

**PENERAPAN ALGORITMA *NAIVE BAYES* UNTUK
KLASIFIKASI KUALITAS PEMAIN SEPAK BOLA
BERDASARKAN DATA STATISTIK PERTANDINGAN**

LAPORAN TUGAS AKHIR

Laporan ini disusun untuk memenuhi salah satu syarat untuk menyelesaikan program studi Teknik Informatika S-1 pada Fakultas Teknologi Industri Universitas Islam Sultan Agung



Disusun Oleh:

Nama : Ivan Suryo Wicaksono

Nim : 32602000070

Program Studi : Teknik Informatika

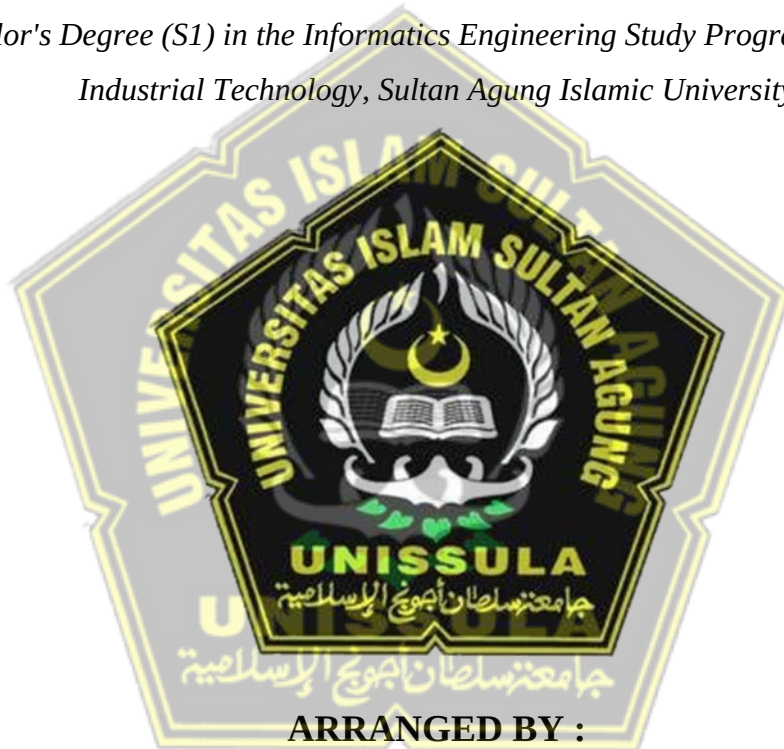
**FAKULTAS TEKNOLOGI INDUSTRI
UNIVERSITAS ISLAM SULTAN AGUNG
SEMARANG**

2025

**APPLICATION OF NAIVE BAYES ALGORITHM FOR
FOOTBALL PLAYER QUALITY CLASSIFICATION BASED
ON MATCH STATISTICAL DATA**

FINAL PROJECT REPORT

*This report has been prepared to fulfill one of the requirements for obtaining
Bachelor's Degree (S1) in the Informatics Engineering Study Program Faculty of
Industrial Technology, Sultan Agung Islamic University*



ARRANGED BY :

IVAN SURYO WICAKSONO

NIM 32602000070

**MAJORING OF INFORMATICS ENGINEERING
INDUSTRIAL TECHNOLOGY FACULTY
SULTAN AGUNG ISLAMIC UNIVERSITY**

2025

LEMBAR PENGESAHAN
TUGAS AKHIR

PENERAPAN ALGORITMA *NAIVE BAYES* UNTUK KLASIFIKASI
KUALITAS PEMAIN SEPAK BOLA BERDASARKAN DATA STATISTIK
PERTANDINGAN

IVAN SURYO WICAKSONO
NIM 32602000070

Telah dipertahankan di depan tim penguji ujian sarjana tugas akhir
Program Studi Teknik Informatika
Universitas Islam Sultan Agung
Pada tanggal : 25-08-2025

TIM PENGUJI UJIAN SARJANA :

Dedy Kurniadi, ST., M.Kom
NIK. 0622058802

(Ketua Penguji)

Andi Riansyah, ST., M.Kom
NIK. 0609108802

(Anggota Penguji)

Mustafa, ST., MM., M.Kom
NIK. 0623117703

(Pembimbing)

25-08-2025

22-08-2025

25-08-2025

Semarang, 25-08-2025

Mengetahui,

Kaprodi Teknik Informatika
Universitas Islam Sultan Agung

Moch Taufik, ST., MIT
NIK. 0622037502

SURAT PERNYATAAN KEASLIAN TUGAS AKHIR

Yang bertanda tangan dibawah ini :

Nama : Ivan Suryo Wicaksono

NIM : 32602000070

Judul Tugas Akhir : Penerapan Algoritma Naïve Bayes Untuk Klasifikasi Kualitas Pemain Sepak Bola Berdasarkan Data Statistik Pertandingan

Dengan bahwa ini saya menyatakan bahwa judul dan isi Tugas Akhir yang saya buat dalam rangka menyelesaikan Pendidikan Strata Satu (S1) Teknik Informatika tersebut adalah asli dan belum pernah diangkat, ditulis ataupun dipublikasikan oleh siapapun baik keseluruhan maupun sebagian, kecuali yang secara tertulis diacu dalam naskah ini dan disebutkan dalam daftar pustaka, dan apabila di kemudian hari ternyata terbukti bahwa judul Tugas Akhir tersebut pernah diangkat, ditulis ataupun dipublikasikan, maka saya bersedia dikenakan sanksi akademis. Demikian surat pernyataan ini saya buat dengan sadar dan penuh tanggung jawab.

Semarang, 25-08-2025

Yang Menyatakan,



Ivan Suryo Wicaksono

PERNYATAAN PERSETUJUAN PUBLIKASI KARYA ILMIAH

Saya yang bertanda tangan dibawah ini :

Nama : Ivan Suryo Wicaksono

NIM : 32602000070

Program Studi : Teknik Informatika

Fakultas : Teknologi industri

Alamat Asal : Semarang

Dengan ini menyatakan Karya Ilmiah berupa Tugas akhir dengan Judul : **Penerapan Algoritma Naïve Bayes Untuk Klasifikasi Kualitas Pemain Sepak Bola Berdasarkan Data Statistik Pertandingan**

Menyetujui menjadi hak milik Universitas Islam Sultan Agung serta memberikan Hak bebas Royalti Non-Eksklusif untuk disimpan, dialihmediakan, dikelola dan pangkalan data dan dipublikasikan diinternet dan media lain untuk kepentingan akademis selama tetap menyantumkan nama penulis sebagai pemilik hak cipta. Pernyataan ini saya buat dengan sungguh-sungguh. Apabila dikemudian hari terbukti ada pelanggaran Hak Cipta/Plagiarisme dalam karya ilmiah ini, maka segala bentuk tuntutan hukum yang timbul akan saya tanggung secara pribadi tanpa melibatkan Universitas Islam Sultan agung.

Semarang, 23-08-2023

Yang menyatakan,



Ivan Suryo Wicaksono

KATA PENGANTAR

Puji syukur penulis panjatkan kepada Allah SWT, yang telah memberikan rahmat, taufik serta hidayah-Nya, sehingga laporan Tugas Akhir ini dapat terselesaikan.

Tanpa lupa penulis mengucapkan terima kasih kepada :

1. Rektor UNISSULA Bapak Prof. Dr. H. Gunarto S.H., M.Hum yang mengizinkan penulis menimba ilmu di kampus ini.
2. Dekan Fakultas Teknologi Industri Ibu Dr. Hj. Novi Marlyana, ST., MT., IPU.
3. Dosen pembimbing 1 Bapak Mustafa, S.T., MM., M.Kom yang telah membimbing selama tugas akhir berlangsung.
4. Orang tua penulis yang telah mengizinkan untuk menyelesaikan laporan ini.
5. Dan kepada semua pihak yang tidak dapat saya sebutkan satu persatu.

Penulis menyadari bahwa dalam penyusunan laporan ini masih terdapat banyak kekurangan, untuk itu penulis mengharap kritik dan saran dari pembaca untuk sempurnanya laporan ini. Semoga dengan ditulisnya laporan ini dapat menjadi sumber ilmu bagi setiap pembaca.

Semarang, 19 Agustus 2025



Ivan Suryo Wicaksono

DAFTAR ISI

HALAMAN JUDUL	i
LEMBAR PENGESAHAN TUGAS AKHIR	ii
SURAT PERNYATAAN KEASLIAN TUGAS AKHIR	iii
PERNYATAAN PERSETUJUAN PUBLIKASI KARYA ILMIAH	iv
KATA PENGANTAR.....	v
DAFTAR ISI.....	vi
DAFTAR GAMBAR.....	viii
DAFTAR TABEL	x
ABSTRAK	xi
ABSTRACT	xi
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang:	1
1.2 Perumusan Masalah	3
1.3 Pembatasan Masalah	3
1.4 Tujuan Penelitian:	4
1.5 Manfaat	4
1.6 Sistematika Penelitian	4
BAB II TINJAUAN PUSTAKA DAN DASAR TEORI	6
2.1 Tinjauan Pustaka	6
2.2 Dasar Teori.....	7
2.2.1 Kualitas pemain sepak bola.....	7
2.2.2 Prediksi.....	7
2.2.3 <i>Naive Bayes</i>	8
2.2.4 Python	8
2.2.5 <i>Website</i>	9
BAB III METODE PENELITIAN	10
3.1 Metodologi Penelitian	10
3.1.1 Studi Literatur	11

3.1.2	Pengumpulan Data	12
3.1.3	Analisa kebutuhan.....	12
3.1.4	Perancangan Sistem	14
3.1.5	Penerapan Algoritma <i>Naïve Bayes</i>	16
3.2	Perancangan User Interface	34
3.2.1	<i>Login</i>	34
3.2.2	<i>Menu</i>	35
3.2.3	<i>Dashboard</i>	35
3.2.4	Klasifikasi Kualitas Pemain	36
3.2.5	Prediksi Kualitas Pemain	37
3.2.6	<i>Logout</i>	43
BAB IV	HASIL DAN ANALISIS PENELITIAN	45
4.1	Analisis Penelitian.....	45
4.1.1	Mempersiapkan data	45
4.1.2	Pembagian data training dan data testing.....	45
4.1.3	Evaluasi Model.....	54
4.2	Hasil Sistem	56
4.2.1	Halaman Login.....	56
4.2.2	Halaman <i>Menu</i>	56
4.2.3	Halaman Utama(<i>Dashboard</i>).....	57
4.2.4	Halaman Prediksi Kualitas Pemain Sepak bola	57
4.2.5	Halaman <i>Logout</i>	66
BAB V	KESIMPULAN DAN SARAN	68
5.1	Kesimpulan	68
5.2	Saran.....	68
DAFTAR PUSTAKA		70

DAFTAR GAMBAR

Gambar 2. 1 Python	8
Gambar 2. 2 <i>Streamlit</i>	9
Gambar 3. 1 <i>Flowchart</i> Penelitian	10
Gambar 3. 2 <i>Flowchart</i> Sistem	15
Gambar 3. 3 Kode <i>Drop Data</i>	18
Gambar 3. 4 Kode Label Encoder.....	19
Gambar 3. 5 Kode Klasifikasi Algoritma Naive Bayes	23
Gambar 3. 6 Kode Memisahkan Fitur x dan y	24
Gambar 3. 7 Kode <i>Library Sklearn</i>	25
Gambar 3. 8 Kode <i>Split Data</i>	25
Gambar 3. 9 Kode x <i>Train</i>	26
Gambar 3. 10 Kode y <i>Train</i>	26
Gambar 3. 11 Kode x <i>Test</i>	27
Gambar 3. 12 Kode Prediksi	28
Gambar 3. 13 Hasil Prediksi	28
Gambar 3. 14 Kode Probabilitas	28
Gambar 3. 15 Hasil Probabilitas	29
Gambar 3. 16 Kode Evaluasi <i>Akurasi</i>	31
Gambar 3. 17 Kode Evaluasi	31
Gambar 3. 18 Kode Laporan Klasifikasi.....	31
Gambar 3. 19 Hasil Klasifikasi Laporan.....	32
Gambar 3. 20 Kode <i>Konfusion Matrix</i>	32
Gambar 3. 21 <i>Confusion Matrix</i>	33
Gambar 3. 22 <i>Login</i>	34
Gambar 3. 23 <i>Menu</i>	35
Gambar 3. 24 <i>Dashboard</i>	35
Gambar 3. 25 Klasifikasi Algoritma <i>Naive Bayes</i>	36
Gambar 3. 26 <i>Input Data</i> Pemain.....	37
Gambar 3. 27 <i>Grafik Posisi</i> Pemain.....	37
Gambar 3. 28 Atribut Data.....	38
Gambar 3. 29 Menghapus Kolom	38
Gambar 3. 30 <i>Label Encoder</i>	39
Gambar 3. 31 Data <i>Training</i> Variabel x	40
Gambar 3. 32 Data <i>Training</i> Variabel y	40
Gambar 3. 33 Data <i>Testing</i>	41
Gambar 3. 34 Hasil Prediksi Pemain Sepak Bola	41
Gambar 3. 35 Probabilitas Prediksi.....	42

Gambar 3. 36 Evaluasi Model.....	42
Gambar 3. 37 Prediksi Pemain Berdasarkan <i>Input</i> manual.....	43
Gambar 3. 38 <i>Logout</i>	43
Gambar 3. 39 Prediksi Pemain Baru Berdasarkan <i>Input</i> Manual	66
Gambar 4. 1 <i>Login</i>	56
Gambar 4. 2 Halaman <i>Menu</i>	56
Gambar 4. 3 Halaman <i>Dashboard</i>	57
Gambar 4. 4 Halaman Prediksi Kualitas Pemain Sepak Bola.....	57
Gambar 4. 5 Upload Data.....	58
Gambar 4. 6 Data yang diunggah.....	58
Gambar 4. 7 Grafik Posisi Pemain.....	59
Gambar 4. 8 Atribut Data.....	59
Gambar 4. 9 Menghapus Kolom.....	60
Gambar 4. 10 Label Encoding	60
Gambar 4. 11 Tahap Klasifikasi <i>Naive Bayes</i>	61
Gambar 4. 12 Variabel <i>y</i>	61
Gambar 4. 13 Pembagian Data Training dan Testing.....	62
Gambar 4. 14 Training Variabel <i>y</i>	62
Gambar 4. 15 <i>Testing Data</i>	63
Gambar 4. 16 Hasil Prediksi	63
Gambar 4. 17 Probabilitas Prediksi.....	64
Gambar 4. 18 Hasil Evaluasi Model	65
Gambar 4. 19 <i>Confusion Matrix</i>	65
Gambar 4. 20 <i>Logout</i>	66

DAFTAR TABEL

Tabel 3. 1 Analisis Sistem.....	12
Tabel 3. 2 Menghapus gambar	18
Tabel 3. 3 Label Encoder	20
Tabel 3. 4 <i>Encoding</i> nominal	20
Tabel 3. 5 Encoding Ordinal	22
Tabel 3. 6 Fitur Variabel x dan y	25
Tabel 3. 7 x <i>Train</i>	26
Tabel 3. 8 y <i>Train</i>	27
Tabel 3. 9 <i>Testing</i>	27
Tabel 4. 1 Memuat Data.....	45
Tabel 4. 2 <i>Training Data</i>	46
Tabel 4. 3 <i>Testing Data</i>	46
Tabel 4. 4 Menghitung Probabilitas Kelas	46
Tabel 4. 5 Probabilitas Negara	46
Tabel 4. 6 Probabilitas Posisi	47
Tabel 4. 7 Probabilitas <i>Yellow Card</i>	47
Tabel 4. 8 Probabilitas <i>Red Card</i>	47
Tabel 4. 9 Probabilitas Gol	47
Tabel 4. 10 Probabilitas Asisst	48
Tabel 4. 11 Probabilitas Jumlah Pertandingan.....	48
Tabel 4. 12 Evaluasi Model	54
Tabel 4. 13 Hasil Prediksi	64

ABSTRAK

Sepak bola merupakan olahraga yang semakin berkembang tidak hanya dalam hal permainan, tetapi juga dalam pemanfaatan teknologi untuk menganalisis performa pemain. Penelitian ini bertujuan untuk mengklasifikasikan kualitas pemain sepak bola ke dalam kategori meningkat atau menurun berdasarkan data statistik pertandingan menggunakan algoritma *Naïve Bayes*. Data yang digunakan diperoleh dari platform Transfermarkt, mencakup variabel seperti jumlah gol, assist, kartu kuning, kartu merah, dan jumlah pertandingan. Dengan data historis sebanyak 1.343 entri (1.098 data latih, 274 data uji), model ini menghasilkan prediksi kategori Meningkatkan (456) dan Menurun (642), dengan akurasi sebesar 91%, presisi 91%, *recall* 91%, *f1-score* 91% yang menunjukkan efektivitas algoritma ini dalam mendukung pengambilan keputusan berbasis data bagi klub dan pelatih. Dengan sistem berbasis web menggunakan *Streamlit*, aplikasi ini diharapkan dapat mempermudah manajer tim dalam melakukan seleksi dan pemantauan kualitas pemain secara efisien dan objektif.

Kata kunci : klasifikasi, kualitas pemain, *naïve bayes*, sepak bola, statistik pertandingan

ABSTRACT

Football is a sport that continues to evolve, not only in terms of gameplay but also in the use of technology to analyze player performance. This study aims to classify football player quality into two categories—improving or declining—based on match statistics using the Naïve Bayes algorithm. The data, obtained from the Transfermarkt platform, includes variables such as goals, assists, yellow cards, red cards, and the number of matches played. Using a historical dataset of 1,343 entries (1,098 training data and 274 testing data), the model produced predictions of 456 “Improving” and 642 “Declining” cases, achieving an accuracy of 91%, precision of 91%, recall of 91%, and F1-score of 91%. These results demonstrate the effectiveness of the algorithm in supporting data-driven decision-making for clubs and coaches. With a web-based system built using Streamlit, this application is expected to assist team managers in selecting and monitoring player quality efficiently and objectively.

Keywords: classification, player quality, naïve bayes, football, match statistics

BAB I

PENDAHULUAN

1.1 Latar Belakang:

Sepak bola merupakan olahraga paling populer di dunia dan terus berkembang, tidak hanya dalam strategi permainan tetapi juga dalam penggunaan teknologi untuk menganalisis performa pemain. Seiring dengan meningkatnya ketersediaan data statistik pertandingan, klub dan pelatih semakin mengandalkan analisis berbasis data untuk mengukur kualitas pemain (P. Indarto dkk., 2021). Proses penilaian yang sebelumnya hanya mengandalkan observasi subjektif kini dapat diperkuat dengan metode berbasis machine learning yang memberikan hasil lebih objektif dan akurat.

Dalam dunia sepak bola profesional, berbagai aspek performa pemain, seperti jumlah gol, assist, kartu kuning, kartu merah, dan menit bermain, menjadi indikator utama dalam menentukan kontribusi seorang pemain terhadap tim. Namun, tanpa pendekatan analitis yang tepat, data ini sulit digunakan secara efektif untuk mengidentifikasi pola performa pemain. Oleh karena itu, diperlukan sistem klasifikasi yang dapat mengolah data statistik ini menjadi informasi yang berguna bagi klub, pelatih, dan manajer tim.

Machine learning menawarkan berbagai metode dalam analisis data, salah satunya adalah Naïve Bayes (Sinaga dkk., 2022), sebuah algoritma klasifikasi berbasis probabilitas yang sederhana namun efektif. Algoritma ini bekerja berdasarkan Teorema Bayes dengan asumsi independensi antar fitur. Dalam konteks sepak bola, Naïve Bayes dapat digunakan untuk mengklasifikasikan pemain berdasarkan performa mereka, membantu dalam pengambilan keputusan yang lebih baik dalam hal strategi tim maupun rekrutmen pemain.

Penggunaan Naïve Bayes dalam analisis sepak bola semakin menarik karena algoritma ini memiliki kecepatan komputasi yang tinggi serta mampu menangani dataset dengan jumlah variabel yang relatif banyak. Dengan membangun model klasifikasi menggunakan algoritma ini, diharapkan klub dapat mengidentifikasi pemain berkualitas berdasarkan data statistik pertandingan secara otomatis. Hal

ini akan mengurangi subjektivitas dalam penilaian dan meningkatkan efisiensi dalam proses seleksi pemain.

Selain itu, penelitian ini juga bertujuan untuk mengevaluasi performa algoritma Naïve Bayes dalam klasifikasi kualitas pemain sepak bola. Evaluasi dilakukan menggunakan metrik seperti akurasi, presisi, recall, dan F1-score untuk memastikan sejauh mana model yang dikembangkan mampu memberikan hasil yang valid (Erinsyah dkk., 2024). Dengan evaluasi yang tepat, model ini diharapkan dapat memberikan prediksi yang akurat dan membantu klub dalam mengambil keputusan berbasis data.

Dalam penelitian ini, data statistik pemain diperoleh dari situs transfermarkt, yang menyediakan beberapa informasi mengenai pertandingan sepak bola. Dengan menambahkan kolom "kualitas pemain" yang berisi label meningkat atau menurun, dataset yang sebelumnya hanya berisi statistik pertandingan kini memiliki target klasifikasi yang lebih jelas. Sebelumnya, dataset belum memiliki indikator langsung mengenai perubahan performa pemain, sehingga sulit untuk menentukan apakah seorang pemain mengalami peningkatan atau penurunan kualitas. Dengan adanya kolom ini, model Naïve Bayes dapat dilatih untuk memahami pola dari variabel statistik seperti jumlah gol, assist, kartu kuning, kartu merah, dan jumlah pertandingan, lalu mengklasifikasikan pemain berdasarkan tren performanya. Tujuan utama penambahan kolom ini adalah untuk menghasilkan target prediksi yang lebih konkret, sehingga klub, pelatih, atau pencari bakat dapat dengan mudah mengidentifikasi pemain yang mengalami perkembangan atau kemunduran dan mengambil keputusan yang lebih tepat dalam pengelolaan tim.

Data ini akan melalui proses preprocessing, termasuk normalisasi, sebelum digunakan untuk pelatihan model. Setelah itu, dataset akan dibagi menjadi data latih dan data uji untuk memastikan model dapat melakukan prediksi dengan baik.

Secara keseluruhan, penelitian ini bertujuan untuk membuktikan efektivitas algoritma Naïve Bayes dalam klasifikasi pemain sepak bola serta mengevaluasi tingkat akurasi. Dengan demikian, diharapkan bahwa metode ini dapat

digunakan sebagai alat bantu dalam pengambilan keputusan di dunia sepak bola profesional, baik dalam aspek teknis maupun manajemen tim.

Sistem ini dirancang khusus untuk mendukung kebutuhan beberapa manajer dalam melakukan pencarian dan pemantauan pemain berdasarkan kualitas performa mereka yang mengalami peningkatan atau penurunan. Dengan memanfaatkan data historis dan analisis performa secara berkala, sistem ini mampu memberikan informasi yang relevan dan terkini mengenai tren perkembangan setiap pemain

Hasil penelitian ini diharapkan dapat memberikan kontribusi bagi dunia sepak bola dalam mengembangkan sistem analisis berbasis kecerdasan buatan. Dengan pemanfaatan machine learning, proses evaluasi pemain dapat menjadi lebih efisien dan objektif, memungkinkan klub untuk lebih cepat dalam mengidentifikasi pemain berbakat serta menentukan strategi permainan yang optimal.

1.2 Perumusan Masalah

1. Bagaimana penerapan algoritma Naïve Bayes dalam mengklasifikasikan kualitas pemain sepak bola berdasarkan data statistik pertandingan?
2. Seberapa akurat hasil klasifikasi yang dihasilkan oleh algoritma Naïve Bayes?

1.3 Pembatasan Masalah

Untuk memastikan penelitian lebih terarah dan fokus, penelitian ini memiliki beberapa batasan sebagai berikut:

1. Data yang digunakan hanya mencakup statistik pemain dalam satu dataset tertentu.
2. Evaluasi model hanya menggunakan metrik akurasi, presisi, dan recall tanpa mempertimbangkan metode validasi lain seperti cross-validation.

1.4 Tujuan Penelitian:

1. Menerapkan algoritma Naïve Bayes untuk klasifikasi kualitas pemain sepak bola.
2. Mengevaluasi akurasi model yang dihasilkan.

1.5 Manfaat

Untuk Mengembangkan penerapan machine learning dalam olahraga, khususnya dalam analisis performa atlet sepak bola.

1.6 Sistematika Penelitian

Sistematika Penulisan untuk pembuatan laporan tugas akhir yang dilakukan oleh penulis adalah sebagai berikut:

BAB I : PENDAHULUAN

Pada bab ini penulis mengutarakan latar belakang pemilihan judul, rumusan masalah, tujuan penelitian, manfaat penelitian, serta sistematika penulisan.

BAB II : TINJAUAN PUSTAKA DAN DASAR TEORI

Pada Bab ini memuat penelitian – penelitian sebelumnya dan dasar teori yang berguna untuk membantu sebuah penulis untuk memahami bagaimana teori yang berhubungan dengan algoritma *naïve bayes* untuk penelitian ini.

BAB III : METODE PENELITIAN

Bab ini mengungkapkan proses tahapan – tahapan penelitian dimulai dari mendapatkan data hingga proses klasifikasi data yang ada.

BAB IV : HASIL PENELITIAN

Pada bab ini penulis mengungkapkan hasil penelitian yaitu hasil klasifikasi menggunakan *naïve bayes* beserta pengujian klasifikasi

BAB V : KESIMPULAN DAN SARAN

Bab ini penulis memaparkan kesimpulan proses penelitian dari awal hingga akhir.

BAB II

TINJAUAN PUSTAKA DAN DASAR TEORI

2.1 Tinjauan Pustaka

Penelitian ini dilakukan oleh Dito Dang Kurniawan dan rekan – rekannya Model ini mampu mengategorikan sentimen menjadi tiga jenis dengan akurasi 85%. Hasil evaluasi menunjukkan bahwa untuk sentimen negatif, precision mencapai 85%, recall 77%, dan F1-score 81%. Pada sentimen netral, precision sebesar 100%, recall 65%, dan F1-score 79%. Sementara itu, sentimen positif memiliki precision 82%, recall 98%, dan F1-score 89%.(Dang Kurniawan, 2025)

Dari penelitian Fabio Fahri Pratama., (2020) hasil rekomendasi dengan hasil akurasi sebesar 65%, sedangkan pengujian pemilihan starting eleven dilakukan menggunakan game Football Manager dengan melakukan pertandingan dengan pemilihan pemain secara default dan pemilihan pemain hasil rekomendasi masing-masing sebanyak sepuluh kali melawan tim dengan komposisi pemain yang sama, hasil dari pertandingan tersebut dihitung selisih (%) dari rata-rata rating pemain. Hasil yang diberikan setelah digunakan perekomendasi pemilihan pemain kenaikan rata-rata rating tim hanya naik sebesar 0.98%.(Pratama & Nurhasanah, 2020)

Penelitian yang dilakukan oleh M.Julkarnain dan rekan – rekannya untuk memprediksi kemungkinan mahasiswa lulus tepat waktu dengan menggunakan algoritma Naïve bayes. Hasil penelitian menunjukkan bahwa algoritma ini mampu memberikan prediksi yang signifikan dalam menentukan mahasiswa yang berpotensi lulus tepat waktu yang memiliki akurasi 94,31%, precision 91%, recall 95%, dan F-1 Score 93%.(julkarnain dkk., 2022)

Dari penelitian Ibrohim dkk., (2022) Penelitian ini mencakup pengukuran performa (akurasi, presisi, recall dan f-score) metode ANN dengan 304 data pasien penyakit jantung yang diperoleh dari pusat dataset Kaggle. Hasil dari pengukuran performa diperoleh nilai akurasi 73,77%, presisi 80,43%, recall 84,09% dan f1-score sebesar 82,22%(Pradana dkk., 2022).

Dari penelitian Kurnia Muludi(2023) data hasil penelitian, pada dataset berbahasa inggris CNN menghasilkan kinerja terbaik dengan nilai akurasi 94%, sementara nilai Naive Bayes hanya 79%. Begitupun pada dataset berbahasa Indonesia, CNN menghasilkan kinerja terbaik dengan nilai akurasi 91%, sementara hasil nilai Naive Bayes ada pada 75% (Humam dkk., 2023).

2.2 Dasar Teori

2.2.1 Kualitas pemain sepak bola

Kualitas pemain sepak bola merupakan indikator penting yang mencerminkan kemampuan dan kontribusi individu terhadap performa tim secara keseluruhan. Kualitas ini tidak hanya diukur dari aspek fisik dan teknik bermain, tetapi juga dari data statistik yang merepresentasikan kinerja pemain selama pertandingan, seperti jumlah gol, assist, kartu kuning, kartu merah, dan total menit bermain. Menurut (Musrifin & Bausad, 2020), analisis kualitas pemain secara objektif sangat penting dalam pengambilan keputusan strategis, baik oleh pelatih, manajemen klub, maupun pencari bakat. Dalam era sepak bola modern, penilaian kualitas pemain tidak lagi mengandalkan observasi subjektif semata, melainkan telah berkembang dengan dukungan teknologi dan metode analisis data. Melalui pendekatan data mining dan machine learning, kualitas pemain dapat diklasifikasikan dan diprediksi secara sistematis berdasarkan pola statistik historis. Dengan demikian, pengukuran kualitas pemain menjadi lebih terukur, transparan, dan efisien, serta dapat digunakan sebagai dasar dalam strategi rekrutmen dan pengembangan tim.

2.2.2 Prediksi

Prediksi adalah proses memperkirakan secara sistematis tentang apa yang kemungkinan besar akan terjadi dimasa depan berdasarkan informasi yang ada dimasa lalu dan sekarang, dengan tujuan meminimalkan kesalahan (perbedaan antara kejadian yang sebenarnya terjadi dan perkiraan).(Damanik dkk., 2021)

2.2.3 Naïve Bayes

Naïve Bayes adalah metode klasifikasi berbasis probabilitas yang didasarkan pada *Teorema Bayes* dengan asumsi independensi antar fitur. Algoritma ini banyak digunakan dalam klasifikasi teks, diagnosis medis, dan berbagai bidang lainnya. Dalam penelitian ini, *Naïve Bayes* digunakan untuk mengklasifikasikan kualitas pemain sepak bola berdasarkan statistik pertandingan mereka. (Sartika dkk., 2023)

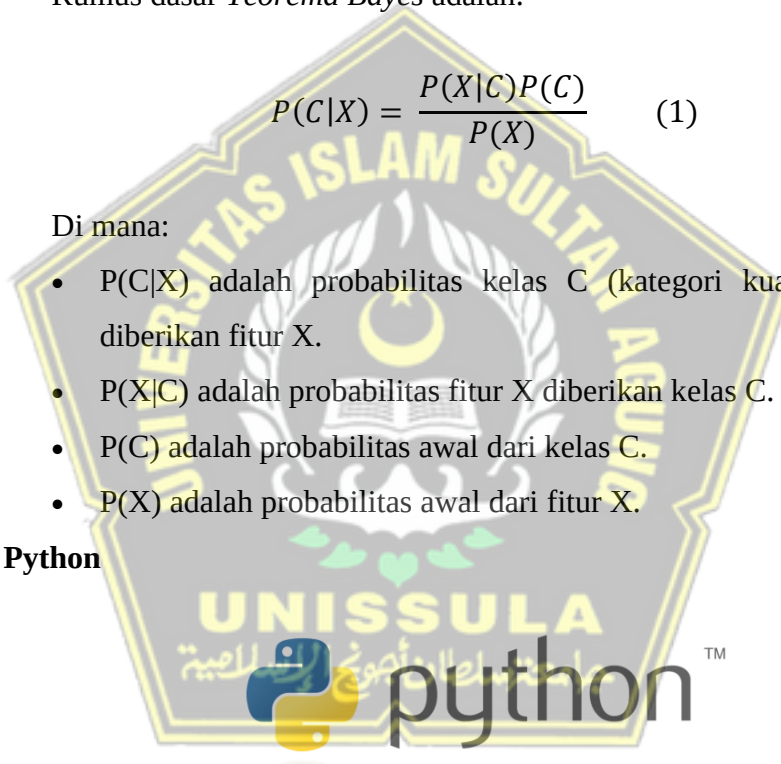
Rumus dasar *Teorema Bayes* adalah:

$$P(C|X) = \frac{P(X|C)P(C)}{P(X)} \quad (1)$$

Di mana:

- $P(C|X)$ adalah probabilitas kelas C (kategori kualitas pemain) diberikan fitur X .
- $P(X|C)$ adalah probabilitas fitur X diberikan kelas C .
- $P(C)$ adalah probabilitas awal dari kelas C .
- $P(X)$ adalah probabilitas awal dari fitur X .

2.2.4 Python



Gambar 2. 1 Python

Python adalah bahasa pemrograman *interpretatif* multiguna dengan filosofi perancangan yang berfokus pada tingkat keterbacaan kode. *Python* diklaim sebagai bahasa yang menggabungkan kapabilitas, kemampuan, dengan sintaksis kode yang besar serta *komperhensif*. *Python* juga didukung oleh komunitas yang besar. (Dania dkk., 2024)

Numpy adalah salah satu pustaka dalam bahasa pemrograman *python* yang digunakan untuk melakukan berbagai perhitungan matematis, seperti aljabar, statistika dan perhitungan kompleks lainnya (Putra dkk., 2023)

2.2.5 Website

Website adalah sekumpulan halaman *web* yang saling terkait dan diakses melalui internet menggunakan *browser web*. Halaman – halaman ini biasanya berbagi *domain* yang sama dan dapat diakses melalui alamat URL (*Unifrom Resource Locator*). (Wahyudin & Rahayu, 2020)



Gambar 2. 2 Streamlit

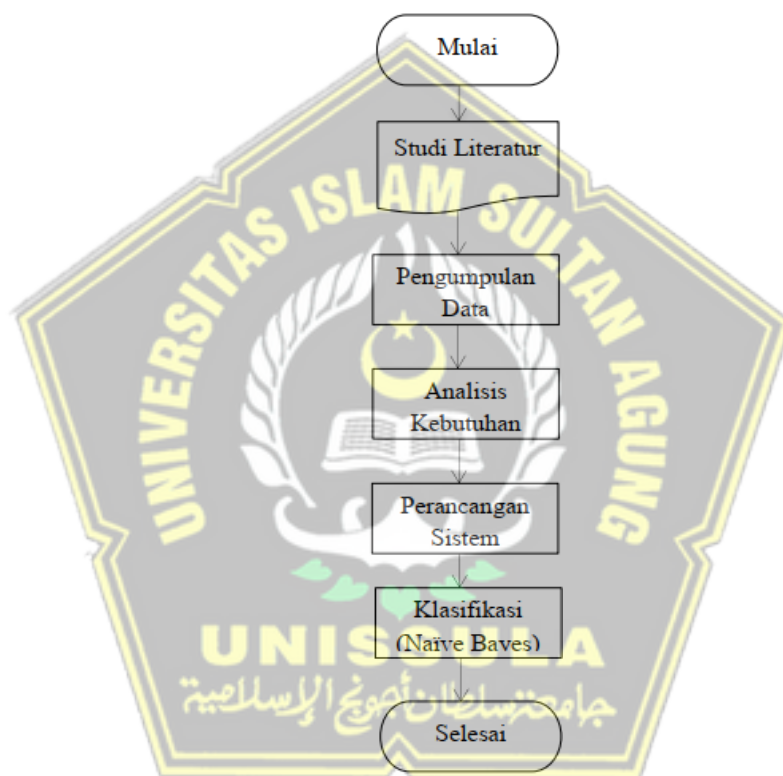
Streamlit adalah *Web Application Framework* untuk memudahkan *developer* dalam pengembangan dan pembuatan aplikasi web yang fokus di bidang *Machine Learning* dan *Data Science*. (Prasetyo & Laksana, 2022)

BAB III

METODE PENELITIAN

3.1 Metodologi Penelitian

Dalam Penelitian ini, diprediksi menggunakan metode *naïve bayes*. Metode ini digunakan untuk mengidentifikasi dan memprediksi performa pemain. Penelitian ini terdiri dari beberapa tahapan yang dilakukan secara sistematis. Berikut tahapan yang dilakukan pada penelitian ini:



Gambar 3. 1 Flowchart Penelitian

Pada Gambar 3. 1 menggambarkan *flowchart* proses analisis data yang komprehensif menggunakan kombinasi metode *Naïve Bayes* untuk mengelompokkan data untuk mengklasifikasikan data. Setiap tahapan disusun secara sistematis mulai dari pengumpulan data hingga evaluasi dan interpretasi hasil, sehingga dapat memberikan wawasan yang akurat dan bermanfaat dalam pengambilan keputusan berbasis data.

3.1.1 Studi Literatur

Dalam sepak bola *modern* diperlukan teknologi yang membuat sebuah klub mengetahui kualitas pemain yang akan direkrut. Teknologi ini menggunakan algoritma *naïve bayes* untuk memprediksi seorang pemain yang memiliki kualitas meningkat atau menurun.(Suryanto dkk., 2023)

Dalam penelitian ini terdapat Penghapusan kolom yang dilakukan sebagai bagian dari tahap pembersihan data (*data cleaning*) untuk meningkatkan kualitas dataset yang akan digunakan dalam Model klasifikasi. Kolom-kolom seperti `no`, `date`, `nama_pemain`, `club_id`, dan `harga_pemain` dihapus karena dianggap tidak memberikan kontribusi signifikan terhadap proses prediksi kualitas pemain, serta untuk menghindari redundansi data. Penghapusan kolom ini bertujuan untuk memastikan bahwa hanya fitur-fitur yang relevan dan bermakna yang dipertahankan dalam dataset, sehingga dapat meningkatkan efisiensi dan akurasi model Naïve Bayes dalam mengklasifikasikan performa pemain sepak bola(Astuti dkk., 2022).

Label Encoding atau pembersihan data dijelaskan sebagai tahap penting sebelum analisis dan pemodelan dilakukan. *Label Encoding* data mencakup beberapa langkah seperti penghapusan data duplikat, penanganan nilai kosong (*missing values*), koreksi kesalahan penulisan, dan standarisasi format data. Selain itu, dilakukan konversi tipe data kategorikal seperti nama pemain dan kompetisi menjadi format *numerik* menggunakan *metode encoding*, untuk memastikan data siap digunakan dalam proses pelatihan model *machine learning*. *Label Encoding* bertujuan untuk menghasilkan dataset yang konsisten, akurat, dan bebas dari noise, sehingga meningkatkan kinerja model klasifikasi *Naïve Bayes* yang digunakan dalam penelitian(Herdian dkk., 2024)

Sistem dalam penelitian ini memanfaatkan *google colab* untuk menganalisis dan menampilkan hasil prediksi. *Google colaboratory* atau sering disebut *google colab* adalah *platform* gratis dari *google* yang mendukung pembelajaran mesin. Layanan ini memungkinkan pengguna untuk menulis dan mengeksekusi kode *python*, serta mengunggah, menyimpan, dan berbagi kode maupun data (Al fatach dkk., 2024). Dalam tahap ini, dilakukan proses

pembersihan data sebelum diolah. Pembersihan mencakup penghapusan beberapa kolom yang tidak diperlukan serta perubahan nama *header* agar lebih sesuai

Penelitian ini bertujuan untuk mengembangkan sistem prediksi performa pemain sepak bola dengan model *naïve bayes* berbasis *web*. Kajian literatur menunjukkan bahwa metode *naïve bayes* sangat efektif untuk memprediksi pemain sepak bola. Diharapkan bahwa penggunaan *naïve bayes* dapat memprediksi performa pemain sepak bola secara akurat dan bermanfaat bagi pemilik klub untuk mengetahui pemain mana yang diprediksi laku pada tahun mendatang, sehingga pengelolaan performa pemain sepak bola dapat dilakukan secara efektif. Selain itu, penelitian ini juga berpotensi menjadi referensi untuk pengembangan sistem prediksi di masa depan.

3.1.2 Pengumpulan Data

Pengumpulan data dalam penelitian ini dilakukan untuk memperoleh informasi yang diperlukan untuk membangun model klasifikasi kualitas pemain sepak bola. Data yang digunakan merupakan data sekunder yang diperoleh dari sumber terpercaya, yaitu Data pemain sepak bola Liga Indonesia diperoleh dari situs Transfermarkt yang menyajikan informasi lengkap seperti profil, statistik, dan nilai pasar pemain. Transfermarkt menjadi sumber data utama karena menyediakan informasi yang akurat dan terkini mengenai pemain Liga Indonesia. (Laia dkk., 2023)

3.1.3 Analisa kebutuhan

Pada tahap penelitian analisis sistem, kebutuhan berbagai alat dan perangkat yang diperlukan untuk pengembangan sistem ini akan diidentifikasi. Berikut ini adalah alat dan perangkat yang akan digunakan tersedia dalam tabel

Tabel 3. 1 Analisis Sistem

No	Tool	Version
1	Laptop Asus A456U	Windows 10
2	Visual Studio Code	1.84.2
3	Excel	Office 2010
4	Microsoft	Office 2010
5	Google Colab	-
6	Web Browser	-
7	Figma	WEB

Untuk membangun aplikasi berbasis web menggunakan Bahasa pemrograman python, diperlukan beberapa *library* yang mendukung pengembangan system tersebut.

Adapun *library* yang digunakan dalam pembuatan sistem adalah sebagai berikut :

1. *Streamlit*

Streamlit adalah sebuah framework open – source yang digunakan untuk membangun antarmuka pengguna interaktif untuk aplikasi data science. Framework ini dirancang untuk mempermudah pengembangan aplikasi web dengan menggunakan python.

2. *Pandas*

Pandas adalah *library* yang biasanya digunakan untuk mempermudah dalam mengolah dan menganalisis data. Dalam *library* ini *pandas* berfungsi untuk mengolah data dalam bentuk *dataframe*, seperti membaca, menulis, dan memproses dataset. *Dataframe* adalah struktur data dua dimensi atau tabel yang terdiri dari baris dan kolom.

3. *Scikit – learn*

Scikit – learn adalah salah satu pustaka yang disediakan oleh *python* untuk memproses algoritma *machine learning*. Dalam pustaka ini terdapat fungsi *train_test_split* yang digunakan untuk membagi dataset menjadi data latih dan data uji dalam proses *machine learning*. Selain itu, terdapat *gaussian naïve bayes* (*GaussianNB*) yang digunakan sebagai model klasifikasi probabilistik. Pustaka ini menyediakan *classification_report* dan *confusion_matrix* yang berfungsi untuk mengukur kinerja model berdasarkan metrik seperti akurasi, presisi, *recall* dan *f1 – score*. Selanjutnya, *labelencoder* digunakan untuk mengonversi data kategorikal menjadi angka sehingga dapat digunakan dalam model *machine learning*.

4. *Matplotlib*

Matplotlib merupakan salah satu pustaka visualisasi data yang paling umum digunakan di *python*. Pustaka ini merupakan ekstensi matematika

numerik dari pustaka numpy. Matplotlib ini digunakan untuk membuat grafik dan visualisasi data.

5. Seaborn

Seaborn adalah *library* visualisasi data berbasis *matplotlib* di python yang digunakan untuk membuat grafik yang lebih informatif dan menarik secara visual. Seaborn menyediakan antarmuka untuk menggambarkan grafik statistik seperti *heatmap confusion matrix*.

6. Plotly express

Plotly express merupakan visualisasi data berbasis plotly di python yang memungkinkan pembuatan grafik interaktif dengan kode yang sederhana dibandingkan dengan *plotly* standar. *Plotly express* ini digunakan untuk membuat grafik interaktif dalam aplikasi web.

7. Numpy

Numpy merupakan salah satu *library* python yang berfungsi untuk proses komputasi *numerik*. Numpy sendiri digunakan untuk mengolah data *numerik* seperti *array* dan perhitungan matematika.

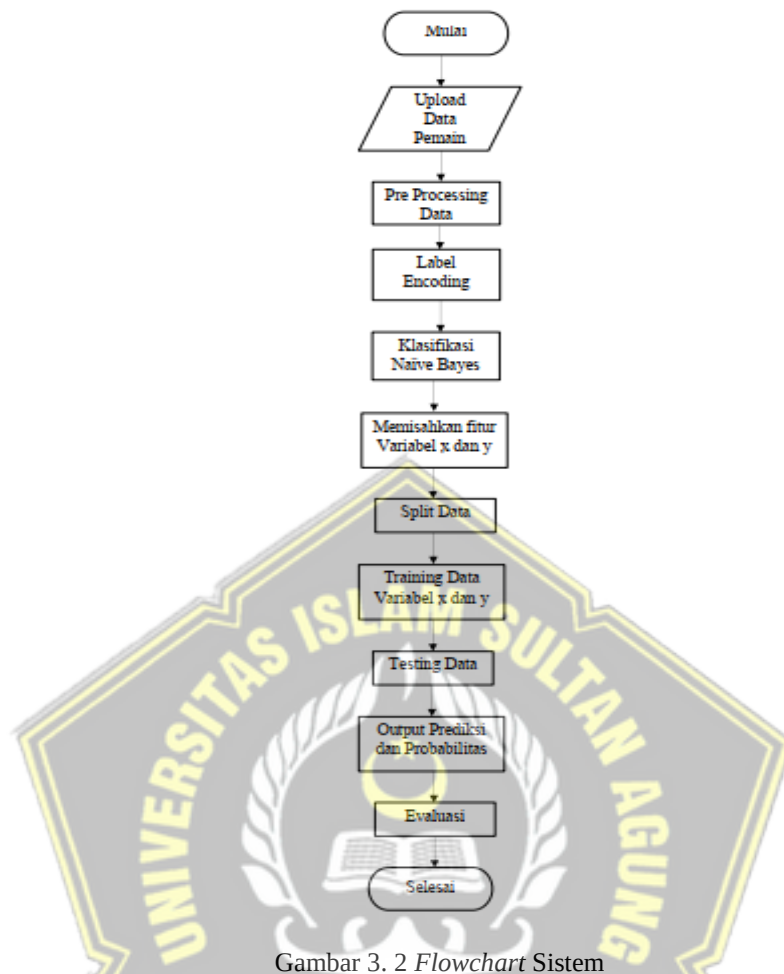
8. Joblib

Joblib adalah *library* python yang digunakan untuk penyimpanan dan pemrosesan paralel yang efisien. *Library* ini digunakan dalam *meachine learning* untuk menyimpan model yang sudah dilatih agar dapat digunakan kembali tanpa perlu melatih ulang kembali.

3.1.4 Perancangan Sistem

1. Analisis Alur Sistem

Dalam analisis alur sistem ini, terdapat *flowchart* yang menguraikan langkah - langkah urutan operasi dari sistem penelitian ini. Proses ini dimulai dari tahap penginputan data hingga menghasilkan keluaran berupa prediksi yang dihasilkan oleh sistem. *Flowchart* ini dapat dilihat pada gambar 3. 2 .



Gambar 3. 2 Flowchart Sistem

Pada gambar 3. 2 dalam *flowchart* sistem, data kualitas pemain sepak bola diinputkan ke dalam sistem. Tahap pertama dalam pemrosesan transformasi data agar siap digunakan dalam proses klasifikasi setelah itu Klasifikasi *Naïve Bayes* Menentukan bahwa metode yang digunakan untuk klasifikasi adalah *Naïve Bayes*, sebuah algoritma berbasis probabilitas. Pada tahap ini, dilakukan persiapan Variabel X dan variabel Y . setelah *variabel* disiapkan, data dibagi menjadi dua bagian yaitu data latih dan data uji. Data latih digunakan untuk melatih model pada *variabel* x dan y . setelah proses pelatihan selesai, model diuji menggunakan data uji untuk menghasilkan prediksi serta probabilitas dari data yang diuji. Tahap terakhir adalah evaluasi hasil prediksi untuk menilai performa model. Setelah melakukan evaluasi proses prediksi telah selesai.

3.1.5 Penerapan Algoritma *Naïve Bayes*

Dalam penelitian ini, proses pengolahan data untuk membangun model prediksi kualitas pemain sepak bola dilakukan menggunakan *google colab* yang berfungsi sebagai *platform* untuk melatih dan menguji guna memperoleh hasil prediksi.

1. Data Statistik Pemain

Penelitian ini menggunakan data statistik pemain untuk mengetahui pemain mana yang mempunyai kualitas permainan bermain sepak bola untuk sebuah klub. Proses input data pada sistem ini adalah mengupload data pemain dengan file excel kemudian menampilkan data pemain yang sesuai dengan file excel. Pada system ini menggunakan *library* dari *pandas*, dan *numpy*.

2. *Preprocessing Data*

Preprocessing data merupakan tahap awal dalam analisis data yang bertujuan untuk menyiapkan data agar dapat digunakan dalam model prediksi. Model *development* (pengembangan model) dimana proses ini membangun, melatih dan mengoptimalkan model *machine learning* menggunakan data yang telah diproses. Proses awal pada system ini dimulai dengan menghapus beberapa kolom seperti *no*, *date*, *nama_pemain*, *club_id*, dan *harga_pemain*. Setelah itu, sistem menyiapkan kolom *negara*, *posisi*, *yellow_card*, *red_card*, *gol*, *assists*, *jumlah_pertandingan*, dan *kualitas_pemain* yang digunakan sebagai fitur dalam proses klasifikasi menggunakan metode *naïve bayes*.

Berikut penjelasan penghapusan kolom disertai penomoran per kolom:

a. *no*

Kolom ini hanya berfungsi sebagai *indeks* atau nomor urut data dalam dataset. Karena tidak memiliki nilai informasi yang relevan terhadap variabel target (*kualitas_pemain*), maka kolom ini tidak diperlukan dalam proses analisis dan klasifikasi.

b. *date*

Walaupun berisi informasi tanggal, kolom ini tidak relevan secara langsung terhadap penentuan kualitas pemain karena waktu pertandingan

bukan merupakan fitur yang memengaruhi performa pemain dalam model ini. Selain itu, kolom ini bersifat *non-numerik* dan tidak kontributif secara langsung terhadap fitur statistik seperti gol atau assist.

c. nama_pemain

Kolom nama_pemain dihapus dari proses pengembangan model karena tidak memiliki nilai prediktif dan hanya berfungsi sebagai identitas. Jika dimasukkan, bisa menyebabkan overfitting karena model bisa menghafal nama, bukan pola data statistik. Namun, nama_pemain tetap ditampilkan kembali setelah prediksi untuk memudahkan interpretasi hasil oleh pengguna.

d. club_id

Merupakan identitas klub dalam bentuk *numerik* atau *string* tertentu. Informasi ini tidak digunakan secara langsung dalam klasifikasi kualitas pemain karena analisis difokuskan pada performa individu (statistik pertandingan) bukan afiliasi klubnya. Penghapusan kolom ini juga mencegah bias terhadap klub tertentu.

e. harga_pemain

Harga pemain bersifat *fluktuatif* dan ditentukan oleh banyak faktor di luar statistik pertandingan (misalnya popularitas, usia, atau kontrak). Kolom ini tidak dijadikan variabel *prediktor* karena dapat menimbulkan bias, dan tidak selalu mencerminkan kualitas performa aktual seorang pemain.

Selanjutnya data yang relevan, seperti negara, posisi, yellow_card, red_card, gol, assists, dan jumlah_pertandingan, dipersiapkan untuk diproses dalam sistem. Faktor – faktor yang mempengaruhi kualitas pemain seperti yellow_card, red_card, gol, assists, dan jumlah_pertandingan akan dianalisis untuk melihat pengaruhnya terhadap tingkat kualitas pemain. Setelah itu, model *naive bayes* dilatih dengan mempertimbangkan faktor – faktor yang mempengaruhi kualitas pemain, kemudian diuji menggunakan data *testing* untuk mengevaluasi kinerja model berdasarkan *akurasi*, *confusion matrix*, dan *clasification report*.


```
[ ] # Menghapus kolom yang tidak dibutuhkan
kolom_dihapus = ['no','date ','nama_pemain','club_id','harga_pemain'] # Fixed typo: 'apperance_id' to 'appearance_id'
df_bersih = data.drop(columns=kolom_dihapus)
```

Gambar 3. 3 Kode Drop Data

Pada gambar 3. 3 kode tersebut berfungsi untuk menghapus kolom yang tidak diperlukan dari sebuah *dataframe* bernama data, dengan terlebih dahulu memeriksa keberadaan kolom yang akan dihapus agar tidak terjadi error.

Tabel 3. 2 Menghapus gambar

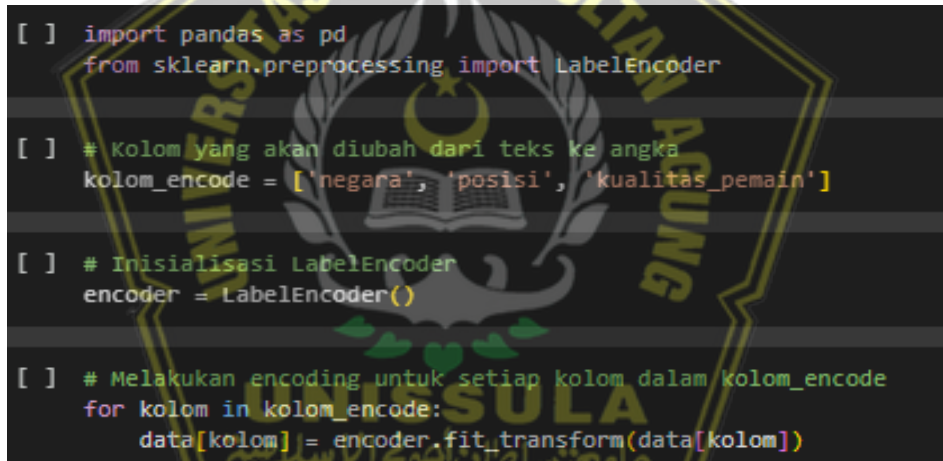
Negara	Posisi	yellow card	red card	Gol	assist	jumlah pertandingan	kualitas pemain
Indonesia	Kiper	2	1	0	0	24	Meningkat
Indonesia	Kiper	2	0	0	0	12	Meningkat
Makedonia Utara	Bek Tengah	2	0	0	0	16	Meningkat
Serbia	Bek Tengah	2	1	0	1	29	Meningkat
Indonesia	Bek Kanan	6	1	0	0	29	Meningkat
...	...	...	...	...	...	...	...
Indonesia	Sayap Kanan	0	0	0	1	13	Meningkat
Indonesia	Sayap Kanan	0	0	1	0	4	Meningkat
Indonesia	Depan Tengah	2	0	0	0	16	Meningkat
Indonesia	Depan Tengah	1	0	1	1	4	Meningkat
Indonesia	Depan Tengah	1	0	3	0	10	Meningkat

Tabel 3. 2 merupakan proses setelah menghapus kolom no, date, nama_pemain, club_id, dan harga_pemain data yang akan diproses ke dalam sistem mencakup negara, posisi, yellow_card, red_card, gol, assist dan jumlah pertandingan.

3. Label Encoding

Dalam proses preprocessing data untuk klasifikasi menggunakan algoritma *Naïve Bayes*, diperlukan konversi terhadap data kategorikal (teks) menjadi data *numerik* agar dapat diproses oleh algoritma *machine learning*. Salah satu teknik yang digunakan dalam penelitian ini adalah *Label Encoding*.

Label Encoding merupakan metode transformasi data yang digunakan untuk mengubah nilai kategorikal (seperti nama negara, posisi pemain, atau label kualitas pemain) ke dalam bentuk angka. Setiap nilai unik pada fitur kategorikal akan diubah menjadi bilangan bulat yang berbeda. Teknik ini sangat berguna terutama ketika nilai-nilai kategorikal tidak memiliki urutan yang eksplisit, namun diperlukan format *numerik* untuk diproses oleh algoritma klasifikasi.



```
[ ] import pandas as pd
    from sklearn.preprocessing import LabelEncoder

[ ] # Kolom yang akan diubah dari teks ke angka
    kolom_encode = ['negara', 'posisi', 'kualitas_pemain']

[ ] # Inisialisasi LabelEncoder
    encoder = LabelEncoder()

[ ] # Melakukan encoding untuk setiap kolom dalam kolom_encode
    for kolom in kolom_encode:
        data[kolom] = encoder.fit_transform(data[kolom])
```

Gambar 3. 4 Kode Label Encoder

Gambar 3. 4 kode ini melakukan proses label encoder dengan mengganti nilai unik dalam beberapa kolom kategorikal dengan angka. Langkah – langkahnya adalah sebagai berikut :

- Menggunakan pandas untuk memanipulasi data dalam bentuk tabel (DataFrame)
- Menggunakan *LabelEncoder* digunakan untuk mengubah data kategori(teks) menjadi angka
- Mengubah kolom teks menjadi numerik, kolom yang dirubah adalah negara, posisi, kualitas pemain

- d. Membuat objek *LabelEncoder* yang nanti digunakan untuk proses encoding
- e. Melakukan loop untuk setiap kolom dalam kolom_encode
- f. *Fit_transform()* akan memetakan setiap nilai unik di kolom menjadi angka, lalu menggantinya di DataFrame data

Tabel 3. 3 Label Encoder

Negara	posisi	Yellow card	Red Card	Gol	Assist	Jumlah pertandingan	Kualitas pemain
18	12	2	1	0	0	24	0
18	12	2	0	0	0	12	0
29	4	2	0	0	0	16	0
47		2	1	0	1	29	0
18	2	6	1	0	0	29	0
...	...	...	...	...	...	...	...
18	14	0	0	0	1	13	0
18	14	0	0	1	0	4	0
18	6	2	0	0	0	16	0
18	6	1	0	1	1	4	0
18	6	1	0	3	0	10	0

Tabel 3. 3 proses mengubah data dari teks menjadi *numerik* pada kolom negara, posisi, kualitas_pemain dengan melalui proses *LabelEncoder*.

Dalam Label Encoding, terdapat dua jenis encoding yang digunakan, yaitu Encoding Nominal dan Encoding Ordinal. Berikut adalah penjelasannya:

a. Encoding nominal

Encoding nominal adalah proses mengubah data kategorikal tanpa urutan menjadi bentuk numerik agar bisa diproses oleh algoritma machine learning. Data nominal biasanya berupa label atau kategori yang memiliki makna berbeda, tetapi tidak memiliki hubungan peringkat. Yang merupakan encoding nominal adalah negara, dan posisi.

Tabel 3. 4 Encoding nominal

negara	Posisi
18	12
18	12
29	4
47	4
18	2

Tabel 3. 4 merupakan hasil dari encoding nominal, yaitu mengubah data kategori non-numerik seperti negara, dan posisi menjadi bentuk angka menggunakan teknik label encoding. Proses ini dilakukan agar data tersebut dapat diproses oleh algoritma machine learning yang hanya menerima input numerik.

Berikut penjelasan yang merupakan encoding nominal

1. Negara

hasil encoding dari nama negara (misal "Brazil", "Argentina", "Indonesia") menjadi angka. Karena negara tidak memiliki urutan tertentu, ini termasuk kategori nominal.

2. Posisi

meskipun posisi pemain bisa memiliki peran yang berbeda seperti kiper, bek, gelandang, atau penyerang, namun tidak ada urutan atau ranking di antara posisi tersebut. Maka jika posisi tersebut diubah menjadi angka, itu juga tergolong encoding nominal.

b. Encoding Ordinal

Encoding Ordinal adalah data kategorikal yang memiliki urutan atau tingkatan. Setiap kategori diberikan nilai numerik berdasarkan tingkatan atau skala tertentu, sehingga urutan antar kategori tetap terjaga secara matematis. Yang merupakan encoding ordinal adalah yellow_card, red_card, gol, assist, jumlah_pertandingan, dan kualitas_pemain

Tabel 3. 5 Encoding Ordinal

yellow_card	red_card	Gol	Assist	jumlah_pertandingan	kualitas_pemain
2	1	0	1	13	1
2	0	1	0	16	0
2	0	2	0	8	0
2	1	3	1	27	0
6	1	0	0	0	1

Pada tabel 3. 5 merupakan hasil encoding ordinal, di mana data kategori yang memiliki urutan atau tingkatan (ordinal) diubah menjadi bentuk angka.

Berikut penjelasan yang merupakan mengenai encoding Ordinal

1. Yellow_card

Merupakan data ordinal, karena terdapat tingkatan pada setiap kategori. Kartu kuning diberikan kepada pemain sebagai peringatan akibat melakukan pelanggaran seperti bermain kasar, protes berlebihan, atau mengulur waktu. Meskipun tidak langsung dikeluarkan dari pertandingan, akumulasi kartu kuning dapat menyebabkan sanksi berupa larangan bermain pada pertandingan berikutnya. Pemain yang sering menerima kartu kuning menunjukkan kurangnya disiplin, yang dapat memengaruhi kontribusinya secara keseluruhan terhadap tim.

2. Red_card

Merupakan data ordinal, karena terdapat tingkatan pada setiap kategori. Kartu merah dalam sepak bola diberikan kepada pemain yang melakukan pelanggaran serius atau tindakan tidak sportif, yang mengakibatkan pemain tersebut langsung dikeluarkan dari lapangan dan tidak dapat melanjutkan pertandingan. Pemain yang sering mendapat kartu merah dianggap kurang disiplin dan memberikan kontribusi yang buruk bagi klub

3. Gol

Kolom gol merupakan encoding ordinal karena terdapat tingkatan pada setiap kategori. Seperti dalam setiap pertandingan bersama klubnya, seorang pemain sepak bola dapat mencatatkan gol. Jumlah gol yang tinggi menunjukkan kontribusi besar terhadap tim, sehingga semakin banyak gol yang dimiliki seorang pemain, maka semakin baik pula kualitasnya

4. Assist

Kolom assist merupakan encoding ordinal karena terdapat tingkatan pada setiap kategori. Seperti dalam setiap pertandingan bersama klubnya, seorang pemain sepak bola dapat mencatatkan assist. Jumlah assist yang tinggi menunjukkan kontribusi besar terhadap tim, sehingga semakin banyak assist yang dimiliki seorang pemain, maka semakin baik pula kualitasnya

5. Jumlah_pertandingan

Kolom jumlah_pertandingan merupakan encoding ordinal karena terdapat tingkatan pada setiap kategori. Seperti setiap pemain mempunyai beberapa jumlah_pertandingan pada saat bermain di sebuah club. Lebih banyak seorang pemain sepak bola mempunyai jumlah pertandingan lebih bagus bagus pemain itu.

6. Kualitas_pemain

Kolom kualitas_pemain merupakan penilaian performa pemain yang telah dikonversi ke dalam bentuk numerik menggunakan encoding ordinal. Nilai pada kolom ini menunjukkan tingkat kualitas pemain berdasarkan kontribusinya di lapangan, seperti jumlah gol, assist, dan jumlah pertandingan yang dimainkan. Dalam data tersebut, nilai 0(meningkat) dan 1(menurun) merepresentasikan kategori kualitas. Karena terdapat urutan atau tingkatan antara nilai-nilai tersebut (meningkat lebih tinggi dari menurun), maka encoding ini termasuk jenis ordinal.

4. Klasifikasi Algoritma Naïve Bayes

Pada tahap klasifikasi, algoritma Naive Bayes diimplementasikan menggunakan library GaussianNB dari sklearn.naive_bayes, di mana model dilatih dengan data fitur dan label untuk kemudian digunakan dalam memprediksi kelas dari data baru berdasarkan distribusi probabilitas Gaussian.

```
[ ] #import algoritma naive bayes
    from sklearn.naive_bayes import GaussianNB

[ ] #import model gaussian
    nbtrain = GaussianNB()
```

Gambar 3. 5 Kode Klasifikasi Algoritma Naive Bayes

Gambar 3. 5 kode ini mengimpor kelas *gaussiannb* dari modul *sklearn.naive_bayes*, yang merupakan implementasi algoritma *naïve bayes* dengan distribusi *gaussian*. Selanjutnya, membuat sebuah *instance* dari model *gaussiannb* dengan nama *nbtrain*, yang artinya dapat digunakan untuk pelatihan dan prediksi data.

5. Memisahkan Fitur Variabel x dan y

Dalam pemodelan ini, variabel X terdiri dari negara, posisi, *yellow_cards*, *red_cards*, gol, *assists*, dan jumlah_pertandingan, yang merupakan fitur-fitur input. Sementara itu, variabel Y(label) adalah *kualitas_pemain*, yang menjadi target atau output dari model.

```
[ ] # Memisahkan fitur (X) dan target (y)
x = data.drop(["kualitas_pemain"], axis=1)
y = data["kualitas_pemain"]
```

Gambar 3. 6 Kode Memisahkan Fitur x dan y

Gambar 3. 6 Kode tersebut untuk memisahkan data menjadi dua bagian, yaitu fitur (X) dan target (y). Pada proses ini, kolom "kualitas_pemain" dihapus dari *DataFrame* dan disimpan dalam variabel x karena kolom tersebut merupakan target yang ingin diprediksi, bukan bagian dari fitur. Sementara itu, kolom "kualitas_pemain" disimpan dalam variabel y sebagai target atau label. Pemisahan ini penting dalam proses *machine learning* agar model dapat dilatih menggunakan fitur sebagai *input* dan mempelajari hubungan terhadap target sebagai *output*.

Tabel 3. 6 Fitur Variabel x dan y

Negara	posisi	Yellow card	red card	Goal	assist	jumlah pertandingan	Kualitas Pemain
18	12	2	1	0	0	24	0
18	12	2	0	0	0	12	0
29	4	2	0	0	0	16	0
47	4	2	1	0	1	29	0
18	2	6	1	0	0	29	0
...	...	...	...	...	...	...	...
18	14	0	0	0	1	13	0
18	14	0	0	1	0	4	0
18	6	2	0	0	0	16	0
18	6	1	0	1	1	4	0
18	6	1	0	3	0	10	0

Tabel 3. 4 merupakan kode diatas merupakan kode memisahkan fitur x dan y .

6. Split Data

Dalam pembagian data ini, proses dilakukan secara acak sebanyak 80% data digunakan untuk pelatihan, sementara 20% data digunakan untuk pengujian.

```
[ ] #import library meachine learning sklearn
    from sklearn.model_selection import train_test_split
```

Gambar 3. 7 Kode Library Sklearn

Gambar 3. 7 kode tersebut mengimpor modul *train_test_split* dari *scikit-learn* (*sklearn*), yang digunakan untuk membagi dataset menjadi data pelatihan (*training*) dan data pengujian (*testing*) dalam proses *meachine learning*.

```
[ ] #meload data x dan y ke dalam data training dan juga data test
    x_train, x_test, y_train, y_test = train_test_split(x, y, test_size = 0.2, random_state = 123)
```

Gambar 3. 8 Kode Split Data

Gambar 3. 8 kode tersebut membagi dataset menjadi data pelatihan (x_{train} , y_{train}) dan data pengujian (x_{test} , y_{test}) menggunakan fungsi

`train_test_split()`, dimana 20% data digunakan untuk pengujian (`test_size=0.2`), sementara 80% digunakan pelatihan. parameter `random_state=123` memastikan bahwa pembagian data selalu konsisten setiap kali kode dijalankan.

7. *Training Data Variabel x dan y*

Data yang digunakan untuk melatih model terdiri dari *variabel x* dan *y*, yang dipilih secara acak. Sebanyak 80% dari total data, yaitu $0,8 * 1.373 = 1.098$ digunakan sebagai data *training* untuk proses pembelajaran model.

```
x_train.head(5)
```

Gambar 3. 9 Kode *x Train*

Gambar 3. 9 kode `x_train` berfungsi untuk menampilkan lima baris pertama dari *dataframe* `x_train`, yang merupakan bagian dari data pelatihan setelah pembagian dataset menggunakan `train_test_split()`.

Tabel 3. 7 *x Train*

Negara	Posisi	yellow card	Red Card	Go I	Assist	Jumlah pertandingan
18	2	4	0	0	1	13
8	4	4	0	1	0	16
18	15	0	0	2	0	8
8	9	7	0	3	1	27
18	12	0	0	0	0	0

Tabel 3. 5 menampilkan lima data dari `x_train` setelah melalui proses pembagian data secara acak untuk melatih data.

```
y_train.head(5)
```

Gambar 3. 10 Kode *y Train*

Gambar 3. 10 kode `y_train` digunakan untuk menampilkan lima baris pertama dari *dataframe* atau *series* `y_train`, yang berisi data target (label) dari data pelatihan setelah pembagian dataset menggunakan `train_test_split()`.

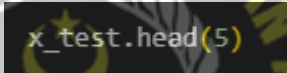
Tabel 3. 8 *y Train*

Kualitas pemain
1
0
0
0
1

Tabel 3. 6 menampilkan lima data dari *y_train* setelah melalui proses pembagian data secara acak untuk melatih data.

8. Testing Data

Data yang digunakan untuk pengujian dipilih secara acak sebanyak 20% dari total data. Sebanyak $0,2 * 1373 = 274$ data digunakan sebagai data *testing* untuk mengevaluasi apakah model dapat memprediksi kategori meningkat atau menurun.



```
x_test.head(5)
```

Gambar 3. 11 Kode *x Test*

Gambar 3. 11 kode *x_test* digunakan untuk menampilkan lima baris pertama dari dataframe *x_test*, yang berisi data uji untuk variabel independen setelah proses pembagian dataset menggunakan *train_test_split()*.

Tabel 3. 9 *Testing*

Negara	Posisi	yellow card	Red Card	gol	assist	jumlah pertandingan
18	4	0	0	0	0	9
18	10	6	0	0	2	17
18	15	1	0	0	0	8
18	15	0	0	0	0	8
18	6	2	0	8	1	22

Tabel 3. 7 menampilkan lima baris data uji yang akan digunakan untuk menguji dan menghasilkan prediksi apakah suatu kualitas pemain meningkat atau menurun.

9. Output Prediksi dan Probabilitas

Hasil prediksi pada *testing* berupa nilai probabilitas yang menunjukkan kemungkinan suatu data masuk ke dalam kategori meningkat atau menurun. Hasil dari *output* ini dari *testing* yang berkategori meningkat atau menurun.

```
#prediksi
y_pred = nbtrain.predict(x_test)
y_pred
```

Gambar 3. 12 Kode Prediksi

Gambar 3. 12 kode ini menggunakan model *naïve bayes* yang telah dilatih (*nbtrain*) untuk memprediksi nilai berdasarkan data uji(*x_test*) dan menyimpan hasilnya dalam variabel *y_pred*. setelah itu, *y_pred* akan menampilkan hasil prediksi tersebut.

```
array([1, 0, 1, 1, 0, 1, 1, 0, 1, 0, 1, 1, 0, 1, 1, 1, 1, 0, 0, 0, 1,
       1, 1, 1, 0, 1, 0, 1, 0, 0, 1, 1, 1, 1, 1, 0, 0, 1, 0, 0, 0, 1, 0,
       0, 1, 1, 1, 1, 1, 0, 1, 1, 0, 1, 1, 1, 0, 1, 1, 0, 1, 0, 0,
       1, 1, 1, 1, 0, 1, 0, 0, 1, 0, 1, 1, 0, 0, 1, 1, 0, 0, 1, 0, 1, 1,
       1, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 1, 0, 1, 1, 0, 0, 1, 1, 0, 0,
       1, 1, 1, 1, 1, 1, 0, 0, 1, 1, 1, 0, 1, 1, 1, 1, 1, 0, 1, 1, 0,
       1, 1, 1, 0, 0, 1, 1, 1, 1, 0, 0, 1, 1, 1, 1, 0, 0, 1, 1, 1, 0, 1,
       0, 1, 0, 0, 1, 1, 0, 1, 1, 1, 0, 0, 1, 1, 1, 1, 0, 0, 1, 0, 1, 0,
       1, 1, 0, 0, 1, 1, 0, 0, 0, 0, 1, 1, 0, 0, 0, 1, 1, 1, 0, 1, 0, 0,
       1, 1, 1, 1, 0, 1, 0, 0, 1, 1, 1, 1, 1, 0, 0, 0, 1, 0, 1, 1, 0,
       0, 1, 1, 0, 0, 1, 0, 1, 0, 1, 0, 1, 1, 1, 0, 0, 1, 1, 1, 0,
       1, 1, 1, 0, 1, 0, 1, 1, 1, 0, 1, 1, 0, 1, 0, 1, 1, 1, 0, 1, 1, 1,
       1, 0, 1, 1, 0, 0, 1, 0, 1, 1, 0])
```

Gambar 3. 13 Hasil Prediksi

Gambar 3. 13 merupakan hasil prediksi ditampilkan dalam bentuk *array* yang berisi kode 0 (meningkat) dan kode 1 (menurun) dengan total sebanyak 275 data.

```
#probabilitas dari data test
probabilitas
```

Gambar 3. 14 Kode Probabilitas

Gambar 3. 14 kode ini bertujuan untuk menampilkan probabilitas hasil prediksi dari data uji. Probabilitas ini menunjukkan seberapa besar

kemungkinan setiap sampel dalam data uji termasuk ke dalam masing – masing kelas.

```
array([[3.04652486e-004, 9.99695348e-001],
       [1.00000000e+000, 1.21656846e-010],
       [4.07401322e-004, 9.99592599e-001],
       [3.38683166e-004, 9.99661317e-001],
       [1.00000000e+000, 0.00000000e+000],
       [1.26638021e-004, 9.99873362e-001],
       [6.94917415e-004, 9.99305083e-001],
       [1.00000000e+000, 3.05337359e-052],
       [1.18195694e-002, 9.88180431e-001],
       [1.00000000e+000, 1.35957762e-036],
       [5.48694316e-004, 9.99451306e-001],
       [3.09948149e-001, 6.90051851e-001],
       [9.99999999e-001, 5.97074703e-010],
       [2.70407386e-004, 9.99729593e-001],
       [2.19212676e-001, 7.80787324e-001],
       [1.18777582e-004, 9.99881222e-001],
       [1.68184948e-004, 9.99831815e-001],
       [3.03922785e-004, 9.99696077e-001],
       [1.00000000e+000, 1.55085014e-142],
       [1.00000000e+000, 3.61762209e-015],
       [1.00000000e+000, 2.43800626e-053],
       [4.45742295e-004, 9.99554258e-001],
       [1.02759131e-004, 9.99897241e-001],
       [7.19910209e-003, 9.92800898e-001],
       [1.78571255e-004, 9.99821429e-001],
       [9.80686806e-001, 1.93131939e-002],
```

Gambar 3. 15 Hasil Probabilitas

Gambar 3. 15 tersebut menunjukkan *output* berupa *array* probabilitas dari hasil prediksi model klasifikasi biner, di mana setiap baris merepresentasikan probabilitas suatu data termasuk ke dalam masing-masing kelas. Nilai pada posisi pertama menunjukkan probabilitas data termasuk ke kelas 0, sedangkan nilai pada posisi kedua menunjukkan probabilitas data termasuk ke kelas 1. Jumlah dari kedua nilai tersebut selalu mendekati 1 karena mewakili distribusi probabilitas antar kelas. Model akan memilih kelas dengan probabilitas tertinggi sebagai hasil prediksi akhir.

9. Evaluasi Model

Setelah mendapatkan hasil prediksi dan dilatih (model *development*), tahap selanjutnya adalah evaluasi model. Tujuan evaluasi model adalah mengukur seberapa baik model prediksi bekerja dalam memproses data. Evaluasi ini memastikan bahwa model memiliki tingkat akurasi yang baik, mampu membuat prediksi yang tepat, serta tidak mengalami *overfitting* atau *underfitting*. Evaluasi dilakukan dengan menghitung metrik seperti akurasi, presisi, *recall*, dan *f1-score* serta menggunakan *confusion matrix* untuk melihat jumlah prediksi yang benar dan salah. berikut merupakan *matrix*nya :

1. Akurasi

$$Accuracy = \frac{(TP + TN)}{(TP + FN + TN)} \quad (2)$$

1. **TP (True Positive)**: Jumlah prediksi positif yang benar.
 2. **TN (True Negative)**: Jumlah prediksi negatif yang benar.
 3. **FN (False Negative)**: Jumlah prediksi negatif yang salah (seharusnya positif).
2. **Presisi**: Mengukur proporsi prediksi positif yang benar.

$$Precision = \frac{(TP)}{TP + FP} \quad (3)$$

- **FP (False Positive)**: Jumlah prediksi positif yang salah (seharusnya negatif).
3. **Recall**: Mengukur seberapa banyak kasus positif yang berhasil diklasifikasikan dengan benar.

$$Recall = \frac{TP}{(TP + FN)} \quad (4)$$

4. **F1-Score**: Menggabungkan *presisi* dan *recall* untuk memberikan evaluasi seimbang

$$F1 - Score = \frac{2 \times recall \times presisi}{(recall + presisi)} \quad (5)$$

```
# Evaluasi Akurasi
#y_true = label asli pada data uji
y_true = y_test # Assign y_test to y_true
accuracy = accuracy_score(y_true, y_pred)
print(f"Akurasi: {accuracy:.2f}")
```

Gambar 3. 16 Kode Evaluasi Akurasi

```
# Evaluasi Precision, Recall, F1-Score
precision = precision_score(y_true, y_pred, average='weighted')
recall = recall_score(y_true, y_pred, average='weighted')
f1 = f1_score(y_true, y_pred, average='weighted')

print(f"Precision: {precision:.2f}")
print(f"Recall: {recall:.2f}")
print(f"F1-Score: {f1:.2f}")
```

Gambar 3. 17 Kode Evaluasi

Gambar 3. 17 Berdasarkan hasil evaluasi, model menghasilkan *akurasi* sebesar 0.91 dengan *precision* 0.91, *recall* 0.91 dan *f1-score* 0.91, yang menunjukkan bahwa hasil prediksi sesuai dengan label asli yang terdapat pada data uji. Nilai *precision*, *recall*, *f1-score* dengan metode *average* yang mempertimbangkan distribusi kelas dalam data uji untuk memberikan evaluasi yang lebih seimbang terhadap performa model.

```
# Laporan Klasifikasi
print("\nLaporan Klasifikasi:\n")
print(classification_report(y_true, y_pred))
```

Gambar 3. 18 Kode Laporan Klasifikasi

Gambar 3. 18 kode tersebut menampilkan laporan klasifikasi yang berisi metrik evaluasi seperti *presisi*, *recall*, dan *f1-score* dengan menggunakan fungsi `classification_report(y_true, y_pred)` dari pustaka *sklearn*.

Laporan Klasifikasi:

	precision	recall	f1-score	support
0	0.83	0.93	0.88	100
1	0.96	0.89	0.92	175
accuracy			0.91	275
macro avg	0.89	0.91	0.90	275
weighted avg	0.91	0.91	0.91	275

Gambar 3. 19 Hasil Klasifikasi Laporan

Gambar 3. 19 hasil klasifikasi menunjukkan bahwa model memiliki *akurasi* sebesar 91% dengan nilai *precision*, *recall* dan *f1-score* untuk masing – masing kelas.

- Kelas 0 (meningkat) memiliki *precision* 83%, *recall* 93% dan *f1-score* 88% menunjukkan bahwa model lebih sering salah dalam mengidentifikasi kelas ini.
- Kelas 1 (menurun) memiliki *precision* 96%, *recall* 89%, dan *f1-score* 92%, yang berarti model lebih baik dalam mendeteksi kelas ini dibanding kelas 1.
- *Macro avg* dan *weighted avg* masing – masing bernilai sekitar 90%, yang menunjukkan performa rata – rata model terhadap seluruh kelas.

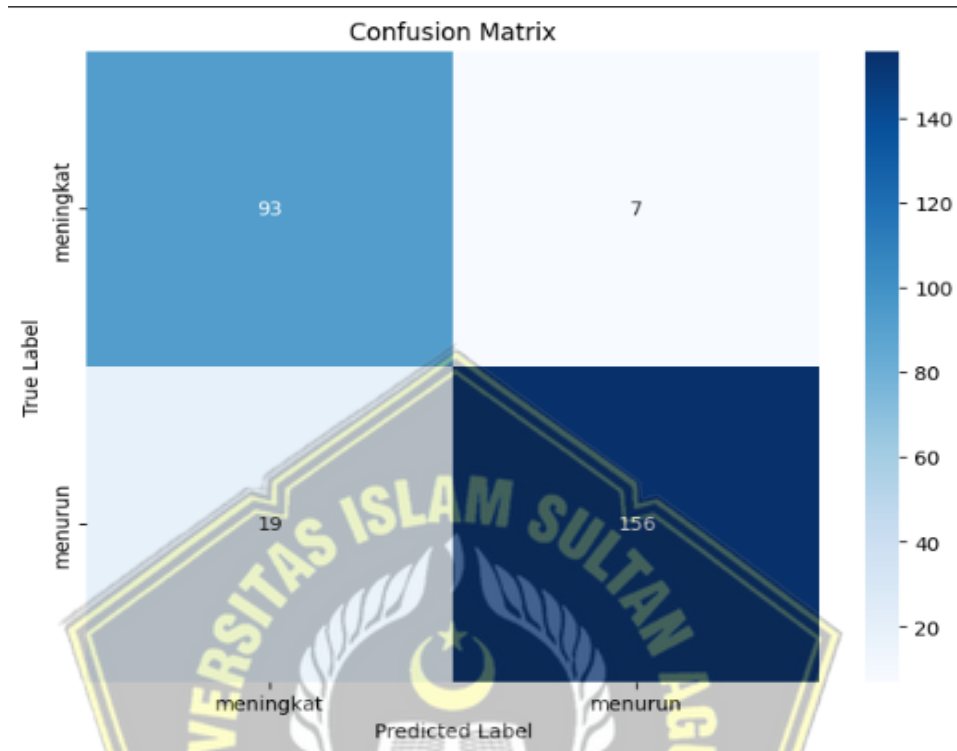
Secara keseluruhan model cenderung lebih ke akurat dalam mengidentifikasi kualitas pemain yang menurun dibandingkan dengan meningkat.

```
# Visualisasi Confusion Matrix
plt.figure(figsize=(8, 6))
sns.heatmap(cm, annot=True, fmt='d', cmap='Blues', xticklabels=['meningkat', 'menurun'], yticklabels=['meningkat', 'menurun'])
plt.xlabel('Predicted Label')
plt.ylabel('True Label')
plt.title('Confusion Matrix')
plt.show()
```

Gambar 3. 20 Kode *Confusion Matrix*

Gambar 3. 20 kode ini digunakan untuk membuat visualisasi *confusion matrix* menggunakan *matplotlib* dan *seaborn* dengan tampilan *heatmap* berwarna biru. Label pada sumbu x menunjukkan hasil prediksi model, sedangkan label pada

sumbu y menunjukkan nilai sebenarnya, sehingga memudahkan dalam mengevaluasi performa klasifikasi.



Gambar 3. 21 *Confusion Matrix*

Gambar 3. 21 menunjukkan *confusion matrix*, yang menggambarkan performa model klasifikasi dalam membedakan kategori meningkat dan menurun.

- 93 pemain yang sebenarnya kualitasnya meningkat diprediksi dengan benar sebagai meningkat (*true positive*)
- 7 pemain yang sebenarnya kualitasnya meningkat salah diprediksi sebagai menurun (*false negative*)
- 19 pemain yang sebenarnya kualitasnya menurun salah diprediksi sebagai meningkat (*false positive*)
- 156 pemain yang sebenarnya kualitasnya menurun diprediksi benar sebagai menurun (*true negative*)

Hasil ini menunjukkan bahwa model lebih sering salah memprediksi pemain kualitas meningkat sebagai kualitas menurun yang dapat berdampak pada keputusan pembelian pemain untuk sebuah klub.

3.2 Perancangan *User Interface*

Dalam pembuatan aplikasi prediksi kualitas pemain sepak bola diperlukan perancangan antarmuka pengguna (*user interface*) yang dirancang dengan baik untuk memudahkan pengguna dalam mengoperasikan aplikasi. Berikut ini adalah tampilan rancangan antarmuka pengguna yang telah disiapkan.

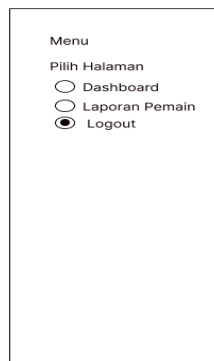
3.2.1 *Login*



Gambar 3. 22 *Login*

Pada gambar 3. 22 merupakan tampilan *login* yang dimana halaman ini hanya bisa dimasukkan oleh *admin* dan *user* karena jika ingin masuk ke halaman ini harus memiliki *username* dan *password*. Untuk masuk ke *login* aplikasi harus memasukkan *username* dan *password* terlebih dahulu lalu klik tombol *login* agar bisa masuk ke dalam aplikasi. Fitur *login* dalam *user interface* berfungsi sebagai mekanisme autentikasi untuk memastikan bahwa hanya pengguna yang berwenang yang dapat mengakses sistem klasifikasi kualitas pemain sepak bola.

3.2.2 Menu



Gambar 3. 23 Menu

Pada gambar 3. 23 menunjukkan *menu* navigasi pada sebuah aplikasi yang memungkinkan pengguna untuk memilih halaman yang ingin diakses. *Menu* ini berisi beberapa opsi berupa tombol yaitu *dashboard*, *upload* data, prediksi kualitas pemain sepak bola dan *logout*. *Menu* dalam *user interface* berfungsi sebagai navigasi utama yang memudahkan pengguna dalam mengakses berbagai *fitur* sistem klasifikasi kualitas pemain sepak bola secara terstruktur dan intuitif.

3.2.3 Dashboard



Gambar 3. 24 Dashboard

Pada gambar 3. 24 halaman *dasboard*, ditampilkan deskripsi mengenai penjelasan aplikasi prediksi penjualan menggunakan algoritma *naïve bayes*. Selanjutnya, halaman ini menyampaikan informasi terkait langkah – langkah prediksi kualitas pemain sepak bola dan pembuatan laporan kualitas pemain sepak

bola. *Dashboard* dalam *user interface* berfungsi sebagai pusat kendali atau tampilan ringkasan dari seluruh proses dan hasil sistem klasifikasi kualitas pemain sepak bola yang dikembangkan. Pada *dashboard*, pengguna dapat melihat visualisasi data seperti grafik performa pemain, hasil prediksi, distribusi kelas kualitas (meningkat atau menurun), serta metrik evaluasi model seperti *akurasi*, *presisi*, *recall*, dan *F1-score* secara *real-time*.

3.2.4 Klasifikasi Kualitas Pemain

Penerapan Klasifikasi Algoritma Naive Bayes

Memisahkan Fitur X dan y (1370 data 9 kolom)

nama_pemain	negara	posisi	yellow_card	red_card	gol	assist	jumlah_pertandingan	kualitas_pemain
410	18	12	2.0	1	0	0	24	0
135	18	12	2.0	0	0	0	12	0
364	29	1	2.0	0	0	0	16	0
1173	41	4	2.0	1	0	1	23	0
160	16	2	0.0	1	0	0	20	0
—	—	—	—	—	—	—	—	—
1128	16	14	0.0	0	0	1	13	0
739	16	14	0.0	0	1	0	4	0
225	16	6	2.0	0	0	0	16	0
627	16	6	1.0	0	1	1	4	0
1174	16	6	1.0	0	3	0	10	0

Gambar 3. 25 Klasifikasi Algoritma *Naive Bayes*

Gambar 3. 25 menunjukkan proses awal penerapan algoritma *Naive Bayes* untuk klasifikasi kualitas pemain sepak bola, yang dimulai dengan pemisahan fitur *X* dan *Y* dari dataset yang berisi 1.370 data dan 9 kolom. Fitur *X* terdiri dari atribut statistik pemain seperti jumlah negara, posisi, yellow_card, red_card, gol, assist, dan jumlah pertandingan, sedangkan fitur *Y* adalah kolom kualitas_pemain yang menunjukkan apakah kualitas pemain meningkat (0) atau menurun (1). Proses ini merupakan tahap penting dalam membangun model klasifikasi, di mana data *input* dan target dipisahkan sebelum dilakukan pelatihan menggunakan algoritma *Naive Bayes*. *Naive Bayes* bekerja dengan cara menghitung kemungkinan setiap data termasuk ke dalam kategori tertentu berdasarkan nilai-nilai fitur yang dimilikinya. Algoritma ini mengasumsikan bahwa setiap fitur saling bebas (*independen*) satu sama lain, sehingga perhitungan probabilitas menjadi lebih sederhana dan cepat. Dalam konteks ini, sistem akan mempelajari hubungan antara statistik permainan dengan kualitas pemain, lalu digunakan

untuk memprediksi performa pemain baru. Metode ini cocok digunakan karena ringan, cepat, dan tetap mampu memberikan hasil prediksi yang cukup akurat untuk analisis awal dalam evaluasi pemain.

3.2.5 Prediksi Kualitas Pemain

1. Input Data Pemain



Gambar 3. 26 Input Data Pemain

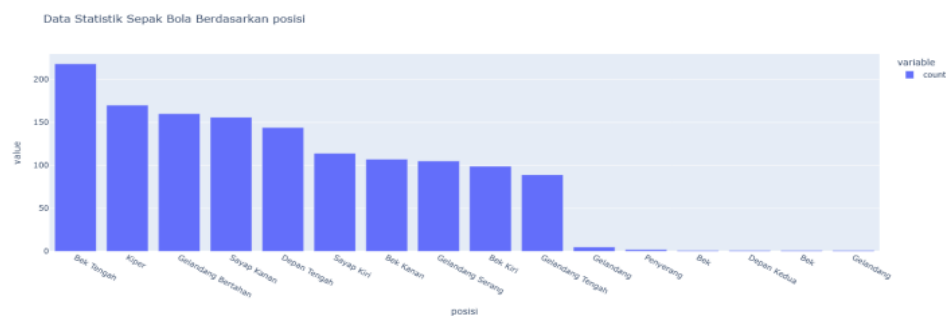
Gambar 3. 26 tahap perencanaan dalam melakukan prediksi kualitas pemain adalah dengan mengunggah file data Pemain Sepak Bola Liga Indonesia dalam format excel. Setelah file berhasil diunggah sistem akan menampilkan data yang telah diunggah untuk memastikan informasi yang dimasukkan sesuai dan siap digunakan dalam proses analisis lebih lanjut.

2. Grafik Pemain Sepak bola

Grafik Data Statistik:

pilih kolom untuk menampilkan grafik:

Posisi



Gambar 3. 27 Grafik Posisi Pemain

Gambar 3. 27 *grafik* posisi pemain dapat ditampilkan dengan memilih parameter berdasarkan bek tengah, kiper, gelandang bertahan, sayap kanan, dan depan tengah. Pilihan ini memungkinkan pengguna untuk memvisualisasikan kualitas pemain secara spesifik sesuai dengan kategori posisi pemain yang diinginkan.

3. Menampilkan atribut dari data

☐ Tampilkan Atribut Data

Atribut Data:

no	date	nama_pemain	negara	club_id	posisi	harga_pemain	yellow_card	red_card	gol	Assist	jumlah_pertandingan	kualitas_pemain
----	------	-------------	--------	---------	--------	--------------	-------------	----------	-----	--------	---------------------	-----------------

Gambar 3. 28 Atribut Data

Gambar 3. 28 tampilkan atribut data yang terdiri dari : negara, club_id, posisi, harga_pemain, yellow_card, red_card, gol, assist, jumlah pertandingan. Atribut-atribut ini dipilih karena berisi informasi statistik yang menggambarkan performa pemain di lapangan. Dengan menggunakan atribut ini, algoritma *Naive Bayes* dapat menghitung probabilitas dan melakukan pada prediksi masing-masing kelas pada label kualitas_pemain berdasarkan pola data historis dari fitur-fitur tersebut

4. Menghapus kolom tidak diperlukan

hapus kolom tidak diperlukan :

nama_pemain	negara	posisi	yellow_card	red_card	gol	Assist	jumlah_pertandingan	kualitas_pemain
Enkrishtul	Indonesia	Kiper	2.0	1	0	0	24	Meningkat
Arshika Ramadani	Indonesia	Kiper	2.0	0	0	0	12	Meningkat
Dani Dini	Makedonia Utara	Bek Tengah	2.0	0	0	0	16	Meningkat
Slavko Džurjanić	Serbia	Bek Tengah	2.0	1	0	1	29	Meningkat
Arif Idrus	Indonesia	Bek Kanan	6.0	1	0	0	29	Meningkat
---	---	---	---	---	---	---	---	---
Sady	Indonesia	Sayap Kanan	0.0	0	0	1	13	Meningkat
Kamarudin	Indonesia	Sayap Kanan	0.0	0	1	0	4	Meningkat
Basojan Latocovina	Indonesia	Depan Tengah	2.0	0	0	0	16	Meningkat
Irsan Widiya	Indonesia	Depan Tengah	1.0	0	1	1	4	Meningkat
Silko Gacbar	Indonesia	Depan Tengah	1.0	0	3	0	10	Meningkat

Gambar 3. 29 Menghapus Kolom

Gambar 3. 29 lakukan penghapusan kolom yang tidak diperlukan seperti nomor, *date*, *nama_pemain*, *club_id*, dan *harga_pemain* untuk menyederhanakan data dan memfokuskan analisis pada atribut yang relevan.

5. Melakukan Label Encoding

LabelEncoder

Setelah melakukan labelencode data :

nama_pemain	negara	posisi	yellow_card	red_card	gol	assist	jumlah_pertandingan	kualitas_pemain
410	18	12	2.0	1	0	0	24	0
135	18	12	2.0	0	0	0	12	0
364	29	4	2.0	0	0	0	16	0
1179	47	4	2.0	1	0	1	29	0
168	18	2	6.0	1	0	0	29	0
...	...	...	...	...	...	...	...	...
1128	18	14	0.0	0	0	1	13	0
736	18	14	0.0	0	1	0	4	0
225	18	6	2.0	0	0	0	16	0
627	18	6	1.0	0	1	1	4	0
1174	18	6	1.0	0	3	0	10	0

Gambar 3. 30 Label Encoder

Gambar 3. 30 menunjukkan hasil penerapan *LabelEncoder* pada dataset pemain sepak bola, di mana data kategorikal seperti negara, posisi, dan kualitas_pemain telah diubah menjadi format numerik agar dapat diproses oleh algoritma *machine learning*. *LabelEncoder* mengonversi setiap nilai unik dalam kolom kategorikal menjadi angka, contohnya negara, posisi dan kualitas_pemain yang awalnya berupa teks kini diwakili oleh angka. Kolom lainnya seperti yellow_card, red_card, gol, assist, jumlah_pertandingan, dan merupakan data numerik asli yang digunakan untuk analisis lebih lanjut. Proses ini penting untuk memastikan semua fitur dalam dataset berada dalam format yang sesuai untuk pelatihan model prediksi atau klasifikasi.

6. Tahapan Algoritma Naïve Bayes

Tahapan algoritma *naïve bayes* meliputi penghitungan probabilitas setiap fitur terhadap kelas menggunakan teorema bayes, mengasumsikan independen antar fitur kemudian menentukan kelas dengan probabilitas tertinggi berdasarkan data latih.

1. Mempersiapkan data dari variabel x dan y

Pada tahap awal klasifikasi menggunakan algoritma *naïve bayes*, data fitur dimuat ke dalam variabel x, yang berisi atribut – atribut seperti negara, posisi, yellow card, red card, gol, assist, dan jumlah_pertandingan sedangkan data label dimasukkan ke dalam variabel y yang mempresentasikan kategori meningkat dan menurun.

2. Pembagian data *training* dan *testing*

Data yang telah dimuat kemudian dibagi menjadi dua bagian, yaitu data *training* untuk melatih model dan data *testing* untuk mengevaluasi performa model *naïve bayes* dalam memprediksi kategori.

- Data *training* variabel *x* dan *y*

Data *training* yang tersimpan dalam variabel *x_train* dan *y_train* merupakan subset dari dataset asli yang digunakan untuk melatih model. Variabel *x_train* berisi atribut – atribut atau fitur, sedangkan variabel *y_train* berisi label atau kategori target yang sesuai dengan data tersebut.

Pembagian Data Training dan Testing

Jumlah data untuk training: 1096 (80%)

Jumlah data untuk testing: 274 (20%)

Data Training Variabel x (1096 data, 8 kolom):

nama_pemain	negara	posisi	yellow_card	red_card	gol	Assist	jumlah_pertandingan
741	18	15	2.0	0	2	0	9
890	18	2	4.0	0	0	1	13
134	8	4	4.0	0	1	0	16
306	18	15	0.0	0	2	0	8
927	6	9	7.0	0	3	1	27

Data Training Variabel y : 1096

kualitas_pemain
0
1
0
0
0

Gambar 3. 31 Data *Training* Variabel x

Gambar 3. 32 Data *Training* Variabel y

- Data *testing*

Setelah proses pelatihan, model diuji menggunakan data *testing* yang tersimpan dalam variabel *x_test* dan *y_test*. Variabel *x_test* digunakan untuk melatih model sedangkan *y_test* menyimpan digunakan untuk

menguji dalam memprediksi kategori berdasarkan target yang sesuai dengan data.

Data Testing (274 data, 6 kolom):

nama_pemain	negara	posisi	yellow_card	red_card	gol	Assist	jumlah_pertandingan
698	18	4	0.0	0	0	0	9
1240	18	10	6.0	0	0	2	17
574	18	15	1.0	0	0	0	8
572	18	15	0.0	0	0	0	8
954	18	6	2.0	0	8	1	22

Gambar 3. 33 Data *Testing*

3. Hasil prediksi pemain sepak bola

Setelah melakukan prediksi selanjutnya menampilkan probabilitas yang digunakan untuk membandingkan probabilitas “meningkat” dan “menurun”. Dalam hal ini, hasil prediksi ditentukan dengan probabilitas tertinggi.

Hasil Prediksi Pemain Sepakbola Berdasarkan Data Statistik

Data Penjualan (274 data, 8 kolom):

```
array([1, 0, 1, 1, 0, 1, 1, 0, 1, 1, 0, 1, 1, 0, 1, 1, 1, 0, 0, 0, 1,
1, 1, 1, 0, 1, 0, 0, 0, 1, 1, 1, 1, 1, 1, 0, 0, 1, 0,
0, 1, 1, 1, 1, 0, 1, 0, 1, 1, 0, 1, 1, 0, 1, 1, 0, 0,
1, 1, 1, 0, 1, 0, 0, 1, 0, 1, 0, 0, 1, 1, 0, 0, 1, 1,
1, 0, 0, 0, 1, 0, 0, 0, 1, 0, 1, 1, 0, 1, 0, 0, 1, 1, 0,
1, 1, 1, 1, 1, 1, 0, 1, 1, 0, 1, 1, 1, 1, 0, 1, 1, 0,
1, 1, 1, 0, 0, 1, 1, 0, 1, 0, 1, 1, 0, 0, 1, 1, 0, 1,
0, 1, 0, 0, 1, 1, 0, 1, 1, 0, 0, 1, 1, 1, 0, 0, 1, 0,
1, 1, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0,
1, 1, 1, 1, 0, 0, 0, 1, 1, 1, 1, 0, 0, 0, 0, 1, 0, 0,
0, 1, 1, 0, 0, 1, 0, 0, 1, 0, 1, 1, 1, 0, 0, 1, 1, 0,
2, 1, 1, 0, 1, 0, 1, 1, 1, 0, 1, 1, 1, 0, 0, 1, 1, 1, 0,
2, 1, 1, 0, 1, 0, 1, 1, 1, 0, 1, 1, 1, 1, 1, 1, 1, 1,
1, 0, 1, 1, 0, 0, 1, 0, 1, 1])
```

Gambar 3. 34 Hasil Prediksi Pemain Sepak Bola

4. Probabilitas Prediksi

Setelah melakukan prediksi selanjutnya menampilkan probabilitas yang digunakan untuk membandingkan probabilitas “meningkat” dan “menurun”. Dalam hal ini, hasil prediksi ditentukan dengan probabilitas tertinggi.

Probabilitas Prediksi
Data Testing (274 data, 2 kolom):

```
array([[3.2124807e-004, 9.9677675e-001],
       [5.99999999e-001, 1.45668071e-000],
       [4.22787195e-004, 9.99577211e-001],
       [3.51344391e-004, 9.99638556e-001],
       [1.00000000e+000, 0.00000000e+000],
       [1.33756132e-004, 9.99866144e-001],
       [2.23626181e-004, 9.99272124e-001],
       [1.00000000e+000, 1.02701099e-047],
       [5.89823807e-003, 9.98309790e-001],
       [1.00000000e+000, 2.16640914e-015],
       [1.73407199e-004, 9.99839594e-001],
       [1.57888284e-001, 0.02191716e-001],
       [0.99999991e-001, 7.47840660e-000],
       [2.85158825e-004, 9.99714810e-001],
       [1.33243341e-001, 0.66750059e-001],
       [1.25086491e-004, 9.99820991e-001],
       [1.70512886e-004, 9.99623467e-001],
       [5.20252000e-004, 9.99079748e-001],
       [1.00000000e+000, 1.69964914e-142],
       [1.00000000e+000, 5.56724688e-014],
       [1.00000000e+000, 4.67175531e-053],
       [4.40175450e-004, 9.99534054e-001],
       [1.00421131e-004, 9.99999960e-001],
       [6.30806414e-003, 9.93791934e-001],
       [4.45545111e-004, 0.00000000e+000]]
```

Gambar 3. 35 Probabilitas Prediksi

5. Evaluasi Model

Evaluasi model dilakukan setelah proses prediksi dengan membandingkan hasil prediksi dengan data aktual menggunakan metrik seperti akurasi, *presisi*, *recall*, dan *f1-score* untuk mengukur seberapa baik model mampu memprediksi kategori target secara tepat.



Gambar 3. 36 Evaluasi Model

6. Prediksi Pemain Berdasarkan Input manual

Tampilan antarmuka form input manual pada aplikasi prediksi kualitas pemain sepak bola, yang dirancang untuk memudahkan pengguna memasukkan data pemain secara langsung. Form tersebut untuk memproses informasi pemain sepak bola untuk menggunakan model

machine learning yang akan menghasilkan prediksi kategori kualitas pemain, seperti meningkat atau menurun.

Prediksi Pemain Baru Berdasarkan Input Manual

Nama Pemain

Pilih Negara
 Indonesia ▾

Pilih Posisi
 Kiper ▾

Jumlah Kartu Kuning
 0

Jumlah Kartu Merah
 0

Jumlah Gol
 0

Jumlah Pertandingan
 0

Prediksi Kualitas Pemain

Gambar 3. 37 Prediksi Pemain Berdasarkan *Input* manual

3.2.6 Logout



Gambar 3. 38 Logout

Gambar 3.37 menunjukkan fitur *logout* dalam *user interface* berfungsi untuk mengakhiri sesi pengguna yang sedang aktif, sehingga mencegah akses tidak sah atau penyalahgunaan akun ketika perangkat ditinggalkan atau digunakan bersama. Setelah pengguna menekan tombol *logout*, sistem akan menghapus status autentikasi yang tersimpan dalam sesi, dan secara otomatis mengarahkan pengguna kembali ke halaman *login*. Dengan adanya fitur ini, keamanan data dan privasi pengguna tetap terjaga, terutama dalam sistem yang memuat informasi penting seperti hasil klasifikasi kualitas pemain. *Logout* juga menjadi bagian penting dari manajemen akses yang profesional dalam aplikasi berbasis *web*.

BAB IV

HASIL DAN ANALISIS PENELITIAN

4.1 Analisis Penelitian

4.1.1 Mempersiapkan data

Langkah pertama sebelum melakukan tahap klasifikasi *naïve bayes* adalah mempersiapkan data untuk dianalisis. Pada tahap ini, data yang digunakan berupa nilai *numerik* atau kode setelah melalui proses pembersihan data. Berikut ini adalah tabel yang digunakan.

Tabel 4. 1 Memuat Data

Negara	Posisi	Yellow Card	Red card	gol	asisst	Jumlah Pertandingan
18	4	0	0	0	0	9
18	10	6	0	0	2	17
18	15	1	0	0	0	8
18	15	0	0	0	0	8
18	6	2	0	8	1	22

Tabel 4.1 berisi data yang digunakan sebagai dasar dalam proses analisis klasifikasi menggunakan metode *Naive Bayes*, yang bertujuan untuk mengidentifikasi pola dan menentukan kategori berdasarkan probabilitas masing-masing fitur.

4.1.2 Pembagian data training dan data testing

Data dibagi menjadi dua bagian, yaitu data pelatihan (training) dan data pengujian (*testing*). Sebanyak 80% dari data yang digunakan sebagai data latih, sementara 20% digunakan sebagai data pengujian. Pembagian data tersebut dilakukan sebagai berikut :

- *Training* data (80%) = $0.8 * 1343 = 1.098$
- *Testing* data (20%) = $0.2 * 1343 = 274$

Pembagian ini dilakukan secara seimbang untuk setiap kategori, yaitu “meningkat” dan “menurun”. pemilihan data pelatihan dilakukan secara acak dari total 1.343 data, sehingga diperoleh 1.098 data untuk dilatih. Sementara itu, data pengujian dipilih dari kategori jumlah terbanyak dalam 274 data, dimana yang paling sering muncul digunakan sebagai pengujian.

a. Membaca *Training data*

Tabel 4. 2 *Training Data*

Negara	Posisi	yellow card	Red Card	Gol	asisst	Jumlah Pertandingan
18	4	0	0	0	0	9
18	10	6	0	0	2	17
18	15	1	0	0	0	8
18	15	0	0	0	0	8
18	6	2	0	8	1	22

Tabel 4. 2 menampilkan data pelatihan yang terdiri dari 1.098 sampel yang dipilih secara acak dengan tujuan melatih data training.

b. Testing data

Tabel 4. 3 *Testing Data*

Negara	Posisi	yellow card	Red Card	gol	asisst	Jumlah pertandingan	kualitas pemain
18	4	0	0	0	0	9	?
18	10	6	0	0	2	17	?
18	15	1	0	0	0	8	?
18	15	0	0	0	0	8	?
18	6	2	0	8	1	22	?

Tabel 4.3 menampilkan data uji yang terdiri dari 274 sampel yang dipilih secara acak untuk menentukan hasil prediksi apakah suatu kualitas pemain meningkat atau menurun.

1. Menghitung Probabilitas Kelas

Tabel 4. 4 Menghitung Probabilitas Kelas

kualitas_pemain	jumlah kemunculan	probabilitas prior
Meningkat	456	$456/1.098 = 0,415300546$
Menurun	642	$642/1.098 = 0,584699454$

Tabel 4.4 menghitung probabilitas kelas berdasarkan jumlah kemunculan masing-masing kategori dalam data pelatihan, di mana kategori “Meningkat” muncul sebanyak 456 kali dengan probabilitas 0,4153, sedangkan kategori “Menurun” muncul sebanyak 642 kali dengan probabilitas 0,5846 dari total 1.098 data pelatihan.

2. Menghitung Probabilitas atribut

a. Probabilitas kategori negara

Tabel 4. 5 Probabilitas Negara

P(negara)	Kualitas pemain	
	Kode 0	Kode 1
Kode 18	2,063596	1,465732
Kode 18	2,063596	1,465732
Kode 18	2,063596	1,465732
Kode 18	2,063596	1,465732
Kode 18	2,063596	1,465732

b. Probabilitas kategori posisi

Tabel 4. 6 Probabilitas Posisi

P(posisi)	Kualitas pemain	
	Kode 0	Kode 1
Kode 4	0,361842	0,257009
Kode 10	0,1864	0,1324
Kode 15	0.201754386	0.143302181
Kode 15	0.201754386	0.143302181
Kode 6	0.25	0.177570093

c. Probabilitas kategori *yellow card*

Tabel 4. 7 Probabilitas Yellow Card

P(yellow_card)	Kualitas pemain	
	Kode 0	Kode 1
0 card	0,947368	0,672897
6 card	0,0943	1,61059
1 card	0.432017	0.306853
0 card	0.947368	0.672897
2 card	0.375	0.266355

d. Probabilitas kategori *red card*

Tabel 4. 8 Probabilitas Red Card

P(red_card)	Kualitas pemain	
	Kode 0	Kode 1
0 card	2,267544	1,610592
0 card	2,267544	1,610592
0 card	2,267544	1,610592
0 card	2,267544	1,610592
0 card	2,267544	1,610592

e. Probabilitas kategori gol

Tabel 4. 9 Probabilitas Gol

P(gol)	Kualitas pemain	
	Kode 0	Kode 1
0 gol	1,638158	1,163551
0 gol	1,638158	1,163551
0 gol	1,638158	1,163551
0 gol	1,638158	1,163551
8 gol	0.013157	0.009345

f. Probabilitas kategori asisst

Tabel 4. 10 Probabilitas Asisst

P(asisst)	Kualitas pemain	
	Kode 0	Kode 1
0 asisst	1,605263	1,140187
2 asisst	0,1886	0,13396
0 asisst	1,605263	1,140187
0 asisst	1,605263	1,140187
1 asisst	0.405701	0.288161

g. Probabilitas kategori jumlah pertandingan

Tabel 4. 11 Probabilitas Jumlah Pertandingan

P(jumlah_pertandingan)	Kualitas pemain	
	Kode 0	Kode 1
9 pertandingan	0,078947	0,056075
17 pertandingan	0,08991	0,06386
8 pertandingan	0.103070	0.073208
8 pertandingan	0.103070	0.073208
22 pertandingan	0.052631	0.037383

Berikut adalah perhitungan pada setiap kategori kualitas pemain :

1. Menghitung probabilitas nama_pemain pada kode 701

a) Menghitung probabilitas prior P(H)

$$P(\text{Meningkat}) = 456/1.098 = 0,415300546$$

$$P(\text{Menurun}) = 642/1.098 = 0,584699454$$

b) Menghitung probabilitas likelihood

$$P(\text{negara "kode 18" | kualitas_pemain kode "0"}) = 941/456 = 2,063596$$

$$P(\text{negara "kode 18" | kualitas_pemain kode "1"}) = 941/642 = 1,465732$$

$$P(\text{posisi "kode 4" | kualitas_pemain kode "0"}) = 165 / 456 = 0,361842$$

$$P(\text{posisi "kode 4" | kualitas_pemain kode "1"}) = 165 / 642 = 0,257009$$

$$P(\text{yellow_card "0" | kualitas_pemain kode "0"}) = 432/456 = 0,947368$$

$$P(\text{yellow_card "0" | kualitas_pemain kode "1"}) = 432/642 = 0,672897$$

$$P(\text{red_card "0" | kualitas_pemain kode "0"}) = 1034/456 = 2,267544$$

$$P(\text{red_card "0" | kualitas_pemain kode "1"}) = 1034/642 = 1,610592$$

$$P(\text{gol "0" | kualitas_pemain kode "0"}) = 747/456 = 1,638158$$

$$P(\text{gol "0" | kualitas_pemain kode "1"}) = 747/642 = 1,163551$$

$$P(\text{asisst "0" | kualitas_pemain kode "0"}) = 732/456 = 1,605263$$

$$P(\text{asisst "0" | kualitas_pemain kode "1"}) = 732/642 = 1,140187$$

$$P(\text{jumlah pertandingan "9" | kualitas_pemain kode "0"}) = 36/456 = 0,078947$$

$$P(\text{jumlah pertandingan "9" | kualitas_pemain kode "1"}) = 36/642 = 0,056075$$

c) Menghitung probabilitas posterior

Probabilitas meningkat

$$P(\text{negara}) * P(\text{Posisi}) * P(\text{yellow_card}) * P(\text{red_card}) * P(\text{gol}) * P(\text{assist}) * P(\text{jumlah_pertandingan})$$

$$2,063596 * 0,361842 * 0,947368 * 2,267544 * 1,638158 * 1,605263 * 0,078947 = 0,333010662$$

Probabilitas menurun

$$P(\text{negara}) * P(\text{Posisi}) * P(\text{yellow_card}) * P(\text{red_card}) * P(\text{gol}) * P(\text{assist}) * P(\text{jumlah_pertandingan})$$

$$1,465732 * 0,257009 * 0,672897 * 1,610592 * 1,638158 * 1,605263 * 0,078947 = 0,030371542$$

d) Menentukan probabilitas tertinggi

Probabilitas meningkat

$$P(X|\text{kualitas_pemain "kode 0"}) * P(X|\text{kualitas_pemain "kode 0"}) = 0,333010662 * 0,415300546 = 0,1383$$

Probabilitas menurun

$$P(X|\text{kualitas_pemain "kode 1"}) * P(X|\text{kualitas_pemain "kode 1"}) = 0,030371542 * 0,584699454 = 0,017758$$

Dengan demikian nama_pemain dengan kode 701 kode 0 lebih besar dari pada 1 (meningkat)

2. Menghitung probabilitas nama_pemain pada kode 1253

a) Menghitung probabilitas prior P(H)

$$P(\text{Meningkat}) = 456/1.098 = 0,415300546$$

$$P(\text{Menurun}) = 642/1.098 = 0,584699454$$

b) Menghitung probabilitas likelihood

$$P(\text{negara "kode 18" | kualitas_pemain kode "0"}) = 2,063596491$$

$$P(\text{negara "kode 18" | kualitas_pemain kode "1"}) = 1,465732087$$

$$\begin{aligned}
P(\text{posisi "kode 10" | kualitas_pemain kode "0"}) &= 0.186403509 \\
P(\text{posisi "kode 10" | kualitas_pemain kode "1"}) &= 0.132398754 \\
P(\text{yellow_card "6" | kualitas_pemain kode "0"}) &= 0.094298246 \\
P(\text{yellow_card "6" | kualitas_pemain kode "1"}) &= 0.066978193 \\
P(\text{red_card "0" | kualitas_pemain kode "0"}) &= 2.26754386 \\
P(\text{red_card "0" | kualitas_pemain kode "1"}) &= 1.6105919 \\
P(\text{gol "0" | kualitas_pemain kode "0"}) &= 1.638157895 \\
P(\text{gol "0" | kualitas_pemain kode "1"}) &= 1.163551402 \\
P(\text{assisst "2" | kualitas_pemain kode "0"}) &= 0.188596491 \\
P(\text{assisst "2" | kualitas_pemain kode "1"}) &= 0.133956386 \\
P(\text{jumlah pertandingan "17" | kualitas_pemain kode "0"}) &= 0.089912281 \\
P(\text{jumlah pertandingan "17" | kualitas_pemain kode "1"}) &= 0.063862928
\end{aligned}$$

c) Menghitung probabilitas posterior $P(H|X)$

Probabilitas meningkat

$$\begin{aligned}
&P(\text{negara}) * P(\text{Posisi}) * P(\text{yellow_card}) * P(\text{red_card}) * P(\text{gol}) * P(\text{assist}) * \\
&P(\text{jumlah_pertandingan}) \\
&2,063596 * 0,361842 * 0,947368 * 2,267544 * 1,638158 * 1,605263 * \\
&0,078947 = 0.002284791
\end{aligned}$$

Probabilitas Menurun

$$\begin{aligned}
&P(\text{negara}) * P(\text{Posisi}) * P(\text{yellow_card}) * P(\text{red_card}) * P(\text{gol}) * P(\text{assist}) * \\
&P(\text{jumlah_pertandingan}) \\
&1,465732 * 0,1324 * 1,61059 * 1,610592 * 1,163551 * 0,13396 * 0,06386 = \\
&0.00020838
\end{aligned}$$

d) Menentukan probabilitas tertinggi (keputusan akhir)

$$\begin{aligned}
&P(X|\text{kualitas_pemain "kode 0"}) * P(X|\text{kualitas_pemain "kode 0"}) = 0.000948875 \\
&P(X|\text{kualitas_pemain "kode 1"}) * P(X|\text{kualitas_pemain "kode 1"}) = 0.133779721 \\
&\text{jadi nama_pemain dengan kode "1253" kode 1 lebih besar dari pada 0} \\
&(\text{menurun})
\end{aligned}$$

3. Menghitung probabilitas nama_pemain pada kode 576

a) Menghitung probabilitas prior $P(H)$

$$P(\text{Meningkat}) = 456/1.098 = 0,415300546$$

$$P(\text{Menurun}) = 642/1.098 = 0,584699454$$

b) Menghitung probabilitas likelihood

$$P(\text{negara "kode 18" | kualitas_pemain kode "0"}) = 2.063596491$$

$$P(\text{negara "kode 18" | kualitas_pemain kode "1"}) = 1.465732087$$

$$P(\text{posisi "kode 15" | kualitas_pemain kode "0"}) = 0.201754386$$

$$P(\text{posisi "kode 15" | kualitas_pemain kode "1"}) = 0.143302181$$

$$P(\text{yellow_card "1" | kualitas_pemain kode "0"}) = 0.432017544$$

$$P(\text{yellow_card "1" | kualitas_pemain kode "1"}) = 0.306853583$$

$$P(\text{red_card "0" | kualitas_pemain kode "0"}) = 2.26754386$$

$$P(\text{red_card "0" | kualitas_pemain kode "1"}) = 1.6105919$$

$$P(\text{gol "0" | kualitas_pemain kode "0"}) = 1.638157895$$

$$P(\text{gol "0" | kualitas_pemain kode "1"}) = 1.163551402$$

$$P(\text{assisst "0" | kualitas_pemain kode "0"}) = 1.605263158$$

$$P(\text{assisst "0" | kualitas_pemain kode "1"}) = 1.140186916$$

$$P(\text{jumlah pertandingan "8" | kualitas_pemain kode "0"}) = 0.103070175$$

$$P(\text{jumlah pertandingan "8" | kualitas_pemain kode "1"}) = 0.073208723$$

c) Menghitung probabilitas posterior $P(H|X)$

Probabilitas meningkat

$$P(\text{negara}) * P(\text{Posisi}) * P(\text{yellow_card}) * P(\text{red_card}) * P(\text{gol}) * P(\text{assist}) * P(\text{jumlah_pertandingan}) \\ 2,063596 * 0.201754386 * 0.432017 * 2,267544 * 1,638158 * 1,605263 * 0.103070 = 0.410630603$$

Probabilitas menurun

$$P(\text{negara}) * P(\text{Posisi}) * P(\text{yellow_card}) * P(\text{red_card}) * P(\text{gol}) * P(\text{assist}) * P(\text{jumlah_pertandingan}) \\ 1,465732 * 0.143302181 * 0.306853 * 1,610592 * 1,163551 * 1,140187 * 0.073208 = 0.010082043$$

d) Menentukan probabilitas tertinggi (keputusan akhir)

$$P(X|\text{kualitas_pemain "kode 0"}) * P(X|\text{kualitas_pemain "kode 0"}) = 0.170535114$$

$$P(X|\text{kualitas_pemain "kode 1"}) * P(X|\text{kualitas_pemain "kode 1"}) = 0.005894965$$

jadi nama_pemain dengan kode "576" kode 0 lebih besar dari pada 1(meningkat)

4. Menghitung probabilitas nama_pemain pada kode 574

a) Menghitung probabilitas prior $P(H)$

$$P(\text{Meningkat}) = 456/1.098 = 0,415300546$$

$$P(\text{Menurun}) = 642/1.098 = 0,584699454$$

b) Menghitung probabilitas likelihood

$$P(\text{negara "kode 18" | kualitas_pemain kode "0"}) = 2.063596491$$

$$P(\text{negara "kode 18" | kualitas_pemain kode "1"}) = 1.465732087$$

$$P(\text{posisi "kode 15" | kualitas_pemain kode "0"}) = 0.201754386$$

$$P(\text{posisi "kode 15" | kualitas_pemain kode "1"}) = 0.143302181$$

$$P(\text{yellow_card "0" | kualitas_pemain kode "0"}) = 0.947368421$$

$$P(\text{yellow_card "0" | kualitas_pemain kode "1"}) = 0.672897196$$

$$P(\text{red_card "0" | kualitas_pemain kode "0"}) = 2.26754386$$

$$P(\text{red_card "0" | kualitas_pemain kode "1"}) = 1.6105919$$

$$P(\text{gol "0" | kualitas_pemain kode "0"}) = 1.638157895$$

$$P(\text{gol "0" | kualitas_pemain kode "1"}) = 1.163551402$$

$$P(\text{assisst "0" | kualitas_pemain kode "0"}) = 1.605263158$$

$$P(\text{assisst "0" | kualitas_pemain kode "1"}) = 1.140186916$$

$$P(\text{jumlah pertandingan "8" | kualitas_pemain kode "0"}) = 0.103070175$$

$$P(\text{jumlah pertandingan "8" | kualitas_pemain kode "1"}) = 0.073208723$$

c) Menghitung probabilitas posterior $P(H|X)$

Probabilitas meningkat

$$P(\text{negara}) * P(\text{Posisi}) * P(\text{yellow_card}) * P(\text{red_card}) * P(\text{gol}) * P(\text{assist}) * P(\text{jumlah_pertandingan})$$

$$P(\text{negara}) * P(\text{Posisi}) * P(\text{yellow_card}) * P(\text{red_card}) * P(\text{gol}) * P(\text{assist}) * P(\text{jumlah_pertandingan})$$

$$2,063596 * 0.201754386 * 0.947368 * 2,267544 * 1,638158 * 1,605263 * 0.103070 = 0.242413822$$

$$0.103070 = 0.242413822$$

Probabilitas menurun

$$P(\text{negara}) * P(\text{Posisi}) * P(\text{yellow_card}) * P(\text{red_card}) * P(\text{gol}) * P(\text{assist}) * P(\text{jumlah_pertandingan})$$

$$P(\text{negara}) * P(\text{Posisi}) * P(\text{yellow_card}) * P(\text{red_card}) * P(\text{gol}) * P(\text{assist}) * P(\text{jumlah_pertandingan})$$

$$1,465732 * 0.143302181 * 0.672897 * 1,610592 * 1,163551 * 1,140187 * 0.073208 = 0.022108846$$

$$0.073208 = 0.022108846$$

d) Menentukan probabilitas tertinggi (keputusan akhir)

$$P(X|\text{kualitas_pemain "kode 0"}) * P(X|\text{kualitas_pemain "kode 0"}) = 0.100674593$$

$$P(X|\text{kualitas_pemain "kode 1"}) * P(X|\text{kualitas_pemain "kode 1"}) = 0.01292703$$

jadi nama_pemain dengan kode "574" kode 0 lebih besar dari pada 1 (meningkat)

5. Menghitung probabilitas nama_pemain pada kode 967

a) Menghitung probabilitas prior $P(H)$

$$P(\text{Meningkat}) = 456/1.098 = 0,415300546$$

$$P(\text{Menurun}) = 642/1.098 = 0,584699454$$

b) Menghitung probabilitas likelihood

$$P(\text{negara "kode 18" | kualitas_pemain kode "0"}) = 2.063596491$$

$$P(\text{negara "kode 18" | kualitas_pemain kode "1"}) = 1.465732087$$

$$P(\text{posisi "kode 6" | kualitas_pemain kode "0"}) = 0.25$$

$$P(\text{posisi "kode 6" | kualitas_pemain kode "1"}) = 0.177570093$$

$$P(\text{yellow_card "2" | kualitas_pemain kode "0"}) = 0.375$$

$$P(\text{yellow_card "2" | kualitas_pemain kode "1"}) = 0.26635514$$

$$P(\text{red_card "0" | kualitas_pemain kode "0"}) = 2.26754386$$

$$P(\text{red_card "0" | kualitas_pemain kode "1"}) = 1.163551402$$

$$P(\text{gol "8" | kualitas_pemain kode "0"}) = 0.013157895$$

$$P(\text{gol "8" | kualitas_pemain kode "1"}) = 0.009345794$$

$$P(\text{assisst "1" | kualitas_pemain kode "0"}) = 0.405701754$$

$$P(\text{assisst "1" | kualitas_pemain kode "1"}) = 0.288161994$$

$$P(\text{jumlah_pertandingan "22" | kualitas_pemain kode "0"}) = 0.052631579$$

$$P(\text{jumlah_pertandingan "22" | kualitas_pemain kode "1"}) = 0.037383178$$

c) Menghitung probabilitas posterior $P(H|X)$

Probabilitas meningkat

$$\begin{aligned} &P(\text{negara}) * P(\text{Posisi}) * P(\text{yellow_card}) * P(\text{red_card}) * P(\text{gol}) * P(\text{assist}) * \\ &P(\text{jumlah_pertandingan}) \\ &2,063596 * 0,25 * 0.375 * 2,267544 * 0.013157 * 0.405701 * 0.052631 = \\ &0.000123251 \end{aligned}$$

Probabilitas menurun

$$\begin{aligned} &P(\text{negara}) * P(\text{Posisi}) * P(\text{yellow_card}) * P(\text{red_card}) * P(\text{gol}) * P(\text{assist}) * \\ &P(\text{jumlah_pertandingan}) \\ &1,465732 * 0.177570093 * 0.266355 * 1,610592 * 0.009345 * 0.288161 * 0.037383 \\ &= 8.12083E-06 \end{aligned}$$

d) Menentukan probabilitas tertinggi (keputusan akhir)

$$\begin{aligned} &P(X|\text{kualitas_pemain "kode 0"}) * P(X|\text{kualitas_pemain "kode 0"}) = \\ &5.11863E-05 \end{aligned}$$

$$P(X|\text{kualitas_pemain "kode 1"}) * P(X|\text{kualitas_pemain "kode 1"}) = 4.74824\text{E-}06$$

jadi nama_pemain dengan kode "976" kode 0 lebih besar dari pada 1 (meningkat)

4.1.3 Evaluasi Model

Tabel 4. 12 Evaluasi Model

<i>True label\ Predicted label</i>	Meningkat	Menurun
Meningkat (TP+FN)	93	7
Menurun (FP+TN)	19	156

Tabel 4.12 menunjukkan evaluasi model berdasarkan perbandingan antara label asli (true label) dan label yang diprediksi (predicted label), di mana terdapat 93 data yang benar terklasifikasi sebagai “Meningkat”, 7 data “Meningkat” yang salah diprediksi sebagai “Menurun”, 19 data “Menurun” yang salah diprediksi sebagai “Meningkat,” dan 156 data yang benar terklasifikasi sebagai “Menurun”.

Dimana :

- TP (True Positive) = 93 (prediksi benar sebagai Meningkat)
- FN (False Negative) = 7 (seharunya Meningkat, tetapi diprediksi Menurun)
- FP (False Positive) = 19 (seharunya Menurun, tetapi diprediksi sebagai Meningkat)
- TN (True Negative) = 156 (prediksi benar sebagai Menurun)

Berikut perhitungan dari akurasi, presisi, recall dan f1 – score :

1. Perhitungan akurasi

Perhitungan akurasi dilakukan dengan membandingkan jumlah prediksi yang benar dengan total data uji, yang dinyatakan dalam persentase menggunakan rumus berikut :

$$\text{akurasi} = \frac{93 + 156}{93 + 156 + 7 + 19} = \frac{249}{275} = 0.9054 = 0.91$$

Dengan demikian hasil perhitungan dari akurasi adalah 0.91

2. Perhitungan *precision*

Precision dihitung dengan membagi jumlah prediksi benar pada kelas positif dengan total prediksi untuk kelas tersebut, menggunakan rumus berikut:

Untuk kelas meningkat dan menurun

$$precision_{meningkat} = \frac{93}{93 + 19} = \frac{93}{112} = 0.8303$$

$$precision_{menurun} = \frac{156}{156 + 7} = \frac{156}{163} = 0.9570$$

Rata – rata *precision* (*macro average*)

$$precision = \frac{0.8303 + 0.9570}{2} = 0.8937 = 0.91$$

Dengan demikian hasil perhitungan dari *precision* adalah 0.91

3. Perhitungan *recall*

Recall dihitung dengan membagi jumlah prediksi benar pada kelas positif dengan total data aktual untuk kelas tersebut, menggunakan rumus berikut:

Untuk kelas meningkat dan menurun

$$Recall_{meningkat} = \frac{93}{93 + 7} = \frac{93}{100} = 0.93$$

$$Recall_{menurun} = \frac{156}{156 + 19} = \frac{156}{175} = 0.8914$$

Rata – rata *recall* (*macro average*)

$$Recall = \frac{0.93 + 0.8914}{2} = 0.9107 = 0.91$$

Dengan demikian hasil perhitungan dari *recall* adalah 0.91

4. Perhitungan *f1-score*

F1-score dihitung sebagai keseimbangan antara *precision* dan *recall* dengan menggunakan rata-rata, sesuai rumus berikut:

$$f1 = 2 * \frac{0.8303 * 0.93}{0.8303 + 0.93} = 0.8773$$

$$f1 = 2 * \frac{0.9570 * 0.8914}{0.9570 + 0.8914} = 0.9232$$

Rata – rata f1-score (*macro average*)

$$= \frac{0.8773 * 0.9232}{2} = 0.912 = 0.91$$

Dengan demikian hasil perhitungan dari *f1-score* adalah 0.91

4.2 Hasil Sistem

Sistem ini menggunakan aplikasi *streamlit* untuk menampilkan prediksi kualitas pemain sepak bola dengan cara yang sederhana dan interaktif.

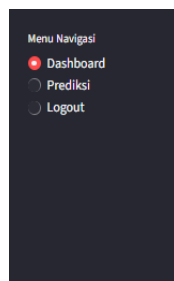
4.2.1 Halaman Login



Gambar 4. 1 Login

Gambar 4. 1 saat pertama kali membuka sistem, pengguna harus *login* terlebih dahulu dengan *username* “admin” dan *password* “admin123”.

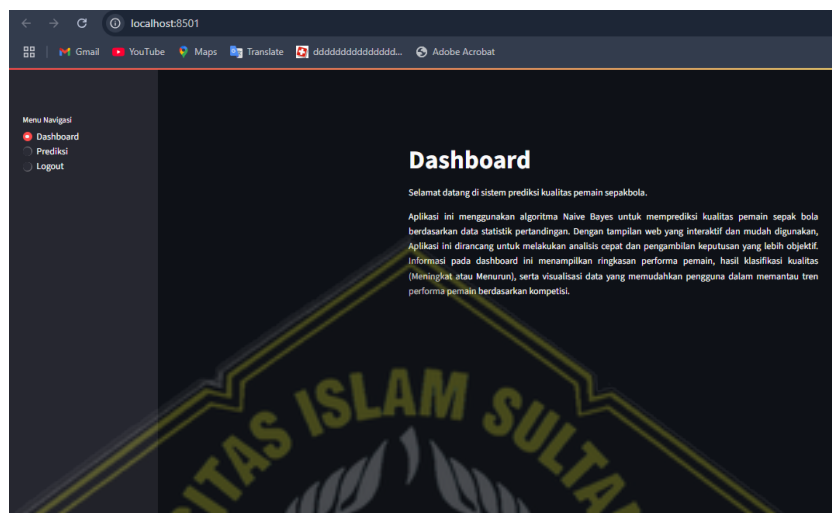
4.2.2 Halaman Menu



Gambar 4. 2 Halaman Menu

Gambar 4. 2 setelah berhasil *login*, pengguna akan melihat *menu* utama yang berisi *dashboard*, prediksi kualitas pemain sepak bola dan *logout* untuk mengelola data serta melakukan analisis kualitas pemain sepak bola.

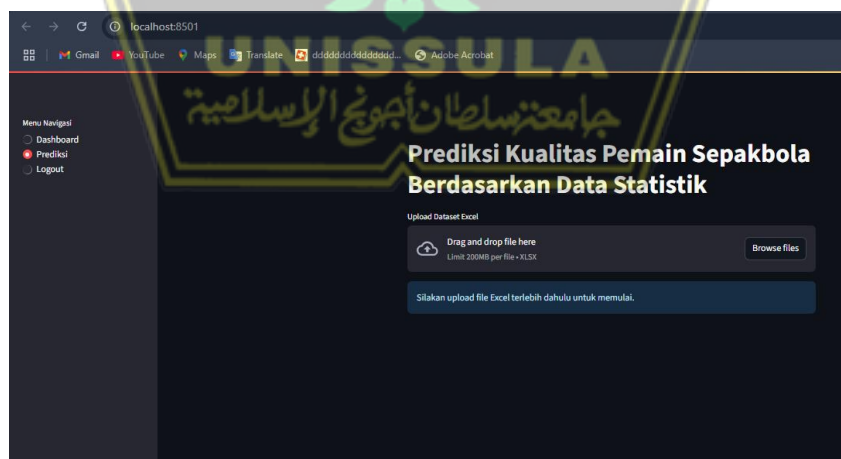
4.2.3 Halaman Utama(Dashboard)



Gambar 4. 3 Halaman *Dashboard*

Gambar 4. 3 pengguna dapat memilih menu *dashboard* pada halaman utama untuk melihat deskripsi mengenai aplikasi prediksi kualitas pemain sepak bola serta informasi terkait fitur sistem.

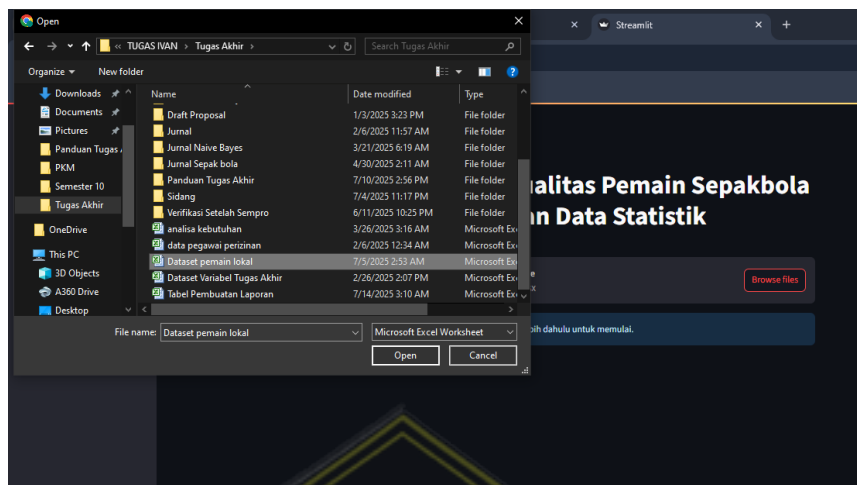
4.2.4 Halaman Prediksi Kualitas Pemain Sepak bola



Gambar 4. 4 Halaman Prediksi Kualitas Pemain Sepak Bola

Gambar 4. 4 pengguna dapat memilih menu prediksi untuk melakukan prediksi kualitas pemain sepak bola berdasarkan data yang telah dimasukkan ke dalam sistem.

1. Upload Data



Gambar 4. 5 Upload Data

Pada gambar 4. 5 langkah pertama dalam melakukan prediksi adalah mengunggah dataset pemain lokal dalam format excel ke dalam sistem.

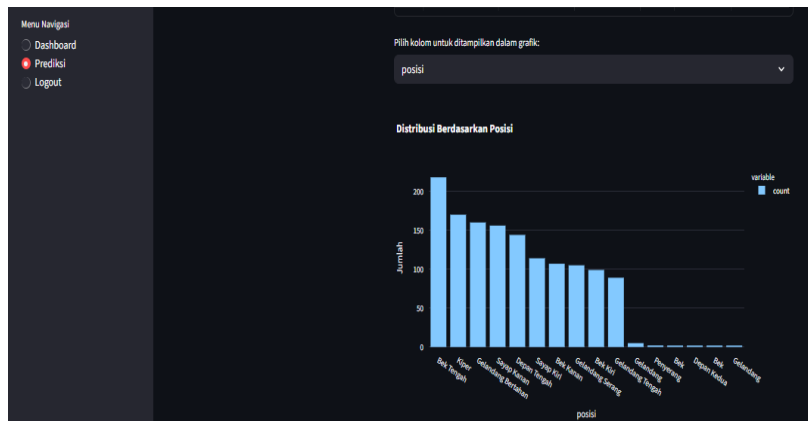
2. Data yang diunggah

no	date	nama_pemain	negara	club_id	posisi	harga_pemain
1	5 Januari 2020	Ernando Ari	Indonesia	PSBYO	Kiper	Rp 4,78 Milliar
2	2020-09-26 00:00:00	Andriko Ramadharini	Indonesia	PSBYO	Kiper	Rp 3,48 Milliar
3	5 Januari 2025	Dime Dimov	Makedonia Utara	PSBYO	Bek Tengah	Rp 7,82 Milliar
4	6 Juli 2024	Shirko Ramjanovic	Serbia	PSBYO	Bek Tengah	Rp 3,91 Milliar
5	1 Juli 2024	Ardi Idros	Indonesia	PSBYO	Bek Kanan	Rp 3,04 Milliar
6	1 Juli 2023	Kadek Raditya	Indonesia	PSBYO	Bek Tengah	Rp 3,04 Milliar
7	2023-11-08 00:00:00	Mikael Tatta	Indonesia	PSBYO	Bek Tengah	Rp 2,61 Milliar
8	2022-04-20 00:00:00	Arief Catur	Indonesia	PSBYO	Bek Kanan	Rp 1,74 Milliar

Gambar 4. 6 Data yang diunggah

Pada gambar 4. 6 setelah data diunggah, sistem akan menampilkan data yang telah dimasukkan.

3. Grafik Posisi Pemain



Gambar 4. 7 Grafik Posisi Pemain

Gambar 4. 7 kemudian pengguna dapat memilih grafik data posisi yang ditampilkan berdasarkan kualitas pemain.

4. Atribut Data



Gambar 4. 8 Atribut Data

Pada gambar 4. 8 selanjutnya, centang atribut data pada tampilan untuk menampilkan *value*.

5. Menghapus Kolom

Menghapus Kolom Tidak Diperlukan

negara	posisi	yellow_card	red_card	gol	assist	jumlah_pertandingan	kualitas_pei
Indonesia	Kiper	2	1	0	0	24	Meningkat
Indonesia	Kiper	2	0	0	0	12	Meningkat
Makedonia Utara	Bek Tengah	2	0	0	0	16	Meningkat
Serbia	Bek Tengah	2	1	0	1	29	Meningkat
Indonesia	Bek Kanan	6	1	0	0	29	Meningkat
Indonesia	Bek Tengah	3	0	0	0	27	Menurun
Indonesia	Bek Tengah	5	0	0	0	18	Meningkat
Indonesia	Bek Kanan	8	0	0	1	26	Meningkat
Indonesia	Bek Tengah	2	0	0	0	18	Menurun
Indonesia	Bek Kanan	3	0	0	0	9	Menurun

Gambar 4. 9 Menghapus Kolom

Gambar 4. 9 sistem telah menghapus beberapa kolom yaitu no, date, nama_pemain, club_id, dan harga_pemain serta menampilkan kolom yang akan dianalisis dalam sistem yaitu negara, posisi, yellow_card, red_card, gol, assist dan jumlah_pertandingan.

6. Label Encoding

Setelah Label Encoding

negara	posisi	yellow_card	red_card	gol	assist	jumlah_pertandingan	kualitas_pemain
18	12	2	1	0	0	24	0
18	12	2	0	0	0	12	0
29	4	2	0	0	0	16	0
47	4	2	1	0	1	29	0
18	2	6	1	0	0	29	0
18	4	3	0	0	0	27	1
18	4	5	0	0	0	18	0
18	2	8	0	0	1	26	0
18	4	2	0	0	0	18	1
18	2	3	0	0	0	9	1

Gambar 4. 10 Label Encoding

Gambar 4. 10 setelah data dianalisis dalam sistem, sistem melakukan proses label encoding dengan mengubah teks menjadi numerik. Dengan data dalam bentuk numerik, sistem dapat mengelolanya menggunakan algoritma *naïve bayes*.

7. Tahap Klasifikasi *Naïve Bayes*

Penerapan Klasifikasi Naive Bayes

Jumlah data pada variabel X: 1373 (8 kolom)

negara	posisi	yellow_card	red_card	gol	assist	jumlah_pertandingan
18	12	2	1	0	0	24
18	12	2	0	0	0	12
29	4	2	0	0	0	16
47	4	2	1	0	1	29
18	2	6	1	0	0	29
18	4	3	0	0	0	27
18	4	5	0	0	0	18
18	2	8	0	0	1	26
18	4	2	0	0	0	18
18	2	3	0	0	0	9

Gambar 4. 11 Tahap Klasifikasi *Naïve Bayes*

Gambar 4. 11 setelah melalui proses label encoding, tahap selanjutnya adalah klasifikasi menggunakan algoritma *naïve bayes*. Pada tahap ini, data disiapkan dengan variabel x dan y untuk proses prediksi. Jumlah data pada variabel x adalah 1.363 dengan 8 kolom.

Jumlah data pada variabel y: 1373 (1 kolom)

kualitas_pemain
0
0
0
0
0
1
0
0
1
1

Gambar 4. 12 Variabel y

Gambar 4. 12 kemudian, variabel y disiapkan dengan 1.373 data dalam 1 kolom, yaitu jumlah kualitas pemain. Variabel ini digunakan untuk menentukan hasil prediksi, apakah termasuk kategori 0 (meningkat) atau 1 (menurun).

8. Pembagian Data

Pembagian Data Training dan Testing

Jumlah data untuk training: 1098 (80%)

Jumlah data untuk testing: 275 (20%)

Data Training Variabel X:

negara	posisi	yellow_card	red_card	gol	assist	jumlah_pertandingan
18	2	4	0	0	1	13
8	4	4	0	1	0	16
18	15	0	0	2	0	8
8	9	7	0	3	1	27
18	12	0	0	0	0	0
18	14	0	0	3	2	15
18	3	5	1	0	0	15
18	9	2	0	0	1	16
18	2	3	0	0	1	11
18	9	5	0	0	0	21

Gambar 4. 13 Pembagian Data Training dan Testing

Gambar 4. 13 selanjutnya, dilakukan pembagian data, dimana data training terdiri dari 1.098 data (80%) dan testing sebanyak 275 data (20%). Data yang digunakan untuk training data pada variabel x berjumlah 1.098 data dengan 8 kolom.

Data Training Variabel y:

kualitas_pemain
1
0
0
0
1
0
1
0
0
1

Gambar 4. 14 Training Variabel y

Pada gambar 4. 14 kemudian, pada variabel y terdapat 1.098 data yang digunakan dalam proses pelatihan model.

9. Testing Data

Data Testing Variabel X:

negara	posisi	yellow_card	red_card	gol	assist	jumlah_pertandingan
18	4	0	0	0	0	9
18	10	6	0	0	2	17
18	15	1	0	0	0	8
18	15	0	0	0	0	8
18	6	2	0	8	1	22
18	10	0	0	0	0	1
18	12	1	0	0	0	13
18	3	5	0	1	4	17
27	4	2	0	0	0	16
29	4	4	1	2	2	30

Gambar 4. 15 Testing Data

Gambar 4. 15 setelah itu, data diuji (testing) menggunakan 1.098 data, yaitu 20% dari keseluruhan data, pada variabel x. proses ini bertujuan untuk memprediksi apakah kualitas pemain tersebut termasuk dalam kategori 0 (meningkat) atau 1 (menurun).

10. Hasil Prediksi

Hasil Prediksi Kualitas Pemain Berdasarkan Data Statistik

negara	posisi	yellow_card	red_card	gol	assist	jumlah_pertandingan	Prediksi kualitas_pemain	Ka
18	4	0	0	0	0	9	1	Me
18	10	6	0	0	2	17	0	Me
18	15	1	0	0	0	8	1	Me
18	15	0	0	0	0	8	1	Me
18	6	2	0	8	1	22	0	Me
18	10	0	0	0	0	1	1	Me
18	12	1	0	0	0	13	1	Me
18	3	5	0	1	4	17	0	Me
27	4	2	0	0	0	16	1	Me
29	4	4	1	2	2	30	0	Me

Gambar 4. 16 Hasil Prediksi

Gambar 4. 16 setelah data diuji, sistem telah menghasilkan prediksi kualitas pemain dengan 1.098 data yang diprediksi masuk ke dalam kategori 0

(menurun) dan 1 (meningkat). Jumlah kualitas pemain berdasarkan prediksi dalam kategori 0 (meningkat) sebanyak 456, sedangkan dalam kategori 1 (menurun) sebanyak 642.

Tabel 4. 13 Hasil Prediksi

Negara	Posisi	Yellow Card	red card	gol	Assist	Jumlah pertandingan	kualitas pemain
Indonesia	Bek Tengah	0	0	0	0	9	Menurun
Indonesia	Gelandang Serang	6	0	0	2	17	Meningkat
Indonesia	Sayap Kiri	1	0	0	0	8	Menurun
Indonesia	Sayap Kiri	0	0	0	0	8	Menurun
Indonesia	Depan Tengah	2	0	8	1	22	Meningkat

Tabel 4.13 menampilkan hasil prediksi kualitas pemain berdasarkan delapan data pemain sepak bola, di mana pemain dengan catatan statistik seperti jumlah gol, assist, kartu kuning, kartu merah, total pertandingan serta kualitas pemain yang dimainkan di klubnya menunjukkan bahwa empat pemain diprediksi memiliki kualitas yang meningkat, sedangkan satu pemain diprediksi mengalami penurunan kualitas.

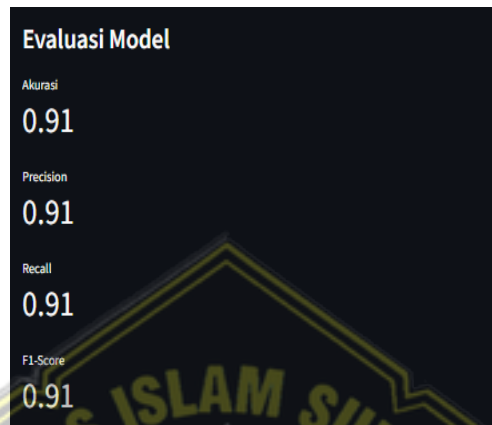
11. Probabilitas Prediksi

Probabilitas Prediksi	
Probabilitas Meningkat	Probabilitas Menurun
3.0114657433e-04	9.9969885343e-01
9.9999999989e-01	1.1483176360e-10
4.0360065997e-04	9.9959639934e-01
3.3553822716e-04	9.9966446177e-01
1.0000000000e+00	0.0000000000e+00
1.2509591358e-04	9.9987490409e-01
6.8730936567e-04	9.9931269063e-01
1.0000000000e+00	2.2574250941e-52
1.1743285833e-02	9.8825671417e-01
1.0000000000e+00	1.0234922599e-36

Gambar 4. 17 Probabilitas Prediksi

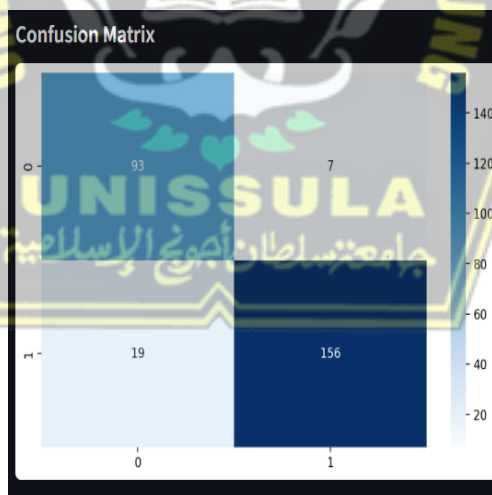
Gambar 4. 17 hasil prediksi kategori menurun dan meningkat menampilkan probabilitas untuk setiap data dari 257 data yang telah diprediksi.

12. Evaluasi Model



Gambar 4. 18 Hasil Evaluasi Model

Pada gambar 4. 18 evaluasi model dilakukan setelah memperoleh hasil prediksi dengan menampilkan nilai *akurasi* 0. 91, *precision* 0. 91, *recall* 0.91 dan *f1-score* 0.91 .



Gambar 4. 19 *Confusion Matrix*

Pada gambar 4. 19 setelah itu, sistem menampilkan visualisasi *confusion matrix* untuk menganalisis performa hasil prediksi.

Prediksi Pemain Baru Berdasarkan Input Manual

Nama Pemain:

Pilih Negara:

Pilih Posisi:

Jumlah Kartu Kuning:

Jumlah Kartu Merah:

Jumlah Gol:

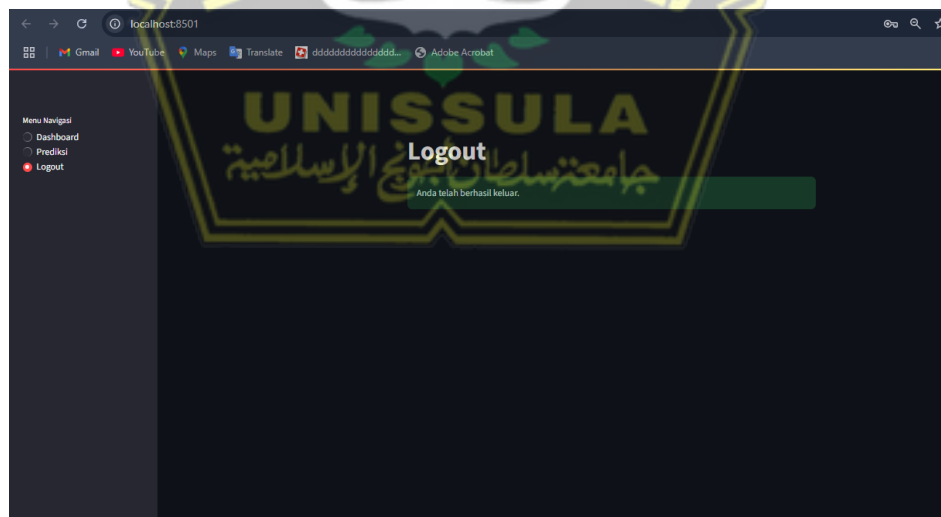
Jumlah Asist:

Jumlah Pertandingan:

Gambar 3. 39 Prediksi Pemain Baru Berdasarkan *Input* Manual

Pada gambar 3.39 fitur ini dapat melakukan prediksi kualitas pemain sepak bola dengan cara memasukkan data secara manual melalui form yang telah disediakan pada sistem. Fitur ini memudahkan manajer tim untuk mengevaluasi pemain baru sehingga dapat digunakan sebagai bahan pertimbangan dalam proses seleksi dan strategi tim.

4.2.5 Halaman *Logout*



Gambar 4. 20 *Logout*

Pada gambar 4. 20 setelah selesai menggunakan sistem, pengguna dapat keluar melalui menu *logout*, dan sistem akan menampilkan notifikasi “Anda telah berhasil keluar”.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil penelitian yang telah dilakukan, dapat disimpulkan bahwa algoritma *Naïve Bayes* mampu memberikan hasil yang cukup akurat dalam mengklasifikasikan kualitas pemain sepak bola berdasarkan data statistik pertandingan. Model ini berhasil memproses data seperti jumlah gol, assist, kartu kuning, kartu merah, dan jumlah pertandingan menjadi informasi klasifikasi yang dapat membedakan antara pemain yang mengalami peningkatan atau penurunan performa. Dengan akurasi sebesar 91% serta nilai *precision*, *recall*, dan *F1-score* yang seimbang, sistem ini terbukti efektif dalam membantu proses evaluasi performa pemain secara objektif, menggantikan penilaian subjektif yang selama ini umum digunakan.

Selain itu, sistem klasifikasi yang dibangun berbasis web dengan menggunakan framework *Streamlit*, memberikan antarmuka yang interaktif dan mudah digunakan oleh pengguna, seperti pelatih atau manajer tim. Fitur-fitur seperti upload data, klasifikasi otomatis, visualisasi hasil prediksi, serta evaluasi model telah diintegrasikan dengan baik. Dengan demikian, aplikasi ini dapat menjadi solusi praktis dalam mendukung proses seleksi pemain dan pengambilan keputusan dalam manajemen tim sepak bola secara data-driven dan efisien. Penelitian ini juga membuka peluang pengembangan lebih lanjut untuk integrasi sistem dengan data *real-time* dan perbandingan dengan algoritma lain.

5.2 Saran

Adapun saran penelitian ini untuk ke depannya sebagai berikut:

1. Pengembangan Data: Penelitian selanjutnya disarankan menggunakan dataset yang lebih besar dan beragam, mencakup lebih banyak musim kompetisi, posisi pemain, dan liga berbeda, agar model lebih general dan akurat.

2. Validasi Model: Untuk hasil evaluasi yang lebih robust, metode validasi lain seperti *k-fold cross-validation* sebaiknya diterapkan guna menghindari bias dari pembagian data latih dan uji yang tetap.
3. Perbandingan Algoritma: Sebaiknya dilakukan perbandingan performa antara *Naïve Bayes* dengan algoritma lain seperti *Decision Tree*, *Random Forest*, atau *SVM* agar dapat diketahui metode mana yang paling optimal dalam kasus ini.
4. Integrasi Sistem Nyata: Aplikasi prediksi yang dibangun sebaiknya dikembangkan lebih lanjut menjadi sistem yang terintegrasi dengan data *real-time* atau *API* dari *platform* seperti *Transfermarkt* agar hasil analisis selalu *up-to-date*.



DAFTAR PUSTAKA

- Astuti, Y., Wulandari, I. R., Putra, A. R., & Kharomadhona, N. (2022). Naïve Bayes untuk Prediksi Tingkat Pemahaman Kuliah Online Terhadap Mata Kuliah Algoritma Struktur Data. *Jurnal Edukasi Dan Penelitian Informatika (JEPIN)*, 8(1), 28. <https://doi.org/10.26418/jp.v8i1.48848>
- Damanik, A. R., Sumijan, S., & Nurcahyo, G. W. (2021). Prediksi Tingkat Kepuasan dalam Pembelajaran Daring Menggunakan Algoritma Naïve Bayes. *Jurnal Sistim Informasi Dan Teknologi*, 3, 88–94. <https://doi.org/10.37034/jsisfotek.v3i3.49>
- Dang Kurniawan, D. (2025). METODE NAÏVE BAYES UNTUK KLASIFIKASI SENTIMEN TWEET PEMAIN NATURALISASI TIM NASIONAL SENIOR SEPAK BOLA INDONESIA. In *Jurnal Mahasiswa Teknik Informatika* (Vol. 9, Issue 2).
- Dania, S., Ishak, R., Kom, M., & Dalai, H. (2024). Penerapan Algoritma Naive Bayes Classifier Untuk Klasifikasi Judul Skripsi Berdasarkan Konsentrasi. *Jurnal Ilmiah Ilmu Komputer Banthayo Lo Komputer*, 3(1), 15–22.
- Erinsyah, M. F., Sasmito, G. W., Wibowo, D. S., & Bakti, V. K. (2024). Sistem Evaluasi Pada Aplikasi Akademik Menggunakan Metode Skala Likert Dan Algoritma Naïve Bayes. *Komputa : Jurnal Ilmiah Komputer Dan Informatika*, 13(1), 74–82. <https://doi.org/10.34010/komputa.v13i1.10940>
- Herdian, C., Kamila, A., & Agung Musa Budidarma, I. G. (2024). Studi Kasus Feature Engineering Untuk Data Teks: Perbandingan Label Encoding dan One-Hot Encoding Pada Metode Linear Regresi. *Technologia : Jurnal Ilmiah*, 15(1), 93. <https://doi.org/10.31602/tji.v15i1.13457>
- Humam, M. N. (2023). Perbandingan Kinerja CNN dan Naïve Bayes pada Analisis Sentimen Performa Manchester United di Twitter. *Journal of Information Engineering and Educational Technology*, 7(2), 83–91. <https://doi.org/10.26740/jieet.v7n2.p83-91>
- Laia, M. P., Hafidz, R. M., & Hestiani, M. D. (2023). Proses Pengumpulan Data Dan Keakuratan Aplikasi Transfermarkt Sebagai Acuan Dari Harga Transfer Pemain Sepakbola Dunia. *OKTAL : Jurnal Ilmu Komputer Dan Science*, 2(5), 1526–1533.
- Material, P., Sandy, B., & Sitohang, S. (2022). Penerapan Algoritma Naive Bayes Dalam Memprediksi. *Jurnal Comasie*, 4(2), 848–858.
- Metode, P., Algoritma, C., & Naive, D. A. N. (2024). Perbandingan metode algoritma c4.5 dan naive bayes untuk memprediksi penjualan kosmetik pada toko jelita 1,2. 7(2), 220–225.

- Musrifin, A. Y., & Bausad, A. A. (2020). Analisis Unsur Kondisi Fisik Pemain Sepak Bola Mataram Soccer Akademi Ntb. *Jurnal Ilmiah Mandala Education*, 6(1), 113–119. <https://doi.org/10.58258/jime.v6i1.1116>
- Pradana, D., Luthfi Alghifari, M., Farhan Juna, M., & Palaguna, D. (2022). Klasifikasi Penyakit Jantung Menggunakan Metode Artificial Neural Network. *Indonesian Journal of Data and Science*, 3(2), 55–60. <https://doi.org/10.56705/ijodas.v3i2.35>
- Prasetyo, A. B., & Laksana, T. G. (2022). Optimasi Algoritma K-Nearest Neighbors dengan Teknik Cross Validation Dengan Streamlit (Studi Data: Penyakit Diabetes). *Journal of Applied Informatics and Computing (JAIC)*, 6(2), 194. <http://jurnal.polibatam.ac.id/index.php/JAIC>
- Pratama, F. F., & Nurhasanah, Y. I. (2020). Penggunaan Metode Profile Matching Dan Naïve Bayes Untuk Menentukan Starting Eleven Pada Sepak Bola. *Jurnal Tekno Insentif*, 14(2), 59–68. <https://doi.org/10.36787/jti.v14i2.268>
- Priyo Utomo, N., & Indarto, P. (2021). Analisis Keterampilan Teknik Dasar Passing dalam Sepak Bola. *Jurnal Porkes*, 4(2), 87–94. <https://doi.org/10.29408/porkes.v4i2.4578>
- Putra, I. N. T. A., Kartini, K. S., Suyitno, Y. K., Sugiarta, I. M., & Puspita, N. K. E. (2023). Penerapan Library Tensorflow, Cyzone, dan Numpy pada Sistem Deteksi Bahasa Isyarat Secara Real Time. *Jurnal Krisnadana*, 2(3), 412–423. <https://doi.org/10.58982/krisnadana.v2i3.335>
- Sartika, D., Andreswari, D., & Anggriani, K. (2023). Penentuan Posisi Ideal Pemain Dalam Cabang Olahraga Sepak Bola Dengan Menggunakan Pendekatan Dua Metode Naïve Bayes & Profile Matching. *Jurnal Rekursif / ISSN 2303-0755*, 4(3), 311–324.
- Sinaga, S., Sembiring, R. W., & Sumarno, S. (2022). Penerapan Algoritma Naive Bayes untuk Klasifikasi Prediksi Penerimaan Siswa Baru. *Journal of Machine ...*, 1(1), 55–64. <https://journal.fkpt.org/index.php/malda/article/view/162%0Ahttps://journal.fkpt.org/index.php/malda/article/download/162/115>
- Suryanto, T. L. M., Nuryananda, P. F., & Wibowo, N. C. (2023). Mangunjaya Watch: Sekolah Sepakbola Berbasis Teknologi dan Analisis Data Digital. *I-Com: Indonesian Community Journal*, 3(4), 1571–1582. <https://doi.org/10.33379/icom.v3i4.3262>
- Wahyudin, Y., & Rahayu, D. N. (2020). Analisis Metode Pengembangan Sistem Informasi Berbasis Website: A Literatur Review. *Jurnal Interkom: Jurnal Publikasi Ilmiah Bidang Teknologi Informasi Dan Komunikasi*, 15(3), 26–40. <https://doi.org/10.35969/interkom.v15i3.74>