

RANCANG BANGUN SISTEM PREDIKSI HARGA BERAS TINGKAT GROSIR DI INDONESIA BERBASIS WEB MENGGUNAKAN METODE AUTOREGRESSIVE INTEGRATED MOVING AVERAGE (ARIMA)

Rahmad firdiansyah¹, Asmunin²

D4 Manajemen Informatika, Universitas Negeri Surabaya
Jl. Ketintang, Ketintang, Kec. Gayungan, Kota Surabaya

¹rahmadfirdiansyah.21023@mhs.unesa.ac.id

²asmunin@unesa.ac.id

Abstrak— Harga beras yang fluktuatif menjadi tantangan dalam distribusi dan ketahanan pangan di Indonesia. Penelitian ini bertujuan membangun sistem prediksi harga beras berbasis web menggunakan metode ARIMA dengan pendekatan Rapid Application Development (RAD). Data diperoleh dari sumber resmi dan diolah sebagai data deret waktu. Model ARIMA dipilih karena mampu menganalisis tren historis dan mengatasi data non-stasioner melalui teknik differencing. Sistem dikembangkan menggunakan Python dan framework Flask, serta menampilkan hasil prediksi dalam bentuk grafik dan evaluasi akurasi menggunakan MAPE, RMSE, dan MAE. Berdasarkan pengujian, model ARIMA(1,1,0) memberikan hasil terbaik dengan nilai MAPE 0,53% (5 bulan), 0,69% (1 tahun), dan 13,00% (5 tahun). Hasil ini menunjukkan kemampuan generalisasi yang baik untuk prediksi jangka pendek hingga panjang. Sistem ini bertujuan dapat membantu pengambilan keputusan terkait harga beras secara tepat dan responsif.

Kata kunci— Prediksi Harga Beras, ARIMA, Time series, Web-Based System, Rapid Application Development

Abstract— Fluctuating rice prices pose a challenge to food distribution and security in Indonesia. This study aims to develop a web-based rice price prediction system using the ARIMA method with a Rapid Application Development (RAD) approach. Data was obtained from official sources and processed as time series data. The ARIMA model was chosen because it can analyze historical trends and overcome non-stationary data through differencing techniques. The system was developed using Python and the Flask framework, and displays prediction results in graphical form and evaluates accuracy using MAPE, RMSE, and MAE. Based on testing, the ARIMA(1,1,0) model yielded the best results with MAPE values of 0.53% (5 months), 0.69% (1 year), and 13.00% (5 years). These results demonstrate good generalization capabilities for short- to long-term predictions. The system aims to assist in making timely and responsive decisions regarding rice prices.

Kata kunci— Rice Price Prediction, ARIMA, Time Series, Web-Based System, Rapid Application Development

I. PENDAHULUAN

Beras adalah salah satu jenis bahan makanan penting di Indonesia yang memainkan peranan krusial dalam mempertahankan kestabilan ekonomi dan kesejahteraan warga. Sebagai sumber makanan utama untuk sebagian besar masyarakat, kebutuhan akan beras

terus bertambah seiring dengan bertambahnya jumlah penduduk. Namun, ketidakstabilan harga beras yang selalu berubah tetap menjadi masalah utama, terutama bagi aktor-aktor dalam jaringan distribusi seperti petani, pengecer, distributor, serta pengambil keputusan. Fenomena ini menjadi lebih rumit ketika pada tahun 2023, Indonesia dinyatakan sebagai negara yang paling banyak mengimpor beras di dunia, dengan proporsi 5,46% (1,79 miliar dolar AS) dari total nilai impor secara global sebesar USD 32,78 miliar, yang dipengaruhi oleh dampak El Nino [1]. Ketidakpastian mengenai harga ini memengaruhi tantangan dalam merencanakan persediaan, manajemen distribusi, serta penetapan kebijakan harga di skala nasional. Untuk itu, sangat diperlukan adanya sistem yang mampu meramalkan nilai beras secara tepat guna membantu proses pengambilan keputusan yang lebih cepat serta didasarkan pada informasi yang valid.

Salah satu cara yang bisa diterapkan untuk memperkirakan harga beras adalah dengan menerapkan pendekatan *Autoregressive Integrated Moving Average* (ARIMA). Metode ini mampu melakukan analisis terhadap data deret waktu yang menunjukkan adanya tren. Dengan menggabungkan komponen *autoregresif* (AR), *differencing* (I), dan *moving average* (MA), ARIMA dapat memodelkan pola historis dan memproyeksikan nilai masa depan secara lebih akurat [2]. Proses *differencing* digunakan untuk mentransformasi data menjadi stasioner, yang merupakan keadaan di mana rata-rata dan variansi data tetap konstan seiring waktu—syarat esensial untuk memastikan keakuratan prediksi. Pada pengujian model, dilakukan penilaian akurasi dengan memanfaatkan metrik MAE, MAPE, dan RMSE, serta verifikasi visual melalui grafik hasil prediksi.

Penelitian ini mengembangkan sistem prediksi harga beras berbasis web dengan metode ARIMA, menggunakan pendekatan *Rapid Application Development* (RAD). RAD dipilih karena memungkinkan pembuatan perangkat lunak dengan cepat, adaptif, dan mengikutsertakan pengguna di setiap fase pengembangan sistem. Implementasi dilakukan menggunakan Python dan *framework* Flask, sementara data diperoleh dari sumber resmi seperti BPS. Berdasarkan temuan dari pengujian model, ARIMA(1,1,0) memberikan performa yang paling unggul dengan nilai MAPE sebesar 0,53% untuk prediksi 5 bulan, 0,69% untuk 1 tahun, dan 13,00% untuk 5 tahun.

Nilai ini mengindikasikan tingkat akurasi sangat baik untuk jangka pendek hingga menengah, dan cukup tepat untuk proyeksi jangka panjang. Konsistensi kinerja dari model ini menunjukkan potensinya dalam mengenali pola perubahan harga tanpa terjebak dalam *overfitting*, sehingga menjadikannya sebagai sarana yang tepat untuk membantu proses pengambilan keputusan terkait distribusi beras dan pengendalian harga pangan nasional.

II. TINJAUAN PUSTAKA

A. Time Series

Deret waktu atau *time series* adalah sekumpulan informasi yang dikumpulkan dalam jadwal tertentu atau interval waktu yang berulang. Deret waktu mengacu pada serangkaian pengamatan yang dalam interval waktu tertentu, biasanya. Dalam periode singkat seperti jam, hari, atau minggu [3].

B. Forecasting

Forecasting adalah proses untuk menilai atau meramalkan situasi ke depan, pemanfaatan informasi historis yang dikombinasikan dengan data terkini akan menjadi landasan pengambilan keputusan yang diperlukan agar dapat menentukan waktu terjadinya suatu peristiwa, sehingga dapat diambil langkah yang sesuai [4].

C. Website

Website adalah halaman yang menyediakan informasi di jalur internet dengan cara yang efisien dan pintar agar dapat diakses secara global melalui jaringan internet, yang mencakup tulisan, gambar, dan suara animasi.[5]. *Website* juga berfungsi sebagai salah satu media yang menyajikan informasi terkini dengan cepat dan akurat. Dalam situasi ini, situs *web* juga dapat dimanfaatkan untuk mengelola informasi seperti data karyawan.. [6].

D. Python

Python merupakan bahasa pemrograman yang gampang dipahami dan memiliki struktur yang tidak rumit.. Bahasa ini dikembangkan oleh seorang bernama Guido van Rossum mengenalkan Python pertama kali di tahun 1991 [7]. Bahasa Python sering dimanfaatkan untuk pengembangan aplikasi web, perangkat lunak, serta dalam bidang ilmu data dan pembelajaran mesin [8].

E. Flask

Flask Termasuk kategori kerangka kerja mikro untuk aplikasi *web* yang dibangun dengan bahasa Python. Disebut sebagai *microframework* karena Flask dirancang untuk memberikan inti yang sederhana namun dapat diperluas dengan berbagai ekstensi sesuai kebutuhan pengembang [9].

F. Rapid Application Development (RAD)

Rapid Application Development merupakan suatu pendekatan dalam merancang dan membangun sistem yang diterapkan dalam penelitian ini, merupakan bagian dari siklus hidup pengembangan sistem (SDLC). RAD adalah

pendekatan dalam pengembangan perangkat lunak yang mengutamakan proses pengembangan yang cepat, cepat dan berurutan.

Pendekatan ini efektif digunakan untuk menciptakan sistem informasi yang efisien dari segi waktu, keakuratan, dan penghematan biaya. Penggunaan metodologi RAD dipilih karena menawarkan sejumlah manfaat, antara lain siklus pengembangan yang lebih pendek, fleksibilitas yang lebih besar, peningkatan keterlibatan pengguna, dan risiko kesalahan dalam proses pengembangan yang lebih rendah.



Gambar 1 Tahapan Metodologi RAD

Sumber : [10]

G. Autoregressive Integrated Moving Average (ARIMA)

ARIMA (*Autoregressive Integrated Moving Average*) merupakan salah satu metode peramalan pada analisis deret waktu yang mengintegrasikan tiga komponen inti *autoregressive* (AR), *moving average* (MA), dan *differencing* (integrated) guna mencapai kondisi stasioner. Model ini diterapkan untuk mengelola data deret waktu yang awalnya tidak stasioner sehingga dapat dikonversikan menjadi stasioner melalui proses perbedaan. kemudian menerapkan model AR dan MA untuk peramalan [11].

H. Model ARIMA

1. Autoregressive Model (AR)

Autoregressive adalah jenis regresi yang menghubungkan nilai-nilai sebelumnya dengan dirinya sendiri berdasarkan berbagai selang waktu. Model *Autoregressive* ini menggambarkan prediksi Sebagai fungsi dari nilai sebelumnya dalam urutan waktu tertentu. Bentuk standar dari model ini adalah model AR(p) dengan urutan p atau model ARIMA(p,0,0). Model umum AR(p) dapat ditemukan pada Persamaan 1:

$$Y_t = \phi_0 + \phi_1 Y_{t-1} + \phi_2 Y_{t-2} + \dots + \phi_p Y_{t-p} + e_t \quad (1)$$

Sumber : [12]

Y_t = deret waktu stasioner
 $Y_{t-1}, Y_{t-2}, \dots, Y_{t-p}$ = Variabel respons pada setiap interval waktu $t - 1, t - 2, \dots, t - p$. Nilai Y berperan sebagai variabel bebas.

ϕ_0 = Suatu konstanta

ϕ_p = parameter *autoregresif* ke-p

e_t = Galat pada waktu t yang mewakili pengaruh dari variabel-variabel yang tidak dimasukkan dalam model.

2. Moving Average Model (MA)

Bentuk umum model moving average orde q (MA(q)) atau ARIMA (0,0,q) dinyatakan pada Persamaan 2 sebagai berikut:

$$Y_t = \theta_0 + e_t - \theta_1 e_{t-1} - \theta_2 e_{t-2} - \dots - \theta_p e_{t-q} \quad (2)$$

Sumber : [12]

Y_t = Deret waktu stasioner
 θ_0 = Konstanta
 $\theta_1, \dots, \theta_q$ = Parameter-parameter *moving average* yang menunjukkan bobot.
 e_{t-q} = Nilai kesalahan pada saat $t - k$.

3. Autoregressive Moving Average (ARMA)

Model *Autoregressive Moving Average* (ARMA) merupakan suatu kombinasi dari model AR dan MA. Bentuk umum dari model ARMA(p,q) :

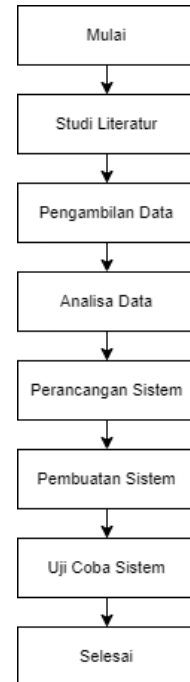
$$Y_t = \gamma_0 + \partial_1 Y_{t-1} + \partial_2 Y_{t-2} + \dots + \partial_n Y_{t-p} - \lambda_1 e_{t-1} - \lambda_2 e_{t-2} - \lambda_n e_{t-q} \quad (3)$$

Sumber : [12]

di mana Y_t merupakan deret waktu yang bersifat stasioner dan e_t sebagai kesalahan. Apabila model tersebut menerapkan dengan 2 lag pada variable dependen dan tiga lag pada residual, model ini dintakan sebagai ARMA. Penambahan preses penstasioneran data akan mengubah model ARMA tersebut menjadi bentuk umum ARIMA (p,d,q)

III. METODE PENELITIAN

Pada bagian ini berisi tentang alur dari rancangan penelitian yang akan dilakukan oleh penulis. Dengan mengimplementasikan metodologi *Rapid Application Development* pada perancangan Sistem Informasi Prediksi Harga Beras tingkat grosir di Indonesia untuk mencapai hasil yang diinginkan untuk gambaran lebih lanjut bisa di lihat di gambar 2.



Gambar 2 Prosedur Alur Penelitian

A. Studi Literatur

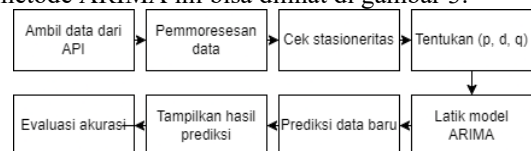
Pada tahap ini, dilakukan kajian teori terkait sistem prediksi harga beras dengan menggunakan metode *Autoregressive Integrated Moving Average* (ARIMA). Metode dipilih karena mampu menangani pola tren dan ketidakstasioneran dalam data deret waktu dengan menerapkan proses *differencing*. Metode ini lebih unggul dibandingkan metode konvensional [13].

B. Pengumpulan Data

Sumber informasi harga beras dalam penelitian ini diambil dari data set di situs www.bps.go.id. Data harga beras yang diambil adalah rata-rata harga beras di tingkat perdagangan besar (Grosir) Indonesia 2025.

C. Analisis Prediksi Metode *Autoregressive Integrated Moving Average* (ARIMA)

Pada bagian ini, analisis data jumlah harga beras akan dilakukan menggunakan metode ARIMA. Adapun tahapan dalam metode ARIMA ini bisa dilihat di gambar 3.



Gambar 3 Proses Alur Metode ARIMA

1. Ambil Data Dari API

Sistem akan merespons permintaan ini dengan mengambil data historis harga beras dari API, dari situs Badan pusat statistik Indonesia. Data yang diambil umumnya berupa data deret waktu (*time series*) dengan format JSON yang kemudian diubah menjadi format tabel atau dataframe.

2. Pemrosesan Data

sistem melakukan proses *preprocessing* untuk menyiapkan data tersebut agar dapat digunakan dalam model prediksi. Tahapan *preprocessing* ini meliputi konversi kolom tanggal ke format waktu, pengurutan berdasarkan waktu.

3. Cek Stasioneritas

Pengujian untuk menentukan apakah data telah stasioner atau tidak. Stasioner artinya data sudah stabil, tidak naik-turun drastis lagi, sehingga bisa dianalisis dengan metode statistik tertentu seperti ARIMA. maka sistem akan melakukan *differencing* atau pengurangan nilai antar waktu agar data menjadi stasioner, sesuai kebutuhan model ARIMA.

4. Membangun Model ARIMA

Setelah data siap, sistem mulai membangun model ARIMA dengan parameter yang telah ditentukan, misalnya ARIMA (0,1,1). Parameter ini berarti model tidak memiliki komponen *autoregressive* ($p=0$), melakukan satu kali *differencing* ($d=1$), dan memiliki satu komponen *moving average* ($q=1$). Model kemudian dilatih menggunakan data historis yang telah diproses sebelumnya.

5. Prediksi Harga

Terakhir, hasil prediksi ditampilkan kepada pengguna dalam bentuk grafik, tabel, atau bahkan file yang dapat diunduh.

6. Evaluasi Akurasi

Setelah model ARIMA dilatih dan menghasilkan prediksi, langkah penting selanjutnya adalah mengevaluasi akurasi model.

Evaluasi performa dilakukan secara manual menggunakan tiga metrik utama MAPE, RMSE, MAE. Nilai MAPE yang sangat rendah menunjukkan bahwa model memiliki tingkat kesalahan prediksi yang rendah terhadap data yang sebenarnya. Dengan hasil tersebut, model ARIMA yang menunjukkan nilai kesalahan paling rendah. dinilai cukup optimal dan layak digunakan untuk melakukan prediksi harga beras dalam sistem yang dibangun.

Berikut adalah formula yang digunakan untuk mengukur performa model:

a. Mean Absolute Percentage Error (MAPE)

MAPE digunakan untuk menilai persentase kesalahan rata-rata dari hasil prediksi dibandingkan nilai yang sebenarnya. Rumus perhitungannya ditunjukkan pada Persamaan 4:

$$MAPE = \frac{100\%}{n} \sum_{t=1}^n \left| \frac{y_t - \hat{y}_t}{y_t} \right| \quad (4)$$

Sumber:[14]

Di mana:

Y_t = nilai aktual pada periode ke-t

$\hat{Y}_t$ = nilai hasil prediksi pada periode ke-t

n = jumlah total data pengamatan

Tujuan MAPE digunakan untuk menunjukkan rata-rata kesalahan dalam bentuk persentase, sehingga memudahkan interpretasi secara umum. Nilai MAPE yang semakin kecil menandakan akurasi model yang meningkat.

b. Root Mean Square Error (RMSE)

RMSE digunakan untuk mengukur seberapa besar deviasi prediksi dibandingkan nilai sebenarnya dengan mempertimbangkan kuadrat dari selisihnya. Rumus perhitungannya ditunjukkan pada Persamaan 5:

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

Sumber :[14]

Tujuan RMSE adalah untuk memperhatikan kesalahan besar (*outlier*), sehingga cocok untuk digunakan pada situasi tersebut. mengetahui kestabilan performa prediksi dalam jangka panjang.

c. Mean Absolute Error (MAE)

MAE menghitung rata-rata nilai absolut dari kesalahan antara hasil prediksi dan data yang sebenarnya. Rumus perhitungannya ditunjukkan pada Persamaan 6:

$$MAE = \frac{1}{n} \sum_{t=1}^n |y_t - \hat{y}_t| \quad (6)$$

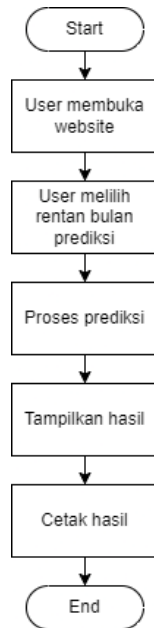
Sumber :[14]

Tujuan MAE menunjukkan seberapa besar rata-rata kesalahan absolut yang terjadi dalam unit yang sama dengan data, dengan memberikan bobot yang sama pada setiap kesalahan.

D. Perancangan Sistem Dan Pembahasan Model ARIMA

Berikut adalah rancangan desain dan alur kerja yang digunakan sebagai acuan pengembangan sistem informasi.

1. Alur Kerja Sistem Informasi



Gambar 4 Alur Kerja Sistem

Gambar 4 menunjukkan alur sistem prediksi harga pada aplikasi *web*, dimulai dari pengguna yang membuka *website* dan memilih rentang waktu. Pilihan ini menjadi input bagi sistem untuk memproses data historis harga dari API menggunakan model ARIMA.

2. Desain Interface

Pada bagian ini ditampilkan desain antarmuka pengguna yang digunakan sebagai prototipe untuk ditunjukkan kepada pihak terkait dan juga sebagai acuan dalam membangun sistem prediksi harga beras. Desain antarmuka ini dirancang oleh penulis menggunakan aplikasi Figma agar sesuai dengan kebutuhan pengguna.

a. Tampilan Utama

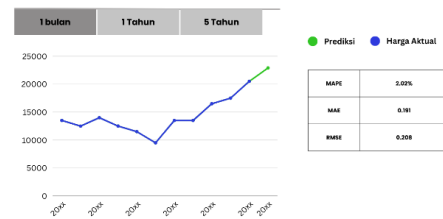
Pada gambar 5 ini menunjukkan desain contoh interface dari halaman utama dari sistem informasi yang berisi grafik harga beras saat ini.



Gambar 5 Interface Halaman Utama

b. Tampilan Hasil Grafik Dan Evaluasi

Pada gambar 6 ini menunjukkan desain *interface* dari hasil prediksi.



Gambar 6 Interface Halaman Evaluasi

c. Tampilan Hasil Prediksi

Pada gambar 7 ini menunjukkan desain *interface* dari hasil prediksi harga beras tahun yang di prediksi dengan tombol cetak.

Hasil Prediksi

20xx - xx	12.0000	20xx - xx	12.0000
20xx - xx	12.0000	20xx - xx	12.0000
20xx - xx	12.0000	20xx - xx	12.0000
20xx - xx	12.0000	20xx - xx	12.0000
20xx - xx	12.0000	20xx - xx	12.0000
20xx - xx	12.0000	20xx - xx	12.0000
20xx - xx	12.0000	20xx - xx	12.0000

Cetak

Gambar 7 Interface Halaman Hasil Prediksi

E. Pembuatan Sistem Prediksi

Pada tahap penerapan sistem dilakukan melalui proses kompilasi kode atau coding menggunakan *framework* Flask untuk membuat *website*.

F. Pengujian sistem

Pada fase ini, sistem ini diuji untuk menjamin bahwa seluruh fitur dalam penerapan beroperasi dengan baik sesuai desain sistem yang sudah dibuat.

IV. HASIL PENELITIAN DAN PEMBAHASAN

A. Tahapan Pengembangan Sistem Prediksi

Tahapan ini menjelaskan seluruh proses yang dilakukan dalam membangun sistem prediksi harga beras, mulai dari pengumpulan data hingga implementasi sistem berbasis *web*. Pengembangan sistem dilakukan melalui beberapa langkah utama, yaitu:

1. Pengumpulan Data, yaitu proses memperoleh data historis harga beras dari sumber yang valid dan terpercaya.
2. Proses Pengambilan Data Dari API menggunakan *library requests*.
3. Preprocessing Data, mencakup pengolahan awal seperti pengubahan data ke format *time series*, penanganan data hilang, dan visualisasi awal.
4. Uji Stasioneritas dan *Differencing*, bertujuan untuk memastikan data memenuhi syarat stasioneritas yang dibutuhkan oleh model ARIMA.
5. Identifikasi Model ARIMA, dengan menganalisis grafik ACF dan PACF serta mengevaluasi beberapa kombinasi parameter.

6. Implementasi *web*, yaitu tahap integrasi model ke dalam aplikasi berbasis *web* menggunakan framework Flask agar dapat diakses dan digunakan secara interaktif oleh pengguna.

- a. Pengumpulan data

Tabel 1 menunjukkan sampel data harga beras rata-rata bulanan selama satu tahun, yaitu pada tahun 2024. Data ini diberikan dari situs Badan Pusat Statistik (BPS) dan digunakan sebagai representasi dari keseluruhan data periode 2010 hingga 2025 yang dijadikan dasar dalam proses pemodelan prediksi

TABEL 1

Sampel Data Rata-Rata Harga Beras Bulanan 2024

Tanggal	Harga (Rp/kg)
Januari 2024	13.588,00
Februari 2024	14.397,00
Maret 2024	14.528,00
April 2024	13.902,00
Mei 2024	13.471,00
Juni 2024	13.433,00
Juli 2024	13.571,00
Agustus 2024	13.600,00
September 2024	13.611,00
Oktober 2024	13.563,00
November 2024	13.453,00
Desember 2024	13.486,00

- b. Proses Pengambilan Data Dari API

Pada sistem ini, data harga beras diperoleh secara otomatis melalui permintaan (*request*) ke API yang tersedia. Pengambilan data dilakukan menggunakan permintaan HTTP GET yang akan mengakses *endpoint* API dan mengembalikan data dalam format JSON. Proses ini diintegrasikan dalam sistem agar data dapat diperbarui secara berkala tanpa perlu dilakukan input manual.

Contoh kode di bawah digunakan untuk mengambil data harga beras secara otomatis dari API resmi BPS menggunakan parameter tahun. Data yang diterima dalam format JSON kemudian diolah menjadi bentuk *DataFrame* menggunakan pustaka Pandas.

```
import requests
import pandas as pd

url
= "https://webapi.bps.go.id/v1/api/list/model/data
/lang/ind/domain/0000/var/295/th/125/key/[Web
API_KEY]"

response = requests.get(url)
```

- c. *Preprocessing* Data

Setelah data harga beras berhasil diambil dari API milik Badan Pusat Statistik (BPS), langkah selanjutnya adalah melakukan *preprocessing* agar data siap digunakan untuk pemodelan. Data yang diperoleh dalam format JSON kemudian diubah menjadi struktur *DataFrame* menggunakan pustaka *pandas*.

- d. Uji Stasioneritas dan *Differencing*

Jika data tidak stasioner (misalnya memiliki tren yang meningkat atau menurun), maka model ARIMA tidak dapat menghasilkan prediksi yang tepat. Dengan demikian, sebelum membangun model, dilakukan pengujian stasioneritas menggunakan *Tes Augmented Dickey-Fuller* (ADF). Apabila pengujian mengindikasikan bahwa data belum berada pada kondisi stasioner, maka dilakukan preses perbedaan, yaitu menghitung selisih antara nilai saat ini dan nilai sebelumnya. berikut adalah contoh kode untuk melakukan uji ADF:

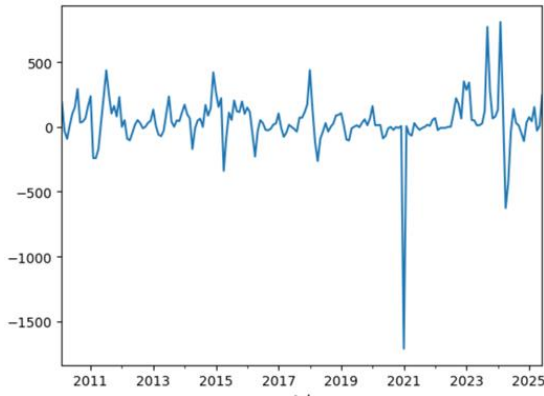
```
# Cek adfuller
result = adfuller(df.harga.dropna())
print('ADF Statistic: %f' % result[0])
print('p-value: %f' % result[1])
```

Berdasarkan hasil uji *Augmented Dickey-Fuller* (ADF) sebelum dilakukan *differencing*, diperoleh nilai p-value sebesar 0.649157. Nilai ini jauh di atas ambang batas signifikansi 0.05, yang berarti data masih bersifat nonstasioner. Dengan demikian, diperlukan proses *differencing* sebanyak yang dibutuhkan sehingga tren dalam data dapat dihilangkan berikut adalah potongan kode yang digunakan untuk melakukan *differencing*:

```
# Differencing pertama
df_harga_diff = df.harga.diff().dropna()

# Visualisasi data setelah differencing
df_harga_diff.plot()
```

Hasil visualisasi pada gambar 13 menunjukkan bahwa tren naik atau turun yang sebelumnya ada dalam data telah berkurang secara signifikan, yang mengindikasikan bahwa data sudah lebih mendekati kondisi stasioner.



Gambar 13 Grafik Stasioner

Setelah proses *differencing* dilakukan satu kali, uji Augmented Dickey-Fuller (ADF) diulang. Hasilnya menunjukkan *p-value* sebesar 4.93×10^{-18} , yang jauh di bawah tingkat signifikansi 0.05 yang berarti bahwa data telah berhasil diubah menjadi stasioner, sehingga dapat digunakan untuk proses pemodelan menggunakan ARIMA.

e. Identifikasi Model ARIMA

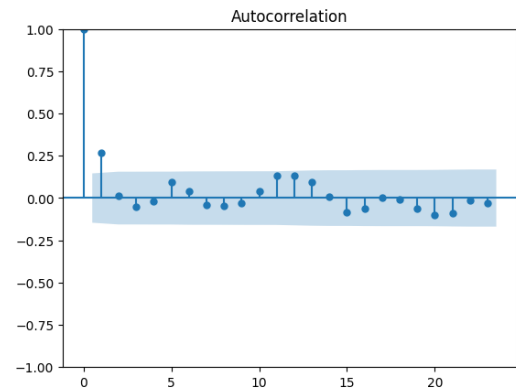
Setelah data dinyatakan stasioner melalui proses *differencing*, langkah berikutnya adalah mengidentifikasi model ARIMA dengan menganalisis grafik ACF (*Autocorrelation Function*) dan PACF (*Partial Autocorrelation Function*). Grafik ini berfungsi untuk menentukan nilai parameter *p* (*autoregressive*) dan *q* (*moving average*) yang optimal, berikut adalah kode yang digunakan untuk menampilkan grafik ACF dan PACF:

```
from statsmodels.graphics.tsaplots import plot_acf,
plot_pacf

acf_diff = plot_acf(df_harga_diff)

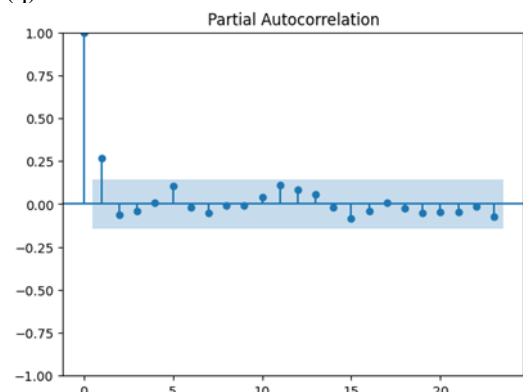
pacf_diff = plot_pacf(df_harga_diff)
```

Grafik ACF menunjukkan keterkaitan antara suatu titik data dengan lag sebelumnya dan membantu dalam menentukan nilai parameter *q*. Sementara itu, grafik PACF menunjukkan hubungan parsial antar lag dan digunakan untuk menentukan nilai parameter *p*. Pola yang terlihat dari kedua grafik ini akan menjadi dasar dalam menentukan beberapa kandidat model ARIMA.



Gambar 14 Grafik ACF

Berdasarkan grafik 14 ACF digunakan untuk melihat apakah data masa lalu berpengaruh terhadap data saat ini. Jika iya, berarti datanya bisa diramal terlihat, *spike* auto korelasi yang signifikan pada Lag 1, dan kemudian autokorelasi secara cepat jatuh ke dalam batas signifikansi (area biru). Ini mengindikasikan adanya *cut-off* pada Lag 1, yang menyarankan orde *Moving Average* non-musiman (*q*) sebesar 1.



Gambar 15 Grafik PACF

sedangkan pada grafik 15 PACF sama seperti grafik ACF ada *spike* autokorelasi yang signifikan pada Lag 1, dan kemudian sebagian besar *spike* berada dalam area biru. Oleh karena itu model menyarankan orde *Autoregressive* non-musiman (*p*) sebesar 1. Hal ini mengindikasikan bahwa kemungkinan model yang sesuai adalah ARIMA(0,1,1), (1,1,0), (1,1,1), dan (2,1,2).

Selanjutnya, dilakukan pengujian terhadap beberapa kombinasi model ARIMA, antara lain ARIMA(0,1,1), ARIMA(1,1,1), ARIMA(1,1,0), dan ARIMA(2,1,2). Pengujian dilakukan menggunakan paket SARIMAX dari pustaka statsmodels di Python. Kriteria dalam memilih model terbaik didasarkan pada nilai AIC (*Akaike Information Criterion*) terendah serta evaluasi akurasi prediksi seperti MAPE, RMSE, dan MAE.

Berikut ringkasan hasil evaluasi terhadap beberapa model ARIMA:

TABEL 2 Evaluasi Model 1 bulan

Model	AIC	MAPE	RMSE	MAE
(0,1,1)	2463.258	1.48%	206.89	206.89
(1,1,1)	2465.093	1.50%	210.15	210.15
(1,1,0)	2463.686	1.53%	214.23	214.23
(1,1,2)	2466.966	1.52%	212.80	212.80
(2,1,1)	2466.688	1.52%	212.40	212.40
(2,1,2)	2467.678	1.58%	221.23	221.23

TABEL 3 Evaluasi Model 5 bulan

Model	AIC	MAPE	RMSE	MAE
(0,1,1)	2413.173	0.56%	112.37	78.77
(1,1,1)	2415.011	0.55%	110.28	76.70
(1,1,0)	2413.608	0.53%	108.40	74.59
(1,1,2)	2416.881	0.54%	109.39	75.96
(2,1,1)	2416.593	0.53%	107.19	73.74
(2,1,2)	2417.646	0.54%	188.178	107.33

TABEL 4 Evaluasi Model 1 Tahun

Model	AIC	MAPE	RMSE	MAE
(0,1,1)	2325.508	0.76%	114.76	103.49
(1,1,1)	2327.295	0.74%	111.97	101.18
(1,1,0)	2325.790	0.69%	104.57	93.82
(1,1,2)	2329.214	0.76%	114.97	103.70
(2,1,1)	2328.992	0.91%	143.47	123.24
(2,1,2)	2330.162	1.34%	207.96	181.99

TABEL 5 Evaluasi Model 5 Tahun

Model	AIC	MAPE	RMSE	MAE
(0,1,1)	1535.874	13.58%	1768.17	1524.29
(1,1,1)	1535.651	13.34%	1742.41	1496.18
(1,1,0)	1537.442	13.00%	1704.99	1456.07
(1,1,2)	1537.308	13.35%	1742.95	1496.68
(2,1,1)	1531.669	14.52%	1872.36	1636.02
(2,1,2)	1531.368	13.86%	1798.61	1557.62

Evaluasi model prediksi dilakukan dengan membandingkan performa ARIMA pada empat rentang waktu berbeda, yaitu prediksi selama periode 1 bulan, 5 bulan, 1 tahun, serta 5 tahun mendatang.

Rangkuman model terbaik berdasarkan jangka waktu prediksi:

- ARIMA (0,1,1) → Terbaik untuk prediksi jangka sangat pendek (1 bulan).
- ARIMA (2,1,1) → Terbaik untuk prediksi jangka pendek (5 bulan).
- ARIMA (1,1,0) → Terbaik untuk prediksi jangka menengah (1 tahun).
- ARIMA (1,1,0) → Cenderung lebih stabil untuk prediksi jangka panjang (5 tahun) meskipun akurasi menurun.

f. Evaluasi Model ARIMA

Evaluasi akurasi dilakukan dengan menghitung tiga metrik utama, yaitu *Mean Absolute Error* (MAE), *Mean Absolute Percentage Error* (MAPE), dan *Root Mean Squared Error* (RMSE) baik dalam jangka pendek (1 bulan), Menengah (1 tahun), Maupun Panjang (5 tahun). Hasil evaluasi mengenai model ARIMA(1,1,0) dapat dilihat pada tabel 6 berikut:

TABEL 6 Evaluasi Model (1,1,0)

Jangka Waktu	Metrik	Nilai
1 Bulan Kedepan	MAPE	1.53%
	RMSE	214.23
	MAE	214.23
5 Bulan Kedepan	MAPE	0.54%
	RMSE	108.4
	MAE	74.59
1 Tahun Kedepan	MAPE	0.69%
	RMSE	104.56
	MAE	93.81
5 Tahun Kedepan	MAPE	13.0%
	RMSE	1704.69
	MAE	1455.74

Evaluasi performa model ARIMA (1,1,0) dilakukan untuk empat rentang waktu prediksi. Berdasarkan hasil evaluasi yang ditampilkan pada Tabel 6, terlihat bahwa model ini memberikan hasil yang sangat baik pada prediksi jangka pendek hingga menengah.

B. Implementasi Sistem

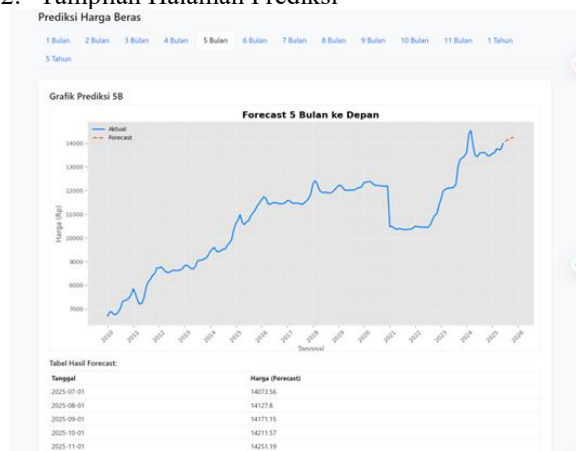
1. Tampilan Halaman Utama



Gambar 16 Tampilan Halaman Utama

Gambar 16 adalah halaman utama yang menampilkan grafik harga beras aktual yang baru.

2. Tampilan Halaman Prediksi



Gambar 17 Tampilan Halaman Prediksi

Pada Gambar 17 ditampilkan halaman prediksi harga beras untuk periode 5 bulan ke depan.

3. Tampilan Halaman Evaluasi Prediksi

Tabel Evaluasi:				
No	Tanggal	Data Aktual	Data Prediksi	Error
1	2025-09-01	12,979.00	12,744.77	-234.23

Nilai Evaluasi:

- MAPE: 1.53%
- RMSE: 214.23
- MAE: 714.23

Gambar 18 Tampilan Halaman Evaluasi Prediksi

Gambar 18 menunjukkan hasil evaluasi prediksi harga beras untuk periode 5 bulan ke depan.

C. Hasil Prediksi

Tabel 7 berikut merupakan hasil prediksi harga beras untuk 5 bulan ke depan dengan penerapan model ARIMA(1,1,0):

TABEL 7 Hasil Prediksi Beras 1 Tahun

Tanggal	Harga (Forecast)
2025-07-01	14073.56
2025-08-01	14127.8
2025-09-01	14171.15
2025-10-01	14211.57
2025-11-01	14251.19

Tanggal	Harga (Forecast)
2025-12-01	14290.6
2026-01-01	14329.96
2026-02-01	14369.3
2026-03-01	14408.63
2026-04-01	14447.96
2026-05-01	14487.3
2026-06-01	14526.63

V. PENUTUP

A. Kesimpulan

Berdasarkan hasil penelitian dan pengujian sistem prediksi harga beras berbasis web menggunakan metode ARIMA (*Autoregressive Integrated Moving Average*), maka dapat disimpulkan sebagai berikut:

1. Prediksi harga beras dengan metode ARIMA dapat dilakukan secara akurat. Dengan menerapkan proses identifikasi, *differencing*, serta analisis ACF dan PACF, model yang terbukti paling optimal adalah ARIMA(1,1,0). Model ini dipilih berdasarkan evaluasi menggunakan metrik akurasi seperti MAPE, RMSE, dan MAE, yang menunjukkan nilai kesalahan yang rendah dalam dalam ke 4 rentan waktu.
2. Tren harga beras dapat diidentifikasi dengan menganalisis data historis. Dengan memanfaatkan data bulanan yang diperoleh dari situs resmi BPS, dilakukan analisis tren melalui visualisasi grafik serta proses stasionerisasi data. Hasilnya menunjukkan adanya pola kenaikan dan penurunan harga yang dapat dimodelkan secara statistik untuk prediksi ke depan.
3. Sistem prediksi yang dibangun bermanfaat dalam pengambilan keputusan. Sistem yang dikembangkan menggunakan bahasa pemrograman Python dengan *framework* Flask berbasis *web* ini memberikan informasi prediktif yang dapat dijangkau secara mudah oleh pengguna. Data tersebut dapat membantu pelaku usaha, petani, distributor, maupun pemerintah dalam mengantisipasi fluktuasi harga, merencanakan distribusi, dan menyusun kebijakan yang lebih tepat berdasarkan proyeksi harga yang ditampilkan sistem.

B. Saran

Untuk pengembangan selanjutnya, peneliti memberikan beberapa rekomendasi sebagai berikut :

1. Penambahan variabel eksternal Di masa mendatang, sistem dapat dikembangkan lebih lanjut menggunakan metode ARIMAX atau SARIMAX dengan mempertimbangkan variabel eksternal seperti data produksi padi, curah hujan, inflasi, dan nilai tukar rupiah, agar prediksi lebih komprehensif.
2. Pengembangan UI/UX Sistem web yang telah dibangun dapat ditingkatkan lagi dari segi tampilan

dan aksesibilitas, agar lebih mudah digunakan oleh pengguna non-teknis seperti petani atau pedagang kecil.

REFERENSI

- [1] T. Pipit Mulyah, Dyah Aminatun, Sukma Septian Nasution, Tommy Hastomo, Setiana Sri Wahyuni Sitepu, “*濟無*No Title No Title,” *J. GEEJ*, vol. 7, no. 2, 2020.
- [2] S. A. Sinaga, “Implementasi Metode Arima (Autoregressive Moving Average) Untuk Prediksi Penjualan Mobil,” *J. Glob. Technol. Comput.*, vol. 2, no. 3, pp. 102–109, 2023, doi: 10.47065/jogtc.v2i3.4013.
- [3] M. Heru Widiyanto, R. Mayasari, and G. Garno, “Implementasi Time Series Pada Data Penjualan Di Gaikindo Menggunakan Algoritma Seasonal Arima,” *JATI (Jurnal Mhs. Tek. Inform.*, vol. 7, no. 3, pp. 1501–1506, 2023, doi: 10.36040/jati.v7i3.6879.
- [4] M. Afrizal, S. Siswanto, and A. Sudarsono, “Implementation of Forecasting Sales of Electronic Goods Using the Semi Average Method at Alex Electronics Store,” *J. Media Comput. Sci.*, vol. 2, no. 2, pp. 319–330, 2023, doi: 10.37676/jmcs.v2i2.4433.
- [5] J. A. Fransiskus Ermando Lado, “Ende Selatan Berbasis Web,” vol. 12, pp. 69–77, 2021.
- [6] E. Nurlailah and K. R. Nova Wardani, “Perancangan Website Sebagai Media Informasi Dan Promosi Oleh-Oleh Khas Kota Pagaralam,” *JUPI (Jurnal Ilm. Penelit. dan Pembelajaran Inform.*, vol. 8, no. 4, pp. 1175–1185, 2023, doi: 10.29100/jupi.v8i4.4006.
- [7] A. Ma’arif, “Buku Ajar Pemrograman Lanjut Bahasa Pemrograman Python Oleh : Alfian Ma ’ Arif,” *Univ. Ahmad Dahlan*, p. 62, 2020, [Online]. Available: http://eprints.uad.ac.id/32743/1/buku_python.pdf
- [8] S. Rahman *et al.*, *Python : Dasar Dan Pemrograman Berorientasi Objek*. 2023.
- [9] E. H. Y. K. Eben Haezer and N. Setiyawati, “Pembangunan Aplikasi Virtual Inventory System (Vis) Berbasis Web Menggunakan Flask Framework,” *J. Mnemon.*, vol. 5, no. 2, pp. 128–135, 2022, doi: 10.36040/mnemonic.v5i2.4799.
- [10] S. Sardiarinto, E. Saputro, A. Syukron, V. Ma’arif, and H. M. Nur, “Rancang Bangun Sistem Informasi Pembayaran Tagihan Pamsimas Menggunakan Metode Rapid Application Development,” *J. Teknol. Inf. dan Terap.*, vol. 10, no. 2, pp. 121–125, 2023, doi: 10.25047/jtit.v10i2.336.
- [11] A. R. Anandayani, D. K. A. Astutik, N. Bariroh, and A. Indrasetyaningsih, “Prediksi Rata-Rata Harga Beras Yang Dijual Oleh Pedagang Besar (Grosir) Menggunakan Metode Arima Box Jenkins,” *Teknosains Media Inf. Sains Dan Teknol.*, vol. 15, no. 2, p. 151, 2021, doi: 10.24252/teknosains.v15i2.17721.
- [12] N. Ulinnuha and Y. Farida, “Prediksi Cuaca Kota Surabaya Menggunakan Autoregressive Integrated Moving Average (Arima) Box Jenkins dan Kalman Filter,” *J. Mat. “MANTIK,”* vol. 4, no. 1, pp. 59–67, 2018, doi: 10.15642/mantik.2018.4.1.59-67.
- [13] E. D. Tarigan, “Peramalan Harga Beras di Indonesia Dengan ARIMA,” *Sepren*, vol. 5, no. 02, pp. 117–126, 2024, doi: 10.36655/sepren.v5i02.1508.
- [14] M. S. Hersyaputra, M. Triyanda, T. Jaya, and U. L. Yuhana, “Prediksi Man-Hours Menggunakan Analisis Regression dan Cyclomatic Complexity,” vol. 11, no. 1, pp. 100–108, 2025.