



Tersedia Online : <http://e-journals.unmul.ac.id/>

**ADOPSI TEKNOLOGI DAN SISTEM INFORMASI (ATASI)**

Alamat Jurnal : <http://e-journals2.unmul.ac.id/index.php/atasi/index>



## Implementasi Sistem Prediksi Pemesanan Stok Barang Menggunakan Algoritma SARIMA Dan Reorder Point (Studi Kasus Devie's Print Copy Center & Digital Printing)

Ajeng Ayu Arumsari <sup>1\*)</sup>, Dwi Hartanti <sup>2)</sup>, Intan Oktaviani <sup>3)</sup>

Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Duta Bangsa Surakarta

E-Mail : [ajengayuarumsari@gmail.com](mailto:ajengayuarumsari@gmail.com) <sup>1)</sup>; [dwhartanti@udb.ac.id](mailto:dwhartanti@udb.ac.id) <sup>2)</sup>; [intan\\_oktaviani@udb.ac.id](mailto:intan_oktaviani@udb.ac.id) <sup>3)</sup>;

### ARTICLE INFO

**Article history:**

Received : 08 May 2025

Revised : 10 May 2025

Accepted : 16 May 2025

Available online : 10 June 2025

**Keywords:**

*Digital Printing*

*Historical Sales*

*Reorder Point*

*Safety Stock*

*SARIMA*

### ABSTRACT

Proper stock management is a key factor in the digital printing industry to avoid operational disruptions caused by stock shortages or surpluses. Devie's Print Copy Center & Digital Printing faces challenges in accurately determining the quantity and timing of stock orders, especially due to seasonal demand fluctuations and procurement lead times. This study aims to develop a stock demand prediction system using the Seasonal Autoregressive Integrated Moving Average (SARIMA) algorithm combined with the Reorder Point (ROP) and Safety Stock methods. Historical sales data from January 2021 to April 2025 were analyzed, taking into account seasonal demand increases in June and December. The forecasting process using SARIMA produced the best model with parameters (0, 1, 0)(0, 0, 2, 12) and a Root Mean Square Error (RMSE) of 1174.81. The developed web-based prediction system features an interactive dashboard that provides automatic notifications when stock levels fall below the ROP threshold, along with recommendations for minimum order quantities. The forecasting results show that the SARIMA model can accurately predict stock needs, thereby helping improve stock management effectiveness and supporting the continuity of digital printing business operations more optimally.

### ABSTRAK

Manajemen stok yang tepat menjadi kunci utama dalam industri digital printing untuk menghindari gangguan operasional akibat kekurangan atau kelebihan persediaan. Devie's Print Copy Center & Digital Printing menghadapi tantangan dalam menentukan jumlah dan waktu pemesanan stok secara akurat, terutama karena adanya fluktuasi permintaan musiman dan waktu tunggu pengadaan barang. Penelitian ini bertujuan untuk mengembangkan sistem prediksi kebutuhan stok berbasis algoritma *Seasonal Autoregressive Integrated Moving Average* (SARIMA) yang dikombinasikan dengan metode *Reorder Point* (ROP) dan *Safety Stock*. Data penjualan historis dari Januari 2021 hingga April 2025 dianalisis dengan mempertimbangkan peningkatan permintaan musiman pada bulan Juni dan Desember. Proses peramalan menggunakan SARIMA menghasilkan model terbaik dengan parameter (0, 1, 0)(0, 0, 2, 12) dan nilai *Root Mean Square Error* (RMSE) sebesar 1174,81. Sistem prediksi yang dikembangkan dalam bentuk aplikasi web ini dilengkapi dengan *dashboard* interaktif yang memberikan notifikasi otomatis saat stok melewati batas ROP, beserta rekomendasi jumlah pemesanan minimum. Hasil peramalan menunjukkan bahwa model SARIMA mampu memprediksi kebutuhan stok dengan akurasi yang baik, sehingga sistem ini dapat membantu meningkatkan efektivitas pengelolaan stok dan mendukung kelangsungan operasional bisnis digital printing secara lebih optimal.

2025 Adopsi Teknologi dan Sistem Informasi (ATASI) with CC BY SA license.

**Kata Kunci :**

*Digital Printing*

*Penjualan Historis*

*Reorder Point*

*Safety Stock*

*SARIMA*

**APA style in citing this article:**

A. A. Arumsari, D. Hartanti, and I. Oktaviani, "Implementasi Sistem Prediksi Pemesanan Stok Barang Menggunakan Algoritma Sarima dan Reorder Point (Studi Kasus Devie's Print Copy Center & Digital Printing)," *ATASI: Adopsi Teknologi dan Sistem Informasi*, vol. 4, no. 1, pp. 8–18, 2025.  
<https://doi.org/10.30872/atasi.v4i1.2900>

\*) Corresponding Author

<https://doi.org/10.30872/atasi.v4i1.2900>

2025 Adopsi Teknologi dan Sistem Informasi (ATASI) with CC BY SA license.

## 1. PENDAHULUAN

Manajemen persediaan merupakan salah satu aspek vital dalam menjaga kelancaran operasional bisnis dan ketersediaan bahan baku yang memadai. Tantangan utama yang dihadapi adalah ketidakakuratan dalam menentukan jumlah stok yang diperlukan serta waktu ideal untuk melakukan pemesanan ulang. Kekurangan stok dapat menghambat proses produksi, menimbulkan keterlambatan pemenuhan pesanan dan menyebabkan hilangnya peluang penjualan. Sebaliknya, kelebihan stok berpotensi meningkatkan biaya penyimpanan, menimbulkan penumpukan barang yang tidak terpakai, serta meningkatkan risiko kerusakan atau kadaluarsa produk (Dewi, Suryandari, Widyari, & Sastrawan, 2024).

Kompleksitas pengelolaan stok semakin bertambah akibat ketidakstabilan permintaan, terutama dipengaruhi oleh faktor musiman seperti peningkatan kebutuhan pada masa masuk sekolah di bulan Juni dan Desember. Selain itu, variabilitas *lead time* dalam pengadaan barang juga menjadi tantangan tersendiri, karena kesalahan dalam memperkirakan waktu kedatangan stok dapat mengakibatkan kekosongan persediaan. Permasalahan ini diperparah dengan ketiadaan sistem prediksi berbasis data yang mampu memperkirakan kebutuhan stok secara akurat berdasarkan tren historis dan pola musiman. Akibatnya, keputusan pemesanan barang sering kali bersifat reaktif, tidak efisien, dan berisiko menimbulkan kerugian operasional.

Untuk mengatasi hal tersebut, penelitian ini mengusulkan pembuatan sistem prediksi kebutuhan stok menggunakan algoritma *Seasonal Autoregressive Integrated Moving Average* (SARIMA). Menggunakan Algoritma ini dipilih berdasarkan kehandalannya menangkap pola tren serta musiman pada data deret waktu. Untuk pengujian keakuratan model digunakan metrik *Root Mean Square Error* (RMSE), mengingat tingkat sensitivitasnya terhadap kesalahan prediksi yang besar, yang sangat relevan dalam konteks pengelolaan stok. Berdasarkan studi sebelumnya, model SARIMA menunjukkan performa terbaik dengan nilai RMSE dan MSE yang paling rendah (Ridla, Azise, & Rahman, 2023).

Sistem prediksi ini akan dikombinasikan dengan metode perhitungan *Reorder Point* (ROP) dan *Safety Stock*. ROP ditentukan berdasarkan total kebutuhan selama periode *lead time* ditambah cadangan stok untuk mengantisipasi lonjakan permintaan tak terduga. Lalu sistem dibuat menjadi aplikasi berbasis web dan dilengkapi dengan *dashboard* notifikasi. *Dashboard* ini secara otomatis menampilkan daftar barang yang telah melewati batas *reorder point* beserta saran jumlah minimum pemesanan. Dengan implementasi sistem prediktif ini, diharapkan pengelolaan stok di industri digital *printing* menjadi lebih akurat, risiko kekurangan maupun kelebihan stok dapat diminimalkan, serta efisiensi operasional perusahaan dapat meningkat secara signifikan.

## 2. TINJAUAN PUSTAKA

### A. Seasonal Autoregressive Integrated Moving Average (SARIMA)

Secara ringkas SARIMA adalah sebuah algoritma pemrosesan data berbentuk deret waktu, yang dipergunakan untuk menganalisa dan memprediksi berdasarkan pola musiman. Algoritma ini adalah pengembangan dari model sebelumnya yaitu ARIMA (*Autoregressive Integrated Moving Average*), namun SARIMA sudah di lengkapi penanganan pola musiman yang tidak sederhana (Tokan & Hermawan, 2023). Bentuk model SARIMA secara umum adalah SARIMA( $p,d,q$ )( $P,D,Q$ ) $s$  (Syafar dkk., 2024), dan dapat dituliskan dalam persamaan 1 dibawah ini (Qisthi, Fitri, Immanuel, & Dewi, 2024):

$$(Bs)(1 - Bd)(1 - Bs)DY_t = (B)^t \quad (1)$$

di mana:

- $p$  : orde *Autoregressive* (AR)
- $d$  : orde *differencing*
- $q$  : orde *Moving Average* (MA)
- $P$  : orde *Autoregressive* (AR) musiman
- $D$  : orde *differencing* musiman
- $Q$  : orde *Moving Average* (MA) musiman
- $s$  : periode musiman
- $Y_t$  : nilai pada waktu ke- $t$
- $B$  : operator lag
- $(B^s)$  : polinom AR musiman
- $(B)$  : polinom MA
- $\varepsilon_t$  : kesalahan acak (*white noise*)

### B. Safety Stock

*Safety stock* merupakan suatu analisis yang bertujuan untuk menentukan jumlah cadangan persediaan yang optimal. Ketidakpastian dalam permintaan pelanggan dapat berdampak signifikan terhadap ketersediaan stok, sehingga diperlukan *safety stock* sebagai langkah antisipatif terhadap potensi kekurangan barang. Keberadaan *safety stock* membantu perusahaan penyedia barang mencegah terjadinya kekosongan stok, terutama saat terjadi keterlambatan pengiriman atau saat waktu tunggu berlangsung lebih lama dari perkiraan (Sagala, Rumapea, & Silalahi, 2023). Model *safety stock* yang akan digunakan yaitu sesuai dengan persamaan 2 dibawah ini.

$$SS = Z \times \sigma_{LT} \quad (2)$$

di mana:

SS : Stok pengaman (*safety stock*)  
Z : Nilai *Z-score* sesuai *service level* yang diinginkan  
(contoh:  $Z = 1.65$  untuk 95% *service level*,  $Z = 2.33$  untuk 99%)  
 $\sigma_{LT}$  : Standar deviasi permintaan.

Kesimpulannya, *safety stock* merupakan stok yang disediakan untuk mencegah terjadinya kekurangan persediaan. Metode ini dilakukan dengan cara menghitung selisih antara penjualan tertinggi dan penjualan rata-rata dalam periode tertentu (seperti per bulan). Selisih tersebut kemudian dikalikan dengan waktu tunggu, sehingga diperoleh jumlah stok pengaman yang dibutuhkan (Setiawan, 2024).

### C. Reorder Point

*Reorder Point* (ROP) merupakan tingkat atau titik persediaan di mana perusahaan harus segera melakukan pemesanan ulang agar tidak terjadi kekosongan stok. Namun, dalam praktiknya, perhitungan ROP dapat disesuaikan dengan kebijakan internal perusahaan. Salah satu kebijakan yang sering memengaruhi perhitungan ROP adalah penambahan *safety stock* sebagai cadangan untuk mengantisipasi ketidakpastian permintaan atau keterlambatan pengiriman. Model ROP yang akan digunakan yaitu sesuai dengan persamaan 3 dibawah ini (Nurfauzia & Rizqiya, 2023):

$$ROP = (d \times Lt) + SS \quad (3)$$

Dimana:

ROP : *Reorder point*  
D : Kebutuhan harian  
Lt : *Lead time*  
SS : *Safety stock*

## 3. METODE PENELITIAN

Penelitian ini memakai pendekatan berbasis data kuantitatif yaitu metode penelitian yang berlandaskan pada paradigma *post-positivisme* dalam upaya pengembangan ilmu pengetahuan. Ciri utama dari pendekatan ini meliputi pengumpulan serta analisis data dalam bentuk angka (kuantitatif), penerapan strategi seperti survei dan eksperimen, pelaksanaan proses pengukuran dan observasi, serta pengujian teori melalui analisis statistik (Muhajirin, Risnita, & Asrulla, 2024). Oleh karena itu, rancangan metode penelitian ini mencakup dua komponen utama, yaitu pengumpulan data dan pengembangan sistem.

### A. Pengumpulan Data

Metode pengumpulan data dilakukan secara dokumentatif, yaitu dengan mengumpulkan data historis penjualan barang digital *printing* selama periode tahun 2021 hingga 2025 yang diperoleh dari sistem *database* internal perusahaan. Selain itu, data stok harian dan waktu tunggu (*lead time*) dari supplier dicatat dari arsip internal dan wawancara singkat dengan bagian pembelian. Data tambahan berupa kalender akademik juga dikumpulkan dari situs resmi sekolah terdekat sebagai acuan dalam menganalisis pola musiman yang berpotensi memengaruhi permintaan.

### B. Metode Pengembangan Sistem

Dalam penelitian ini, sistem dikembangkan menggunakan metode Waterfall. Waterfall merupakan model dalam *Software Development Life Cycle* yang umum digunakan. Metode ini menerapkan pendekatan bertahap, di mana proses pengembangan dibagi menjadi sejumlah fase yang terstruktur dan saling berurutan. Setiap tahap menghasilkan output yang menjadi acuan bagi tahap selanjutnya, dan setelah suatu fase diselesaikan, proses tidak kembali ke tahap sebelumnya (Duma & Pusvita, 2023). Metode ini dilakukan secara bertahap dengan lima langkah utama.

1. Analisis kebutuhan, yaitu proses mengidentifikasi permasalahan dalam sistem pengelolaan stok yang masih bersifat manual, serta merumuskan kebutuhan fungsional sistem, termasuk kebutuhan akan prediksi permintaan dan peringatan pemesanan ulang. Dalam tahap ini juga dilakukan analisis pola permintaan berdasarkan musim masuk sekolah.
2. Perancangan Sistem, yaitu tahap di mana struktur *database*, antarmuka pengguna, dan alur logika sistem dirancang. Untuk menggambarkan fungsi dan interaksi antar komponen, perancangan dilakukan menggunakan model UML, seperti *use case diagram*, *class diagram*, *sequence diagram*, dan *activity diagram*. Pada tahap ini juga dirancang metode peramalan dengan menggunakan algoritma SARIMA (*Seasonal Autoregressive Integrated Moving Average*).
  - a) *Preprocessing Data*, sebelum dilakukan peramalan, data penjualan melalui beberapa tahapan *preprocessing*. Tahapan ini meliputi pembersihan data (penghapusan data duplikat dan nilai hilang) dan uji stasioneritas menggunakan *Augmented Dickey-Fuller* (ADF). Jika data tidak stasioner, dilakukan proses *differencing* baik secara musiman maupun non-musiman untuk memastikan data memenuhi asumsi stasioneritas model SARIMA.

- b) Pemilihan Parameter SARIMA, dilakukan melalui analisis *Autocorrelation Function* (ACF) dan *Partial Autocorrelation Function* (PACF), serta *grid search* terhadap kombinasi parameter musiman dan non-musiman  $(p, d, q) \times (P, D, Q, s)$  untuk mendapatkan model dengan RMSE terendah.
- c) Platform Pengembangan Aplikasi, sistem dikembangkan menggunakan *python* sebagai bahasa pemrograman untuk perhitungan peramalan, dengan *framework flask*. *Library* tambahan seperti *pmdarima*, *statsmodels*, dan *scikit-learn* digunakan untuk proses peramalan dan evaluasi model. Sedangkan untuk membangun aplikasi web menggunakan PHP dengan *framework laravel*.
- d) Evaluasi model, mencakup uji normalitas residual dan plot ACF/PACF residual untuk memastikan tidak ada autokorelasi tersisa. Sementara itu, pengujian akurasi dilakukan terhadap hasil prediksi model SARIMA. Akurasi prediksi diukur menggunakan *Root Mean Square Error* (RMSE) yang merupakan rata-rata dari jumlah kuadrat selisih antara nilai yang diprediksi dan nilai aktual. RMSE digunakan untuk mengukur seberapa besar kesalahan dalam suatu model prediksi. Nilai RMSE yang lebih rendah menunjukkan bahwa model menghasilkan prediksi yang lebih akurat. Dalam pengujian ini, data dibagi menggunakan teknik *train-test split*, yaitu sekitar 80% data digunakan untuk pelatihan dan 20% untuk pengujian. Pembagian ini bertujuan mengukur performa model terhadap data yang tidak pernah dilihat selama pelatihan. RMSE dihitung menggunakan rumus pada persamaan 4 di bawah ini (Nurani, Setiawan, & Susanto, 2023):

$$RMSE = \frac{\sum_{t=1}^n (Y'_t - Y_t)^2}{n} \quad (4)$$

di mana:

$Y'_t$  : nilai data aktual ke- $t$  ( $t = 1, \dots, n$ ),

$Y_t$  : nilai aktual ke- $t$  ( $t = 1, \dots, n$ ),

$n$  : banyaknya data uji.

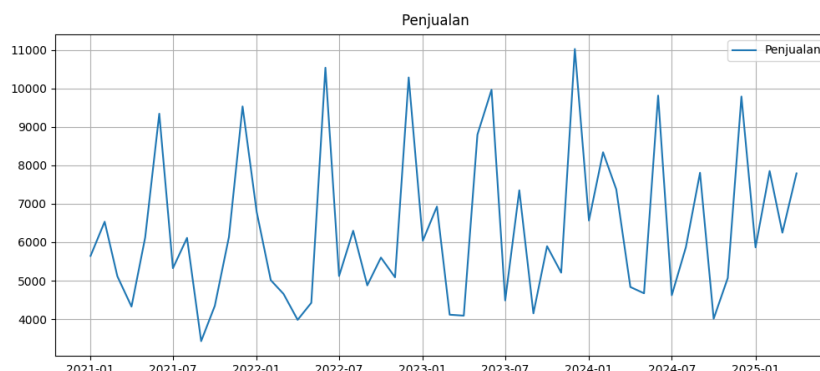
Data prediksi dari model SARIMA kemudian digunakan dalam perhitungan *Reorder Point* (ROP), yaitu metode penentuan titik aman untuk melakukan pemesanan ulang barang sebelum stok habis, di mana *demand* harian diperoleh dari hasil prediksi SARIMA dan *safety stock* dihitung berdasarkan standar deviasi permintaannya.

3. Implementasi Sistem, yaitu proses membangun sistem berbasis web. Sistem ini mengintegrasikan fitur manajemen data, proses peramalan SARIMA, perhitungan ROP, serta pemberian notifikasi jika stok telah mencapai titik pemesanan ulang.
4. Pengujian Sistem, *Black Box Testing* digunakan untuk menguji apakah semua fitur sistem berfungsi dengan sesuai, mulai dari input data, proses peramalan, hingga perhitungan titik pemesanan ulang (ROP).
5. Pemeliharaan, yaitu tahap terakhir yang dilakukan setelah sistem digunakan oleh pengguna. Pada tahap ini, sistem dipantau secara berkala untuk memastikan tetap berjalan sesuai harapan, serta dilakukan perbaikan jika ditemukan bug atau kesalahan sistem. Selain itu, pemeliharaan juga mencakup pembaruan fitur berdasarkan kebutuhan baru dari pengguna, serta peningkatan performa sistem agar tetap relevan dengan kondisi operasional yang berkembang.

#### 4. HASIL DAN PEMBAHASAN

##### A. Hasil Analisa Kebutuhan

Dalam Penelitian ini, digunakan data yang bersumber dari *database* penjualan di devie's *print*. Data penjualan diambil dari bulan januari 2021 sampai dengan april 2025 dengan jumlah data sebanyak 52 baris terdiri dari jumlah penjualan jasa *fotocopy* per bulan. Data tersebut ditampilkan ke dalam grafik seperti yang terlihat pada gambar 1 di bawah ini.



Gambar 1. Data Historis Penjualan Tahun 2021-2025

## B. Perancangan Sistem

Pada tahap ini akan dijelaskan tahapan dari *pre-processing* data sampai dengan peramalan menggunakan SARIMA dan penentuan *reorder point*. Proses analisis SARIMA mencakup tahapan uji stationaritas, analisis ACF dan PACF, pembangunan model SARIMA, dan evaluasi model SARIMA (Kuswanto, Eko, Utomo, Khaira, & Waladi, 2025). Kemudian setelah peramalan SARIMA sudah terpenuhi, maka di lanjutkan dengan penentuan pemesanan ulang barang.

### 1. Uji Stasionaritas

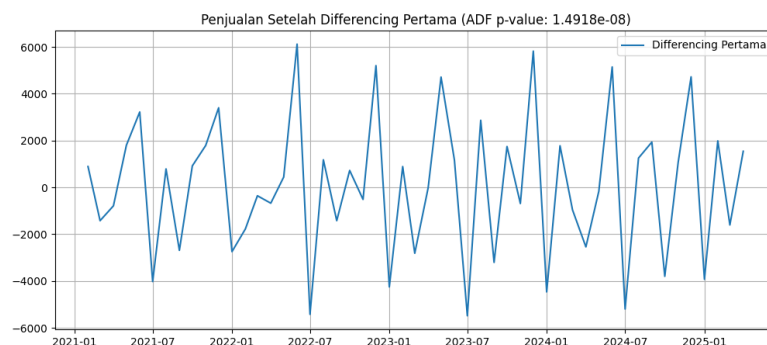
*Augmented Dickey Fuller* (ADF Test) digunakan untuk menguji kestasioneritasan data (Ramadhan & Fauzan, 2023). Pengujian dilakukan dengan menggunakan *python*, dengan data uji seperti yang sudah disebutkan diatas, sehingga menghasilkan grafik dan hasil test sesuai dengan tabel 1 di bawah ini.

Tabel 1. Hasil ADF Test

| Augmented Dickey-Fuller | Result |
|-------------------------|--------|
| Test Statistic:         | -1.749 |
| p-value:                | 0.406  |
| Critical Value (1%):    | -3.581 |
| Critical Value (5%):    | -2.927 |
| Critical Value (10%):   | -2.601 |

Pada uji ADF (Augmented Dickey-Fuller), suatu data dianggap stasioner jika menunjukkan pola yang stabil seiring waktu. Apabila data deret waktu tidak stasioner, maka hasil prediksi cenderung menyimpang dari nilai sebenarnya dan dapat menyebabkan kesalahan. Dalam uji ini, hipotesis nol ( $H_0$ ) menyatakan bahwa data tidak stasioner, sedangkan hipotesis alternatif ( $H_1$ ) menyatakan bahwa data bersifat stasioner.  $H_0$  ditolak jika nilai p-value kurang dari tingkat signifikansi alpha (0,05) (Zamil, Anzani, & Arifin, 2023). Seperti yang terlihat pada tabel 1, setelah dilakukan perhitungan ADF Test, p-value bernilai 0.4058, sehingga dinyatakan data belum stasioner, karena nilai p-value masih lebih besar dari 0.05. Kemudian Hasil dari uji ADF dibandingkan dengan nilai kritis McKinnon pada level signifikansi 1%, 5%, dan 10%. Apabila nilai t-statistik lebih rendah dari nilai kritis McKinnon, maka hipotesis nol ( $H_0$ ) ditolak, yang menunjukkan bahwa data sudah stasioner (Kuswanto dkk., 2025). Seperti pada tabel 1 di atas nilai *statistic* -1.749 masih lebih kecil dari *critical value* 10%, jadi data belum stasioner.

Karena data masih belum stasioner, maka kita lakukan proses *differencing* menggunakan *python*. Grafik data penjualan setelah dilakukan *differencing* seperti yang terlihat pada gambar 2 di bawah ini.



Gambar 2. Data Penjualan Setelah Differencing

Seperti pada tabel 2, p-value sangatlah kecil bernilai 0.0000000149 lebih kecil dari 0.05, sehingga data sudah bisa disebut stasioner. Kemudian nilai *statistic* -6.455 lebih kecil dari semua *critical value*, sehingga data juga sudah bisa disebut stasioner. Setelah data sudah stasioner, maka bisa dilanjutkan ke proses selanjutnya yaitu analisis ACF dan PACF.

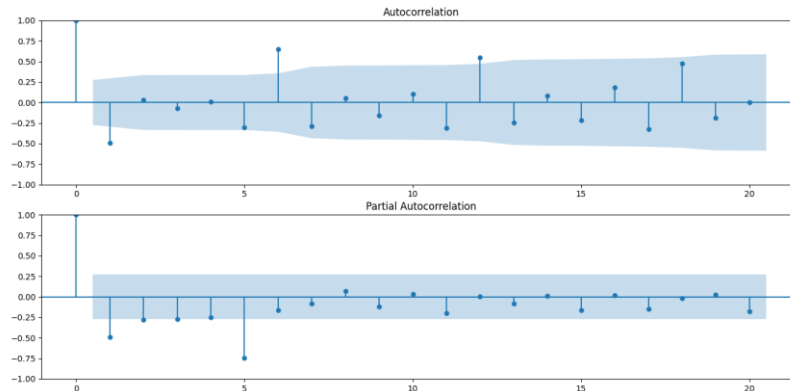
Tabel 2. Hasil ADF Test Setelah Differencing

| Augmented Dickey-Fuller | Result       |
|-------------------------|--------------|
| Test Statistic:         | -6.455       |
| p-value:                | 0.0000000149 |
| Critical Value (1%):    | -3.585       |
| Critical Value (5%):    | -2.928       |
| Critical Value (10%):   | -2.602       |

### 2. Analisis ACF dan PACF

Analisis ACF dan PACF untuk penentuan parameter SARIMA bertujuan untuk membantu proses identifikasi parameter optimal dalam membangun model SARIMA, *plotting* ACF (*Autocorrelation Function*) dan PACF (*Partial Autocorrelation Function*) digunakan untuk menganalisis hubungan antar data pada lag yang berbeda. Plot

ACF memberikan gambaran tentang korelasi antar nilai data pada berbagai lag, sedangkan plot PACF membantu menentukan jumlah lag yang relevan untuk komponen autoregresif (AR) dalam model. Dengan menggunakan visualisasi ini, parameter utama seperti order (p, d, q) dan seasonal\_order (P, D, Q, s) dapat ditentukan secara lebih efektif (Ikhsani, Martanto, Dikananda, & Mulyawan, 2025).

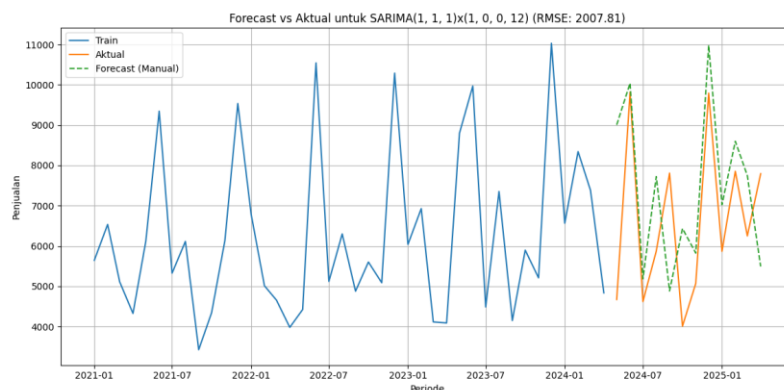


Gambar 3. Plot ACF & PACF

Seperti pada gambar 3 di atas, untuk menentukan parameter p dilihat pada plot PACF, dimana terdapat lonjakan signifikan pada lag 1 sampai melewati blok biru, blok biru di sini adalah menandakan tingkat kepercayaan. Sehingga bisa didapatkan kemungkinan untuk parameter p adalah 1. Lalu parameter d diambil dari uji ADF Test, karena data stasioner pada *differencing* pertama, maka nilai d adalah 1. Kemudian parameter q dilihat dari plot ACF, dimana terdapat lonjakan signifikan juga pada lag 1, sehingga memungkinkan jika nilai q adalah 1. Kemudian untuk parameter seasonal, nilai P di dapat dari plot PACF, karena pada lag 1 terdapat lonjakan maka nilai P memungkinkan bernilai 1, lalu untuk D karena tidak ada proses seasonal *differencing* data sudah stasioner maka nilai D adalah 0, kemudian nilai Q dilihat dari plot ACF, karena data musiman pada lag 6, 12, dan 18 tidak terlalu signifikan maka Q memungkinkan untuk bernilai 0. Kemudian karena ada data musiman tahunan, maka nilai s adalah 12. Jadi sudah di dapat nilai parameter sementara yang bisa digunakan untuk perhitungan SARIMA, yaitu (1,1,1)(1,0,0,12).

### 3. Pembangunan Model SARIMA

Pada tahap sebelumnya sudah di dapat parameter sementara yang bisa di gunakan untuk prediksi dengan SARIMA, disini proses perhitungan menggunakan *python*, berikut hasil grafik yang menunjukkan data actual dan hasil *forecasting* dari prediksi SARIMA.



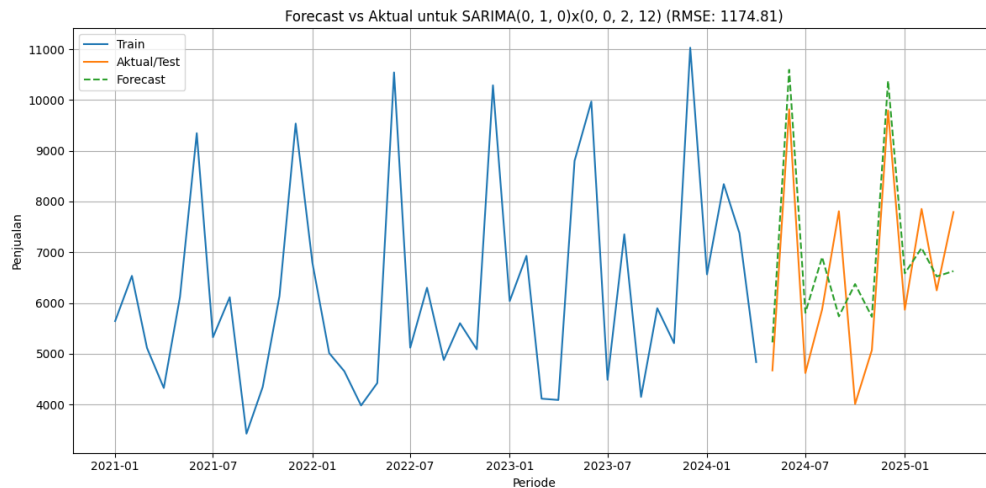
Gambar 4. Grafik Forecast vs Aktual

Terlihat pada gambar 4 diatas, untuk menguji hasil *error* pada pemodelan SARIMA digunakan RMSE (*Root Mean Squared Error*), dan RMSE menghasilkan nilai 2007.81, di mana nilai tersebut masih bisa di bilang cukup tinggi. Untuk menentukan parameter terbaik, pada penelitian ini di gunakan grid search, dari proses *search* tersebut di dapatkan 5 parameter terbaik seperti pada tabel 3 dibawah ini.

Tabel 3. Top 5 Parameter

| Order             | Seasonal                | RSME    |
|-------------------|-------------------------|---------|
| (p,d,q)=(0, 1, 0) | (P,D,Q,s)=(0, 0, 2, 12) | 1174.81 |
| (p,d,q)=(1, 1, 0) | (P,D,Q,s)=(0, 0, 2, 12) | 1306.44 |
| (p,d,q)=(2, 1, 1) | (P,D,Q,s)=(0, 0, 2, 12) | 1456.38 |
| (p,d,q)=(0, 1, 1) | (P,D,Q,s)=(0, 0, 2, 12) | 1515.86 |
| (p,d,q)=(1, 1, 1) | (P,D,Q,s)=(0, 0, 2, 12) | 1516.82 |

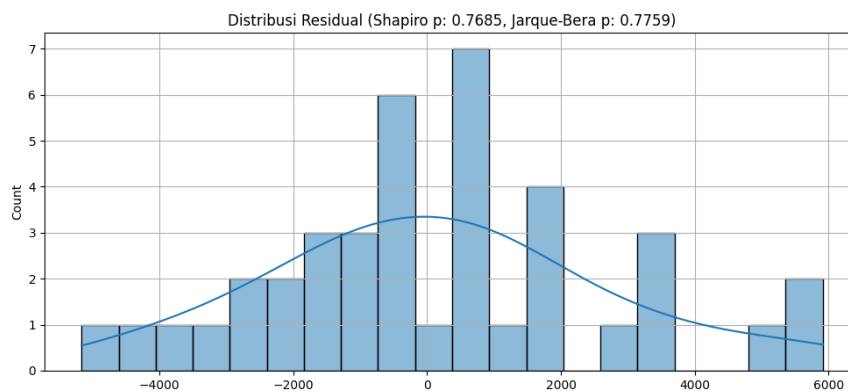
Berdasarkan tabel 3 diatas, dapat disimpulkan bahwa parameter yang paling bagus adalah (0,1,0)(0,0,2,12) dengan nilai RSME 1174.81, lebih kecil dari parameter sebelumnya. Setelah ditentukan parameter yang paling kecil nilai RSMEnya, maka kita bisa proses ulang model SARIMA-nya. Hasil perbandingan grafik antara data actual dan forecast bisa dilihat pada gambar 5 di bawah ini.



Gambar 5. Grafik Forecast vs Aktual Parameter Terbaik

#### 4. Evaluasi model SARIMA

Setelah mendapatkan model SARIMA, perlu dilakukan uji normalitas residual dan *white noise*, agar model prediksi siap di gunakan untuk memprediksi penjualan barang. Untuk menguji normalitas residual digunakan Shapiro-Wilk dan Jarque-Bera. Setelah dilakukan pengujian dengan menggunakan *python* di dapatkan hasil p-value Shapiro-Wilk bernilai 0.7685 lebih tinggi dari 0.05 yang berarti residual berdistribusi normal, kemudian untuk hasil p-value Jarque-Bera bernilai 0.7759 lebih besar juga dari 0.05, yang berarti residual berdistribusi normal, grafik yang menunjukkan distribusi residual dapat dilihat pada gambar 6 dibawah. Untuk menguji *white noise* di gunakan Ljung-Box. Setelah dilakukan pengujian dengan menggunakan *python*, di dapatkan hasil p-value 0.1335, karena hasilnya lebih besar dari 0.05 maka residual dianggap *white noise*. Sehingga model SARIMA(0,1,0)(0,0,2,12) siap di gunakan untuk peramalan penjualan barang.



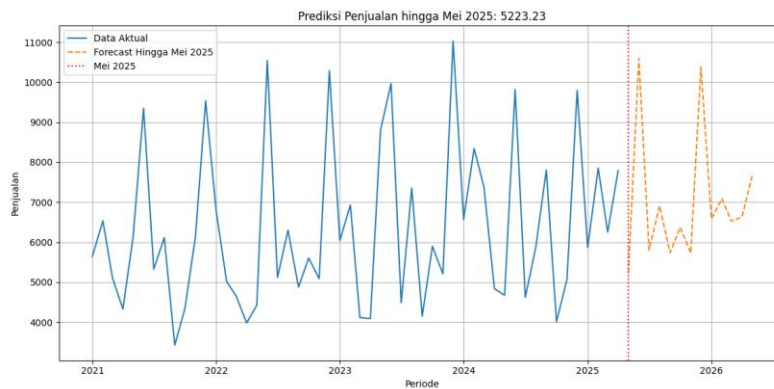
Gambar 6. Grafik Distribusi Residual

#### 5. Penentuan Pemesanan Ulang Barang

Setelah melakukan peramalan dengan SARIMA, didapatkan total penjualan pada bulan Mei sebesar 5223, bisa dilihat pada gambar 7. Untuk mendapatkan rata-rata penjualan perhari, maka 5223 dibagi 30, hasilnya 174. Data penjualan yang di gunakan di awal adalah penjualan untuk jasa fotokopi, fotokopi memerlukan barang kertas hvs, *lead time* nya 3 hari, dan stok saat ini pada tanggal 3 Mei adalah 1000, untuk *safety stock* senilai 605. Sehingga *reorder point* bisa dihitung menggunakan rumus pada persamaan 3, seperti di bawah ini.

$$\begin{aligned} \text{ROP} &= (d \times \text{Lt}) + \text{SS} \\ &= (174 \times 3) + 605 \\ &= 522 + 605 \\ &= 1127 \end{aligned}$$

Karena stok sekarang lebih kecil dari hasil ROP, maka akan memunculkan notifikasi pada *dashboard* sistem, untuk segera memesan barang, dan akan muncul estimasi minimum kuantiti pemesanan yaitu 1127 di kurangi 1000, jadi hasilnya 127 yang harus di pesan.



Gambar 7. Prediksi Penjualan Bulan Mei 2025

### C. Implementasi Sistem

Berikut merupakan hasil pembuatan sistem berbasis web yang telah dirancang sebelumnya, berikut beberapa antarmuka dari sistem tersebut.

Gambar 8. Form Login

Pada gambar 8 diatas merupakan antarmuka login untuk *user* yang akan mengakses aplikasi, dengan cara memasukkan email dan *password* yang sudah didaftarkan sebelumnya.

Gambar 9. Form Penerimaan Barang

Pada gambar 9 diatas merupakan antarmuka untuk *user* admin memasukkan data penerimaan barang, dengan cara memilih nama produk, satuan, nama *supplier*, memasukkan harga beli, qty dan harga jual. Untuk harga per pcs akan otomatis mengkalkulasi dari satuan yang di pilih. Setelah semua diisi, tekan tombol tambah, maka data akan masuk ke dalam tabel, setelah semua sudah di masukkan, tekan tombol simpan, untuk menyimpan semua data dan otomatis menambah jumlah stok barang.

Gambar 10. Form Point of Sale

Pada gambar 10 diatas, menunjukkan antarmuka untuk *user* kasir memasukkan data barang yang terjual, dengan cara memilih nama produk, satuan, dan memasukkan qty. Untuk keterangan dan kepada bersifat *optional*, dan untuk harga akan otomatis mengambil harga jual sesuai nama produk dan satuan yang dipilih. Kemudian tekan tombol tambah, maka data akan masuk ke dalam tabel, dan otomatis terhitung sub totalnya. Masukkan diskon jika sedang ada promo, maka akan otomatis terhitung total *amount* nya. Jika di bayar tunai, maka masukkan nominal bayarnya, maka akan otomatis terhitung kembaliannya jika ada. Jika pembayaran dengan *down payment* (DP) maka masukkan total DP, akan terhitung otomatis kekurangan bayarnya. Kemudian tekan tombol simpan, untuk menyimpan semua data, dan akan mengurangi jumlah stok barang tersebut, setelahnya akan tampil struk belanjanya yang siap untuk di cetak.

| Nama Barang   | Stok Saat Ini | Reorder Point | Rekomendasi Minimal Order |
|---------------|---------------|---------------|---------------------------|
| Kertas HVS    | 1000          | 1127          | 127                       |
| Tinta Hitam   | 10            | 15            | 5                         |
| Stiker Glossy | 1             | 5             | 4                         |

Gambar 11. Dashboard

Pada gambar 11 diatas, merupakan antarmuka *dashboard*, yang digunakan untuk melihat notifikasi stok barang yang kurang dari *reorder point*, sehingga bisa digunakan oleh admin sebagai acuan pemesanan barang ke *supplier*, agar stok dapat terpenuhi dengan baik.

#### D. Pengujian Sistem

Pada tahap ini, dilakukan proses pengujian terhadap sistem yang telah selesai dibuat menggunakan metode Black Box Testing. Metode ini merupakan teknik pengujian perangkat lunak yang menilai fungsi sistem tanpa memperhatikan struktur internal atau kode programnya. Tujuan dari pengujian ini adalah untuk memastikan bahwa sistem berjalan sesuai dengan kebutuhan dan spesifikasi yang telah ditentukan.

Tabel 4. Test Case Form Login

| Nama Test Case | Input                                 | Langkah Uji                                        | Output Yang Diharapkan          | Hasil Uji |
|----------------|---------------------------------------|----------------------------------------------------|---------------------------------|-----------|
| Login Berhasil | Username: admin<br>Password: admin123 | Masukkan <i>username</i> dan <i>password</i> valid | Masuk ke <i>dashboard</i> utama | Sesuai    |
| Login Gagal    | Username: admin<br>Password: salah    | Masukkan <i>password</i> salah                     | Muncul pesan "Login gagal"      | Sesuai    |

Tabel 5. Test Case Form Penerimaan Barang

| Nama Test Case                         | Input                                                                                              | Langkah Uji                                                 | Output Yang Diharapkan                       | Hasil Uji |
|----------------------------------------|----------------------------------------------------------------------------------------------------|-------------------------------------------------------------|----------------------------------------------|-----------|
| Simpan Data Penerimaan Barang Berhasil | Nama barang, jumlah, nama supplier, satuan, harga beli, harga jual, harga beli pcs, harga jual pcs | Isi form lalu klik "Simpan"                                 | Data berhasil disimpan dan stok bertambah    | Sesuai    |
| Simpan Gagal Karena Data Kosong        | Kosong semua <i>field</i>                                                                          | Kosongkan form lalu klik "Simpan"                           | Muncul pesan error "Data tidak boleh kosong" | Sesuai    |
| Edit Data Penerimaan Barang Berhasil   | Data penerimaan lama & data baru                                                                   | Pilih data penerimaan → klik edit → ubah data → klik simpan | Data penerimaan terupdate sesuai input baru  | Sesuai    |
| Hapus Data Penerimaan Barang Berhasil  | Pilih salah satu data penerimaan                                                                   | Klik tombol hapus lalu konfirmasi                           | Data penerimaan terhapus dari daftar         | Sesuai    |

Tabel 6. Test Case Form Penjualan Barang

| Nama Test Case                      | Input                                         | Langkah Uji                      | Output Yang Diharapkan                           | Hasil Uji |
|-------------------------------------|-----------------------------------------------|----------------------------------|--------------------------------------------------|-----------|
| Input Transaksi Penjualan Berhasil  | Nama barang, jumlah, <i>remark</i> (optional) | Isi form dan klik "Simpan"       | Data penjualan tercatat, stok berkurang otomatis | Sesuai    |
| Input Gagal Karena Stok Tidak Cukup | Jumlah penjualan > stok                       | Simulasi penjualan melebihi stok | Muncul pesan <i>error</i> "Stok tidak mencukupi" | Sesuai    |

Tabel 7. Test Case Form Dashboard / Notifikasi

| Nama Test Case                    | Input             | Langkah Uji                              | Output Yang Diharapkan                           | Hasil Uji |
|-----------------------------------|-------------------|------------------------------------------|--------------------------------------------------|-----------|
| Notifikasi Saat Stok < ROP        | Stok aktual < ROP | Simulasikan stok rendah, akses dashboard | Muncul List Nama Barang, Beserta Stok Saat Ini   | Sesuai    |
| Rekomendasi Pemesanan Ditampilkan | Stok < ROP        | Akses dashboard                          | Rekomendasi jumlah pemesanan minimum ditampilkan | Sesuai    |

Sesuai hasil pengujian sistem dengan metode black box testing yang terlihat pada tabel 4 sampai dengan tabel 7 diatas, semua fitur utama seperti *form login*, *form* penerimaan barang, *form* penjualan, dan *dashboard* notifikasi telah diuji dengan skenario sesuai dan tidak sesuai. Hasil uji menunjukkan bahwa sistem merespons sesuai dengan harapan pada semua kondisi input yang diuji. Dengan demikian, sistem dapat dinyatakan berjalan secara fungsional sesuai dengan rancangan dan layak untuk dipergunakan.

## 5. KESIMPULAN

Dari hasil penelitian dan proses pengujian yang telah dilakukan, dapat ditarik kesimpulan bahwa sistem prediksi kebutuhan stok yang dikembangkan menggunakan algoritma SARIMA dengan parameter (0, 1, 0)(0, 0, 2, 12) mampu memberikan hasil peramalan yang akurat, dengan nilai RMSE sebesar 1174,81. Integrasi metode *Reorder Point* (ROP) dan *Safety Stock* ke dalam sistem membantu memberikan informasi yang lebih praktis dalam pengambilan keputusan pemesanan barang. Implementasi sistem dalam bentuk aplikasi web dengan fitur notifikasi otomatis dan *dashboard* interaktif terbukti meningkatkan efisiensi pengelolaan stok pada Devie's *Print Copy Center & Digital Printing*. Sistem ini mampu meminimalisir risiko kehabisan atau kelebihan stok, serta mendukung kelangsungan operasional bisnis secara lebih optimal dan terukur, terutama dalam menghadapi fluktuasi permintaan musiman. Untuk penelitian selanjutnya disarankan untuk mengeksplorasi dan membandingkan performa algoritma peramalan lainnya seperti Prophet, LSTM, atau *Exponential Smoothing* untuk meningkatkan akurasi dan ketahanan sistem terhadap variasi pola data yang lebih kompleks.

## 6. DAFTAR PUSTAKA

- Dewi, N. M. S. S., Suryandari, N. N. A., Widyari, N. Y. A., & Sastrawan, I. K. (2024). Optimalisasi Pencatatan Stok Persediaan Dengan Kartu Stok Dan Pemeriksaan Omzet Harian Di Rockpizz Music. *Prosiding Seminar Nasional Pengabdian Masyarakat*, 3(1), 457–463.
- Duma, A., & Pusvita, E. A. (2023). Pengembangan Sistem Informasi Data Siswa Berbasis Web Pada Smpn 09 Nabire Dengan Metode Waterfall. *Journal of Information System Management (JOISM)*, 5(1), 70–76.

- Ikhsani, I., Martanto, Dikananda, A. R., & Mulyawan. (2025). Prediksi Harga Emas Menggunakan Algoritma Seasonal Auto Regressive Integrated Moving Average. *Jurnal informasi dan Komputer*, 13(1), 1–10.
- Kuswanto, H., Eko, P., Utomo, P., Khaira, U., & Waladi, A. (2025). Prediksi Nilai Ekspor Migas Indonesia Menggunakan Metode SARIMA dan LSTM. *Jurnal Sains Teknologi dan Sistem Informasi*, 5(1), 69–79. <https://doi.org/10.54259/satesi.v5i1.4103>
- Muhajirin, Risnita, & Asrulla. (2024). Pendekatan Penelitian Kuantitatif Dan Kualitatif Serta Tahapan Penelitian. *Journal Genta Mulia*, 15(1), 82–92.
- Nurani, A. T., Setiawan, A., & Susanto, B. (2023). Perbandingan Kinerja Regresi Decision Tree dan Regresi Linear Berganda untuk Prediksi BMI pada Dataset Asthma. *Jurnal Sains dan Edukasi Sains*, 6(1), 34–43. <https://doi.org/10.24246/juses.v6i1p34-43>
- Nurfauzia, F., & Rizqiya, S. (2023). Reorder Point Analysis On Fasajaya Wholesale Store Inventory Using POM QM Application. *JOBM-Journal Of Bussines Management Basic*, 5(2). <https://doi.org/https://doi.org/10.33481/jobm.v5i2.1113>
- Qisthi, N., Fitri, S. L., Immanuel, A., & Dewi, D. D. (2024). Prediksi Harga Emas Untuk Investasi Masa Depan Menggunakan Model Seasonal Autoregressive Integrated Moving Average (SARIMA). *Journal of Innovation Research and Knowledge*, 4(7), 4183–4194.
- Ramadhan, A. D., & Fauzan, A. (2023). Prediksi Nilai Ekspor Non-Migas Di Jawa Barat Menggunakan Metode Seasonal Auto Regresif Integrated Moving Average (SARIMA). *Emerging Statistics and Data Science Journal*, 1(1), 10–19.
- Ridla, M. A., Azise, N., & Rahman, M. (2023). Perbandingan Model Time Series Forecasting Dalam Memprediksi Jumlah Kedatangan Wisatawan Dan Penumpang Airport. *SIMKOM*, 8(1), 1–14. <https://doi.org/10.51717/simkom.v8i1.103>
- Sagala, P. R., Rumapea, Y., & Silalahi, A. P. (2023). Pengendalian Persediaan Gas dengan Metode Economic Order Quantity untuk Menentukan Safety Stock dan Reorder Point. *METHOSISFO: Jurnal Ilmiah Sistem Informasi*, 3(2), 33–39. Diambil dari <https://ejurnal.methodist.ac.id/index.php/methosisfo/33>
- Setiawan, F. (2024). Perancangan Aplikasi Pengendalian Persediaan Barang Dengan Metode Safety Stock Dan Reorder Point (Studi Kasus : PT. Airlangga Jaya Mandiri). *Jurnal Ilmu Komputer dan Pendidikan*, 2(2), 401–408. Diambil dari <https://journal.mediapublikasi.id/index.php/logic>
- Syafar, M., Wulandari, J., Amanda, M. F., Shavira, N., Aulia, H. M., Syahfitri, D., & Dalimunthe, D. Y. (2024). Perbandingan Metode Sarima Dan Exponential Smoothing Dalam Memprediksi Curah Hujan Di Kabupaten Bangka Tengah. *Jurnal Fraction*, 4(2), 41–48.
- Tokan, L. F., & Hermawan, A. (2023). Implementasi Model SARIMA Untuk Memprediksi Produksi Minyak Kelapa Sawit. *JURNAL FASILKOM*, 13(3), 456–463. Diambil dari <https://ditjenbun.pertanian.go.id/template/uploads/202>
- Zamil, A. S., Anzani, L., & Arifin, W. A. (2023). Prediksi Tinggi Gelombang Laut Jakarta Utara Menggunakan Machine Learning: Perbandingan Algoritma Arima & Sarima. *JURNAL ILMIAH TEKNOLOGI INFORMASI DAN KOMUNIKASI (JTIK)*, 14(2), 286–294. Diambil dari <http://ejurnal.provisi.ac.id/index.php/JTIKP>