduran atau pengambilalihan tulisan atau pikiran orang lain yang saya akui sebagai tulisan atau pikiran saya kecuali yang secara tertulis diacu dalam karya ilmiah saya dengan menyebutkan sumber di dalam daftar pustaka.

Demikian pernyataan ini yang saya buat dengan sebenar-benarnya tanpa ada unsur paksaan dari pihak manapun.

Purbalingga, 20 November 2025

Yang membuat pernyataan,


10000
METERAI
TEMPEL
0B38BANX060429396 Julia Utami

H1D021022

HALAMAN PERSEMBAHAN

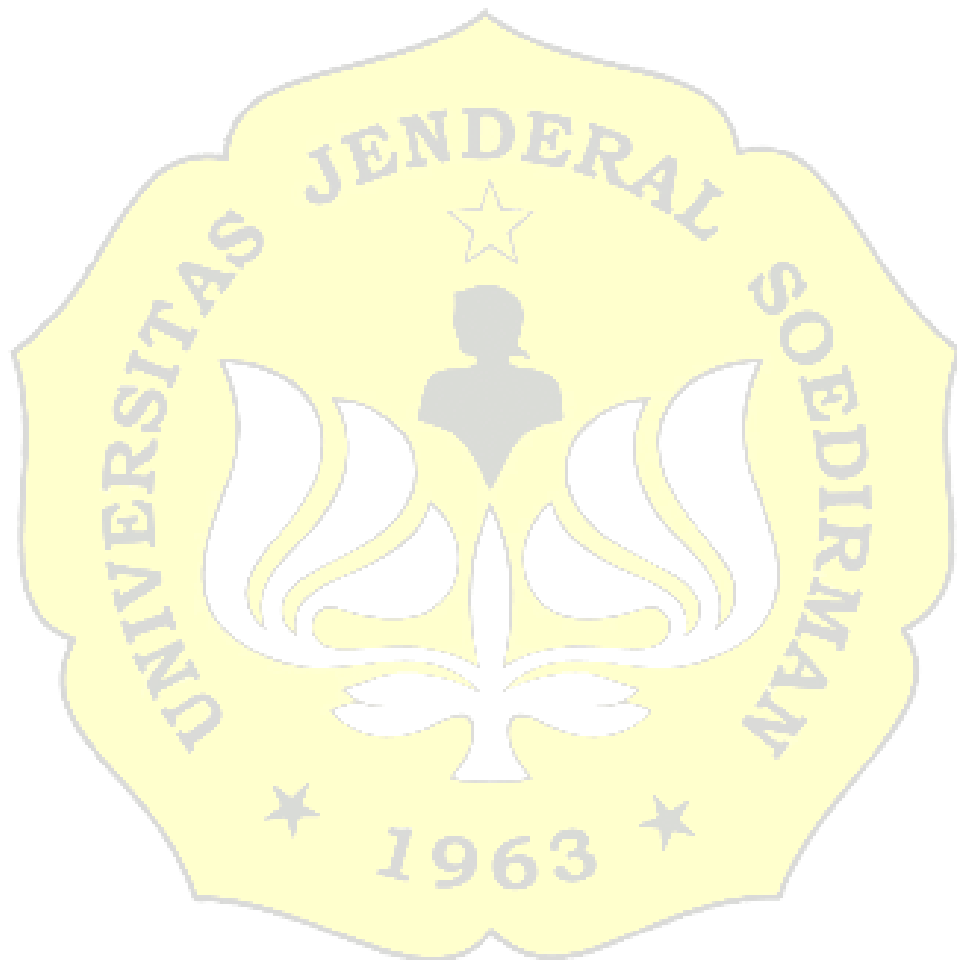
Segala puji dan syukur penulis panjatkan kepada Allah SWT atas limpahan rahmat, kesehatan, dan kesempatan yang diberikan hingga penulis dapat menyelesaikan skripsi ini. Perjalanan panjang ini telah menjadi pengalaman berharga yang penuh dengan pembelajaran dan makna. Walaupun jauh dari kata sempurna, penulis bangga telah menyelesaikan skripsi ini dengan baik. Karya sederhana ini penulis persembahkan kepada:

1. Orang tua, yang selalu menjadi sumber kekuatan, teladan, serta doa yang tak pernah putus. Terima kasih atas kasih sayang dan kesabaran yang tiada batas.
2. Bapak/Ibu Dosen Pembimbing, yang telah meluangkan waktu, tenaga, dan pikiran untuk memberikan arahan, masukan, serta motivasi selama proses penyusunan skripsi ini. Bimbingan dan kesabaran Bapak/Ibu menjadi bekal berharga bagi penulis dalam menyelesaikan karya ilmiah ini.
3. Orang tersayang, yang telah menemani dalam suka dan duka, berbagi tawa, cerita, serta motivasi hingga akhir perjalanan.
4. Semua pihak yang telah membantu, baik secara langsung maupun tidak langsung dalam proses penyusunan skripsi ini.

Semoga karya sederhana ini menjadi bentuk kecil dari rasa terima kasih dan kebanggaan atas semua doa, dukungan, dan kasih yang telah diberikan.

HALAMAN MOTTO

*“Jangan meminta izin untuk terbang.
Sayap itu milikmu dan langit bukan milik siapapun”*



KATA PENGANTAR

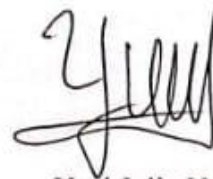
Puji syukur penulis panjatkan ke hadirat Allah SWT atas segala rahmat, taufik, dan hidayah-Nya, sehingga skripsi yang berjudul “Implementasi Metode *Autoregressive Integrated Moving Average* (ARIMA) untuk Prediksi Jumlah Wisatawan Objek Wisata Guci Tegal dengan Visualisasi *Website* Menggunakan *Framework Streamlit*” dapat diselesaikan sebagai salah satu syarat untuk menyelesaikan Pendidikan di jurusan Informatika Universitas Jenderal Soedirman.

Penulis menyadari adanya berbagai tantangan selama penyusunan, namun berkat doa, dukungan, dan bimbingan dari pihak, penelitian ini dapat terselesaikan dengan baik. Oleh karena itu, dengan segala kerendahan hati, penulis ingin menyampaikan rasa terima kasih yang sebesar-besarnya kepada:

1. Tuhan Yang Maha Esa.
2. Bapak Ir. Ipung Permadi, S.Si., M.Cs. selaku dosen pembimbing I dan Ibu Ir. Nur Alfi Ekowati, S.Kom., M.Sc. selaku dosen pembimbing II, yang telah memberikan bimbingan, nasihat, serta masukan yang baik selama proses penyusunan tugas akhir ini.
3. Kedua orang tua, atas doa, kasih sayang, dan dukungan yang tiada henti dalam setiap langkah penulis.
4. Keluarga, teman, dan semua pihak yang tidak dapat saya sebutkan satu-persatu yang bersangkutan dalam penyelesaian tugas akhir ini.

Penulis menyadari bahwa skripsi ini masih jauh dari kata sempurna, sehingga kritik dan saran yang membangun sangat diharapkan demi perbaikan ke depan. Semoga karya ini bermanfaat dan berkontribusi bagi pengembangan ilmu, khususnya dalam analisis data dan prediksi pariwisata.

Purbalingga, 20 November 2025



Yosi Julia Utami

ABSTRAK

IMPLEMENTASI METODE *AUTOREGRESSIVE INTEGRATED MOVING AVERAGE* (ARIMA) UNTUK PREDIKSI JUMLAH WISATAWAN OBJEK WISATA GUCI TEGAL DENGAN VISUALISASI *WEBSITE* MENGGUNAKAN *FRAMEWORK STREAMLIT*

Yosi Julia Utami
H1D021022

Pariwisata merupakan sektor penting yang berkontribusi terhadap perekonomian daerah, salah satunya adalah objek wisata Guci Tegal yang memiliki potensi besar dalam menarik wisatawan. Jumlah kunjungan wisatawan yang fluktuatif dan tidak menentu menyebabkan pengelola kesulitan dalam mengambil keputusan strategis karena belum adanya sistem prediksi yang terukur. Penelitian ini bertujuan mengimplementasikan metode *Autoregressive Integrated Moving Average* (ARIMA) untuk memprediksi jumlah wisatawan pada objek wisata Guci Tegal berdasarkan data historis, mengevaluasi tingkat akurasi hasil prediksi menggunakan metrik RMSE dan MAPE, serta menyajikan hasil prediksi dalam bentuk *dashboard* interaktif berbasis *website* menggunakan *framework Streamlit*. Sistem prediksi dikembangkan dengan pendekatan *prototyping* untuk mempermudah proses perancangan dan penyesuaian kebutuhan pengguna, serta diuji menggunakan metode *blackbox testing* guna memastikan setiap fungsi berjalan sesuai yang diharapkan. Dampak penerapan sistem ini adalah membantu pengelola dalam menganalisis tren wisatawan secara akurat sehingga mendukung pengambilan keputusan strategis dalam perencanaan promosi, infrastruktur, serta pengelolaan. Hasil penelitian menunjukkan bahwa model terbaik adalah ARIMA(5,0,0) dengan nilai RMSE sebesar 294,179 dan MAPE sebesar 10,71%. Hasil prediksi divisualisasikan melalui *dashboard* interaktif berbasis web dengan *framework Streamlit* yang informatif dan mudah dipahami oleh pengguna.

Kata kunci: ARIMA, *blackbox testing*, Guci Tegal, *prototyping*, *Streamlit*, wisatawan.

ABSTRACT

IMPLEMENTATION OF THE AUTOREGRESSIVE INTEGRATED MOVING AVERAGE (ARIMA) METHOD FOR PREDICTING THE NUMBER OF TOURISTS AT GUCI TOURISM OBJECT, TEGAL, WITH WEBSITE-BASED VISUALIZATION USING THE STREAMLIT FRAMEWORK

Yosi Julia Utami
H1D021022

Tourism is an important sector that contributes to regional economic development, one of which is the Guci Tegal tourist destination that has great potential to attract visitors. The fluctuating and unpredictable number of tourist arrivals causes difficulties for managers in making strategic decisions due to the absence of a measurable prediction system. This study aims to implement the Autoregressive Integrated Moving Average (ARIMA) method to predict the number of tourists at the Guci Tegal tourist destination based on historical data, evaluate the accuracy level of prediction results using RMSE and MAPE metrics, and present the forecasting results in an interactive web-based dashboard using the Streamlit framework. The prediction system was developed using the prototyping approach to facilitate the design process and meet user requirements, and tested using the Blackbox Testing method to ensure that each function works as expected. The impact of this system implementation is to assist managers in analyzing tourist trends accurately, thereby supporting strategic decision-making in promotion planning, infrastructure development, and tourism management. The results show that the best model is ARIMA(5,0,0) with an RMSE value of 294.179 and a MAPE value of 10.71%. The prediction results are visualized through an interactive web-based dashboard built with the Streamlit framework, which is informative and easy for users to understand.

Keywords: ARIMA, blackbox testing, Guci Tegal, prototyping, Streamlit, tourists.

DAFTAR ISI

LEMBAR PENGESAHAN	ii
LEMBAR PERNYATAAN KEASLIAN SKRIPSI	ii
HALAMAN PERSEMBAHAN	iv
HALAMAN MOTTO	v
KATA PENGANTAR	vi
ABSTRAK	vii
<i>ABSTRACT</i>	viii
DAFTAR ISI	ix
DAFTAR GAMBAR	xi
DAFTAR TABEL	xii
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	4
1.3 Batasan Masalah	5
1.4 Tujuan Penelitian	5
1.5 Manfaat Penelitian	5
BAB II TINJAUAN PUSTAKA	7
2.1 Pariwisata	7
2.2 Objek Wisata Guci Tegal	7
2.4 Metode ARIMA	8
2.4.1 <i>Autoregressive</i> (AR)	9
2.4.2 <i>Moving Average</i> (MA)	9
2.4.3 <i>Autoregressive Moving Average</i> (ARMA)	10
2.5 <i>Time Series</i>	11
2.6 Uji Diagnostik	12
2.7 <i>Python</i>	14
2.9 <i>Streamlit</i>	15
2.10 Penelitian Sejenis	15

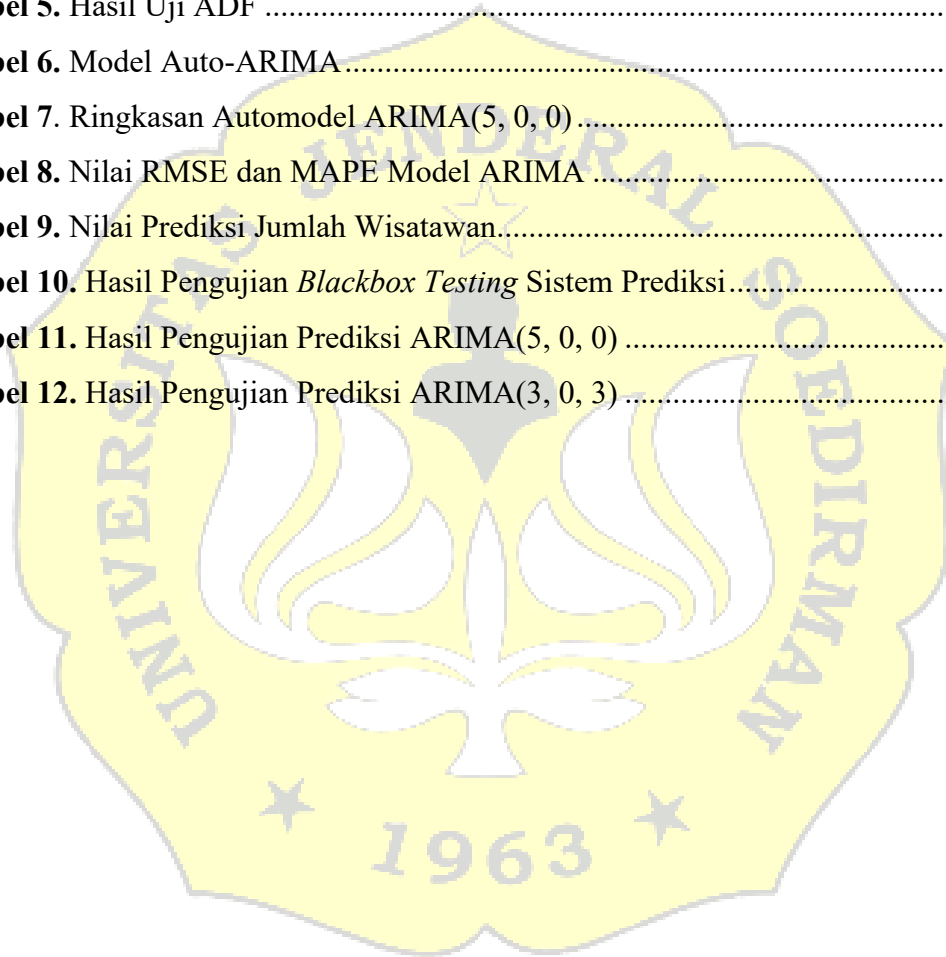
BAB III METODOLOGI PENELITIAN	19
3.1 Waktu Penelitian	19
3.2 Mekanisme Penelitian	19
3.3 Data dan Alat Penelitian	24
BAB VI HASIL DAN PEMBAHASAN	26
4.1 Kebutuhan Data dan Informasi Pendukung	26
4.2 Analisis Kebutuhan	26
4.3 Perancangan Sistem	27
4.4 ARIMA	35
4.5 Identifikasi Model	38
4.6 Pembuatan Model.....	44
4.7 Pemilihan Model Terbaik.....	47
4.8 Prediksi.....	55
4.9 Implementasi <i>Dashboard</i>	59
4.10 Pengujian <i>Website</i>	70
4.11 Pengujian Hasil Prediksi	72
BAB V PENUTUP.....	75
5.1 Kesimpulan	75
5.2 Saran.....	76
DAFTAR PUSTAKA.....	77

DAFTAR GAMBAR

Gambar 1. Tahapan Penelitian Pengembangan Sistem	19
Gambar 2. Tahapan metode ARIMA	20
Gambar 3. <i>Use Case Diagram</i> Sistem Prediksi Jumlah Wisatawan	28
Gambar 4. <i>Activity Diagram</i> Sistem Prediksi Jumlah Wisatawan	30
Gambar 5. <i>Sequence Diagram</i> Sistem Prediksi Jumlah Wisatawan	31
Gambar 6. Rancangan Antarmuka Halaman <i>Homepage</i>	32
Gambar 7. Rancangan Antarmuka Halaman Plot Data	33
Gambar 8. Rancangan Antarmuka Halaman <i>Performance</i>	33
Gambar 9. Rancangan Antarmuka Halaman Hasil Prediksi.....	34
Gambar 10. Pergerakan Jumlah Wisatawan Guci	39
Gambar 11. Grafik ACF Orde p	42
Gambar 12. Grafik PACF Orde q	43
Gambar 13. Grafik Perbandingan Data <i>Training</i> dan Data <i>Testing</i>	44
Gambar 14. Plot Residual Standar Model ARIMA(5, 0, 0)	49
Gambar 15. Histogram dan Estimasi Kepadatan Model ARIMA(5, 0, 0)	49
Gambar 16. Q-Q Plot Model ARIMA(5, 0, 0)	50
Gambar 17. Korelogram Model ARIMA(5, 0, 0).....	51
Gambar 18. Grafik Performa Model ARIMA(5, 0, 0).....	53
Gambar 19. Grafik Nilai Aktual dan Nilai Prediksi Jumlah Wisatawan.....	59
Gambar 20. Halaman <i>Homepage</i> Sistem Prediksi Jumlah Wisatawan	62
Gambar 21. Halaman Input Data Jumlah Wisatawan.....	64
Gambar 22. Halaman Plot Data Jumlah Wisatawan.....	64
Gambar 23. Halaman <i>Performance</i> Model ARIMA	66
Gambar 24. Halaman Hasil Prediksi Grafik Jumlah Wisatawan.....	69
Gambar 25. Halaman Hasil Prediksi Tabel Jumlah Wisatawan.....	69

DAFTAR TABEL

Tabel 1. Identifikasi Model <i>Time Series</i>	12
Tabel 2. Penelitian Sejenis	15
Tabel 3. Data Historis Jumlah Wisatawan	35
Tabel 4. Statistik Deskripsi Data Wisatawan	37
Tabel 5. Hasil Uji ADF	41
Tabel 6. Model Auto-ARIMA	47
Tabel 7. Ringkasan Automodel ARIMA(5, 0, 0)	48
Tabel 8. Nilai RMSE dan MAPE Model ARIMA	54
Tabel 9. Nilai Prediksi Jumlah Wisatawan.....	56
Tabel 10. Hasil Pengujian <i>Blackbox Testing</i> Sistem Prediksi	70
Tabel 11. Hasil Pengujian Prediksi ARIMA(5, 0, 0)	73
Tabel 12. Hasil Pengujian Prediksi ARIMA(3, 0, 3)	73



BAB 1

PENDAHULUAN

1.1 Latar Belakang

Jawa Tengah merupakan provinsi yang memiliki keindahan alam dan keanekaragaman budaya, serta memiliki wilayah yang sangat luas dan didukung dengan beragamnya sumber daya alam yang sangat berpotensi untuk diolah dan dimanfaatkan (Khamidah & Azakia, 2024). Salah satu sektor yang dapat dikembangkan untuk mendukung pembangunan daerah adalah pariwisata, yang melibatkan peran pemerintah, masyarakat, dan swasta (Husna, 2022). Salah satu destinasi wisata yang menjadi daya tarik di Jawa Tengah adalah objek wisata Guci di Tegal. Tempat ini dikenal dengan pemandian air panasnya yang bermanfaat bagi kesehatan serta panorama alam pegunungan yang indah, sehingga menjadi pilihan favorit bagi wisatawan yang ingin menikmati keindahan alam sekaligus merasakan manfaat terapi air panas.

Objek penelitian ini difokuskan pada objek wisata Guci Tegal, yang merupakan salah satu destinasi wisata unggulan di Jawa Tengah. Objek wisata ini menarik minat wisatawan lokal maupun mancanegara karena keunikan alamnya dan fasilitas yang tersedia. Menurut data dari Dinas Pariwisata Kabupaten Tegal, jumlah kunjungan wisatawan ke Guci Tegal mengalami fluktuasi dari tahun ke tahun yang dipengaruhi

oleh berbagai faktor seperti musim, promosi, dan kondisi infrastruktur. Sebagai contoh, pada tahun 2019, jumlah pengunjung mencapai 991.742 orang, menurun drastis pada tahun 2020 menjadi 386.352 orang akibat pandemi COVID-19, dan kembali meningkat pada tahun 2022 dengan total 536.614 pengunjung hingga Oktober (Khamidah & Azakia, 2024). Untuk mengoptimalkan potensi wisata ini, diperlukan upaya prediksi jumlah wisatawan yang akurat guna mendukung perencanaan dan pengambilan keputusan.

Prediksi jumlah kunjungan wisatawan menjadi salah satu aspek penting dalam perencanaan pengelolaan objek wisata. Objek wisata Guci mengalami ketidakpastian jumlah kunjungan wisatawan. Hal ini berdampak pada kesulitan dalam merencanakan strategi pemasaran, pengelolaan fasilitas, dan alokasi sumber daya. Oleh karena itu, suatu metode yang dapat memberikan prediksi jumlah wisatawan dengan tingkat akurasi yang tinggi diperlukan untuk mendukung pengambilan keputusan yang lebih baik.

Salah satu metode yang dapat digunakan untuk memprediksi jumlah kunjungan wisatawan adalah metode *Autoregressive Integrated Moving Average* (ARIMA). Metode ARIMA adalah salah satu metode yang mudah dan cepat dengan menggunakan pola data yang ada. Selain itu, metode ARIMA mempunyai tingkat kedekatan yang tinggi dan mempunyai kesalahan yang kecil karena proses perhitungan yang secara bertahap (Mardiyah dkk., 2021). Dengan memanfaatkan data

historis, metode ARIMA dapat mengidentifikasi pola dan tren yang terjadi dalam data, sehingga dapat menghasilkan prediksi yang lebih akurat. Implementasi metode ARIMA dalam memprediksi jumlah wisatawan di objek wisata Guci Tegal diharapkan dapat memberikan informasi yang berguna bagi pengelola dalam mengambil keputusan.

Penelitian sejenis yang mendukung penggunaan metode ARIMA telah dilakukan oleh beberapa peneliti sebelumnya. Penelitian oleh (Rianto & Yunis, 2021) berhasil memprediksi jumlah mahasiswa baru dengan model ARIMA, diperoleh empat model *time series* yang digunakan yakni model ARIMA terbaik yaitu ARIMA (2,1,1) dengan kesalahan atau nilai MAPE sebesar 7.066147 dengan akurasi sebesar 93% yang berarti model bisa digunakan untuk meramalkan jumlah mahasiswa baru dengan sangat akurat. Penelitian sejenis juga dilakukan oleh (Ramadhani dkk., 2022), dapat diketahui bahwa model ARIMA (0,1,1) memiliki nilai MAPE sebesar 6,23% dan model ARIMA (1,1,0) memiliki nilai *Mean Absolute Percentage Error* (MAPE) sebesar 6,38%. Dengan demikian, model terbaik diperoleh model ARIMA (0,1,1) karena memiliki nilai MAPE terkecil. Hal ini menunjukkan bahwa hasil prediksi yang didapatkan cukup akurat sehingga dapat menjadi referensi dalam penelitian serupa terkait prediksi jumlah wisatawan di objek Guci, Tegal.

Selain itu, visualisasi data prediksi dalam bentuk yang interaktif dan mudah dipahami menjadi kebutuhan penting bagi pengelola objek

wisata. Penelitian ini akan menggunakan *framework Streamlit* untuk membangun *website* interaktif dapat menjadi solusi yang efektif. *Streamlit* merupakan sebuah *framework open- source* yang bermanfaat untuk menciptakan aplikasi web yang dapat diakses menggunakan bahasa pemrograman *Python* (Azhar dkk., 2024). Dengan menggabungkan metode ARIMA dan visualisasi berbasis web, pengelola objek wisata dapat mengakses informasi prediksi secara *real-time* dan interaktif. Berdasarkan latar belakang permasalahan tersebut, maka akan dilakukan sebuah penelitian untuk mengimplementasikan metode ARIMA dalam prediksi jumlah wisatawan di objek wisata Guci Tegal.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang sudah disampaikan, terdapat beberapa rumusan masalah sebagai berikut:

- a. Bagaimana proses mengimplementasikan metode *Autoregressive Integrated Moving Average* (ARIMA) untuk memprediksi jumlah wisatawan yang berkunjung ke objek wisata Guci Tegal?
- b. Bagaimana hasil prediksi jumlah wisatawan yang berkunjung ke objek wisata Guci Tegal menggunakan metode ARIMA yang diperoleh?
- c. Bagaimana penerapan *website* menggunakan *framework Streamlit* untuk visualisasi hasil analisis prediksi jumlah wisatawan yang berkunjung ke objek wisata Guci Tegal dengan metode ARIMA?

1.3 Batasan Masalah

Batasan masalah pada penelitian ini di antaranya adalah:

- a. Penelitian ini menggunakan data sekunder yaitu data historis jumlah kunjungan wisatawan ke objek wisata Guci Tegal yang diperoleh dari Dinas Pariwisata Kabupaten Tegal.
- b. Data yang digunakan untuk pemodelan prediksi yang dibuat adalah dari tahun 2019-2024.
- c. Penelitian ini menggunakan bahasa pemrograman *Python*.

1.4 Tujuan Penelitian

Tujuan-tujuan yang ingin dicapai dari penelitian ini sebagai berikut:

- a. Mengimplementasikan metode ARIMA dalam memprediksi jumlah wisatawan yang berkunjung ke objek wisata Guci Tegal berdasarkan data historis.
- b. Mengevaluasi akurasi prediksi jumlah wisatawan yang dihasilkan oleh metode ARIMA.
- c. Menyajikan visualisasi hasil analisis ke dalam sebuah *website*.

1.5 Manfaat Penelitian

Berikut ini adalah beberapa manfaat yang dapat diperoleh dari penelitian ini, antara lain:

- a. Bagi Pemerintah Daerah

Hasil penelitian ini dapat dijadikan sebagai bahan pertimbangan dalam merumuskan kebijakan dan program pengembangan pariwisata

di Kabupaten Tegal menggunakan metode ARIMA.

b. Bagi Pengelola Objek Wisata

Memberikan informasi prediksi jumlah wisatawan yang akurat untuk mendukung perencanaan strategis, seperti pengelolaan fasilitas, promosi, dan alokasi sumber daya.

c. Bagi Mahasiswa

Berikut ini merupakan manfaat yang dapat dirasakan untuk mahasiswa yaitu:

- 1) Dapat mengaplikasikan pengetahuan tentang metode ARIMA untuk menganalisis dan memprediksi data *time series*, seperti jumlah kunjungan wisatawan.
- 2) Mampu menerapkan keterampilan dalam pembuatan visualisasi data menggunakan *Streamlit* yang dapat digunakan untuk menyajikan informasi prediksi secara efektif.

BAB II

TINJAUAN PUSTAKA

2.1 Pariwisata

Definisi pariwisata menurut Undang-Undang Nomor 10 Tahun 2009 (Undang-Undang No. 10 tahun 2009 tentang Pariwisata, 2009) “Pariwisata adalah berbagai macam kegiatan wisata dan didukung berbagai fasilitas serta layanan yang disediakan oleh masyarakat, pengusaha, pemerintah, dan pemerintah daerah”. Jadi pariwisata merupakan perjalanan yang dilakukan manusia ke daerah yang bukan merupakan tempat tinggalnya dalam waktu paling tidak satu malam dengan tujuan perjalanannya bukan untuk mencari nafkah, pendapatan atau penghidupan di tempat tujuan (Lumansik dkk., 2022).

Menurut Badan Pusat Statistik (BPS), pariwisata Indonesia mengalami peningkatan yang signifikan setelah pandemi COVID-19, dengan jumlah kunjungan wisatawan baik domestik maupun mancanegara terus bertambah. Oleh karena itu, peramalan jumlah wisatawan menjadi penting untuk mendukung kebijakan dan strategi pengelolaan pariwisata.

2.2 Objek Wisata Guci Tegal

Objek wisata adalah destinasi yang memiliki daya tarik bagi wisatawan, baik dari segi alam, budaya, sejarah, maupun rekreasi. Objek Wisata Guci adalah salah satu destinasi wisata terkenal di

Kabupaten Tegal, Jawa Tengah. Tempat ini dikenal dengan sumber air panas alaminya yang berasal dari pegunungan Slamet. Selain pemandian air panas, di Guci juga terdapat air terjun atau curug, bukit dan kebun stroberi. Wisatawan juga dapat berkeliling lokasi wisata dengan menunggangi kuda yang disewakan dengan tarif yang relatif murah. Fasilitas lain yang tidak kalah menarik adalah terdapat kolam renang (*waterpark*) dan *outbound*. Wisatawan juga dapat menyewa villa atau hotel di sekitar guci bagi mereka yang ingin menginap di sana (Maulana, 2021).

2.3 Forecasting/Prediksi

Definisi dari peramalan (*forecasting*) adalah suatu teknik untuk memperkirakan suatu nilai pada masa yang akan datang dengan memperhatikan masa lalu maupun data saat ini (Ramadhani dkk., 2022). Hal tersebut dapat dilakukann dengan menggunakan data historis dan proses kalkulasi untuk memprediksikan sebuah proyeksi atas kejadian di masa datang (Hilhami, 2020). Dalam konteks pariwisata, peramalan jumlah wisatawan berguna untuk mempersiapkan fasilitas, sumber daya manusia, dan strategi kedepannya.

2.4 Metode ARIMA

Metode ARIMA merupakan salah satu teknik analisis deret waktu yang banyak digunakan untuk peramalan data masa depan (Abidin dkk., 2022). Model ARIMA sepenuhnya tidak mempertimbangkan variabel independen dalam proses peramalan dan memanfaatkan data historis

serta nilai saat ini dari variabel dependen untuk menghasilkan prediksi jangka pendek yang lebih akurat. Metode ini sering disebut juga sebagai metode runtun waktu *Box-Jenkins*. Model ARIMA (p,d,q) pertama kali diperkenalkan oleh Box dan Jenkins (1970) (Fauzani & Rahmi, 2023).

2.4.1 Autoregressive (AR)

Model *Autoregressive* (AR) diperkenalkan pertama kali oleh Yule pada tahun 1962 dan kemudian dikembangkan oleh Walker pada tahun 1931 (Tasna Yunita, 2020). Proses AR berguna dalam menggambarkan situasi dimana nilai sekarang dari deret waktu tergantung pada nilai-nilai sebelumnya yang bermacam-macam (Wei, 2006). Bentuk umum model AR (p) dapat dilihat pada persamaan 1.

$$Z_t = \phi_1 Z_{t-1} + \dots + \phi_p Z_{t-p} + a_t \quad (1)$$

Keterangan:

Z_t : nilai variable pada waktu ke- t

$Z_t, Z_{t-1}, Z_{t-2}, \dots, Z_{t-p}$: nilai masa lalu dari *time series* yang bersangkutan pada waktu $t, t-1, t-2, \dots, t-p$

ϕ_i : koefisien regresi, $i: 1, 2, 3, \dots, p$

a_t : nilai *error* pada waktu ke- t

2.4.2 Moving Average (MA)

Moving Average (MA) menyatakan bahwa model yang

memprediksi suatu nilai berdasarkan rata-rata kesalahan dari periode sebelumnya. Adapun bentuk umum model MA (q) bisa dilihat pada persamaan 2.

$$Z_t = a_t + \theta_1 a_{t-1} + \theta_2 a_{t-2} + \dots + \theta_q a_{t-q} \quad (2)$$

Keterangan:

Z_t : nilai variabel pada waktu ke- t

$a_t, a_{t-1}, a_{t-2}, a_{t-p}$: nilai-nilai dari *error* pada waktu $t, t-1, t-2, t-p$

θ_i : parameter model MA ke- i

q : orde MA

2.4.3 Autoregressive Moving Average (ARMA)

Autoregressive Moving Average (ARMA) adalah model yang menggabungkan AR dan MA untuk meningkatkan akurasi prediksi. Berikut bentuk model ARMA (p,q) dapat dilihat pada persamaan 3.

$$Z_t = \phi_1 Z_{t-1} + \dots + \phi_p Z_{t-p} + a_t + \theta_1 a_{t-1} + \theta_2 a_{t-2} + \dots + \theta_q a_{t-q} \quad (3)$$

Keterangan:

Z_t : nilai variable pada waktu ke- t

ϕ_i : koefisien regresi, $i: 1, 2, 3, \dots, p$

p : orde AR

q : orde MA

θ_i : parameter model MA ke- i

a_t : nilai *error* pada waktu ke- t

$a_t, a_{t-1}, a_{t-2}, a_{t-p}$: nilai-nilai dari *error* pada waktu $t, t-1, t-2, t-p$

2.5 Time Series

Time series atau deret waktu adalah sekumpulan data yang dikumpulkan dalam rentang waktu tertentu dengan interval yang tetap, seperti harian, mingguan, bulanan, atau tahunan. Analisis deret waktu bertujuan untuk memahami pola data masa lalu guna memprediksi nilai di masa depan. Langkah penting dalam memilih suatu metode *time series* yang tepat adalah mempertimbangkan jenis pola data, sehingga metode yang paling cocok dengan pola tersebut dapat di uji dan diterapkan secara optimal. Pada pola trend, terdapat 3 jenis bentuk trend yaitu ada *uptrend*, *downtrend*, dan *sideway trend* (Heru Widiyanto dkk., 2023).

Secara umum, identifikasi model *time series* dapat dilakukan dengan melihat plot ACF dan plot PACF. Sifat-sifat model deret waktu dapat dilihat berdasarkan bentuk atau pola plot ACF dan PACF (Seymour dkk., 2016). Apabila ACF menurun perlahan sementara PACF terputus setelah beberapa lag maka menggunakan AR (p). Sebaliknya, jika ACF memiliki terputus setelah beberapa lag sedangkan PACF menurun secara perlahan, maka menggunakan MA (q). Namun, jika keduanya menunjukkan pola menurun secara eksponensial maka menggunakan ARMA (p,q). Sementara itu, jika ACF tidak menemukan

pola yang jelas, maka data tidak stasioner sehingga perlu dilakukan *differencing* sebelum menerapkan ARIMA. Identifikasi model *time series* bisa dilihat pada Tabel 1.

Tabel 1. Identifikasi Model *Time Series*

Model	ACF	PACF
AR (p)	Menurun secara eksponensial mendekati nol	Signifikan pada semua lag p
MA (q)	Signifikan pada seluruh lag p	Menurun secara eksponensial hingga nol
ARMA (p,q)	Menurun secara eksponensial hingga nol	Menurun secara eksponensial hingga nol
ARIMA (p,d,q)	Menurun secara eksponensial hingga nol dengan proses <i>differencing</i> (pembedaan)	Menurun secara eksponensial hingga nol dengan proses <i>differencing</i> (pembedaan)

2.6 Uji Diagnostik

Untuk menentukan model peramalan terbaik, model tersebut harus memenuhi ketiga uji sebelumnya. Pemilihan model terbaik didasarkan pada tingkat terkecil, yang diukur menggunakan *Mean Absolute Percentage Error* (MAPE) dan *Root Mean Square Error* (RMSE) yang memiliki nilai kesalahan terkecil (Hendrik Hidayatullah dkk., 2023).

a. *Mean Absolute Percentage Error* (MAPE)

MAPE adalah ukuran statistik tentang akurasi prediksi pada

metode peramalan. MAPE dinyatakan sebagai presentase kesalahan rata-rata prediksi dalam hubungan dengan nilai aktual. MAPE dapat dihitung pada persamaan 4 (Wei, 2006).

$$MAPE = \frac{\sum_{t=1}^n \frac{|z_t - \hat{z}_t|}{z_t}}{n} \times 100 \quad (4)$$

Keterangan:

z_t : *real value*

\hat{z} : *forecast*

n : periode waktu

b. *Root Mean Square Error (RMSE)*

RMSE adalah mengukur rata-rata nilai kesalahan antara nilai yang diprediksi dan nilai aktual dalam satuan yang sama dengan data yang diukur, sehingga memberikan gambaran tentang seberapa besar kesalahan prediksi dalam skala aslinya. RMSE dapat dihitung pada persamaan 5 (Wei, 2006).

$$RMSE = \frac{\sum_{t=1}^n (z_t - \hat{z}_t)^2}{n} \quad (5)$$

Keterangan:

z_t : *real value*

\hat{z} : *forecast*

n : periode waktu

2.7 Python

Python adalah bahasa pemrograman yang populer dalam analisis data dan pemodelan statistik, termasuk implementasi model ARIMA. *Python* merupakan salah satu bahasa pemrograman yang populer pada suatu bidang *data science*, *machine learning*, *big data*, *data mining*, hingga *deep learning* (Mangilang, A, 2024). *Python* juga memiliki ekosistem yang luas untuk pengembangan aplikasi web, termasuk *framework Streamlit* yang memungkinkan pembuatan aplikasi web dengan cepat dan mudah. Keunggulan *python* terletak pada kemampuannya menghasilkan hasil yang lebih akurat dan efisien dibandingkan dengan metode manual (Maulida Surbakti dkk., 2024).

2.8 Use Case Diagram

Use Case Diagram yaitu diagram yang digunakan untuk menggambarkan hubungan antara sistem dengan aktor. Secara umum *use case diagram* digunakan untuk memberikan gambaran akan fungsi dari setiap sistem dan juga untuk mengetahui hak akses dalam menggunakan sistem tersebut (Setiaji, S., & Sastra, 2021).

2.9 Visualisasi Website

Visualisasi data melalui *website* memungkinkan penyajian informasi yang interaktif dan mudah dipahami oleh pengguna. Dalam konteks prediksi jumlah wisatawan, visualisasi dapat berupa grafik tren atau plot prediksi. Visualisasi data juga dapat mengubah kumpulan-kumpulan data tersebut menjadi sebuah data yang lebih sederhana

(Madyatmadja dkk., 2021).

2.9 *Streamlit*

Streamlit adalah *library python* yang berfungsi sebagai aplikasi *website* interaktif bagi pengguna *python* (Christian & Idrus, 2023). Penelitian ini akan diterapkan dengan menggunakan *framework* web yang disebut *Streamlit*. *Streamlit* adalah *library Python open-source* dan gratis untuk pengembangan aplikasi web, karena kode-kodenya juga tersedia untuk umum di *Github*, para *developer* dapat dengan mudah menyesuaikan kode-kode tersebut untuk mengubah dan memperbarui apapun yang diinginkan (Koh dkk., 2021).

2.10 Penelitian Sejenis

Penelitian ini mengacu pada beberapa referensi dari penelitian-penelitian sebelumnya yang memiliki topik serupa. Referensi tersebut digunakan sebagai bahan pertimbangan dan perbandingan untuk memperkuat landasan serta metodologi dalam penelitian ini. Adapun penelitian yang digunakan disajikan pada Tabel 2.

Tabel 2. Penelitian Sejenis

No	Peneliti	Judul	Hasil	Metode	Perbedaan
1.	Rianto & Yunis (2021)	Runtun Waktu Untuk Memprediksi Jumlah Mahasiswa Baru Dengan Model ARIMA	Diperoleh empat model <i>time series</i> yang digunakan yakni model ARIMA terbaik yaitu ARIMA (2,1,1) dengan kesalahan atau nilai MAPE sebesar	ARIMA	a. Topik yang diangkat b. Tanpa visualisasi data

No	Peneliti	Judul	Hasil	Metode	Perbedaan
			7.066147 dengan akurasi sebesar 93% yang berarti model bisa digunakan untuk meramalkan jumlah mahasiswa baru dengan sangat akurat.		
2.	Ramadhani dkk (2022)	Peramalan Jumlah Kunjungan Wisatawan Mancanegara Ke Indonesia Menggunakan <i>Autoregressive Integrated Moving Average</i> (ARIMA)	Hasil analisis dapat diketahui bahwa model ARIMA (0,1,1) memiliki nilai MAPE sebesar 6,23% dan model ARIMA (1,1,0) memiliki nilai MAPE sebesar 6,38%. Dengan demikian, model terbaik diperoleh model ARIMA (0,1,1) karena memiliki nilai MAPE terkecil.	ARIMA	a. Tanpa visualisasi data
3.	Daratullaila & Sari (2024)	Prediksi Jumlah Kejahatan di Indonesia dengan Metode ARIMA	Diperoleh model terbaik untuk meramalkan jumlah kejahatan di Indonesia menggunakan metode ARIMA, yaitu model ARIMA (3,1,3) dengan MAPE terkecil, yaitu 19%.	ARIMA	a. Topik yang digunakan b. Tanpa Visualisasi data
4.	Mardiyah	Analisis	Hasil penelitian	ARIMA	a. Topik

No	Peneliti	Judul	Hasil	Metode	Perbedaan
	dkk (2021)	Prediksi Jumlah Penduduk di Kota Pasuruan Menggunakan Metode ARIMA	mengenai prediksi jumlah penduduk Kota Pasuruan menggunakan metode ARIMA menunjuk-kan bahwa model terbaik untuk melakukan prediksi adalah ARIMA (1,1,1) dengan MSE sebesar 10.542507 dan MAPE sebesar 1,52%.		yang diguna-kan b. Tanpa visuali-sasi data
5.	Mukhamad Riziq Maulana (2021)	Peramalan (<i>forecasting</i>) jumlah wisatawan objek wisata guci tegal dengan metode dekomposisi dan <i>Winter's Exponential Smoothing</i>	Hasil penelitian ini menyimpulkan bahwa hasil perhitungan nilai <i>error</i> dari kedua metode, metode dekomposisi menghasilkan nilai MAPE sebesar 22,48%, sedangkan metode <i>winter exponential smoothing</i> menghasilkan MAPE sebesar 62,97%.	Dekomposisi dan <i>Winters Exponential Smoothing</i>	a. Topik yang diguna-kan b. Tanpa Visuali-sasi data
6.	Agus Dwi Milniadi dkk (2023)	Analisis Perbandingan Model ARIMA dan LSTM dalam Peramalan Harga Penutupan	Hasil penelitian ini membanding-kan metode ARIMA dengan data <i>train-test</i> 90:10, 80:20, dan 70:30. Hasilnya,	ARIMA dan LSTM	a. Topik yang diguna-kan b. Metode yang diguna-kan c. Tanpa

No	Peneliti	Judul	Hasil	Metode	Perbedaan
		Saham (Studi kasus: 6 Kriteria Kategori Saham Menurut Peter Lynch)	pembagian 90:10 memberikan nilai RMSE 198,62 dan MAPE 1,79%, yang lebih baik dibandingkan 80:20 dan 70:30. Jadi, data <i>training</i> lebih besar (90%) menghasilkan model ARIMA yang lebih akurat dalam peramalan dibanding pembagian yang lebih kecil.		Visualisasi data
6.	Yosi Julia Utami (2025)	Implementasi Metode <i>Autoregressive Integrated Moving Average</i> (ARIMA) untuk Prediksi Jumlah Wisatawan Objek Wisata Guci Tegal dengan Visualisasi <i>Website</i> menggunakan <i>Framework Streamlit</i>	-	ARIMA	a. Metode yang digunakan b. Topik yang diangkat c. Visualisasi dalam <i>website</i>

BAB III

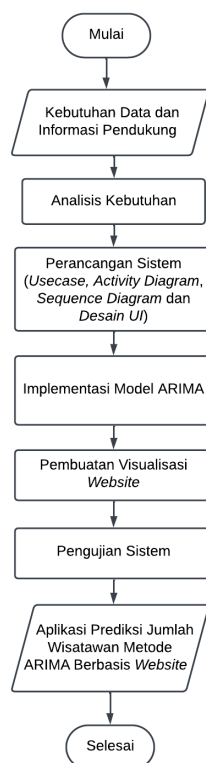
METODOLOGI PENELITIAN

3.1 Waktu Penelitian

Penelitian dilaksanakan pada bulan Juli 2025 sampai dengan Oktober 2025, sesuai dengan jadwal yang telah direncanakan untuk menyelesaikan seluruh tahapan penelitian.

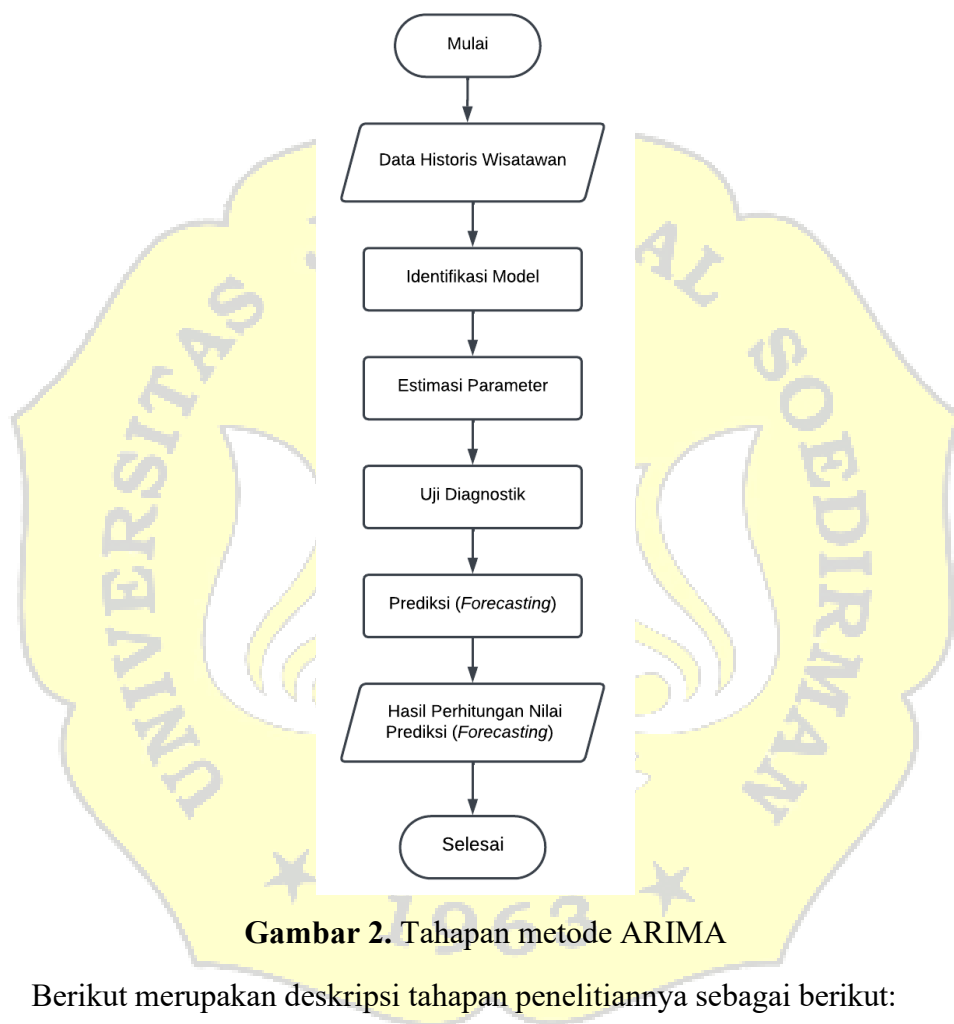
3.2 Mekanisme Penelitian

Penelitian ini menggunakan pendekatan pengembangan sistem yang melibatkan beberapa tahapan utama. Berikut ini adalah gambaran tahapan penelitian pengembangan sistem yang ditunjukkan pada Gambar 1.



Gambar 1. Tahapan Penelitian Pengembangan Sistem

Selain tahapan pengembangan sistem, penelitian ini juga menggunakan metode statistik ARIMA (*Autoregressive Integrated Moving Average*). Berikut adalah gambaran tahapan metode ARIMA ditunjukkan pada Gambar 2.



1) Kebutuhan Data dan Informasi Pendukung

Tahapan ini berfokus pada pengumpulan kebutuhan awal penelitian berupa data dan informasi pendukung. Data yang digunakan adalah data historis jumlah kunjungan wisatawan objek wisata Guci

Tegal periode 2019-2024, yang diperoleh dari Dinas Pariwisata Kabupaten Tegal. Selain itu, dilakukan studi literatur dan wawancara untuk memperkuat landasan teori serta memastikan data relevan dengan tujuan penelitian.

2) Analisis Kebutuhan

Setelah data dan informasi pendukung terkumpul, dilakukan analisis kebutuhan sistem. Tahapan ini bertujuan untuk mendefinisikan kebutuhan pengguna dan sistem, serta fungsi-fungsi utama yang tersedia pada aplikasi prediksi wisatawan. Hasilnya berupa dokumen kebutuhan sistem yang menjadi acuan pada tahap perancangan.

3) Perancangan Sistem

Pada tahap ini, perancangan sistem dilakukan dengan menggunakan berbagai diagram pemodelan, seperti *Use Case Diagram*, *Activity Diagram*, *Sequence Diagram* dan perancangan antarmuka (*UI Design*). Tujuan tahapan ini adalah untuk menggambarkan alur kerja sistem, interaksi antara aktor, serta rancangan antarmuka *website* yang akan digunakan oleh pengguna.

4) Implementasi Model ARIMA

Setelah melakukan persiapan data, langkah selanjutnya menganalisis data tersebut menggunakan metode ARIMA. Adapun langkah-langkahnya adalah sebagai berikut:

a. Identifikasi Model

Dalam mengidentifikasi model pada metode ARIMA dilakukan dengan menguji stasioneritas data menggunakan metode uji *Augmented Dickey-Fuller* (ADF). Pengujian ini dilakukan untuk memastikan data *time-series* yang digunakan telah stasioner. Hipotesis uji ADF:

H0 : Data tidak stasioner

H1 : Data stasioner

Data yang belum stasioner dalam ragam akan ditransformasi, sedangkan data yang belum stasioner dalam rata-rata akan dilakukan *differencing* (Rizki, 2023).

b. Estimasi Parameter

Setelah model teridentifikasi, langkah selanjutnya adalah mencari estimasi terbaik untuk parameter-parameter dalam model ARIMA. Tahap ini melibatkan proses dengan memilih metode ARIMA dengan parameter yang memberikan hasil terbaik berdasarkan kriteria seperti *Akaike Information Criterion* (AIC).

Berikut ini merupakan nilai AIC dapat dihitung dengan menggunakan persamaan 6.

$$AIC = 2k - 2\log(L) \quad (6)$$

Keterangan:

k : jumlah parameter yang digunakan dalam model

ARIMA

L : nilai *log-likelihood* dari model ARIMA

c. Uji Diagnostik

Uji diagnostik yaitu memeriksa atau menguji model telah dispesifikasi secara benar atau telah dipilih p , d dan q yang benar. Langkah-langkahnya meliputi uji residual untuk memastikan residual model berdistribusi normal, tidak memiliki autokorelasi dan mengukur model menggunakan metrik seperti *Mean Absolute Percentage Error* (MAPE) dan *Root Mean Squared Error* (RMSE).

d. Prediksi

Pada tahap ini prediksi jumlah kunjungan wisatawan untuk periode waktu selanjutnya dilakukan dengan model ARIMA terbaik yang dihasilkan pada tahapan sebelumnya. Hasil prediksi ini kemudian akan ditampilkan pada sebuah *website* menggunakan *framework Streamlit*.

e. Hasil Perhitungan Nilai Prediksi

Hasil prediksi ditampilkan dalam bentuk angka dan divisualisasikan dalam grafik. Nilai ini menjadi dasar untuk pembuatan sistem prediksi berbasis *website*, sehingga dapat digunakan sebagai informasi pendukung pengambilan keputusan oleh pihak terkait.

5) Pembuatan Visualisasi *Website*

Hasil prediksi jumlah wisatawan akan ditampilkan dalam bentuk

dashboard berbasis *website*. Tahap awal dari proses ini dimulai dengan pembuatan diagram *use case* serta rancangan antarmuka *dashboard* menggunakan alat bantu seperti *Figma*. Rancangan tersebut selanjutnya diimplementasikan ke dalam *website* menggunakan *framework Streamlit*.

6) Pengujian Sistem

Setelah model dan *website* selesai diimplementasikan, tahap berikutnya adalah pengujian sistem. Pengujian fungsionalitas aplikasi dilakukan menggunakan metode *Blackbox Testing*.

7) Aplikasi Prediksi Jumlah Wisatawan

Hasil dari seluruh tahapan adalah sebuah aplikasi berbasis *website* yang mampu menampilkan hasil prediksi jumlah wisatawan Guci Tegal dengan metode ARIMA. Aplikasi ini dilengkapi visualisasi data berupa grafik dan tabel untuk mempermudah interpretasi hasil.

3.3 Data dan Alat Penelitian

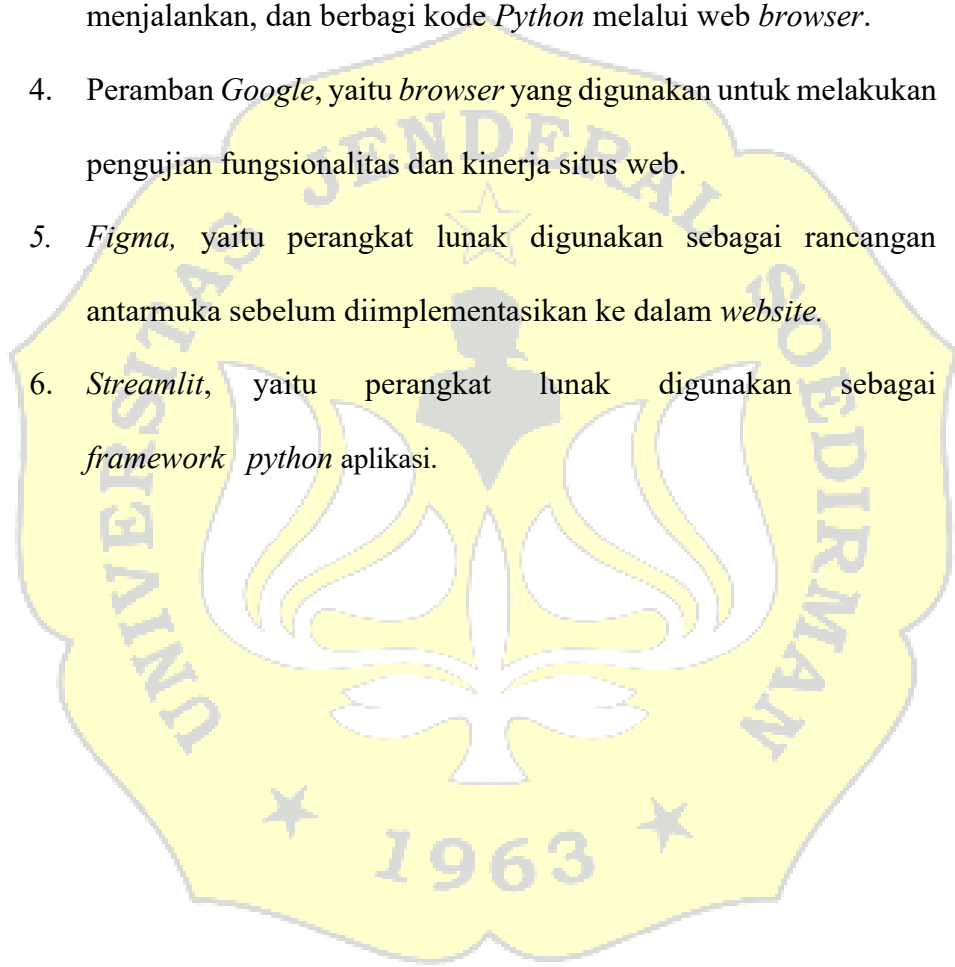
Penelitian ini menggunakan perangkat keras laptop ASUS *VivoBook X415EA_A416EA* dengan spesifikasi pemroses *11th Gen Intel(R) Core(TM) i3- 1115G4*, RAM 4 GB, dan *Windows 11 Home Single Language 64-bit*.

Dalam penelitian ini beberapa perangkat lunak yang digunakan adalah sebagai berikut:

1. *Microsoft Word Home and Student 2021*, yaitu perangkat lunak yang digunakan untuk membantu pengolahan dan penulisan

laporan.

2. *Lucidchart*, yaitu perangkat lunak yang digunakan untuk membuat visualisasi seperti *flowchart*, diagram metodologi penelitian, dan lain-lain.
3. *Google Colab*, yaitu *platform* berbasis *cloud* untuk menulis, menjalankan, dan berbagi kode *Python* melalui web *browser*.
4. Peramban *Google*, yaitu *browser* yang digunakan untuk melakukan pengujian fungsionalitas dan kinerja situs web.
5. *Figma*, yaitu perangkat lunak digunakan sebagai rancangan antarmuka sebelum diimplementasikan ke dalam *website*.
6. *Streamlit*, yaitu perangkat lunak digunakan sebagai *framework python* aplikasi.



BAB IV

HASIL DAN PEMBAHASAN

4.1 Kebutuhan Data dan Informasi Pendukung

Tahap awal penelitian ini adalah melakukan pengumpulan data. Data diperoleh dari Dinas Pariwisata Kabupaten Tegal melalui wawancara dan observasi kunjungan wisatawan ke objek wisata Guci Tegal. *Dataset* mencakup jumlah wisatawan harian periode 2019-2024 dengan kategori dewasa, anak, dan wisatawan mancanegara. Selain itu, informasi tambahan diperoleh dari observasi lapangan yang menunjukkan bahwa fluktuasi jumlah wisatawan dipengaruhi oleh faktor musim liburan, kondisi cuaca, dan infrastruktur.

4.2 Analisis Kebutuhan

Analisis kebutuhan bertujuan untuk mendefinisikan apa saja yang harus disediakan oleh aplikasi prediksi wisatawan agar sesuai dengan tujuan penelitian. Analisis ini dibagi menjadi dua bagian, yaitu kebutuhan fungsional dan non-fungsional.

a. Kebutuhan Fungsional

Kebutuhan fungsional menjelaskan fungsi utama yang harus tersedia pada sistem, yaitu:

- 1) Sistem dapat menampilkan data historis wisatawan dalam bentuk grafik.
- 2) Sistem dapat melakukan prediksi jumlah wisatawan menggunakan metode ARIMA.

- 3) Sistem dapat menampilkan hasil prediksi dalam bentuk tabel dan grafik.
- 4) Sistem dapat menampilkan nilai evaluasi akurasi model (MAPE dan RMSE).
- 5) Sistem menyediakan menu navigasi utama berupa *Home*, *Plot Data*, *Performance*, dan *Prediksi*.

b. Kebutuhan Non-Fungsional

Kebutuhan non-fungsional menjelaskan aspek pendukung agar sistem berjalan dengan baik, meliputi:

- 1) *Usability*, Antarmuka sistem sederhana, responsif, dan mudah dipahami.
- 2) *Performance*, Sistem mampu menampilkan hasil prediksi dengan waktu respon kurang dari 5 detik.
- 3) *Compatibility*, Sistem dapat dijalankan pada *browser modern* seperti *Google Chrome*.
- 4) *Reliability*, Sistem dapat berjalan stabil tanpa *error* pada kondisi normal.
- 5) *Maintainability*, Sistem mudah dikembangkan lebih lanjut, misalnya dengan menambahkan metode prediksi lain.

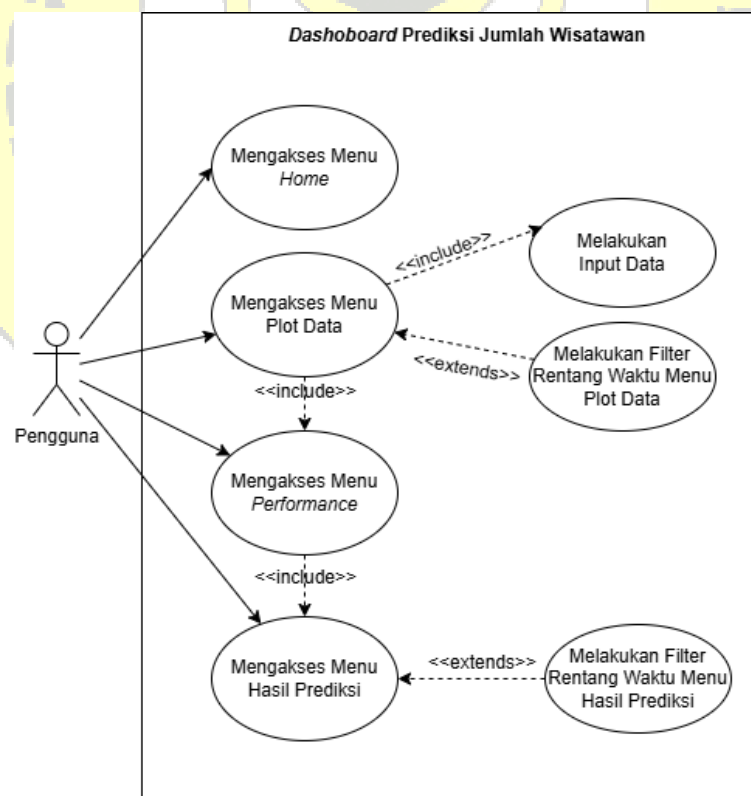
1.3 Perancangan Sistem

Peneliti mengembangkan *dashboard* untuk memvisualisasikan hasil analisis dan prediksi yang diperoleh pada tahap sebelumnya. Proses pengembangan diawali dengan pembuatan *use case*, *activity*

diagram, sequence diagram, serta perancangan antarmuka dashboard menggunakan figma. Desain antarmuka yang telah dibuat diimplementasikan peneliti ke dalam website menggunakan framework streamlit. Selanjutnya, peneliti menguji website tersebut dengan metode blackbox testing memastikan fungsionalitas dan keandalan sistem.

a. *Use Case Diagram*

Use case diagram digunakan untuk menggambarkan interaksi antara pengguna dengan sistem atau website yang dikembangkan. Gambar 3 memperlihatkan rancangan use case diagram dari website prediksi jumlah wisatawan yang akan dibangun.

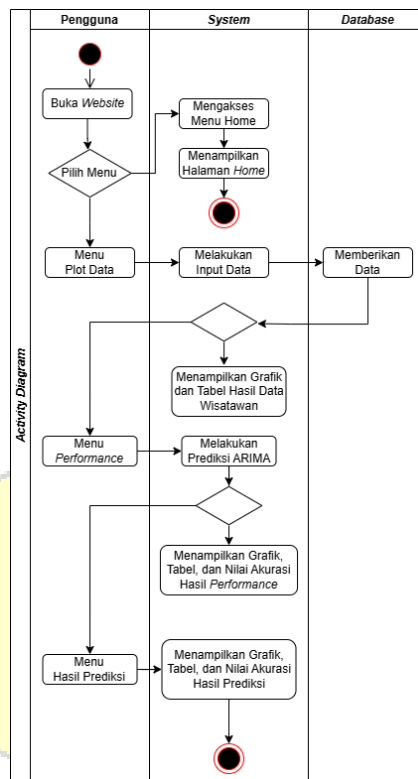


Gambar 3. *Use Case Diagram* Sistem Prediksi Jumlah Wisatawan

Gambar 3 merupakan *use case diagram* dari *website* prediksi jumlah wisatawan yang sedang dirancang. *Website* ini hanya memiliki satu aktor yaitu pengguna, yang dapat melakukan aksi untuk mengakses seluruh menu dan fitur filter yang tersedia pada *website*, seperti mengakses menu *home*, plot data, performa model, dan hasil prediksi.

b. *Activity Diagram*

Activity diagram digunakan untuk menggambarkan alur aktivitas pengguna ketika berinteraksi dengan sistem. Diagram ini memperlihatkan langkah-langkah proses dari mulai pengguna memilih menu hingga sistem menampilkan *output*. Pada aplikasi prediksi jumlah wisatawan, *activity diagram* mencakup aktivitas seperti membuka menu *Home*, melihat data historis pada menu *Plot Data*, menampilkan performa model pada menu *Performance*, serta menghasilkan *output* prediksi pada menu *Prediksi*. Gambar 4 memperlihatkan rancangan *activity diagram* dari *website* prediksi jumlah wisatawan yang akan dibangun.

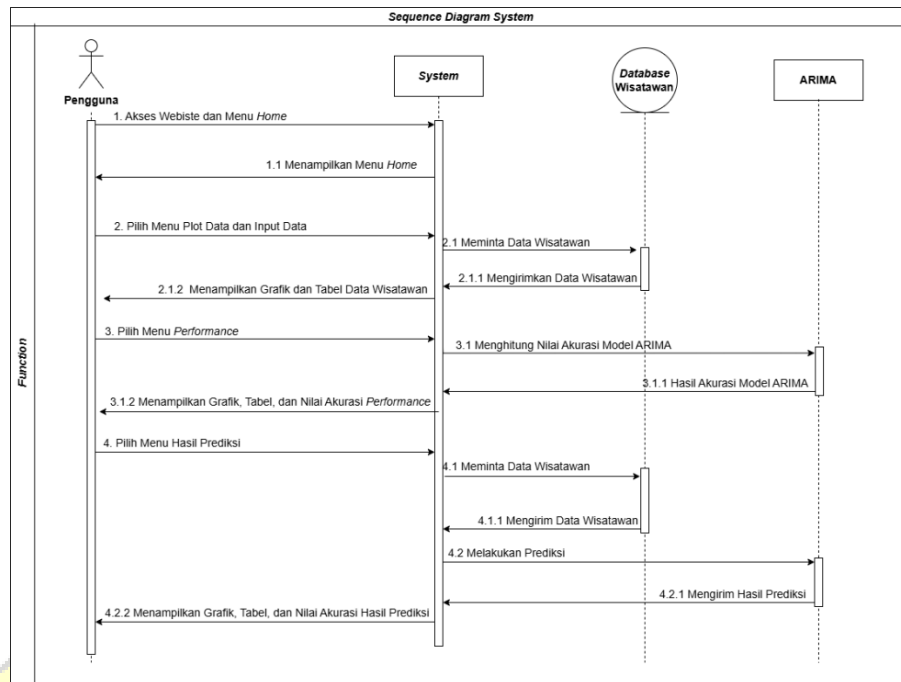


Gambar 4. Activity Diagram Sistem Prediksi Jumlah Wisatawan

Gambar 4 merupakan rancangan *activity diagram* yang memperlihatkan alur aktivitas antara pengguna, *system*, dan *database* yang saling terhubung dalam menampilkan data historis, evaluasi model ARIMA, hingga hasil prediksi jumlah wisatawan.

c. Sequence Diagram

Sequence diagram digunakan untuk menggambarkan interaksi antar objek atau aktor dengan sistem secara berurutan berdasarkan waktu. Pada sistem ini, *sequence diagram* digunakan untuk menjelaskan pengguna mengakses setiap menu dan sistem memproses permintaan tersebut hingga menghasilkan *output*. Gambar 5 memperlihatkan rancangan *sequence diagram* dari *website* prediksi jumlah wisatawan yang akan dibangun.



Gambar 5. *Sequence Diagram* Sistem Prediksi Jumlah Wisatawan

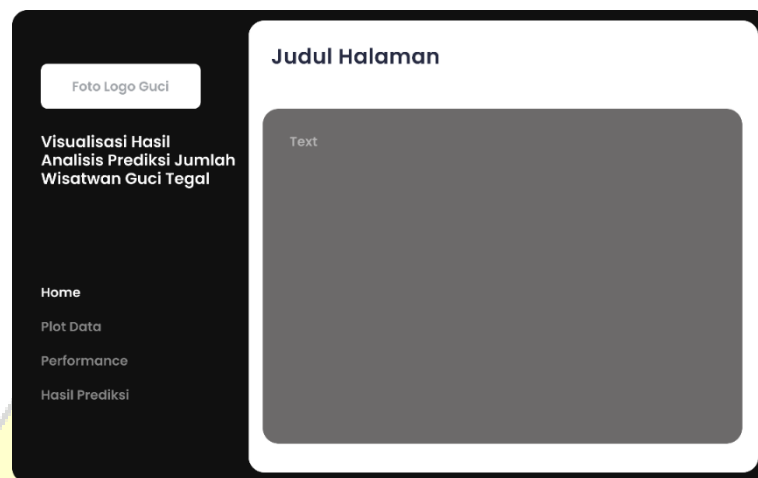
Gambar 5 merupakan rancangan *sequence diagram* yang menggambarkan interaksi antara pengguna, *website*, *database*, dan model ARIMA dalam menjalankan setiap menu. Diagram ini memperlihatkan alur proses ketika pengguna memilih plot data, *performance*, maupun hasil prediksi hingga sistem menampilkan grafik maupun tabel hasil kepada pengguna.

d. Desain Antarmuka (*User Interface*)

Peneliti merancang antarmuka *dashboard* prediksi jumlah wisatawan sebelum mengimplementasikannya ke dalam *website*. Perancangan ini dilakukan menggunakan *tools figma* untuk membuat rancangan tampilan yang interaktif dan memudahkan pengguna dalam mengakses menu serta informasi prediksi.

1) Rancangan Antarmuka Halaman *Homepage*

Hasil dari rancangan antarmuka halaman *homepage* dapat dilihat pada Gambar 6.

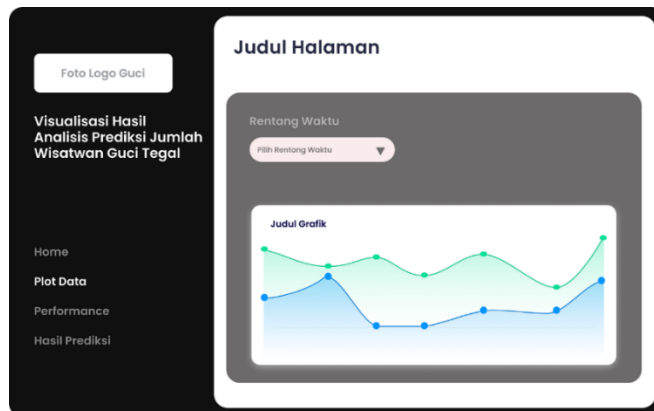


Gambar 6. Rancangan Antarmuka Halaman *Homepage*

Gambar 6 menampilkan rancangan antarmuka halaman *Home* yang memuat beberapa elemen, seperti sidebar yang memuat nama *website*, logo, serta menu navigasi berbentuk *dropdown*. Sementara itu, pada bagian utama halaman ditampilkan judul halaman dan konten utama.

2) Rancangan Antarmuka Halaman Plot Data

Hasil dari rancangan antarmuka halaman plot data dapat dilihat pada Gambar 7, yang menampilkan tampilan input data, pemilihan rentang waktu, serta grafik tren jumlah wisatawan berdasarkan data yang telah diunggah.

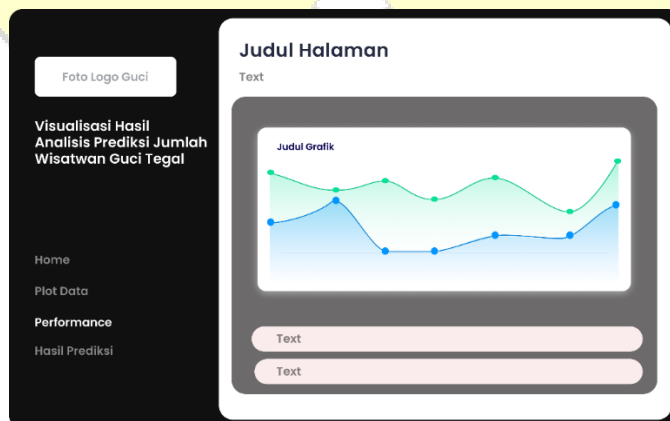


Gambar 7. Rancangan Antarmuka Halaman Plot Data

Gambar 7 menampilkan rancangan antarmuka halaman plot data pada *dashboard* prediksi jumlah wisatawan. Pada halaman ini disediakan beberapa komponen, yaitu judul halaman, filter rentang waktu untuk menampilkan data sesuai periode tertentu, serta grafik visualisasi yang digunakan untuk menampilkan informasi data jumlah wisatawan.

3) Rancangan Antarmuka Halaman *Performance*

Hasil dari rancangan antarmuka halaman *performance* dapat dilihat pada Gambar 8.

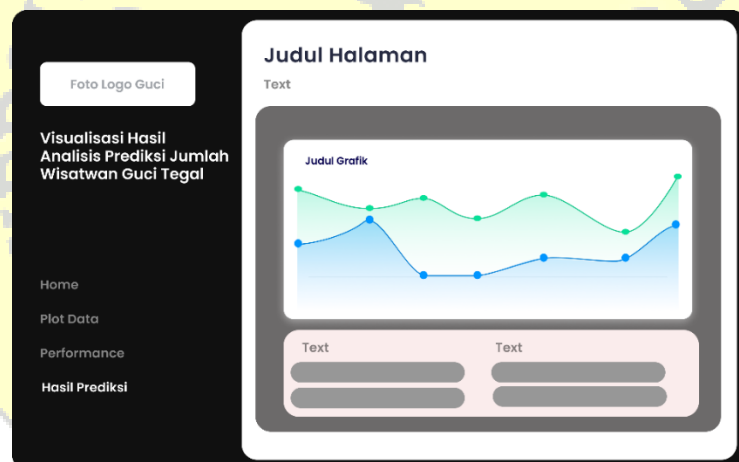


Gambar 8. Rancangan Antarmuka Halaman *Performance*

Gambar 8 merupakan rancangan antarmuka halaman *performance* pada *dashboard* prediksi jumlah wisatawan. Halaman ini memuat judul halaman dan teks penjelasan mengenai hasil prediksi yang telah dihasilkan oleh model ARIMA. Selain itu, halaman ini juga dilengkapi grafik visualisasi performa model serta informasi nilai akurasi berupa RMSE, MAPE, MAE dan akurasi model untuk memberikan gambaran tingkat akurasi prediksi.

4) Rancangan Antarmuka Halaman Hasil Prediksi

Hasil dari rancangan antarmuka halaman hasil prediksi dapat dilihat pada Gambar 9.



Gambar 9. Rancangan Antarmuka Halaman Hasil Prediksi

Gambar 9 merupakan rancangan antarmuka halaman hasil prediksi pada *dashboard* prediksi jumlah wisatawan. Pada halaman ini terdapat judul halaman, grafik visualisasi prediksi, serta tabel yang digunakan untuk menampilkan hasil prediksi jumlah wisatawan secara lebih detail dan terstruktur.

1.4 ARIMA

Metode ARIMA (*Autoregressive Integrated Moving Average*) digunakan dalam penelitian ini untuk melakukan prediksi jumlah wisatawan pada objek wisata Guci Tegal. Data yang digunakan dalam penelitian ini adalah data sekunder mengenai jumlah kunjungan wisatawan ke objek wisata Guci Tegal yang diperoleh dari Dinas Pariwisata Kabupaten Tegal. Data historis jumlah wisatawan Guci Tegal disajikan pada Tabel 4.

Tabel 3. Data Historis Jumlah Wisatawan

Tanggal	Dewasa	Anak	Wisatawan Mancanegara	Jumlah
2019-01-01	1485	126	2	1613
2019-01-02	1905	161	7	2073
2019-01-03	1666	141	6	1813
2019-01-04	2215	187	8	2410
...
...
2024-12-28	1871	15	0	1886
2024-12-29	1665	14	0	1679
2024-12-30	1700	14	0	1714
2024-12-31	2311	19	0	2330

Data pada tabel 4 merupakan data historis jumlah wisatawan objek wisata Guci Tegal pada periode 1 Januari 2019 – 31 Desember 2024. *Dataset* ini memiliki 2 variabel utama, yaitu tanggal dan jumlah, dengan total sebanyak 2193 baris data. Penelitian ini berfokus pada

variabel jumlah, yang menggambarkan total kunjungan harian untuk dilakukan proses prediksi.

Tahap awal pengolahan data meliputi beberapa langkah penting, yaitu mengimpor *dataset*, mengecek struktur tipe data, menampilkan informasi detail mengenai tipe data, melakukan pengecekan terhadap nilai kosong (*null*), serta memperoleh gambaran statistik deskriptif data. Seluruh proses awal ini didukung dengan pemrograman *Python*, yang dapat dilihat pada *Listing Program 1*.

```
#import dataset yang akan dilakukan analisis
data =
pd.read_csv('/content/drive/MyDrive/SKRIPSI/data/w
isatawan5.csv')
#pengecekan nilai null dan tipe data
data.info()

#statistik deskriptif
data.Jumlah.describe()
```

Listing Program 1. Persiapan Data Wisatawan

Pada *listing program 1*, tahap awal dilakukan dengan mengimpor *dataset* menggunakan fungsi *read_csv()* sesuai dengan format file yang tersedia. Selanjutnya, fungsi *head()* digunakan untuk menampilkan sejumlah baris awal dari dataset guna memastikan data telah terbaca dengan benar. Fungsi *info()* berperan dalam menampilkan struktur data, termasuk tipe data setiap kolom serta mendeteksi adanya nilai kosong (*null*). Sementara itu, fungsi *describe()* digunakan untuk memperoleh statistik deskriptif dari variabel yang diteliti. Terakhir, perintah '*data.set_index('Tanggal')*' digunakan untuk menjadikan kolom Tanggal sebagai indeks pada *dataset*. Ringkasan hasil statistik

deskriptif dari data penelitian ini dapat dilihat pada Tabel 5.

Tabel 4. Statistik Deskripsi Data Wisatawan

Statistik	Nilai
<i>Count</i>	2192,000000
<i>Mean</i>	1793,114964
<i>Std</i>	1348,618261
<i>Min</i>	0,000000
25%	876,750000
50%	1622,500000
75%	2406,500000
max	11233,000000

Statistik deskriptif pada tabel 5 merupakan ringkasan statistik dari data Jumlah pada jumlah wisatawan. Berikut adalah penjelasan untuk setiap metrik:

1. *Count* (Jumlah Data), Menunjukkan total entri data wisatawan yang tersedia dalam periode penelitian.
2. *Mean* (Rata-rata), Menggambarkan nilai rata-rata jumlah wisatawan yang berkunjung ke objek wisata Guci Tegal selama periode pengamatan.
3. *Std* (Standar Deviasi), Mengindikasikan seberapa besar variasi atau penyimpangan jumlah wisatawan dari nilai rata-ratanya. Semakin besar nilai ini, semakin tinggi fluktuasi jumlah kunjungan.
4. *Min* (Nilai Minimum), Jumlah wisatawan paling sedikit yang tercatat dalam satu periode pengamatan.
5. 25% (Kuartil Pertama), Batas nilai di mana 25% data jumlah wisatawan berada di bawahnya.

6. 50% (Median), Nilai tengah dari distribusi data jumlah wisatawan, yaitu titik di mana separuh data berada di bawah dan separuh lainnya berada di atas nilai tersebut.
7. 75% (Kuartil Ketiga), Batas nilai di mana 75% data berada di bawahnya, menunjukkan jumlah wisatawan pada kategori atas sebelum mencapai puncak kunjungan.
8. *Max* (Nilai Maksimum), Jumlah wisatawan terbanyak yang tercatat dalam satu periode pengamatan.

Ringkasan statistik ini memberikan gambaran awal mengenai pola distribusi jumlah wisatawan Guci, sehingga dapat diketahui tingkat rata-rata kunjungan, penyebaran data, serta titik-titik penting seperti median dan kuartil yang berperan dalam analisis lanjutan.

1.5 Identifikasi Model

a. Plot Data

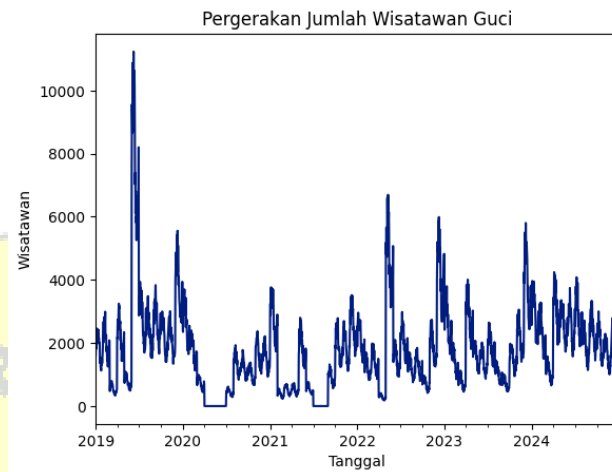
Tahap identifikasi diawali dengan melakukan pemetaan data ke dalam bentuk grafik untuk melihat pola pergerakan jumlah wisatawan. Dari visualisasi ini dapat diamati data menunjukkan adanya tren peningkatan atau penurunan, serta adanya pola musiman atau hanya bersifat nonmusiman. Pada tahapan ini disajikan pada *Listing Program*

2.

```
data['Jumlah'].  
plot() plt.title('Pergerakan Jumlah Wisatawan Guci')  
plt.ylabel('Wisatawan')
```

Listing Program 2. Plot Data Jumlah Wisatawan

Listing program 2 digunakan untuk membuat grafik tren data dengan memanfaatkan *library matplotlib* dan fungsi *plot*. Adapun Plot data pada data historis jumlah wisatawan objek wisata Guci Tegal dari 1 Januari 2019 – 31 Desember 2024 dapat dilihat pada gambar 10.



Gambar 10. Pergerakan Jumlah Wisatawan Guci

Gambar 10 menampilkan pergerakan jumlah wisatawan Guci dari tahun 2019 hingga 2024. Grafik menunjukkan fluktuasi jumlah kunjungan yang cukup signifikan sepanjang periode tersebut. Pada awal tahun 2019 terdapat lonjakan jumlah wisatawan yang sangat tinggi hingga lebih dari 10.000 orang. Namun setelah itu jumlah kunjungan menurun drastis dan mengalami pola naik-turun yang berulang.

Secara keseluruhan, grafik memperlihatkan adanya pola musiman, di mana terdapat periode puncak kunjungan yang cenderung terjadi setiap tahun. Meskipun demikian, besar puncak kunjungan tidak selalu sama dari tahun ke tahun, menunjukkan variasi yang cukup tinggi. Pola ini mengindikasikan bahwa jumlah wisatawan Guci Tegal dipengaruhi

oleh faktor musiman dan cenderung berfluktuasi tanpa adanya tren jangka panjang yang stabil meningkat atau menurun.

b. Identifikasi Stasioneritas Data

Setelah pola data teramati, tahapan berikutnya dalam pemodelan deret waktu menggunakan metode *Autoregressive Integrated Moving Average* (ARIMA) adalah melakukan proses identifikasi terhadap variabel yang akan dianalisis. Implementasi kode program pada tahap identifikasi ini ditampilkan pada *Listing Program 3*.

```
#Testing For Stationarity

from statsmodels.tsa.stattools import
adfuller
test_result=adfuller(data.Jumlah)

def adfuller_test(jumlah):
    result=adfuller(jumlah)
    labels = ['ADF Test Statistic','p-value','#Lags
for value,label in zip(result,labels):
        print(label+' : '+str(value) )
        if result[1] <= 0.05:
            print("strong evidence against the null
hypothesis(Ho), reject the null hypothesis. Data has
no unit root and is stationary")
        else:
            print("weak evidence against null hypothesis, time
series has a unit root, indicating it is non-
stationary")
    print('Used', 'Number of Observations Used']
plot_acf(diff, ax=ax2);
```

Listing Program 3. Uji ADF

Pada *listing program 3* ditunjukkan proses impor fungsi *adfuller* dari *library statsmodels.tsa.stattools*. Fungsi ini kemudian digunakan untuk melakukan *Augmented Dickey-Fuller Test* (ADF Test) pada data deret waktu variabel *data.Jumlah*. Hasil pengujian tersebut disimpan dalam variabel 'test_result'. Adapun ringkasan hasil dari eksekusi

Listing Program 3 ditampilkan pada Tabel 6.

Tabel 5. Hasil Uji ADF

Parameter	Hasil
ADF Test Statistic	-5.244345533793014
p-value	6.716892293708159e-23

Dari hasil uji ADF pada tabel 6 dapat dilihat bahwa nilai nilai ADF *Test Statistic* sebesar -5.244345533793014 dengan *p-value* sebesar 6.716892293708159e-23. Karena nilai $p\text{-value} < 0,05$, maka H_0 ditolak sehingga data dinyatakan telah memenuhi asumsi stasioner. Dengan demikian, data tidak memiliki akar unit dan tidak diperlukan proses *differencing* lebih lanjut. Hal ini menunjukkan bahwa akan menggunakan model ARIMA dengan nilai orde $d = 0$ untuk melakukan analisis lebih lanjut.

c. Identifikasi *Autocorrelation Function* (ACF) dan *Partial Autocorrelation Function* (PACF)

Pada tahap ini, dilakukan analisis menggunakan plot *Autocorrelation Function* (ACF) dan *Partial Autocorrelation Function* (PACF). Plot ACF digunakan untuk membantu menentukan orde p pada model *Autoregressive* (AR), sedangkan plot PACF berfungsi dalam menentukan orde q pada model *Moving Avarage* (MA). Pola dari kedua plot tersebut menjadi acuan dalam identifikasi model, di mana model AR maupun MA dapat dikenali apabila grafik ACF dan PACF menunjukkan penurunan yang signifikan. Implementasi kode program

pada tahapan ini ditampilkan pada *Listing Program 4*.

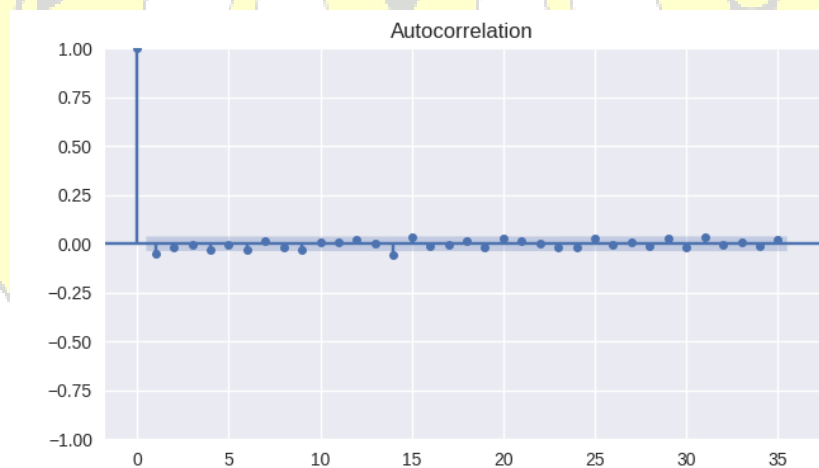
```
fig, (ax1, ax2) = plt.subplots(1, 2, figsize=(16,4))

ax1.plot(diff)
ax1.set_title('Difference once')

#add ; at the end of the plot function so that the
plot is not duplicated
plot_acf(diff, ax=ax2);
```

Listing Program 4. Plot ACF Orde p

Pada *listing program 4*, perintah ‘plot_acf(data.Jumlah, ax=ax2)’ digunakan untuk menampilkan grafik fungsi autokorelasi (ACF) dari data *time series* yang telah melalui proses ADF Test. Fungsi plot_acf ini berasal dari ‘library statsmodels.graphics.tsaplots’. Argumen ‘ax=ax2’ menunjukkan bahwa hasil plot ACF ditempatkan pada *subplot* kedua (‘ax2’). Visualisasi hasil plot ACF ditampilkan pada gambar 11.



Gambar 11. Grafik ACF Orde p

Berdasarkan hasil plot ACF diperlihatkan bahwa data tidak memiliki pola autokorelasi yang kuat pada *lag* tertentu, sehingga komponen *moving average* (q) dalam model arima kemungkinan kecil atau bahkan bisa bernilai nol. Langkah selanjutnya adalah membuat plot PACF. *Listing Program* pada langkah ini disajikan pada *Listing*

Program 5.

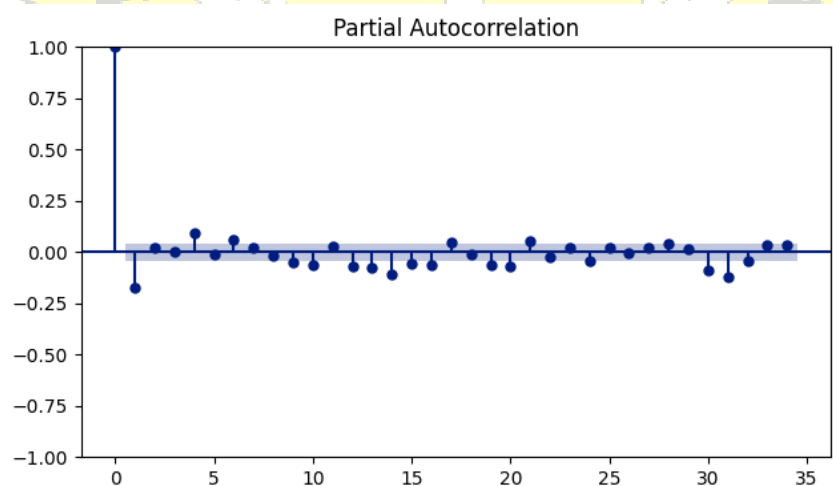
```
fig, (ax1, ax2) = plt.subplots(1, 2, figsize=(16,4))

ax1.plot(diff)
ax1.set_title('Difference once')

#add ; at the end of the plot function so that the
plot is not duplicated
plot_acf(diff, ax=ax2);
```

Listing Program 5. Plot PACF Orde q

Pada Listing Program 5, perintah ‘ax1.plot(diff)’ digunakan untuk menampilkan grafik data *time series* yang telah melalui proses *differencing* satu kali (‘diff’) pada subplot pertama (‘ax1’). Sementara itu, perintah ‘ax2.set_ylim(0, 1)’ berfungsi untuk membatasi rentang sumbu y pada subplot kedua (‘ax2’) agar hanya berada pada kisaran 0 hingga 1. Ini mempermudah dalam membaca serta menginterpretasikan nilai pada grafik PACF. Hasil visualisasi dari plot PACF dapat dilihat pada Gambar 12.



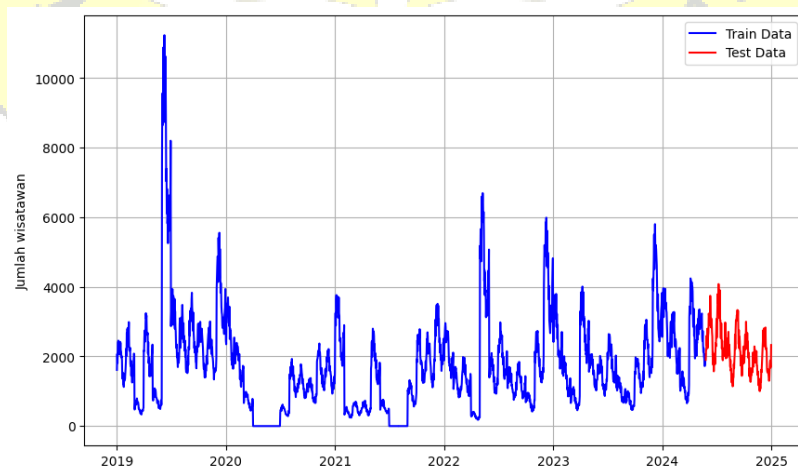
Gambar 12. Grafik PACF Orde q

Berdasarkan hasil plot PACF setelah dilakukan *differencing* satu kali, terlihat bahwa hampir seluruh *lag* berada dalam batas interval

kepercayaan, sehingga tidak ada *lag* yang signifikan. Hal ini menunjukkan bahwa tidak terdapat korelasi parsial yang kuat antar *lag*. Dengan demikian, nilai orde *autoregresif* (p) dapat diidentifikasi mendekati nol.

1.6 Pembuatan Model

Pada tahap ini, data dibagi menjadi dua bagian yaitu data *training* dan data *testing*. Sebanyak 90% data (1.972 observasi) digunakan sebagai data *training*, sedangkan 10% data (220 observasi) digunakan sebagai data *testing*. Proporsi 90:10 dipilih agar model memiliki cukup banyak data untuk proses pengujian (*training*) sehingga mampu menangkap pola yang kompleks dan umum, namun tetap menyisakan data uji (*testing*) yang memadai untuk mengevaluasi kinerja model secara akurat dan representatif. Grafik perbandingan antara data *training* dan data *testing* ditunjukkan pada Gambar 13.



Gambar 13. Grafik Perbandingan Data *Training* dan Data *Testing*

Gambar 13 menyajikan grafik perbandingan antara data *training* dan data *testing* yang digunakan dalam penelitian. Pada grafik tersebut,

garis berwarna biru merepresentasikan data *training*, sedangkan garis berwarna merah merepresentasikan data *testing*.

c. ARIMA Manual

Pada tahap identifikasi model, dilakukan analisis ACF dan PACF untuk menentukan nilai parameter (p,d,q). Hasil pengujian stasioneritas menunjukkan bahwa data sudah stasioner, sehingga nilai $d=0$. Berdasarkan plot ACF, tidak ditemukan lag yang signifikan sehingga nilai $q=0$. Sementara dari plot PACF terlihat adanya indikasi signifikan pada beberapa lag awal, sehingga kandidat model AR diperoleh dengan nilai p yang bervariasi. Dari hasil identifikasi, kandidat model manual yang dibentuk antara lain:

1. ARIMA(1,0,0)
2. ARIMA(2,0,0)
3. ARIMA(3,0,0)
4. ARIMA(4,0,0)
5. ARIMA(5,0,0)

Selanjutnya, setiap model diuji dan dibandingkan menggunakan kriteria AIC serta nilai *error* prediksi RMSE dan MAPE. Dari hasil perbandingan diperoleh bahwa model ARIMA(5,0,0) merupakan model terbaik karena menghasilkan nilai AIC paling rendah dan akurasi prediksi terbaik dibanding kandidat lainnya.

d. *Auto* ARIMA

Fungsi *auto-arima* memilih model terbaik berdasarkan nilai

Akaike Information Criterion (AIC). Proses penentuan ini dapat dilakukan dengan menjalankan kode yang ditunjukkan pada *Listing Program 6*.

```
def arimamodel(timeseries):  
    automodel = pm.auto_arima(timeseries,  
                               start_p=0,  
                               start_q=0,  
                               test='adf',  
                               seasonal=True,  
                               trace=True,  
                               stepwise=False)  
    return automodel
```

Listing Program 6. Auto-ARIMA

Listing program 6 mendefinisikan sebuah fungsi bernama *arimamodel* yang menggunakan pustaka *pmdarima* (diimpor sebagai *pm*) untuk secara otomatis menentukan model ARIMA terbaik bagi data *time series* yang diberikan. Parameter '*start_p=0*' menunjukkan bahwa pencarian nilai awal untuk parameter AR (*autoregressive*) dimulai dari 0, sedangkan '*start_q=0*' menandakan pencarian nilai awal untuk parameter MA (*moving average*) juga dimulai dari 0. Pengujian stasioneritas data dilakukan dengan metode ADF test '(*test='adf'*)'. Argumen *seasonal=True* menunjukkan bahwa model mempertimbangkan komponen musiman dalam data, sedangkan '*stepwise=False*' berarti pencarian model tidak menggunakan pendekatan *stepwise*, sehingga fungsi akan mengeksplorasi semua kemungkinan kombinasi parameter secara menyeluruh. Hasil dari proses ini menghasilkan model *auto-arima*, yang ditampilkan pada Tabel 8.

Tabel 6. Model *Auto*-ARIMA

Model ARIMA	AIC
ARIMA(0,0,0)	34168,275
ARIMA(0,0,1)	32351,764
ARIMA(0,0,2)	31221,222
ARIMA(0,0,3)	30743,571
ARIMA(0,0,4)	30237,309
ARIMA(0,0,5)	30066,892
ARIMA(1,0,0)	29374,944
ARIMA(1,0,1)	29339,887
ARIMA(1,0,2)	29332,232
ARIMA(1,0,3)	29332,725
ARIMA(1,0,4)	29315,629
ARIMA(2,0,0)	29334,761
ARIMA(2,0,1)	29332,725
ARIMA(2,0,2)	29333,719
ARIMA(2,0,3)	29333,898
ARIMA(3,0,1)	29332,308
ARIMA(3,0,1)	29334,931
ARIMA(3,0,2)	29331,964
ARIMA(4,0,0)	29333,776
ARIMA(4,0,1)	29331,061
ARIMA(5,0,0)	29309,820

Tabel 7 merupakan hasil dari *Listing Program 6* yang menampilkan model-model ARIMA secara otomatis. Tabel 7 juga menyajikan informasi mengenai nilai AIC pada masing-masing model.

1.7 Pemilihan Model Terbaik

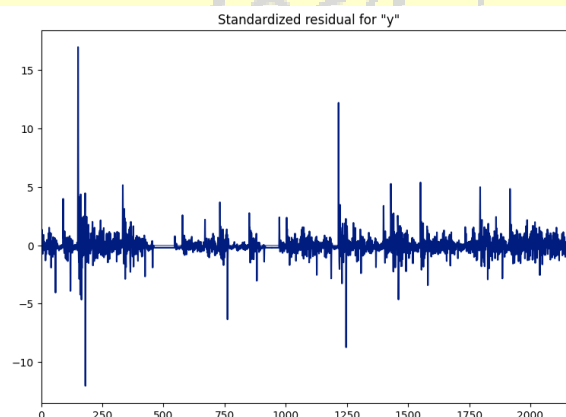
Pada tahap sebelumnya telah diperoleh beberapa kandidat model ARIMA beserta nilai AIC masing-masing. Pemilihan model terbaik ditentukan berdasarkan nilai AIC terkecil. Pada tabel 8 terlihat bahwa model ARIMA (5,0,0) memiliki nilai AIC paling kecil yaitu 29309.820. Dengan demikian, model ARIMA (5,0,0) dipilih sebagai model terbaik. Model ini kemudian dilatih menggunakan data *training* untuk menguji data baru dan selanjutnya dievaluasi. Ringkasan hasil pelatihan model

ARIMA (5,0,0) yang diperoleh melalui fungsi *automodel_summary()* dapat dilihat pada tabel 8.

Tabel 7. Ringkasan Automodel ARIMA(5, 0, 0)

Parameter	Coef	Std error	z	P> z	0.025	0.975
ar.L1	0,8120	0,007	122,164	0,000	0,799	0,825
ar.L2	0,1867	0,018	10,459	0,000	0,152	0,222
ar.L3	-0,0038	0,020	-0,192	0,848	-0,043	0,035
ar.L4	0,0777	0,022	3,561	0,000	0,033	0,114
ar.L5	-0,1155	0,011	-10,487	0,000	-0,137	-0,094
Sigma2	1,576e+05	1090,414	144,516	0,000	1,55e+05	1,6e+05

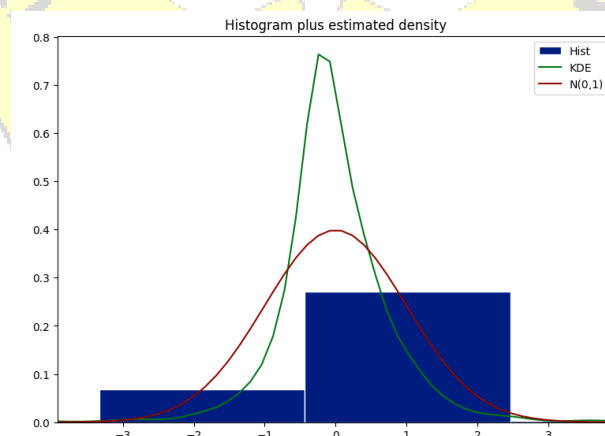
Hasil ini memberikan gambaran mengenai kecocokan model ARIMA (5,0,0) terhadap data yang dianalisis. Beberapa parameter model menunjukkan signifikansi, namun hasil diagnostik mengindikasikan adanya beberapa asumsi yang mungkin belum sepenuhnya terpenuhi, seperti normalitas residual. Oleh karena itu, diperlukan evaluasi lebih lanjut untuk memperoleh model yang lebih akurat dengan melakukan diagnostik residual melalui perintah ‘*model_fit.plot_diagnostics(figsize=(20, 14))*’. Hasil plot residual dari model ARIMA (5,0,0) ditampilkan pada Gambar 14.



Gambar 14. Plot Residual Standar Model ARIMA(5, 0, 0)

Gambar 14 merupakan plot *standardized residual* dari model ARIMA (5,0,0) terhadap waktu. Residual yang baik seharusnya berfluktuasi secara acak di sekitar nol tanpa pola tertentu. Dari plot tersebut terlihat bahwa sebagian besar residual menyebar di sekitar garis nol, meskipun terdapat beberapa lonjakan besar (*outlier*) pada titik-titik tertentu. Hal ini menunjukkan bahwa secara umum model sudah mampu menangkap pola utama dalam data, namun masih terdapat ketidakaturan pada beberapa periode tertentu.

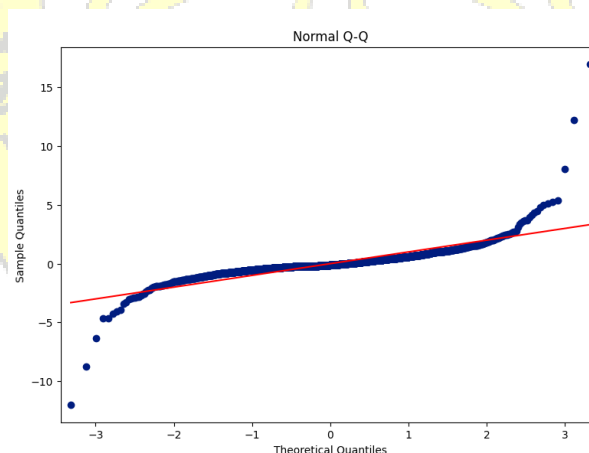
Plot residual ini juga memperlihatkan bahwa tidak ada pola musiman yang jelas, sehingga dapat dikatakan bahwa model telah cukup baik dalam menangkap pola musiman pada data, jika memang ada. Dengan demikian, model ARIMA (5,0,0) masih layak digunakan, meskipun evaluasi lebih lanjut tetap diperlukan untuk memastikan keakuratan prediksi. Plot selanjutnya adalah histogram yang disajikan pada gambar 15.



Gambar 15. Histogram dan Estimasi Kepadatan Model ARIMA(5, 0, 0)

Gambar 15 menunjukkan histogram residual yang dikombinasikan dengan estimasi kepadatan (*Kernel Density Estimation/KDE*) dan kurva distribusi normal standar ($N(0,1)$). Histogram (berwarna biru tua) menggambarkan distribusi frekuensi dari residual model ARIMA (5,0,0). Garis hijau merupakan estimasi kepadatan menggunakan metode KDE, sedangkan garis merah menunjukkan kurva distribusi normal standar sebagai pembanding.

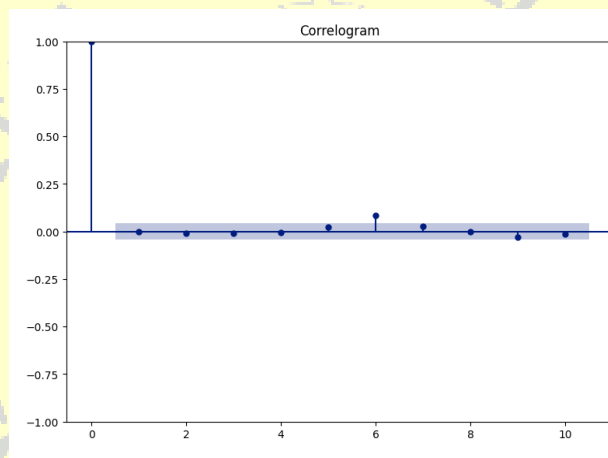
Dari plot ini terlihat bahwa distribusi residual cenderung mendekati distribusi normal, meskipun masih terdapat perbedaan bentuk, khususnya pada puncak distribusi yang lebih tajam dan pada bagian ekor kiri dan kanan (*tail*) yang menunjukkan adanya sedikit penyimpangan. Hal ini mengindikasikan bahwa residual tidak sepenuhnya berdistribusi normal.



Gambar 16. Q-Q Plot Model ARIMA(5, 0, 0)

Gambar 16 menampilkan Q-Q Plot residual yang membandingkan kuantil residual dengan distribusi normal. Apabila

residual benar-benar normal, titik-titik akan sejajar dengan garis merah lurus. Pada plot terlihat bahwa sebagian besar titik pada bagian tengah mengikuti distribusi normal dengan baik, tetapi terdapat penyimpangan pada ekor kiri dan kanan. Hal ini menunjukkan adanya *outlier* atau sifat distribusi dengan ekor lebih berat (*heavy tails*), sehingga residual tidak sepenuhnya normal. Meskipun demikian, secara keseluruhan pola residual masih cukup mendekati distribusi normal, sehingga model ARIMA yang digunakan tetap dapat dianggap sesuai dengan data.



Gambar 17. Korelogram Model ARIMA(5, 0, 0)

Gambar di atas merupakan Korelogram yang menunjukkan autokorelasi dari residual pada berbagai *lag*. Batang autokorelasi yang signifikan ditunjukkan apabila melewati batas interval kepercayaan (*confidence interval*). Dari plot ini terlihat bahwa seluruh batang autokorelasi berada dalam batas interval kepercayaan, sehingga tidak terdapat autokorelasi yang signifikan pada residual di berbagai *lag*. Hal ini mengindikasikan bahwa residual bersifat acak, tidak menyisakan pola tertentu, serta memiliki rata-rata mendekati nol. Dengan demikian,

model ARIMA yang digunakan telah cukup baik dalam menangkap pola data utama, dan tidak ada struktur temporal yang tertinggal dalam residual.

Setelah dilakukan uji diagnostik residual, langkah selanjutnya adalah menghitung performa model yang telah dipilih. *Listing program* untuk menampilkan grafik performa model disajikan pada *Listing Program 7*.

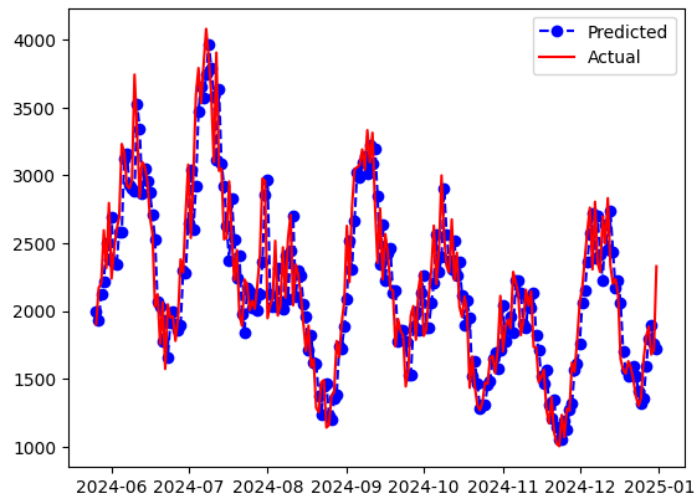
```
#Ambil tanggal sesuai panjang data uji
date_range = data[to_row:].index

#Pastikan prediksi juga hanya sepanjang data uji
model_predictions =
model_predictions[:len(date_range)]

# Plot
plt.plot(date_range, model_predictions, color='blue',
marker='o', ls='dashed', label='Predicted')
plt.plot(date_range, test, color='red',
label='Actual')
plt.legend()
plt.show()
```

Listing Program 7. Grafik Performa Model ARIMA

Pada *listing program 7* ‘plt.plot(date_range, model_predictions, color='blue', marker='o', ls='dashed', label='Predicted')’ membuat plot dari prediksi model. Sedangkan ‘plt.plot(date_range, test, color='red', label='Actual')’ membuat plot dari data harga aktual. Grafik performa setelah dilakukan diagnostik residual disajikan pada gambar 18.



Gambar 18. Grafik Performa Model ARIMA(5, 0, 0)

Gambar 18 memperlihatkan plot hasil prediksi terhadap 280 data uji jumlah wisatawan Guci Tegal. Kedekatan antara garis merah dengan garis biru putus-putus menunjukkan bahwa model prediksi cukup baik dalam mengikuti pola jumlah kunjungan wisatawan aktual. Model juga mampu menangkap sebagian besar perubahan tren, baik saat terjadi peningkatan maupun penurunan jumlah wisatawan.

Langkah selanjutnya adalah mengukur kinerja model dengan menghitung nilai *Mean Absolute Percentage Error* (MAPE) dan *Root Mean Square Error* (RMSE). Perhitungan kedua nilai tersebut disajikan pada *Listing Program 8*.

```
from math import sqrt
from sklearn.metrics import mean_squared_error

#Menghitung RMSE
rmse = sqrt(mean_squared_error(test,
model_predictions))
print('Test RMSE: %.3f' % rmse)
```

Listing Program 8. Menghitung nilai MAPE dan RMSE

Pada *listing program* 8 dilakukan impor fungsi ‘sqrt’ dari modul *math* untuk menghitung akar kuadrat. Selain itu, perintah ‘from sklearn.metrics import mean_squared_error’ digunakan untuk mengimpor fungsi *mean_squared_error* dari *scikit-learn*, yang berfungsi menghitung *Mean Squared Error* (MSE). Nilai MSE diperoleh dari perbandingan antara data aktual jumlah wisatawan (*test*) dengan hasil prediksi model (*model_predictions*). Selanjutnya, fungsi ‘sqrt(...)’ digunakan untuk mengambil akar kuadrat dari MSE sehingga diperoleh nilai *Root Mean Square Error* (RMSE). Untuk perhitungan *Mean Absolute Percentage Error* (MAPE), digunakan fungsi ‘np.mean(...)’ yang menghitung rata-rata dari persentase kesalahan absolut. Nilai RMSE dan MAPE ini digunakan sebagai indikator untuk menilai seberapa baik model ARIMA (5,0,0) dalam memprediksi jumlah wisatawan Guci. Hasil perhitungan kedua nilai tersebut disajikan pada Tabel 9.

Tabel 8. Nilai RMSE dan MAPE Model ARIMA(5, 0, 0)

RMSE	MAPE
294,179	0,1071

Nilai RMSE sebesar 294.179 menunjukkan bahwa rata-rata kesalahan prediksi model dalam memperkirakan jumlah wisatawan Guci adalah sekitar 294.179 unit. Nilai RMSE yang relatif kecil terhadap skala data menunjukkan bahwa model memiliki tingkat akurasi yang cukup baik. Sementara itu, nilai MAPE sebesar 0,1071 atau sekitar 10,71% menunjukkan bahwa rata-rata persentase kesalahan

absolut antara data aktual dan hasil prediksi model adalah sekitar 10,71%. Hal ini mengindikasikan bahwa model ARIMA yang digunakan memiliki tingkat kesalahan yang masih dapat diterima, sehingga dapat dikatakan cukup layak untuk melakukan peramalan jumlah wisatawan di objek wisata Guci.

4.8 Prediksi

Prediksi dilakukan dengan menggunakan model (5,0,0) untuk memperkirakan jumlah wisatawan selama 10 hari ke depan. Kode program untuk tahap ini ditampilkan pada *Listing Program 9*.

```
# 5. Forecasting
# =====
forecast_steps = 10 # Jumlah hari ke depan
start_date = data.index[-1] + pd.Timedelta(days=1)
forecast_dates = pd.date_range(start=start_date,
                                periods=forecast_steps, freq='D')

forecast_res =
model_fit.get_forecast(steps=forecast_steps)
forecast_mean = forecast_res.predicted_mean
conf_int = forecast_res.conf_int()

# Menggunakan metode get_forecast() untuk
memprediksi output, lower, dan upper bound dengan
level kepercayaan 90%
forecast_results = model_fit2.get_forecast(steps=10,
alpha=0.1) # Menggunakan level kepercayaan 90%

# Mengisi hasil prediksi ke dalam DataFrame future
future['forecast'] = forecast_results.predicted_mean
future['lower_bound'] =
forecast_results.conf_int(alpha=0.1)['lower Jumlah']
future['upper_bound'] =
forecast_results.conf_int(alpha=0.1)['upper Jumlah']
future.tail(15)
```

Listing Program 9. Prediksi Jumlah Wisatawan

Pada *listing program* 9, perintah ‘model_fit2.get_forecast(steps=10, alpha=0.1)’ digunakan untuk menghasilkan prediksi 10 langkah ke depan berdasarkan model yang

telah dilatih (*model_fit2*). Parameter 'alpha=0.1' menunjukkan tingkat kepercayaan sebesar 90% (karena $1 - \alpha = 0,9$). Sementara itu, 'forecast_results.conf_int(alpha=0.1)' memberikan interval kepercayaan 90% untuk hasil prediksi. Selain itu, kolom 'lower_bound' dan 'upper_bound' ditambahkan ke *DataFrame future* untuk menampilkan batas bawah dan batas atas dari interval kepercayaan. Hasil dari kode tersebut ditampilkan pada Tabel 10.

Tabel 9. Nilai Prediksi Jumlah Wisatawan

Tanggal	Prediksi
2025-01-01	2219
2025-01-02	2220
2025-01-03	2225
2025-01-04	2270
2025-01-05	2229
2025-01-06	2217
2025-01-07	2199
2025-01-08	2186
2025-01-09	2163
2025-01-10	2146

Tabel 10 menunjukkan nilai prediksi jumlah wisatawan selama 10 hari, mulai dari 1 Januari 2025 hingga 10 Januari 2025. Visualiasi perbandingan antara data aktual dengan hasil prediksi dapat dilakukan menggunakan kode pada *Listing Program 10*.


```

# Slicing rentang dari bulan Juni 2024 hingga data
terakhir
future_data_sliced = future['2024-06-04':]

# Plot data asli
plt.plot(future_data_sliced.index,
future_data_sliced['Jumlah'], label='Actual',
color='blue')

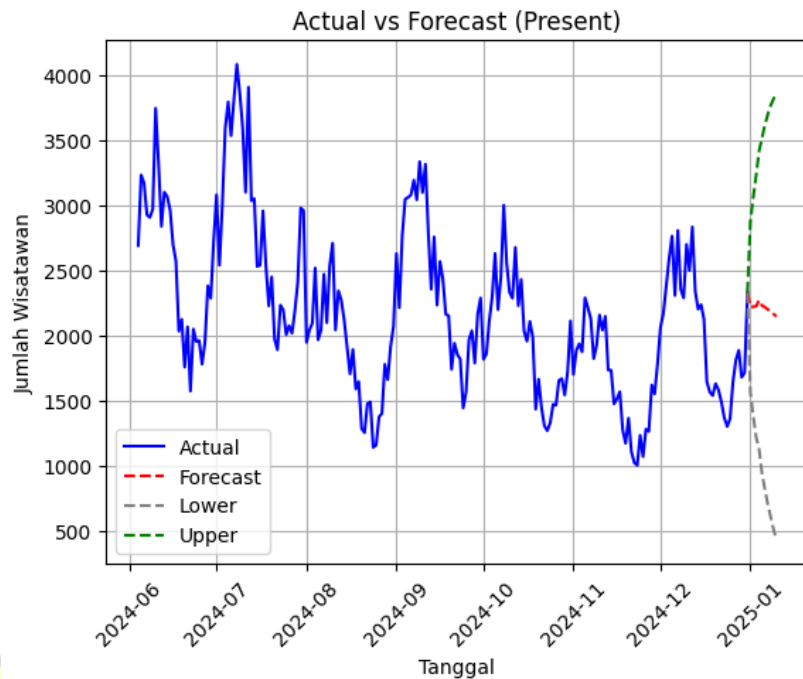
# Plot forecast
plt.plot(future_data_sliced.index,
future_data_sliced['forecast'], label='Forecast',
linestyle='--', color='red')
plt.plot(future_data_sliced.index,
future_data_sliced['lower_bound'], label='Lower',
linestyle='--', color='grey')
plt.plot(future_data_sliced.index,
future_data_sliced['upper_bound'], label='Upper',
linestyle='--', color='green')

# Label dan judul plot
plt.grid(True)
plt.xlabel('Tanggal')
plt.ylabel('Jumlah Wisatawan')
plt.title('Actual vs Forecast (Present)')
plt.legend()
# Membuat label sumbu x miring
plt.xticks(rotation=45)
# Tampilkan plot
plt.show()

```

Listing Program 10. Grafik Hasil Prediksi Jumlah Wisatawan

Listing program 10 digunakan untuk menampilkan grafik perbandingan antara data aktual jumlah wisatawan (Jumlah), hasil prediksi (*forecast*), dan interval kepercayaan (*lower_bound* dan *upper_bound*). Potongan kode `future['2024-06-04':]` dilakukan untuk *slicing* pada *Dataframe future*, mengambil data mulai 4 Juni 2024 hingga data terakhir. Hasil *slicing* tersebut disimpan ke dalam *variable future_data_sliced*. Visualisasi dari data aktual dan hasil prediksi jumlah wisatawan ditunjukkan pada Gambar 19, sehingga memudahkan analisis tren dari tanggal 4 Juni 2024 hingga periode prediksi.



Gambar 19. Grafik Nilai Aktual dan Nilai Prediksi Jumlah Wisatawan

Gambar 19 memperlihatkan grafik perbandingan antara data aktual jumlah wisatawan (*actual*), hasil prediksi (*forecast*), serta batas interval kepercayaan (*lower* dan *upper*) mulai Juni 2024 hingga periode akhir data. Garis prediksi (*forecast*) tampak relatif stabil dengan sedikit fluktuasi, yang mengindikasikan model memproyeksikan tren wisatawan yang cenderung konstan dalam jangka pendek. Interval kepercayaan terlihat semakin melebar menjelang akhir periode prediksi, menunjukkan adanya tingkat ketidakpastian yang lebih tinggi untuk prediksi jangka panjang. Batas atas interval kepercayaan (*upper*) mengalami peningkatan signifikan, sedangkan batas bawah (*lower*) menurun tajam, mencerminkan potensi variasi jumlah wisatawan yang cukup besar. Grafik ini memberikan gambaran visual mengenai kinerja model ARIMA terhadap data aktual dan mengilustrasikan tingkat

ketidakpastian prediksi, di mana semakin jauh periode prediksi, ketidakpastian yang dihasilkan semakin besar.

4.9 Implementasi Dashboard

Pada tahap ini dilakukan pengembangan *dashboard* visualisasi atau GUI (*Graphical User Interface*) dengan memanfaatkan *framework Python*, yaitu *Streamlit*, melalui IDE *Visual Studio Code*. Pembuatan *dashboard* bertujuan untuk menyajikan hasil prediksi saham agar lebih mudah dipahami oleh pengguna. *Dashboard* ini dilengkapi dengan beberapa menu atau halaman yang menampilkan informasi terkait analisis hasil prediksi. Kode program yang digunakan untuk mengatur navigasi antar halaman ditampilkan pada *Listing Program 11*.

```
import streamlit as st
import page_home, page1, page2, page3

# --- CONFIG ---
st.set_page_config(
    page_title="Visualisasi Prediksi Wisatawan Guci Tegal",
    page_icon="🌊",
    layout="wide"
)

# --- TEMA WARNA ---
st.markdown("""
<style>
    [data-testid="stAppViewContainer"] {background-color: #FFF9F0; color: #000;}
    [data-testid="stSidebar"] {background-color: #FFEEF5; color: #000;}
    [data-testid="stHeader"] {background-color: #FFF9F0;}
    h1, h2, h3, h4, h5, h6, p, li, span, label
    {color: #000 !important;}
    .block-container {padding-top: 1rem; padding-bottom: 2rem;}
</style>
""", unsafe_allow_html=True)
```

```
# --- SIDEBAR ---
with st.sidebar:
    st.image("assets/guci_logo.png", width=150)
    st.markdown("### 🟢 Visualisasi Prediksi Wisatawan Guci Tegal")
    page = st.radio("Navigasi", ["Home", "Plot Data", "Performance", "Hasil Prediksi"])

# --- HALAMAN ---
pages = {
    "Home": page_home,
    "Plot Data": page1,
    "Performance": page2,
    "Hasil Prediksi": page3
}

pages[page].app()
```

Listing Program 11. Navigasi Sistem Prediksi Jumlah Wisatawan

Pada *listing program 11* dilakukan impor modul *streamlit*, *pandas*, dan *plotly.express* untuk membangun tampilan antarmuka dan visualisasi data, serta modul *page_home*, *page1*, *page2*, dan *page3* sebagai halaman terpisah dalam aplikasi. Selanjutnya dilakukan konfigurasi tampilan dengan ‘*st.set_pageconfig()*’ untuk mengatur judul, ikon, dan tata letak halaman. Bagian sidebar dibuat menggunakan ‘*st.sidebar*’ yang menampilkan judul, logo, dan menu navigasi berbentuk *button* untuk berpindah antarhalaman. Terakhir, navigasi diatur menggunakan *dictionary pages* yang memetakan setiap menu dengan modulnya masing-masing, lalu perintah ‘*pages[page].app()*’ dijalankan untuk menampilkan halaman sesuai pilihan pengguna.

a. Halaman *Homepage*

Halaman *homepage* berfungsi sebagai tampilan utama dari *website* ini. Pada halaman tersebut ditampilkan informasi umum mengenai keseluruhan isi *website* serta berbagai menu yang tersedia di

dalamnya. Kode program untuk halaman *homepage* ditunjukkan pada

Listing Program 12.

```
import streamlit as st

def app():
    st.title("🌍 Prediksi Jumlah Wisatawan Objek  
Wisata Guci")
    st.write("Selamat datang di dashboard  
interaktif ini!")

    st.markdown(
        """
        ...
        """
    )

    st.success("Gunakan menu di sebelah kiri untuk  
menjelajahi dashboard.")
```

Listing Program 12. Homepage Sistem Prediksi Jumlah Wisatawan

Pada *listing program 12* ini dibuat fungsi ‘app()’ yang menampilkan halaman utama dari *dashboard* prediksi jumlah wisatawan. Baris ‘st.title()’ digunakan untuk menampilkan judul utama halaman, sedangkan ‘st.write()’ memberikan sambutan singkat kepada pengguna. Selanjutnya, ‘st.markdown()’ digunakan untuk menampilkan deskripsi isi *dashboard* yang menjelaskan bahwa analisis dilakukan menggunakan metode ARIMA serta mencantumkan fitur-fitur utama seperti Plot Data, *Performance*, dan Hasil Prediksi. Terakhir, ‘st.success()’ menampilkan pesan petunjuk agar pengguna menggunakan menu *sidebar* untuk menjelajahi fitur yang tersedia dalam *dashboard*.



Gambar 20. Halaman *Homepage* Sistem Prediksi Jumlah Wisatawan

Pada gambar 20 disajikan tampilan *dashboard* utama yang berisi menu *Home*, *Plot Data*, *Performance*, dan Hasil Prediksi. *Dashboard* ini juga memuat metode ARIMA beserta penjelasan istilah seperti MAE, MAPE, RMSE, dan Akurasi untuk evaluasi model. Selain itu, ditampilkan pula nilai *lower CI* dan *upper CI* sebagai batas kepercayaan hasil prediksi jumlah wisatawan.

b. Halaman Plot Data

Halaman ini menampilkan grafik prediksi jumlah wisatawan Guci Tegal menggunakan metode ARIMA. Sistem menyediakan fitur input data CSV dan filter tanggal untuk menampilkan data sesuai rentang waktu tertentu. Pengguna dapat menganalisis tren kunjungan wisatawan secara interaktif melalui halaman ini. *Listing Program 13* menampilkan kode program halaman plot data prediksi.

```

import streamlit as st
import pandas as pd
import plotly.express as px
import os

def app():
    st.title("🇮🇩 Plot Data Jumlah Wisatawan")

    # Upload data
    uploaded_file = st.file_uploader("Upload file CSV:",
type=["csv"])
    if uploaded_file:
        df = pd.read_csv(uploaded_file)
        df["Tanggal"] = pd.to_datetime(df["Tanggal"])
        os.makedirs("data", exist_ok=True)
        df.to_csv("data/wisatawan_input.csv",
index=False)

        # Pilih rentang waktu & filter
        start_date, end_date = st.date_input("Pilih
rentang waktu:",

[df["Tanggal"].min(), df["Tanggal"].max()])
        filtered = df.query("@start_date <= Tanggal <=
@end_date")

        # Tampilkan grafik dan data
        st.plotly_chart(px.line(filtered, x="Tanggal",
y="Jumlah", title="Tren Jumlah Wisatawan
Guci"),use_container_width=True)
        st.dataframe(filtered)
    else:
        st.info("Silakan upload file CSV terlebih
dahulu.")

```

Listing Program 13. Plot Data Jumlah Wisatawan

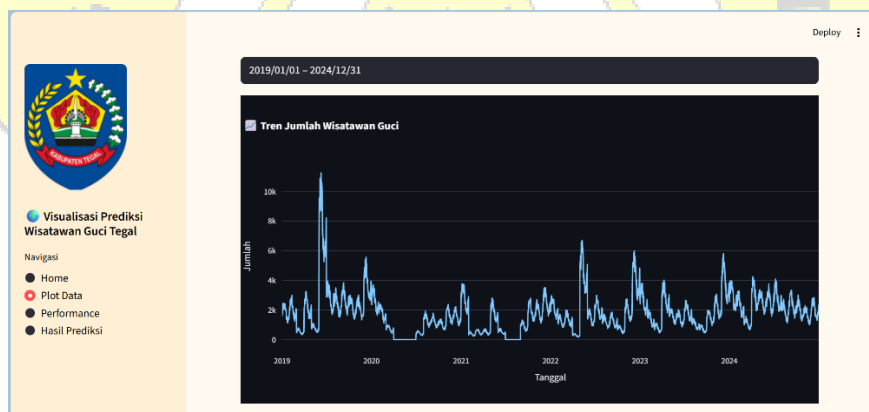
Pada *listing program* ini, modul *Streamlit*, *Pandas*, dan *Ploty Express* digunakan untuk menampilkan halaman plot data wisatawan. Data diunggah melalui ‘`st.file_uploader()`’ dan kolom “Tanggal” dikonversi ke format *datetime*. Pengguna dapat memilih rentang waktu dengan ‘`st.date_input()`’, kemudian data difilter sesuai tanggal yang dipilih. Hasilnya divisualisasikan dalam grafik garis menggunakan ‘`px.line()`’ dan ditampilkan dalam tabel interaktif melalui ‘`st.dataframe()`’. Hasil dari kode pada *Listing Program 13* ditampilkan

pada Gambar 21 dan Gambar 22.



Gambar 21. Halaman Input Data Wisatawan

Pada Gambar 21 ditampilkan halaman input data pada menu plot data jumlah wisatawan. Pada halaman ini mengunggah file CSV berisi data jumlah wisatawan untuk menampilkan grafik tren dan tabel data.



Gambar 22. Halaman Plot Data Jumlah Wisatawan

Gambar 22 menampilkan halaman Plot Data Prediksi Wisatawan dengan fitur input data dan filter tanggal. Setelah pengguna mengunggah data dan memilih rentang waktu, sistem menampilkan grafik dan tabel data aktual jumlah wisatawan untuk memudahkan pemahaman pola dan tren kunjungan secara interaktif.

c. Halaman *Performance*

Pada halaman ini ditampilkan grafik performa dari model ARIMA yang dihasilkan pada tahap analisis. Grafik tersebut memperlihatkan perbandingan antara data aktual dan hasil prediksi, sehingga pengguna dapat melihat sejauh mana model mampu memprediksi dengan akurat. Selain itu, halaman ini juga menampilkan nilai akurasi model, yaitu MAE, MAPE, RMSE dan akurasi sebagai pengukur tingkat performa model. *Listing program* untuk halaman *performance* disajikan pada *Listing Program* 14.

```
import streamlit as st
import pandas as pd
import numpy as np
import plotly.express as px
from statsmodels.tsa.arima.model import ARIMA
from sklearn.metrics import mean_squared_error,
mean_absolute_error, mean_absolute_percentage_error

def app():
    st.title("📊 Evaluasi Model ARIMA (5,0,0)")

    df = pd.read_csv("data/wisatawan_input.csv")
    df["Tanggal"] = pd.to_datetime(df["Tanggal"])

    # Split data train dan test
    split_idx = int(len(df) * 0.9)
    train, test = list(df["Jumlah"][:split_idx]),
list(df["Jumlah"][split_idx:])
    test_dates = df["Tanggal"].iloc[split_idx:]

    # Rolling forecast
    history, predictions = train.copy(), []
    for t in range(len(test)):
        model = ARIMA(history, order=(5,0,0)).fit()
        yhat = model.forecast()[0]
        predictions.append(yhat)
        history.append(test[t])
```

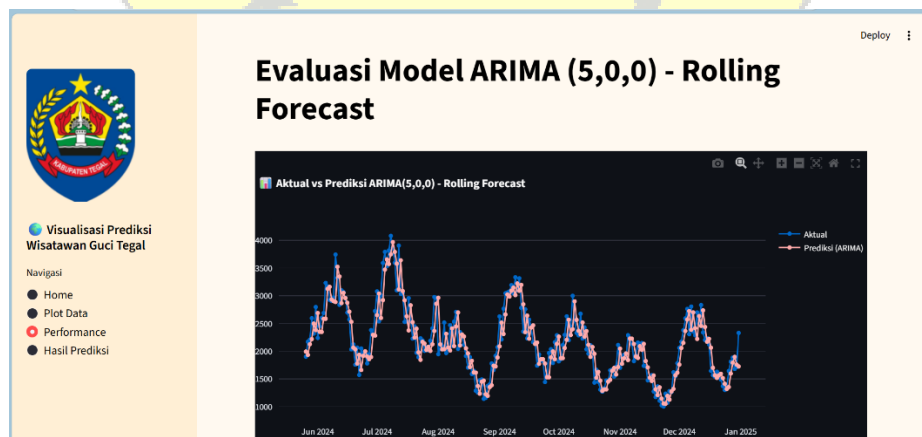
```
# Evaluasi model
mae = mean_absolute_error(test, predictions)
mape = mean_absolute_percentage_error(test, predictions)

rmse = np.sqrt(mean_squared_error(test,
predictions))
akurasi = 100 - mape

# Tampilkan hasil dan grafik
st.write(f"MAE: {mae:.2f} | MAPE: {mape:.2f}% |
RMSE: {rmse:.2f} | Akurasi: {akurasi:.2f}%")
fig = px.line(x=test_dates, y=[test, predictions],
labels={'x': 'Tanggal', 'value': 'Jumlah'}, title="Aktual
vs Prediksi ARIMA(5,0,0)")
st.plotly_chart(fig, use_container_width=True)
```

Listing Program 14. Performance Model ARIMA

Pada *listing program 14*, langkah pertama yang dilakukan adalah mengimpor modul-modul yang dibutuhkan untuk membangun halaman *performance*. Data jumlah wisatawan dibaca dari file CSV dan dibagi menjadi data *train* dan *test*. Proses *rolling forecast* dilakukan untuk menghasilkan nilai prediksi berdasarkan model ARIMA. Selanjutnya, didefinisikan fungsi *main* yang berfungsi untuk menampilkan nilai MAE, MAPE, RMSE, akurasi serta grafik performa model ARIMA. Hasil dari kode pada *Listing Program 14* ditampilkan pada Gambar 23.



Gambar 23. Halaman *Performance* Model ARIMA

Gambar 23 menampilkan tampilan halaman *performance* secara visual, di mana grafik performa menunjukkan perbandingan antara nilai aktual dan nilai hasil prediksi jumlah wisatawan. Selain itu, nilai MAE, MAPE, RMSE dan akurasi yang dihasilkan dari model ARIMA juga ditampilkan pada halaman tersebut sebagai indikator akurasi dan keandalan model prediksi.

d. Halaman Hasil Prediksi

Pada halaman hasil prediksi ditampilkan grafik jumlah wisatawan yang telah dianalisis menggunakan metode ARIMA. Grafik tersebut memperlihatkan perbandingan antara data aktual dan hasil prediksi. Selain itu, pada halaman ini juga terdapat tabel hasil prediksi jumlah wisatawan untuk 10 hari ke depan, yang dihasilkan dari model ARIMA. *Listing program* untuk halaman Hasil Prediksi disajikan pada *Listing Program 15*.

```
import streamlit as st
import pandas as pd
import plotly.express as px
import joblib
import numpy as np
from sklearn.metrics import mean_absolute_error,
mean_squared_error

def app():
    st.title("📊 Prediksi Jumlah Wisatawan")

    # Load data dan model ARIMA
    df = pd.read_csv("data/wisatawan5.csv")
    df["Tanggal"] = pd.to_datetime(df["Tanggal"])
    model = joblib.load("models/arima_model.pkl")

    # Prediksi berdasarkan rentang tanggal
    start, end = st.date_input("Pilih rentang
    tanggal prediksi:",
    [df["Tanggal"].max() + pd.Timedelta(days=1),
```

```

df["Tanggal"].max() + pd.Timedelta(days=10))
steps = (end - start).days + 1
forecast = model.get_forecast(steps=steps)
future_dates = pd.date_range(start,
periods=steps)

hasil = pd.DataFrame({
    "Tanggal": future_dates,
    "Prediksi": forecast.predicted_mean,
    "Lower CI": forecast.conf_int().iloc[:, 0],
    "Upper CI": forecast.conf_int().iloc[:, 1]
})

# Visualisasi hasil prediksi
fig = px.line(df, x="Tanggal", y="Jumlah",
title="Prediksi Jumlah Wisatawan")
fig.add_scatter(x=hasil["Tanggal"],
y=hasil["Prediksi"], mode="lines+markers",
name="Prediksi")
st.plotly_chart(fig, use_container_width=True)

# Evaluasi model
mae = mean_absolute_error(df["Jumlah"],
model.fittedvalues)
mape = np.mean(np.abs((df["Jumlah"] -
model.fittedvalues) / df["Jumlah"])) * 100
rmse = np.sqrt(mean_squared_error(df["Jumlah"],
model.fittedvalues))
akurasi = 100 - mape

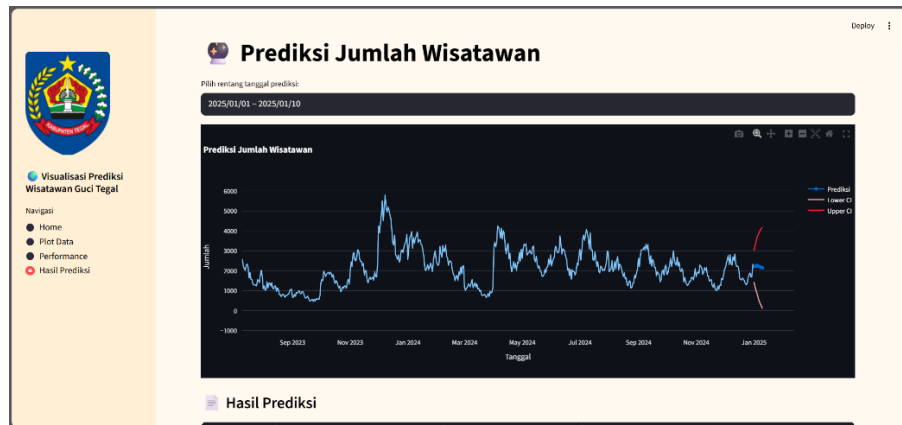
st.write(f"MAE: {mae:.2f} | MAPE: {mape:.2f}% |
RMSE: {rmse:.2f} | Akurasi: {akurasi:.2f}%")

```

Listing Program 15. Hasil Prediksi Jumlah Wisatawan

Pada *listing program 15*, tahap pertama yang dilakukan yaitu mengimpor berbagai modul yang diperlukan untuk membangun halaman hasil prediksi. Data dibaca dari file CSV, kemudian model ARIMA yang telah dilatih digunakan untuk memprediksi jumlah wisatawan berdasarkan rentang tanggal yang dipilih pengguna. Hasil prediksi divisualisasikan dalam grafik dan tabel. Selain itu, menampilkan nilai metrik MAE, MAPE, RMSE, serta akurasi untuk menilai kinerja model. Tampilan hasil dari listing program tersebut yang merepresentasikan halaman hasil prediksi dapat dilihat pada

gambar 24 dan gambar 25.



Gambar 24. Halaman Hasil Prediksi Grafik Jumlah Wisatawan

Gambar 24 menampilkan halaman Hasil Prediksi yang berisi grafik hasil prediksi jumlah wisatawan. Pada grafik tersebut terlihat plot data aktual pada periode sebelumnya serta nilai untuk prediksi beberapa hari ke depan. Grafik ini membantu dalam melihat tren perubahan jumlah wisatawan berdasarkan hasil prediksi menggunakan metode ARIMA.

Hasil Prediksi			
Tanggal	Prediksi	Lower CI (Batas Prediksi Terendah)	Upper CI (Batas Prediksi Tertinggi)
2025-01-01 00:00:00	7719 Wisatawan	3441 Wisatawan	7997 Wisatawan
2025-01-02 00:00:00	2230 Wisatawan	1217 Wisatawan	3222 Wisatawan
2025-01-03 00:00:00	3223 Wisatawan	2022 Wisatawan	3424 Wisatawan
2025-01-04 00:00:00	7770 Wisatawan	3677 Wisatawan	3634 Wisatawan
2025-01-05 00:00:00	2229 Wisatawan	694 Wisatawan	3765 Wisatawan
2025-01-06 00:00:00	2217 Wisatawan	550 Wisatawan	3884 Wisatawan
2025-01-07 00:00:00	7109 Wisatawan	417 Wisatawan	3937 Wisatawan
2025-01-08 00:00:00	2186 Wisatawan	304 Wisatawan	4067 Wisatawan
2025-01-09 00:00:00	2163 Wisatawan	344 Wisatawan	4133 Wisatawan
2025-01-10 00:00:00	7146 Wisatawan	307 Wisatawan	4191 Wisatawan

Hasil Evaluasi Model			
MAE	MAPE	MSE	RMSE
233.69	14.82%	85.18%	410.20

Model ARIMA memiliki akurasi sebesar 85.18% berdasarkan data historis.

Gambar 25. Halaman Hasil Prediksi Tabel Jumlah Wisatawan

Gambar 25 menampilkan tabel hasil prediksi jumlah wisatawan untuk 10 hari ke depan. Tabel tersebut memuat informasi tanggal dan

nilai hasil prediksi jumlah wisatawan yang dihasilkan oleh model ARIMA. Selain itu, halaman ini juga menampilkan nilai evaluasi model berupa MAE, MAPE, RMSE, serta akurasi yang digunakan untuk mengukur tingkat kesalahan serta performa model. Halaman ini merupakan halaman terakhir dari *website*, yang berfungsi untuk menyajikan hasil akhir prediksi dalam bentuk tabel agar mudah dibaca dan dianalisis oleh pengguna.

4.10 Pengujian *Website*

Pada tahap ini dilakukan pengujian menggunakan metode *blackbox testing*. Tujuan pengujian ini adalah untuk memastikan bahwa seluruh fungsi pada *website* telah berjalan sesuai dengan kebutuhan dan perintah yang telah ditentukan. Hasil dari pengujian *blackbox testing* pada *website* prediksi ditampilkan pada Tabel 10.

Tabel 10. Hasil Pengujian *Blackbox Testing* Sistem Prediksi

No	Deskripsi Pengujian	Hal yang diharapkan	Hasil Pengujian	Kesimpulan
1	Klik menu “Home” pada sidebar navigasi.	<i>Website</i> menampilkan halaman utama berisi judul, deskripsi, dan fitur utama.	Halaman “Home” berhasil tampil sesuai deskripsi.	Sesuai.
2	Klik menu “Plot Data” pada sidebar navigasi.	<i>Webiste</i> menampilkan grafik tren jumlah wisatawan dan tabel data berdasarkan	Grafik dan tabel data tampil dengan benar sesuai rentang	Sesuai.

No	Deskripsi Pengujian	Hal yang diharapkan	Hasil Pengujian	Kesimpulan
		rentang tanggal.	waktu yang dipilih.	
3	Klik tombol “ <i>Upload File CSV</i> ” pada halaman “Plot Data” dan pilih file berformat CSV yang valid.	Sistem memproses file dan menampilkan grafik serta tabel data sesuai isi file.	Grafik dan tabel berhasil ditampilkan sesuai data yang diunggah.	Sesuai.
3	Klik menu “ <i>Performance</i> ” pada <i>sidebar</i> navigasi.	<i>Website</i> menampilkan grafik perbandingan data aktual dan hasil prediksi serta nilai RMSE dan MAPE.	Grafik dan nilai metrik berhasil ditampilkan dengan benar.	Sesuai.
4	Klik menu “Hasil Prediksi” pada <i>sidebar</i> navigasi.	<i>Website</i> menampilkan grafik hasil prediksi jumlah wisatawan dan tabel hasil prediksi berdasarkan tanggal.	Grafik dan tabel hasil prediksi tampil dengan benar.	Sesuai.
5	Memilih rentang tanggal pada fitur kalender di halaman Plot Data	Grafik dan tabel menyesuaikan data sesuai tanggal yang dipilih.	Grafik dan tabel berhasil berubah sesuai rentang waktu.	Sesuai.

No	Deskripsi Pengujian	Hal yang diharapkan	Hasil Pengujian	Kesimpulan
6	Memilih rentang tanggal prediksi pada halaman Hasil Prediksi.	Grafik hasil prediksi menyesuaikan dengan tanggal yang dipilih.	Grafik berhasil berubah sesuai dengan rentang tanggal yang diatur.	Sesuai.
7	Menampilkan grafik <i>performance</i> perbandingan data aktual dan hasil prediksi.	Grafik menunjukkan dua garis berbeda dengan label yang sesuai.	Grafik tampil sesuai dengan data aktual dan hasil prediksi.	Sesuai.
8	Membandingkan nilai pada tabel dengan grafik hasil prediksi.	Nilai pada grafik sesuai dengan nilai di tabel hasil prediksi.	Data pada grafik dan tabel konsisten dan sesuai.	Sesuai.

4.11 Pengujian Hasil Prediksi

Pada tahap ini dilakukan pengujian hasil prediksi menggunakan dua model yang berbeda, yaitu ARIMA(5, 0, 0) dan ARIMA(3, 0, 3). Tujuan dari pengujian ini adalah untuk mengetahui model mana yang memiliki performa paling baik dalam memprediksi jumlah wisatawan pada objek wisata Guci Tegal. Kedua model dilatih menggunakan data historis jumlah wisatawan dan diuji menggunakan data uji (*testing*) yang telah dipisahkan sebelumnya. Pengujian dilakukan dengan menghitung nilai *Mean Absolute Error* (MAE), *Root Mean Squared Error* (RMSE), *Mean Absolute Percentage Error* (MAPE), serta tingkat

akurasi dari hasil prediksi masing-masing model. Adapun hasil perbandingan kedua model dapat dilihat pada tabel 12 dan tabel 13.

Tabel 11. Hasil Pengujian Prediksi ARIMA(5, 0, 0)

Tanggal	Harga Prediksi	Harga Sebenarnya	<i>Absolute Error</i> Prediksi
2025-01-01	2219	1950	269
2025-01-02	2220	2590	370
2025-01-03	2225	2605	380
2025-01-04	2270	2700	430
2025-01-05	2229	2405	176
2025-01-06	2217	2487	270
2025-01-07	2199	2530	331
2025-01-08	2186	2605	419
2025-01-09	2163	2800	637
2025-01-10	2146	2615	469
MAE			233,87
MAPE			14,83%
Akurasi			85,17%

Tabel 12 menampilkan hasil prediksi jumlah wisatawan selama sepuluh hari ke depan beserta nilai aktual dan *absolute error*. Berdasarkan hasil pengujian, model ini menghasilkan MAE sebesar 233,87, MAPE sebesar 14,83%, dan akurasi sebesar 85,17%.

Selanjutnya, pada pengujian pada model ARIMA(3, 0, 3) dilakukan untuk melihat performa prediksi yang dihasilkan dan membandingkannya dengan model sebelumnya. Hasil pengujian tersebut dapat dilihat pada Tabel 13.

Tabel 12. Hasil Pengujian Prediksi ARIMA(3, 0, 3)

Tanggal	Harga Prediksi	Harga Sebenarnya	<i>Absolute Error</i> Prediksi
2025-01-01	2428	1950	478
2025-01-02	2086	2590	504
2025-01-03	3136	2605	531

Tanggal	Harga Prediksi	Harga Sebenarnya	<i>Absolute Error</i> Prediksi
2025-01-04	2143	2700	557
2025-01-05	2989	2405	584
2025-01-06	1877	2487	610
2025-01-07	3167	2530	637
2025-01-08	1942	2605	663
2025-01-09	3490	2800	690
2025-01-10	1899	2615	716
MAE			505,97
MAPE			23,64%
Akurasi			76,36%

Tabel 13 menampilkan hasil pengujian model ARIMA(3, 0, 3) terhadap data jumlah wisatawan selama sepuluh hari ke depan beserta nilai aktual dan *absolute error*. Berdasarkan hasil pengujian, model ini menghasilkan MAE sebesar 505.97, MAPE sebesar 23,64%, dan akurasi sebesar 76,36%.

Berdasarkan hasil kedua model dapat disimpulkan bahwa model ARIMA(5, 0, 0) memiliki tingkat kesalahan lebih kecil dan akurasi lebih tinggi dibandingkan ARIMA(3, 0, 3). Dengan demikian, model ARIMA(5, 0, 0) dinyatakan sebagai model yang lebih sesuai untuk digunakan dalam memprediksi jumlah wisatawan pada penelitian ini.

BAB V

PENUTUP

5.1 Kesimpulan

Berdasarkan hasil penelitian yang telah dilakukan, maka dapat disimpulkan sebagai berikut:

1. Proses implementasi metode *Autoregressive Integrated Moving Average* (ARIMA) untuk memprediksi jumlah wisatawan objek wisata Guci Tegal meliputi tahapan pra-pemrosesan data, uji stasioneritas, identifikasi parameter ACF dan PACF, estimasi model menggunakan *auto-ARIMA*, serta evaluasi berdasarkan nilai AIC, RMSE, dan MAPE. Dari tahapan tersebut diperoleh model terbaik ARIMA(5, 0, 0) dengan tingkat akurasi prediksi tertinggi.
2. Berdasarkan model terbaik tersebut, hasil prediksi jumlah wisatawan diperoleh untuk sepuluh hari ke depan yaitu 2.219; 2.220; 2.225; 2.270; 2.229; 2.217; 2.199; 2.186; 2.163; dan 2.146, yang menunjukkan nilai relatif stabil serta masih mengikuti tren data aktual.
3. Selain itu, visualisasi hasil prediksi berhasil diterapkan melalui aplikasi berbasis *website* menggunakan *framework Streamlit* yang menampilkan representasi data prediksi dalam bentuk visual interaktif dan komponen informatif mengenai jumlah wisatawan. Sistem ini dikembangkan menggunakan pendekatan *prototype* untuk

mempermudah proses perancangan dan penyesuaian kebutuhan pengguna, serta telah diuji menggunakan metode *blackbox testing* guna memastikan seluruh fungsi berjalan dengan baik dan sesuai harapan.

5.2 Saran

Adapun saran dari penelitian ini adalah agar dilakukan penelitian lebih lanjut terhadap data jumlah wisatawan pada objek wisata Guci Tegal dengan menggunakan metode lain selain ARIMA, sehingga dapat dibandingkan tingkat keakuratannya. Selain itu, pengembangan sistem masih memiliki beberapa kekurangan, seperti performa *website* yang terkadang lambat pada menu *performance* dan waktu pemuatan data yang cukup lama. Oleh karena itu, disarankan agar dilakukan optimalisasi pada sisi performa dan efisiensi sistem agar aplikasi dapat berjalan lebih responsif dan stabil di masa mendatang.

DAFTAR PUSTAKA

- Abidin, A. A., Buiney, P. F., & Nohe, D. A. (2022). Peramalan Data Ekspor Kalimantan Barat Dengan Metode Autoregressive Integrated Moving Average (Arima). *Prosiding Seminar Nasional Matematika dan Statistika*, 96–107. <https://jurnal.fmipa.unmul.ac.id/index.php/SNMSA/article/view/900>
- Azhar, P. A., Pratama, M. A., & Fitriani, R. (2024). *Prediksi Harga Mobil Audi Bekas Menggunakan Model Regresi Linear dengan Framework Streamlit*. 6(1). <https://doi.org/10.37802/joti.v6i1.763>
- Christian, J., & Idrus, S. I. Al. (2023). Introduction to Citrus Fruit Ripens Using the Deep Learning Convolutional Neural Network (CNN) Learning Method. *Asian Journal of Applied Education (AJAE)*, 2(3), 459–470. <https://doi.org/10.55927/ajae.v2i3.5003>
- Daratullaila, & Sari, R. P. (2024). Prediksi Jumlah Kejahatan Di Indonesia Pada Tahun 2023 Dengan Metode Autoregressive Integrate Moving Average (Arima). *Jurnal Gamma-Pi*, 5(2), 60–67. <https://doi.org/10.33059/gamma-pi.v5i2.9523>
- Fauzani, S. P., & Rahmi, D. (2023). Penerapan Metode ARIMA Dalam Peramalan Harga Produksi Karet di Provinsi Riau. *Jurnal Teknologi dan Manajemen Industri Terapan*, 2(4), 269–277. <https://doi.org/10.55826/tmit.v2i4.283>
- Hendrik Hidayatullah, Fitri Sukaesih, Yanuar Arif Hizbulloh, & Tb Ai Munandar. (2023). Implementasi Metode Arima Data Warehouse Untuk Prediksi Permintaan Suku Cadang. *Jurnal Riset Informatika dan Teknologi Informasi*, 1(1), 30–37. <https://doi.org/10.58776/jriti.v1i1.48>

- Heru Widiyanto, M., Mayasari, R., & Garno, G. (2023). Implementasi Time Series Pada Data Penjualan Di Gaikindo Menggunakan Algoritma Seasonal Arima. *JATI (Jurnal Mahasiswa Teknik Informatika)*, 7(3), 1501–1506. <https://doi.org/10.36040/jati.v7i3.6879>
- Hilhami, M. S. A. A. (2020). *Forecasting Harga Saham PT. Astra Agro Lestari Dengan Metode Simple Moving Average Dan Weighted Moving Average*. <https://id.investing.com/equities/astra->
- Husna, F. K. (2022). Analisis dampak sektor pariwisata bagi perekonomian warga sekitar kawasan wisata Siblarak Polanharjo Kabupaten Klaten. *Journal of Economics Research and Policy Studies*, 2(2), 104–117. <https://doi.org/10.53088/jerps.v2i2.577>
- Khamidah, K., & Azakia, K. (2024). Strategi Pengembangan Quality Tourism Pada Objek Wisata Guci Sebagai Potensi Pertumbuhan Ekonomi Daerah. *Jurnal Ekonomi Syari'Ah*, 5(2), 273–286.
- Koh, C. W., Ooi, J. S., Joly, G. L., & Rong Chan, K. (2021). Gene Updater: A Streamlit web tool that autocorrects and updates for Excel misidentified gene names. *Research Square*, 1–8. <https://doi.org/10.21203/rs.3.rs-1146062/v1>
- Lumansik, J. R. C., Kawung, G. M. V, & ... (2022). Analisis Potensi Sektor Pariwisata Air Terjun Di Desa Kali Kecamatan Pineleng Kabupaten Minahasa. *Jurnal Berkala Ilmiah ...*, 22(1), 13–23. <https://ejournal.unsrat.ac.id/index.php/jbie/article/view/38177>
- Madyatmadja, E. D., Kusumawati, L., Jamil, S. P., Kusumawardhana, W., Informasi, S., & Nusantara, U. B. (2021). Infotech: journal of technology information. *Raden Ario Damar*, 7(1), 55–62.
- Mangilang, A. (2024). *Peramalan Permintaan Telur pada PT. Rijen di Tallunglipu Menggunakan Metode Seasonal Autoregressive*

Integrated Moving Average (SARIMA) (Doctoral dissertation, Universitas Kristen Indonesia Toraja).

Mardiyah, I., Dianita Utami, W., Rini Novitasari, D. C., Hafiyusholeh, M., & Sulistiyawati, D. (2021). Analisis Prediksi Jumlah Penduduk Di Kota Pasuruan Menggunakan Metode Arima. *BAREKENG: Jurnal Ilmu Matematika dan Terapan*, 15(3), 525–534.

<https://doi.org/10.30598/barekengvol15iss3pp525-534>

Maulana, M. R. (2021). *Peramalan (forecasting) jumlah wisatawan objek wisata Guci Tegal dengan metode dekomposisi dan winter's exponential smoothing*. 75(17), 399–405.
<https://eprints.walisongo.ac.id/id/eprint/16704>

Maulida Surbakti, N., Talia, A., Br Perangin-Angin, C., Olivia Nainggolan, D., Devi Friskauly, N., & Ruth Br Tumorang, S. (2024). Penggunaan Bahasa Pemrograman Python dalam Pembelajaran Kalkulus Fungsi Dua Variabel. *Kebumian dan Angkasa*, 2(3), 98–107. <https://doi.org/10.62383/algorithm.v2i3.67>

Milniadi, A. D., & Adiwijaya, N. O. (2023). Analisis perbandingan model arima dan lstm dalam peramalan harga penutupan saham (studi kasus: 6 kriteria kategori saham menurut peter lynch). *SIBATIK JOURNAL: Jurnal Ilmiah Bidang Sosial, Ekonomi, Budaya, Teknologi, dan Pendidikan*, 2(6), 1683-1692.

Ramadhani, A., Wahyuningsih, S., & Siringoringo, M. (2022). Peramalan Jumlah Kunjungan Wisatawan Mancanegara Ke Indonesia Menggunakan Autoregressive Integrated Moving Average (ARIMA). *Ekspansional*, 13(2), 103.
<https://doi.org/10.30872/ekspansional.v13i2.1049>

- Rianto, M., & Yunis, R. (2021). Analisis Runtun Waktu Untuk Memprediksi Jumlah Mahasiswa Baru Dengan Model Random Forest. *Paradigma Jurnal Komputer dan Informatika*, 23(1). <https://doi.org/10.31294/p.v23i1.9781>
- Rizki, A. (2023). Aplikasi Model ARIMA dalam Peramalan Data Harga Emas Dunia Tahun 2010-2022. *Jurnal Statistika dan Aplikasinya*, 7(1), 84–92. <https://doi.org/10.21009/jsa.07108>
- Setiaji, S., & Sastra, R. (2021). Implementasi Diagram UML (Unified Modelling Language) Pada Perancangan Sistem Informasi Penggajian. *Jurnal Teknik Komputer AMIK BSI*, 7(1), 160–111. <https://doi.org/10.31294/jtk.v4i2>
- Seymour, L., Brockwell, P. J., & Davis, R. A. (2016). Introduction to Time Series and Forecasting. In *Journal of the American Statistical Association* (Vol. 92, Nomor 440). <https://doi.org/10.2307/2965440>
- Tasna Yunita. (2020). Peramalan Jumlah Penggunaan Kuota Internet Menggunakan Metode Autoregressive Integrated Moving Average (ARIMA). *Journal of Mathematics: Theory and Applications*, 1(2), 16–22. <https://doi.org/10.31605/jomta.v2i1.777>
- Wei, W. W. S. (2006). Time Series Analysis Univariate and Multivariate Methods. In *Applied Time Series Analysis* (hal. 203–220). <https://doi.org/10.1201/b11459-9>