

**STUDI KOMPARASI PERFORMA ALGORITMA *EXTREME GRADIENT BOOSTING* (XGBOOST) DAN *ARTIFICIAL NEURAL NETWORK* (ANN) PADA SISTEM PREDIKSI PRODUKSI PADI BERBASIS WEB**

**SKRIPSI**

**Diajukan untuk memenuhi sebagian persyaratan mendapatkan gelar Strata Satu Program Studi Informatika**



**Disusun oleh :  
RHEIKA MOSA LARASATI  
22EO10022**

**PROGRAM STUDI INFORMATIKA  
FAKULTAS MATEMATIKA DAN ILMU KOMPUTER  
UNIVERSITAS NAHDLATUL ULAMA AL GHAZALI  
CILACAP  
2026**

**STUDI KOMPARASI PERFORMA ALGORITMA *EXTREME GRADIENT BOOSTING* (XGBOOST) DAN *ARTIFICIAL NEURAL NETWORK* (ANN) PADA SISTEM PREDIKSI PRODUKSI PADI BERBASIS WEB**

**SKRIPSI**

**Diajukan untuk memenuhi sebagian persyaratan mendapatkan gelar Strata Satu Program Studi Informatika**



**RHEIKA MOSA LARASATI  
22EO10022**

**PROGRAM STUDI INFORMATIKA  
FAKULTAS MATEMATIKA DAN ILMU KOMPUTER  
UNIVERSITAS NAHDLATUL ULAMA AL GHAZALI  
CILACAP  
2026**

## HALAMAN PERNYATAAN KEASLIAN SKRIPSI

Dengan ini saya menyatakan bahwa penulisan skripsi dengan judul “Studi Komparasi Performa Algoritma *Extreme Gradient Boosting* (XGBoost) dan *Artificial Neural Network* (ANN) pada Sistem Prediksi Produksi Padi Berbasis Web” adalah hasil karya saya dengan arahan dari pembimbing dan belum diajukan kepada pihak manapun. Sumber informasi yang dikutip dalam skripsi ini telah dicantumkan dalam Daftar Pustaka.

Demikian pernyataan ini dibuat dengan sebenarnya. Apabila di kemudian hari terdapat ketidaksesuaian dalam pernyataan ini, maka saya bersedia menerima sanksi sesuai dengan peraturan yang berlaku.

Cilacap, 24 Juni 2026



Rheika Mosa Larasati  
22EO10022

## LEMBAR PENGESAHAN

Skripsi Saudara

Nama : Rheika Mosa Larasati  
NIM : 22EO10022  
Fakultas/Prodi : Fakultas MIKOM / Informatika  
Judul : Studi Komparasi Performa Algoritma *Extreme Gradient Boosting* (XGBoost) dan *Artificial Neural Network* (ANN) pada Sistem Prediksi Produksi Padi Berbasis Web

Telah disidangkan oleh Dewan Penguji Fakultas Matematika dan Ilmu Komputer Universitas Nahdlatul Ulama Al Ghazali (UNUGHA) Cilacap pada hari / tanggal :  
**Senin, 27 Juli 2026**

Dan dapat diterima sebagai pemenuhan tugas akhir mahasiswa Program Strata Satu (S.1) Program Studi Informatika (S.Kom) Fakultas Matematika dan Ilmu Komputer (FMikom) pada Universitas Nahdlatul Ulama Al Ghazali (UNUGHA) Cilacap.

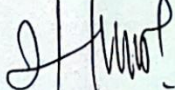
Cilacap, 5 Agustus 2026

Dewan Sidang

Ketua

  
M.T. Abdul Aziz Zein, S.Si., M.Kom.  
NIDN. 2125098601

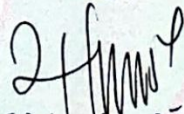
Sekretaris

  
Riski Aspriyani, M.Pd  
NIDN. 0616118901

Penguji 1

  
Abdul Haq, SE., M.Cs.  
NIDN. 0606067701

Penguji 2

  
Riski Aspriyani, M.Pd  
NIDN. 0616118901

Pembimbing I

  
M.T. Abdul Aziz Zein, S.Si., M.Kom.  
NIDN. 2125098601

Pembimbing II

  
Dr. M. Kharif, M.Pd.B.I  
NIDN. 0617058703

Mengetahui,  
Dekan Fakultas Matematika dan Ilmu Komputer



M.T. Abdul Aziz Zein, S.Si., M.Kom.  
NIDN. 2125098601

## HALAMAN NOTA KONSULTAN

### NOTA KONSULTAN

Hal : Skripsi Saudari Rheika Mosa Larasati  
Lampiran : -

Kepada  
Yth. Dekan Fakultas Matematika dan Ilmu Komputer  
Universitas Nahdlatul Ulama Al Ghazali Cilacap  
di Cilacap

*Assalamu'alaikum Wr. Wb.*

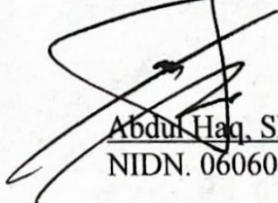
Setelah membaca, memeriksa, dan melakukan perbaikan seperlunya maka skripsi saudara:

Nama : Rheika Mosa Larasati  
NIM : 22EO10022  
Prodi : Informatika  
Judul : Studi Komparasi Performa Algoritma *Extreme Gradient Boosting* (XGBoost) dan *Artificial Neural Network* (ANN) pada Sistem Prediksi Produksi Padi Berbasis Web

Dapat diajukan ke Fakultas Matematika dan Ilmu Komputer, Universitas Nahdlatul Ulama Al Ghazali Cilacap untuk memenuhi syarat memperoleh gelar Strata Satu (S1).

*Wassalamu'alaikum Wr. Wb.*

Cilacap, 5 Agustus 2026  
Konsultan



Abdul Haq, SE., M.Cs.  
NIDN. 0606067701

## NOTA PEMBIMBING

Cilacap, 24 Juni 2026

Kepada Yth :  
Fakultas Matematika dan Komputer (FMIKOM)  
UNUGHA Cilacap  
di Cilacap

*Assalamu'alaikum Wr. Wb.*

Setelah melakukan bimbingan, telaah, arahan, dan koreksi tahap penulisan skripsi saudara:

Nama : Rheika Mosa Larasati  
NIM : 22EO10022  
Fakultas : Matematika dan Ilmu Komputer  
Prodi : Informatika  
Judul : Studi Komparasi Performa Algoritma *Extreme Gradient Boosting* (XGBoost) dan *Artificial Neural Network* (ANN) pada Sistem Prediksi Produksi Padi Berbasis Web

Kami berpendapat bahwa skripsi tersebut sudah dapat diajukan ke sidang skripsi. Bersamaan ini kami kirimkan skripsi tersebut, semoga dapat segera disidangkan.

Atas perhatiannya kami ucapkan terima kasih.  
*Wassalamu'alaikum Wr. Wb.*

Mengetahui,

Pembimbing I



M.T. Abdul Aziz Zein, S.Si., M.Kom.  
NIDN. 2125098601

Pembimbing II



Dr. M. Khanif, M.Pd.B.I  
NIDN. 0617058703

## HALAMAN MOTO

*“Maka, sesungguhnya beserta kesulitan ada kemudahan. Sesungguhnya beserta kesulitan ada kemudahan.”*

*(QS. Al-Insyirah: 5-6)*

*“Bahwa manusia hanya memperoleh apa yang telah diusahakannya.”*

*(QS. An-Najm: 39)*

*“Saat terasa berat-beratnya. Kutahu kau pun berjuang juga. Hadapi semuanya langsung di muka. Apa pun yang terjadi tidak apa.”*

*(Baskara Putra – Hindia)*

*“Hidup yang tidak sesuai impian bukanlah hidup yang gagal, dan hidup yang sesuai impian belum tentu hidup yang berhasil. Aku hanya ingin melakukan tugas yang diberikan kepadaku dengan baik”*

*(Baek Yijin – 2521)*

## HALAMAN PERSEMBAHAN

Segala puji bagi Allah SWT, Rabb semesta alam yang senantiasa memberikan karunia sehingga penulis mampu menyelesaikan penulisan skripsi ini. Karya ini saya persembahkan kepada :

1. Kedua orang tua tercinta, Bapak Narso dan Ibu Kasinah, yang selalu menjadi alasan saya untuk terus berjuang. Terima kasih atas setiap doa yang dipanjatkan, kasih sayang yang diberikan tanpa syarat, serta pengorbanan yang tidak akan pernah mampu saya balas dengan apa pun. Setiap langkah yang saya tempuh hingga berada pada titik ini tidak lepas dari *ridha*, dukungan, dan kepercayaan yang selalu kalian berikan. Terima kasih telah menjadi rumah paling aman dalam hidup saya, tempat saya kembali ketika lelah, tempat saya mengadu ketika merasa tidak mampu, dan tempat saya mendapatkan kekuatan untuk bangkit kembali. Tidak ada kata yang benar-benar mampu menggambarkan betapa besar peran kalian dalam perjalanan ini, namun skripsi ini adalah salah satu bentuk kecil dari rasa terima kasih dan usaha saya untuk membanggakan kalian.
2. Adik laki-laki saya, Abror Fauzi Ardhani, yang selalu membuat saya termotivasi untuk bisa terus belajar menjadi sosok kakak yang dapat memberikan pengaruh *positif*, baik dalam bidang akademik maupun *non-akademik*, serta berusaha menjadi panutan di masa yang akan datang.
3. Dosen pembimbing skripsi, bapak Mochamad T. A. Aziz Zein, S.Si., M.Kom dan bapak Dr. M. Khanif, M.Pd.B.I., yang dengan penuh kesabaran, ketulusan, dan perhatian telah membimbing, mengarahkan, serta memberikan ilmu dan motivasi kepada penulis selama proses penyusunan karya ini. Terima kasih atas setiap masukan, nasihat, dan waktu yang telah diberikan, sehingga penulis mampu menyelesaikan skripsi ini dengan baik.
4. Sahabat-sahabat tercinta, Faisya, Bela, dan Robbi. Terima kasih atas kehadiran kalian yang selalu menjadi warna dalam setiap perjalanan ini. Di tengah lelahnya proses, kalian selalu hadir membawa tawa, semangat, dan dukungan yang tidak pernah habis. Terima kasih karena selalu ada dalam suka maupun duka, menjadi tempat berbagi cerita, serta penguat ketika langkah mulai goyah.
5. “*TripleKill*” dan “*BMKG Squad*”, teman-teman seperjuangan yang dipertemukan sejak masa PKL. Terima kasih telah berjuang bersama dalam satu project, melewati setiap proses, kesulitan, kebingungan, dan tekanan hingga akhirnya dapat menyelesaikan semuanya bersama. Terima kasih atas kerja sama, dukungan, doa, dan kebersamaan yang hingga saat ini masih tetap terjaga. Semoga silaturahmi ini tetap bertahan, di mana pun langkah kita nantinya.
6. Seseorang yang tak kalah penting kehadirannya, Ichsan Wahyu Barokah. Terima kasih atas kesabaran, dukungan, doa, serta kesediaannya untuk selalu hadir dan menemani setiap proses, meluangkan waktu, tenaga, pikiran, maupun materi sejak awal perkuliahan hingga penulis berhasil menyelesaikan skripsi ini.

Terima kasih telah menjadi bagian dari perjalanan yang penuh pembelajaran dan perjuangan. Semoga segala doa, harapan, dan cita-cita baik yang kita impikan dapat terwujud dikemudian hari.

7. Teman-teman Informatika A, terima kasih atas kebersamaan, bantuan, dan semangat yang telah diberikan selama masa perkuliahan. Kalian telah menjadi bagian dari perjalanan ini yang penuh pengalaman dan kenangan berharga.
8. Keluarga besar Himafortic dan BEM FMIKOM, yang telah menjadi tempat bagi penulis untuk belajar, bertumbuh, dan berkembang selama menjalani proses perkuliahan. Terima kasih atas setiap kesempatan, kebersamaan, pengalaman, serta dukungan yang diberikan, sehingga penulis dapat memperoleh banyak pengalaman, baik di dalam maupun di luar kampus, serta menjadi pribadi yang lebih baik. Segala proses dan perjalanan yang telah dilewati bersama akan selalu menjadi bagian berharga dalam perjalanan hidup penulis.
9. “Keluarga Pohon Pete”, yang hingga saat ini masih terus menjaga silaturahmi dan tetap berteman dengan baik. Terima kasih atas kebersamaan, kerja sama, serta pengalaman yang telah dilalui bersama selama masa KKN. Meskipun waktu kebersamaan hanya sementara, namun cerita, tawa, dan proses yang kita jalani telah meninggalkan kesan yang mendalam. Semoga hubungan baik ini tetap terjaga dan terus berlanjut sebagai keluarga yang saling mendukung di masa depan.
10. *Last but not least*, anak perempuan pertama dan harapan orang tuanya, Rheika Mosa Larasati. Ya, diri saya sendiri. Apresiasi sebesar-besarnya untuk diri saya sendiri yang telah berjuang tanpa henti. Terima kasih telah bekerja keras dan bertahan hingga sejauh ini. Untuk setiap malam yang dihabiskan dalam kelelahan, setiap pagi yang disambut kekhawatiran, namun tetap dijalani dan berhasil dilalui. Terima kasih kepada hati yang tetap ikhlas, meski banyak hal yang di luar prediksi. Terima kasih kepada jiwa yang tetap kuat, meski berkali-kali hamper menyerah. Terima kasih kepada raga yang terus melangkah, meski lelah menghadapi tekanan tapi tetap diperjuangkan. Teruslah belajar dan menyukuri nikmat yang Tuhan berikan. Tetap semangat untuk terus berusaha, berbahagialah dimanapun kamu berada. Semoga langkah kecilmu selalu diperkuat, dikelilingi oleh orang-orang baik, serta mimpimu satu persatu terwujud. Aamiin.

## KATA PENGANTAR

Puji syukur kepada Allah SWT atas segala limpahan nikmat dan karunia-Nya, sehingga penulis dapat menyelesaikan penulisan skripsi. Sholawat dan salam senantiasa penulis haturkan kepada Nabi Muhammad SAW sebagai pembimbing seluruh umat manusia.

Skripsi ini tidak akan selesai tanpa adanya bantuan dari banyak pihak, karena itu penulis menyampaikan terima kasih kepada :

1. Bapak M.T. Abdul Aziz Zein, S.Si., M.Kom., Dekan FMIKOM UNUGHA dan dosen pembimbing I yang telah dengan sabar memberikan bimbingan dan petunjuk dalam menyelesaikan skripsi ini.
2. Bapak Dr. M. Khanif, M.Pd.B.I, pembimbing II yang telah dengan sabar memberikan bimbingan dan petunjuk dalam menyelesaikan skripsi ini.
3. Bapak Safiq Rosad, S.T., M.Kom., selaku Ketua Program Studi Informatika yang telah memberikan ilmu dan arahan selama masa perkuliahan.
4. Bapak Bagus Pramujo, S.Si., M.Sc., selaku Kepala Stasiun Meteorologi Tunggal Wulung Cilacap yang telah memberikan izin, dukungan, serta membantu menyediakan data iklim yang digunakan dalam penelitian ini.
5. Bapak Adnan Dendy Mardika, S.Kom., M.Si., selaku pembimbing Praktik Kerja Lapangan (PKL) yang telah memberikan arahan, bimbingan, serta masukan sejak pelaksanaan PKL hingga proses penyusunan skripsi.
6. Dinas Pertanian Kabupaten Cilacap yang telah memberikan dukungan dalam pelaksanaan penelitian sehingga penelitian ini dapat berjalan baik.
7. Bapak dan Ibu Dosen Program Studi Informatika FMIKOM UNUGHA Cilacap yang telah memberikan ilmu, pengalaman, serta wawasan selama penulis menempuh pendidikan.
8. Kedua orang tua yang selalu memberikan doa, dukungan, serta motivasi yang tiada henti kepada penulis.
9. Teman-teman seperjuangan yang telah memberikan bantuan, semangat, serta dukungan selama proses perkuliahan dan penyusunan skripsi.
10. Semua pihak yang tidak dapat penulis sebutkan satu per satu yang telah membantu dan memberikan dukungan dalam penyelesaian skripsi ini.

Semoga Allah SWT membalas jerih payah dan pengorbanan yang telah diberikan dengan balasan yang lebih baik. Aamiin. Penulis berharap semoga karya kecil ini bermanfaat bagi pembaca.

Cilacap, 5 Agustus 2026

Penulis

## ABSTRAK

RHEIKA MOSA LARASATI. Studi Komparasi Performa Algoritma *Extreme Gradient Boosting* (XGBoost) dan *Artificial Neural Network* (ANN) pada Sistem Prediksi Produksi Padi Berbasis Web. Dibimbing oleh M.T. ABDUL AZIZ ZEIN, S.Si., M.Kom. dan Dr. M. KHANIF, M.Pd.B.I.

Produksi padi merupakan salah satu faktor penting dalam menjaga ketahanan pangan dan mendukung sektor pertanian di Indonesia. Namun, produksi padi dipengaruhi oleh berbagai faktor iklim seperti lama penyinaran matahari, curah hujan, *Oceanic Niño Index* (ONI), suhu udara, dan kelembaban udara yang cenderung mengalami fluktuasi. Perubahan pola iklim yang tidak menentu dapat menyebabkan penurunan hasil panen sehingga diperlukan suatu metode yang mampu memprediksi produksi padi secara akurat sebagai dasar dalam mendukung perencanaan produksi dan pengambilan keputusan di bidang pertanian.

Penelitian ini bertujuan untuk membandingkan performa algoritma *Extreme Gradient Boosting* (XGBoost) dan *Artificial Neural Network* (ANN) dalam memprediksi produksi padi di Kabupaten Cilacap berdasarkan data iklim. Data yang digunakan merupakan data bulanan periode 2020–2024 yang diperoleh dari instansi terkait. Proses penelitian meliputi pengumpulan data dari instansi terkait, tahap pra-pemrosesan data, pembangunan dan pelatihan model, evaluasi performa menggunakan metrik *R-Square* ( $R^2$ ), *Mean Absolute Error* (MAE), dan *Root Mean Squared Error* (RMSE). Model dengan performa terbaik selanjutnya diimplementasikan ke dalam sistem berbasis web yang dikembangkan menggunakan metode *prototyping* dengan arsitektur monolitik.

Hasil penelitian menunjukkan bahwa algoritma XGBoost memiliki performa yang lebih baik dibandingkan ANN. Model XGBoost memperoleh nilai  $R^2$  sebesar 0,870, MAE sebesar 7.997,27 ton, dan RMSE sebesar 22.849,18 ton, sedangkan model ANN memperoleh nilai  $R^2$  sebesar 0,824, MAE sebesar 10.767,52 ton, dan RMSE sebesar 26.589,44 ton. Berdasarkan hasil tersebut, XGBoost dipilih sebagai model terbaik dan berhasil diimplementasikan ke dalam sistem prediksi produksi padi berbasis web. Hasil pengujian *Black Box Testing* menunjukkan bahwa seluruh fitur sistem dapat berjalan sesuai dengan kebutuhan fungsional yang telah ditetapkan. Dengan demikian, sistem yang dikembangkan mampu mendukung proses prediksi produksi padi secara efektif berdasarkan data iklim yang tersedia.

Kata kunci : *Artificial Neural Network*, *Extreme Gradient Boosting*, Prediksi Produksi Padi, Sistem Berbasis Web, Variabel Iklim.

## ABSTRACT

*RHEIKA MOSA LARASATI. Comparative Study of the Performance of Extreme Gradient Boosting (XGBoost) and Artificial Neural Networks (ANN) in a Web-Based Rice Production Prediction System. Supervised by M.T. ABDUL AZIZ ZEIN, S.Si., M.Kom. and Dr. M. KHANIF, M.Pd.B.I.*

*Rice production is one of the important factors in maintaining food security and supporting the agricultural sector in Indonesia. However, rice production is influenced by various climatic factors, such as sunshine duration, rainfall, Oceanic Niño Index (ONI), air temperature, and humidity, which tend to fluctuate over time. Unpredictable climate pattern changes can lead to decreased crop yields; therefore, a method capable of accurately predicting rice production is needed to support production planning and decision-making in the agricultural sector.*

*This study aims to compare the performance of the Extreme Gradient Boosting (XGBoost) and Artificial Neural Network (ANN) algorithms in predicting rice production in Cilacap Regency based on climate data. The data used consist of monthly climate and rice production data from 2020 to 2024 obtained from relevant institutions. The research process includes data collection, data preprocessing, model development and training, and performance evaluation using R-Square ( $R^2$ ), Mean Absolute Error (MAE), and Root Mean Squared Error (RMSE). The model with the best performance was then implemented in a web-based system developed using the prototyping method with a monolithic architecture.*

*The results show that the XGBoost algorithm outperformed ANN in predicting rice production. The XGBoost model achieved an  $R^2$  value of 0.870, an MAE of 7,997.27 tons, and an RMSE of 22,849.18 tons, while the ANN model achieved an  $R^2$  value of 0.824, an MAE of 10,767.52 tons, and an RMSE of 26,589.44 tons. Based on these results, XGBoost was selected as the best-performing model and successfully implemented in a web-based rice production prediction system. Black Box Testing results indicate that all system features functioned according to the specified functional requirements. Therefore, the developed system is capable of effectively supporting rice production prediction based on the available climate data.*

*Keywords: Artificial Neural Network, Climate Variables, Extreme Gradient Boosting, Rice Production Prediction, Web-Based System.*

## DAFTAR ISI

|                                                 |      |
|-------------------------------------------------|------|
| HALAMAN PERNYATAAN KEASLIAN SKRIPSI .....       | iii  |
| LEMBAR PENGESAHAN .....                         | iv   |
| HALAMAN NOTA KONSULTAN.....                     | v    |
| NOTA PEMBIMBING .....                           | vi   |
| HALAMAN MOTO.....                               | vii  |
| HALAMAN PERSEMBAHAN.....                        | viii |
| KATA PENGANTAR .....                            | x    |
| ABSTRAK .....                                   | xi   |
| <i>ABSTRACT</i> .....                           | xii  |
| DAFTAR ISI.....                                 | xiii |
| DAFTAR GAMBAR .....                             | xv   |
| DAFTAR TABEL.....                               | xvi  |
| BAB I PENDAHULUAN .....                         | 1    |
| A. Latar Belakang .....                         | 1    |
| B. Rumusan Masalah .....                        | 3    |
| C. Batasan Masalah.....                         | 3    |
| D. Tujuan Penelitian .....                      | 3    |
| E. Manfaat Penelitian .....                     | 4    |
| BAB II TINJAUAN PUSTAKA.....                    | 5    |
| A. Penelitian Terkait .....                     | 5    |
| B. Landasan Teori.....                          | 13   |
| 1. Produksi Padi.....                           | 13   |
| 2. Iklim .....                                  | 13   |
| 3. <i>Machine Learning</i> .....                | 14   |
| 4. Metode <i>Boosting</i> .....                 | 14   |
| 5. <i>Artificial Neural Network</i> (ANN).....  | 17   |
| 6. Bahasa Pemrograman Python.....               | 19   |
| 7. Parameter Evaluasi ( $R^2$ , MAE, RMSE)..... | 19   |
| 8. <i>Random Split</i> .....                    | 21   |
| 9. Metode <i>Quantile</i> .....                 | 21   |

|                                   |                                        |    |
|-----------------------------------|----------------------------------------|----|
| 10.                               | Sistem Berbasis Web .....              | 22 |
| 11.                               | <i>Framework FastAPI</i> .....         | 22 |
| 12.                               | <i>Metode Prototyping</i> .....        | 23 |
| 13.                               | <i>Black Box Testing</i> .....         | 23 |
| 14.                               | <i>Kolmogorov–Smirnov</i> .....        | 24 |
| 15.                               | <i>Wilcoxon Signed-Rank Test</i> ..... | 24 |
| BAB III METODOLOGI .....          |                                        | 25 |
| A.                                | Waktu dan Tempat Penelitian .....      | 25 |
| B.                                | Alat dan Bahan.....                    | 25 |
| C.                                | Prosedur Penelitian.....               | 26 |
| 1.                                | Studi Literatur.....                   | 27 |
| 2.                                | Identifikasi Kebutuhan .....           | 28 |
| 3.                                | Pembangunan dan Pelatihan Model .....  | 29 |
| 4.                                | Evaluasi dan Analisis Komparatif ..... | 36 |
| 5.                                | Implementasi Web.....                  | 36 |
| 6.                                | Pengujian Sistem .....                 | 44 |
| D.                                | Jadwal Penelitian.....                 | 47 |
| BAB IV HASIL DAN PEMBAHASAN ..... |                                        | 49 |
| A.                                | Hasil .....                            | 49 |
| 1.                                | Pengambilan Data.....                  | 49 |
| 2.                                | Pembangunan dan Pelatihan Model .....  | 52 |
| 3.                                | Evaluasi dan Analisis Komparatif ..... | 54 |
| 4.                                | Implementasi Web.....                  | 56 |
| 5.                                | Pengujian Sistem .....                 | 65 |
| B.                                | Pembahasan.....                        | 65 |
| BAB V KESIMPULAN .....            |                                        | 70 |
| A.                                | Kesimpulan .....                       | 70 |
| B.                                | Saran.....                             | 70 |
| DAFTAR PUSTAKA .....              |                                        | 72 |

## DAFTAR GAMBAR

|                                                                          |    |
|--------------------------------------------------------------------------|----|
| Gambar 1 MindMap Penelitian.....                                         | 12 |
| Gambar 2 Algoritma XGBoost .....                                         | 16 |
| Gambar 3 Algoritma ANN.....                                              | 17 |
| Gambar 4 Metode <i>Prototyping</i> .....                                 | 23 |
| Gambar 5 Alur Penelitian.....                                            | 27 |
| Gambar 6 Pembangunan dan Pelatihan Model .....                           | 29 |
| Gambar 7 Tahapan <i>Prototyping</i> .....                                | 37 |
| Gambar 8 <i>Use case</i> Web Sistem Prediksi .....                       | 38 |
| Gambar 9 <i>Activity Diagram</i> User .....                              | 39 |
| Gambar 10 <i>Activity Diagram</i> Admin.....                             | 40 |
| Gambar 11 <i>Sequence Diagram</i> User .....                             | 41 |
| Gambar 12 <i>Sequence Diagram</i> Admin.....                             | 42 |
| Gambar 13 <i>Gantt Chart</i> Jadwal Penelitian .....                     | 48 |
| Gambar 14 Korelasi Variabel Iklim dan Produksi Padi .....                | 52 |
| Gambar 15 Grafik Prediksi Produksi Padi (XGBoost) .....                  | 53 |
| Gambar 16 Grafik Prediksi Produksi Padi (ANN).....                       | 54 |
| Gambar 17 Grafik Hasil Prediksi XGBoost dan ANN .....                    | 55 |
| Gambar 18 Grafik Perbandingan <i>Absolute Error</i> XGBoost dan ANN..... | 55 |
| Gambar 19 Arsitektur Sistem.....                                         | 57 |
| Gambar 20 Halaman Beranda .....                                          | 59 |
| Gambar 21 Halaman Pemilihan Komoditas.....                               | 60 |
| Gambar 22 Halaman Informasi Prediksi.....                                | 60 |
| Gambar 23 Halaman Grafik Hasil Pelatihan Model .....                     | 61 |
| Gambar 24 Halaman Prediksi Produksi Padi .....                           | 61 |
| Gambar 25 Halaman Rekomendasi.....                                       | 62 |
| Gambar 26 Halaman <i>Login</i> Admin.....                                | 62 |
| Gambar 27 Halaman <i>Dashboard</i> Admin .....                           | 63 |
| Gambar 28 Halaman Kelola Prediksi Padi.....                              | 63 |
| Gambar 29 Halaman Kelola Dataset.....                                    | 64 |
| Gambar 30 Halaman Simulasi Prediksi .....                                | 64 |
| Gambar 31 Halaman Riwayat Dataset .....                                  | 65 |

## DAFTAR TABEL

|                                                               |    |
|---------------------------------------------------------------|----|
| Tabel 1 Penelitian Terkait .....                              | 5  |
| Tabel 2 Parameter Perbandingan .....                          | 19 |
| Tabel 3 Alat dan Bahan.....                                   | 25 |
| Tabel 4 Tabel Hasil Evaluasi .....                            | 36 |
| Tabel 5 Skenario Pengujian User .....                         | 44 |
| Tabel 6 Skenario Pengujian Admin .....                        | 45 |
| Tabel 7 Spesifikasi Data.....                                 | 49 |
| Tabel 8 Contoh Data .....                                     | 50 |
| Tabel 9 Hasil Analisis .....                                  | 56 |
| Tabel 10 Struktur Database .....                              | 59 |
| Tabel 11 Hasil Uji Normalitas <i>Kolmogorov-Smirnov</i> ..... | 67 |
| Tabel 12 Hasil Uji <i>Wilcoxon Signed-Rank Test</i> .....     | 68 |

# BAB I PENDAHULUAN

## A. Latar Belakang

Pertanian merupakan salah satu sektor strategis yang memiliki peran penting dalam menjaga ketahanan pangan nasional sebagai bahan pangan utama masyarakat Indonesia [1]. Produksi padi yang stabil menjadi faktor krusial dalam mendukung pasokan pangan dan kestabilan ekonomi [2]. Namun, produksi padi sangat dipengaruhi oleh berbagai faktor, terutama kondisi iklim seperti lama penyinaran matahari, curah hujan, suhu udara, kelembaban, dan fenomena iklim global seperti El Niño dan La Niña yang dapat menyebabkan anomali cuaca [3][4][5]. Perubahan pola iklim yang tidak menentu berpotensi memengaruhi pertumbuhan tanaman padi dan berdampak langsung terhadap hasil produksi [3][6].

Produksi padi cenderung mengalami fluktuasi akibat ketidakpastian kondisi iklim yang terjadi setiap tahunnya [7]. Perubahan curah hujan yang ekstrem, peningkatan suhu, maupun gangguan iklim lainnya dapat menyebabkan penurunan hasil panen atau bahkan gagal panen di beberapa wilayah [8]. Berdasarkan data dari Dinas Pertanian dan Badan Meteorologi, Klimatologi, dan Geofisika (BMKG) mengenai produksi padi dan variabel iklim selama periode 2020-2024 di Kabupaten Cilacap, terlihat adanya variasi hasil panen yang menunjukkan kecenderungan tidak stabil dari waktu ke waktu. Fluktuasi tersebut menunjukkan adanya hubungan yang erat antara perubahan kondisi iklim dengan tingkat produksi padi, sehingga sulit untuk memprediksi hasil panen secara konvensional. Kondisi ini menimbulkan tantangan dalam perencanaan produksi, distribusi pangan, serta pengambilan kebijakan oleh instansi terkait seperti Badan Meteorologi, Klimatologi, dan Geofisika (BMKG) serta Dinas Pertanian. Oleh karena itu, diperlukan suatu pendekatan yang mampu memodelkan hubungan antara variabel iklim dan produksi padi secara lebih akurat untuk mendukung proses prediksi.

Untuk menangani permasalahan tersebut, diperlukan metode analisis yang mampu mengidentifikasi pola serta hubungan non-linear antara variabel iklim dan produksi padi. Seiring dengan perkembangan teknologi, *machine learning* menjadi salah satu pendekatan yang digunakan dalam proses pemodelan dan prediksi [9][10][11]. Pendekatan ini memiliki kemampuan dalam menangani data kompleks dan dapat memodelkan hubungan *non-linear* yang sulit dijelaskan melalui metode statistik konvensional [12]. Menurut Indri Hapsari dan Shifa Pandya Wisesa, algoritma *machine learning* berperan penting dalam analisis prediktif dengan memanfaatkan data historis dan pola data saat ini untuk memprediksi kejadian di masa depan [13]. Dengan memanfaatkan data historis variabel iklim dan produksi padi, model prediksi dapat dikembangkan untuk menghasilkan estimasi produksi berdasarkan pola yang dipelajari dari data [14].

*Extreme Gradient Boosting* (XGBoost) memiliki kemampuan dalam menghasilkan performa prediksi yang tinggi serta stabil pada data numerik yang tersusun dalam bentuk tabel, seperti data iklim dan produksi pertanian, yang umumnya memiliki karakteristik dan hubungan antarvariabel yang kompleks [15][16]. XGBoost merupakan algoritma berbasis *ensemble learning* yang mengembangkan metode *gradient boosting* dengan menerapkan teknik optimasi dan regularisasi untuk meningkatkan performa model serta mengurangi risiko *overfitting* [17][18]. Algoritma ini membangun sejumlah *pohon* secara bertahap, yang setiap pohonnya memperbaiki kesalahan model sebelumnya sehingga mampu menangkap pola hubungan yang sulit dideteksi secara linear [11][19][20]. Karakteristik tersebut menjadikan XGBoost sesuai digunakan pada penelitian yang berbentuk data tabular numerik dan memiliki hubungan *non-linear* antarvariabel.

Sementara itu, *Artificial Neural Network* (ANN) merupakan algoritma yang memiliki kemampuan untuk memodelkan hubungan *non-linear* pada data yang sangat kompleks melalui struktur jaringan yang terdiri dari beberapa lapisan *neuron* [21][22]. Pendekatan ini efektif dalam mengenali pola tersembunyi (*hidden layer*) dan mampu menghasilkan prediksi berdasarkan pola yang dipelajari dari data historis [23][24][25]. Dengan karakteristik tersebut, ANN dinilai relevan untuk mempelajari hubungan antara variabel iklim dan produksi padi yang cenderung fluktuatif [26]. Kemampuan ANN dalam mempelajari pola *non-linear* menjadikan metode ini sesuai untuk digunakan pada data penelitian yang terdiri atas beberapa variabel iklim sebagai variabel input dan produksi padi sebagai variabel target.

Pemilihan XGBoost dan ANN dalam penelitian ini didasarkan pada kesesuaian karakteristik kedua metode dengan data yang digunakan. XGBoost sesuai untuk mengolah data tabular numerik serta mempelajari hubungan dan interaksi antarvariabel, sedangkan ANN mampu mempelajari pola serta hubungan *non-linear* yang kompleks melalui jaringan neuron. Meskipun memiliki mekanisme pembelajaran yang berbeda, kedua algoritma dapat digunakan untuk menyelesaikan permasalahan regresi, yaitu memprediksi nilai produksi padi berdasarkan variabel input yang tersedia. Oleh karena itu, kedua algoritma dibandingkan untuk mengetahui metode yang menghasilkan performa lebih baik pada dataset produksi padi Kabupaten Cilacap periode 2020–2024.

Implementasi model prediksi ke dalam sistem berbasis web dipilih karena beberapa alasan strategis. Sistem berbasis web memudahkan akses dari berbagai perangkat tanpa perlu instalasi *software*, cukup dengan menggunakan browser internet yang tersedia di berbagai perangkat seperti komputer, tablet, maupun *smartphone* [27]. Pembaruan model dan data dapat dilakukan secara terpusat di server sehingga seluruh pengguna otomatis mengakses versi terbaru tanpa perlu update secara manual [28]. Platform web juga merupakan antarmuka yang familiar

dan mudah dipahami oleh pengguna dengan berbagai tingkat literasi teknologi, mulai dari petani, petugas dinas pertanian, serta pihak-pihak terkait lainnya [29].

Karena pentingnya akurasi dan efisiensi dalam proses pengambilan keputusan berbasis data, penelitian ini akan membandingkan dua pendekatan *machine learning*, yaitu *Extreme Gradient Boosting* (XGBoost) dan *Artificial Neural Network* (ANN), dalam memprediksi produksi padi berdasarkan variabel iklim. Perbandingan dilakukan menggunakan dataset, variabel input, variabel target, pembagian data, dan metrik evaluasi yang sama agar performa kedua model dapat dievaluasi pada kondisi pengujian yang setara. Metode dengan performa terbaik selanjutnya akan diimplementasikan ke dalam sistem prediksi produksi padi berbasis web agar dapat diakses lebih mudah oleh pengguna. Sistem ini dirancang untuk membantu proses estimasi produksi padi berdasarkan data variabel iklim serta mendukung penentuan langkah strategis dan upaya antisipatif dalam menghadapi potensi risiko penurunan hasil panen. Dengan adanya sistem berbasis web, hasil prediksi dapat dimanfaatkan oleh instansi terkait seperti Badan Meteorologi, Klimatologi, dan Geofisika (BMKG) dan Dinas Pertanian dalam mendukung perencanaan produksi dan pengendalian risiko pertanian.

## **B. Rumusan Masalah**

Berdasarkan latar belakang yang telah disebutkan, maka perumusan masalah pada penelitian ini adalah:

1. Bagaimana melakukan perbandingan performa algoritma *Extreme Gradient Boosting* (XGBoost) dan *Artificial Neural Network* (ANN) dalam memprediksi produksi padi berdasarkan variabel iklim yang digunakan?
2. Bagaimana mengimplementasikan algoritma dengan performa terbaik ke dalam sistem prediksi produksi padi berbasis web?

## **C. Batasan Masalah**

Batasan masalah yang diambil pada penelitian ini yaitu :

1. Data yang digunakan mencakup variabel iklim sebagai variabel input yaitu lama penyinaran matahari, curah hujan, suhu, kelembaban, dan *Oceanic Niño Index* (ONI), serta produksi padi sebagai variabel target. Data yang merupakan data bulanan dari 2020-2024 dan hanya mencakup wilayah Kabupaten Cilacap.
2. Implementasi model dilakukan menggunakan bahasa pemrograman Python.
3. Sistem berbasis web berfokus pada prediksi jumlah produksi padi Kabupaten Cilacap dalam satuan ton, yang dilengkapi dengan kategori produksi dan rekomendasi. Sistem tidak mencakup fitur analisis spasial atau integrasi dengan sistem eksternal lainnya.

## **D. Tujuan Penelitian**

Tujuan penelitian berdasarkan rumusan masalah di atas adalah :

1. Mengetahui dan membandingkan performa algoritma *Extreme Gradient Boosting* (XGBoost) dan *Artificial Neural Network* (ANN) dalam memprediksi produksi padi berdasarkan variabel iklim.
2. Mengimplementasikan algoritma dengan performa terbaik ke dalam sistem prediksi produksi padi berbasis web.

#### **E. Manfaat Penelitian**

Manfaat teoritis dalam penelitian ini yaitu:

1. Memberikan penjelasan mengenai perbandingan algoritma *Extreme Gradient Boosting* (XGBoost) dan *Artificial Neural Network* (ANN) dalam memprediksi produksi padi berdasarkan variabel iklim.
2. Menjadi literatur bagi mahasiswa atau peneliti selanjutnya dalam penelitian terkait dengan komparasi metode *machine learning*.
3. Menjelaskan mengenai penerapan model prediksi pada sistem berbasis web.

Manfaat praktis dalam penelitian ini yaitu:

1. Mengetahui algoritma yang paling optimal dalam memprediksi produksi padi.
2. Menghasilkan sistem prediksi berbasis web yang mendukung proses analisis dan estimasi produksi secara sistematis.
3. Mendukung instansi terkait seperti BMKG serta Dinas Pertanian dalam melakukan perencanaan produksi dan upaya antisipatif terhadap potensi penurunan hasil panen di Kabupaten Cilacap.

## BAB II TINJAUAN PUSTAKA

### A. Penelitian Terkait

Berikut penelitian-penelitian terdahulu yang berkaitan dengan penerapan metode *Extreme Gradient Boosting* (XGBoost), *Artificial Neural Network* (ANN), serta pengembangan sistem prediksi berbasis web.

Tabel 1 Penelitian Terkait

| No  | Nama (Tahun)                                        | Judul                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
|-----|-----------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| (1) | (2)                                                 | (3)                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
| 1   | R. Illaihiyah, G. Abdurrahman, Daryanto (2026)      | <b>Judul:</b> Implementasi Algoritma <i>XGBoost Classifier</i> untuk Klasifikasi Kerumunan Jemaah Haji dan Umrah<br><b>Metode:</b> XGBoost<br><b>Hasil Penelitian:</b><br>Berdasarkan penelitian ini, Algoritma <i>Extreme Gradient Boosting</i> (XGBoost) menunjukkan kinerja klasifikasi yang sangat tinggi dalam memprediksi kepadatan kerumunan jemaah, dengan akurasi 98,7% serta presisi, recall, dan f1-score pada rentang 98–99% setelah dioptimasi menggunakan <i>Randomized Search CV</i> dan pembobotan kelas. |
| 2   | S. Ruscikasani, R. Roro, N. Oktalivia et al. (2025) | <b>Judul:</b> Prediksi Pembelian <i>E-Commerce</i> menggunakan XGBoost Berbasis Perilaku Sesi Pengguna<br><b>Metode:</b> XGBoost<br><b>Hasil Penelitian:</b><br>Penelitian ini menunjukkan bahwa algoritma <i>Extreme Gradient Boosting</i> (XGBoost) mampu memprediksi keputusan pembelian pada platform <i>e-commerce</i> dengan akurasi 0,785 dan ROC-AUC 0,804, serta precision, recall, dan F1-score yang menunjukkan                                                                                                |

| No  | Nama (Tahun)                           | Judul                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
|-----|----------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| (1) | (2)                                    | (3)                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
|     |                                        | kemampuan model dalam membedakan sesi berpotensi pembelian.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
| 3   | G. Cepi, F. Umbara, H. Ashaury (2025)  | <p><b>Judul:</b> Prediksi Pembatalan Reservasi Hotel menggunakan Metode Xgboost</p> <p><b>Metode:</b> XGBoost</p> <p><b>Hasil Penelitian:</b></p> <p>Penelitian ini menunjukkan bahwa algoritma <i>Extreme Gradient Boosting</i> (XGBoost) mampu memprediksi pembatalan reservasi hotel dengan akurasi 94,15% serta unggul dalam presisi, recall, dan F1-score. Faktor utama yang memengaruhi pembatalan adalah riwayat pembatalan, perubahan reservasi, dan permintaan khusus, dengan potensi pengembangan melalui teknik <i>oversampling</i> dan studi kasus yang lebih luas.</p> |
| 4   | A. Saputra, B. Sari, C. Rozikin (2024) | <p><b>Judul:</b> Penerapan Algoritma <i>Extreme Gradient Boosting</i> (XGBoost) untuk Analisis Risiko Kredit</p> <p><b>Metode:</b> XGBoost</p> <p><b>Hasil Penelitian:</b></p> <p>Berdasarkan penelitian ini, hasil evaluasi menunjukkan bahwa algoritma <i>Extreme Gradient Boosting</i> (XGBoost) efektif untuk analisis kelayakan kredit dan mengalami peningkatan performa setelah penerapan SMOTE, dengan skenario terbaik (90% data latih dan 10% data uji) menghasilkan <i>accuracy</i> 0,83 dan AUC 0,918.</p>                                                              |
| 5   | W. Kurniawan, C. Indahyanti (2024)     | <p><b>Judul:</b> Prediksi Angka Harapan Hidup Penduduk menggunakan Metode XGBoost</p> <p><b>Metode:</b> XGBoost</p> <p><b>Hasil Penelitian:</b></p>                                                                                                                                                                                                                                                                                                                                                                                                                                 |

| No  | Nama (Tahun)                                  | Judul                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
|-----|-----------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| (1) | (2)                                           | (3)                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
|     |                                               | Penelitian ini menghasilkan model <i>Extreme Gradient Boosting</i> (XGBoost Regressor) yang mampu memprediksi angka harapan hidup dengan akurasi 96,8% dan MAE sebesar 0,97, menunjukkan efektivitasnya sebagai alat estimasi untuk mendukung perencanaan kebijakan berbasis data di kawasan Asia.                                                                                                                                                                                                                                                                                                                                                                                                |
| 6   | H. Fadil, E. Rouza, D. Suherna et al. (2026)  | <p><b>Judul:</b> Implementasi <i>Artificial Neural Network</i> ( ANN ) untuk Memprediksi Cuaca Harian berdasarkan Data Suhu dan Kelembapan</p> <p><b>Metode:</b> ANN</p> <p><b>Hasil Penelitian:</b></p> <p>Penelitian ini menunjukkan bahwa <i>Artificial Neural Network</i> (ANN) mampu memprediksi kondisi cuaca harian berbasis suhu dan kelembapan dengan akurasi 89% serta kemampuan generalisasi yang baik dalam menangkap pola non-linear. Model ini juga berhasil diimplementasikan dalam aplikasi web sebagai sistem prediksi yang praktis, dengan potensi pengembangan melalui penambahan variabel meteorologi dan penggunaan arsitektur <i>deep learning</i> yang lebih kompleks.</p> |
| 7   | S. Anugrahaini, R. Fatchan, T. Wiyatno (2025) | <p><b>Judul:</b> Model Prediksi Tingkat Stres di Kalangan Siswa dengan Pendekatan Algoritma <i>Artificial Neural Network</i></p> <p><b>Metode:</b> ANN</p> <p><b>Hasil Penelitian:</b></p> <p>Penelitian ini menunjukkan bahwa algoritma <i>Neural Network</i> mampu memprediksi kelulusan mahasiswa dengan presisi 72,32%, recall 82,53%, dan akurasi 73,21%, sehingga</p>                                                                                                                                                                                                                                                                                                                       |

| No  | Nama (Tahun)                                         | Judul                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
|-----|------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| (1) | (2)                                                  | (3)                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
|     |                                                      | cukup baik dalam mengidentifikasi mahasiswa yang berpotensi lulus. Untuk meningkatkan performa, diperlukan peningkatan kualitas data, optimasi <i>hyperparameter</i> , serta penerapan validasi silang dan evaluasi berkala.                                                                                                                                                                                                                                                                                                |
| 8   | R. Pratama, R. Herdiana, R. Hamonangan et al. (2024) | <p><b>Judul:</b> Analisis Prediksi Kelulusan Mahasiswa menggunakan Metode <i>Artificial Neural Network</i></p> <p><b>Metode:</b> ANN</p> <p><b>Hasil Penelitian:</b></p> <p>Penelitian ini menunjukkan bahwa algoritma <i>Neural Network</i> mampu memprediksi kelulusan mahasiswa dengan presisi 72,32%, recall 82,53%, dan akurasi 73,21%, sehingga cukup andal dalam mengidentifikasi mahasiswa yang berpotensi lulus.</p>                                                                                               |
| 9   | J. Veronika, Andri (2022)                            | <p><b>Judul:</b> Penerapan Metode Algoritma <i>Neural Network</i> untuk Memprediksi Penjualan Bahan Bakar Minyak</p> <p><b>Metode:</b> ANN</p> <p><b>Hasil Penelitian:</b></p> <p>Penelitian ini menggunakan 4.324 data transaksi dari PT Dempo Migas / SPBU 24.31550 dan menghasilkan akurasi tertinggi pada produk Biosolar sebesar 99,83% (skema 80:20) melalui pengolahan data mining menggunakan RapidMiner dengan atribut Nozzel, nomor penjualan, jumlah penjualan, bulan, tahun, dan bahan bakar sebagai label.</p> |
| 10  | S. Prasetyo, T. Efendi, Muqorobin (2024)             | <p><b>Judul:</b> Implementasi Sistem Prediksi Curah Hujan dengan Penerapan Jaringan Syaraf Tiruan Berbasis Website</p>                                                                                                                                                                                                                                                                                                                                                                                                      |

| No  | Nama (Tahun)                   | Judul                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
|-----|--------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| (1) | (2)                            | (3)                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
|     |                                | <p><b>Metode:</b> ANN</p> <p><b>Hasil Penelitian:</b></p> <p>Penelitian ini membangun sistem prediksi curah hujan bernama RainCast ID menggunakan <i>Neural Network</i> metode <i>Backpropagation</i> dengan arsitektur 4-4-1 berdasarkan data historis cuaca Indonesia tahun 2023–Mei 2024. Hasil pengujian menunjukkan bahwa peningkatan epoch dan <i>learning rate</i> (0,5) mampu meningkatkan akurasi dari 54,34% (epoch 20) menjadi 87,43% (epoch 500), dan sistem diimplementasikan menggunakan Flask berbasis Python.</p>                                                                     |
| 11  | A. Pangestu, S. Mujiono (2025) | <p><b>Judul:</b> Implementasi Algoritma XGBoost untuk Prediksi Status Gizi Balita Berbasis Website</p> <p><b>Metode:</b> XGBoost</p> <p><b>Hasil Penelitian:</b></p> <p>Penelitian ini berhasil mengembangkan sistem prediksi status gizi balita menggunakan <i>Extreme Gradient Boosting</i> (XGBoost) dengan akurasi 97,17% serta presisi, recall, dan F1-score sekitar 97% setelah optimasi <i>Grid Search</i>, berdasarkan 5.169 data berkualitas hasil data cleaning. Sistem berbasis web ini berpotensi menjadi alat skrining awal yang efisien untuk mendukung program penurunan stunting.</p> |
| 12  | S. Grace, D. Gunawan (2025)    | <p><b>Judul:</b> Perbandingan CNN, Resnet50, dan <i>Vision Transformer</i> Untuk Klasifikasi Kanker Payudara Berbasis Web</p> <p><b>Metode:</b> CNN, Resnet50, dan <i>Vision Transformer</i></p>                                                                                                                                                                                                                                                                                                                                                                                                      |

| No  | Nama (Tahun)                   | Judul                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
|-----|--------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| (1) | (2)                            | (3)                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
|     |                                | <p><b>Hasil Penelitian:</b></p> <p>Berdasarkan evaluasi tiga model <i>deep learning</i>, ResNet50 menunjukkan performa terbaik dan paling konsisten dalam deteksi kanker payudara dari citra mammografi, diikuti SimpleCNN, sementara Vision Transformer (ViT) memiliki performa terendah akibat keterbatasan data pelatihan. Ketiga model berhasil diimplementasikan dalam aplikasi berbasis Flask.</p>                                                                                                                                               |
| 13  | W. Su, F. Jiang et al. (2023)  | <p><b>Judul:</b> <i>An XGBoost - Based Knowledge Tracing Model</i></p> <p><b>Metode:</b> XGBoost</p> <p><b>Hasil Penelitian:</b></p> <p>Penelitian ini menunjukkan bahwa algoritma <i>Extreme Gradient Boosting</i> (XGBoost) dapat diterapkan secara efektif pada model <i>knowledge tracing</i> dengan nilai AUC terbaik mencapai 0,9855 saat menggunakan multi-fitur. Dibandingkan model <i>deep learning</i> sebelumnya, XGBoost juga lebih efisien dalam waktu pelatihan serta lebih mudah diimplementasikan pada platform pendidikan daring.</p> |
| 14  | S. Arya, Anju, N. Ramli (2024) | <p><b>Judul:</b> <i>Predicting the stress level of students using Supervised Machine Learning and Artificial Neural Network (ANN)</i></p> <p><b>Metode:</b> ANN</p> <p><b>Hasil Penelitian:</b></p> <p>Penelitian ini menggabungkan berbagai model <i>machine learning</i> dan <i>deep learning</i> seperti <i>Support Vector Machine</i>, <i>Naive Bayes</i>, dan <i>Extreme Gradient Boosting</i> untuk mengidentifikasi faktor penyebab stres pada</p>                                                                                              |

| No  | Nama (Tahun) | Judul                                                                                                         |
|-----|--------------|---------------------------------------------------------------------------------------------------------------|
| (1) | (2)          | (3)                                                                                                           |
|     |              | mahasiswa, dengan akurasi tertinggi diperoleh <i>Naïve Bayes</i> sebesar 90% dan terendah SVM sebesar 85,45%. |

Berdasarkan kajian terhadap penelitian-penelitian terdahulu, diketahui bahwa metode *Extreme Gradient Boosting* (XGBoost) dan *Artificial Neural Network* (ANN) telah banyak diterapkan dalam berbagai bidang seperti klasifikasi, analisis risiko, dan prediksi berbasis web. Namun, sebagian besar penelitian tersebut hanya berfokus pada penerapan satu metode tanpa melakukan perbandingan performa secara langsung pada permasalahan yang sama. Selain itu, penelitian komparatif yang ada umumnya dilakukan pada bidang kesehatan, pendidikan, atau klasifikasi citra, sehingga masih terbatas penelitian yang membandingkan kedua metode tersebut dalam konteks prediksi produksi pertanian berbasis variabel iklim.

Selain itu, belum banyak penelitian yang mengintegrasikan hasil komparasi algoritma ke dalam sistem prediksi berbasis web sebagai bentuk implementasi nyata untuk mendukung pengambilan keputusan. Oleh karena itu, penelitian ini bertujuan untuk membandingkan performa metode XGBoost dan ANN dalam memprediksi produksi padi berdasarkan variabel iklim, serta mengimplementasikan metode dengan performa terbaik ke dalam sistem prediksi berbasis web guna mendukung perencanaan produksi dan pengambilan kebijakan berbasis data. Diagram *mind map* dijelaskan pada Gambar 1.



Gambar 1 *MindMap* Penelitian

## B. Landasan Teori

Landasan teori pada penelitian “Studi Komparasi Performa *Algoritma Extreme Gradient Boosting (XGBoost)* dan *Artificial Neural Network (ANN)* pada Sistem Prediksi Produksi Padi Berbasis Web” yaitu:

### 1. Produksi Padi

Padi (*Oryza sativa L.*) merupakan tanaman pangan pokok bagi sebagian besar penduduk di Asia, termasuk Indonesia. Sebagai komoditas strategis, padi menjadi tulang punggung ketahanan pangan nasional yang menyumbang kebutuhan kalori terbesar masyarakat Indonesia. Menurut Badan Pusat Statistik, produksi padi di Kabupaten Cilacap pada tahun 2024 sebanyak 651,14 ribu ton gabah kering giling (GKG), dan mengalami penurunan 69,78 ribu ton (15,71 persen) dibandingkan dengan produksi tahun 2023. Fluktuasi produksi dari tahun ke tahun akibat pengaruh iklim dan kondisi lahan [30].

Produksi padi dipengaruhi oleh dua kelompok faktor utama, yaitu teknis dan sosial-ekonomi. Faktor teknis meliputi varietas padi yang digunakan, kualitas benih, teknik budidaya yang tepat, serta efisiensi pemupukan. Sementara itu, sosial-ekonomi mencakup kondisi tanah, ketersediaan air irigasi, ketidakpastian musim tanam, serta kondisi iklim [1]. Di antara faktor-faktor tersebut, kondisi iklim menjadi variabel yang paling sulit dikendalikan namun memberikan pengaruh yang dominan terhadap fluktuasi produksi padi.

### 2. Iklim

Badan Meteorologi, Klimatologi, dan Geofisika (BMKG) mendefinisikan iklim sebagai pola rata-rata cuaca dalam jangka waktu panjang (biasanya 30 tahun atau lebih) yang terjadi di suatu wilayah. Iklim berbeda dengan cuaca, karena cuaca bersifat harian dan dapat berubah dalam waktu singkat, sedangkan iklim menunjukkan kecenderungan jangka panjang [31]. Secara lebih spesifik, iklim merupakan sintesis atau rangkuman kejadian cuaca dalam periode waktu yang panjang, yang kemudian dapat digunakan untuk mengidentifikasi karakteristik dan pola cuaca suatu wilayah [32].

Indonesia memiliki iklim tropis yang dicirikan oleh curah hujan tinggi, suhu udara relatif stabil sepanjang tahun, kelembaban relatif tinggi, dan lama penyinaran matahari yang relatif konstan [33]. Curah hujan merupakan sumber air utama bagi pertanian padi dengan kebutuhan optimal berkisar 200-300 mm per bulan selama musim tanam, sementara suhu optimal untuk pertumbuhan padi berkisar antara 25-30°C dengan suhu ekstrem di atas 35°C atau di bawah 15°C dapat menyebabkan gangguan pada proses reproduksi tanaman[3][34]. Kelembaban relatif yang optimal berada pada kisaran 70-80%. Kelembaban terlalu tinggi dapat meningkatkan risiko penyakit jamur, sedangkan lama penyinaran matahari yang cukup (6-8 jam per hari) sangat diperlukan untuk proses fotosintesis dan pengisian bulir padi [35][36].

Pola iklim Indonesia sangat dipengaruhi oleh berbagai fenomena atmosfer dan oseanik seperti *El Nino-Southern Oscillation* (ENSO) yang berperan penting dalam variabilitas iklim tahunan. Indeks ENSO menjadi indikator penting dalam memprediksi variabilitas iklim yang berpengaruh terhadap produktivitas pertanian di Indonesia. Pada fase *El Niño* terjadi kecenderungan kemarau panjang yang berdampak pada penurunan produksi padi, sedangkan pada fase *La Niña* terjadi peningkatan curah hujan yang berpotensi menimbulkan banjir dan gagal panen. [37]. Kelima variabel iklim tersebut—lama penyinaran matahari, curah hujan, suhu udara, kelembaban relatif, dan Indeks ENSO—memiliki korelasi dengan produksi padi dan menjadi variabel input utama dalam penelitian ini.

### 3. *Machine Learning*

*Machine Learning* (pembelajaran mesin) merupakan cabang dari kecerdasan buatan (*Artificial Intelligence*) yang memungkinkan sistem komputer memiliki kemampuan untuk belajar secara otomatis dari data dan pengalaman tanpa harus diprogram secara eksplisit untuk setiap tugas tertentu [22]. Konsep ini pertama kali diperkenalkan oleh Arthur Samuel pada tahun 1959 yang mendefinisikan *machine learning* sebagai bidang studi yang memberikan kemampuan komputer untuk belajar tanpa diprogram secara eksplisit. Melalui proses pembelajaran dari data, *machine learning* dapat mengidentifikasi pola dan hubungan antarvariabel untuk menghasilkan prediksi atau keputusan [12].

Secara umum, *machine learning* terdiri atas beberapa kategori, salah satunya adalah *supervised learning* yang merupakan metode pembelajaran yang menggunakan data yang telah memiliki variabel input dan variabel target. Model mempelajari hubungan antara variabel input dan variabel target berdasarkan data historis, kemudian menggunakan pola yang telah dipelajari untuk menghasilkan prediksi terhadap data baru [38].

Permasalahan dalam *supervised learning* dapat diselesaikan menggunakan berbagai algoritma, seperti *Linear Regression*, *Support Vector Regression* (SVR), *Decision Tree*, *Random Forest*, *XGBoost*, *AdaBoost*, *CatBoost*, dan *Artificial Neural Network* (ANN). Setiap algoritma memiliki mekanisme pembelajaran dan karakteristik yang berbeda sehingga pemilihannya perlu disesuaikan dengan karakteristik data serta tujuan penelitian [25]. Dalam penelitian ini, *machine learning* digunakan untuk mempelajari hubungan antara variabel iklim sebagai variabel input dan jumlah produksi padi sebagai variabel target.

### 4. *Metode Boosting*

*Boosting* merupakan salah satu metode yang menggabungkan beberapa model sederhana (*weak learner*) untuk membentuk model prediksi yang lebih

kuat dan akurat. Pada metode *boosting*, model dibangun secara bertahap dan berurutan (*sequential*). Setiap model baru berfokus untuk memperbaiki kesalahan yang dihasilkan oleh model sebelumnya sehingga kemampuan prediksi dapat meningkat secara bertahap. Hasil dari seluruh model kemudian digabungkan untuk menghasilkan prediksi akhir [39].

Beberapa algoritma yang menggunakan metode *boosting* antara lain AdaBoost, CatBoost, LightGBM dan XGBoost.

a. *Adaptive Boosting* (AdaBoost)

*Adaptive Boosting* (AdaBoost) merupakan salah satu algoritma *boosting* yang membangun beberapa model secara bertahap. Pada setiap proses pembelajaran, AdaBoost memberikan bobot yang lebih besar pada data yang sebelumnya mengalami kesalahan prediksi. Dengan demikian, model berikutnya akan memberikan perhatian lebih terhadap data yang sulit diprediksi. Hasil prediksi dari seluruh model kemudian digabungkan untuk menghasilkan prediksi akhir. AdaBoost memiliki mekanisme yang relatif sederhana, tetapi dapat sensitif terhadap data yang mengandung *noise* atau *outlier* [40].

b. *Categorical Boosting* (CatBoost)

CatBoost (*Categorical Boosting*) merupakan algoritma *boosting* yang dirancang untuk menangani data kategorikal secara efektif. CatBoost menggunakan mekanisme *ordered boosting* untuk membantu mengurangi risiko *overfitting* serta dapat mengolah variabel kategorikal tanpa memerlukan proses *encoding* secara manual. Karakteristik tersebut menjadikan CatBoost sesuai digunakan pada dataset yang memiliki banyak variabel kategorikal. Selain itu, CatBoost juga dapat digunakan untuk menyelesaikan permasalahan regresi maupun klasifikasi [41].

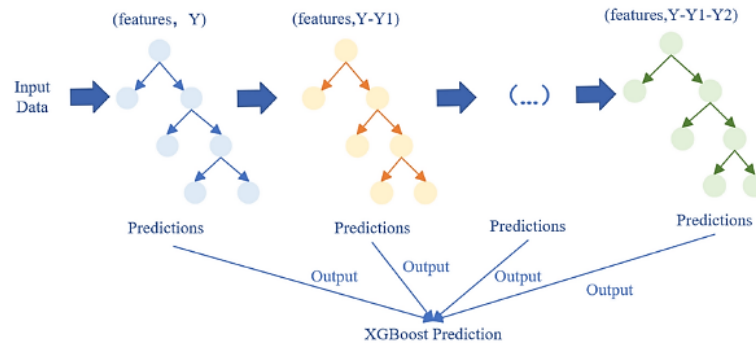
c. *Light Gradient Boosting Machine* (LightGBM)

*Light Gradient Boosting Machine* (LightGBM) merupakan algoritma *gradient boosting* yang dirancang untuk menghasilkan proses pelatihan yang cepat dan efisien. LightGBM menggunakan strategi pertumbuhan pohon secara *leaf-wise*, yaitu memilih daun dengan potensi penurunan kesalahan terbesar untuk dikembangkan terlebih dahulu. Strategi tersebut memungkinkan LightGBM menghasilkan proses pembelajaran yang efisien, terutama pada dataset berukuran besar dan memiliki jumlah fitur yang banyak. Namun, pada data yang relatif kecil, strategi *leaf-wise* dapat meningkatkan risiko *overfitting* apabila parameter model tidak diatur dengan baik [42].

d. *Extreme Gradient Boosting* (XGBoost)

*Extreme Gradient Boosting* (XGBoost) adalah algoritma *machine learning* berbasis ensemble yang dikembangkan oleh Tianqi Chen dan Carlos Guestrin pada tahun 2016 [11][43]. XGBoost bekerja dengan

membangun model prediksi secara bertahap dan berurutan (*sequential*), yang setiap pohon keputusan (*decision tree*) baru dibangun untuk memperbaiki kesalahan yang dibuat oleh pohon-pohon sebelumnya [20][44]. Proses ini dilakukan secara bertahap hingga diperoleh model dengan tingkat kesalahan yang semakin kecil.



Gambar 2 Algoritma XGBoost

Berdasarkan Gambar 2, proses pembelajaran XGBoost dilakukan secara bertahap melalui pembentukan sejumlah *decision tree*. Model awal menghasilkan prediksi, kemudian kesalahan prediksi digunakan sebagai dasar untuk membangun *decision tree* berikutnya. Setiap *decision Tree* baru berfungsi untuk memperbaiki kesalahan dari model sebelumnya. Hasil prediksi dari seluruh *decision tree* kemudian digabungkan untuk menghasilkan prediksi akhir [18].

XGBoost menerapkan fitur regularisasi L1 (*Lasso*) dan L2 (*Ridge*) yang berfungsi untuk mengurangi risiko *overfitting*, sehingga model menjadi lebih stabil ketika digunakan pada data baru [11][19]. Selain itu, XGBoost mendukung komputasi paralel sehingga proses pelatihan dapat dilakukan untuk bekerja lebih cepat dan efisien [45][46]. XGBoost juga memiliki kemampuan yang baik dalam mengolah data numerik berbentuk tabel (*tabular data*) dan dapat digunakan untuk menyelesaikan permasalahan regresi maupun klasifikasi. Keunggulan tersebut menjadikan XGBoost banyak digunakan dalam berbagai studi kasus prediksi, salah satunya pada bidang pertanian [39][47].

Peneliti memilih XGBoost karena data yang digunakan berupa data numerik berbentuk tabel (*tabular data*) dengan jumlah data yang relatif terbatas serta memiliki hubungan yang kompleks dan nonlinier antara variabel iklim dan produksi padi. Dibandingkan AdaBoost, XGBoost memiliki mekanisme optimasi dan regularisasi yang lebih lengkap. Sementara itu, LightGBM memiliki keunggulan dalam efisiensi proses pelatihan, terutama pada dataset berukuran besar, sedangkan CatBoost memiliki kemampuan yang baik dalam menangani data kategorikal. Karakteristik XGBoost yang mampu

mengolah data numerik, memodelkan hubungan nonlinier, serta menerapkan regularisasi dinilai sesuai dengan karakteristik data dan tujuan penelitian. Oleh karena itu, XGBoost digunakan sebagai algoritma *boosting* untuk memprediksi produksi padi.

Secara sistematis, persamaan prediksi XGBoost adalah sebagai berikut:

$$\hat{y}_i = \sum_{k=1}^K f_k(x_i)$$

Keterangan:

$\hat{y}_i$  = nilai prediksi

$x_i$  = data masukan

$f_k$  = *decision tree* ke- $k$

$k$  = jumlah *decision Tree*

Berdasarkan persamaan di atas, setiap *decision Tree* ditambahkan secara bertahap untuk memperbaiki kesalahan prediksi dari *tree* sebelumnya.

Dalam proses pelatihan, XGBoost menggunakan fungsi objektif yang terdiri atas fungsi kesalahan (*loss function*) dan fungsi regularisasi. Persamaannya adalah sebagai berikut:

$$\text{Obj} = \sum_{i=1}^n l(y_i, \hat{y}_i) + \sum_{k=1}^K \Omega(f_k)$$

Keterangan:

Obj = fungsi objektif

$l(y_i, \hat{y}_i)$  = fungsi kesalahan

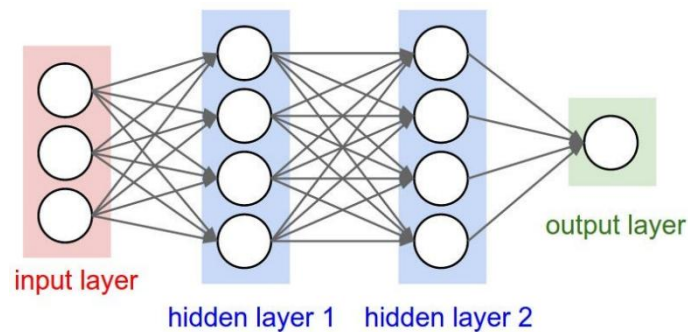
$\Omega(f_k)$  = fungsi regularisasi

$n$  = jumlah data

$k$  = jumlah *decision tree*

Berdasarkan persamaan tersebut, XGBoost tidak hanya berfokus pada minimisasi kesalahan prediksi, tetapi juga mengurangi risiko *overfitting*.

## 5. Artificial Neural Network (ANN)



Gambar 3 Algoritma ANN

*Artificial Neural Network* (ANN) atau Jaringan Syaraf Tiruan merupakan model komputasi yang terinspirasi dari struktur dan fungsi jaringan saraf biologis pada otak manusia [24]. Konsep dasar ANN pertama kali diperkenalkan oleh McCulloch dan Pitts (1943) melalui model *neuron* formal, dan kemudian berkembang pesat dengan diperkenalkannya algoritma *backpropagation* yang memungkinkan pelatihan jaringan berlapis. ANN terdiri dari unit-unit pemrosesan yang disebut *neuron*, yang terdiri dari lapisan input yang menerima data masukan, satu atau lebih lapisan tersembunyi (*hidden layer*) yang melakukan transformasi non-linear terhadap data, dan lapisan output yang menghasilkan prediksi akhir [21]. Proses pembelajaran ANN dilakukan melalui algoritma *backpropagation* yang memperbarui bobot-bobot koneksi antar *neuron* secara berulang untuk meminimalkan *error*, hingga mencapai model dengan tingkat kesalahan yang lebih kecil [22].

Dalam proses pembelajaran, ANN menyesuaikan bobot koneksi antar neuron berdasarkan nilai kesalahan (*error*) yang dihasilkan selama pelatihan [26][48]. Struktur berlapis memungkinkan model menangkap pola tersembunyi dalam data sehingga memungkinkan model untuk membuat prediksi akurat [49]. Selain itu, arsitektur ANN bersifat fleksibel karena dapat disesuaikan berdasarkan jumlah *neuron*, jumlah *hidden layer*, dan parameter pelatihan, sehingga dapat dioptimalkan sesuai karakteristik data yang digunakan [50].

Secara matematis, proses pada ANN dimulai dengan perhitungan nilai masukan pada setiap *neuron*. Nilai tersebut diperoleh dari penjumlahan hasil perkalian antara data masukan dan bobot, kemudian ditambahkan dengan nilai bias. Persamaan perhitungan *neuron* ditunjukkan sebagai berikut:

$$z = \sum_{i=1}^n w_i x_i + b$$

Keterangan:

$z$  = nilai masukan *neuron*

$x_i$  = nilai input

$w_i$  = bobot

$b$  = bias

Nilai  $z$  kemudian diproses menggunakan fungsi aktivasi untuk menghasilkan keluaran *neuron*. Pada penelitian ini, fungsi aktivasi yang digunakan adalah *Rectified Linear Unit* (ReLU), yang dirumuskan sebagai berikut:

$$f(z) = \max(0, z)$$

Keterangan:

$f(z)$  = keluaran *neuron*

$z$  = nilai masukan *neuron*

Selanjutnya, proses *backpropagation* digunakan untuk memperbarui bobot berdasarkan nilai kesalahan yang dihasilkan selama pelatihan. Persamaan pembaruan bobot dapat dituliskan sebagai berikut:

$$w_{\text{baru}} = w_{\text{lama}} - \eta \frac{\partial E}{\partial w}$$

Keterangan:

$w_{\text{baru}}$  = bobot setelah diperbarui

$w_{\text{lama}}$  = bobot sebelumnya

$\eta$  = *learning rate*

$E$  = nilai kesalahan

## 6. Bahasa Pemrograman Python

Python merupakan bahasa pemrograman yang diciptakan oleh Guido van Rossum pada tahun 1991 [14]. Python merupakan bahasa pemrograman tingkat tinggi (*high-level programming language*) yang bersifat interpretatif, mudah dipelajari, dan memiliki sintaks yang jelas serta ringkas. Python bersifat terbuka (*open-source*) sehingga dapat digunakan dan dikembangkan secara bebas oleh berbagai kalangan. Python banyak digunakan dalam pengembangan aplikasi, analisis data, serta pemodelan *machine learning* [51].

Dalam penelitian ini, Python digunakan sebagai bahasa pemrograman utama dalam proses pengolahan data variabel iklim, pembangunan model XGBoost dan ANN, serta evaluasi performa model menggunakan metrik seperti  $R^2$ , MAE, dan RMSE. Selain itu, Python juga berperan dalam proses integrasi model ke dalam sistem berbasis web melalui *framework* tertentu, sehingga hasil prediksi dapat diakses oleh pengguna secara lebih mudah [47].

## 7. Parameter Evaluasi ( $R^2$ , MAE, RMSE)

Parameter perbandingan berdasarkan penelitian terdahulu dapat dilihat pada Tabel 2 berikut [16][26][44].

Tabel 2 Parameter Perbandingan

| No  | Parameter                        | Keterangan                                                                     | Output          |
|-----|----------------------------------|--------------------------------------------------------------------------------|-----------------|
| (1) | (2)                              | (3)                                                                            | (4)             |
| 1.  | <i>R-Squared</i> ( $R^2$ )       | Mengukur seberapa besar variasi data produksi padi dapat dijelaskan oleh model | Skor (0-1)      |
| 2.  | <i>Mean Absolute Error</i> (MAE) | Mengukur rata-rata selisih absolut antara nilai prediksi dan nilai aktual      | Rata-rata (ton) |

| No  | Parameter                             | Keterangan                                              | Output          |
|-----|---------------------------------------|---------------------------------------------------------|-----------------|
| (1) | (2)                                   | (3)                                                     | (4)             |
| 3.  | <i>Root Mean Squared Error</i> (RMSE) | Mengukur akar dari rata-rata kuadrat kesalahan prediksi | Rata-rata (ton) |

Metrik evaluasi yang telah ditentukan pada tabel tersebut menjadi dasar dalam pengukuran kinerja model regresi. Untuk memberikan gambaran yang lebih jelas, berikut penjelasan dari masing-masing parameter.

a. *R-Squared* ( $R^2$ )

*R-Squared* atau koefisien determinasi merupakan metrik yang digunakan untuk mengukur seberapa besar variasi pada data aktual dapat dijelaskan oleh model prediksi. Nilai  $R^2$  berada pada rentang 0 sampai 1. Semakin mendekati 1, maka semakin besar kemampuan model dalam menjelaskan hubungan antara variabel input (variabel iklim) dan variabel output (produksi padi). Sebaliknya, nilai yang mendekati 0 menunjukkan bahwa model kurang mampu menjelaskan variasi data. Secara matematis,  $R^2$  dapat dirumuskan sebagai berikut:

$$R^2 = 1 - \frac{\sum (y_i - \hat{y}_i)^2}{\sum (y_i - \bar{y})^2}$$

Keterangan:

$y_i$  = nilai aktual

$\hat{y}_i$  = nilai prediksi

$\bar{y}$  = rata-rata nilai aktual

b. *Mean Absolute Error* (MAE)

*Mean Absolute Error* (MAE) digunakan untuk mengukur rata-rata selisih absolut antara nilai prediksi dan nilai aktual. MAE memberikan gambaran besarnya kesalahan prediksi secara umum tanpa memperhatikan arah kesalahan (positif atau negatif). Nilai MAE yang lebih kecil menunjukkan bahwa hasil prediksi model semakin mendekati nilai sebenarnya. Rumus MAE adalah sebagai berikut:

$$MAE = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i|$$

Keterangan:

$n$  = jumlah data

$y_i$  = nilai aktual

$\hat{y}_i$  = nilai prediksi

c. *Root Mean Squared Error (RMSE)*

*Root Mean Squared Error (RMSE)* merupakan metrik yang mengukur akar dari rata-rata kuadrat selisih antara nilai prediksi dan nilai aktual. RMSE memberikan penalti yang lebih besar terhadap kesalahan yang bernilai besar karena proses pengkuadratan. Nilai RMSE yang lebih kecil menunjukkan performa model yang lebih baik. Rumus RMSE adalah sebagai berikut:

$$\text{RMSE} = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2}$$

Keterangan:

$n$  = jumlah data

$y_i$  = nilai aktual

$\hat{y}_i$  = nilai prediksi

8. *Random Split*

*Random Split* merupakan metode pembagian data yang dilakukan secara acak menjadi data latih (*training*) dan data uji (*testing*). Data *training* digunakan untuk membentuk model, sedangkan data *testing* digunakan untuk menguji model. Tujuan dari pembagian data ini agar model dapat dilatih menggunakan sebagian data dan kemudian dievaluasi menggunakan data yang berbeda sehingga performa model dapat diukur secara lebih objektif. Dengan metode ini, setiap data memiliki peluang yang sama untuk terpilih sebagai data latih maupun data uji, sehingga dapat mengurangi potensi bias dalam proses pembentukan model. Oleh karena itu, *random split* banyak digunakan dalam penelitian *machine learning* untuk menguji kemampuan model dalam mengenali pola data dan menghasilkan prediksi yang baik pada data yang belum pernah dipelajari sebelumnya [18][19][51].

Pada penelitian ini, data dibagi menggunakan 80:20, yaitu 80% data digunakan sebagai data latih dan 20% data digunakan sebagai data uji. Rasio tersebut dipilih karena memberikan proporsi data yang lebih besar untuk proses pelatihan, sekaligus menyediakan data yang cukup untuk pengujian. Dengan demikian, rasio 80:20 diharapkan dapat memberikan keseimbangan antara jumlah data pelatihan dan pengujian serta mendukung proses pembentukan model yang lebih stabil pada algoritma XGBoost dan ANN [20].

9. *Metode Quantile*

*Quantile* merupakan metode pengelompokan data berdasarkan distribusi nilai pada suatu dataset. Pembagian dilakukan berdasarkan nilai persentil tertentu sehingga batas antar kelompok ditentukan oleh sebaran data yang ada, bukan oleh nilai yang ditetapkan secara manual. Penggunaan metode *quantile*

memungkinkan setiap kategori memiliki proporsi data yang lebih seimbang dan representatif terhadap karakteristik dataset. Dalam penelitian ini, metode *quantile* digunakan untuk mengelompokkan data produksi padi ke dalam tiga kategori, yaitu rendah, sedang, dan tinggi berdasarkan distribusi data historis produksi padi. Pengelompokan tersebut bertujuan untuk memudahkan pengguna dalam memahami hasil prediksi produksi padi yang ditampilkan pada sistem serta sebagai dasar dalam pemberian rekomendasi yang sesuai dengan kategori produksi yang dihasilkan [52][53].

#### 10. Sistem Berbasis Web

Sistem berbasis web merupakan aplikasi yang dapat diakses melalui jaringan internet atau intranet menggunakan peramban (browser) tanpa memerlukan instalasi khusus pada perangkat pengguna. Arsitektur sistem berbasis web mengikuti model *client-server*. *Client* (browser pengguna) berperan dalam mengirimkan permintaan (*request*) ke server melalui protokol HTTP/HTTPS, sedangkan server bertugas memproses permintaan tersebut dan mengirimkan respon kembali kepada *client*. Keunggulan sistem berbasis web terletak pada kemudahan akses, fleksibilitas penggunaan, serta kemampuannya untuk diakses dari berbagai perangkat seperti komputer maupun *smartphone* [27].

Dalam pengembangan sistem berbasis web, diperlukan teknologi *backend* untuk mengatur proses pengolahan data dan komunikasi antara pengguna dengan server. Teknologi *backend* tersebut dapat diimplementasikan menggunakan *framework* berbasis Python yang mendukung integrasi dengan model *machine learning* sehingga sistem prediksi dapat berjalan dengan baik.

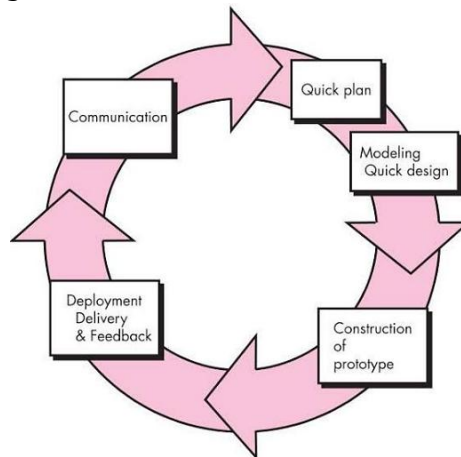
#### 11. *Framework* FastAPI

FastAPI merupakan *framework backend* berbasis Python yang digunakan untuk membangun *Application Programming Interface* (API) secara cepat dan efisien. *Framework* ini banyak digunakan dalam pengembangan sistem berbasis *machine learning* karena mampu menangani proses pertukaran data antara model prediksi dan antarmuka web dengan baik. Selain itu, FastAPI mampu melakukan validasi data secara otomatis sehingga dapat membantu menjaga konsistensi dan integritas data sebelum diproses oleh model prediksi [54].

Pada penelitian ini, FastAPI digunakan sebagai penghubung antara sistem berbasis web dengan model prediksi *machine learning*. Ketika pengguna memasukkan data variabel iklim melalui antarmuka web, sistem akan mengirimkan permintaan (*request*) ke server menggunakan format JSON. Selanjutnya, FastAPI menerima data tersebut dan meneruskannya ke proses prediksi yang telah dibangun menggunakan Python. Setelah model menghasilkan nilai prediksi produksi padi, FastAPI mengirimkan hasil tersebut kembali ke antarmuka web dalam bentuk respons sehingga dapat ditampilkan

kepada pengguna. Dengan mekanisme tersebut, proses integrasi antara sistem web dan model *machine learning* dapat berjalan secara terstruktur dan efisien.

## 12. Metode *Prototyping*



Gambar 4 Metode *Prototyping*

Metode pengembangan sistem *prototyping* merupakan salah satu model pengembangan perangkat lunak yang berfokus pada pembuatan versi awal (*prototype*) dari sistem untuk menggambarkan kebutuhan dan fungsionalitas yang diinginkan pengguna. *Prototype* yang dibuat bersifat sementara dan digunakan sebagai media komunikasi antara pengembang dan pengguna agar kebutuhan sistem dapat dipahami secara lebih jelas sebelum sistem akhir dikembangkan. Melalui pendekatan ini, pengguna dapat memberikan umpan balik secara langsung terhadap tampilan maupun fungsi sistem sehingga risiko kesalahan dalam perancangan dapat diminimalkan [55].

Tahapan dalam metode *prototyping* umumnya meliputi proses komunikasi untuk mengetahui kebutuhan, perancangan awal (*quick design*), pembuatan *prototype*, evaluasi oleh pengguna, perbaikan berdasarkan umpan balik, dan pengembangan sistem akhir. Metode ini sesuai diterapkan pada sistem yang memerlukan keterlibatan aktif pengguna, seperti sistem berbasis web, karena memungkinkan proses pengembangan dilakukan secara bertahap melalui evaluasi dan penyempurnaan hingga sistem memenuhi kebutuhan yang diharapkan [56].

## 13. *Black Box Testing*

*Black box testing* merupakan metode pengujian perangkat lunak yang berfokus pada pengujian fungsi sistem tanpa melihat atau menganalisis struktur kode program di dalamnya. Pengujian ini dilakukan dengan cara memberikan inputan ke dalam sistem, kemudian mengamati apakah output yang dihasilkan telah sesuai dengan kebutuhan yang telah dirancang sebelumnya. Pada pendekatan ini, penguji tidak perlu mengetahui proses internal sistem,

melainkan hanya memastikan bahwa sistem memberikan respons yang benar berdasarkan input yang diberikan [24][57].

Pada penelitian ini, metode *black box testing* digunakan untuk memastikan bahwa seluruh proses pada sistem prediksi produksi padi berbasis web berjalan dengan benar sesuai alur yang telah dirancang. Pengujian dilakukan dengan memberikan berbagai variasi input parameter seperti tahun, bulan tanam, serta variabel iklim, kemudian diamati apakah sistem mampu memproses data tersebut tanpa *error* dan menghasilkan output prediksi yang sesuai dengan logika model yang telah diintegrasikan. Melalui pengujian ini, dapat dipastikan bahwa fungsi sistem bekerja sebagaimana mestinya dan tidak terjadi kesalahan pada proses maupun hasil yang ditampilkan kepada pengguna.

#### 14. Kolmogorov–Smirnov

Uji *Kolmogorov–Smirnov* merupakan salah satu metode statistik yang digunakan untuk menguji apakah suatu data mengikuti distribusi normal. Pengujian dilakukan dengan membandingkan distribusi data yang diamati dengan distribusi normal. Hasil pengujian dapat dilihat berdasarkan nilai signifikansi (Sig.). Apabila nilai signifikansi lebih besar dari 0,05, data dinyatakan berdistribusi normal. Sebaliknya, apabila nilai signifikansi kurang dari atau sama dengan 0,05, data dinyatakan tidak berdistribusi normal [58]. Dalam penelitian ini, uji *Kolmogorov–Smirnov* digunakan untuk menguji normalitas data selisih nilai *Absolute Error* (AE) antara model XGBoost dan ANN sebelum dilakukan pengujian statistik komparatif.

#### 15. Wilcoxon Signed-Rank Test

*Wilcoxon Signed-Rank Test* merupakan uji statistik nonparametrik yang digunakan untuk membandingkan dua kelompok data yang saling berpasangan apabila data selisih tidak berdistribusi normal. Pengujian dilakukan berdasarkan perbedaan nilai pada setiap pasangan data untuk mengetahui apakah terdapat perbedaan yang signifikan antara kedua kelompok. Dalam penelitian ini, *Wilcoxon Signed-Rank Test* digunakan untuk membandingkan nilai *Absolute Error* (AE) model XGBoost dan ANN pada periode pengamatan yang sama. Pengambilan keputusan dilakukan berdasarkan nilai signifikansi (*Asymp. Sig. (2-tailed)*). Apabila nilai signifikansi kurang dari 0,05, maka terdapat perbedaan yang signifikan antara nilai AE kedua model. Sebaliknya, apabila nilai signifikansi lebih besar dari atau sama dengan 0,05, maka tidak terdapat perbedaan yang signifikan [59].

## BAB III METODOLOGI

### A. Waktu dan Tempat Penelitian

#### 1. Waktu Penelitian

Penelitian ini dilakukan pada bulan Februari 2026 sampai dengan Juni 2026.

#### 2. Tempat Penelitian

Penelitian ini dilakukan di Kabupaten Cilacap sebagai wilayah studi dalam analisis prediksi produksi padi berdasarkan variabel iklim.

### B. Alat dan Bahan

Pada penelitian ini, digunakan perangkat keras (*hardware*) dan perangkat lunak (*software*) untuk mendukung proses pengolahan data, pembangunan model *machine learning*, serta implementasi sistem berbasis web. Adapun perangkat yang digunakan dalam penelitian ini yaitu:

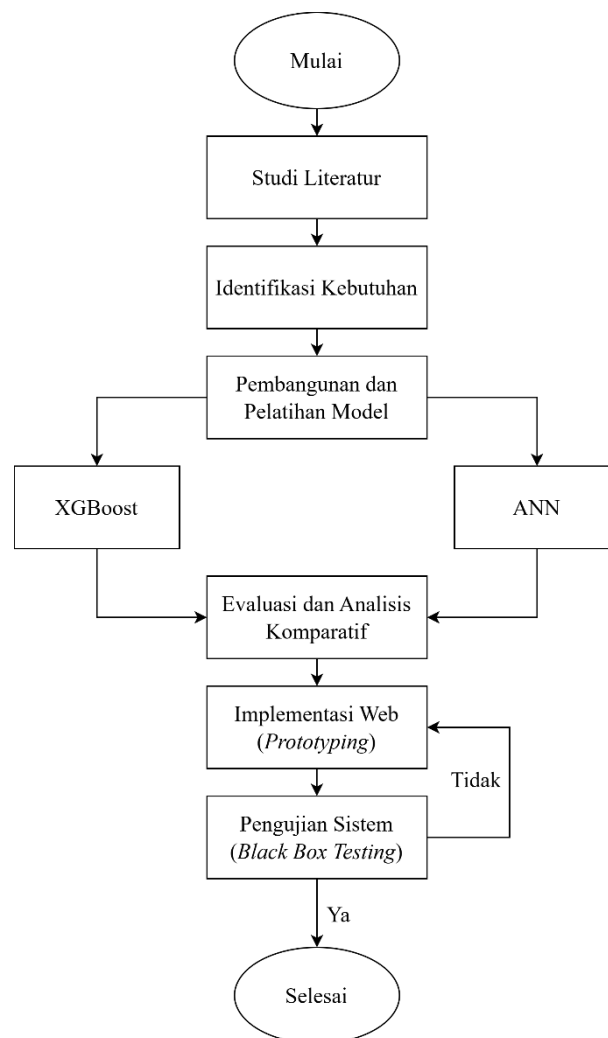
Tabel 3 Alat dan Bahan

| No  | Alat/Bahan                        | Keterangan                                                                                                                                                                                                                                               |
|-----|-----------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| (1) | (2)                               | (3)                                                                                                                                                                                                                                                      |
| 1   | Laptop                            | Digunakan untuk proses pengolahan data, pembangunan model <i>machine learning</i> , implementasi sistem berbasis web, dan pengujian sistem dengan spesifikasi minimum prosesor setara Intel® Core i5, RAM 8 GB, serta sistem operasi Windows 10 ke atas. |
| 2   | <i>Visual Studio Code</i> 1.130.0 | Digunakan sebagai text editor untuk penulisan kode program.                                                                                                                                                                                              |
| 3   | Microsoft Excel 2016              | Digunakan untuk penyimpanan, pengelolaan, dan pengecekan data.                                                                                                                                                                                           |
| 4   | Python 3.12                       | Digunakan sebagai bahasa pemrograman untuk proses pengolahan data, pembangunan model XGBoost dan ANN, visualisasi data, serta implementasi <i>backend</i> sistem prediksi.                                                                               |
| 5   | <i>Pandas</i> 2.2.2               | <i>Library</i> untuk membaca, mengelola, dan memproses data dari file <i>Microsoft Excel</i> .                                                                                                                                                           |
| 6   | <i>Numpy</i> 2.1.1                | <i>Library</i> untuk melakukan operasi numerik pada data.                                                                                                                                                                                                |

| No  | Alat/Bahan                               | Keterangan                                                                                                                                                           |
|-----|------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| (1) | (2)                                      | (3)                                                                                                                                                                  |
| 7   | <i>Matplotlib</i> 3.9.2                  | <i>Library</i> untuk membuat visualisasi data dan grafik hasil prediksi.                                                                                             |
| 8   | <i>Xgboost (XGBRegressor)</i> 3.2.0      | <i>Library</i> untuk membangun dan melatih model prediksi menggunakan algoritma XGBoost.                                                                             |
| 9   | <i>Scikit-learn (MLPRegressor)</i> 1.5.2 | <i>Library</i> yang digunakan untuk pembagian data, normalisasi data, pembangunan model ANN, serta evaluasi performa model menggunakan metrik $R^2$ , MAE, dan RMSE. |
| 10  | OS 3.12.6                                | <i>Library</i> untuk mengelola folder dan penyimpanan file hasil visualisasi.                                                                                        |
| 11  | FastAPI 0.136.1                          | Digunakan sebagai <i>framework backend</i> untuk menghubungkan sistem berbasis web dengan model prediksi <i>machine learning</i> .                                   |
| 12  | Browser                                  | Digunakan untuk menjalankan dan menguji sistem berbasis web yang dikembangkan.                                                                                       |
| 13  | <i>Supabase</i> 2.30.1                   | Platform database berbasis PostgreSQL yang digunakan untuk menyimpan dan mengelola data pada sistem berbasis web.                                                    |

### C. Prosedur Penelitian

Penelitian ini dilakukan melalui beberapa tahap untuk memastikan hasil akurat dan sesuai dengan tujuan penelitian. Adapun tahapan yang dilakukan ditunjukkan pada Gambar 4.



Gambar 5 Alur Penelitian

## 1. Studi Literatur

Untuk mengidentifikasi suatu masalah, diperlukan studi literatur yang berhubungan dengan topik penelitian. Literatur menjadi sumber/landasan mengenai teori, konsep, dan metode yang dijadikan dasar dalam proses pengembangan dan pelaksanaan penelitian.

Pada tahap ini dilakukan analisis terhadap berbagai sumber ilmiah yang berfokus pada topik produksi padi, faktor iklim, metode *machine learning* khususnya *Extreme Gradient Boosting* (XGBoost) dan *Artificial Neural Network* (ANN), serta pengembangan sistem berbasis web menggunakan bahasa pemrograman Python. Studi literatur ini juga mencakup pembahasan mengenai teknik evaluasi model regresi seperti  $R^2$ , MAE, dan RMSE yang umum digunakan dalam pengukuran performa model prediksi.

Berdasarkan studi yang dilakukan, algoritma XGBoost dan ANN telah banyak dimanfaatkan untuk berbagai kebutuhan prediksi di sejumlah bidang. Namun, penerapannya umumnya berfokus pada satu metode saja dan lebih

banyak dilakukan pada sektor kesehatan, pendidikan, maupun klasifikasi citra. Penelitian yang membandingkan kedua metode tersebut secara langsung dalam konteks prediksi produksi pertanian berbasis variabel iklim masih relatif terbatas. Selain itu, kajian yang mengintegrasikan hasil perbandingan model ke dalam sistem yang dapat digunakan secara langsung juga belum banyak dilakukan.

Oleh karena itu, penelitian ini diarahkan pada dua fokus utama, yaitu melakukan analisis komparatif performa XGBoost dan ANN dalam memprediksi produksi padi berdasarkan variabel iklim, serta mengimplementasikan metode dengan performa terbaik ke dalam sistem prediksi berbasis web. Pendekatan ini diharapkan tidak hanya menghasilkan evaluasi model secara akademis, tetapi juga memberikan manfaat praktis dalam mendukung perencanaan produksi dan pengambilan keputusan berbasis data.

## 2. Identifikasi Kebutuhan

Tahap identifikasi kebutuhan menjelaskan apa saja yang diperlukan dalam penelitian ini. Tahap ini meliputi pengumpulan data, pra-pemrosesan data, serta kebutuhan perangkat yang digunakan.

### a. Pengumpulan Data

Data dalam penelitian ini terdiri dari data iklim dan data produksi padi. Data iklim diambil dari Badan Meteorologi, Klimatologi, dan Geofisika (BMKG) Cilacap, yang meliputi variabel lama penyinaran matahari, curah hujan, suhu udara, dan kelembaban udara. Sedangkan Indeks ENSO diambil dari web resmi <https://ggweather.com/enso/oni.htm>. Data produksi padi diperoleh dari Dinas Pertanian Kabupaten Cilacap, berupa data jumlah produksi padi dalam satuan ton. Data yang digunakan merupakan data bulanan selama 5 tahun, yaitu periode 2020-2024. Data tersebut berdasarkan ketersediaan resmi dari instansi terkait.

### b. Pra-pemrosesan Data

Pada tahap pra-pemrosesan data, seluruh data yang telah dikumpulkan akan disesuaikan sebelum digunakan dalam proses pemodelan. Proses ini meliputi pengecekan serta penanganan data yang hilang dan penyelarasan periode waktu antara data iklim dan data produksi padi. Tahap ini bertujuan untuk memastikan data dalam kondisi optimal sebelum dilakukan pembangunan dan perbandingan model.

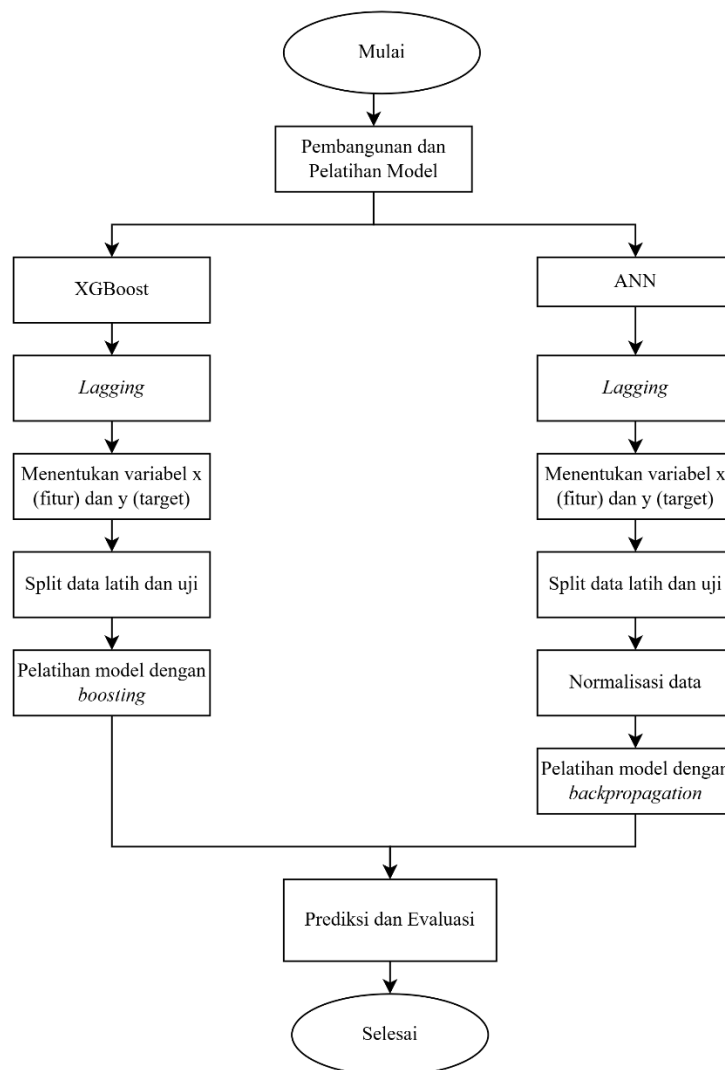
### c. Kebutuhan Perangkat

Perangkat keras yang digunakan dalam penelitian ini berupa satu unit laptop dengan *device* HP. Adapun spesifikasi minimum yang direkomendasikan setara dengan Intel® Core i5-8350U, RAM 8,00 GB, serta sistem operasi Windows 11 Pro ke atas. Spesifikasi minimum tersebut

dinilai cukup untuk melakukan proses pengolahan data, pembangunan model *machine learning*, serta implementasi dan pengujian sistem berbasis web. Dengan terpenuhinya spesifikasi tersebut, sistem yang di kembangkan dalam penelitian ini dapat dijalankan dengan baik.

Perangkat lunak yang digunakan meliputi bahasa pemrograman Python sebagai alat utama dalam pengolahan data dan pembangunan model prediksi, Visual Studio Code sebagai media penulisan dan pengembangan kode program, Microsoft Excel untuk pengelolaan dan penyusunan data awal, serta web browser untuk menjalankan sistem berbasis web yang dikembangkan.

### 3. Pembangunan dan Pelatihan Model



Gambar 6 Pembangunan dan Pelatihan Model

Pada tahap ini dilakukan proses pembangunan model prediksi menggunakan dua algoritma, yaitu *Extreme Gradient Boosting* (XGBoost) dan *Artificial Neural Network* (ANN). Data yang telah melalui tahap pra-pemrosesan kemudian dibagi menjadi data latih (*training*) dan data uji (*testing*) untuk memastikan bahwa model dapat dievaluasi secara objektif. Proses pelatihan dilakukan menggunakan data latih dengan tujuan agar model dapat mempelajari pola hubungan antara variabel iklim terhadap produksi padi.

Seluruh proses pembangunan dan pelatihan model pada penelitian ini diimplementasikan menggunakan bahasa pemrograman Python. Python digunakan untuk membaca dan mengolah data produksi padi serta variabel iklim dengan memanfaatkan *library* seperti *pandas* dan *numpy*. Pada tahap pemodelan, diterapkan teknik *lagging* hingga tiga periode (*lag-1*, *lag-2*, dan *lag-3*) untuk merepresentasikan pengaruh kondisi tiga bulan sebelumnya terhadap hasil produksi padi, sesuai dengan siklus tanam hingga panen [60]. Data yang telah diproses kemudian digunakan sebagai fitur (*input*) dan target (*output*) dalam proses pelatihan model.

Selanjutnya, dilakukan pembangunan dan pelatihan dua model, yaitu XGBoost dan ANN. Untuk metode XGBoost, model dibangun dengan pendekatan *boosting* yang membentuk sejumlah pohon keputusan secara bertahap, yang setiap pohon berfungsi untuk memperbaiki kesalahan prediksi sebelumnya [20][44]. Sementara itu, metode ANN dilakukan terlebih dahulu proses normalisasi data menggunakan teknik *StandardScaler* untuk menyamakan skala antar variabel sehingga proses pelatihan jaringan saraf dapat berlangsung lebih optimal. Setelah normalisasi dilakukan, model dibangun menggunakan arsitektur yang terdiri dari lapisan input, satu atau lebih *hidden layer*, dan lapisan output [21]. Proses pelatihan ANN dilakukan dengan metode *backpropagation* untuk menyesuaikan bobot berdasarkan nilai kesalahan (*error*) hingga model menghasilkan prediksi yang mendekati nilai aktual [22].

Pada penelitian ini, pembangunan model difokuskan pada perbandingan performa kedua algoritma tanpa melakukan optimasi parameter. Model yang telah dilatih kemudian dievaluasi menggunakan metrik  $R^2$ , MAE, dan RMSE untuk mengukur tingkat akurasi prediksi. Selain itu, hasil prediksi divisualisasikan dalam bentuk grafik untuk memudahkan analisis perbandingan antara data aktual dan hasil prediksi. Hasil dari tahap ini berupa dua model terlatih yang selanjutnya akan dibandingkan untuk menentukan model dengan performa terbaik.

a. Model *Extreme Gradient Boosting* (XGBoost)

1) Inisialisasi *Library*

Pada proses ini dilakukan *import library* yang digunakan dalam pembangunan model prediksi. Library *pandas* digunakan untuk membaca dan mengolah dataset, *numpy* digunakan untuk operasi numerik, *matplotlib* digunakan untuk visualisasi grafik, sedangkan *xgboost* dan *scikit-learn* digunakan dalam proses pembangunan serta evaluasi model *machine learning*.

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import os

from sklearn.neural_network import MLPRegressor
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.metrics import r2_score, mean_absolute_error,
mean_squared_error
```

## 2) *Lagging* Data

Tahap berikutnya yaitu penerapan teknik *lagging* pada seluruh variabel iklim. Proses ini bertujuan untuk merepresentasikan pengaruh kondisi iklim pada periode sebelumnya terhadap hasil produksi padi. Penerapan *lagging* didasarkan pada siklus pertumbuhan tanaman padi yang umumnya berlangsung sekitar tiga bulan sejak masa tanam hingga panen.

Pada penelitian ini digunakan tiga periode *lag*, yaitu *lag-1*, *lag-2*, dan *lag-3*. *Lag-1* merepresentasikan kondisi iklim satu bulan sebelum panen yang umumnya berkaitan dengan fase pematangan dan pengisian bulir padi. *Lag-2* merepresentasikan kondisi iklim dua bulan sebelum panen yang berkaitan dengan fase pertumbuhan vegetatif hingga pembentukan malai. Sementara itu, *lag-3* merepresentasikan kondisi iklim tiga bulan sebelum panen yang berkaitan dengan fase awal pertumbuhan setelah penanaman. Dengan demikian, model dapat mempelajari pengaruh kondisi iklim selama satu siklus tanam terhadap hasil produksi padi yang diperoleh pada saat panen.

Sebagai contoh, produksi padi pada bulan April dipengaruhi oleh kondisi iklim pada bulan Januari (*lag-3*), Februari (*lag-2*), dan Maret (*lag-1*). Oleh karena itu, informasi iklim pada ketiga periode tersebut digunakan sebagai fitur masukan dalam proses pelatihan model.

```
kolom_fitur = ['LPM (%)', 'CH (mm)', 'ONI', 'Suhu (°C)',
               'Kelembaban (%)']

for col in kolom_fitur:
    df[f'{col}_Lag1'] = df[col].shift(1)
    df[f'{col}_Lag2'] = df[col].shift(2)
```

```
df[f'{col}_Lag3'] = df[col].shift(3)

df_clean = df.dropna().reset_index(drop=True)
```

### 3) Penentuan Fitur dan Target

Pada tahap ini dilakukan pemisahan antara variabel fitur (*input*) dan variabel target (*output*). Variabel fitur terdiri dari seluruh variabel hasil dari proses *lagging*, sedangkan variabel target berupa data produksi padi dalam satuan ton.

```
features = [col for col in df_clean.columns if 'Lag' in col]

X = df_clean[features]
y = df_clean['P (ton)']
```

### 4) Pembagian Data

Dataset kemudian dibagi menjadi data latih (*training*) dan data uji (*testing*) menggunakan metode *random split* dengan rasio 80:20. Data latih digunakan untuk proses pelatihan model, sedangkan data uji digunakan untuk mengevaluasi performa model.

```
X_train_raw, X_test_raw, y_train_raw, y_test_raw =
train_test_split(
    X, y,
    test_size=0.2,
    random_state=42
)
```

### 5) *Traning* Model XGBoost

Pada tahap ini dilakukan pembangunan dan pelatihan model menggunakan algoritma XGBoost. Pemilihan *hyperparameter* dilakukan dengan metode *trial and error* melalui beberapa kombinasi nilai. Kombinasi yang menghasilkan performa terbaik digunakan pada proses pelatihan model. Pada penelitian ini digunakan parameter *n\_estimators* sebesar 100 yang menunjukkan bahwa model membangun 100 pohon keputusan selama proses pelatihan. Parameter *learning\_rate* sebesar 0,5 digunakan untuk mengatur besarnya kontribusi setiap pohon terhadap model akhir. Nilai *max\_depth* sebesar 5 digunakan untuk membatasi kedalaman pohon agar model tidak terlalu kompleks dan mengurangi risiko *overfitting*. Selain itu, parameter *subsample* dan *colsample\_bytree* masing-masing bernilai 1 yang berarti seluruh data dan seluruh fitur digunakan dalam pembentukan setiap pohon keputusan.

Setelah model selesai dilatih, tahap berikutnya yaitu melakukan prediksi terhadap seluruh data yang digunakan dalam penelitian untuk mengetahui hasil estimasi produksi padi yang dihasilkan model.

```
xgb = XGBRegressor(
    n_estimators=100,
    learning_rate=0.5,
    max_depth=5,
    subsample=1,
    colsample_bytree=1,
    random_state=42
)
xgb.fit(X_train, y_train)
```

#### 6) Evaluasi Model

Evaluasi model dilakukan menggunakan metrik  $R^2$ , MAE, dan RMSE untuk mengetahui tingkat performa model dalam memprediksi.

```
r2_all = r2_score(y, y_all_pred)
mae_all = mean_absolute_error(y, y_all_pred)
rmse_all = np.sqrt(mean_squared_error(y, y_all_pred))
```

### b. Model Artificial Neural Network (ANN)

#### 1) Inisialisasi Library

Pada tahap ini dilakukan *import library* yang dibutuhkan untuk proses pembangunan model ANN. *Library pandas* dan *numpy* digunakan untuk pengolahan data, *matplotlib* digunakan untuk visualisasi hasil prediksi, *MLPRegressor* dari *scikit-learn* digunakan untuk membangun model ANN, *StandardScaler* digunakan untuk proses normalisasi data, sedangkan *train\_test\_split*, *r2\_score*, *mean\_absolute\_error*, dan *mean\_squared\_error* digunakan untuk pembagian data serta evaluasi performa model.

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import os

from sklearn.neural_network import MLPRegressor
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.metrics import r2_score, mean_absolute_error,
mean_squared_error
```

#### 2) Lagging Data

Tahap *lagging* pada model ANN dilakukan dengan prosedur yang sama seperti pada model XGBoost. Teknik ini digunakan untuk merepresentasikan pengaruh kondisi iklim pada periode sebelumnya terhadap produksi padi dengan menerapkan *lag-1*, *lag-2*, dan *lag-3* pada seluruh variabel iklim. Hasil proses lagging digunakan sebagai fitur masukan dalam pelatihan model ANN.

### 3) Penentuan Fitur dan Target

Pada tahap ini dilakukan pemisahan antara variabel fitur (*input*) dan variabel target (*output*). Tahapan ini dilakukan dengan prosedur yang sama seperti pada model XGBoost, yaitu menggunakan seluruh variabel hasil lagging sebagai fitur dan data produksi padi sebagai target prediksi.

### 4) Pembagian Data

Pada tahap ini dilakukan pembagian dataset menjadi data latih (*training*) dan data uji (*testing*) menggunakan metode *random split* dengan rasio 80:20. Tahapan ini dilakukan dengan prosedur yang sama seperti pada model XGBoost.

### 5) Normalisasi Data

Pada tahap ini dilakukan proses normalisasi data menggunakan metode *StandardScaler*. Normalisasi bertujuan untuk menyamakan skala antar variabel sehingga setiap fitur memiliki rata-rata mendekati 0 dan standar deviasi mendekati 1. Proses ini diperlukan karena model ANN sensitif terhadap perbedaan rentang nilai antar variabel, sehingga normalisasi dapat membantu mempercepat proses pembelajaran dan meningkatkan stabilitas model selama pelatihan.

Berbeda dengan algoritma XGBoost yang mampu mengolah data dengan berbagai rentang nilai tanpa memerlukan normalisasi, model ANN memerlukan data yang telah dinormalisasi agar proses penyesuaian bobot pada setiap *neuron* dapat berlangsung secara lebih optimal.

Pada penelitian ini, normalisasi dilakukan terhadap variabel fitur dan variabel target menggunakan data latih (*training*). Selanjutnya, parameter normalisasi yang diperoleh dari data latih digunakan kembali untuk mentransformasi data uji (*testing*) guna menghindari terjadinya data *leakage*. Selain itu, seluruh data fitur juga ditransformasikan menggunakan parameter yang sama untuk keperluan prediksi dan visualisasi hasil model.

```
scaler_X = StandardScaler()
scaler_y = StandardScaler()

X_train = scaler_X.fit_transform(X_train_raw)
y_train = scaler_y.fit_transform(y_train_raw.values.reshape(-1,1)).ravel()

X_test = scaler_X.transform(X_test_raw)
y_test = scaler_y.transform(y_test_raw.values.reshape(-1,1)).ravel()

X_scaled = scaler_X.transform(X)
```

#### 6) Pelatihan Model ANN

Pada tahap ini dilakukan pembangunan dan pelatihan model menggunakan algoritma ANN dengan metode *Multi-Layer Perceptron* (MLP). *Hyperparameter* dipilih menggunakan metode *trial and error* berdasarkan kombinasi nilai yang menghasilkan performa terbaik. Model dibangun menggunakan satu *hidden layer* yang terdiri dari 100 *neuron* (*hidden\_layer\_sizes* = 100) untuk mempelajari pola hubungan antara variabel iklim dan produksi padi yang bersifat kompleks. Parameter *learning\_rate\_init* sebesar 0,01 digunakan untuk mengatur besarnya langkah pembaruan bobot selama proses pelatihan, sedangkan parameter *alpha* sebesar 0,0005 digunakan sebagai regularisasi untuk mengurangi risiko *overfitting*. Selain itu, parameter *max\_iter* sebesar 100 digunakan sebagai batas maksimum iterasi pelatihan agar model memiliki kesempatan yang cukup untuk mencapai performa yang optimal selama proses pelatihan, dan *random\_state* sebesar 42 digunakan untuk memastikan hasil pelatihan yang konsisten pada setiap proses pengujian.

```
ann = MLPRegressor(  
    hidden_layer_sizes=(100),  
    alpha=0.0005,  
    learning_rate_init=0.01,  
    max_iter=100,  
    random_state=42  
)  
ann.fit(X_train, y_train)
```

#### 7) Denormalisasi Hasil Prediksi

Setelah proses pelatihan selesai, model ANN digunakan untuk menghasilkan prediksi terhadap seluruh data. Prediksi dilakukan menggunakan fungsi *predict()*, sehingga diperoleh nilai prediksi dalam bentuk data yang masih berada pada skala normalisasi. Hasil prediksi tersebut disimpan pada variabel *y\_all\_pred\_scaled*.

Selanjutnya dilakukan proses denormalisasi menggunakan fungsi *inverse\_transform()* untuk mengembalikan hasil prediksi dari skala normalisasi ke satuan ton. Proses ini dilakukan karena nilai prediksi yang dihasilkan model ANN masih berada dalam bentuk data yang telah dinormalisasi. Hasil denormalisasi kemudian digunakan pada tahap evaluasi dan visualisasi model.

```
y_all_pred_scaled = ann.predict(X_scaled)  
  
y_all_pred = scaler_y.inverse_transform(  
    y_all_pred_scaled.reshape(-1,1)  
)  
.ravel()
```

#### 8) Evaluasi Model

Evaluasi model dilakukan menggunakan metrik  $R^2$ , MAE, dan RMSE untuk mengetahui tingkat performa model dalam memprediksi produksi padi.

```
r2_all = r2_score(y, y_all_pred)
mae_all = mean_absolute_error(y, y_all_pred)
rmse_all = np.sqrt(mean_squared_error(y, y_all_pred))
```

#### 4. Evaluasi dan Analisis Komparatif

Pada tahap ini, kedua model dilakukan evaluasi untuk mengukur tingkat akurasi dan kesalahan prediksi dalam memperkirakan produksi padi berdasarkan variabel iklim. Untuk mengukur performa model secara objektif, digunakan beberapa metrik evaluasi pada kasus regresi. Metrik tersebut meliputi *R-Square* ( $R^2$ ), *Mean Absolute Error* (MAE), dan *Root Mean Squared Error* (RMSE).

Hasil evaluasi kemudian dicatat dalam bentuk tabel untuk memudahkan analisis. Tabel hasil evaluasi kedua metode dapat dilihat pada Tabel 3 berikut.

Tabel 4 Tabel Hasil Evaluasi

| No  | Metode                                     | $R^2$ | MAE (ton) | RMSE (ton) |
|-----|--------------------------------------------|-------|-----------|------------|
| (1) | (2)                                        | (3)   | (4)       | (5)        |
| 1.  | <i>Extreme Gradient Boosting</i> (XGBoost) |       |           |            |
| 2.  | <i>Artificial Neural Network</i> (ANN)     |       |           |            |

Hasil evaluasi dari kedua model kemudian dibandingkan secara komparatif untuk menentukan metode dengan performa terbaik. Perbandingan dilakukan berdasarkan nilai  $R^2$  tertinggi serta nilai MAE dan RMSE terendah. Model yang menunjukkan performa paling baik berdasarkan hasil evaluasi tersebut selanjutnya dipilih untuk diimplementasikan ke dalam sistem prediksi produksi padi berbasis web. Dengan demikian, proses komparasi tidak hanya berhenti pada tahap pengujian akademik, tetapi juga berlanjut pada tahap implementasi sebagai bentuk penerapan praktis dari hasil penelitian.

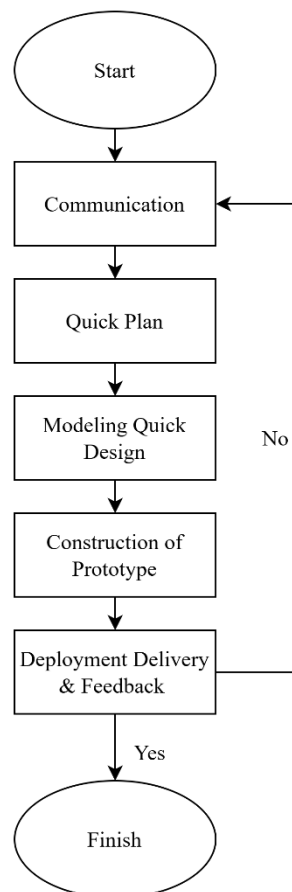
#### 5. Implementasi Web

Setelah proses pembangunan dan evaluasi model prediksi selesai dilakukan, tahap selanjutnya adalah implementasi model terpilih ke dalam sistem berbasis web menggunakan metode pengembangan *prototyping*. Dengan pendekatan ini, sistem tidak langsung dikembangkan secara penuh,

melainkan diawali dengan pembuatan *prototype* sebagai gambaran awal fungsionalitas dan tampilan sistem.

Implementasi berbasis web dipilih karena sistem dapat diakses melalui browser tanpa memerlukan instalasi khusus, sehingga lebih fleksibel dan mudah digunakan. Dengan demikian, pengguna dapat memanfaatkan sistem ini kapan saja dan di mana saja untuk mendapatkan estimasi produksi padi secara cepat dan akurat. Selain itu, sistem web memungkinkan integrasi model prediksi dengan antarmuka yang interaktif serta penyajian hasil prediksi dalam bentuk yang informatif dan mudah dipahami.

Adapun tahapan implementasi sistem menggunakan metode *prototyping* adalah sebagai berikut:



Gambar 7 Tahapan *Prototyping*

a. *Communication*

Tahap komunikasi (*communication*) merupakan tahap awal yang melibatkan interaksi antara pengembang dan pengguna untuk mengidentifikasi kebutuhan sistem secara rinci. Pada tahap ini dilakukan diskusi untuk mengetahui permasalahan yang ada, menentukan fitur utama sistem prediksi produksi padi, serta menetapkan kebutuhan fungsional yang

meliputi fitur input variabel iklim, proses prediksi, serta tampilan hasil dan rekomendasi yang informatif.

b. *Quick Plan*

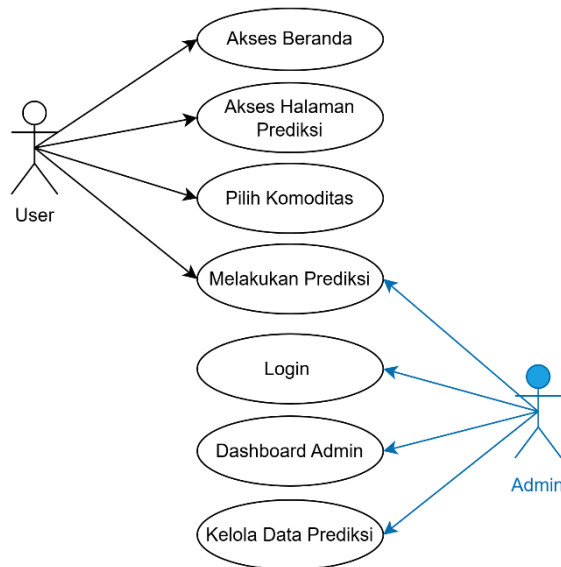
Berdasarkan hasil komunikasi, tahap selanjutnya adalah perencanaan cepat (*quick plan*). Pada tahap ini dilakukan perancangan awal alur sistem secara umum, mulai dari proses pengguna mengakses halaman web hingga sistem menampilkan hasil prediksi. Selain itu, ditentukan ruang lingkup sistem, dan kebutuhan teknologi yang digunakan, serta integrasi model *machine learning* yang telah dilatih sebelumnya.

c. *Modeling Quick Design*

Tahap *modeling quick design* dilakukan dengan membuat desain sistem secara sederhana dan cepat berdasarkan perencanaan sebelumnya. Desain ini meliputi pembuatan diagram pemodelan sistem seperti *use case diagram*, *activity diagram*, dan *sequence diagram*.

1) *Use case Diagram*

*Use case diagram* digunakan untuk menggambarkan interaksi antara aktor dengan sistem serta menunjukkan fungsi-fungsi yang dapat dijalankan oleh masing-masing aktor. Diagram ini menunjukkan hak akses serta aktivitas yang dapat dilakukan oleh masing-masing aktor dalam sistem.



Gambar 8 *Use case* Web Sistem Prediksi

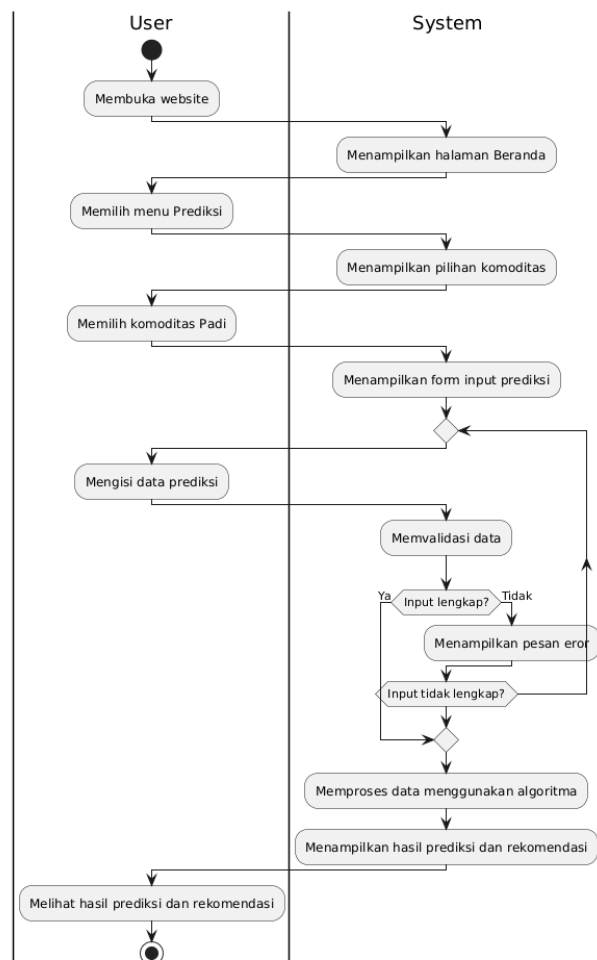
*Use case diagram* tersebut menunjukkan aktivitas utama yang dapat dilakukan oleh pengguna dan administrator pada sistem prediksi produksi padi berbasis web. Pengguna dapat mengakses halaman beranda, membuka halaman prediksi, memilih komoditas yang akan diprediksi, serta melakukan prediksi produksi berdasarkan parameter

iklim yang dimasukkan. Setelah proses prediksi dilakukan, sistem akan menampilkan hasil prediksi beserta kategori produksi dan rekomendasi yang sesuai. Sementara itu, administrator dapat melakukan login untuk mengakses *dashboard* admin, mengelola data produksi yang digunakan dalam sistem dan melakukan simulasi prediksi, serta melihat riwayat prediksi yang telah dilakukan. Dengan rancangan ini, sistem tidak hanya mendukung proses prediksi bagi pengguna, tetapi juga menyediakan fasilitas pengelolaan data bagi administrator.

## 2) Activity Diagram

Activity diagram digunakan untuk menggambarkan alur proses atau urutan aktivitas yang terjadi dalam sistem dari awal hingga akhir. Diagram ini menunjukkan bagaimana pengguna maupun administrator berinteraksi dengan sistem serta bagaimana sistem merespons setiap aktivitas yang dilakukan.

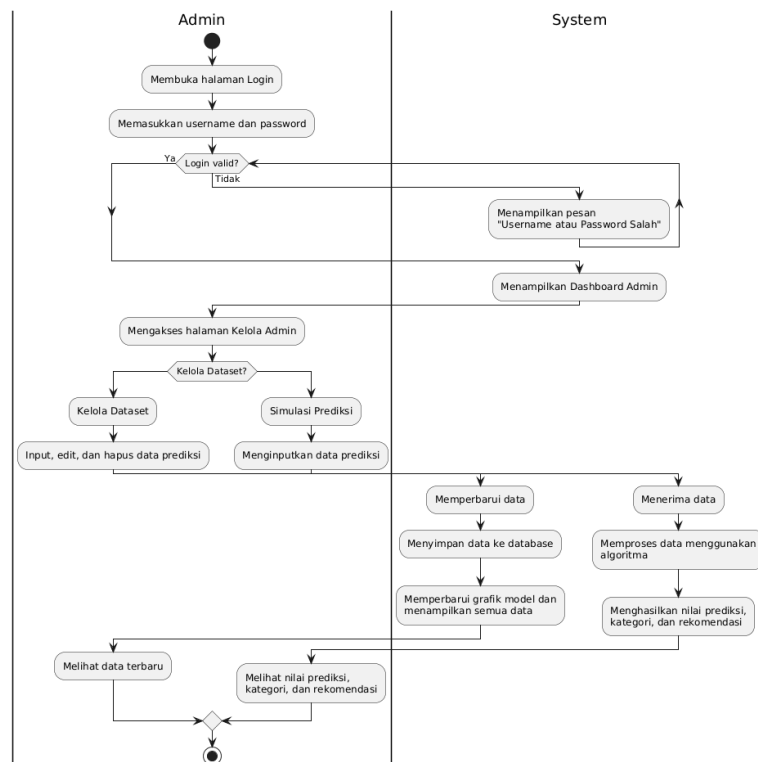
### a) Activity Diagram User



Gambar 9 Activity Diagram User

Activity diagram tersebut menunjukkan alur kerja sistem prediksi produksi padi berbasis web dari sisi pengguna. Proses dimulai saat pengguna membuka website dan sistem menampilkan halaman beranda. Pengguna kemudian mengakses menu prediksi dan memilih komoditas padi sehingga sistem menampilkan formulir input prediksi. Setelah pengguna mengisi parameter yang dibutuhkan, sistem melakukan validasi terhadap data yang dimasukkan. Jika terdapat data yang belum lengkap, sistem akan menampilkan pesan kesalahan dan pengguna diminta untuk memperbaiki input. Sebaliknya, apabila seluruh data telah valid, sistem akan melakukan proses prediksi menggunakan model yang telah diimplementasikan dan menampilkan hasil prediksi produksi padi beserta rekomendasi. Proses berakhir setelah hasil prediksi berhasil ditampilkan kepada pengguna.

#### b) Activity Diagram Admin



Gambar 10 Activity Diagram Admin

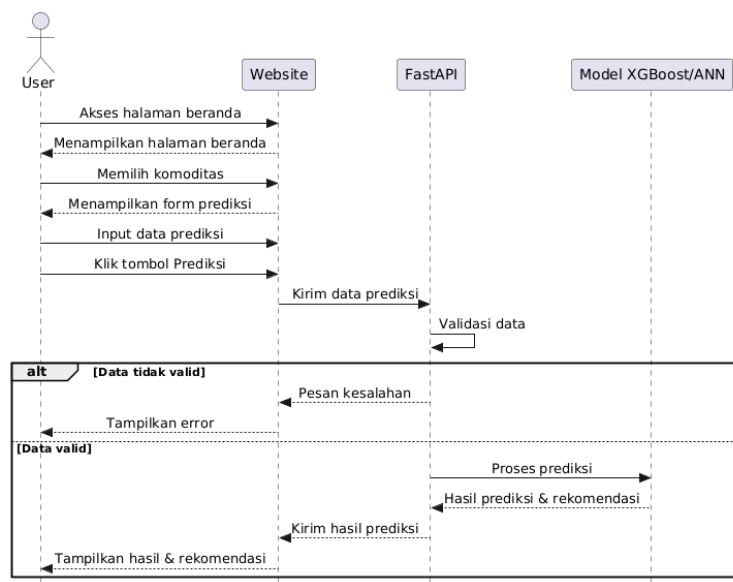
Activity diagram tersebut menggambarkan alur pengelolaan sistem yang dilakukan oleh admin pada sistem prediksi produksi padi berbasis web. Proses diawali dengan login ke sistem dan validasi kredensial oleh sistem. Setelah berhasil masuk ke *dashboard* admin dapat memilih dua aktivitas utama, yaitu

mengelola dataset atau melakukan simulasi prediksi. Pada proses pengelolaan dataset, admin dapat menambah, mengubah, dan menghapus data produksi. Sistem kemudian memperbarui data yang tersimpan pada database, serta memperbarui grafik model dan tampilan data pada sistem. Sementara itu, pada proses simulasi prediksi, admin memasukkan data prediksi yang kemudian diproses oleh sistem menggunakan model yang telah diimplementasikan untuk menghasilkan nilai prediksi, kategori produksi, dan rekomendasi yang sesuai. Setelah proses selesai, admin dapat melihat data terbaru maupun hasil simulasi prediksi yang ditampilkan oleh sistem.

### 3) Sequence Diagram

*Sequence* diagram digunakan untuk menggambarkan urutan pertukaran pesan (*message*) antara aktor dan sistem selama suatu proses berlangsung. Diagram ini menunjukkan interaksi yang terjadi secara berurutan berdasarkan waktu sehingga dapat memperjelas aliran data dan proses yang dilakukan oleh sistem.

#### a) Sequence Diagram User

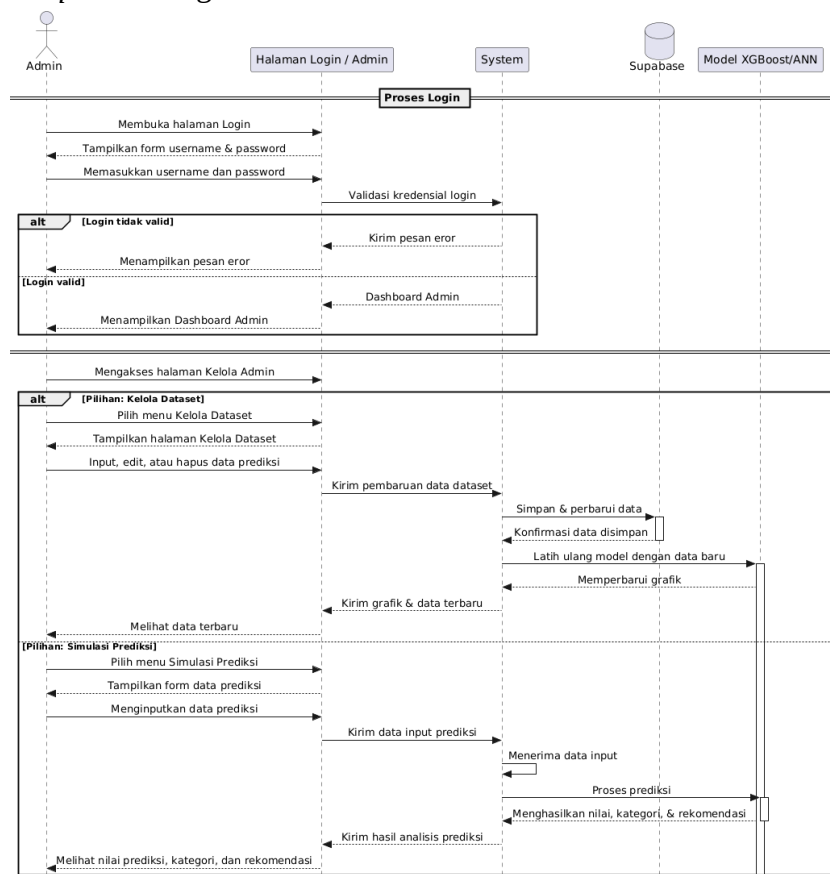


Gambar 11 *Sequence* Diagram User

*Sequence* diagram tersebut menggambarkan urutan interaksi antara pengguna, website, FastAPI, dan model model yang telah diimplementasikan dalam proses prediksi produksi padi. Proses diawali ketika pengguna mengakses website dan sistem menampilkan halaman beranda. Selanjutnya pengguna memilih komoditas yang akan diprediksi sehingga website menampilkan formulir input prediksi. Setelah pengguna mengisi data prediksi dan menekan tombol prediksi,

data dikirimkan dari website ke FastAPI untuk dilakukan validasi. Apabila terdapat data yang tidak valid atau tidak lengkap, FastAPI akan mengirimkan pesan kesalahan ke website yang kemudian ditampilkan kepada pengguna. Namun apabila data valid, data tersebut diteruskan ke model model yang telah diimplementasikan untuk dilakukan proses prediksi. Model kemudian menghasilkan nilai prediksi beserta rekomendasi yang dikembalikan ke FastAPI, selanjutnya diteruskan ke website untuk ditampilkan kepada pengguna. Proses berakhir setelah hasil prediksi dan rekomendasi berhasil ditampilkan pada halaman sistem.

#### b) Sequence Diagram Admin



Gambar 12 Sequence Diagram Admin

Sequence diagram tersebut menunjukkan urutan interaksi antara administrator, sistem, database Supabase, dan model model yang telah diimplementasikan. Proses dimulai dari login administrator yang divalidasi oleh sistem. Setelah berhasil masuk ke *dashboard*, administrator dapat memilih dua aktivitas utama, yaitu mengelola dataset atau melakukan simulasi. Pada pengelolaan dataset, perubahan data disimpan ke database Supabase, kemudian sistem melakukan pelatihan ulang model dan memperbarui grafik berdasarkan data

terbaru. Pada simulasi prediksi, data input yang dimasukkan administrator diproses menggunakan model model yang telah diimplementasikan untuk menghasilkan nilai prediksi, kategori produksi, dan rekomendasi yang kemudian ditampilkan pada halaman admin.

d. *Construction of Prototype*

Pada tahap ini dilakukan pembangunan *prototype* sistem berbasis web sesuai dengan desain yang telah dibuat. *Prototype* yang dibangun mencakup halaman beranda, halaman prediksi, halaman login administrator, *dashboard* admin, serta halaman pengelolaan data produksi. Selain itu, dikembangkan fitur simulasi prediksi yang dapat digunakan oleh administrator untuk melakukan pengujian prediksi secara langsung melalui sistem. Model *machine learning* yang telah dilatih sebelumnya kemudian diintegrasikan ke dalam sistem web melalui layanan *backend*. Proses integrasi ini memungkinkan sistem menerima data input dari pengguna maupun administrator, melakukan pengolahan data secara otomatis menggunakan model prediksi, kemudian menampilkan hasil prediksi produksi padi, kategori produksi, dan rekomendasi yang sesuai. Pada tahap ini juga dilakukan integrasi dengan database untuk mendukung penyimpanan, pengelolaan, dan pembaruan data yang digunakan dalam sistem.

Untuk mendukung proses prediksi produksi padi berbasis web, dibangun arsitektur sistem menggunakan pendekatan monolitik (*monolithic architecture*) yang terdiri dari tiga lapisan utama, yaitu lapisan presentasi (*frontend*), lapisan aplikasi (*backend*), dan lapisan data (database). Pada pendekatan ini, seluruh fungsi sistem dijalankan dalam satu aplikasi yang terintegrasi sehingga memudahkan proses pengembangan, pemeliharaan, dan implementasi. Lapisan *frontend* dibangun menggunakan HTML, CSS, dan JavaScript yang berfungsi sebagai antarmuka bagi pengguna dan administrator. Lapisan *backend* menggunakan *framework* FastAPI yang menangani proses autentikasi, pengelolaan data, komunikasi dengan database, serta integrasi dengan model *machine learning*. Sementara itu, lapisan data menggunakan Supabase sebagai media penyimpanan data pengguna, data produksi pertanian, riwayat prediksi, dan berbagai informasi yang digunakan oleh sistem. Dengan pendekatan monolitik ini, seluruh komponen aplikasi dapat berjalan dalam satu kesatuan sistem sehingga mendukung proses pengelolaan data dan prediksi produksi padi secara terintegrasi.

e. *Deployment Delivery & Feedback*

Tahap terakhir adalah implementasi *prototype* kepada pengguna untuk diuji dan dievaluasi. Pada tahap ini, pengguna melakukan peninjauan terhadap *prototype* yang telah dibangun guna memastikan kesesuaiannya dengan kebutuhan yang telah ditentukan pada tahap komunikasi. Apabila ditemukan ketidaksesuaian atau diperlukan penambahan fitur, maka proses akan kembali ke tahap *communication* untuk dilakukan perbaikan. Namun, jika *prototype* telah sesuai dengan kebutuhan, maka sistem akan dilanjutkan ke tahap penyempurnaan hingga menjadi sistem akhir yang siap digunakan.

6. Pengujian Sistem

Tahap pengujian dilakukan setelah proses implementasi model ke dalam sistem berbasis web selesai. Pengujian ini bertujuan untuk memastikan bahwa seluruh fungsi sistem berjalan sesuai dengan perancangan serta mampu menghasilkan keluaran prediksi dengan benar berdasarkan input yang diberikan pengguna.

Metode pengujian yang digunakan adalah *black box testing*, yaitu pengujian yang berfokus pada fungsi sistem tanpa melihat struktur kode program di dalamnya. Pengujian dilakukan dengan memberikan berbagai kombinasi input parameter seperti tahun, bulan tanam, serta variabel iklim, kemudian mengamati apakah sistem mampu memproses data tersebut dan menampilkan hasil prediksi sesuai dengan yang diharapkan.

Berikut merupakan skenario pengujian pada sistem prediksi produksi padi berbasis web.

a. Skenario Pengujian User

Tabel 5 Skenario Pengujian User

| <b>Skenario Pengujian</b> | <b>Langkah Pengujian</b>  | <b>Hasil yang diharapkan</b>           | <b>Hasil Pengujian</b>             |
|---------------------------|---------------------------|----------------------------------------|------------------------------------|
| (1)                       | (2)                       | (3)                                    | (4)                                |
| Akses beranda             | Mengakses halaman website | Menampilkan halaman beranda            | [ ] Berhasil<br>[ ] Tidak Berhasil |
| Akses menu prediksi       | Klik menu “Prediksi”      | Menampilkan pilihan komoditas prediksi | [ ] Berhasil<br>[ ] Tidak Berhasil |
| Akses prediksi padi       | Klik komoditas “Padi”     | Menampilkan halaman prediksi padi      | [ ] Berhasil<br>[ ] Tidak Berhasil |

| <b>Skenario Pengujian</b>    | <b>Langkah Pengujian</b>                            | <b>Hasil yang diharapkan</b>                                             | <b>Hasil Pengujian</b>             |
|------------------------------|-----------------------------------------------------|--------------------------------------------------------------------------|------------------------------------|
| (1)                          | (2)                                                 | (3)                                                                      | (4)                                |
| Input variabel prediksi      | Menginputkan variabel prediksi                      | Data variabel prediksi dapat diinput                                     | [ ] Berhasil<br>[ ] Tidak Berhasil |
| Validasi input tidak lengkap | Klik tombol “Hitung Prediksi” tanpa melengkapi data | Sistem menampilkan pesan eror                                            | [ ] Berhasil<br>[ ] Tidak Berhasil |
| Proses prediksi              | Klik tombol “Hitung Prediksi” dengan data lengkap   | Sistem memproses data dan menampilkan hasil prediksi beserta kategorinya | [ ] Berhasil<br>[ ] Tidak Berhasil |
| Tampilan rekomendasi         | Setelah prediksi berhasil                           | Sistem menampilkan rekomendasi sesuai kategorinya                        | [ ] Berhasil<br>[ ] Tidak Berhasil |

b. Skenario Pengujian Admin

Tabel 6 Skenario Pengujian Admin

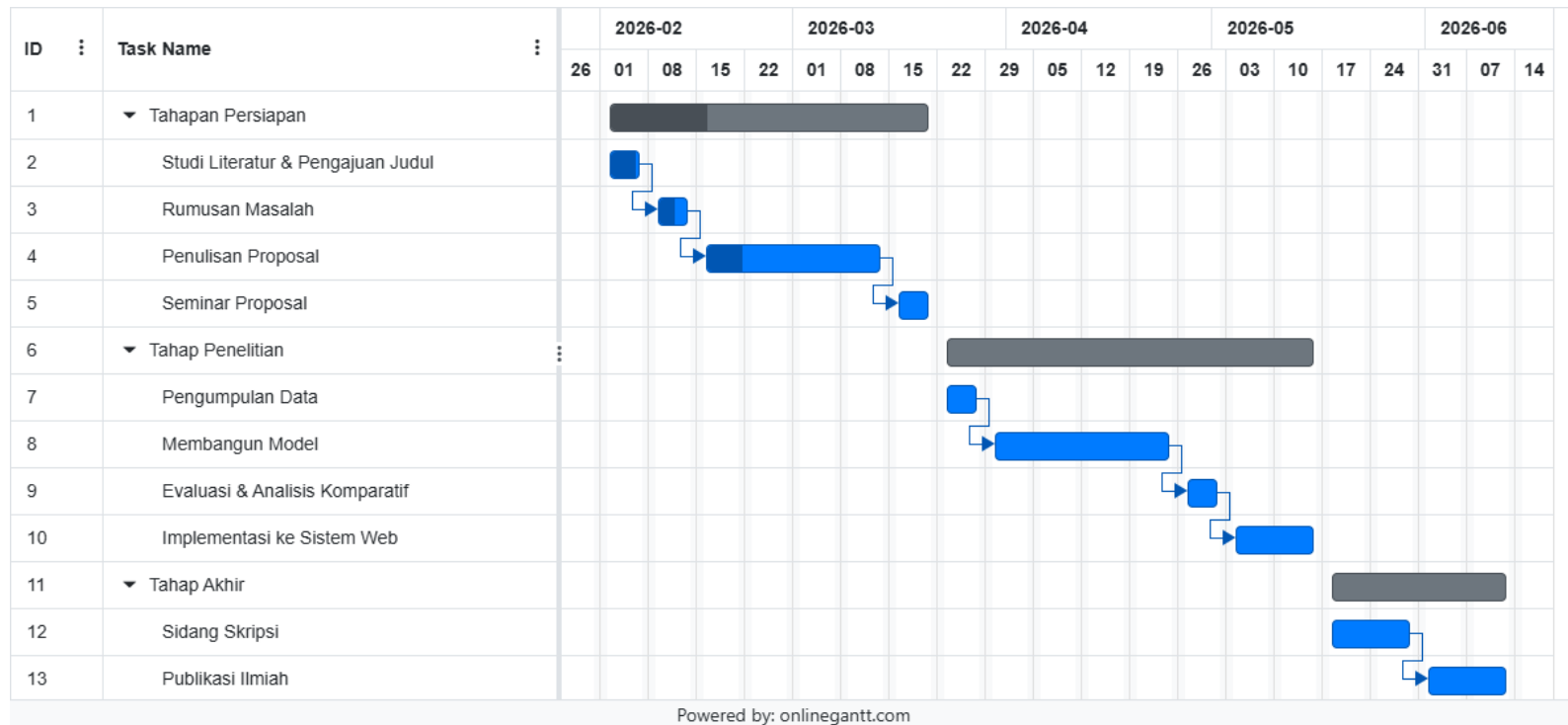
| <b>Skenario Pengujian</b>                 | <b>Langkah Pengujian</b>                                       | <b>Hasil yang diharapkan</b>                      | <b>Hasil Pengujian</b>             |
|-------------------------------------------|----------------------------------------------------------------|---------------------------------------------------|------------------------------------|
| (1)                                       | (2)                                                            | (3)                                               | (4)                                |
| Login Admin                               | Masuk ke halaman login admin                                   | Menampilkan halaman login admin                   | [ ] Berhasil<br>[ ] Tidak Berhasil |
| Input <i>username</i> dan <i>password</i> | Mengisi kolom <i>username</i> dan <i>password</i>              | <i>Username</i> dan <i>password</i> dapat diinput | [ ] Berhasil<br>[ ] Tidak Berhasil |
| Validasi salah inputan                    | Klik tombol “Masuk ke <i>Dashbord</i> ” dengan data yang salah | Sistem menampilkan pesan eror                     | [ ] Berhasil<br>[ ] Tidak Berhasil |
| Akses <i>dashboard</i> admin              | Setelah login berhasil                                         | Menampilkan <i>dashboard</i> admin                | [ ] Berhasil<br>[ ] Tidak Berhasil |

| <b>Skenario Pengujian</b>   | <b>Langkah Pengujian</b>                                                | <b>Hasil yang diharapkan</b>                                                    | <b>Hasil Pengujian</b>             |
|-----------------------------|-------------------------------------------------------------------------|---------------------------------------------------------------------------------|------------------------------------|
| (1)                         | (2)                                                                     | (3)                                                                             | (4)                                |
| Akses kelola prediksi       | Klik menu “Kelola Prediksi”                                             | Menampilkan halaman kelola prediksi                                             | [ ] Berhasil<br>[ ] Tidak Berhasil |
| Akses kelola prediksi padi  | Klik pilihan kelola prediksi padi                                       | Menampilkan halaman kelola prediksi padi                                        | [ ] Berhasil<br>[ ] Tidak Berhasil |
| Akses kelola dataset        | Klik pilihan “Kelola Dataset”                                           | Menampilkan halaman kelola dataset                                              | [ ] Berhasil<br>[ ] Tidak Berhasil |
| Input data baru             | Mengisi kolom tahun, bulan, prediksi, variabel iklim, dan produksi padi | Data tahun, bulan, prediksi, variabel iklim, dan produksi padi dapat diinputkan | [ ] Berhasil<br>[ ] Tidak Berhasil |
| Konfirmasi simpan data baru | Klik button “Simpan ke Dataset”                                         | Sistem menampilkan konfirmasi                                                   | [ ] Berhasil<br>[ ] Tidak Berhasil |
| Simpan dataset baru         | Klik button “Simpan”                                                    | Data tersimpan di database dan riwayat dataset terdaftar                        | [ ] Berhasil<br>[ ] Tidak Berhasil |
| Akses Edit data             | Pilih salah satu data lalu klik tombol edit                             | Menampilkan data yang akan diedit                                               | [ ] Berhasil<br>[ ] Tidak Berhasil |
| Melakukan edit data         | Mengganti data yang ada dengan data baru                                | Data dapat diubah ke data baru                                                  | [ ] Berhasil<br>[ ] Tidak Berhasil |
| Konfirmasi edit data        | Klik button “Simpan Perubahan”                                          | Sistem menampilkan konfirmasi                                                   | [ ] Berhasil<br>[ ] Tidak Berhasil |
| Simpan edit data            | Klik button “Simpan”                                                    | Data tersimpan di database dan riwayat dataset terdaftar                        | [ ] Berhasil<br>[ ] Tidak Berhasil |

| <b>Skenario Pengujian</b>                 | <b>Langkah Pengujian</b>                            | <b>Hasil yang diharapkan</b>                                                                          | <b>Hasil Pengujian</b>             |
|-------------------------------------------|-----------------------------------------------------|-------------------------------------------------------------------------------------------------------|------------------------------------|
| (1)                                       | (2)                                                 | (3)                                                                                                   | (4)                                |
| Akses hapus data                          | Pilih salah satu data lalu klik tombol edit         | Sistem menampilkan konfirmasi                                                                         | [ ] Berhasil<br>[ ] Tidak Berhasil |
| Konfirmasi edit data                      | Klik “Hapus Data”                                   | Data terhapus dari database dan riwayat dataset terdaftar                                             | [ ] Berhasil<br>[ ] Tidak Berhasil |
| Akses simulasi prediksi                   | Klik menu “Simulasi Prediksi”                       | Menampilkan halaman simulasi prediksi                                                                 | [ ] Berhasil<br>[ ] Tidak Berhasil |
| Input variabel prediksi                   | Mengisi variabel prediksi                           | Data variabel prediksi dapat diinput                                                                  | [ ] Berhasil<br>[ ] Tidak Berhasil |
| Validasi input tidak lengkap              | Klik tombol “Hitung Prediksi” tanpa melengkapi data | Sistem menampilkan pesan eror                                                                         | [ ] Berhasil<br>[ ] Tidak Berhasil |
| Proses prediksi                           | Klik tombol “Hitung Prediksi” dengan data lengkap   | Sistem memproses data dan menampilkan hasil prediksi                                                  | [ ] Berhasil<br>[ ] Tidak Berhasil |
| Tampilan hasil, kategori, dan rekomendasi | Setelah prediksi berhasil                           | Sistem menampilkan estimasi bulan panen, prediksi panen dalam ton, kategori prediksi, dan rekomendasi | [ ] Berhasil<br>[ ] Tidak Berhasil |

#### **D. Jadwal Penelitian**

Jadwal penelitian disusun untuk mengatur alokasi waktu agar seluruh tahapan berjalan terarah dan terstruktur. Dengan adanya jadwal ini, peneliti dapat Memastikan bahwa setiap kegiatan penelitian dilaksanakan sesuai dengan rencana. Jadwal penelitian dibuat dalam bentuk *Gantt Chart* yang memudahkan visualisasi tahapan kegiatan beserta durasi waktunya secara rinci. Berikut ini *Gantt Chart* jadwal penelitian dapat dilihat pada Gambar 11.



Gambar 13 *Gantt Chart* Jadwal Penelitian

## BAB IV HASIL DAN PEMBAHASAN

### A. Hasil

#### 1. Pengambilan Data

Pada tahap ini dilakukan proses pengumpulan data yang digunakan dalam penelitian prediksi produksi padi berbasis variabel iklim. Data yang digunakan merupakan data sekunder yang diperoleh dari Badan Meteorologi, Klimatologi, dan Geofisika (BMKG) Cilacap serta Dinas Pertanian Kabupaten Cilacap. Data dikumpulkan dalam periode bulanan selama lima tahun, yaitu mulai tahun 2020 hingga 2024. Berdasarkan data yang diperoleh, terlihat adanya fluktuasi pada variabel iklim maupun hasil produksi padi dari waktu ke waktu. Perubahan tersebut menunjukkan bahwa kondisi iklim memiliki keterkaitan terhadap naik turunnya produksi padi, sehingga data historis tersebut dapat digunakan dalam proses pembangunan model prediksi.

##### a. Spesifikasi Data

Data yang digunakan dalam penelitian ini terdiri dari beberapa variabel iklim serta data produksi padi. Adapun variabel yang digunakan ditunjukkan pada tabel berikut.

Tabel 7 Spesifikasi Data

| No  | Variabel                        | Deskripsi                                                                             | Sumber Data                                                                         |
|-----|---------------------------------|---------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------|
| (1) | (2)                             | (3)                                                                                   | (4)                                                                                 |
| 1.  | Lama Penyinaran Matahari (LPM)  | Data rata-rata lama penyinaran matahari bulanan dalam satuan persen (%).              | BMKG Cilacap                                                                        |
| 2.  | Curah Hujan (CH)                | Data jumlah curah hujan bulanan dalam satuan milimeter (mm).                          | BMKG Cilacap                                                                        |
| 3.  | <i>Oceanic Niño Index</i> (ONI) | Data indeks ENSO untuk merepresentasikan fenomena <i>El Niño</i> dan <i>La Niña</i> . | <a href="https://ggweather.com/enso/oni.htm">https://ggweather.com/enso/oni.htm</a> |
| 4.  | Suhu                            | Data rata-rata suhu bulanan dalam satuan derajat <i>Celcius</i> (°C).                 | BMKG Cilacap                                                                        |

| No  | Variabel      | Deskripsi                                                  | Sumber Data                       |
|-----|---------------|------------------------------------------------------------|-----------------------------------|
| (1) | (2)           | (3)                                                        | (4)                               |
| 5.  | Kelembaban    | Data rata-rata kelembaban bulanan dalam satuan persen (%). | BMKG Cilacap                      |
| 6.  | Produksi Padi | Data rata-rata produksi padi bulanan dalam satuan ton.     | Dinas Pertanian Kabupaten Cilacap |

Tabel 8 Contoh Data

| Tahun | Bulan    | LPM (%) | Curah Hujan (mm) | ONI   | Suhu (°C) | Kelembaban (%) | P (ton) |
|-------|----------|---------|------------------|-------|-----------|----------------|---------|
| (1)   | (2)      | (3)     | (4)              | (5)   | (6)       | (7)            | (8)     |
| 2020  | Januari  | 59      | 159,0            | 0,5   | 28,1      | 83             | 13.726  |
| 2020  | Februari | 59      | 342,5            | 0,48  | 27,9      | 83             | 32.621  |
| 2020  | Maret    | 63      | 519,0            | 0,4   | 27,9      | 84             | 143.882 |
| 2020  | April    | 74      | 274,0            | 0,19  | 28,1      | 84             | 186.587 |
| 2020  | Mei      | 66      | 380,4            | -0,08 | 28,2      | 84             | 100.636 |
| 2020  | Juni     | 80      | 520,2            | -0,3  | 27,3      | 84             | 16.157  |
| 2020  | Juli     | 80      | 54,0             | -0,41 | 26,8      | 81             | 70.449  |
| 2020  | Agustus  | 77      | 35,0             | -0,57 | 26,6      | 82             | 173.063 |
| ⋮     | ⋮        | ⋮       | ⋮                | ⋮     | ⋮         | ⋮              | ⋮       |
| 2024  | Desember | 40      | 464              | -0,53 | 27,9      | 84             | 7.797   |

b. Alasan Pemilihan Data

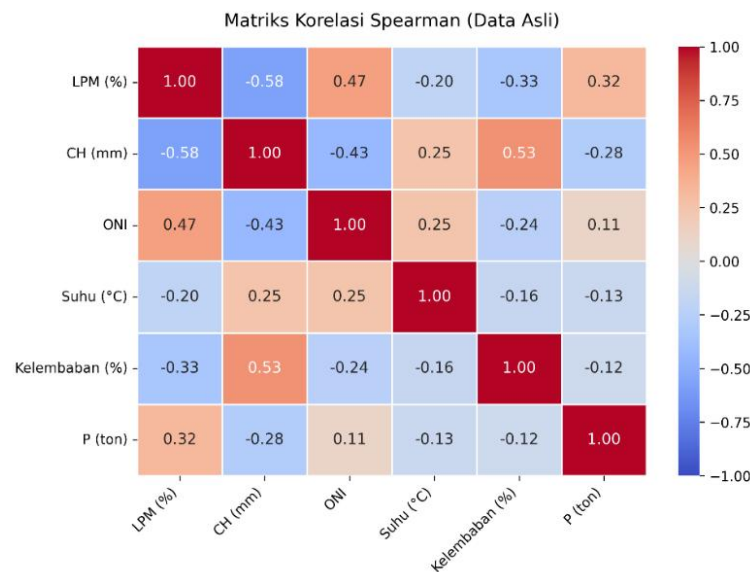
Pemilihan data dalam penelitian ini dilakukan berdasarkan keterkaitan variabel iklim terhadap produksi padi. Data yang digunakan merupakan data sekunder yang diperoleh dari instansi resmi, yaitu BMKG Cilacap dan Dinas Pertanian Kabupaten Cilacap, sehingga memiliki tingkat validitas dan kredibilitas yang baik untuk digunakan dalam penelitian. Variabel iklim yang dipilih meliputi lama penyinaran matahari, curah hujan, suhu udara, kelembaban udara, dan *Oceanic Niño Index* (ONI) karena variabel tersebut memiliki pengaruh terhadap pertumbuhan tanaman padi dan hasil produksi panen.

- 1) Lama penyinaran matahari dipilih karena berpengaruh terhadap proses fotosintesis tanaman padi yang memengaruhi pertumbuhan dan pembentukan hasil panen
- 2) Curah hujan dipilih karena ketersediaan air menjadi salah satu faktor utama dalam pertumbuhan tanaman padi, terutama pada fase tanam hingga panen.
- 3) Suhu udara dipilih karena perubahan suhu dapat memengaruhi perkembangan hama yang berpotensi menurunkan produksi padi.
- 4) Kelembaban udara dipilih karena kondisi kelembaban tertentu dapat memengaruhi perkembangan tanaman maupun risiko penyakit jamur pada tanaman padi.
- 5) *Oceanic Niño Index* (ONI) dipilih karena merepresentasikan fenomena *El Niño* dan *La Niña* yang berpengaruh terhadap perubahan pola iklim di Indonesia. Pada fase *El Niño* terjadi kecenderungan kemarau panjang yang berdampak pada penurunan produksi padi, sedangkan pada fase *La Niña* terjadi peningkatan curah hujan yang berpotensi menimbulkan banjir dan gagal panen.
- 6) Data produksi padi dipilih sebagai variabel target karena digunakan untuk mengetahui hasil prediksi produksi berdasarkan pengaruh variabel iklim yang digunakan dalam penelitian.

Dengan menggunakan parameter tersebut, penelitian ini diharapkan mampu memodelkan hubungan antara kondisi iklim dan produksi padi sehingga dapat digunakan dalam proses prediksi produksi padi yang akurat.

Untuk memperkuat pemilihan variabel yang digunakan dalam penelitian, dilakukan analisis korelasi antara variabel iklim dan produksi padi menggunakan matriks korelasi *spearman*. Analisis ini bertujuan untuk mengetahui tingkat hubungan antar variabel serta melihat sejauh mana masing-masing variabel iklim memiliki keterkaitan dengan produksi padi. Korelasi *spearman* dipilih karena mampu mengukur kekuatan hubungan antar variabel serta dapat menangkap hubungan yang bersifat nonlinier, sehingga lebih sesuai untuk menganalisis keterkaitan antara faktor iklim dan produksi padi yang

tidak selalu memiliki pola hubungan linear. Berikut merupakan hasil matriks korelasi *spearman* pada variabel penelitian.



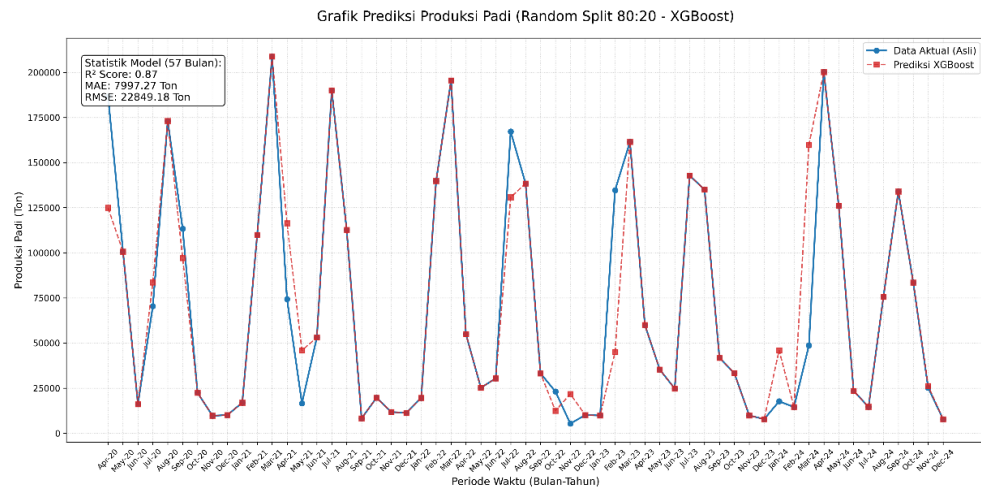
*Gambar 14 Korelasi Variabel Iklim dan Produksi Padi*

Berdasarkan grafik di atas, hasil analisis korelasi *spearman* menunjukkan bahwa lama penyinaran matahari (LPM) memiliki korelasi positif tertinggi terhadap produksi padi dengan nilai 0,32, sedangkan curah hujan memiliki korelasi negatif sebesar -0,28. Variabel ONI memiliki korelasi positif sebesar 0,11, sedangkan suhu udara dan kelembaban udara masing-masing memiliki korelasi negatif sebesar -0,13 dan -0,12 terhadap produksi padi. Secara umum, seluruh variabel menunjukkan tingkat korelasi yang relatif lemah, namun tetap digunakan dalam penelitian karena secara teoritis berpengaruh terhadap produksi padi serta dapat dimanfaatkan oleh model XGBoost dan ANN untuk mempelajari pola hubungan yang kompleks faktor iklim dan produksi padi.

## 2. Pembangunan dan Pelatihan Model

Pada tahap ini dilakukan proses pembangunan dan pelatihan model prediksi produksi padi menggunakan algoritma XGBoost dan ANN. Proses pembangunan model diimplementasikan menggunakan bahasa pemrograman Python dengan bantuan beberapa *library* seperti *pandas*, *numpy*, *matplotlib*, dan *scikit-learn*. Tahap ini meliputi proses inisialisasi library, pemanggilan dataset, penerapan lagging data, penentuan fitur dan target, pembagian data latih dan data uji, pelatihan model, evaluasi model, hingga visualisasi hasil prediksi.

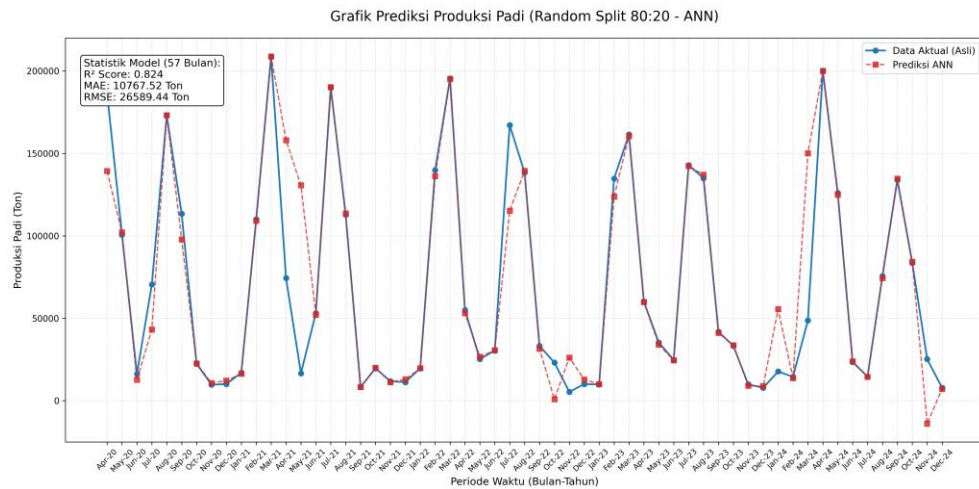
a. Hasil Model *Extreme Gradient Boosting* (XGBoost)



Gambar 15 Grafik Prediksi Produksi Padi (XGBoost)

Berdasarkan grafik di atas, model XGBoost mampu mengikuti pola perubahan produksi padi pada sebagian besar periode pengamatan. Hal ini terlihat dari kedekatan antara garis prediksi dan data aktual pada beberapa titik, terutama saat terjadi kenaikan maupun penurunan produksi. Nilai prediksi XGBoost terlihat mengalami perubahan yang cukup bervariasi antarperiode, sejalan dengan fluktuasi produksi padi pada data aktual. Meskipun masih terdapat beberapa selisih prediksi pada periode tertentu, secara umum model dapat menangkap pola hubungan antara variabel iklim dan produksi padi dengan cukup baik. Hasil visualisasi ini didukung oleh nilai  $R^2$  sebesar 0.87, MAE sebesar 7.997,27 ton, dan RMSE sebesar 22.849,18 ton, yang menunjukkan bahwa model XGBoost memiliki kemampuan prediksi yang baik terhadap data produksi padi.

## b. Hasil Model Artificial Neural Network (ANN)

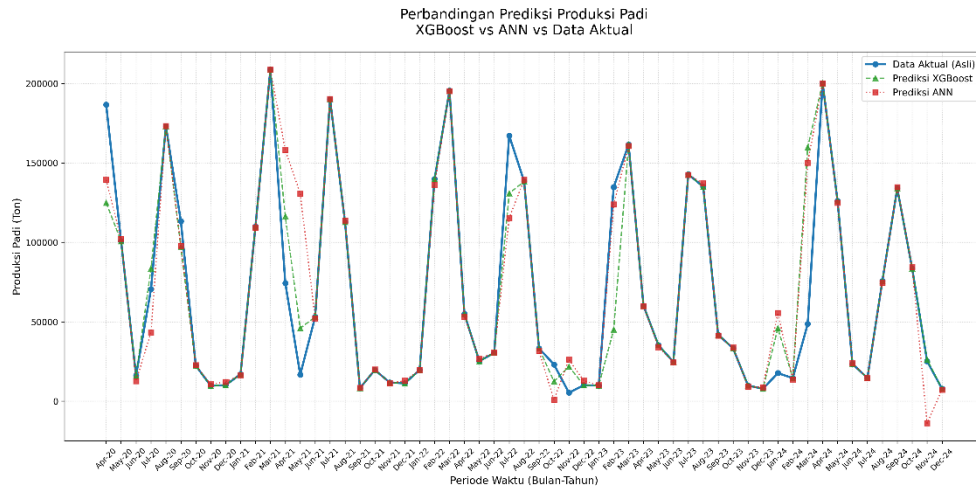


*Gambar 16 Grafik Prediksi Produksi Padi (ANN)*

Berdasarkan grafik di atas, model ANN mampu mengikuti pola perubahan produksi padi pada sebagian besar periode pengamatan. Hal ini terlihat dari kedekatan antara garis prediksi dan data aktual pada berbagai periode, baik saat terjadi peningkatan maupun penurunan produksi. Meskipun masih terdapat beberapa perbedaan antara nilai prediksi dan data aktual pada beberapa titik, secara umum model ANN dapat mempelajari hubungan antara variabel iklim dan produksi padi dengan cukup baik. Hasil visualisasi ini didukung oleh nilai  $R^2$  sebesar 0,824, MAE sebesar 10.767,52 ton, dan RMSE sebesar 26.589,44 ton. Nilai tersebut menunjukkan bahwa model ANN mampu menjelaskan sebagian besar variasi produksi padi, meskipun masih terdapat kesalahan prediksi pada beberapa periode pengamatan.

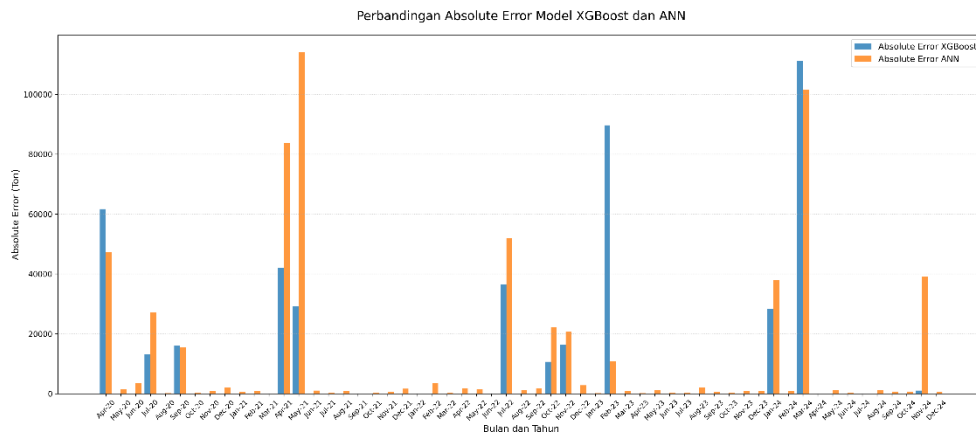
### 3. Evaluasi dan Analisis Komparatif

Untuk memberikan gambaran visual mengenai perbandingan hasil prediksi kedua model, dilakukan visualisasi hasil prediksi XGBoost dan ANN dalam satu grafik. Visualisasi ini menampilkan hasil prediksi kedua model beserta data aktual pada setiap periode pengamatan sehingga perbedaan kedekatan hasil prediksi terhadap data aktual dapat diamati secara langsung. Hasil perbandingan prediksi XGBoost dan ANN disajikan pada Gambar berikut.



Gambar 17 Grafik Hasil Prediksi XGBoost dan ANN

Berdasarkan Gambar 17, hasil prediksi model XGBoost dan ANN secara umum mampu mengikuti pola perubahan produksi padi pada berbagai periode pengamatan. Pada beberapa periode, hasil prediksi kedua model berada dekat dengan data aktual. Namun, terdapat beberapa periode yang menunjukkan perbedaan antara hasil prediksi XGBoost dan ANN. Secara visual, hasil prediksi XGBoost cenderung lebih mendekati data aktual dibandingkan ANN pada beberapa periode pengamatan. Secara visual, hasil prediksi XGBoost cenderung lebih mendekati data aktual dibandingkan ANN. Untuk memberikan gambaran mengenai tingkat kesalahan prediksi yang dihasilkan oleh masing-masing model, dilakukan perbandingan nilai *Absolute Error* (AE) XGBoost dan ANN yang disajikan pada Gambar 18.



Gambar 18 Grafik Perbandingan *Absolute Error* XGBoost dan ANN

Berdasarkan Gambar 18, nilai *Absolute Error* (AE) kedua model menunjukkan variasi pada setiap periode pengamatan. Pada 53 dari 57 periode pengamatan atau sebesar 92,98%, model XGBoost menunjukkan hasil yang lebih baik dibandingkan ANN karena menghasilkan nilai AE yang lebih rendah. Hal tersebut menunjukkan bahwa hasil prediksi XGBoost memiliki selisih yang lebih kecil terhadap data aktual pada sebagian besar periode

pengamatan. Namun, pada 4 dari 57 periode atau sebesar 7,02%, model ANN menghasilkan nilai AE yang lebih rendah dibandingkan XGBoost. Oleh karena itu, perbandingan performa kedua model tidak hanya dilakukan berdasarkan visualisasi hasil prediksi dan nilai AE, tetapi juga menggunakan metrik  $R^2$ , MAE, dan RMSE. Hasil evaluasi kedua model disajikan pada Tabel berikut.

Tabel 9 Hasil Analisis

| No  | Metode                                     | $R^2$ | MAE (ton) | RMSE (ton) |
|-----|--------------------------------------------|-------|-----------|------------|
| (1) | (2)                                        | (3)   | (4)       | (5)        |
| 1.  | <i>Extreme Gradient Boosting</i> (XGBoost) | 0,870 | 7.997,27  | 22.849,18  |
| 2.  | <i>Artificial Neural Network</i> (ANN)     | 0,824 | 10.767,52 | 26.589,44  |

Berdasarkan hasil evaluasi yang ditunjukkan pada Tabel 9, model XGBoost menghasilkan nilai  $R^2$  sebesar 0,870, sedangkan model ANN menghasilkan nilai  $R^2$  sebesar 0,824. Nilai tersebut menunjukkan bahwa kedua model mampu menjelaskan lebih dari 80% variasi data produksi padi, namun XGBoost memiliki kemampuan yang lebih baik dalam memodelkan hubungan antara variabel iklim dan produksi padi pada dataset penelitian ini.

Pada metrik MAE, model XGBoost memperoleh nilai sebesar 7.997,27 ton, lebih rendah dibandingkan ANN yang memperoleh nilai sebesar 10.767,52 ton. Hal ini menunjukkan bahwa rata-rata kesalahan prediksi yang dihasilkan XGBoost lebih kecil dibandingkan ANN. Hasil serupa juga terlihat pada metrik RMSE, di mana XGBoost menghasilkan nilai sebesar 22.849,18 ton, sedangkan ANN menghasilkan nilai sebesar 26.589,44 ton. Nilai RMSE yang lebih rendah menunjukkan bahwa kesalahan prediksi yang besar dapat diminimalkan dengan lebih baik oleh model XGBoost.

Secara keseluruhan, model XGBoost menunjukkan performa yang lebih unggul dibandingkan model ANN karena memiliki nilai  $R^2$  yang lebih tinggi serta nilai MAE dan RMSE yang lebih rendah. Hasil ini menunjukkan bahwa pada dataset penelitian ini, algoritma XGBoost lebih efektif dalam mempelajari pola hubungan antara variabel iklim dan produksi padi. Oleh karena itu, model XGBoost dipilih untuk diimplementasikan pada sistem prediksi produksi padi berbasis web.

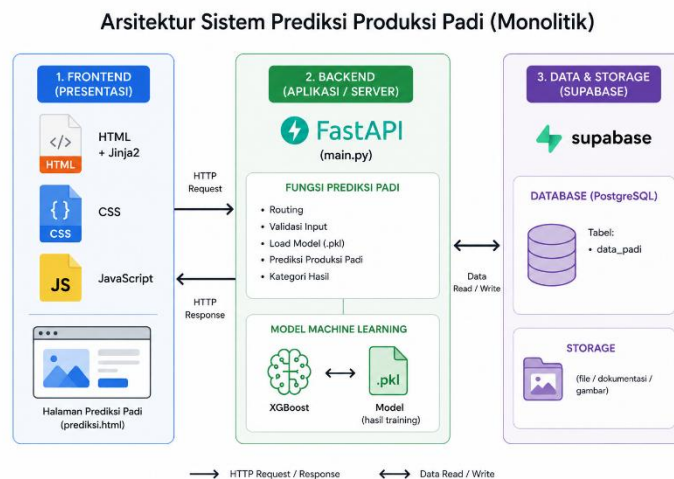
#### 4. Implementasi Web

Setelah diperoleh model terbaik berdasarkan hasil evaluasi dan analisis komparatif, tahap selanjutnya adalah implementasi model ke dalam sistem prediksi produksi padi berbasis web. Proses implementasi dilakukan

menggunakan metode *prototyping*, yaitu dengan membangun sistem secara bertahap berdasarkan kebutuhan yang telah dirancang pada tahap sebelumnya. Pada tahap ini, model prediksi diimplementasikan ke dalam sistem berbasis web yang telah dirancang sehingga pengguna dapat melakukan prediksi produksi padi secara langsung melalui antarmuka yang telah disediakan.

#### a. Arsitektur Sistem

Arsitektur sistem digunakan untuk menggambarkan struktur dan hubungan antar komponen yang membangun sistem prediksi produksi padi berbasis web. Pada penelitian ini, sistem dibangun menggunakan arsitektur monolitik (*monolithic architecture*), yaitu seluruh komponen aplikasi seperti antarmuka pengguna, logika aplikasi, model *machine learning*, dan akses database dijalankan dalam satu aplikasi yang terintegrasi. Pendekatan ini dipilih karena lebih sederhana dalam proses pengembangan, implementasi, dan pemeliharaan sistem. Arsitektur sistem terdiri dari tiga lapisan utama, yaitu lapisan presentasi (*frontend*), lapisan aplikasi (*backend*), dan lapisan database.



Gambar 19 Arsitektur Sistem

##### 1) Lapisan Presentasi (*Frontend*)

Lapisan presentasi (*frontend*) merupakan bagian sistem yang berinteraksi secara langsung dengan pengguna dan administrator melalui web browser. Pada penelitian ini, *frontend* dibangun menggunakan HTML, CSS, dan JavaScript, sedangkan proses *rendering* halaman dilakukan menggunakan *template engine Jinja2* yang terintegrasi dengan *framework* FastAPI. HTML digunakan untuk menyusun struktur dan konten halaman web, CSS digunakan untuk mengatur tampilan dan tata letak antarmuka agar lebih menarik dan mudah digunakan, sedangkan *JavaScript* digunakan untuk menambahkan fungsi interaktif pada halaman web. Lapisan ini berfungsi untuk menampilkan antarmuka

sistem, menerima masukan dari pengguna, serta menyajikan hasil prediksi yang dihasilkan oleh model *machine learning*.

Halaman yang tersedia pada sisi pengguna meliputi halaman beranda dan halaman prediksi. Sementara itu, pada sisi administrator tersedia halaman login, *dashboard* admin, serta halaman pengelolaan data produksi. Ketika pengguna mengakses suatu halaman, browser akan mengirimkan permintaan (*request*) ke server FastAPI. Selanjutnya, FastAPI memproses permintaan tersebut dan mengembalikan halaman HTML yang telah dirender menggunakan Jinja2 sehingga dapat ditampilkan kepada pengguna.

## 2) Lapisan Aplikasi (*Backend*)

Lapisan aplikasi (*backend*) merupakan bagian sistem yang bertugas mengelola logika bisnis, memproses permintaan dari pengguna, menghubungkan sistem dengan database, serta menjalankan proses prediksi menggunakan model *machine learning*. Pada penelitian ini, *backend* dibangun menggunakan *framework* FastAPI dengan seluruh fungsi utama sistem diimplementasikan dalam satu file utama, yaitu *main.py*. Pendekatan ini sejalan dengan arsitektur monolitik yang digunakan, di mana seluruh komponen aplikasi dijalankan dalam satu sistem yang terintegrasi.

*Backend* bertanggung jawab untuk menangani berbagai proses, seperti autentikasi pengguna dan administrator, pengelolaan data produksi pertanian, pengambilan data dari database Supabase, proses prediksi produksi padi, serta penyimpanan riwayat prediksi. Selain itu, *backend* juga berfungsi sebagai penghubung antara antarmuka web dan model *machine learning* yang telah dilatih sebelumnya.

Pada proses prediksi, *backend* menerima data parameter iklim yang dimasukkan oleh pengguna, kemudian memuat model XGBoost yang telah disimpan dalam file *pickle* (.pkl). Selanjutnya, model digunakan untuk menghasilkan nilai prediksi produksi padi yang kemudian dikategorikan ke dalam tingkat produksi tertentu dan ditampilkan kembali kepada pengguna beserta rekomendasi yang sesuai.

## 3) Lapisan Database

Lapisan database berfungsi sebagai media penyimpanan data yang digunakan oleh sistem. Pada penelitian ini, database dibangun menggunakan Supabase yang berbasis PostgreSQL. Supabase digunakan untuk menyimpan data historis produksi padi, data pengguna, riwayat prediksi, serta berbagai informasi pendukung lainnya. Selain itu, Supabase dapat diakses oleh *backend* melalui API sehingga proses pengelolaan data dapat dilakukan secara terpusat.

Salah satu tabel utama yang digunakan adalah data\_padi, yang berisi data produksi padi dan variabel iklim sebagai sumber data dalam proses pelatihan model dan prediksi. Struktur tabel tersebut ditunjukkan pada tabel berikut.

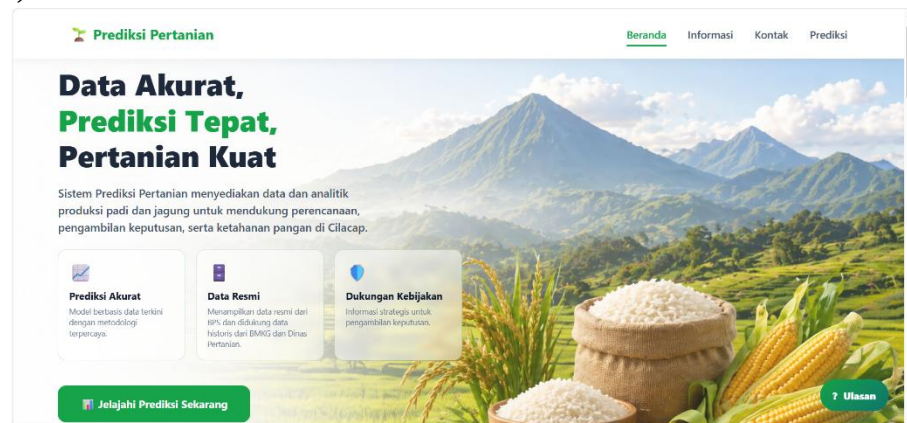
Tabel 10 Struktur Database

| No  | Nama Kolom | Tipe Data | Keterangan                   |
|-----|------------|-----------|------------------------------|
| (1) | (2)        | (3)       | (4)                          |
| 1   | tahun      | number    | Tahun produksi               |
| 2   | bulan      | text      | Bulan produksi               |
| 3   | lpm        | varchar   | Lama penyinaran matahari (%) |
| 4   | ch         | varchar   | Curah hujan (mm)             |
| 5   | oni        | varchar   | <i>Oceanic Niño Index</i>    |
| 6   | suhu       | varchar   | Suhu udara (°C)              |
| 7   | kelembaban | varchar   | Kelembaban udara (%)         |
| 8   | produksi   | varchar   | Produksi padi (ton)          |

Data pada tabel data\_padi digunakan sebagai sumber data utama dalam proses pelatihan model XGBoost dan implementasi prediksi pada sistem berbasis web.

## b. Implementasi Antarmuka Pengguna

### 1) Halaman Beranda

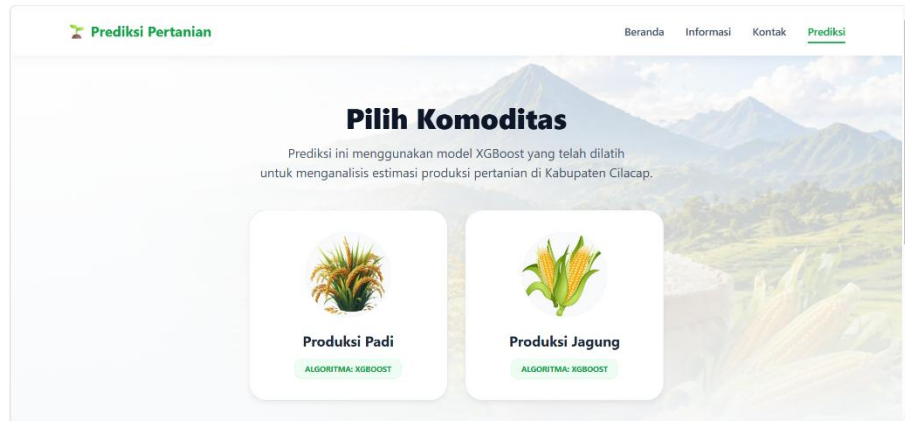


Gambar 20 Halaman Beranda

Halaman beranda merupakan halaman utama yang ditampilkan saat pengguna pertama kali mengakses website. Halaman ini berisi informasi mengenai sistem prediksi produksi pertanian serta menyediakan menu

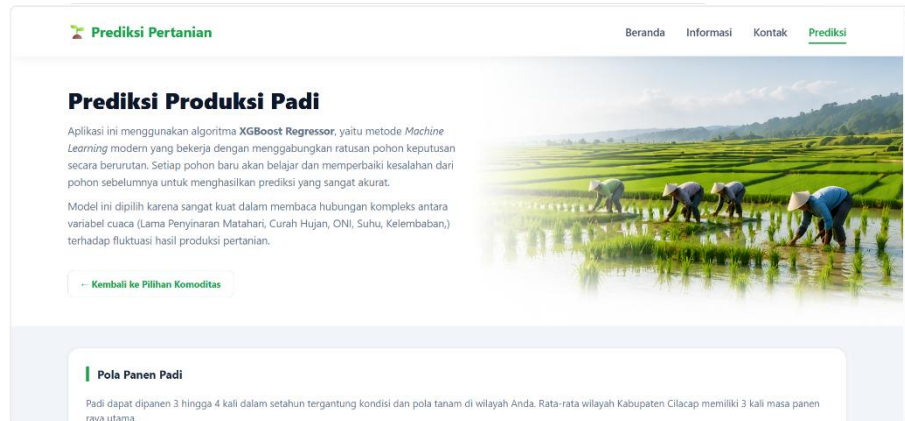
navigasi menuju fitur-fitur yang tersedia, termasuk halaman prediksi yang digunakan dalam penelitian.

## 2) Halaman Prediksi



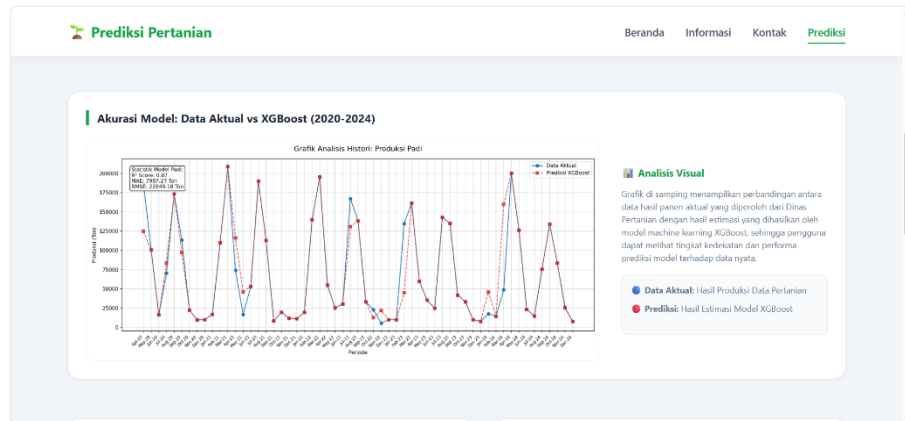
Gambar 21 Halaman Pemilihan Komoditas

Halaman ini digunakan untuk memilih komoditas yang akan diprediksi oleh sistem. Pengguna dapat memilih komoditas padi atau jagung sebelum melanjutkan ke proses prediksi. Pada penelitian ini, komoditas yang digunakan sebagai studi kasus adalah padi.



Gambar 22 Halaman Informasi Prediksi

Halaman ini menampilkan informasi mengenai prediksi produksi padi, variabel yang digunakan dalam proses prediksi, serta penjelasan singkat mengenai model yang diterapkan pada sistem.

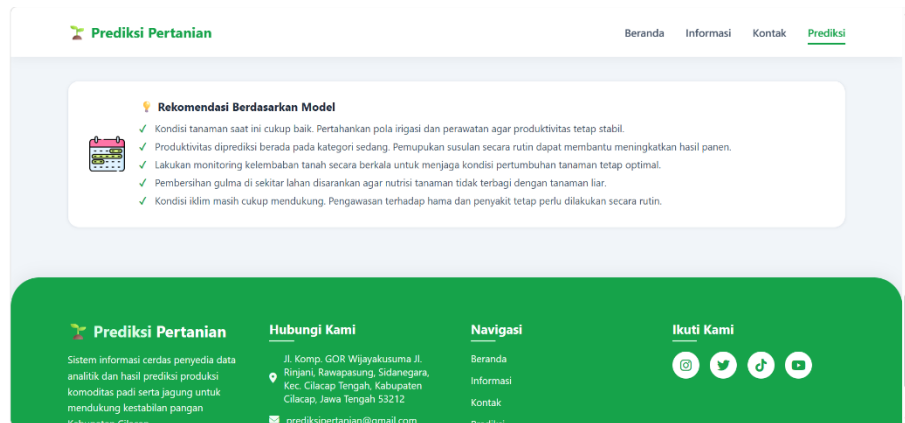


Gambar 23 Halaman Grafik Hasil Pelatihan Model

Halaman ini menampilkan grafik perbandingan antara data aktual dan hasil prediksi model XGBoost. Grafik digunakan untuk memberikan gambaran kepada pengguna mengenai kemampuan model dalam mengikuti pola produksi padi berdasarkan data historis yang digunakan pada proses pelatihan.

Gambar 24 Halaman Prediksi Produksi Padi

Halaman ini digunakan untuk melakukan prediksi produksi padi berdasarkan parameter iklim yang dimasukkan oleh pengguna. Pada halaman ini tersedia formulir input yang terdiri dari tahun tanam, bulan tanam, lama penyinaran matahari, curah hujan, *Oceanic Niño Index* (ONI), suhu udara, dan kelembaban udara. Setelah pengguna menekan tombol prediksi, sistem akan memproses data menggunakan model XGBoost yang telah diimplementasikan dan menampilkan hasil prediksi produksi padi dalam satuan ton. Selain itu, sistem menampilkan kategori produksi, yaitu rendah, sedang, atau tinggi, yang ditentukan menggunakan metode *quantile* dengan membagi data produksi ke dalam tiga kelompok sehingga hasil prediksi lebih mudah diinterpretasikan.

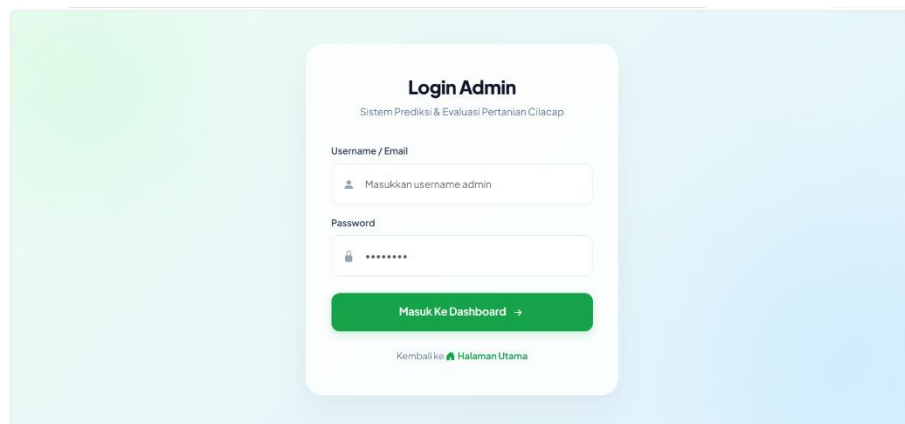


Gambar 25 Halaman Rekomendasi

Halaman rekomendasi menampilkan saran yang dihasilkan sistem berdasarkan kategori produksi yang diperoleh dari hasil prediksi. Setiap kategori memiliki rekomendasi yang berbeda sesuai dengan kondisi produksi yang diprediksi, sehingga pengguna dapat memperoleh informasi pendukung untuk membantu perencanaan dan pengambilan keputusan dalam pengelolaan produksi padi.

### c. Implementasi Antarmuka Admin

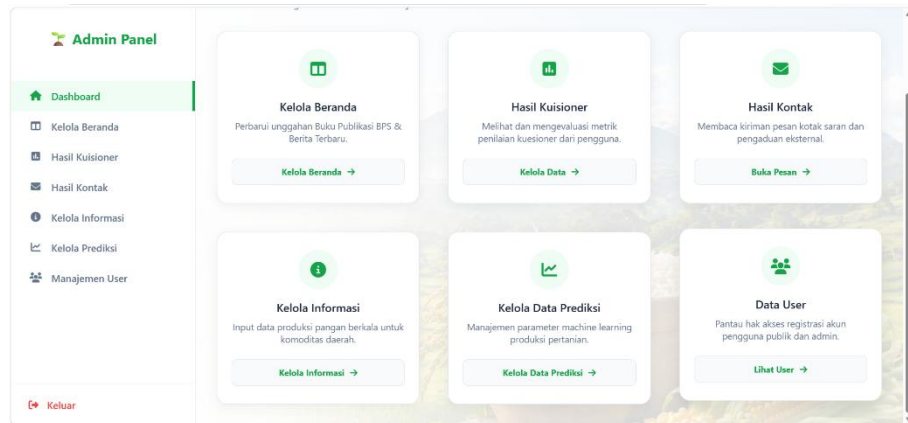
#### 1) Login Admin



Gambar 26 Halaman Login Admin

Halaman login admin digunakan sebagai mekanisme autentikasi sebelum administrator dapat mengakses fitur-fitur pengelolaan sistem. Administrator diwajibkan memasukkan *username* dan *password* yang valid untuk masuk ke *dashboard* admin.

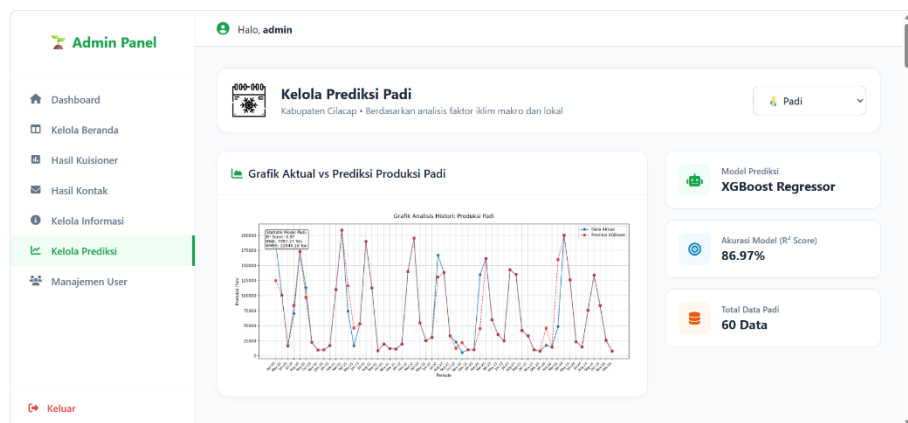
## 2) Dashboard Admin



Gambar 27 Halaman *Dashboard Admin*

Halaman *dashboard* admin merupakan halaman utama yang ditampilkan setelah administrator berhasil melakukan login. Halaman ini berfungsi sebagai pusat pengelolaan sistem yang menyediakan akses ke berbagai fitur administrasi, seperti pengelolaan beranda, informasi, kontak, kuisisioner, prediksi, dan manajemen pengguna. Pada penelitian ini, fokus implementasi diarahkan pada fitur pengelolaan data prediksi yang digunakan untuk mendukung proses prediksi produksi padi.

## 3) Kelola Data Produksi



Gambar 28 Halaman *Kelola Prediksi Padi*

Halaman *kelola prediksi* digunakan untuk menampilkan hasil pelatihan model prediksi yang telah diimplementasikan pada sistem. Halaman ini menampilkan grafik perbandingan antara data aktual dan hasil prediksi model XGBoost beserta informasi performa model seperti nilai  $R^2$ , MAE, dan RMSE. Informasi tersebut digunakan administrator

untuk memantau kualitas model prediksi yang digunakan dalam sistem sebelum diterapkan pada proses prediksi produksi padi.

The screenshot shows the 'Kelola Dataset' page with the following input fields and example values:

| Field                                   | Example Value     |
|-----------------------------------------|-------------------|
| Tahun                                   | -- Pilih Tahun -- |
| Bulan                                   | -- Pilih Bulan -- |
| LPM (Penyinaran %)                      | Contoh: 78        |
| Curah Hujan (mm)                        | Contoh: 225       |
| Suhu (°C)                               | Contoh: 29        |
| Kelembaban (%)                          | Contoh: 88        |
| ONI (Oceanic Niño Index)                | Contoh: -0.2      |
| Hasil Produksi Hasil Panen Aktual (Ton) | Contoh: 25700     |

Gambar 29 Halaman Kelola Dataset

Halaman kelola dataset digunakan untuk menambahkan data produksi dan variabel iklim yang akan digunakan oleh sistem. Administrator dapat melakukan input data baru sehingga dataset yang tersimpan pada database selalu terbaru dan siap digunakan dalam proses pelatihan model maupun prediksi.

The screenshot shows the 'Simulasi Prediksi' page with the following data:

| Parameter                    | Value |
|------------------------------|-------|
| Tahun Tanam                  | 2026  |
| Bulan Tanam                  | April |
| Lama Penyinaran Matahari (%) | 59    |
| Curah Hujan (mm)             | 159   |
| Suhu (°C)                    | 28,4  |
| Kelembaban (%)               | 85    |

**57.750,01 Ton**  
Prediksi Produksi Padi Bulan Juli  
Kategori: Sedang  
33% - 66% dari rata-rata

**Rekomendasi**

- ✓ Pertahankan pola irigasi dan perawatan.
- ✓ Pemupukan susulan rutin untuk menjaga hasil.

Gambar 30 Halaman Simulasi Prediksi

Halaman simulasi prediksi digunakan oleh administrator untuk melakukan prediksi produksi padi dengan memasukkan parameter iklim tertentu. Fitur ini memiliki fungsi yang sama dengan halaman prediksi pada sisi pengguna, namun disediakan dalam *dashboard* admin untuk kebutuhan pengujian dan validasi hasil prediksi.

| KOMODITAS | PERIODE       | LPM          | HUJAN    | SUHU          | KELEMBABAN   | ONI   | PRODUKSI AKTUAL | AKSI |
|-----------|---------------|--------------|----------|---------------|--------------|-------|-----------------|------|
|           | Januari 2020  | 58,5483871%  | 159 mm   | 28,11290323°C | 82,5372067%  | 0,5   | 13726 Ton       |      |
|           | Februari 2020 | 59,44827586% | 342,5 mm | 27,91206897°C | 83,16135047% | 0,48  | 32621 Ton       |      |
|           | Maret 2020    | 63,12903226% | 519 mm   | 27,88951613°C | 84,47309101% | 0,4   | 143882 Ton      |      |
|           | April 2020    | 73,66666667% | 274 mm   | 28,09°C       | 84,01201269% | 0,19  | 186587 Ton      |      |
|           | Mei 2020      | 65,93548387% | 380,4 mm | 28,2483871°C  | 84,0483871%  | 0,08  | 100636 Ton      |      |
|           | Juni 2020     | 80,1%        | 520,2 mm | 27,33666667°C | 84,10962353% | -0,3  | 16157 Ton       |      |
|           | Juli 2020     | 80%          | 54 mm    | 26,79516129°C | 81,33842568% | -0,41 | 70449 Ton       |      |

Gambar 31 Halaman Riwayat Dataset

Halaman riwayat data digunakan untuk mengelola data prediksi yang tersimpan pada sistem. Administrator dapat melihat, menambah, mengubah, maupun menghapus data (CRUD) sehingga seluruh riwayat prediksi dan data terkait dapat dikelola secara terpusat melalui *dashboard* admin.

## 5. Pengujian Sistem

Setelah proses implementasi sistem selesai dilakukan, tahap berikutnya adalah pengujian sistem untuk memastikan seluruh fitur dapat berfungsi dengan baik sesuai kebutuhan. Pengujian dilakukan menggunakan metode *black box testing* pada fitur-fitur utama yang tersedia pada sistem, baik dari sisi pengguna maupun administrator. Detail hasil pengujian disajikan pada Lampiran 1.

Hasil pengujian menunjukkan bahwa seluruh fitur yang diimplementasikan dapat beroperasi dengan baik dan menghasilkan keluaran sesuai dengan yang diharapkan. Sistem mampu menjalankan setiap proses secara terintegrasi, mulai dari penerimaan data masukan hingga penyajian hasil prediksi dan rekomendasi. Dengan demikian, sistem yang dikembangkan telah memenuhi kebutuhan fungsional dan siap digunakan sebagai media prediksi produksi padi berbasis web.

## B. Pembahasan

Berdasarkan hasil penelitian yang telah dilakukan, variabel iklim yang terdiri dari lama penyinaran matahari, curah hujan, *Oceanic Niño Index* (ONI), suhu udara, dan kelembaban udara dapat digunakan sebagai dasar dalam membangun model prediksi produksi padi di Kabupaten Cilacap. Pemilihan variabel tersebut didasarkan pada keterkaitannya dengan pertumbuhan dan perkembangan tanaman padi selama masa tanam hingga panen. Hasil analisis korelasi *spearman* menunjukkan bahwa hubungan masing-masing variabel terhadap produksi padi

cenderung lemah. Namun demikian, seluruh variabel tetap digunakan karena secara teoritis memiliki pengaruh terhadap produksi padi dan dapat dimanfaatkan oleh algoritma *machine learning* untuk mempelajari hubungan yang lebih kompleks.

Untuk merepresentasikan pengaruh kondisi iklim terhadap siklus pertumbuhan tanaman padi, penelitian ini menerapkan teknik *lagging* pada seluruh variabel iklim. *Lagging* dilakukan hingga tiga periode sebelumnya, yaitu *lag-1*, *lag-2*, dan *lag-3*. Penerapan teknik ini didasarkan pada siklus pertumbuhan tanaman padi yang umumnya berlangsung sekitar tiga bulan sejak masa tanam hingga panen. Dengan adanya *lagging*, model tidak hanya memanfaatkan kondisi iklim pada saat panen, tetapi juga mempertimbangkan kondisi iklim pada fase awal pertumbuhan, fase vegetatif, dan fase pematangan tanaman. Pendekatan tersebut memungkinkan model memperoleh informasi yang lebih lengkap mengenai faktor-faktor yang memengaruhi hasil produksi padi.

Hasil penelitian menunjukkan bahwa kedua model yang digunakan, yaitu *Extreme Gradient Boosting* (XGBoost) dan *Artificial Neural Network* (ANN), mampu mempelajari pola hubungan antara variabel iklim dan produksi padi dengan cukup baik. Hal ini terlihat dari nilai koefisien determinasi ( $R^2$ ) yang diperoleh masing-masing model. Model XGBoost menghasilkan nilai  $R^2$  sebesar 0,870, sedangkan model ANN menghasilkan nilai  $R^2$  sebesar 0,824. Nilai tersebut menunjukkan bahwa kedua model mampu menjelaskan lebih dari 80% variasi data produksi padi berdasarkan variabel iklim yang digunakan. Dengan demikian, pendekatan *machine learning* yang diterapkan pada penelitian ini dapat digunakan untuk melakukan prediksi produksi padi dengan tingkat akurasi yang cukup baik.

Meskipun demikian, perbandingan performa kedua model tidak hanya dilakukan berdasarkan nilai metrik evaluasi, tetapi juga diperkuat melalui analisis statistik komparatif. Berdasarkan hasil evaluasi, model XGBoost menghasilkan nilai  $R^2$  sebesar 0,870 dengan MAE sebesar 7.997,27 ton dan RMSE sebesar 22.849,18 ton. Sementara itu, model ANN menghasilkan nilai  $R^2$  sebesar 0,824 dengan MAE sebesar 10.767,52 ton dan RMSE sebesar 26.589,44 ton. Secara deskriptif, hasil tersebut menunjukkan bahwa XGBoost memiliki kemampuan prediksi yang lebih baik dibandingkan ANN karena menghasilkan nilai koefisien determinasi yang lebih tinggi dan tingkat kesalahan prediksi yang lebih rendah.

Perbedaan tingkat kesalahan kedua model juga dapat diamati melalui nilai *Absolute Error* (AE) pada setiap bulan pengamatan. Nilai AE XGBoost dan ANN menunjukkan variasi pada setiap bulan, yang mengindikasikan bahwa tingkat kesalahan prediksi kedua model dapat berubah sesuai dengan karakteristik data yang dihadapi. Pada sebagian besar bulan, nilai AE XGBoost cenderung lebih rendah dibandingkan ANN, meskipun pada beberapa bulan tertentu ANN menghasilkan nilai AE yang lebih kecil. Pola tersebut menunjukkan bahwa XGBoost tidak selalu menghasilkan kesalahan yang lebih rendah pada setiap

pengamatan, tetapi secara keseluruhan memiliki tingkat kesalahan yang lebih kecil. Hal ini selaras dengan nilai MAE dan RMSE XGBoost yang lebih rendah dibandingkan ANN.

Hasil prediksi XGBoost yang terlihat lebih bervariasi antarperiode menunjukkan bahwa model mampu merespon perubahan pola pada data produksi padi dan variabel input. Karakteristik XGBoost yang membangun sejumlah *decision tree* secara bertahap memungkinkan model mempelajari hubungan *non-linear* serta pola yang kompleks pada data. Oleh karena itu, perubahan nilai prediksi yang cukup besar pada beberapa periode tidak secara langsung menunjukkan kelemahan model. Hal tersebut juga didukung oleh hasil evaluasi yang menunjukkan bahwa XGBoost memperoleh nilai  $R^2$  yang lebih tinggi serta nilai MAE dan RMSE yang lebih rendah dibandingkan ANN.

Selain itu, pada beberapa periode pengamatan, model ANN menghasilkan nilai prediksi di bawah nol. Kondisi tersebut dapat terjadi karena model ANN yang digunakan melalui MLPRegressor menghasilkan output regresi yang tidak dibatasi pada nilai minimum tertentu, sehingga nilai prediksi dapat berada di luar rentang produksi yang sebenarnya. Keterbatasan jumlah data setelah penerapan teknik lagging serta karakteristik pembagian data secara acak juga diduga dapat memengaruhi kemampuan model dalam mempelajari pola produksi secara optimal. Nilai prediksi tersebut tetap digunakan dalam proses evaluasi untuk mempertahankan hasil asli model dan memastikan perbandingan performa dilakukan secara objektif.

Untuk memastikan perbedaan performa kedua model tersebut signifikan secara statistik, dilakukan studi komparasi menggunakan uji statistik berpasangan terhadap nilai *Absolute Error* (AE) yang dihasilkan oleh model XGBoost dan ANN pada setiap periode pengamatan. Tahap pertama yang dilakukan adalah uji normalitas *Kolmogorov–Smirnov* terhadap selisih nilai AE antara model XGBoost dan ANN. Hasil uji normalitas disajikan pada Tabel 11.

Tabel 11 Hasil Uji Normalitas *Kolmogorov–Smirnov*

| Variabel                                              | Statistik <i>Kolmogorov–Smirnov</i> | Sig.   |
|-------------------------------------------------------|-------------------------------------|--------|
| (1)                                                   | (2)                                 | (3)    |
| Selisih <i>Absolute Error</i> (AE)<br>XGBoost dan ANN | 0,370                               | <0,001 |

Berdasarkan Tabel 11, hasil pengujian menunjukkan nilai signifikansi kurang dari 0,001 ( $p < 0,05$ ), sehingga data selisih AE tidak berdistribusi normal. Oleh karena itu, pengujian dilanjutkan menggunakan *Wilcoxon Signed-Rank Test* sebagai metode nonparametrik untuk membandingkan nilai *Absolute Error* (AE)

model XGBoost dan ANN yang berpasangan pada setiap periode pengamatan yang sama. Pengujian dilakukan untuk mengetahui apakah perbedaan tingkat kesalahan prediksi antara kedua model bersifat signifikan secara statistik. Hasil pengujian *Wilcoxon Signed-Rank Test* disajikan pada Tabel 12.

Tabel 12 Hasil Uji *Wilcoxon Signed-Rank Test*

| Statistik                     | Nilai  |
|-------------------------------|--------|
| (1)                           | (2)    |
| <i>Negative Ranks</i>         | 4      |
| <i>Positive Ranks</i>         | 53     |
| <i>Ties</i>                   | 0      |
| Total                         | 57     |
| Z                             | -5,152 |
| <i>Asymp. Sig. (2-tailed)</i> | <0,001 |

Hipotesis yang digunakan dalam pengujian ini adalah sebagai berikut:

$H_0$  = Tidak terdapat perbedaan yang signifikan antara nilai *Absolute Error* model XGBoost dan ANN

$H_1$  = Terdapat perbedaan yang signifikan antara nilai *Absolute Error* model XGBoost dan ANN

Hasil *Wilcoxon Signed-Rank Test* menunjukkan nilai statistik Z sebesar -5,152 dengan nilai *Asymp. Sig. (2-tailed)* kurang dari 0,001. Nilai tersebut lebih kecil dari taraf signifikansi 0,05 sehingga  $H_0$  ditolak dan  $H_1$  diterima. Dengan demikian, terdapat perbedaan yang signifikan antara nilai *Absolute Error* (AE) model XGBoost dan ANN. Selain itu, hasil peringkat (*Ranks*) menunjukkan bahwa terdapat 53 data yang termasuk dalam *positive ranks* dan 4 data yang termasuk dalam *negative ranks*. *Positive ranks* menunjukkan bahwa nilai AE model ANN lebih besar daripada nilai AE XGBoost, sedangkan *negative ranks* menunjukkan kondisi sebaliknya. Hasil tersebut mengindikasikan bahwa model XGBoost menghasilkan tingkat kesalahan prediksi yang lebih rendah pada sebagian besar periode pengamatan.

Performa XGBoost yang lebih baik diduga dipengaruhi oleh karakteristik algoritma tersebut yang mampu menangani hubungan *nonlinier* antarvariabel secara efektif melalui mekanisme *boosting*. Algoritma ini membangun sejumlah pohon keputusan secara bertahap untuk memperbaiki kesalahan prediksi pada proses sebelumnya sehingga mampu menghasilkan model yang lebih akurat. Selain itu, XGBoost juga memiliki mekanisme regularisasi yang membantu mengurangi

risiko *overfitting* dan pada penelitian ini menunjukkan kemampuan yang baik dalam memodelkan dataset yang relatif terbatas. Kondisi tersebut sesuai dengan penelitian ini yang menggunakan data bulanan selama periode 2020–2024. Sebaliknya, ANN umumnya membutuhkan jumlah data yang lebih besar agar proses pembelajaran jaringan saraf dapat berlangsung secara optimal. Keterbatasan jumlah data yang digunakan diduga menjadi salah satu faktor yang menyebabkan performa ANN masih berada di bawah XGBoost.

## BAB V KESIMPULAN

### A. Kesimpulan

Berdasarkan hasil penelitian yang telah dilakukan, dapat ditarik kesimpulan sebagai berikut:

1. Algoritma *Extreme Gradient Boosting* (XGBoost) memiliki performa yang lebih baik dibandingkan *Artificial Neural Network* (ANN) dalam memprediksi produksi padi berdasarkan variabel-variabel lama penyinaran matahari, curah hujan, Oceanic Niño Index (ONI), suhu udara, dan kelembaban udara pada dataset produksi padi Kabupaten Cilacap periode 2020-2024. Model XGBoost memperoleh nilai  $R^2$  sebesar 0,870, MAE sebesar 7.997,27 ton, dan RMSE sebesar 22.849,18 ton. Sementara itu, model ANN menghasilkan nilai  $R^2$  sebesar 0,824, MAE sebesar 10.767,52 ton, dan RMSE sebesar 26.589,44 ton. Hasil uji *Wilcoxon Signed-Rank Test* juga menunjukkan nilai signifikansi kurang dari 0,001 ( $p < 0,05$ ), sehingga terdapat perbedaan tingkat kesalahan prediksi yang signifikan antara kedua model. Berdasarkan hasil evaluasi tersebut, XGBoost dipilih sebagai model dengan performa terbaik pada dataset penelitian ini.
2. XGBoost berhasil diimplementasikan ke dalam sistem prediksi produksi padi berbasis web menggunakan metode *prototyping* dengan arsitektur monolitik. Sistem dibangun menggunakan FastAPI sebagai *backend*, HTML, CSS, dan *JavaScript* sebagai antarmuka pengguna, serta Supabase sebagai database. Sistem yang dikembangkan menyediakan fitur prediksi produksi padi, simulasi prediksi, pengelolaan data produksi, serta penyajian hasil prediksi berupa nilai produksi, kategori produksi, dan rekomendasi. Berdasarkan hasil pengujian *black box testing*, seluruh fitur dapat berjalan sesuai dengan kebutuhan fungsional yang telah ditetapkan. Dengan demikian, sistem dapat digunakan sebagai alat bantu dalam melakukan prediksi produksi padi berdasarkan data iklim pada dataset penelitian ini.

### B. Saran

Berdasarkan hasil penelitian yang telah dilakukan, saran yang dapat diberikan adalah sebagai berikut:

1. Penelitian ini masih terbatas pada cakupan periode data tahun 2020–2024 dan penggunaan variabel iklim sebagai variabel input. Oleh karena itu, pengembang sistem prediksi produksi padi dapat menggunakan data dengan periode yang lebih panjang serta menambahkan variabel lain yang berkaitan dengan produksi padi, seperti luas panen, penggunaan pupuk, dan kondisi irigasi. Penambahan data dan variabel tersebut diharapkan dapat memberikan

informasi yang lebih lengkap, mendukung proses prediksi, serta meningkatkan kinerja model dan pengembangan sistem selanjutnya.

2. Bagi peneliti selanjutnya, disarankan untuk membandingkan Algoritma XGBoost dengan Algoritma lain yang memiliki karakteristik dan pendekatan yang sejenis, seperti *Light Gradient Boosting Machine* (LightGBM) atau *Categorical Boosting* (CatBoost). Perbandingan tersebut dapat memberikan hasil evaluasi yang lebih terarah mengenai performa algoritma berbasis *gradient boosting* pada prediksi produksi padi.

## DAFTAR PUSTAKA

- [1] T. W. Damayanti, S. Isma, and S. A. Mardila, “Strategi Terpadu Peningkatan Produksi Padi di Indonesia : Suatu Analisis Literatur Komprehensif,” *J. Ilmu Pertan. Dan Teknol. Dalam Ilmu Tanam.*, vol. 2, no. November, pp. 01–12, 2025, doi: <https://doi.org/10.62951/hidroponik.v2i3.503>.
- [2] R. Rahim, A. Dela, R. Nurfalah, Y. Anggraeni, and S. Pasaribu, “Dinamika Ketahanan Pangan: Analisis Pengaruh Luas Panen Padi, Konsumsi Beras, Harga Beras, dan Jumlah Penduduk terhadap Produksi Padi di Wilayah Sentra Padi di Indonesia Tahun 2017-2021,” *J. Soc. Sci. Res.*, vol. 4, pp. 17083–17093, 2024.
- [3] M. Y. Dawud and W. Abryandoko, “Pengaruh Fluktuasi Unsur Iklim Tahunan terhadap Produksi Padi : Studi Empiris berdasarkan Data Historis,” *J. Agrik.*, vol. 36, no. 2, pp. 216–227, 2025.
- [4] A. I. Widiawati, N. Vania, A. Harini, D. Musytari, I. Irpawa, and N. Rizki, “Dampak Anomali Iklim terhadap Produksi Padi di Kecamatan Seputih Mataram Kabupaten Lampung Tengah,” *J. Agric. Anim. Sci.*, vol. 5, no. 1, pp. 79–88, 2025, doi: <https://doi.org/10.47637/agrimals.v5i1.1734>.
- [5] S. Aziz, Sudrajat, and B. Setia, “Strategi Adaptasi Perubahan Iklim (Upaya Mempertahankan Produksi Padi di Kabupaten Ciamis),” *J. Konserv. dan Budaya*, vol. 1, no. 1, pp. 74–82, 2024.
- [6] D. Prafi and K. Manokwari, “Dampak Perubahan Iklim terhadap Usaha Tani Padi : Studi Adaptasi Petani Kampung Prafi,” *J. Sustain. Agric. Ext.*, vol. 3, no. 2, pp. 107–119, 2025, doi: <https://doi.org/10.47687/JoSAE.v3i2.1646>.
- [7] S. L. Tefa, M. Samin, and M. H. Hasan, “Strategi Adaptasi Petani Padi Ladang terhadap Perubahan Musim (Suatu Studi pada Komunitas Petani Padi Ladang di Kelurahan Oesao Kecamatan Kupang Timur Kabupaten Kupang),” *J. Geogr.*, vol. 21, no. 1, pp. 14–22, 2025.
- [8] L. M. Harahap, Y. I. B. Manurung, J. B. Situngkir, and N. A. Simanungkalit, “Pengelolaan Risiko Iklim dalam Sektor Pertanian: Strategi dan Implementasi,” *J. Ilmu Manajemen, Bisnis dan Ekon.*, vol. 1, no. 5, pp. 117–125, 2024.
- [9] A. K. Walya, H. R. Sulisty, I. A. Pratama, and S. Akmal, “Analisa Prediksi Kelulusan Mahasiswa menggunakan Metode *Machine Learning*,” *J. Pengabd. Masy. dan Ris. Pendidik.*, vol. 4, no. 3, pp. 16576–16579, 2026, doi: <https://doi.org/10.31004/jerkin.v4i3.4959>.
- [10] C. Prianto, R. Angelina, and P. Hutabarat, “Penerapan Algoritma *Machine Learning* untuk Prediksi Gangguan Kesehatan Mental: *Systematic Literature Review*,” *J. Sist. dan Teknol. Inf.*, vol. 13, no. 4, pp. 510–518, 2025, doi: [10.26418/justin.v13i4.95911](https://doi.org/10.26418/justin.v13i4.95911).
- [11] L. Nur, I. Surala, A. L. Hananto, E. Novalia, and F. Nurapriani, “Analisis

Status Pembayaran Group Order NooBlue menggunakan Algoritma XGBoost,” *J. Sist. Inf. TGD*, vol. 4, no. 4, pp. 838–847, 2025.

- [12] M. Apriani, E. Satira, D. W. Sari, J. Wati, M. Afriyanti, and S. Martadinata, “Sistematikasi Penerapan *Machine Learning* dalam Prediksi Risiko Kebangkrutan Perusahaan : *Systematic Literature Review*,” *J. Sains Student Res.*, vol. 4, no. 1, pp. 41–55, 2026, doi: <https://doi.org/10.61722/jssr.v4i1.7281>.
- [13] I. Hapsari and S. P. Wisesa, “Evaluasi Model Prediksi Curah Hujan berbasis *Machine Learning* di Kota Bandung,” *J. Nas. Teknol. dan Sist. Inf.*, vol. 11, no. 2, pp. 136–143, 2025, doi: <https://doi.org/10.25077/TEKNOSI.v11i2.2025.136-143>.
- [14] S. F. Prasetyo, T. F. Efendi, and Muqorobin, “Implementasi Sistem Prediksi Curah Hujan dengan Penerapan Jaringan Syaraf Tiruan Berbasis Website,” *J. Ris. Tek. Komput.*, vol. 1, no. 2, pp. 80–96, 2024, doi: <https://doi.org/10.69714/gn6nym06>.
- [15] W. Su *et al.*, “*An XGBoost - Based Knowledge Tracing Model*,” *Int. J. Comput. Intell. Syst.*, vol. 16, no. 13, pp. 1–9, 2023, doi: <https://doi.org/10.1007/s44196-023-00192-y>.
- [16] P. M. Izzati and Fitriyani, “Implementasi Algoritma XGBoost Untuk Prediksi Capaian Bulanan Pendapatan Daerah Kota Bandung,” *J. Comput. Sci. Inf. Technol.*, vol. 6, no. 2, pp. 104–111, 2025, doi: <https://doi.org/10.37859/coscitech.v6i2.9578>.
- [17] S. N. Ruscikasani, R. Roro, N. Oktalivia, F. R. Putra, and A. J. Wahidin, “Prediksi Pembelian *E-Commerce* menggunakan XGBoost berbasis Perilaku Sesi Pengguna,” *J. Artif. Intell. Digit. Bus.*, vol. 4, no. 4, pp. 5666–5672, 2025, doi: <https://doi.org/10.31004/riggs.v4i4.4287>.
- [18] D. M. Puspita, K. I. Santoso, A. Triyono, E. Supriyadi, and S. Agus, “Algoritma *Random Forest*, *Decision Tree*, dan XGboost untuk Klasifikasi Stunting pada Balita,” *J. Transform.*, vol. 23, no. 1, pp. 67–76, 2025, doi: <https://doi.org/10.26623/transformatika.v23i1.12202>.
- [19] R. Illaihiyah, G. Abdurrahman, and Daryanto, “Implementasi Algoritma XGBoost *Classifier* untuk Klasifikasi Kerumunan Jemaah Haji dan Umrah,” *Indones. J. Mach. Learn. Comput. Sci.*, vol. 6, no. 1, pp. 106–115, 2026, doi: <https://doi.org/10.57152/malcom.v6i1.2245>.
- [20] F. Ananda, K. Hidayat, N. Sulistianingsih, and R. Hammad, “Perbandingan Algoritma XGBoost dan *Random Forest* dalam Klasifikasi Surat Masuk Pemerintahan,” *J. Artif. Intell. Digit. Bus.*, vol. 4, no. 4, pp. 10390–10397, 2026, doi: <https://doi.org/10.31004/riggs.v4i4.5044>.
- [21] S. R. Anugrahaini, M. Fatchan, and T. N. Wiyatno, “Model Prediksi Tingkat Stres di Kalangan Siswa dengan Pendekatan Algoritma *Artificial Neural Network*,” *J. Mhs. Tek. Inform.*, vol. 9, no. 3, pp. 4742–4748, 2025.
- [22] R. Pratama, R. Herdiana, R. Hamonangan, and S. Anwar, “Analisis Prediksi

Kelulusan Mahasiswa menggunakan Metode *Artificial Neural Network*,” vol. 8, no. 1, pp. 687–693, 2024.

- [23] J. Veronika and Andri, “Penerapan Metode Algoritma *Neural Network* untuk Memprediksi Penjualan Bahan Bakar Minyak,” *J. Inf. Technol. Ampera*, vol. 3, no. 2, pp. 235–243, 2022.
- [24] R. Setiabudi, F. S. Utomo, and I. Tahyudin, “Sistem Klasifikasi Perintah Suara menggunakan *Artificial Network* pada Aplikasi G-MOOC 4D,” *J. Informatics Interact. Technol.*, vol. 1, no. 1, pp. 19–28, 2024.
- [25] S. Arya, Anju, and N. A. Ramli, “*Predicting the Stress Level of Students using Supervised Machine Learning and Artificial Neural Network (ANN)*,” *Indian J. Eng.*, vol. 1684, pp. 1–24, 2024.
- [26] T. B. Harahap, L. S. Harahap, and N. N. Hasibuan, “Penerapan Jaringan Saraf Tiruan dengan Metode *Backpropagation* untuk Memprediksi Curah Hujan di Kota Medan,” *J. Ilmu Komput. dan Ilmu Pengetah. Alam*, vol. 4, no. 1, pp. 87–100, 2026, doi: <https://doi.org/10.62383/polygon.v4i1.934>.
- [27] I. Efendi, S. Khairunnisa, P. Studi, and M. Informatika, “Perancangan dan Implementasi Website Informasi Nagari Koto Tangah,” *J. Sains Student Res.*, vol. 3, no. 3, pp. 832–850, 2025, doi: <https://doi.org/10.61722/jssr.v3i5.6077>.
- [28] S. Sudirman, A. Fauzan, and R. A. W. Mustakim, “Pembuatan Website sebagai Media Pencitraan dan Promosi pada Desa Kamiri Kec. Balusu Kab. Barru,” *Ilmu Komput. Untuk Masy.*, vol. 4, no. 1, pp. 1–8, 2023.
- [29] D. Hamdani and B. R. Aditya, “Perancangan Aplikasi Berbasis Website untuk Edukasi dan Pengelolaan Pertanian Biofarmaka menggunakan Metode FAST (*Framework for the Application System Thinking*),” *J. Teknol. Inf. dan Ilmu Komput.*, vol. 12, no. 3, pp. 505–516, 2025.
- [30] B. P. Statistik, “Luas Panen dan Produksi Padi di Kabupaten Cilacap,” vol. 3, Cilacap, 2025.
- [31] BMKG Stasiun Pemantau Atmosfer Global (SPAG) Lore Lindu Bariri, “Iklim.” [Online]. Available: <https://gaw-bariri.bmkg.go.id/index.php/karya-tulis-dan-artikel/gawsarium/228-iklim>
- [32] M. Astuti, “Hubungan Faktor Iklim terhadap Kejadian Demam Berdarah Dengue (DBD) di Kabupaten Lampung Barat Tahun 2022-2024,” Politeknik Kesehatan Kemenkes Tanjung Karang, 2025.
- [33] N. A. P. Indaryono, R. R. Saedudin, and F. Hamami, “Analisa Perbandingan Algoritma *Random Forest* dan *Naïve Bayes* untuk Klasifikasi Curah Hujan berdasarkan Iklim di Indonesia,” *J. Ilm. Penelit. dan Pembelajaran Inform.*, vol. 9, no. 1, pp. 158–167, 2024, doi: <https://doi.org/10.29100/jipi.v9i1.4421>.
- [34] M. R. F. Daulay *et al.*, “Pemetaan Curah Hujan untuk Tanaman Padi di Kabupaten Serdang Bedagai,” *J. Pemberdaya. Masy.*, vol. 12, no. 2, pp. 237–246, 2025, doi: <http://dx.doi.org/10.37064/jpm.v12i2.22670>.

- [35] K. L. Himayanta and D. F. Wardhani, "Prediksi Hasil Panen Padi dengan *Machine Learning*," *J. Kelitbangan*, vol. 13, no. 1, pp. 1–14, 2023.
- [36] S. E. GIRLANG, "Dampak Perubahan Iklim terhadap Produksi dan Produktivitas Tanaman Padi (*Oryza sativa L.*) di Dataran Rendah dan Tinggi Kabupaten Deli Serdang, Sumatera Utara," Universitas Islam Sumatera Utara, 2024.
- [37] R. P. Ramadhani and Sudirman, "Evaluasi Pola dan Tren Curah Hujan Harian dalam Hubungannya dengan Fenomena ENSO (*El Niño–Southern Oscillation*) di Wilayah Papua, Indonesia," *J. Ilm. Multi Sci.*, vol. 18, no. 1, pp. 30–39, 2026, doi: <https://doi.org/10.30599/031fvn57>.
- [38] O. O. Palit *et al.*, "Akurasi Metode Mesin Pembelajaran Dalam Analisis Variabel Penting Faktor Risiko Sindrom Down," *Indones. J. Comput. Sci.*, vol. 13, no. 5, pp. 8001–8016, 2024, doi: <https://doi.org/10.33022/ijcs.v13i5.4354>.
- [39] G. A. Syafarina and M. Amin, "Perbandingan Tree-Based Ensemble (*Random Forest*, *XGBoost*, *LightGBM*) dan *Deep Learning* untuk Prediksi Klik Iklan Online," *J. Technol.*, vol. 17, no. 1, pp. 257–265, 2026, doi: <http://dx.doi.org/10.31602/tji.v17i1.21627>.
- [40] M. R. P. Putra, S. Juwariyah, M. Ridwan, and R. Marco, "Optimasi Prediksi Kelayakan Pinjaman dengan Teknik *Resampling* dan Algoritma *Boosting*," *J. Sist. Komput.*, vol. 14, no. 1, pp. 41–51, 2025, doi: [10.34010/komputika.v14i2.15485](https://doi.org/10.34010/komputika.v14i2.15485).
- [41] D. Fudhlalina and F. Budiman, "Peningkatan Akurasi Prediksi Curah Hujan menggunakan *Gradient Boosting* dan CatBoost dengan Pendekatan *Voting Classifier*," *J. Pendidik. Inform.*, vol. 9, no. 1, pp. 51–59, 2025, doi: [10.29408/edumatic.v9i1.28988](https://doi.org/10.29408/edumatic.v9i1.28988).
- [42] F. F. Amatullah, H. U. My, T. A. Rizqi, S. T. Wahyuni, Bagus Sartono, and A. R. Firdawanti, "Studi Komparatif Metode *Boosting* Dalam Pengklasifikasian Penerima Bantuan Program Keluarga Harapan ( PKH )," *J. Telekomun. Elektron. Komputasi, dan Kontrol*, vol. 11, no. 3, pp. 315–326, 2025, doi: <https://doi.org/10.15575/telka.v11n3.315-326>.
- [43] A. A. Saputra, B. N. Sari, and C. Rozikin, "Penerapan Algoritma *Extreme Gradient Boosting* (XGboost) untuk Analisis Risiko Kredit," *J. Ilm. Wahana Pendidik.*, vol. 10, no. 7, pp. 27–36, 2024, doi: <https://doi.org/10.5281/zenodo.10960080>.
- [44] W. Kurniawan and U. Indahyanti, "Prediksi Angka Harapan Hidup Penduduk menggunakan Metode XGBoost," *Indones. J. Appl. Technol.*, vol. 1, no. 2, pp. 1–18, 2024, doi: <https://doi.org/10.47134/ijat.v1i2.3045>.
- [45] P. S. Rizky, R. H. Hirzi, and U. Hidayaturrohmah, "Perbandingan Metode LightGBM dan XGBoost dalam Menangani Data dengan Kelas Tidak Seimbang," *J. Stat.*, vol. 15, no. 2, pp. 228–236, 2022.
- [46] S. Elina, H. Yulianti, O. Soesanto, and Y. Sukmawaty, "Penerapan Metode

*Extreme Gradient Boosting (XGBoost) pada Klasifikasi Nasabah Kartu Kredit,” J. Math. Theory Appl.*, vol. 4, no. 1, pp. 21–26, 2022.

- [47] A. Pangestu and S. Mujiyono, “Implementasi Algoritma XGBoost untuk Prediksi Status Gizi Balita Berbasis Website,” *J. Algoritma.*, vol. 22, no. 2, pp. 176–187, 2025, doi: 10.33364/algoritma/v.22-2.2390.
- [48] F. Rohman and M. S. Al Amin, “Prediksi Beban Listrik dengan menggunakan Jaringan Syaraf Tiruan Metode *Backpropagation*,” *J. Surta Energy*, vol. 5, no. 2, pp. 55–60, 2022, doi: <https://doi.org/10.32502/jse.v5i2.3092>.
- [49] S. E. Wahyuni, R. Buaton, and S. Ramadani, “Jaringan Saraf Tiruan ( JST ) Memprediksi Penjualan UMKM Kota Binjai dengan menggunakan Metode *Backpropagation*,” *J. Publ. Sist. Inf. dan Telekomun.*, vol. 2, no. 4, pp. 47–59, 2024, doi: <https://doi.org/10.62951/bridge.v2i4.214>.
- [50] J. J. J. Manajemen, I. Komputer, and N. W. Putri, “Optimasi Jaringan Syaraf Tiruan Metode *Backpropagation* dengan Algoritma Genetik untuk Prediksi Tingkat Pengangguran di Provinsi Sumatera Utara,” *J. Manajemen, Pendidik. dan Ilmu Komput.*, vol. 2, no. 1, pp. 29–36, 2025.
- [51] M. A. Dinata, H. Yenni, W. A. A, and Guston, “Prediksi Jumlah Titik Ruang Terbuka Hijau (RTH) menggunakan Regresi Linier dan Model *Random Forest*,” *J. Ilm. Besemah Teknol. Inf. Komput.*, vol. 16, no. 2, pp. 144–156, 2025.
- [52] R. H. Cokro, D. A. Ahmad, R. A. Sanchia, Trimono, and M. Nasrudin, “Segmentasi Provinsi Penghasil Padi di Indonesia Berdasarkan Karakteristik Produksi Produksi Tahun 2024 menggunakan *Clustering K-Means*,” *J. Multidisiplin Saintek*, vol. 8, no. 10, pp. 1–9, 2025, doi: <https://doi.org/10.2238/5jv9bq37>.
- [53] M. N. Yullah, “Analisis Spasial Kasus Persebaran Tuberkolosis Kabupaten dan Kota di Provinsi Jawa Barat Tahun 2022-2024,” *J. Tek. Inform. dan Terap.*, vol. 3, no. 4, pp. 106–119, 2025, doi: <https://doi.org/10.62951/router.v3i4.748>.
- [54] R. N. Nugraha *et al.*, “Implementasi Prediksi Stok Barang Menggunakan Algoritma *Support Vector Regression ( SVR )* pada *Marketplace* UMKM Berbasis *Microservices*,” *Informatics Comput. Eng. J.*, vol. 6, no. 1, pp. 16–25, 2026, doi: <https://doi.org/10.31294/icej.v6i1.11508>.
- [55] M. Syarif and D. Risdiansyah, “Pemanfaatan Metode *Prototype* dalam Perancangan Sistem Informasi Penjualan Berbasis Website,” *J. Mhs. Tek. Inform.*, vol. 8, no. 4, pp. 7945–7952, 2024.
- [56] I. P. A. S. Utama, I. N. Y. A. Wijaya, and A. A. G. A. M. Putra, “Rancang Bangun Sistem Informasi Manajemen Berbasis Website di PT Bali Tresna Cemerlang dengan Metode *Prototype*,” *J. Sos. dan Teknol.*, vol. 4, no. 8, pp. 546–565, 2024.
- [57] M. A. M. A. Rachman *et al.*, “Penerapan *Black Box Testing* untuk Evaluasi

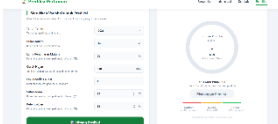
Fungsionalitas Website Maggoplast,” *J. Mhs. Tek. Inform.*, vol. 9, no. 1, pp. 169–176, 2025.

- [58] F. Al Fatih, N. P. Lubis, K. N. Hsb, Zulpan, and Arianto, “Konsep Homogenitas dan Normalitas dalam Statistik serta Teknik Pengujiannya,” *Educ. J.*, vol. 1, no. 3, pp. 805–817, 2026, doi: <https://doi.org/10.63822/b98bqe88>.
- [59] D. A. Saputra and A. Gunasti, “Analisis Uji Beda Berpasangan Antara Mobil,” *J. Media Akad.*, vol. 4, no. 1, pp. 1–9, 2026, doi: <https://doi.org/10.62281/etfwxp25>.
- [60] C. J. Chandra, C. F. Hariyadi, N. Aprilian, and L. J. J. Laipiopa, “Prediksi Kecelakaan Lalu Lintas Polandia dengan XGBoost, CatBoost dan *Random Forest*,” *J. Ilmu Komput. dan Sist. Inf.*, vol. 14, no. 1, pp. 1–9, 2026, doi: <https://doi.org/10.24912/csmqwt31>.

## Lampiran 1 *BlackBox Testing*

### A. Pengujian User

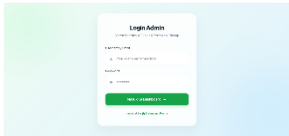
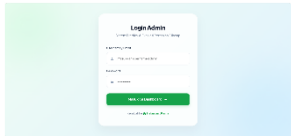
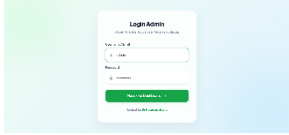
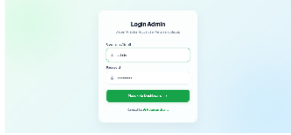
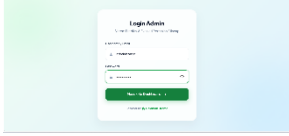
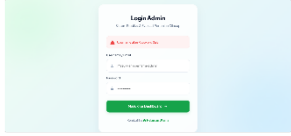
Tabel 13 Hasil Pengujian User

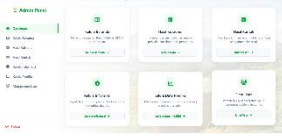
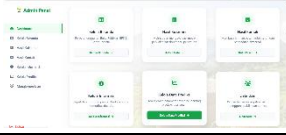
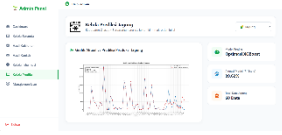
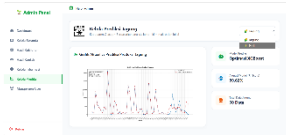
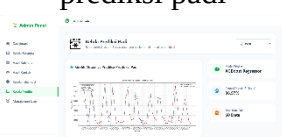
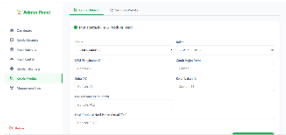
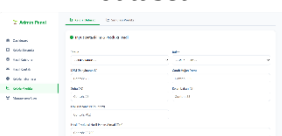
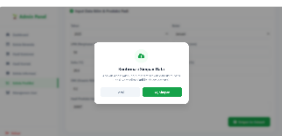
| Skenario Pengujian           | Langkah Pengujian                                                                                                                          | Hasil yang diharapkan                                                    | Hasil Pengujian |
|------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------|-----------------|
| (1)                          | (2)                                                                                                                                        | (3)                                                                      | (4)             |
| Akses beranda                | Mengakses halaman website<br>                             | Menampilkan halaman beranda                                              | Berhasil        |
| Akses menu prediksi          | Klik menu “Prediksi”<br>                                 | Menampilkan pilihan komoditas prediksi                                   | Berhasil        |
| Akses prediksi padi          | Klik komoditas “Padi”<br>                               | Menampilkan halaman prediksi padi                                        | Berhasil        |
| Input variabel prediksi      | Menginputkan variabel prediksi<br>                      | Data variabel prediksi dapat diinput                                     | Berhasil        |
| Validasi input tidak lengkap | Klik tombol “Hitung Prediksi” tanpa melengkapi data<br> | Sistem menampilkan pesan error                                           | Berhasil        |
| Proses prediksi              | Klik tombol “Hitung Prediksi” dengan data lengkap<br>   | Sistem memproses data dan menampilkan hasil prediksi beserta kategorinya | Berhasil        |

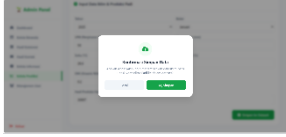
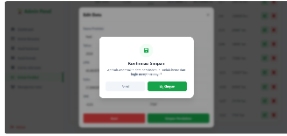
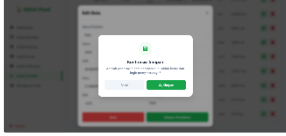
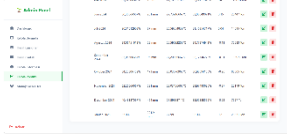
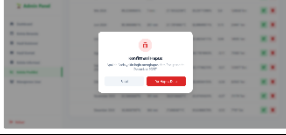
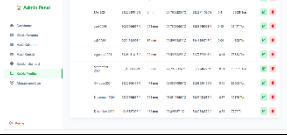
| Skenario Pengujian   | Langkah Pengujian         | Hasil yang diharapkan                                                                                                                   | Hasil Pengujian |
|----------------------|---------------------------|-----------------------------------------------------------------------------------------------------------------------------------------|-----------------|
| (1)                  | (2)                       | (3)                                                                                                                                     | (4)             |
|                      |                           |                                                       |                 |
| Tampilan rekomendasi | Setelah prediksi berhasil | Sistem menampilkan rekomendasi sesuai kategorinya<br> | Berhasil        |

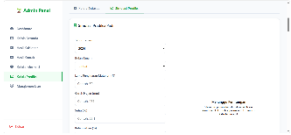
## B. Pengujian Admin

Tabel 14 Hasil Pengujian Admin

| Skenario Pengujian                        | Langkah Pengujian                                                                                                                                     | Hasil yang diharapkan                                                                                                                     | Hasil Pengujian |
|-------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------|-----------------|
| (1)                                       | (2)                                                                                                                                                   | (3)                                                                                                                                       | (4)             |
| Login Admin                               | Masuk ke halaman login admin<br>                                   | Menampilkan halaman login admin<br>                   | Berhasil        |
| Input <i>username</i> dan <i>password</i> | Mengisi kolom <i>username</i> dan <i>password</i><br>              | <i>Username</i> dan <i>password</i> dapat diinput<br> | Berhasil        |
| Validasi salah inputan                    | Klik tombol “Masuk ke <i>Dashbord</i> ” dengan data yang salah<br> | Sistem menampilkan pesan eror<br>                     | Berhasil        |
| Akses <i>dashboard</i> admin              | Setelah login berhasil                                                                                                                                | Menampilkan <i>dashboard</i> admin                                                                                                        | Berhasil        |

| Skenario Pengujian          | Langkah Pengujian                                                                                                                                              | Hasil yang diharapkan                                                                                                                                                   | Hasil Pengujian |
|-----------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------|
| (1)                         | (2)                                                                                                                                                            | (3)                                                                                                                                                                     | (4)             |
|                             |                                                                                                                                                                |                                                                                       |                 |
| Akses kelola prediksi       | Klik menu “Kelola Prediksi”<br>                                               | Menampilkan halaman kelola prediksi<br>                                               | Berhasil        |
| Akses kelola prediksi padi  | Klik pilihan kelola prediksi padi<br>                                        | Menampilkan halaman kelola prediksi padi<br>                                         | Berhasil        |
| Akses kelola dataset        | Klik pilihan “Kelola Dataset”<br>                                           | Menampilkan halaman kelola dataset<br>                                              | Berhasil        |
| Input data baru             | Mengisi kolom tahun, bulan, prediksi, variabel iklim, dan produksi padi<br> | Data tahun, bulan, prediksi, variabel iklim, dan produksi padi dapat diinputkan<br> | Berhasil        |
| Konfirmasi simpan data baru | Klik button “Simpan ke Dataset”<br>                                         | Sistem menampilkan konfirmasi<br>                                                   | Berhasil        |
| Simpan dataset baru         | Klik button “Ya Simpan”                                                                                                                                        | Data tersimpan di database dan riwayat dataset terdaftar                                                                                                                | Berhasil        |

| Skenario Pengujian   | Langkah Pengujian                                                                                                                   | Hasil yang diharapkan                                                                                                                             | Hasil Pengujian |
|----------------------|-------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------|-----------------|
| (1)                  | (2)                                                                                                                                 | (3)                                                                                                                                               | (4)             |
|                      |                                                    |                                                                 |                 |
| Akses Edit data      | Pilih salah satu data lalu klik tombol edit<br>    | Menampilkan data yang akan diedit<br>                           | Berhasil        |
| Melakukan edit data  | Mengganti data yang ada dengan data baru<br>       | Data dapat diubah ke data baru<br>                              | Berhasil        |
| Konfirmasi edit data | Klik button “Simpan Perubahan”<br>               | Sistem menampilkan konfirmasi<br>                             | Berhasil        |
| Simpan edit data     | Klik button “Ya Simpan”<br>                      | Data tersimpan di database dan riwayat dataset terdaftar<br>  | Berhasil        |
| Akses hapus data     | Pilih salah satu data lalu klik tombol hapus<br> | Sistem menampilkan konfirmasi<br>                             | Berhasil        |
| Konfirmasi edit data | Klik “Ya Hapus Data”<br>                         | Data terhapus dari database dan riwayat dataset terdaftar<br> | Berhasil        |

| Skenario Pengujian                        | Langkah Pengujian                                                                                                                         | Hasil yang diharapkan                                                                                                                                                                         | Hasil Pengujian |
|-------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------|
| (1)                                       | (2)                                                                                                                                       | (3)                                                                                                                                                                                           | (4)             |
| Akses simulasi prediksi                   | Klik menu “Simulasi Prediksi”<br>                        | Menampilkan halaman simulasi prediksi<br>                                                                   | Berhasil        |
| Input variabel prediksi                   | Mengisi variabel prediksi<br>                            | Data variabel prediksi dapat diinput<br>                                                                    | Berhasil        |
| Validasi input tidak lengkap              | Klik tombol “Hitung Prediksi” tanpa melengkapi data<br> | Sistem menampilkan pesan eror<br>                                                                          | Berhasil        |
| Proses prediksi                           | Klik tombol “Hitung Prediksi” dengan data lengkap<br>  | Sistem memproses data dan menampilkan hasil prediksi<br>                                                  | Berhasil        |
| Tampilan hasil, kategori, dan rekomendasi | Setelah prediksi berhasil                                                                                                                 | Sistem menampilkan estimasi bulan panen, prediksi panen dalam ton, kategori prediksi, dan rekomendasi<br> | Berhasil        |

## Lampiran 2 Source Code Komparasi Algoritma

### A. Algoritma XGBoost

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import os

from xgboost import XGBRegressor
from sklearn.model_selection import train_test_split
from sklearn.metrics import r2_score, mean_absolute_error,
mean_squared_error

# =====
# 1. LOAD DATA
# =====
df = pd.read_excel("data_padi.xlsx")

# =====
# 2. PROSES LAGGING (SEMUA VARIABEL IKLIM)
# =====
kolom_fitur = ['LPM (%)', 'CH (mm)', 'ONI', 'Suhu (°C)', 'Kelembaban (%)']

for col in kolom_fitur:
    df[f'{col}_Lag1'] = df[col].shift(1)
    df[f'{col}_Lag2'] = df[col].shift(2)
    df[f'{col}_Lag3'] = df[col].shift(3)

df_clean = df.dropna().reset_index(drop=True)

# =====
# 3. FITUR (X) & TARGET (y)
# =====
features = [col for col in df_clean.columns if 'Lag' in col]
X = df_clean[features]
y = df_clean['P (ton)']

# =====
# 4. SPLIT DATA (RANDOM APPROACH)
# =====
X_train, X_test, y_train, y_test = train_test_split(
    X, y,
    test_size=0.2,
    random_state=42
)

# =====
# 5. MODEL XGBOOST
```

```

# =====
xgb = XGBRegressor(
    n_estimators=100, #Jumlah pohon keputusan
    learning_rate=0.5, #Kecepatan belajar
    max_depth=5, #kedalaman maksimum pohon
    subsample=1, #Data yg digunakan untuk membangun setiap pohon
    colsample_bytree=1, #Fitur untuk membangun setiap pohon
    random_state=42 #Agar hasil prediksi konsisten setiap dijalankan
)

xgb.fit(X_train, y_train)

# =====
# 6. PREDIKSI SELURUH DATA (57 BARIS)
# =====
y_all_pred = xgb.predict(X)

# =====
# 7. EVALUASI (SELURUH DATA)
# =====
r2_all = r2_score(y, y_all_pred)
mae_all = mean_absolute_error(y, y_all_pred)
rmse_all = np.sqrt(mean_squared_error(y, y_all_pred))

# =====
# 8. VISUALISASI GRAFIK 57 BULAN
# =====
date_range = pd.date_range(start='2020-04-01', periods=len(y),
                             freq='MS')
date_labels = [d.strftime('%b-%y') for d in date_range]

plt.figure(figsize=(16, 8))

plt.plot(date_labels, y.values,
         label='Data Aktual (Asli)',
         color='#1f77b4',
         marker='o',
         linewidth=2)

plt.plot(date_labels, y_all_pred,
         label='Prediksi XGBoost',
         color='#d62728',
         linestyle='--',
         marker='s',
         alpha=0.8)

info_box = (f"Statistik Model (57 Bulan):\n"
            f"R2 Score: {round(r2_all, 3)}\n"
            f"MAE: {round(mae_all, 2)} Ton\n"
            f"RMSE: {round(rmse_all, 2)} Ton")

```

```

plt.gca().text(0.02, 0.95,
               info_box,
               transform=plt.gca().transAxes,
               fontsize=12,
               verticalalignment='top',
               bbox=dict(boxstyle='round', facecolor='white',
alpha=0.9))

plt.title('Grafik Prediksi Produksi Padi (Random Split 80:20 -
XGBoost)',
         fontsize=16, pad=20)

plt.xlabel('Periode Waktu (Bulan-Tahun)', fontsize=12)
plt.ylabel('Produksi Padi (Ton)', fontsize=12)
plt.xticks(rotation=45, fontsize=9)
plt.legend(loc='upper right', fontsize=11)
plt.grid(True, linestyle=':', alpha=0.6)

plt.tight_layout()

# Pastikan folder grafik tersedia sebelum menyimpan file gambar
if not os.path.exists("grafik"):
    os.makedirs("grafik")

# Menyimpan grafik aktual vs prediksi XGBoost
plt.savefig("grafik/xg_random80.png", dpi=300, bbox_inches='tight')
print("✓ Grafik dengan pendekatan Random Split 80:20 berhasil
disimpan di folder /grafik!")

plt.show()
plt.close()

# =====
# 9. OUTPUT HASIL DI TERMINAL
# =====
print("\n" + "="*40)
print(f"HASIL AKHIR R²: {r2_all:.3f}")
print(f"HASIL AKHIR MAE: {mae_all:.2f} Ton")
print(f"HASIL AKHIR RMSE: {rmse_all:.2f} Ton")
print("="*40)

# =====
# 10. FEATURE IMPORTANCE (HANYA DITAMPILKAN DI TERMINAL)
# =====
importances = pd.Series(xgb.feature_importances_, index=features)
importances = importances.sort_values(ascending=False)

# Mengubah ke skala persentase murni (dikali 100)

```

```

top5_xgb = importances.head(5) * 100

print("\nTOP 5 VARIABEL PALING BERPENGARUH (XGBOOST):")
for rank, (nama_fitur, nilai) in enumerate(top5_xgb.items(), 1):
    print(f"{rank}. {nama_fitur}: {nilai:.2f}%")
print("="*40)

```

## B. Algoritma ANN

```

import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import os

from sklearn.neural_network import MLPRegressor
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.metrics import r2_score, mean_absolute_error,
mean_squared_error

# =====
# 1. LOAD DATA
# =====
df = pd.read_excel("data_padi.xlsx")

# =====
# 2. PROSES LAGGING (SEMUA VARIABEL IKLIM)
# =====
kolom_fitur = ['LPM (%)', 'CH (mm)', 'ONI', 'Suhu (°C)', 'Kelembaban (%)']

for col in kolom_fitur:
    df[f'{col}_Lag1'] = df[col].shift(1) # 1 bulan sebelum panen
    df[f'{col}_Lag2'] = df[col].shift(2) # 2 bulan sebelum panen
    df[f'{col}_Lag3'] = df[col].shift(3) # 3 bulan sebelum panen

# Hapus data NaN akibat lagging (Menghasilkan 57 baris bersih)
df_clean = df.dropna().reset_index(drop=True)

# =====
# 3. FITUR (X) & TARGET (y)
# =====
features = [col for col in df_clean.columns if 'Lag' in col]

X = df_clean[features]
y = df_clean['P (ton)']

# =====
# 4. SPLIT DATA (RANDOM APPROACH 80:20)

```

```

# =====
X_train_raw, X_test_raw, y_train_raw, y_test_raw = train_test_split(
    X,
    y,
    test_size=0.2,
    random_state=42
)

# =====
# 5. SCALING / NORMALISASI
# =====
scaler_X = StandardScaler()
scaler_y = StandardScaler()

# fit_transform HANYA diaplikasikan pada data latih (Train)
X_train = scaler_X.fit_transform(X_train_raw)
y_train = scaler_y.fit_transform(y_train_raw.values.reshape(-1,1)).ravel()

# Data uji (Test) hanya di-transform menggunakan parameter skala
dari data latih
X_test = scaler_X.transform(X_test_raw)
y_test = scaler_y.transform(y_test_raw.values.reshape(-1,1)).ravel()

# Mempersiapkan skala seluruh data untuk prediksi total di grafik
X_scaled = scaler_X.transform(X)

# =====
# 6. MODEL ANN (Baseline Tanpa Optimasi)
# =====
ann = MLPRegressor(
    hidden_layer_sizes=(100),
    alpha=0.0005,
    learning_rate_init=0.01,
    max_iter=100, #Jumlah maksimum iterasi training
    random_state=42
)

# Proses pelatihan 80% data latih yang sudah discaling bersih
ann.fit(X_train, y_train)

# =====
# 7. PREDIKSI SELURUH DATA (57 BARIS)
# =====
y_all_pred_scaled = ann.predict(X_scaled)

# Denormalisasi: Mengembalikan angka skala ke satuan Ton asli
y_all_pred = scaler_y.inverse_transform(
    y_all_pred_scaled.reshape(-1,1)
).ravel()

```

```

# =====
# 8. EVALUASI AKURASI TOTAL
# =====
r2_all = r2_score(y, y_all_pred)
mae_all = mean_absolute_error(y, y_all_pred)
rmse_all = np.sqrt(mean_squared_error(y, y_all_pred))

# =====
# 9. VISUALISASI GRAFIK 57 BULAN
# =====
date_range = pd.date_range(start='2020-04-01', periods=len(y),
                             freq='MS')
date_labels = [d.strftime('%b-%y') for d in date_range]

plt.figure(figsize=(16, 8))

# Plot Garis Aktual Asli
plt.plot(date_labels, y.values, label='Data Aktual (Asli)',
         color='#1f77b4', marker='o', linewidth=2)

# Plot Garis Hasil Prediksi ANN
plt.plot(date_labels, y_all_pred, label='Prediksi ANN',
         color='#d62728', linestyle='--', marker='s', alpha=0.8)

# Informasi Statistik Performa Model
info_box = (f"Statistik Model (57 Bulan):\n"
            f"R2 Score: {round(r2_all, 3)}\n"
            f"MAE: {round(mae_all, 2)} Ton\n"
            f"RMSE: {round(rmse_all, 2)} Ton")

plt.gca().text(0.02, 0.95, info_box, transform=plt.gca().transAxes,
               fontsize=12,
               verticalalignment='top', bbox=dict(boxstyle='round',
               facecolor='white', alpha=0.9))

plt.title('Grafik Prediksi Produksi Padi (Random Split 80:20 -
ANN)', fontsize=16, pad=20)
plt.xlabel('Periode Waktu (Bulan-Tahun)', fontsize=12)
plt.ylabel('Produksi Padi (Ton)', fontsize=12)
plt.xticks(rotation=45, fontsize=9)
plt.legend(loc='upper right', fontsize=11)
plt.grid(True, linestyle=':', alpha=0.6)
plt.tight_layout()

# --- SIMPAN GRAFIK PREDIKSI ---
os.makedirs("grafik", exist_ok=True)
plt.savefig("grafik/ann_random80.png", dpi=300, bbox_inches='tight')

```

```

print("✓ Grafik ANN dengan pendekatan Random Split (80:20) berhasil
disimpan!")

# Menampilkan grafik prediksi aktual ke layar
plt.show()
plt.close('all')

# =====
# 10. OUTPUT HASIL AKHIR DI TERMINAL
# =====
print("\n" + "="*40)
print(f"HASIL AKHIR R²: {r2_all:.3f}")
print(f"HASIL AKHIR MAE: {mae_all:.2f} Ton")
print(f"HASIL AKHIR RMSE: {rmse_all:.2f} Ton")
print("="*40)

# =====
# 11. WEIGHT ANALYSIS IMPORTANCE (MURNI TERMINAL)
# =====
bobot_input_hidden = np.abs(ann.coefs_[0])
kontribusi_fitur = np.mean(bobot_input_hidden, axis=1)
skor_importance = (kontribusi_fitur / np.sum(kontribusi_fitur)) *
100

importances = pd.Series(skor_importance,
index=features).sort_values(ascending=False)

print("\nWEIGHT ANALYSIS IMPORTANCE (TOP 5 VARIABEL PALING
BERPENGARUH):")
for rank, (nama_fitur, nilai) in
enumerate(importances.head(5).items(), 1):
    print(f"{rank}. {nama_fitur}: {nilai:.2f}%")
print("="*40)

```