

# **ANALISIS WEBSITE PREDIKSI CURAH HUJAN KOTA BOGOR JAWA BARAT MENGGUNAKAN METODE NAÏVE BAYES PADA SEKTOR PERTANIAN.**



**Dosen Pengampu :**

**Mirza Sutrisno, S.SI., M.KOM**

**Disusun Oleh :**

**Rahmita Zahro (22040700046)**

**Muhammad Arief Abean (22040700100)**

**Muhammad Alfarizhi Gato (22040700117)**

**PROGRAM STUDI INFORMATIKA  
FAKULTAS TEKNIK  
UNIVERSITAS MUHAMMADIYAH JAKARTA**

**2025**

## **KATA PENGANTAR**

Puji serta syukur kehadirat Allah SWT, karena atas rahmat dan karunia-Nya, laporan akhir ini dapat selesai tepat waktu tanpa hambatan yang berarti. Laporan akhir ini disusun sebagai bentuk pertanggungjawaban atas pelaksanaan kegiatan Studi Independen Bersertifikat di Orbit Future Academy yang dilaksanakan selama 4 bulan dalam kurun waktu 6 September - 31 Desember 2024. Laporan akhir ini berisi gambaran lengkap mengenai seluruh kegiatan yang telah dilakukan, termasuk tujuan, metode, pelaksanaan, serta hasil yang telah dicapai salah satunya hasil proyek akhir berjudul “Prediksi Curah Hujan Kota Bogor Jawa Barat Menggunakan Metode Naïve Bayes pada Sektor Pertanian”.

Kami menyadari bahwa laporan ini masih memiliki kekurangan dan keterbatasan. Oleh karena itu, kami mengharapkan kritik an saran yang konstruktif demi penyempurnaan laporan ini. Semoga laporan akhir ini dapat menjadi bahan evaluasi dan referensi yang bermanfaat bagi semua pihak serta memberikan kontribusi positif dan menjadi dasar untuk perbaikan di masa yang akan datang.

Hormat kami,

## DAFTAR ISI

<b>KATA PENGANTAR.....</b>	<b>2</b>
<b>DAFTAR ISI.....</b>	<b>3</b>
<b>DAFTAR GAMBAR.....</b>	<b>4</b>
<b>DAFTAR TABEL .....</b>	<b>5</b>
<b>BAB I PENDAHULUAN.....</b>	<b>6</b>
1.1    Latar Belakang.....	6
1.2    Rumusan Masalah.....	7
1.3    Tujuan .....	7
1.4    Manfaat.....	8
<b>BAB II LANDASAN TEORI.....</b>	<b>9</b>
2.1    Prediksi Curah Hujan .....	9
<b>BAB III GAMBARAN UMUM .....</b>	<b>11</b>
3.1    Pengolahan Data .....	11
<b>BAB IV ANALISA DAN HASIL.....</b>	<b>19</b>
4.1    Desain.....	19
4.2    Source code.....	22
<b>BAB V PENUTUP.....</b>	<b>37</b>
Kesimpulan.....	37
<b>DAFTAR PUSTAKA.....</b>	<b>39</b>

## DAFTAR GAMBAR

Gambar 3, 1 Data Exploration .....	13
Gambar 3, 2 Analisis Korelasi .....	14
Gambar 3, 3 Distribusi data training.....	16
Gambar 3, 4 Distribusi data testing.....	17
Gambar 3, 5 Evaluasi Model .....	17
Gambar 4. 1 Use Case Diagram admin dan user .....	19
Gambar 4. 2 Activity Diagram admin dan user .....	20
Gambar 4. 3 Home .....	20
Gambar 4. 4 About.....	21
Gambar 4. 5 Prediksi Curah Hujan .....	22
Gambar 4. 6 Dataset.....	23
Gambar 4. 7 Style .....	25
Gambar 4. 8 Javascript.....	28
Gambar 4. 9 App.py .....	30
Gambar 4. 10 Model.app .....	34

## DAFTAR TABEL

Tabel 3. 1 Variabel Data .....	11
Tabel 3. 2 Fitur Model .....	15

# **BAB I**

## **PENDAHULUAN**

### **1.1 Latar Belakang**

Curah hujan merupakan salah satu faktor iklim yang sangat berpengaruh di berbagai bidang. Bidang pertanian, pelayaran, penerbangan dan beberapa bidang lainnya melibatkan faktor iklim sebagai faktor penentu, khususnya curah hujan. Untuk memprakirakan kondisi curah hujan bulanan/musiman, Badan Meteorologi Klimatologi dan Geofisika (BMKG) secara rutin memperhatikan ketiga faktor pengendali curah hujan yaitu el nino/la nina, IDM dan suhu permukaan perairan Indonesia. Pemerintah telah membangun stasiun hujan untuk mengukur curah hujan di titik atau lokasi tertentu di berbagai daerah di Indonesia yang dianggap memiliki potensi dan dapat mewakili daerah sekitarnya. Namun curah hujan di luar daerah stasiun hujan tidak diketahui secara pasti, sedangkan untuk kepentingan tertentu, dibutuhkan informasi tentang curah hujan di titik lain.

Bogor merupakan salah satu kota besar di Indonesia yang terletak di dataran tinggi. Dengan topografinya yang beragam, Kota Bogor mengalami variasi curah hujan yang signifikan. Stasiun Citeko, misalnya, sering mencatat curah hujan yang tinggi. Untuk mengantisipasi potensi bencana hidrometeorologi dan merencanakan pembangunan yang berkelanjutan, diperlukan prediksi curah hujan yang akurat di setiap kecamatan di Bogor. Informasi ini dapat digunakan sebagai dasar untuk mengambil langkah-langkah mitigasi bencana, seperti peringatan dini dan pengelolaan sumber daya air. Tidak hanya penting untuk mengantisipasi bencana hidrometeorologi dan merencanakan pembangunan berkelanjutan, tetapi juga sangat bermanfaat bagi sektor pertanian. Petani dapat menyusun jadwal tanam yang optimal, memilih varietas tanaman yang sesuai, dan mengelola irigasi secara efektif berdasarkan prediksi curah hujan. Dengan demikian, produktivitas pertanian dapat ditingkatkan dan risiko kerugian akibat cuaca ekstrem dapat diminimalisir.

Salah satu pendekatan untuk memprediksi curah hujan adalah dengan menggunakan analisis deret waktu dan menerapkan metode klasifikasi Naive Bayes. Analisis deret waktu memungkinkan kita mengidentifikasi pola dalam data curah hujan yang terurut berdasarkan waktu. Dengan Naive Bayes, kita dapat mengetahui kemungkinan terjadinya suatu kondisi curah hujan di masa depan berdasarkan data historis. Naïve Bayes Classifier (NBC), atau yang sering disebut klasifikasi Bayesian, adalah suatu metode untuk mengatasi masalah dengan mencari nilai peluang. Naïve Bayes adalah klasifikasi menggunakan metode statistik dan probabilistik yang diusulkan oleh ilmuwan Inggris Thomas Bayes. Naïve Bayes adalah perhitungan statistik untuk memprediksi peluang masa depan berdasarkan pengalaman atau masalah yang dihadapi sebelumnya, sehingga disebut teorema Bayes. Metode ini mengasumsikan bahwa setiap faktor yang mempengaruhi curah hujan bersifat independen satu sama lain. Karena diasumsikan sebagai variabel bebas, maka hanya diperlukan varian variabel dalam suatu kelas untuk menentukan klasifikasi, tidak seluruh matriks kovarians.

## **1.2 Rumusan Masalah**

1. Bagaimana tingkat akurasi metode Naïve Bayes dalam memprediksi curah hujan di Kota Bogor?
2. Apakah metode Naïve Bayes dapat diimplementasikan secara efektif untuk mendukung
3. Faktor-faktor apa saja yang mempengaruhi hasil prediksi curah hujan menggunakan metode Naïve Bayes di Kota Bogor?
4. Bagaimana pengaruh prediksi curah hujan terhadap pengambilan keputusan di sector pertanian di Kota Bogor?

## **1.3 Tujuan**

1. Memberikan rekomendasi dan informasi cuaca yang dapat membantu petani dalam merencanakan aktivitas pertanian, seperti penanaman, pemupukan, dan panen, sesuai dengan kondisi curah hujan

2. Dengan adanya prediksi cuaca yang lebih baik, petani dapat meminimalkan kerugian akibat cuaca ekstrem, meningkatkan efisiensi penggunaan sumber daya, dan mengoptimalkan hasil produksi pertanian.

#### **1.4 Manfaat**

1. Membantu masyarakat mendapatkan informasi terkini tentang curah hujan, sehingga dapat mempersiapkan diri menghadapi potensi cuaca ekstrem.
2. Dengan adanya prediksi cuaca yang lebih baik, petani dapat meminimalkan kerugian akibat cuaca ekstrem, meningkatkan efisiensi penggunaan sumber daya, dan mengoptimalkan hasil produksi pertanian.



## **BAB II**

### **LANDASAN TEORI**

#### **Prediksi Curah Hujan**

Prediksi adalah suatu kegiatan yang secara terstruktur memperkirakan apa yang kemungkinan akan terjadi di masa yang akan datang berdasarkan data dari masa lalu dan saat ini yang dimiliki. Ramalan melibatkan estimasi peristiwa yang telah terjadi sebelumnya secara terorganisir dan informatif untuk mengantisipasi peristiwa yang akan datang. Curah hujan adalah jumlah air hujan yang terkumpul dalam wadah pengukur hujan di area datar, yang tidak menguap, tidak meresap, dan tidak mengalir. Hujan juga dapat didefinisikan sebagai total air hujan yang terakumulasi dalam alat ukur hujan pada permukaan datar yang tidak menguap atau mengalir (suparji, Sriyanto, dan Lestari, 2022).

Menurut (Rizqi, Kusumaningsih, 2022) perkiraan jumlah hujan telah menjadi salah satu tantangan ilmiah dan teknologi terpenting di seluruh dunia dalam seratus tahun terakhir. Ini disebabkan oleh dua penyebab utama, yaitu perannya dalam berbagai aktivitas manusia dan kemajuan dalam perhitungan komputer. Hujan adalah salah satu komponen penting dari iklim. Hujan juga merupakan elemen yang berperan langsung dalam perubahan cuaca, baik yang bersifat positif maupun negatif. Ada beberapa faktor utama yang dapat memengaruhi terjadinya hujan, seperti temperatur udara, kelembapan, kecepatan angin, dan durasi sinar matahari.

#### **2.1 NAÏVE BAYES**

Menurut (Hubert, 2020) metode Naïve Bayes adalah teknik pengklasifikasian yang berbasis pada probabilitas dan statistik, yang ditemukan oleh ilmuwan Inggris bernama Thomas Bayes. Naïve Bayes berfungsi untuk menghitung kemungkinan dengan cara mengevaluasi nilai probabilitas dalam setiap keanggotaan dari kelas tertentu. Dalam teori Naïve Bayes, probabilitas dianggap independen dan tidak terpengaruh oleh keberadaan fitur lain. Contohnya, suatu buah bisa diidentifikasi sebagai apel jika berwarna merah, berbentuk bulat, dan memiliki diameter sekitar 3 inci. Semua karakteristik ini memberikan

kontribusi independen terhadap kemungkinan bahwa buah tersebut merupakan apel, sehingga metode ini disebut 'naif'. Metode Naïve Bayes mudah diimplementasikan dan sangat bermanfaat untuk dataset yang sangat besar. Selain kesederhanaannya, Naïve Bayes juga diketahui mampu mengungguli metode klasifikasi yang lebih kompleks (Pheonix, Suhartono, 2021).

## BAB III GAMBARAN UMUM

### 3.1 Pengolahan Data

#### 3.1.1 Data Acquisition

Data meliputi serangkaian proses pengukuran, pengumpulan, dan validasi data yang dibutuhkan dalam suatu proyek Artificial Intelligence. Pada proyek Prediksi Curah Hujan Kota Bogor Jawa Barat Menggunakan Metode Naïve Bayes pada Sektor Pertanian, dataset yang diperlukan berupa data sekunder dengan jenis data time series yang diambil dari situs BMKG Indonesia. Diambil sampel data iklim harian dari Januari 2013 hingga Desember 2023. Dataset yang diambil memiliki 11 variabel dengan rincian sebagai berikut.

*Tabel 3. 1 Variabel Data*

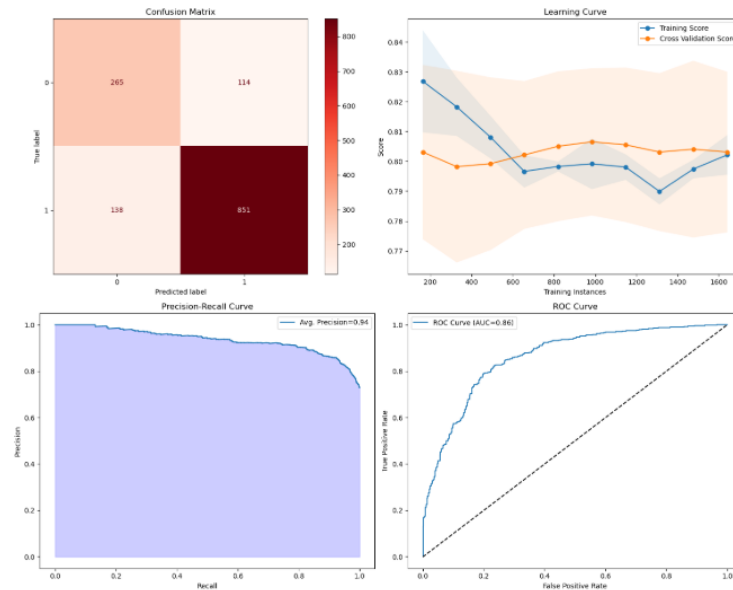
Variabel	Definisi Operasional Variabel
Tanggal	Periode waktu (dd-mm-yyyy).
Temperatur minimum (Tn)	Suhu udara terendah yang tercatat pada periode satu hari dalam derajat celcius (°C).
Temperatur maksimum (Tx)	Suhu udara tertinggi yang tercatat pada periode satu hari dalam derajat celcius (°C).
Temperatur rata-rata (Tavg)	Nilai rata-rata dari semua pengukuran suhu yang tercatat pada periode satu hari dalam derajat celcius (°C).
Kelembapan rata-rata (RH_avg)	Nilai rata-rata dari semua pengukuran kelembaban udara pada periode satu hari dalam persen (%).
Curah hujan (RR)	Jumlah air hujan yang jatuh dalam

	periode satu hari, diukur dalam milimeter (mm)
Lamanya penyinaran matahari (ss)	Jumlah waktu matahari bersinar langsung dalam periode satu hari, dinyatakan dalam jam.
Kecepatan angin maksimum	Kecepatan angin tertinggi yang tercatat pada periode satu hari dalam meter per detik (m/s).
Arah angin saat kecepatan maksimum	Arah dari mana angin bertiup saat mencapai kecepatan maksimum dalam derajat ( $^{\circ}$ ).
Kecepatan angin rata-rata	Nilai rata-rata dari semua pengukuran kecepatan angin dalam periode satu hari, diukur dalam meter per detik (m/s).
Arah angin terbanyak	Arah angin yang paling sering terjadi pada periode satu hari dalam derajat ( $^{\circ}$ )

4

### 3.1.2 Data Exploration

*Data exploration* merupakan proses menjelajahi dataset untuk memahami isi, komponen dan karakteristiknya sehingga kita dapat mengetahui pola data tersebut sehingga membantu kita untuk memilih *tools* yang tepat untuk melakukan pra- pemrosesan atau dalam melakukan analisis. Kami menggunakan teknik *visualization* untuk eksplorasi data agar lebih mudah untuk membaca data.



Gambar 3, 1 Data Exploration

### 3.1.3 Preprocessing Data

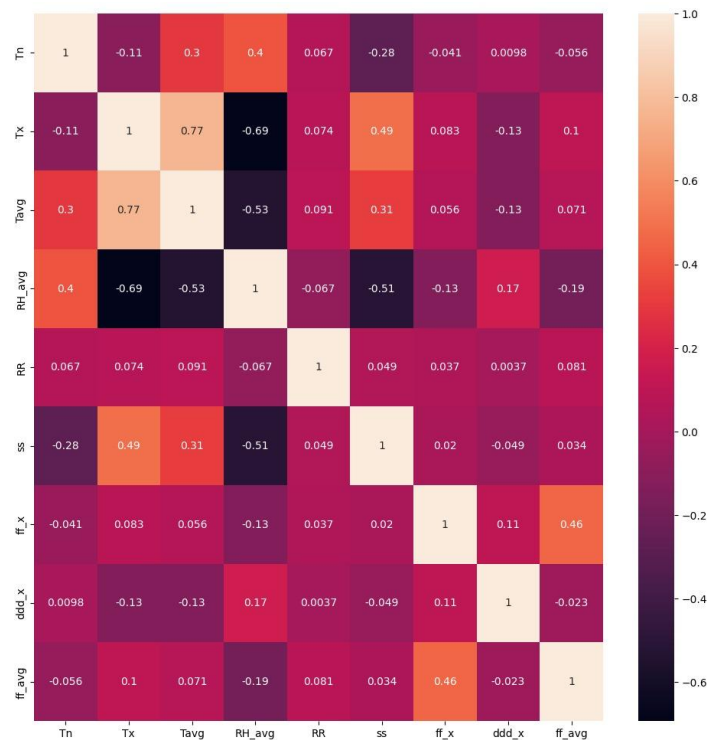
*Preprocessing data* adalah tahap awal yang penting dalam *data mining*, bertujuan untuk mempersiapkan data mentah sebelum memasuki proses berikutnya. Tahap ini melibatkan penghilangan data yang tidak relevan atau mengubahnya menjadi format yang lebih mudah diolah oleh sistem. Selain itu, praproses dilakukan untuk meningkatkan akurasi hasil, mempercepat proses komputasi pada masalah skala besar, dan mengurangi ukuran data tanpa menghilangkan informasi penting yang dikandungnya. Proses ini mencakup berbagai langkah, seperti pembersihan data (*data cleaning*), integrasi data (*data integration*), reduksi data (*data reduction*), serta transformasi data (*data transformation*).

*Target data* dalam machine learning merujuk pada variabel atau kolom dalam dataset yang ingin diprediksi atau dipelajari oleh model. Dalam konteks *supervised learning*, *target data* sering disebut sebagai variabel dependen atau label karena ia merupakan nilai yang bergantung pada input yang diberikan oleh fitur lainnya dalam dataset (variabel independen). Target data dapat berupa berbagai tipe variabel, seperti kategori (untuk klasifikasi) atau nilai kontinu (untuk regresi). Pada proyek Prediksi Curah Hujan Kota Bogor, variabel target yang digunakan adalah variabel curah hujan (*rr*) yang merepresentasikan jumlah air hujan yang jatuh dalam periode satu hari, diukur dalam milimeter (mm).

Adapun pra-pemrosesan data pada penelitian ini mencakup tiga langkah. Pertama, melakukan pengecekan data yang duplikat. Data duplikat adalah data yang identik atau hampir identik yang muncul lebih dari satu kali dalam suatu dataset. Setelah dilakukan pengecekan, tidak ditemukan data duplikat pada dataset iklim harian. Oleh karena itu, proses pra-pemrosesan data dapat dilanjutkan ke tahap berikutnya.

Kedua melakukan pengecekan *missing value*. Terdapat *missing value* pada semua variabel dalam *dataset*. Untuk mengatasi hal ini, *missing value* tersebut diisi dengan rata-rata (*mean*) dari masing-masing variabel. Dengan menambahkan nilai *mean* pada setiap data *missing value* akan cenderung mempertahankan bentuk distribusi data secara keseluruhan.

Ketiga melakukan analisis korelasi. Analisis korelasi ini bertujuan untuk penentuan fitur dan target. Kami melakukan analisis korelasi menggunakan visualisasi *heatmap*. *Heatmap* adalah visualisasi yang sangat berguna untuk memahami hubungan antara banyak variabel sekaligus. Dalam konteks analisis korelasi, heatmap menampilkan kekuatan dan arah hubungan antara setiap pasangan variabel dalam bentuk warna. Diperoleh koefisien korelasi dari visualisasi *heatmap* sebagai berikut.



Gambar 3, 2 Analisis Korelasi

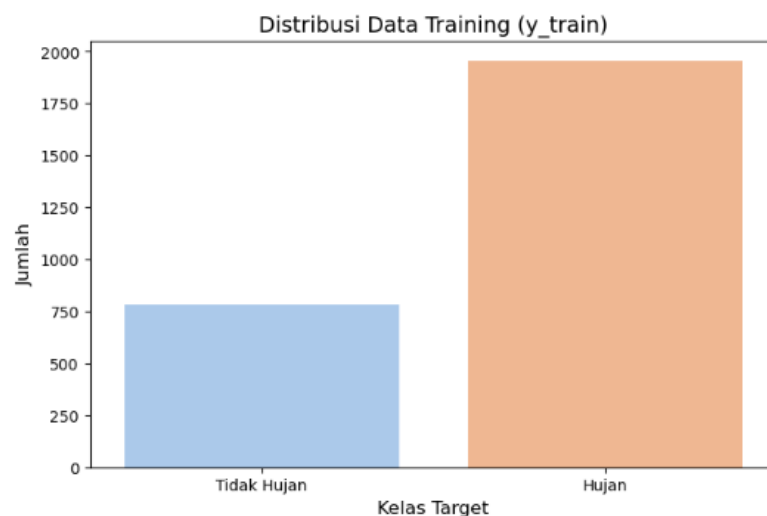
Berdasarkan visualisasi *heatmap* di atas, diperoleh koefisien yang paling berpengaruh terhadap variabel curah hujan (RR) adalah suhu rata-rata (Tavg), kecepatan angin rata-rata (ff\_avg), suhu maksimum (Tx), suhu minimum (Tn), kelembapan rata-rata (RH\_avg), dan lama penyinaran matahari (ss). Oleh karena itu, kami mengambil keenam variabel tersebut sebagai fitur model dan variabel curah hujan sebagai target dengan rincian pada tabel 1.5.

**Tabel 3. 2 Fitur Model**

<b>Variabel</b>	<b>Koefisien Korelasi</b>	<b>Keterangan</b>
suhu rata-rata (Tavg)	0.091	suhu rata-rata mungkin sedikit berpengaruh terhadap curah hujan, mungkin terkait dengan pola atmosfer yang memengaruhi intensitas hujan.
kecepatan angin rata-rata (ff_avg)	0.081	kecepatan angin dapat mempengaruhi distribusi awan dan hujan, sehingga masih bisa dipertimbangkan sebagai fitur.
suhu maksimum (Tx)	0.074	suhu maksimum dapat berpengaruh terhadap dinamika atmosfer yang berkontribusi pada pembentukan hujan.
suhu minimum (Tn)	0.067	suhu minimum dapat berpengaruh terhadap dinamika atmosfer yang berkontribusi pada pembentukan hujan.
kelembapan rata-rata (RH_avg),	-0.067	menunjukkan hubungan terbalik di mana penurunan kelembapan

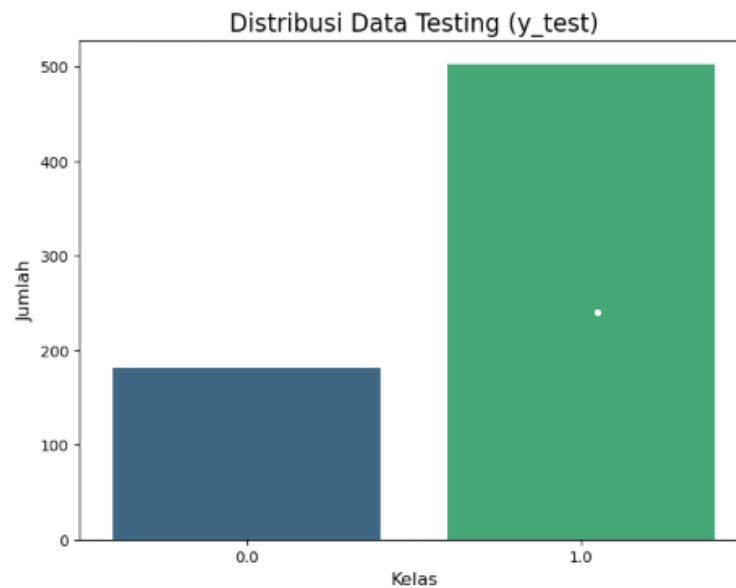
		rata-rata dapat sedikit mempengaruhi penurunan curah hujan.
lama penyinaran matahari (ss)	0.049	sinar matahari sering kali berkaitan dengan pembentukan awan dan penguapan yang dapat mempengaruhi curah hujan.

*Split data* adalah proses pembagian *dataset* menjadi dua bagian utama, yaitu data pelatihan (*training set*) dan data pengujian (*testing set*). Data pelatihan digunakan untuk melatih model *machine learning*, yaitu tahap di mana model belajar mengenali pola dan hubungan dalam data. Sementara itu, data pengujian digunakan untuk mengevaluasi kinerja model yang telah dilatih, memastikan bahwa model mampu melakukan prediksi dengan baik pada data yang belum pernah dilihat sebelumnya. Fungsi utama dari *split data* adalah untuk mengevaluasi kemampuan model secara objektif dan menghindari *overfitting*, yaitu kondisi di mana model terlalu baik dalam menyesuaikan diri dengan data pelatihan tetapi gagal memberikan prediksi yang akurat pada data baru. Dengan memisahkan data pelatihan dan pengujian, kita dapat memastikan bahwa model memiliki kemampuan generalisasi yang baik, yaitu kemampuan untuk bekerja dengan data di luar data yang digunakan untuk pelatihan.



*Gambar 3, 3 Distribusi data training*





*Gambar 3, 4 Distribusi data testing*

Kinerja model diukur menggunakan berbagai metrik evaluasi. Pada proyek ini, tahap evaluasi bertujuan untuk mengukur seberapa baik model dapat mengklasifikasikan data baru dan mengidentifikasi area-area yang perlu ditingkatkan. Kami menggunakan tiga pengukuran yaitu pengukuran akurasi, analisis matriks konfusi, dan *classification report* pada gambar berikut.

```

Accuracy: 81.58%
Confusion Matrix:
[[265 114]
 [138 851]]
Classification Report:

```

	precision	recall	f1-score	support
0.0	0.66	0.70	0.68	379
1.0	0.88	0.86	0.87	989
accuracy			0.82	1368
macro avg	0.77	0.78	0.77	1368
weighted avg	0.82	0.82	0.82	1368

*Gambar 3, 5 Evaluasi Model*

Pada pengukuran akurasi, *accuracy* menunjukkan proporsi prediksi yang benar terhadap total prediksi. *Accuracy* memberikan gambaran umum tentang seberapa baik model dapat memprediksi kelas yang benar. Berdasarkan gambar, diperoleh *accuracy* sebesar 81.58% artinya model dapat memprediksi kelas dengan persentase 81.58% benar.

Pengukuran yang kedua adalah *confusion matrix* dimana akan ditampilkan rincian tentang jumlah prediksi benar dan salah untuk setiap kelas. *Confusion matrix* membantu mengidentifikasi tipe kesalahan yang sering terjadi, seperti *false positive* (prediksi positif yang sebenarnya negatif) dan *false negative* (prediksi negatif yang sebenarnya positif). Berdasarkan gambar, diperoleh nilai kuadran kiri atas 265 bernilai *True Negative* (TN), kuadran kanan atas 114 bernilai *False Positive* (FP), kuadran kiri bawah 138 bernilai *False Negative* (FN), dan kuadran kanan bawah 851 bernilai *True Positive* (TP). Jadi, model cenderung lebih baik dalam memprediksi kelas 1 karena jumlah *True Positive* (851) lebih besar dibanding jumlah *False Negative* (138) dan terdapat beberapa kesalahan dalam memprediksi kelas 0 karena jumlah *False Positive* (114) yang cukup signifikan.

Pengukuran yang ketiga adalah berdasarkan *classification report*. Terdapat tiga metrik yang digunakan yaitu *precision*, *recall*, dan *F1-score*. Berdasarkan gambar, model memiliki kinerja yang cukup baik, terutama dalam memprediksi kelas 1. Namun, masih terdapat sedikit kesalahan dalam memprediksi kelas 0.

## BAB IV

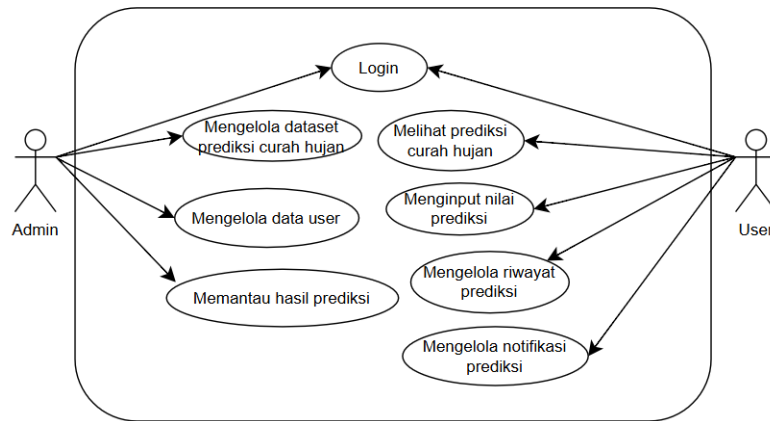
### ANALISA DAN HASIL

#### Desain

##### 4.1.1 Arsitektur Sistem

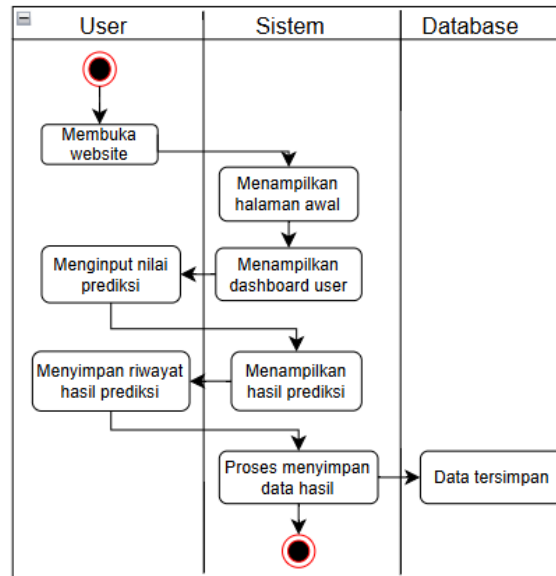
Arsitektur sistem dirancang untuk skalabilitas dan juga keamanan sistem dari berbagai segi. Perancangan arsitektur sistem ini menggunakan alat bantu UML (Unified Modeling Language) yang terdiri dari beberapa diagram berikut:

##### 1. Use Case Diagram



*Gambar 4. 1 Use Case Diagram admin dan user*

##### 2. Activity Diagram

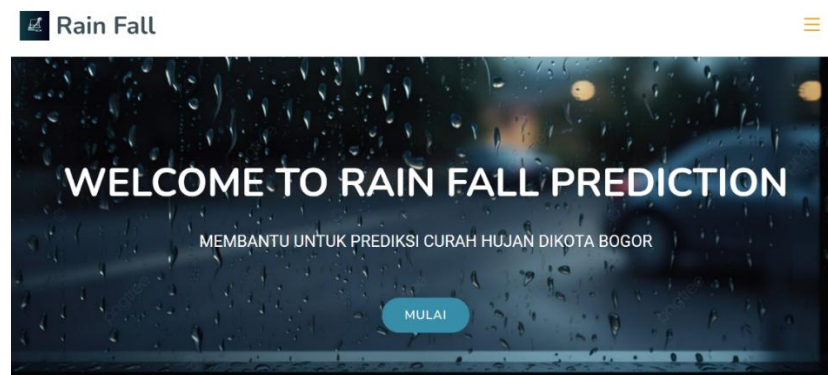


*Gambar 4. 2 Activity Diagram admin dan user*

#### 4.1.2 Tampilan Website Antarmuka

- Tampilan Utama

Landing page atau Tampilan awal bertindak sebagai pintu masuk pertama untuk pengguna. Ini adalah kesempatan untuk memberikan kesan pertama yang baik, yang bisa menentukan apakah pengguna akan melanjutkan menggunakan aplikasi atau tidak. Terdapat tombol “mulai” yang bertujuan agar pengguna aplikasi dapat langsung menuju halaman prediksi. Terdapat empat menu pilihan, yaitu menu “Home”, “About”, dan “Prediksi”.



*Gambar 4. 3 Home*

- About



Pada halaman about, dijelaskan secara singkat mengenai website ”Prediksi Curah Hujan” mulai dari deskripsi singkat aplikasi, kegunaan dalam berbagai sektor, serta keunggulan yang dimiliki oleh website.



*Gambar 4. 4 About*

- Prediksi curah hujan

Selanjutnya yaitu halaman “Prediksi”, dimana halaman ini adalah prediksi curah hujan. Apabila pengguna ingin melakukan prediksi curah hujan, pengguna cukup mengisi variabelvariabel yang tertera dan menekan tombol “Prediksi Curah Hujan”. Setelahnya akan muncul prediksi berupa “Ya” atau “Tidak”.

 Rain Fall

## PREDIKSI CURAH HUJAN

Masukkan nilai berikut untuk memprediksi curah hujan di Kota Bogor

CONTOH PREDIKSI

"Tn": "19",  
"Tx": "26.7",  
"Tavg": "21.3",  
"RH\_avg": "89",  
"ss": "1.4",  
"ff\_avg": "1",  
PREDIKSI: "1a",

Nilai TN

Nilai TX


Nilai TAVG

Nilai RH AVG

Nilai SS

Nilai FF AVG

MULAI PREDIKSI

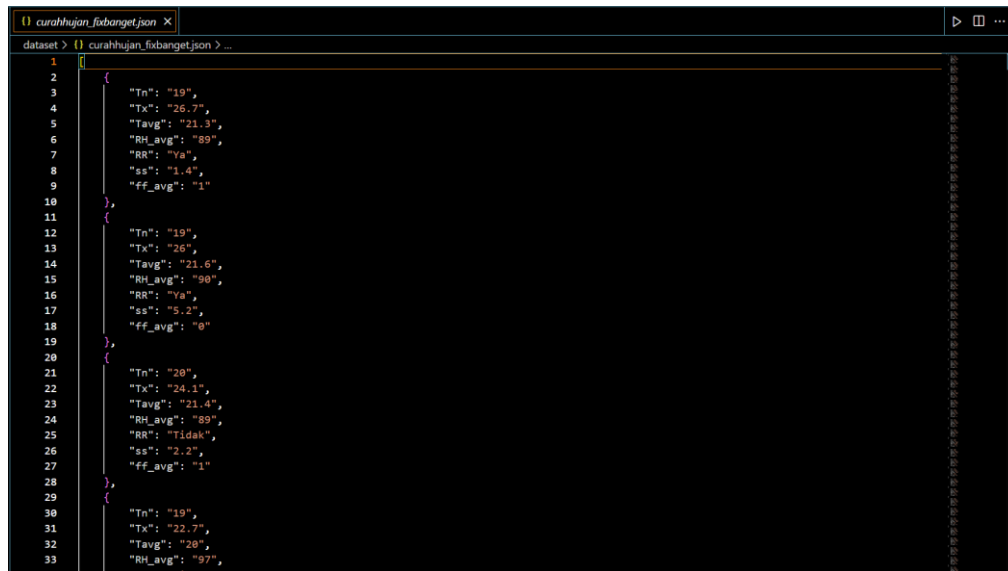


*Gambar 4. 5 Prediksi Curah Hujan*

## Source code

### **Dataset:**

dataset ini merupakan dataset yang sudah diubah ke json dan kami tidak bisa menampilkan semua dikarenakan terlalu banyak



Gambar 4. 6 Dataset

```

[
  {
    "Tn": "19",
    "Tx": "26.7",
    "Tavg": "21.3",
    "RH_avg": "89",
    "RR": "Ya",
    "ss": "1.4",
    "ff_avg": "1"
  },
  {
    "Tn": "19",
    "Tx": "26",
    "Tavg": "21.6",
    "RH_avg": "90",
    "RR": "Ya",
    "ss": "5.2",
    "ff_avg": "0"
  },
  {
    "Tn": "28",
    "Tx": "24.1",
    "Tavg": "21.4",
    "RH_avg": "89",
    "RR": "Tidak",
    "ss": "2.2",
    "ff_avg": "1"
  },
  {
    "Tn": "19",
    "Tx": "22.7",
    "Tavg": "20",
    "RH_avg": "97",
    "RR": "Ya",
    "ss": "1.4",
    "ff_avg": "1"
  }
]

```

```

},
{
  "Tn": "20",
  "Tx": "24.1",
  "Tavg": "21.4",
  "RH_avg": "89",
  "RR": "Tidak",
  "ss": "2.2",
  "ff_avg": "1"
},
{
  "Tn": "19",
  "Tx": "22.7",
  "Tavg": "20",
  "RH_avg": "97",
  "RR": "Tidak",
  "ss": "2",
  "ff_avg": "1"
},
{
  "Tn": "19",
  "Tx": "25.6",
  "Tavg": "20.8",
  "RH_avg": "85",
  "RR": "Ya",
  "ss": "0",
  "ff_avg": "2"
},
{
  "Tn": "19",
  "Tx": "23.3",

```



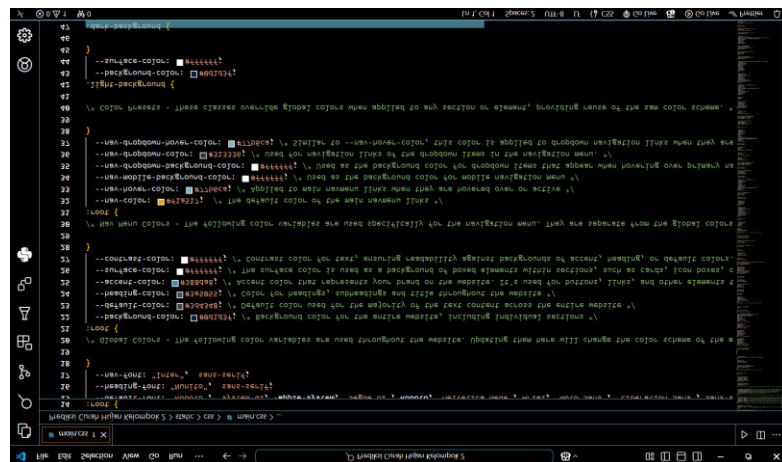
```

    "Tavg": "20",
    "RH_avg": "94",
    "RR": "Ya",
    "ss": "3.4",
    "ff_avg": "1"
  },
  {
    "Tn": "19",
    "Tx": "23.1",
    "Tavg": "20.4",
    "RH_avg": "91",
    "RR": "Ya",
    "ss": "0",
    "ff_avg": "1"
  }
}

```

## Style:

Ini merupakan css dari website kami dan kami tidak bisa menampilkan semua dikarenakan terlalu banyak



Gambar 4. 7 Style

```
:root {

    --default-font: "Roboto", system-ui, -apple-system, "Segoe UI", Roboto, "Helvetica Neue", Arial, "Noto Sans", "Liberation Sans", sans-serif, "Apple Color Emoji", "Segoe UI Emoji", "Segoe UI Symbol", "Noto Color Emoji";

    --heading-font: "Nunito", sans-serif;

    --nav-font: "Inter", sans-serif;

}
```

**/\* Global Colors - The following color variables are used throughout the website. Updating them here will change the color scheme of the entire website \*/**

```
:root {

    --background-color: #0d1d3f; /* Background color for the entire website, including individual sections */

    --default-color: #3d4348; /* Default color used for the majority of the text content across the entire website */

    --heading-color: #3e5055; /* Color for headings, subheadings and title throughout the website */

    --accent-color: #388da8; /* Accent color that represents your brand on the website. It's used for buttons, links, and other elements that need to stand out */

    --surface-color: #ffffff; /* The surface color is used as a background of boxed elements within sections, such as cards, icon boxes, or other elements that require a visual separation from the global background. */

    --contrast-color: #ffffff; /* Contrast color for text, ensuring readability against backgrounds of accent, heading, or default colors. */

}
```

**/\* Nav Menu Colors - The following color variables are used specifically for the navigation menu. They are separate from the global colors to allow for more customization options \*/**

```
:root {  
  
  --nav-color: #f1a517; /* The default color of the main navmenu links */  
  
  --nav-hover-color: #77b6ca; /* Applied to main navmenu links when they are hovered over or active */  
  
  --nav-mobile-background-color: #ffffff; /* Used as the background color for mobile navigation menu */  
  
  --nav-dropdown-background-color: #ffffff; /* Used as the background color for dropdown items that appear when hovering over primary navigation items */  
  
  --nav-dropdown-color: #313336; /* Used for navigation links of the dropdown items in the navigation menu. */  
  
  --nav-dropdown-hover-color: #77b6ca; /* Similar to --nav-hover-color, this color is applied to dropdown navigation links when they are hovered over. */  
  
}
```

**/\* Color Presets - These classes override global colors when applied to any section or element, providing reuse of the same color scheme. \*/**

```
.light-background {  
  
  --background-color: #0d1d3f;  
  
  --surface-color: #ffffff;  
  
}
```

```
.dark-background {  
  
  --background-color: #060606;
```

```

--default-color: #ffffff;

--heading-color: #ffffff;

--surface-color: #252525;

--contrast-color: #ffffff;

}

/* Smooth scroll */

:root {

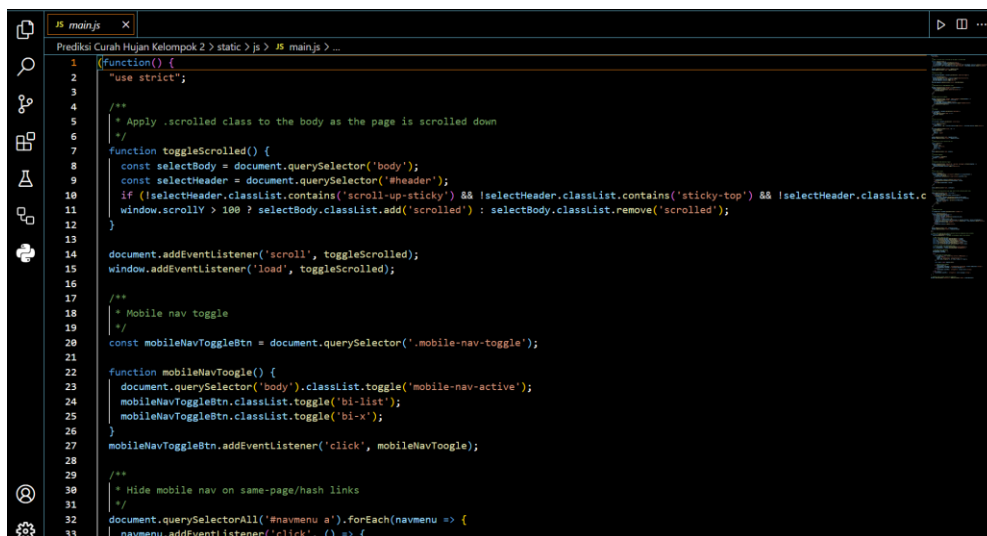
    scroll-behavior: smooth;

}

```

### Javascript:

Ini merupakan javascript dari website kami dan kami tidak bisa menampilkan semua dikarenakan terlalu banyak



Gambar 4. 8 Javascript

```

(function() {

    "use strict";

    /**

    * Apply .scrolled class to the body as the page is scrolled down

    */

    function toggleScrolled() {

        const selectBody = document.querySelector('body');

        const selectHeader = document.querySelector('#header');

        if (!selectHeader.classList.contains('scroll-up-sticky') &&
!selectHeader.classList.contains('sticky-top') &&
!selectHeader.classList.contains('fixed-top')) return;

        window.scrollY > 100 ? selectBody.classList.add('scrolled') :
selectBody.classList.remove('scrolled');

    }

    document.addEventListener('scroll', toggleScrolled);

    window.addEventListener('load', toggleScrolled);

    /**

    * Mobile nav toggle

    */

    const mobileNavToggleBtn = document.querySelector('.mobile-nav-toggle');

    function mobileNavToogle() {

        document.querySelector('body').classList.toggle('mobile-nav-active');

        mobileNavToggleBtn.classList.toggle('bi-list');
    }

```

```

mobileNavToggleBtn.classList.toggle('bi-x');

}

mobileNavToggleBtn.addEventListener('click', mobileNavToggle);

/**

* Hide mobile nav on same-page/hash links

*/

document.querySelectorAll('#navmenu a').forEach(navmenu => {

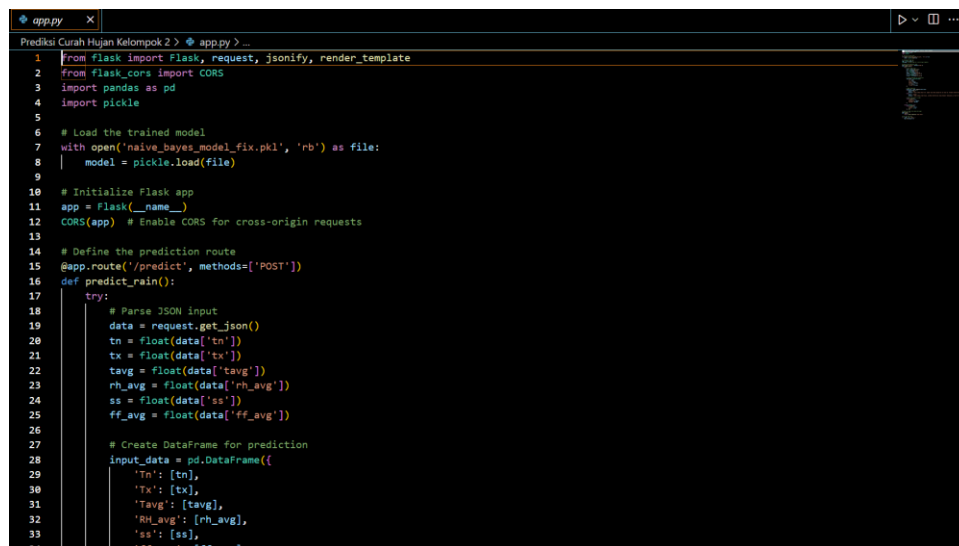
    navmenu.addEventListener('click', () => {

```

## Python:

Ini merupakan code python yang mana kami gunakan sebagai backend yang bertanggung jawab untuk mengelola logika aplikasi, komunikasi dengan database, otentikasi pengguna, pengolahan data, dan penyajian data

App.py



```

1 from flask import Flask, request, jsonify, render_template
2 from flask_cors import CORS
3 import pandas as pd
4 import pickle
5
6 # Load the trained model
7 with open('naive_bayes_model_fix.pkl', 'rb') as file:
8     model = pickle.load(file)
9
10 # Initialize Flask app
11 app = Flask(__name__)
12 CORS(app) # Enable CORS for cross-origin requests
13
14 # Define the prediction route
15 @app.route('/predict', methods=['POST'])
16 def predict_rain():
17     try:
18         # Parse JSON input
19         data = request.get_json()
20         tn = float(data['tn'])
21         tx = float(data['tx'])
22         tavg = float(data['tavg'])
23         rh_avg = float(data['rh_avg'])
24         ss = float(data['ss'])
25         ff_avg = float(data['ff_avg'])
26
27         # Create DataFrame for prediction
28         input_data = pd.DataFrame({
29             'tn': [tn],
30             'tx': [tx],
31             'tavg': [tavg],
32             'rh_avg': [rh_avg],
33             'ss': [ss],
34             'ff_avg': [ff_avg]
35         })

```

Gambar 4. 9 App.py

```

from flask import Flask, request, jsonify, render_template

from flask_cors import CORS

import pandas as pd

import pickle


# Load the trained model

with open('naive_bayes_model_fix.pkl', 'rb') as file:

    model = pickle.load(file)

# Initialize Flask app

app = Flask(__name__)

CORS(app) # Enable CORS for cross-origin requests


# Define the prediction route

@app.route('/predict', methods=['POST'])

def predict_rain():

    try:

        # Parse JSON input

        data = request.get_json()

        tn = float(data['tn'])

        tx = float(data['tx'])

        tavg = float(data['tavg'])

        rh_avg = float(data['rh_avg'])

        ss = float(data['ss'])

```

```

ff_avg = float(data['ff_avg'])

# Create DataFrame for prediction

input_data = pd.DataFrame({

    'Tn': [tn],

    'Tx': [tx],

    'Tavg': [tavg],

    'RH_avg': [rh_avg],

    'ss': [ss],

    'ff_avg': [ff_avg]

})

# Make prediction

prediction = model.predict(input_data)

if prediction[0] == 1:

    result = "Ya"

    message = "Cuaca sedang hujan nih, jangan lupa sedia payung dan jas hujan ya.
Selamat beraktivitas!"

else:

    result = "Tidak"

    message = "Cuaca sedang tidak hujan, silahkan beraktivitas tanpa khawatir
kehujanan ya. Have fun!"

# Return the result as JSON

return jsonify({

```



```

        'success': True,

        'prediction': result,

        'message': message

    })

except Exception as e:

    return jsonify({

        'success': False,

        'error': str(e)

    }), 400

# Route to render the index.html page

@app.route('/')

def index():

    return render_template('index.html')

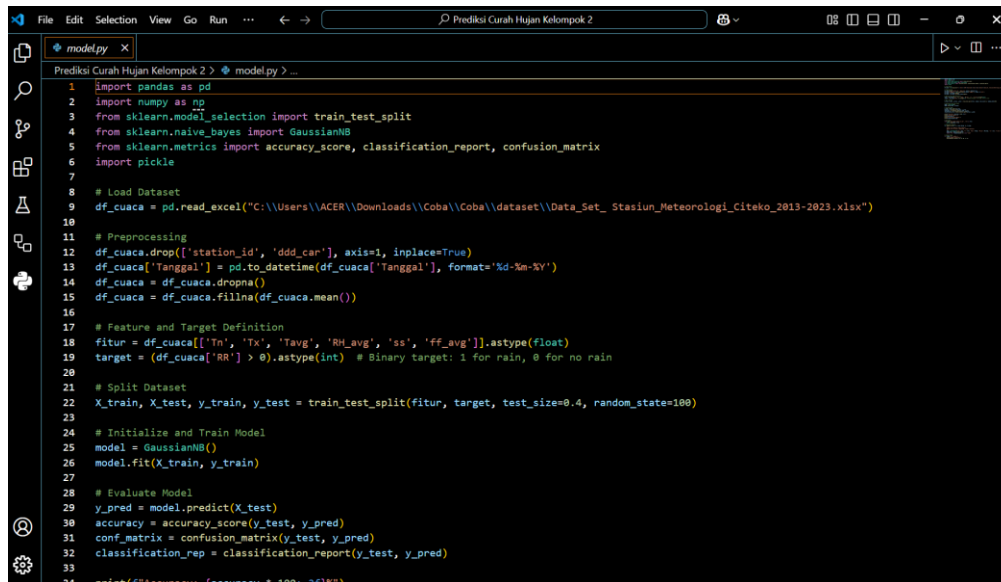
# Run the Flask app

if __name__ == '__main__':

    app.run(debug=True)

```

Model.app



*Gambar 4. 10 Model.app*

```
import pandas as pd
```

```
import numpy as np
```

```
from sklearn.model_selection import train_test_split
```

```
from sklearn.naive_bayes import GaussianNB
```

```
from sklearn.metrics import accuracy_score, classification_report, confusion_matrix
```

```
import pickle
```

```
# Load Dataset
```

```
df_cuaca =
```

```
pd.read_excel("C:\\Users\\ACER\\Downloads\\Coba\\Coba\\dataset\\Data_Set_  
Stasiun_Meteorologi_Citeko_2013-2023.xlsx")
```

```
# Preprocessing
```

```
df_cuaca.drop(['station_id', 'ddd_car'], axis=1, inplace=True)
```

```

df_cuaca['Tanggal'] = pd.to_datetime(df_cuaca['Tanggal'], format='%d-%m-%Y')

df_cuaca = df_cuaca.dropna()

df_cuaca = df_cuaca.fillna(df_cuaca.mean())


# Feature and Target Definition

fitur = df_cuaca[['Tn', 'Tx', 'Tavg', 'RH_avg', 'ss', 'ff_avg']].astype(float)

target = (df_cuaca['RR'] > 0).astype(int) # Binary target: 1 for rain, 0 for no rain


# Split Dataset

X_train, X_test, y_train, y_test = train_test_split(fitur, target, test_size=0.4,
random_state=100)


# Initialize and Train Model

model = GaussianNB()

model.fit(X_train, y_train)


# Evaluate Model

y_pred = model.predict(X_test)

accuracy = accuracy_score(y_test, y_pred)

conf_matrix = confusion_matrix(y_test, y_pred)

classification_rep = classification_report(y_test, y_pred)


print(f'Accuracy: {accuracy * 100:.2f}%')

```

```

print("Confusion Matrix:")

print(conf_matrix)

print("Classification Report:")

print(classification_rep)


# Save Model

with open('naive_bayes_model_fix.pkl', 'wb') as file:

    pickle.dump(model, file)


# Prediction Function

def predict_rain(Tn, Tx, Tavg, RH_avg, ss, ff_avg):

    """

    Predict rain based on input features.

    Returns "Ya" if rain is predicted, "Tidak" otherwise.

    """

    data = pd.DataFrame({'Tn': [Tn], 'Tx': [Tx], 'Tavg': [Tavg], 'RH_avg': [RH_avg],
'ss': [ss], 'ff_avg': [ff_avg]})

    prediction = model.predict(data)

    return "Ya" if prediction[0] == 1 else "Tidak"


# Example Usage

if __name__ == "__main__":

    print("Example Prediction:")

    print(predict_rain(20, 30, 25, 80, 5, 3))

```

## **BAB V**

### **PENUTUP**

#### **Kesimpulan**

Berdasarkan hasil pengolahan data dan evaluasi model menggunakan metode Naïve Bayes pada proyek prediksi curah hujan di Kota Bogor, dapat ditarik beberapa kesimpulan yaitu:

1. Akurasi Model: Model Naïve Bayes menunjukkan tingkat akurasi sebesar 81.58%, yang mengindikasikan bahwa model ini mampu memprediksi kondisi curah hujan dengan benar pada 81.58% dari total prediksi yang dilakukan. Tingkat akurasi ini menunjukkan bahwa metode Naïve Bayes cukup efektif dalam menangkap pola dan tren dalam data curah hujan yang dianalisis.
2. Kinerja pada Kelas Positif dan Negatif:
  - o Kelas Positif (True Positive): Model menunjukkan performa yang baik dalam memprediksi kelas 1 (True Positive) dengan jumlah prediksi benar sebanyak 851. Hal ini menunjukkan kemampuan model dalam mengidentifikasi kondisi curah hujan yang sebenarnya terjadi.
  - o Kelas Negatif (True Negative): Terdapat 265 prediksi benar untuk kelas 0 (True Negative), namun juga terdapat sejumlah kesalahan prediksi sebesar 114 False Positive dan 138 False Negative. Kesalahan ini menunjukkan bahwa model masih memiliki ruang untuk perbaikan dalam membedakan kondisi tanpa curah hujan secara akurat.
3. Metrik Evaluasi Tambahan:
  - o Precision, Recall, dan F1-Score: Berdasarkan classification report, model menunjukkan kinerja yang baik terutama dalam memprediksi kelas 1. Precision dan recall yang tinggi untuk kelas ini menunjukkan bahwa model efektif dalam mengidentifikasi kondisi hujan tanpa banyak kesalahan positif palsu atau negatif palsu. Namun, terdapat sedikit kesalahan dalam prediksi kelas 0, yang dapat diatasi dengan optimalisasi lebih lanjut pada model.
4. Analisis Korelasi: Proses analisis korelasi menggunakan heatmap membantu dalam menentukan fitur-fitur yang paling berpengaruh terhadap prediksi curah hujan. Hal ini memungkinkan pemilihan variabel yang relevan dan peningkatan efisiensi model dalam proses prediksi.
5. Proses Preprocessing: Tahap pra-pemrosesan data, termasuk penanganan missing values dan analisis korelasi, berkontribusi signifikan terhadap peningkatan kualitas data dan akurasi model. Dengan memastikan data bebas dari duplikasi dan missing values, serta memilih fitur yang relevan, model dapat bekerja lebih optimal dalam memprediksi curah hujan.

Secara keseluruhan, metode Naïve Bayes terbukti cukup efektif dalam memprediksi curah hujan di Kota Bogor dengan tingkat akurasi yang memadai. Meskipun terdapat beberapa kesalahan prediksi, terutama pada kelas 0, model ini tetap memberikan dasar yang kuat untuk implementasi lebih lanjut dalam mendukung pengambilan keputusan di sektor pertanian dan mitigasi bencana hidrometeorologi. Untuk peningkatan lebih lanjut, dapat dilakukan optimasi model dan eksplorasi metode klasifikasi lainnya guna meningkatkan akurasi dan keandalan prediksi.

## DAFTAR PUSTAKA

IPCC. (2021). Climate Change 2021: The Physical Science Basis. Intergovernmental Panel on Climate Change.

World Meteorological Organization. (2020). WMO Guidelines on Multi-Hazard Impact-Based Forecast and Warning Services.

A. Rizqi and D. Kusumaningsih, (2022). “Classification of Rainfall in Bogor City West Java Province using Naive Bayes Method,” Semin. Nas. Mhs. Fak. Teknol. Inf., no. September, pp. 542–550.

S. Suparji, S. Sriyanto, dan S. Lestari, “Metode Pendugaan Curah Hujan Dasarian Menggunakan Jaringan Syaraf Tiruan Dengan Algoritma Back-Propagation ( Studi Kasus : Stasiun Klimatologi Pesawaran Lampung ),” hal. 139–144, 2022

P. Hubert, R. Sudaryono Phoenix, D. Suhartono, Classifying promotion images using optical character recognition and Naïve Bayes classifier, Procedia. Comput. Sci. 179 (2020) (2021) 498–506