BAB I

PENDAHULUAN

A. Latar Belakang

Permintaan dalam industri transportasi, terutama di sektor kapal laut, cenderung berubah-ubah. Pemesanan tiket dipengaruhi oleh sejumlah faktor, mulai dari musim, tanggal keberangkatan, harga tiket, hingga kejadian-kejadian khusus lainnya. Bagi PT. Abadi yang bergerak di bidang penjualan tiket kapal laut, seringkali mengalami kekurangan tiket untuk penumpang, terutama pada musim puncak atau saat ada permintaan mendadak yang tidak terprediksi. Karena itu, ketidakpastian dalam permintaan tiket tetap menjadi tantangan utama yang harus diatasi. Untuk menjawab tantangan ini, penerapan teknologi machine learning seperti sistem prediksi dapat memberikan solusi yang efektif.

Prediksi pemesanan tiket merupakan proses analisis dan peramalan tren pembelian tiket oleh konsumen dalam rentang waktu tertentu. Menurut Wahid (2021), Keakuratan dalam memprediksi pemesanan tiket tidak hanya memberikan keuntungan finansial bagi perusahaan, tetapi juga meningkatkan pengalaman pelanggan. Menurut firmansyah dan kolega (2020) bahwa penggunaan algoritma untuk menganalisis data penjualan tiket dapat menyediakan wawasan yang lebih mendetail mengenai pola pembelian. Lebih lanjut, Septiadi et al. (2016) menegaskan bahwa model prediksi tidak hanya membantu dalam perencanaan kapasitas, tetapi juga dalam strategi pemasaran yang ditargetkan. Oleh karena itu, penerapan metode prediksi tidak hanya berkisar pada analisis data, tetapi juga

berperan sebagai pilar dalam pengambilan keputusan strategis untuk memaksimalkan potensi pasar dan meningkatkan profitabilitas perusahaan.

Untuk meningkatkan akurasi dan efisiensi dalam sistem prediksi pemesanan tiket, penerapan algoritma *machine learning* yang tepat sangat diperlukan. Salah satu metode yang terbukti efektif dalam mengatasi kompleksitas dan variabilitas data adalah algoritma *Random Forest*. Menerapkan algoritma *random forest* pada sistem prediksi pemesanan tiket merupakan langkah penting dalam meningkatkan akurasi dan efisiensi sistem prediksi. Algoritma *random forest*, yang dikenal karena kemampuannya dalam menangani dataset yang kompleks dan besar, dapat memberikan hasil prediksi yang lebih akurat dibandingkan metode lain. Menurut Rianto, Marchell, dan Roni Yunis (2021), penerapan *random forest* telah terbukti mampu mengatasi variabilitas data yang tinggi dan meminimalisir kesalahan prediksi. Keandalan algoritma *random forest* dalam menangani berbagai tipe data dan menghasilkan prediksi yang akurat menjadikannya pilihan yang ideal untuk sistem prediksi pemesanan tiket.

Keputusan untuk mengintegrasikan algoritma *random forest* ke dalam sistem prediksi menjadi sangat relevan dalam konteks bisnis saat ini. Studi oleh Pramesti dan Baihaqi (2023) menegaskan bahwa *random forest* dapat memberikan keunggulan kompetitif dalam memprediksi jumlah transaksi ojek online dibandingkan regresi linier. Hal ini menunjukkan bahwa *random forest* mampu beradaptasi dengan perubahan pola data yang dinamis. Sementara itu, Siburian dan Mulyana (2019) juga menemukan bahwa metode ini efektif dalam prediksi harga ponsel, menyoroti fleksibilitas dan akurasinya. Dengan demikian, dalam konteks PT. Abadi, penerapan *random forest* diharapkan dapat meningkatkan keakuratan

prediksi pemesanan tiket, mengurangi risiko *overbooking,* serta mengoptimalkan strategi pemasaran dan pengelolaan kapasitas. Oleh karena itu, penerapan algoritma ini bukan hanya sekadar peningkatan teknologi, tetapi juga esensial untuk mencapai efisiensi operasional dan meningkatkan kepuasan pelanggan.

Penerapan algoritma *random forest* dalam sistem prediksi pemesanan tiket di PT. Abadi bertujuan untuk menguji pengaruh penggunaan teknologi ini terhadap peningkatan keakuratan prediksi pemesanan tiket, sekaligus mengurangi risiko kekurangan tiket dan mengoptimalkan strategi pemasaran serta pengelolaan kapasitas. Dengan meningkatnya kompleksitas dan volume data yang dihadapi PT. Abadi, algoritma *random forest* menawarkan jalan keluar yang lebih efektif dalam menangani keanekaragaman dan variabilitas data dibandingkan dengan metode tradisional. Pentingnya penelitian ini terletak pada kemampuannya untuk menyesuaikan diri dengan dinamika pasar yang terus berubah, yang akhirnya dapat mencapai efisiensi operasional lebih tinggi dan meningkatkan kepuasan pelanggan. Selain itu, penerapan metode ini juga dapat membuka peluang baru dalam pengembangan strategi bisnis perusahaan yang lebih inovatif dan responsif terhadap kebutuhan pasar. Dalam konteks bisnis modern, kemampuan untuk memanfaatkan big data melalui penerapan *machine learning* seperti *random forest* menjadi tidak hanya sekadar keunggulan teknis, tetapi juga suatu keharusan untuk menjamin keberlanjutan dan pertumbuhan perusahaan. Dengan demikian, penelitian ini tidak hanya berkontribusi terhadap literatur ilmiah, tetapi juga memberikan dampak nyata bagi pengembangan strategi bisnis berbasis data di PT. Abadi.

B. Rumusan Masalah

Berdasarkan latar belakang yang sudah diuraikan, dapat disimpulkan bahwa rumusan masalah pada penelitian ini yaitu :

1. Bagaimana merancang sistem prediksi menggunakan algoritam *random forest* pada PT.Abadi polacitra mandiri?
2. Bagaimana mengimplementasikan sistem prediksi sehingga memberikan kontribusi kepada PT. Abadi polacitra mandiri?

C. Tujuan Penelitian

Tujuan dari penelitian ini adalah

1. Membuat sebuah sistem prediksi pemesanan tiket kapal menggunakan algoritma *random forest* pada PT.Abadi polacitra mandiri.
2. Sistem yang kami bangun diharapkan mempu memberikan kontribusi pada PT. Abadi polacitra mandiri dalam hal ketersediaan tiket, dalam rangka meningkatkan kepuasan konsumen sebagai salah satu standar dalam sistem pelayanan dan penjualan.

D. Batasan Masalah

Dalam penelitian ini ruang lingkup analisis dibatasi secera spesifik untuk memastikan terfokus dan tidak meluas dari pembahasan. Adapun batasan-batasan masalah telah diterapkan meliputi:

1. Metode prediksi yang digunakan hanya berfokus pada algoritma *random forest*, dan tidak membahas secara mendalam metode prediksi lainnya seperti *k-nearest neighbor* dan apriori.

2. Prediksi yang dilakukan hanya mencakup jumlah pemesanan tiket dan tidak mencakup prediksi pendapatan, biaya operasional, atau aspek keuangan lainnya.
3. Penelitian ini hanya ditujukan pada satu perusahaan *travel* yaitu PT. Abadi polacitra mandiri.
4. Data yang digunkan untuk sistem prediksi yaitu pemesanan tiket pada 4 tahun terakhir sehingga faktor eksternal seperti kondisi cuaca atau bencana alam tidak dimasukkan dalam analisis.

E. Manfaat Penelitian

Penelitian ini diharapkan dapat memberikan manfaat bagi berbagai pihak. Adapun manfaatnya sebagai berikut :

1. Dengan menggunakan sistem prediksi berbasis *Random Forest*, perusahaan *travel* dapat lebih mudah memprediksi jumlah pemesanan tiket kapal secara akurat dan mengurangi terjadinya ketidaktersediaan tiket
2. Penelitian ini memberikan pemahaman lebin dalam mengenal kelebihan dan kelemahan penggunaan algoritma *Random Forest* dalam prediksi permintaan tiket di industri transportasi laut, yang dapat digunakan untuk pengembangan penelitian lebih lanjut dalam bidang yang sama.

BAB II

TINJAUAN PUSTAKA

A. Kajian Teori

1. Sistem Prediksi

Menurut Maulana dan rekan-rekan (2021), prediksi merupakan proses sistematis untuk memperkirakan kemungkinan kejadian di masa depan dengan memanfaatkan data masa lalu dan informasi saat ini, sehingga dapat meminimalkan potensi kesalahan. Meskipun tidak selalu menghasilkan jawaban yang pasti, prediksi bertujuan untuk mendekati hasil yang paling mungkin terjadi. Dalam konteks perencanaan dan pengambilan keputusan, prediksi berfungsi sebagai alat untuk memperkirakan apa yang mungkin terjadi dalam kondisi tertentu.

Roza (2020) menyatakan bahwa prediksi adalah gabungan antara seni dan ilmu untuk memperkirakan kejadian yang akan datang. Peramalan biasanya melibatkan penggunaan data masa lalu, seperti data penjualan tahun sebelumnya, yang kemudian diterapkan dalam model matematika untuk memprediksi masa depan. Bisnis membutuhkan prediksi untuk mengetahui berapa banyak permintaan yang akan ada untuk produk mereka. Dalam Kamus Besar Bahasa Indonesia, prediksi diartikan sebagai hasil dari aktivitas meramalkan atau memperkirakan nilai atau kejadian yang akan datang berdasarkan informasi yang ada.

Peramalan (*forecasting*) adalah suatu proses yang digunakan untuk menghasilkan informasi yang akurat mengenai keadaan sosial di masa depan, dengan berlandaskan pada informasi yang sudah ada terkait masalah kebijakan.

Dalam penelitiannya, Saras (2023) menjelaskan bahwa ada tiga bentuk utama dalam ramalan:

a. Proyeksi

   Proyeksi merupakan ramalan yang dibangun dengan cara meng-ekstrapolasi data dari kecenderungan masa lalu dan saat ini untuk meramalkan keadaan di masa depan. Proyeksi berfokus pada pertanyaan yang jelas berdasarkan argumen yang didapat melalui metode tertentu dan kasus serupa.

b. Prediksi

   Prediksi adalah ramalan yang didasari oleh asumsi teoritis yang spesifik, yang dapat berupa hukum atau prinsip teori, proposisi, atau analogi yang relevan.

c. Perkiraan

   Perkiraan (*conjecture*) adalah ramalan yang didasarkan pada evaluasi yang informatif atau pendapat ahli mengenai apa yang akan terjadi di masa depan

2. Data Mining

Data mining adalah proses yang mencakup pengumpulan dan penggunaan data historis untuk menemukan pola, keteraturan, dan hubungan dalam kumpulan data besar. Proses ini juga dikenal sebagai serangkaian langkah untuk menggali nilai tambah berupa pengetahuan yang sebelumnya tidak diketahui secara manual dari kumpulan data. Data mining merupakan langkah untuk menemukan pola atau informasi dalam dataset dengan menggunakan teknik dan algoritma tertentu (Kamal, dkk, 2017).

Berdasarkan definisi data mining menurut para ahli, dapat disimpulkan bahwa data mining adalah suatu proses otomatis untuk mencari data yang dapat menghasilkan model dari basis data yang besar.

Data mining lebih dari sekadar proses analisis data konvensional; ia merupakan upaya sistematis untuk mengungkap informasi penting, pola, dan pengetahuan yang mungkin tersembunyi dalam dataset yang umumnya besar dan kompleks (Fauzi, 2024).

Menurut Noviayanto (2020) data Mining dibagi menjadi dua kategori utama yaitu:

a. Dapat Prediktif Proses untuk menemukan karakteristik penting dari data dalam satu basis data. Teknik data mining yang termasuk *descriptive* mining adalah *clustering*, *asosiation*, dan *sequential* mining.
b. Deskriptif Proses untuk menemukan pola dari data dengan menggunakan beberapa variabel lain di masa depan. Salah satu teknik yang terdapat dalam *predictive* mining adalah klasifikasi.

3. *Big Data*

Menurut Santoso (2020) *Big data* hadir sebagai kekuatan yang dapat mengubah paradigma di dunia bisnis, memengaruhi sektor-sektor, industri, dan ekonomi secara luas. Pemanfaatan analisis data yang lebih baik bisa mempercepat kemajuan dalam berbagai disiplin ilmu dan membantu banyak bisnis meraih keuntungan yang lebih tinggi

Big data merujuk pada koleksi data yang sangat besar dan kompleks, yang mencakup beragam informasi dan membutuhkan teknologi serta metode analisis

khusus untuk dapat diproses dan dimanfaatkan secara optimal. Pertumbuhannya didorong oleh perkembangan pesat dalam teknologi digital, peningkatan jumlah perangkat yang terhubung dengan internet, dan kemajuan dalam kemampuan penyimpanan serta pengolahan data (Ibna dkk, 2024).

Menurut Putra et al (2023) *Big data* memiliki tiga karakteristik utama yang dikenal dengan istilah Tiga V: yaitu :

a. *Volume*

   Big data berhubungan dengan data dalam jumlah besar, di mana ukuran data memainkan peran kunci dalam menentukan apakah data tersebut termasuk dalam kategori *Big data*. Karena itu, *volume* data harus dipertimbangkan dalam pengelolaannya.

b. *Velocity*

   Ini merujuk pada kecepatan penghasilannya, pemrosesannya, dan analisisnya untuk memenuhi tujuan tertentu. Kecepatan dalam pengumpulan, transfer, dan analisis data sangat vital. *Big data* yang diproses dengan cepat memungkinkan penggunaannya dalam waktu nyata.

c. *Variety*

   Variety mencakup berbagai tipe data dalam *Big data*. Sementara data tradisional lebih terstruktur, *Big data* juga mencakup data yang tidak terstruktur dan semi-terstruktur, seperti audio, video, dan teks, yang membutuhkan pemrosesan lebih lanjut untuk mengungkapkan makna dan nilai data tersebut.

4. *Excel*

Microsoft Office Excel merupakan program yang digunakan untuk memproses data angka dengan menggunakan *spreadsheet* yang menyusun data dalam bentuk baris dan kolom untuk menjalankan perintah. *Excel* pada dasarnya memanfaatkan spreadsheet untuk mengelola data serta melakukan berbagai fungsi yang lebih dikenal dengan formula *Excel*. Program ini juga memungkinkan pembuatan aplikasi yang berhubungan dengan matematika.

Bill Gates memperkenalkan *Microsoft Office* sekitar tahun 1988. Jika tidak ada *Microsoft Office*, komputer mungkin tidak akan dapat melakukan proses, penyimpanan, dan kompilasi data dengan efisien, yang menjadikannya perangkat lunak penting bagi komputer. *Microsoft Exce*l, yang merupakan aplikasi untuk mengolah data atau angka, sudah digunakan secara luas. Ini adalah aplikasi spreadsheet yang dikembangkan oleh Microsoft Corporation dan dapat dijalankan di *Microsoft Windows* maupun Mac OS..

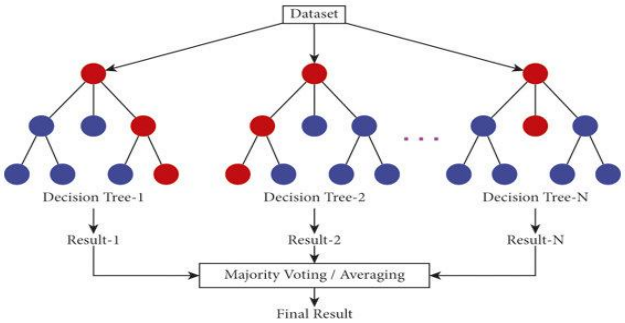
Excel dimanfaatkan oleh banyak profesi, mulai dari bidang akuntansi, keuangan, industri, statistik, penelitian, hingga sektor akademik. *Microsoft Excel* adalah aplikasi yang dikenal dengan fitur kalkulasi dan pembuatan grafik yang paling banyak digunakan pada komputer mikro hingga saat ini. Bahkan, Excel adalah *software spreadsheet* yang paling banyak digunakan di seluruh dunia. Sebagai *spreadsheet, Excel* memiliki berbagai fungsi, mulai dari mengolah, mengubah, mengurutkan, hingga menganalisis data, dan juga dapat menyajikan data dalam bentuk grafik atau diagram.

5. *Website*

Situs web merujuk pada serangkaian halaman web yang umumnya beroperasi di bawah satu nama domain atau subdomain di *World Wide Web* (WWW) yang dapat diakses melalui Internet (Noviantoro, dkk, 2022). Halaman web adalah dokumen yang dibuat dengan format HTML (*Hyper Text Markup Language*), yang hampir selalu diakses menggunakan protokol HTTP, yang mengirimkan data dari *server website* dan menampilkannya pada web *browser* pengguna. Halaman-halaman ini bisa bersifat statis maupun dinamis, di mana masing-masing halaman saling terhubung melalui jaringan halaman web (Haryono, dkk, 2021).

Website dapat didefinisikan sebagai kumpulan halaman web yang saling terhubung dan dapat diakses melalui jaringan internet. Setiap halaman web umumnya berisi teks, gambar, video, dan elemen multimedia lainnya yang dirancang untuk memberikan informasi atau pengalaman interaktif kepada pengunjungnya. *Website* dapat diakses melalui peramban web (*browser*) seperti *Google Chrome, Mozilla Firefox*, atau Safari (Rizaldi, dkk, 2023).

6. *Random Forest*



Gambar 2. 1 Model Arsitektur Random Forest
(Sumber : https://www.trivusi.web.id/2022/08/algoritma-random-forest.html)

Random Forest merupakan algoritma dalam *machine learning* yang digunakan untuk pengklasifikasian dataset dalam jumlah besar. Karena fungsinya bisa dipakai dengan banyak dimensi dan dengan berbagai skala dan performa yang tinggi (Khalim, dkk, 2023). Algoritma *random forest* (RF) menggunakan pemisah biner secara rekursif untuk mencapai node akhir dalam struktur pohon keputusan, yang pada dasarnya berfungsi sebagai pohon klasifikasi dan regresi. Dalam metode *random forest*, beberapa pohon dibuat untuk membentuk hutan (*forest*), dan analisis dilakukan pada seluruh pohon tersebut. *Random forest* menghasilkan sejumlah pohon yang saling independen dengan memilih subset acak dari data pelatihan melalui teknik *bootstrap* dan memilih subset acak dari variabel input. (Rakhmat, dkk, 2023).

Random forest adalah pengklasifikasi yang terdiri dari pohon *{h(x, k), k = 1, . . .}* di mana k adalah vektor acak yang didistribusikan secara independen dan setiap pohon komponen memilih kelas yang paling populer dengan nilai input x (Listiono et al., 2021). Cara kerja algoritma ini adalah dengan membuat beberapa pohon keputusan dan menyartukannya untuk memperoleh prediksi yang lebih stabil serta akurat. Hutan (Forest) yang dibangun oleh algoritma tersebut merupakan kumpulan pohon keputusan dimana biasanya dilatih dengan metode *bagging*, yang diharapkan menghasilkan model *machine learning* yang lebih baik (Sudrajat et al., 2022)

Menurut Breiman (2001) *Random Forest* adalah sebuah pengklasifikasi yang terdiri dari kumpulan pohon pengklasifikasi yang *terstruktur* $\{h(x, \Theta_k), k = 1,. . .\}$ *dimana* $\{\Theta_k\}$ adalah vektor acak independen yang terdistribusi secara identik

dan setiap pohon akan memberikan satu suara untuk kelas yang paling populer pada input x. Dengan kata lain, *Random Forest* merupakan sekelompok pohon klasifikasi atau regresi yang belum dipangkas, yang dilatih pada sampel *bootstrap* dari data training menggunakan pemilihan fitur acak saat proses pembuatan pohon.

Dalam penelitiannya, Okta (2022) menguraikan bahwa *Random forest* dibagi dalam bebrapa karakteristik yaitu :

a. Memusatkan *Random Forest*

Random Forest berfokus pada pengelompokan data yang lebih kuat dan akurat dengan menggunakan beberapa pohon keputusan yang saling bekerja sama untuk meminimalkan kesalahan dan meningkatkan ketepatan prediksi.

b. *Random Forest* Menggunakan Seleksi Input yang *Random*

Bagging digunakan untuk memilih fitur secara acak. Setiap set pelatihan diganti dengan set pelatihan asli. Satu set pelatihan kemudian diunggulkan menggunakan pemilihan fitur acak.

c. *Random Forest* Menggunakan Kombinasi Input yang *Linear*

Seandainya terdapat inputan bernilai M, F mengambil unit pada M yang akan menjadi tujuan utama dalam peningkatan hubungan yang tinggi. Variabel L secara *random* diseleksi dan ditambahkan bersama dengan koefisien.

Algoritma *Random Forest* dipilih karena kemampuannya dalam menangani data yang kompleks, tidak memerlukan asumsi linearitas, dan memiliki interpretabilitas yang baik.

Langkah-langkah pengembangan model meliputi:

1. Pembagian data membagi dataset menjadi data training (80%) dan data test (20%).

2. Pelatihan model melatih model *Random Forest* pada data latih dengan menentukan jumlah pohon sebanyak 200 pohon
3. Evaluasi model mengevaluasi kinerja model pada data uji menggunakan metrik seperti *Mean Absolute Error* (MAE), *Root Mean Squared Error* (RMSE), dan *R-squared* (R^2). Metrik-metrik ini memberikan gambaran mengenai seberapa baik model dalam memprediksi harga berdasarkan data yang ada.
4. Prediksi akhir dihitung dengan mengambil rata-rata dari semua hasil prediksi pohon keputusan, yaitu dengan rumus:

$$\hat{y}\mathcal{RF} = \frac{1}{n}\sum_{i=1}^{n}\hat{y} \quad \text{.....(1)}$$

Keterangan

$\hat{y}\mathcal{RF}$: prediksi akhir yang dihasilkan *random forest*

n : jumlah pohon keputusan dalam *random forest*

$\hat{y}i$: prediksi yang dihasilakn pohon-i

5. *Python*



Gambar 2. 2 Logo *python*
(Sumber : https://algorit.ma/blog/belajar-apa-itu-python)

Python merupakan bahasa pemrograman interpretatif yang memiliki banyak kegunaan, dengan fokus desain pada keterbacaan kode. *Python* dikenal sebagai bahasa yang menggabungkan berbagai kapabilitas dan kemampuan dengan

sintaksis yang mudah dipahami, serta menyediakan pustaka standar yang besar dan menyeluruh. (Syahrudin, dkk, 2018). *Python* mendukung banyak paradigma pemrograman, termasuk pemrograman berorientasi objek, imperatif, dan fungsional, meskipun tidak terbatas pada ketiganya. Salah satu keunggulan *Python* adalah sebagai bahasa pemrograman dinamis yang memiliki manajemen memori otomatis.

Bahasa pemrograman *Python* pertama kali diperkenalkan oleh Guido van Rossum pada tahun 1991, setelah dikembangkan sejak tahun 1989. Nama *Python* tidak diambil langsung dari nama ular piton, melainkan terinspirasi dari sebuah acara komedi di BBC pada tahun 1980-an yang berjudul "*Monty Python's Flying Circus*".

Menurut Jamaluddin (2021), *Python* menawarkan berbagai keunggulan yang membuatnya sangat disukai oleh penggunanya dan menjadi pilihan utama dibandingkan bahasa pemrograman lainnya. Kode dalam *Python* dirancang agar mudah dibaca, dipelajari, dapat digunakan kembali, dan mudah dirawat. Selain itu, *Python* bersifat gratis, *open source*, dan dapat digunakan untuk tujuan komersial. Keunggulan-keunggulan ini membuat *Python* menjadi salah satu bahasa pemrograman yang banyak dipilih oleh para pengguna.

6. PHP (*Hypertext Preprocessor*)



Gambar 2. 3 Logo PHP
(Sumber : https://seeklogo.com/vector-logo/108600/php)

PHP (*Hypertext Preprocessor*) adalah bahasa scripting yang berjalan di sisi *server* dan terintegrasi dengan HTML untuk menghasilkan halaman web dinamis. Sebagai *server-side* scripting, kode dan perintah PHP dieksekusi di server, lalu hasilnya dikirimkan ke *browser* dalam format HTML. Hal ini memastikan bahwa kode PHP tidak terlihat oleh pengguna, yang pada gilirannya meningkatkan tingkat keamanan halaman web.

Keunggulan PHP terletak pada kemampuannya untuk terhubung dengan berbagai sistem manajemen basis data (DBMS), yang memungkinkan pembuatan halaman web dinamis. PHP memiliki koneksi yang baik dengan sejumlah DBMS, seperti Oracle, Sybase, mSQL, MySQL, Microsoft SQL Server, Solid, PostgreSQL, Adabas, FilePro, Velocis, dBase, Unix dbm, dan juga semua *database* yang menggunakan antarmuka ODBC. Selain itu, PHP dapat terintegrasi dengan berbagai pustaka eksternal yang memungkinkan Anda untuk melakukan berbagai hal, mulai dari menghasilkan dokumen PDF hingga memparsing XML. PHP juga mendukung komunikasi dengan layanan lain melalui protokol IMAP, SNMP, NNTP, POP3, atau bahkan HTTP.

7. *Visual Studio Code*



Visual Studio Code

Gambar 2. 4 Logo *Visual Studio Code*
(Sumber : https://canonical.com/blog/visual-studio-code-launches-as-a-snap)

Visual Studio adalah *integrated development environment* (IDE) yang dikembangkan oleh *Microsoft* untuk memudahkan pengembang perangkat lunak dalam mengembangkan aplikasi pada platform *Microsoft. Visual Studio* dapat digunakan untuk membuat aplikasi *mobile*, web, *desktop*, dan cloud. Sedangkan, Visual Studio Code adalah editor kode sumber *multiplatform* yang dapat digunakan di sistem operasi Windows, Linux, dan Mac OSX. *Visual Studio Code* mendukung berbagai bahasa pemrograman, termasuk bahasa yang ada di *Visual Studio* 2015, ditambah dengan PHP, Node.js, dan lainnya.

Keunggulan utama dari *Visual Studio Code* adalah kemampuannya untuk diperluas. Pengguna dapat menambahkan ekstensi untuk mendukung bahasa baru, tema, dan bahkan alat *debugging*. Fitur integrasi dengan Git memungkinkan pengembang mengelola kode sumber dan kontrol versi langsung dari editor. Selain itu, *Visual Studio Code* dilengkapi dengan *debugger* untuk men-*debug* aplikasi secara langsung tanpa menggunakan alat eksternal. Pengembang juga dapat menjalankan perintah shell langsung dari terminal terintegrasi yang ada di *Visual Studio Code* (Faisal, 2017).

8. UML (*Unified Modeling Language*)

UML *(Unified Modeling Language)* bahasa pemodelan yang digunakan untuk merancang sistem atau perangkat lunak dengan pendekatan berorientasi objek. Tujuan utama dari UML adalah untuk menyediakan bahasa pemodelan yang tidak tergantung pada bahasa pemrograman atau proses rekayasa tertentu, menggabungkan praktik terbaik dalam pemodelan, serta menyediakan model yang dapat langsung digunakan. UML juga menawarkan bahasa pemodelan visual yang

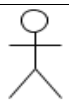
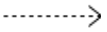
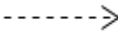
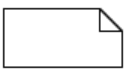
ekspresif, memudahkan pengembangan dan pertukaran model secara mudah dan dapat dipahami secara umum.

Adapun jenis jenis UML yaitu :

a. *Use case*

Use case berfungsi untuk mendeskripsikan sebuah interaksi antara satu atau lebih actor dengan sistem informasi yang akan dibangun, dan dapat menggambarkan fungsi apa saja yang ada pada sebuah system informasi (Julianto & Setiawan, 2019).

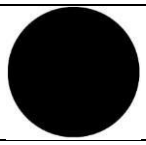
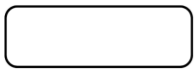
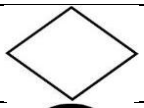
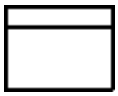
Tabel 2. 1 Simbol *Use Case Diagram*

GAMBAR	NAMA	KETERANGAN
	*Actor*	Menspesifikasikan himpuan peran yang pengguna mainkan ketika berinteraksi dengan *use case*.
	*Dependency*	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (*independent*) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri (*independent*).
	Generalization	Menspesifikasikan bahwa *use case* sumber secara eksplisit.
	*Include*	Menspesifikasikan bahwa *use case* target memperluas perilaku dari *use case* sumber pada suatu titik yang diberikan.
	Extend	Menunjukkan bahwa sebuah *Use Case* dapat menambahkan alur tambahan pada *Use Case* lain secara opsional.
	Association	Hubungan yang menghubungkan antara aktor dengan *Use Case* untuk menggambarkan interaksi di antara keduanya.
	*Use Case*	Deskripsi tentang urutan aksi atau interaksi antara aktor dengan sistem yang menghasilkan output tertentu bagi aktor
	*Note*	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi

b. *Activity diagram*

Activity diagram menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah system yang ada pada perangkat lunak.

Tabel 2. 2 Simbol *Activity diagram*

GAMBAR	NAMA	KETERANGAN
	*(initial Node) Status Awal*	Sebuah diagram aktivitas memiliki sebuah status awal.
	*Activity state*	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
	Control Flow	menjelaskan aliran kontrol dari satu tindakan ke tindakan lain.
	*Marge/Decision*	Percabangan dimana ada pilihan aktivitas yang lebih dari satu.
	*Final Node (Status Akhir)*	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir.
	Fork Node/Join Node	Pergabungan dimana leboh dari satu aktivitas lalu digabungkan jdi satu.
	*Swimlane*	Sebuah organisasi bisnis yang bertanggung terhadap aktivitas yang terjadi.

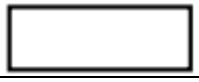
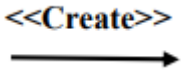
c. *Sequence diagram*

Sequence diagram merupakan pembentukkan utama dari system berorientasi objek yang mempresentasikan suatu class berserta dengan antribut dan operasinya (Haviluddin, 2011).

Tabel 2. 3 Simbol *Sequence Diagram*

GAMBAR	NAMA	KETERANGAN
	LifeLine	Menyatakan kehidupan suatu objek.

Lanjutan Tabel 2.3

GAMBAR	NAMA	KETERANGAN
	*Object*	Menyatakan objek yang berinteraksi pesan.
	*Activation*	Menyatakan objek dalam keadaan aktif dan berinteraksi, semua yang terhubung dengan waktu aktif ini adalah sebuah tahapan yang dilakukan didalmnya.
	Pesan tipe *Create*	Menyatakan suatu objek membuat objek yang lain, arah panah mengarah pada objek yang dibuat.

9. Flowchart

Flowchart adalah sebuah alat pemetaan yang digunakan untuk menunjukkan urutan tindakan dalam sebuah proses dengan cara yang mudah dibaca dan dipahami. Sebagai bahasa pemodelan untuk sistem atau perangkat lunak berorientasi objek, *flowchart* dikategorikan menjadi beberapa jenis berdasarkan fungsinya, proses yang terlibat, dan tingkat kepentingannya bagi pengguna.

Flowchart terbagi atas lima jenis, yaitu:

a. *Flowchart* Sistem *(System Flowchart*)

Flowchart Sistem adalah diagram yang menggambarkan alur kerja atau aktivitas yang sedang berlangsung dalam suatu sistem secara keseluruhan, serta menjelaskan urutan prosedur-prosedur yang ada di dalam sistem tersebut.

b. *Flowchart Paperwork / Flowchart* Dokumen (*Document Flowchart*)

Flowchart Dokumen (*Paperwork)* mengikuti alur data yang diproses dalam sistem. Tujuan utamanya adalah untuk melacak jalur formulir dan laporan sistem, dari satu bagian ke bagian lainnya, serta menjelaskan bagaimana formulir dan laporan tersebut diproses, dicatat, dan disimpan.

c. *Flowchart* Skematik (*Schematic Flowchart*)

Flowchart Skematik berfungsi sebagai alat komunikasi yang membantu analis sistem berinteraksi dengan orang yang tidak terbiasa dengan simbol-simbol Flowchart yang biasa digunakan.

d. *Flowchart* Program (*Program Flowchart)*

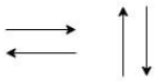
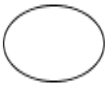
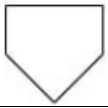
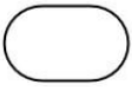
Flowchart Program memberikan penjelasan rinci tentang bagaimana setiap langkah dalam program atau prosedur dijalankan. Flowchart ini menggambarkan urutan yang tepat dari setiap langkah yang terjadi dalam program.

e. *Flowchart* Proses (*Process Flowchart*)

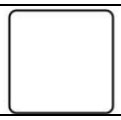
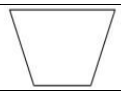
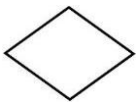
Flowchart Proses adalah teknik yang digunakan dalam rekayasa industri untuk memecah dan menganalisis tahapan-tahapan dalam suatu prosedur atau sistem sehingga memudahkan identifikasi potensi masalah dan peluang perbaikan dalam proses atau sistem tersebut.

Berikut adalah simbol-simbol yang sering digunakan dalam proses pembuatan *flowchart*

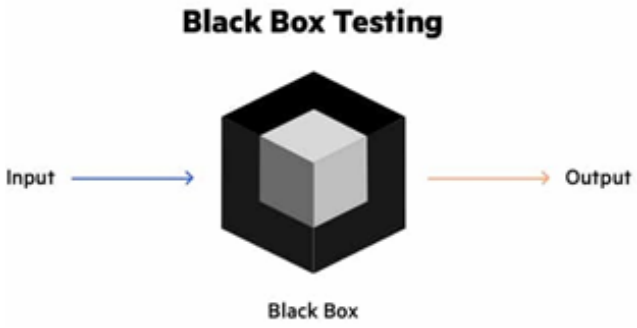
Tabel 2. 4 Simbol *Flowchart*

GAMBAR	NAMA	KETERANGAN
	*Flow*	Simbol ini menjelaskan penggabungan antara satu simbol dengan lainnya.
	*On page reference*	simbol ini menyatakan keluar masuknya atau proses penyambungan di dalam lembar kerja yang sama
	*Off page reference,*	Simbol ini menyatakan keluar masuknya proses dalam lembar kerja yang berbeda
	*Terminator*	Berfungsi sebagai penanda sebuah awal ataupun akhir dari suatu program

Lanjutan Tabel 2.4

GAMBAR	NAMA	KETERANGAN
	*Process*	Simbol ini mewakili sebuah proses yang tengah dijalankan komputer
	*Input-output*	Simbol ini memiliki arti proses input maupun output tanpa memperhatikan device atau perangkat yang digunakan
	*Manual operation*	Menjelaskan bahwa proses tersebut tidak melalui komputer.
	*Document*	Fungsi dari simbol ini, yakni menjelaskan bahwasanya input berasal dari dokumen penyimpanan fisik, serta output yang perlu untuk dicetak
	*Decision*	Fungsinya yaitu menjelaskan suatu kondisi tertentu yang dapat memunculkan dua jawaban yaitu ya dan juga tidak
	*Preparation*	Fungsinya untuk menunjukkan penyediaan penyimpanan, serta pengolahan yang memunculkan niat awal

10. *Black Box* Testing



Gambar 2. 5 *Black Box* Testing
(Sumber : https://www.imperva.com/learn/application-security/black-box-testing)

Black box Testing merupakan metode yang digunakan untuk mengevaluasi sebuah perangkat lunak tanpa memerlukan pemahaman mendalam tentang struktur internalnya. Pengujian ini dilakukan dengan cara mengisi data pada setiap form

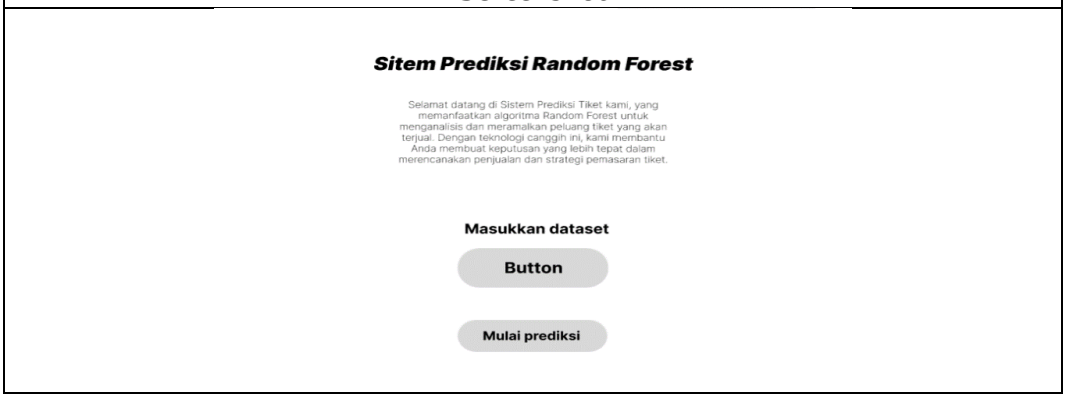
yang ada pada program yang telah dibuat. Pengujian semacam ini penting untuk memastikan bahwa aplikasi berjalan sebagaimana mestinya, sesuai dengan kebutuhan yang telah ditetapkan oleh perusahaan (Wahyudi, dkk. 2016).

Black box testing bertujuan untuk menilai apakah sistem dapat menghasilkan output yang sesuai dengan input yang diberikan. Jika output yang dihasilkan sesuai dengan yang diharapkan, maka sistem dianggap berfungsi dengan baik. Sebaliknya, jika terdapat kegagalan dalam pelaksanaan prosedur, maka perbaikan diperlukan. Metode ini juga sering disebut sebagai *behavioral* testing, *closed-box* testing, atau *specification-based* testing.

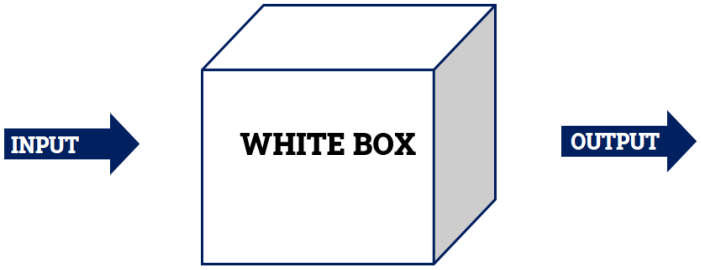
Dalam penerapannya, *Black box* testing memiliki berbagai kelebihan dan kekurangan. Salah satu kelebihannya adalah kemampuannya untuk menemukan aspek-aspek kebutuhan yang belum tercapai sesuai dengan spesifikasi yang ditetapkan pada tahap pengembangan perangkat lunak (Praniffa, dkk, 2023).

Contoh : pengujian fitur login

Tabel 2. 5 Tabel Contoh Pengujian Fitur Login *Black Box*

Test Faktor	Hasil	Keterangan
User pertama kali mengakses website	✓	Sukses, tampilkan halaman utama
Screenshot		
		

11. *White Box* Testing



Gambar 2. 6 *White Box* Testing
(Sumber : https://www.softwaretestingmaterial.com/white-box-testing)

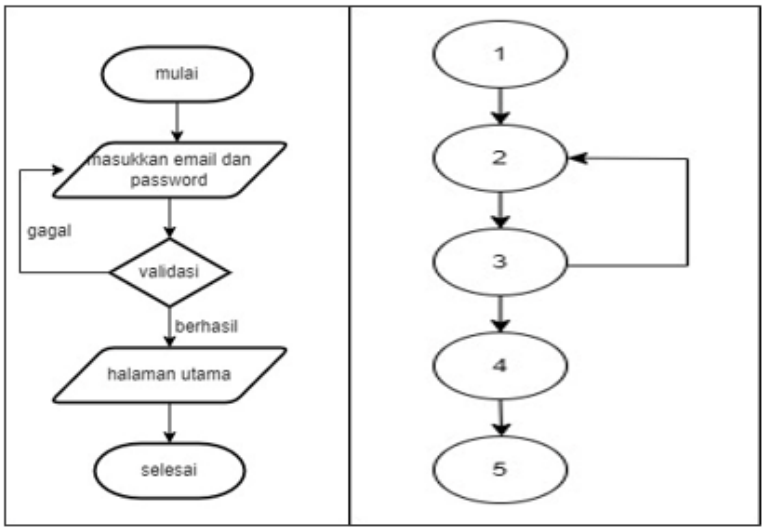
White box testing adalah metode pengujian perangkat lunak yang fokus pada komponen internal, seperti struktur kode, desain, dan cara kerja sistem dari awal hingga akhir. Tujuan utama dari pengujian ini adalah untuk memeriksa alur data yang mengalir dari input ke output, mengevaluasi fungsionalitas, dan memastikan perangkat lunak aman digunakan. Penguji dalam *white box* testing memiliki akses penuh untuk menganalisis kode yang digunakan dalam proses pengembangan, sehingga metode ini juga dikenal dengan sebutan *clear box* testing, *open box testing, code-based* testing, atau *structural* testing.

White box testing difokuskan pada pemeriksaan logika internal dan struktur kode perangkat lunak. Uji kasus dirancang untuk menguji fungsi-fungsi internal perangkat lunak dari perspektif pengembang. Teknik ini diterapkan ketika pengembang memiliki pemahaman yang mendalam mengenai struktur program. Dengan menggunakan pendekatan ini, setiap cabang dan keputusan dalam program dapat diuji secara rinci (Wintana & Nurhadi, 2022).

Kelebihan dari *White box* testing meliputi kemampuannya untuk mengidentifikasi dan menghapus kode yang tidak perlu atau tersembunyi. Metode

ini memungkinkan pengujian secara menyeluruh karena setiap komponen dari struktur atau logika program diperiksa secara detail. Selain itu, *White box* testing juga membantu dalam mengoptimalkan kode dan bisa diterapkan bahkan ketika antarmuka pengguna grafis (GUI) masih dalam tahap pengembangan (Fedianto, dkk, 2024).

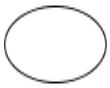
Contoh Pengujian Fungsi Login:



Gambar 2. 7 Contoh Pengujian *White Box*

Flowgraph terdiri dari beberapa elemen penting yang mewakili logika program, di antaranya:

Tabel 2. 6 Simbol *Flowgraph*

GAMBAR	NAMA	KETERANGAN
	*Node*	Simbol yang menggambarkan langkah atau keputusan dalam alur program.
	*Edge*	Menghubungkan antar node, menunjukkan alur atau urutan eksekusi dari satu langkah ke langkah lainnya.

12. *Mean Absolute Percentage Error* (MAPE)

Dalam proses prediksi, sering kali diperlukan suatu metode untuk mengukur tingkat kesalahan antara nilai aktual dan prediksi. Salah satu metode yang digunakan untuk mengukur kesalahan tersebut adalah *Mean Absolute Percentage Error* (MAPE). MAPE adalah indikator kesalahan prediksi yang digunakan untuk menilai tingkat akurasi suatu model, di mana semakin kecil nilai MAPE, semakin baik tingkat akurasi model tersebut.

Mean Absolute Percentage Error (MAPE) dihitung dengan membagi kesalahan absolut pada setiap periode dengan nilai observasi nyata pada periode tersebut, lalu menghitung rata-rata dari kesalahan persentase absolut yang diperoleh. Metode ini sangat berguna ketika ukuran atau skala variabel yang diprediksi memiliki peran penting dalam menilai ketepatan prediksi. MAPE memberikan gambaran seberapa besar selisih antara hasil ramalan dengan nilai aktual (Ginantra & Anandita, 2019). Semakin kecil nilai MAPE yang dihasilkan berarti nilai prediksi semakin mendekati nilai sebenarnya. Untuk menghitung MAPE menggunakan persamaan

$$MAPE = \frac{1}{\mathrm{N}} \sum_{i=1}^{n} \left| \frac{yi - \hat{y}i}{yi} \right| \times 100 \quad \text{.....(2)}$$

Dimana:

yi adalah nilai data asli (actual values) untuk titik data ke-i.

$\hat{y}i$ adalah nilai prediksi (predicted values) untuk titik data ke-i.

N adalah jumlah data (jumlah titik data yang digunakan).

Hasil MAPE berupa persentase, yang menunjukkan kesalahan rata-rata dalam prediksi dibandingkan dengan data asli.

13. *Usability Testing*

Usability Testing adalah metode yang digunakan untuk menilai sejauh mana suatu perangkat lunak, situs web, atau produk dapat digunakan dengan efektif. Kemudahan akses ke situs web sangat penting, karena jika sebuah situs web sulit diakses atau digunakan, maka situs tersebut menjadi tidak layak untuk digunakan, karena tidak dapat memberikan informasi yang dibutuhkan oleh penggunanya.

Usability Testing melibatkan penentuan evaluator, pengumpulan data melalui survei dan kuisioner, pembuatan tugas-tugas usability, serta penyusunan skrip untuk pengujian usability. Pelaksanaan usability testing ini dilakukan dengan mempertimbangkan aspek *learnability, efficiency, memorability, errors*, dan *satisfaction.*

Usability dapat diukur berdasarkan indikator-indikator berikut

a. Kemudahan (*learnability*) didefinisikan seberapa cepat pengguna mahir dalam menggunakan sistem serta kemudahan dalam penggunaan menjalankan suatu fungsi serta apa yang pengguna inginkan dapat meraka dapatkan.
b. Efisiensi (*efficiency*) didefenisikan sebagai sumber daya yang dikeluarkan guna mencapai ketepatan dan kelengkapan tujuan.
c. Mudah diingat (*memorability*) didefinisikan bagaimana kemapuan pengguna mempertahankan pengetahuannya setelah jangka waktu tertentu, kemampuan mengngat didapatkan dari peletakkan menu yang selalu tetap.
d. Kesalahan dan keamanan (*errors)* didefinisikan berapa banyak kesalahan-kesalahan apa saja yang dibuat pengguna, kesalahan yang dibuat pengguna

mencangkup ketidaksesuaian apa yang pengguna pikirkan dengan apa yang sebenarnya disajikan oleh sistem.

e. Kepuasan (*satisfaction*) didefinisikan kebebasan dari ketidaknyamanan, dan sikap positif terhadap penggunaan produk atau ukuran subjektif

B. Penelitian Terdahulu

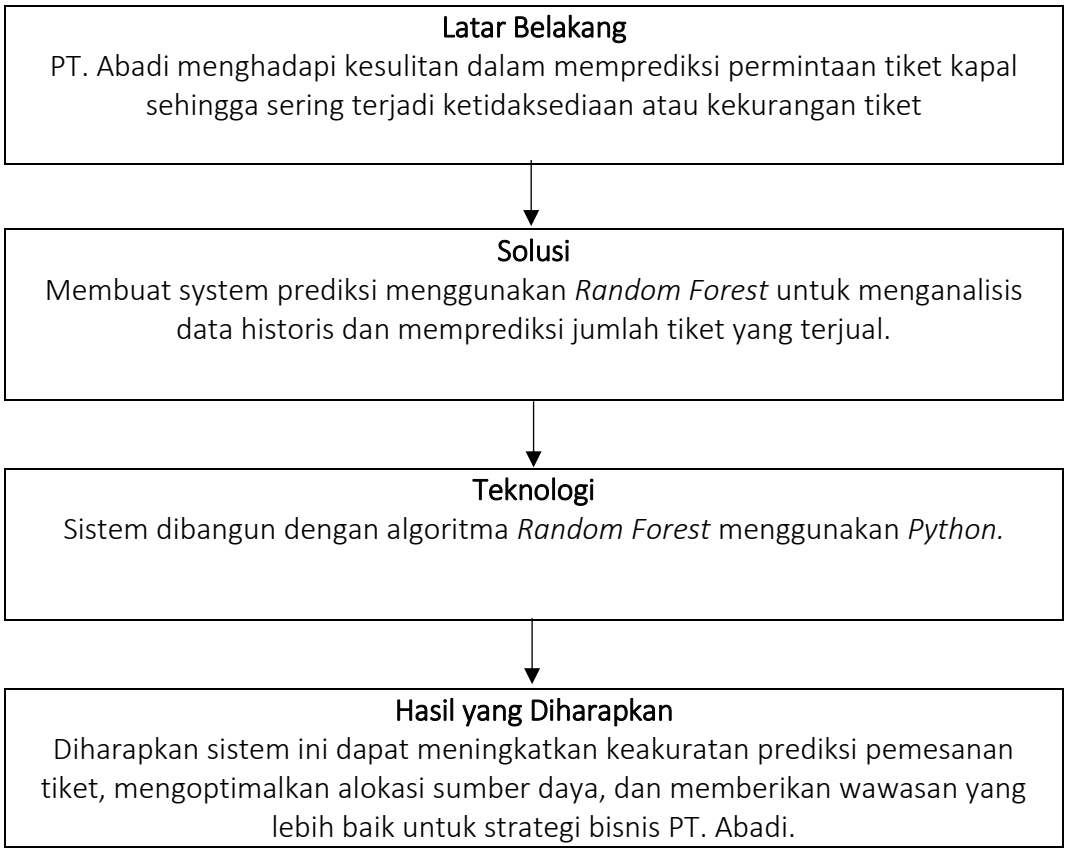
Penelitian sebelumnya sangat penting untuk digunakan sebagai acuan, pertimbangan, dan data pendukung dalam melaksanakan penelitian. Berikut ini adalah beberapa studi terdahulu yang telah dilakukan:

1. Muhammad Syahrul Efendi, Sarwido, dan Akhmad Khanif Zyen (2024) dalam jurnal "**PENERAPAN ALGORITMA RANDOM FOREST UNTUK PREDIKSI PENJUALAN DAN SISTEM PERSEDIAAN PRODUK**" meneliti penggunaan algoritma Random Forest untuk memprediksi penjualan dan mengelola persediaan produk Bolen Crispy di Desa Pekalongan. Penelitian ini bertujuan untuk mengoptimalkan pengelolaan persediaan dengan mengurangi kesalahan prediksi yang sering menyebabkan kelebihan atau kekurangan stok, yang berdampak pada peningkatan biaya operasional dan menurunnya kepuasan pelanggan. Hasil penelitian menunjukkan bahwa penerapan algoritma Random Forest dapat meningkatkan akurasi prediksi penjualan hingga 85%, lebih baik dibandingkan dengan metode konvensional. Dengan menggunakan data penjualan historis dan pembagian data menjadi 80% untuk pelatihan dan 20% untuk pengujian, algoritma ini mampu memberikan prediksi yang lebih

akurat, sehingga membantu pengusaha dalam merencanakan stok dengan lebih efisien.

2. Siti Saadah dan Haifa Salsabila (2021) dalam jurnal "**PREDIKSI HARGA BITCOIN MENGGUNAKAN METODE RANDOM FOREST (STUDI KASUS: DATA ACAK PADA AWAL MASA PANDEMIC COVID-19**)" meneliti penggunaan metode Random Forest untuk memprediksi harga Bitcoin pada awal pandemi COVID-19. Penelitian ini bertujuan untuk menganalisis fluktuasi harga Bitcoin yang volatil dan memberikan solusi prediksi yang lebih akurat bagi investor. Hasil penelitian menunjukkan bahwa algoritma Random Forest Regression berhasil mencapai akurasi 98,50% dan MAPE 1,50%, membuktikan bahwa Random Forest efektif dalam memprediksi harga Bitcoin pada data yang acak dan fluktuatif.
3. Othman Alydrus, Syarifah Putri Agustini Alkadri, dan Istikoma (2024) dalam jurnal "**SISTEM PREDIKSI JUMLAH PENUMPANG KAPAL KM DHARMA FERRY 2 DI KETAPANG MENGGUNAKAN METODE DOUBLE EXPONENTIAL SMOOTHING**" mengembangkan sistem untuk memprediksi jumlah penumpang kapal KM Dharma Ferry 2 di Pelabuhan Sukabangun, Ketapang. Penelitian ini bertujuan mengatasi fluktuasi jumlah penumpang, terutama saat arus mudik Lebaran. Dengan menggunakan metode Double Exponential Smoothing, hasil penelitian menunjukkan bahwa nilai alpha terbaik untuk prediksi adalah 0,9, yang menghasilkan prediksi jumlah penumpang sebanyak 1392 orang untuk bulan berikutnya dengan MAPE 6,59%.

C. Kerangka Berfikir



Gambar 2. 8 Kerangka berpikir

BAB III

METODE PENELITIAN

A. Jenis Penelitian

Penelitian ini termasuk dalam kategori penelitian eksperimental yang bertujuan untuk menguji hubungan antara berbagai variabel melalui analisis statistik, dalam hal ini dengan fokus pada penerapan sistem prediksi pemesanan tiket kapal menggunakan algoritma *random forest.* Tujuan dari penelitian ini adalah untuk mengembangkan dan menguji model prediksi yang dapat membantu PT. Abadi dalam memproyeksikan jumlah pemesanan tiket kapal berdasarkan data historis yang telah tersedia.

Penelitian eksperimental dipilih karena penelitian ini bertujuan untuk mengevaluasi efektivitas penerapan model prediksi berbasis *random forest* dalam konteks pemesanan tiket kapal. Dengan menggunakan dataset yang ada, peneliti akan membangun model prediksi menggunakan algoritma *random forest* dan menguji sejauh mana model tersebut mampu menghasilkan prediksi yang akurat terkait jumlah pemesanan tiket.

B. Lokasi dan Waktu

Penelitian dilaksanakan di PT.Abadi Polacitra mandiri di jalan andi cammi no.108, Parepare, Sulawesi Selatan. Penelitian dilakukan selama 3 (tiga) bulan yang dimulai pada bulan Mei sampai dengan bulan juli 2025, rincian pelaksanaan penelitian dapat dilihat pada Tabel.

Tabel 3. 1 Lokasi dan Waktu

No.	Uraian Kegiatan	Bulan		
		Mei	Juni	Juli
1	Perancangan konsep sistem prediksi			
2	Pengumpulan Data			
3	Pengembangan Sistem Prediksi			
4	Pengujian dan Evaluasi			

C. Alat dan Bahan

Penelitian ini membutuhkan beberapa alat dan bahan untuk mendukung berjalannya perancangan dan implementasi aplikasi. Adapun alat dan bahan yang dibutuhkan terdiri dari

1. Alat

Alat yang digunakan berupa perangkat keras *(hardware*) dan perangkat lunak *(software).*

a. Perangkat Keras *(Hardware)*

1) Personal Computer (PC) atau Laptop ADVAN WORKPLUS
    a) AMD Ryzen 5 6600H *with Radeon Graphics* (12 CPU), ~3.3GHz
    b) Kapasitas *RAM 16GB*
    c) Perangkat *mouse* dan *keyboard* standar
2) Printer Dokumen untuk mencetak laporan

b. Perangkat Lunak *(Software)*

1) Sistem Operasi *Windows* 11 *Home*
2) Text Editor Visual studio Code
3) Bahasa Pemrograman *Pyhton* dan PHP

4) Web *Browser Brave, Crhome, Microsoft Edge*

2. Bahan

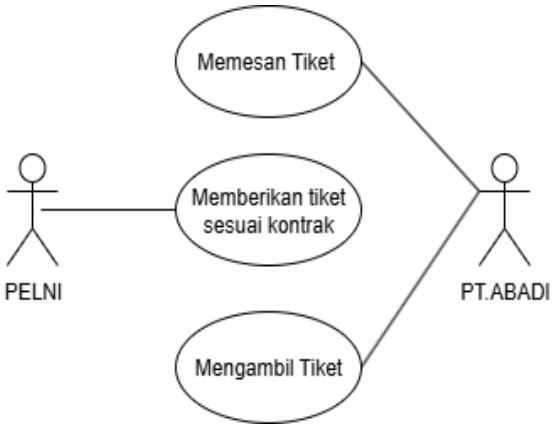
Dalam penelitian bahan yang digunakan adalah data pemesanan tiket yang diperoleh di PT. Abadi Polacitra Mandiri

D. Rancangan Sistem

Rancangan sistem adalah proses mendefinisikan arsitektur, komponen, modul, antarmuka, dan data dari sebuah sistem guna memenuhi kebutuhan tertentu. Proses perancangan ini mencakup berbagai aspek yang harus diperhatikan agar sistem yang dibangun dapat berjalan dengan efektif dan efisien. Salah satu aspek yang sangat penting dalam perancangan sistem adalah pemahaman yang mendalam terhadap kebutuhan pengguna, sehingga sistem yang dikembangkan dapat memberikan solusi yang relevan dan bermanfaat.

1. *Usecase Diagram*

a. Rancangan sistem yang berjalan



Gambar 3. 1 sistem yang berjalan

Berikut adalah tabel *use case* berdasarkan gambar diagram sistem yang berjalan :

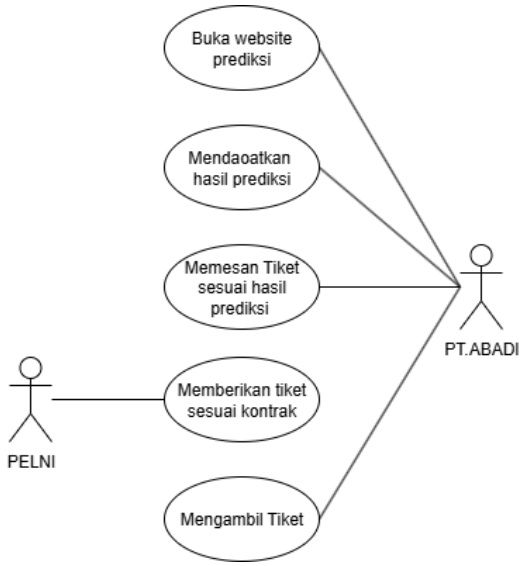
Tabel 3. 2 Tabel *Use Case* Sistem Berjalan

Nama *Use case*	Deskripsi
Memasan Tiket	PT.ABADI memulai dengan memesan tiket. Proses ini bisa dilakukan melalui sistem yang disediakan oleh PELNI atau penyedia lainnya.
Memberikan Tiket Sesuai Kontrak	Setelah tiket dipesan, PELNI bertanggung jawab untuk memberikan tiket yang sesuai dengan kontrak yang telah disepakati.
Mengambil Tiket	PT.ABADI kemudian mengambil tiket yang telah diberikan oleh PELNI sesuai dengan kontrak yang berlaku.

Sistem yang sedang berjalan memiliki kelemahan dalam hal memprediksi permintaan tiket secara efektif, menyebabkan tiket sering kali tidak terjual habis atau kekurangan tiket.

b. Rancangan sistem yang diusulkan

Untuk mengatasi keterbatasan sistem yang berjalan, diusulkan sistem prediksi pemesana tiket kapal menggunakan random forest untuk membantu PT.ABADI dalam menentukan jumlah tiket yang akan digunakan



Gambar 3. 2 Sistem yang diusulkan

Berikut adalah tabel *use case* berdasarkan gambar diagram sistem yang diusulkan :

Tabel 3. 3 Tabel *Use Case* Sistem yang diusulkan

Nama *Use case*	**Deskripsi**
Buka website prediksi	PT.ABADI sebagai pengguna membuka website prediksi untuk mendapatkan informasi mengenai tiket yang akan dipesan..
Mendapatkan hasil prediksi	Setelah membuka website, PT.ABADI akan mendapatkan hasil prediksi yang dapat berupa rekomendasi tentang keputusan pemesanan berdasarkan analisis data yang ada.
Memesan tiket sesuai hasil prediksi	Berdasarkan hasil prediksi yang diperoleh dari website, PT.ABADI kemudian memutuskan untuk memesan tiket yang sesuai dengan informasi yang telah diberikan.
Memberikan tiket sesuai kontrak	Setelah tiket dipesan, PELNI bertanggung jawab untuk memberikan tiket yang sesuai dengan kontrak yang telah disepakati
Mengambil Tiket	PT.ABADI kemudian mengambil tiket yang telah diberikan oleh PELNI sesuai dengan kontrak yang berlaku.

2. Rancangan diagram alur sistem

Perancangan sistem ini menggunakan diagram alur (*flowchart*) yang menggambarkan proses dan tahapan dalam perangkat lunak. Flowchart membantu memvisualisasikan alur kerja sistem dengan cara yang lebih jelas dan terstruktur. Selain itu, *flowchart* juga berfungsi untuk menyederhanakan dan memecah proses yang rumit menjadi bagian-bagian yang lebih kecil dan mudah diatur. Dengan cara ini, alur kerja yang kompleks dalam sistem dapat diuraikan menjadi langkah-langkah yang lebih spesifik dan terperinci.

Berikut adalah diagram alur (*flowchart)* yang menggambarkan langkah-langkah dan proses dalam sistem yang sedang dirancang.



Gambar 3. 3 Diagram Alur Sistem

Diagram alur ini menggambarkan langkah-langkah dalam proses prediksi yang menggunakan metode pohon keputusan. Proses dimulai dengan penginputan data historis selama empat tahun ke dalam sistem. Kemudian, data tersebut diproses melalui beberapa pohon keputusan, di mana setiap pohon menghasilkan prediksi berdasarkan aturan yang telah ditetapkan. Setiap pohon keputusan, mulai dari yang pertama hingga yang terakhir, memberikan hasil prediksi masing-masing. Hasil prediksi dari semua pohon keputusan kemudian dihitung rata-ratanya untuk menghasilkan prediksi akhir. Dengan cara ini, penggunaan beberapa pohon

keputusan meningkatkan akurasi dan keandalan prediksi. Setelah prediksi akhir dihitung, kemudian hasil *Random Forest* disesuaikan dengan pola musiman, proses pun selesai dan hasilnya siap digunakan sesuai kebutuhan.

3. Desain *Interface*

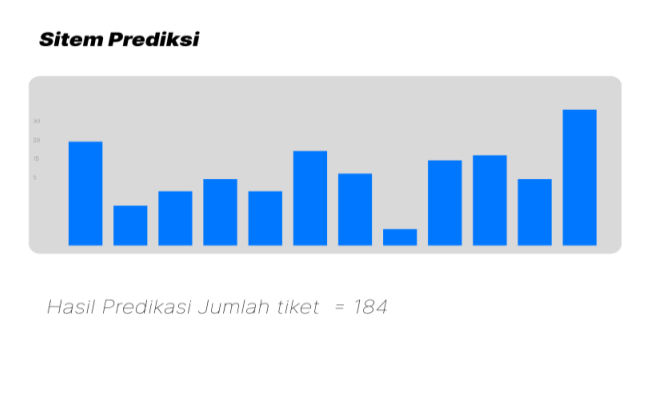
Perancangan tampilan desain antarmuka sistem ini dilakukan dengan menggunakan pendekatan desain antarmuka pengguna yang fokus pada pengalaman pengguna (*user experience*). Salah satu elemen yang digunakan dalam perancangan tampilan ini adalah *Wireframe. Wireframe* berfungsi untuk menggambarkan layout dan struktur tampilan sistem, dengan menyoroti elemen-elemen antarmuka yang digunakan pengguna untuk berinteraksi dengan sistem. Desain *interface* perancangan sistem ditunjukkan pada gambar berikut ini



Gambar 3. 4 Tampilan menu utama

Desain halaman utama memperhatikan kemudahan navigasi, memastikan bahwa setiap elemen interaktif mudah ditemukan dan digunakan, sehingga menciptakan pengalaman pengguna yang efisien dan lancar, serta memastikan bahwa pengguna dapat segera memulai proses tanpa kebingungan. Pada tampilan utama ini, terdapat beberapa opsi utama yang dapat dipilih sesuai dengan kebutuhan pengguna. Opsi pertama adalah Input Dataset, yang memungkinkan pengguna

untuk mengunggah dataset yang berisi data historis penjualan tiket kapal untuk dianalisis menggunakan algoritma Random Forest. Pengguna dapat memilih file dataset dalam format yang diterima, Excel.Selanjutnya, terdapat Hasil Prediksi, yang memberikan hasil prediksi tiket kapal berdasarkan data yang telah diunggah.



Gambar 3. 5 Tampilan menu hasil

Pada menu hasil prediksi, pengguna akan melihat diagram yang menampilkan hasil prediksi total jumlah tiket yang akan terjual berdasarkan analisis algoritma *Random Forest* dengan cara yang jelas dan mudah dipahami. Dibagian ini juga terdapat fitur untuk melihat data historis yang telah di upload. Desain antarmuka ini dirancang untuk memberikan pengguna informasi yang terstruktur dan memungkinkan mereka untuk dengan mudah membandingkan data *historis* dengan prediksi yang dihasilkan.

E. Teknik Pengumpulan Data

1. Pengumpulan Data

   Pada tahap awal penelitian, dilakukan pengumpulan data historis pemesanan tiket kapal. Data yang dikumpulkan mencakup volume pemesanan per bulan dan kebijakan perusahaan. Data ini sangat penting untuk membangun

model prediksi yang akurat. Tujuan utamanya adalah memastikan sistem yang dikembangkan sesuai dengan kebutuhan operasional perusahaan.

2. Pemilihan Fitur

   Proses pemilihan fitur merupakan tahap penting dalam pembangunan model prediksi. Tahap ini mengidentifikasi variabel yang paling berpengaruh terhadap pemesanan tiket kapal. Fitur yang dipilih meliputi tanggal pemesanan, jumlah pemesanan, nama kapal, serta faktor eksternal seperti musim liburan. Metode yang digunakan bisa berupa analisis korelasi, teknik statistik, atau algoritma *machine learning* seperti *decision trees*. Dengan pemilihan fitur yang tepat, performa model meningkat dan kompleksitas dapat dikurangi.

3. Implementasi Model *Random Forest*

   Pada tahap ini, algoritma *Random Forest* diterapkan untuk membangun model prediksi. *Random Forest* cocok untuk prediksi dengan data yang banyak variabel dan hubungan kompleks antar fitur. Data dibagi menjadi dua set: data latih (*training set*) untuk melatih model, dan data uji (*test set*) untuk menguji akurasi. Dengan menggunakan Random Forest, diharapkan model dapat memprediksi permintaan tiket kapal secara tepat.

4. Pengujian dan Uji Coba

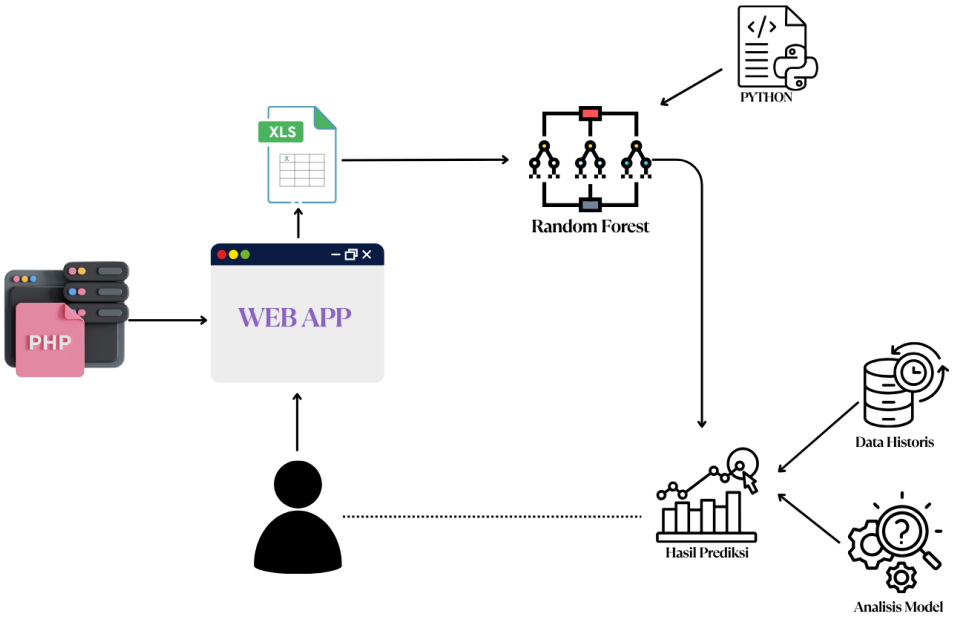
   Tahap uji coba bertujuan untuk memastikan setiap elemen di situs berfungsi dengan baik.Fitur yang diuji meliputi formulir input, tombol interaktif, navigasi, dan tampilan responsif di berbagai perangkat dan browser.Pengujian melibatkan pengguna untuk menilai apakah interaksi dengan situs berjalan lancar.Hasil uji coba akan mengidentifikasi apakah fitur-fitur berfungsi dengan baik atau perlu perbaikan.

BAB IV

PEMBAHASAN DAN HASIL

A. Analisis Sistem Yang Diusulkan

1. Alur Rancangan Sistem



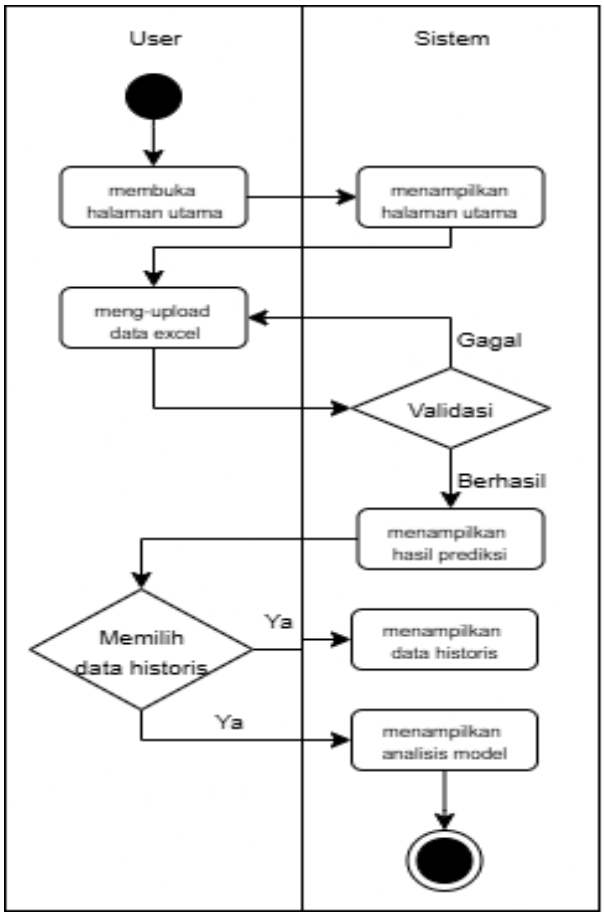
Gambar 4. 1 Alur Rancangan Sistem

Gambaran alur sistem ini menunjukkan bagaimana implementasi sistem dapat diterapkan dalam website untuk meramalkan pemesanan tiket kapal. Penjabaran lebih lanjut mengenai setiap bagian sistem akan disampaikan dalam tabel berikut, yang mencakup peran, fungsi, serta cara kerja sistem dalam menghasilkan prediksi :

Tabel 4. 1 Daftar Kebutuhan Sistem

Komponen	Keterangan
User	Pengguna utama sistem
Web App	Begitu aplikasi dijalankan, user akan melihat tampilan utama aplikasi, yang menjadi layar pertama yang muncul setelah masuk.
PHP	Bahasa pemrograman yang digunakan untuk membuat tampilan pada website
Upload data	User meng-upload data dari excel yang diperlukan, yang kemudian akan diproses oleh sistem.
Random Forest	Machine learning yang digunakan oleh sistem untuk mendapatkan hasil prediksi
Python	Bahasa pemrograman yang digunakan untuk meng-oprasikan random forest
Hasil Prediksi	Menampilakan hasil prediksi dalam bentuk grafik per-bulan
Data Historis	Menampilakan data historis excel yang di upload sebelumnya
Analisis Model	Menampilkan cara random forest menentukan hasil prediksi

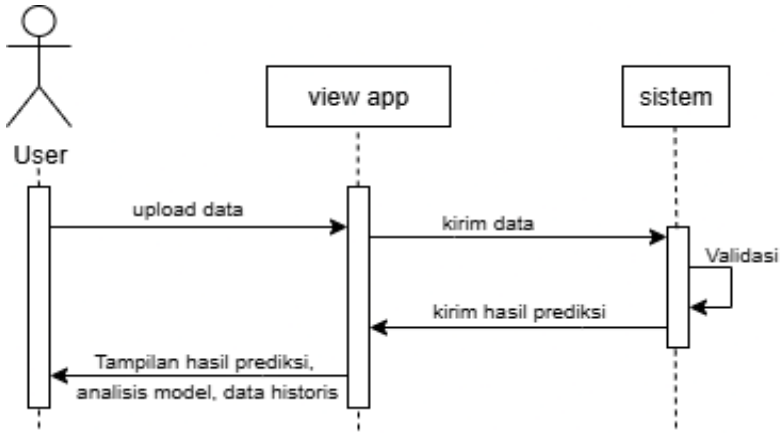
2. Activity Diagram



Gambar 4. 2 Activity Diagram Prediksi

User memulai dengan membuka halaman utama aplikasi, yang kemudian ditampilkan oleh sistem. Selanjutnya, pengguna meng-upload file *Excel* yang berisi data untuk dianalisis. Sistem akan memvalidasi data yang di-upload tersebut. Jika validasi berhasil, hasil prediksi yang berasal dari data yang diproses akan ditampilkan oleh sistem. Namun, jika validasi gagal, sistem akan memberikan status kegagalan dan meminta pengguna untuk memeriksa kembali data yang di-upload. Pada halaman prediksi, sistem juga menunjukkan analisis yang meliputi model *Random Forest* yang digunakan dan data historis yang mencakup pemesanan selama empat tahun yang telah di-upload sebelumnya.

3. Sequence Diagram



Gambar 4. 3 Sequence Diagram Prediksi

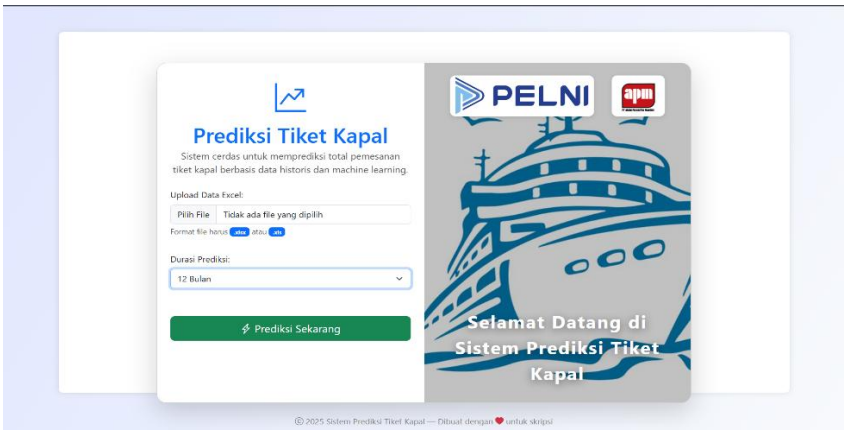
Diagram sequence ini menunjukkan interaksi antara tiga pihak: *User*, *View App*, dan Sistem. Proses dimulai ketika User mengunggah data ke dalam aplikasi (*View App*). Aplikasi kemudian menerima data tersebut dan meneruskannya ke Sistem untuk diproses lebih lanjut. Setelah menerima data, Sistem melakukan proses validasi guna memastikan bahwa data tersebut sesuai dan layak untuk dianalisis. Setelah validasi selesai, Sistem mengirimkan hasil prediksi yang telah

diproses kembali ke *View App*. Terakhir, *View App* menyajikan hasil prediksi, analisis model, dan data historis kepada User sebagai output dari seluruh rangkaian proses.

B. Implementasi Interface

1. Halaman Utama

Halaman utama dari aplikasi Sistem Prediksi Tiket Kapal menampilkan fitur inti berupa form unggah data, pilihan durasi prediksi, dan tombol aksi untuk menjalankan proses analisis. Pengguna dapat langsung mengunggah file Excel berisi data historis, menentukan jangka waktu prediksi, dan memulai proses dengan satu klik. Sistem ini dirancang untuk memudahkan prediksi jumlah tiket kapal yang akan dipesan dengan menggunakan pendekatan *machine learning* berbasis data historis. Visual kapal dan logo pelayaran yang ditampilkan di bagian kanan halaman memperkuat identitas dan konteks penggunaan aplikasi.

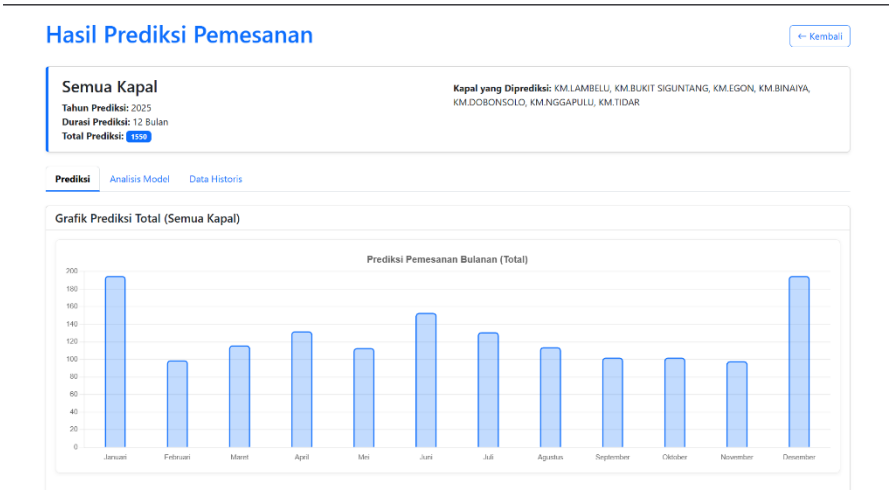


Gambar 4. 4 Tampilan Halaman Utama

2. Halaman Predikasi

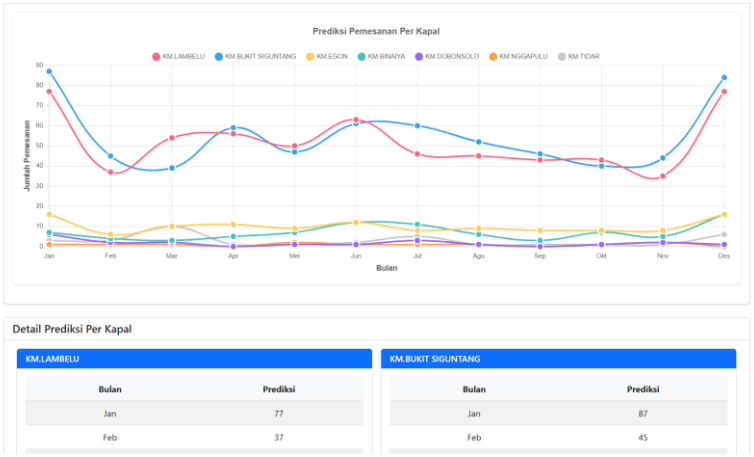
Tampilan ini merupakan halaman hasil dari aplikasi Sistem Prediksi Tiket Kapal, yang berfungsi untuk menyajikan prediksi pemesanan tiket selama tahun

2025 dalam rentang waktu 12 bulan. Judul "Hasil Prediksi Pemesanan" di bagian atas halaman menunjukkan bahwa pengguna sedang berada di tahap akhir dari proses analisis. Di bawah judul tersebut, terdapat informasi yang merangkum tahun prediksi, durasi estimasi, jumlah total hasil prediksi, dan juga tabel detail yang menampilkan data hasil prediksi secara lebih terperinci.



Gambar 4. 5 Tampilan Halaman Hasil Prediksi

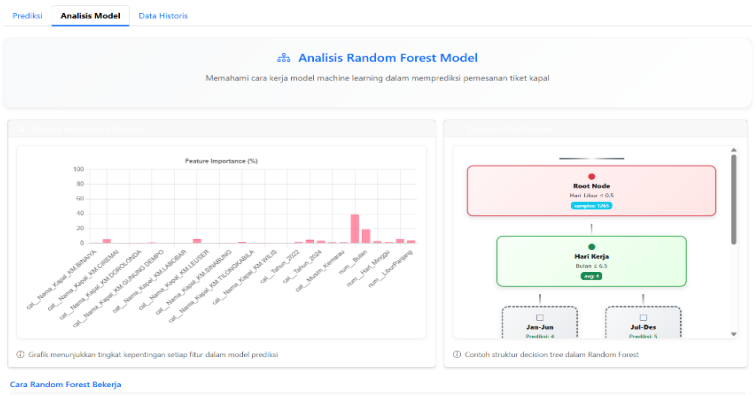
Di bagian bawahg grafik di atas terdapat grafik baru yang menggambarkan tren pemesanan untuk beberapa kapal yang berbeda, dengan garis-garis yang menunjukkan jumlah pemesanan yang diprediksi untuk setiap bulan. Setiap kapal diberi warna yang berbeda, seperti KM. LAMBELU yang berwarna merah dan KM. BUKIT SIGUNTANG yang berwarna biru, yang membantu membedakan prediksi masing-masing kapal. Di bawah grafik, terdapat tabel yang memberikan detail prediksi per kapal untuk bulan Januari dan Februari, seperti 77 pemesanan untuk KM. LAMBELU pada bulan Januari dan 87 untuk KM. BUKIT SIGUNTANG. Dengan informasi ini, pengguna dapat melihat gambaran yang jelas mengenai prediksi pemesanan tiket sepanjang tahun dan dapat membuat keputusan berdasarkan data yang tersedia.



Gambar 4. 6 Tampilan Hasil Prediksi Per-Kapal

3. Tab Analisis Model

Halaman ini dibuat untuk memberikan pemahaman yang lebih terbuka dan informatif kepada pengguna tentang bagaimana model prediksi bekerja dalam sistem. Melalui visualisasi feature importance dan struktur decision tree, pengguna tidak hanya melihat hasil akhir, tetapi juga dapat menelusuri proses logis yang digunakan model untuk menghasilkan prediksi tersebut.

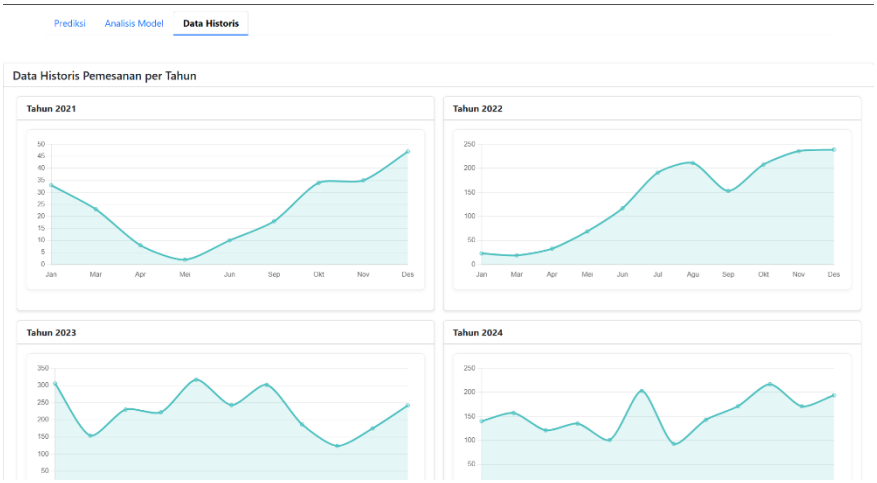


Gambar 4. 7 Tampilan Analisis Model

4. Tab Data Historis

Halaman ini dirancang untuk memudahkan visualisasi dan analisis data pemesanan tahunan, melalui grafik yang jelas dan mudah dipahami, sehingga

pengguna dapat dengan cepat melihat tren pemesanan sepanjang tahun. Selain grafik tersebut, terdapat juga tabel data historis yang menyajikan detail pemesanan per bulan, yang memberi kesempatan bagi pengguna untuk mengakses data numerik yang mendasari grafik untuk pemahaman yang lebih mendalam.



Gambar 4. 8 Tampilan Data Historis

C. Proses Prediksi

1. Load Data dari Excel

Proses ini dimulai dengan tujuan untuk mengumpulkan dan menyiapkan data historis sebagai dasar untuk melatih model machine learning. Langkah pertama adalah membaca parameter durasi prediksi dari command line argument, yang kemudian diikuti dengan mengakses file Excel berisi data pemesanan tiket kapal yang terdapat pada beberapa sheet.

```
bulan_prediksi = int(sys.argv[1]) if len(sys.argv) > 1 else
12

file_path = 'data/pemesanan.xlsx'
sheets = pd.ExcelFile(file_path).sheet_names
data_frames = []

for sheet in sheets:
    df = pd.read_excel(file_path, sheet_name=sheet)
```

```
    df['Tahun'] = sheet
    data_frames.append(df)

data = pd.concat(data_frames, ignore_index=True)
```

2. Deteksi Skala Data

Sistem melakukan evaluasi mendalam terhadap rata-rata dan nilai maksimum jumlah pemesanan historis untuk mengategorikan data dalam empat tingkat skala: normal (di bawah 100), medium (100-299), high (300-699), dan very high (700 ke atas). Setiap kategori skala diberi multiplier dan rentang target yang berbeda, yang dirancang untuk memastikan prediksi yang dihasilkan tetap sesuai dengan pola bisnis yang berlaku dan realistis.

```
avg_jumlah_pemesanan = data['Jumlah_pemesanan'].mean()
max_jumlah_pemesanan = data['Jumlah_pemesanan'].max()

if avg_jumlah_pemesanan >= 50 or max_jumlah_pemesanan >= 100:
    scale_category = "medium"
    scale_multiplier = 1.0
else:
    scale_category = "normal"
    scale_multiplier = 1.0

is_high_scale = scale_category in ["high", "very_high",
"medium"]
```

3. Data Cleaning

Di tahap ini, sistem melakukan pembersihan dan validasi data, dengan penekanan pada konversi format tanggal yang mengikuti standar Indonesia (format hari lebih dulu). Proses ini sangat penting karena ketidakkonsistenan atau ketidakvalidan data tanggal dapat menyebabkan kesalahan pada tahap feature engineering dan prediksi.

```
data['Tanggal'] = pd.to_datetime(data['Tanggal'],
dayfirst=True, errors='coerce')
data = data.dropna(subset=['Tanggal'])
```

4. Train Model

Tahap training model merupakan inti dari proses machine learning dimana sistem membangun dan mengoptimalkan model Random Forest untuk memprediksi jumlah pemesanan tiket kapal. Proses dimulai dengan pemisahan data menjadi fitur input (X) yang berisi semua kolom kecuali target, dan variabel target (y) yang berisi kolom Jumlah_Pemesanan yang akan diprediksi. Sistem kemudian membangun pipeline yang mengintegrasikan tahap preprocessing dan model Random Forest untuk memastikan konsistensi dalam pemrosesan data.

```
X = data.drop('Jumlah_pemesanan', axis=1)
y = data['Jumlah_pemesanan']

model = Pipeline([
    ('preprocessor', preprocessor),
    ('regressor', RandomForestRegressor(random_state=42))
])

param_grid = {
    'regressor__n_estimators': [100, 200],
    'regressor__max_depth': [None, 10, 20],
    'regressor__min_samples_split': [2, 5]
}

tscv = TimeSeriesSplit(n_splits=5)
grid_search = GridSearchCV(model, param_grid, cv=tscv,
scoring='neg_mean_squared_error')
grid_search.fit(X, y)
best_model = grid_search.best_estimator_
```

5. Generate Future Data untuk Prediksi

Tahap generate future data merupakan proses pembentukan dataset untuk periode prediksi yang akan datang dengan struktur fitur yang identik dengan data training. Sistem menentukan tanggal terakhir dari data historis sebagai titik awal, kemudian membuat rentang tanggal harian untuk periode prediksi yang diinginkan.

```
last_date = data['Tanggal'].max()
periods = bulan_prediksi * 30
```

```
unique_ships = data['Nama_Kapal'].unique()
all_predictions = []

for kapal in unique_ships:
    future_data = pd.DataFrame({
        'Tanggal': pd.date_range(start=last_date +
timedelta(days=1), periods=periods, freq='D'),
        'Nama_Kapal': kapal,
        'Tahun': str(last_date.year + 1)
    })

    future_data['Hari'] = future_data['Tanggal'].dt.day
    future_data['Bulan'] = future_data['Tanggal'].dt.month
    future_data['Hari_Minggu'] =
(future_data['Tanggal'].dt.dayofweek == 6).astype(int)
    future_data['Akhir_Pekan'] =
(future_data['Tanggal'].dt.dayofweek >= 5).astype(int)
    future_data['Musim'] = future_data['Bulan'].apply(lambda
x: 'Hujan' if x in [11,12,1,2,3] else 'Kemarau')
    future_data['LiburPanjang'] =
future_data['Bulan'].isin([6,12]).astype(int)
```

6. Optimasi Seasonal Pattern dengan Hari Libur

Proses untuk menciptakan pola musiman yang akurat berdasarkan analisis mendalam terhadap data historis dan karakteristik kalender Indonesia. Sistem menganalisis rata-rata hari libur per bulan dari data historis, kemudian membangun pola seasonal dengan menggabungkan base seasonal factor dengan holiday boost tambahan yang dihitung berdasarkan rasio hari libur historis.

```
holiday_months_analysis =
data.groupby(data['Tanggal'].dt.month)['Hari_Libur'].mean()

seasonal_patterns = {}
for month in range(1, 13):
    holiday_ratio = holiday_months_analysis.get(month, 0)

    if month == 1:
        base_factor = 1.4
    elif month == 12:
        base_factor = 1.5
    elif month in [6, 7]:
        base_factor = 1.2
    elif month == 8:
        base_factor = 1.1
    elif month in [2, 11]:
        base_factor = 0.9
    else:
        base_factor = 1.0
```

```
    if month in [1, 12]:
        holiday_boost = 1.0 + (holiday_ratio * 0.3)
    else:
        holiday_boost = 1.0 + (holiday_ratio * 0.2)

    seasonal_patterns[month] = base_factor * holiday_boost
```

7. Distribusi Prediksi Berdasarkan Pola Historis

Proses untuk memastikan bahwa prediksi per kapal tetap realistis dan proporsional sesuai dengan karakteristik historis masing-masing kapal. Sistem menghitung bobot global per kapal berdasarkan total pemesanan historis relatif terhadap keseluruhan armada, kemudian menganalisis share bulanan setiap kapal dari data historis yang dirata-rata lintas tahun untuk memahami pola kontribusi bulanan masing-masing kapal.

```
hist_ship_total =
data.groupby('Nama_Kapal')['Jumlah_pemesanan'].sum()
hist_total_all = float(hist_ship_total.sum()) if
float(hist_ship_total.sum()) > 0 else 1.0
global_weights = {str(k): float(v) / hist_total_all for k, v
in hist_ship_total.items()}

hist_month_ship = data.groupby(['Nama_Kapal',
data['Tanggal'].dt.month])['Jumlah_pemesanan'] \
                      .sum() \
                      .unstack(fill_value=0)
monthly_totals_hist = hist_month_ship.sum(axis=0)
monthly_shares = {}
for ship in hist_month_ship.index:
    monthly_shares[str(ship)] = {}
    for m in range(1, 13):
        denom = float(monthly_totals_hist.get(m, 0.0))
        share = float(hist_month_ship.at[ship, m]) / denom if
denom > 0 else 0.0
        monthly_shares[str(ship)][m] = share

ship_p90 = hist_month_ship.quantile(0.9, axis=1)
ship_avg = hist_month_ship.mean(axis=1)
ship_max = hist_month_ship.max(axis=1)
```

8. Aplikasi Target Range dan Seasonal Pattern

Pada Tahap aplikasi target range dan seasonal pattern merupakan proses kalibrasi akhir yang menggabungkan semua analisis sebelumnya untuk

menghasilkan prediksi bulanan yang realistis dan dapat diandalkan. Sistem menetapkan target range berdasarkan kategori skala data kemudian menerapkan seasonal patterns yang telah dioptimasi dengan boost hari libur untuk setiap bulan, dimana base prediction dihitung dari target average dikalikan seasonal factor.

```
if scale_category == "medium":
    target_min = 65
    target_max = 150
    target_avg = (target_min + target_max) / 2
else:
    target_min = 50
    target_max = 120
    target_avg = (target_min + target_max) / 2

for i, row in monthly_pred.iterrows():
    month_num = row['Bulan']

    seasonal_factor = seasonal_patterns.get(month_num, 1.0)
    base_prediction = target_avg * seasonal_factor

    import random
    random.seed(month_num * 42)
    variation = random.uniform(0.95, 1.05)

    predicted_value = base_prediction * variation

    max_limit = target_max
    if month_num in [1, 12]:
        if scale_category == "medium":
            max_limit = target_max * 1.3
        else:
            max_limit = target_max * 1.3
    elif seasonal_factor > 1.1:
        if scale_category == "medium":
            max_limit = target_max * 1.2
        else:
            max_limit = target_max * 1.2

    predicted_value = min(max(predicted_value, target_min),
max_limit)

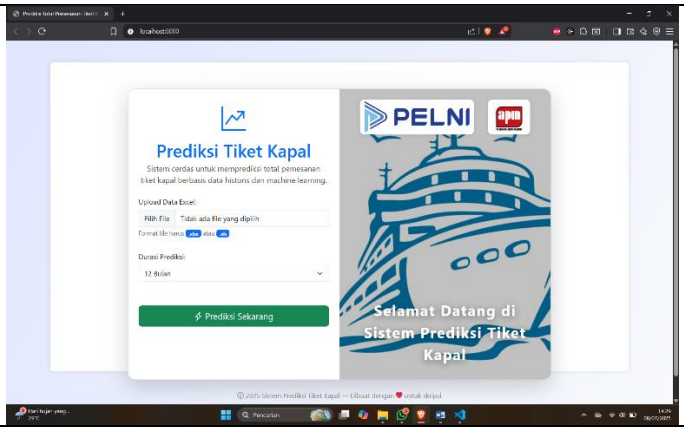
    monthly_pred.at[i, 'Prediksi'] = predicted_value
```

D. Pengujian Sistem

Pengujian sistem ini dilakukan dengan menggunakan 4 metode, yaitu *Blackbox* dan *Whitebox*, untuk menjelaskan serta membuktikan sistem berjalan dengan lancar, kemudian *Mean Absolute Percentage Error*, untuk membandingkan hasil prediksi dengan data asli, dan *Usability Testing* untuk menguji interaksi antara pengguna dan website.

1. Pengujian *Black Box*

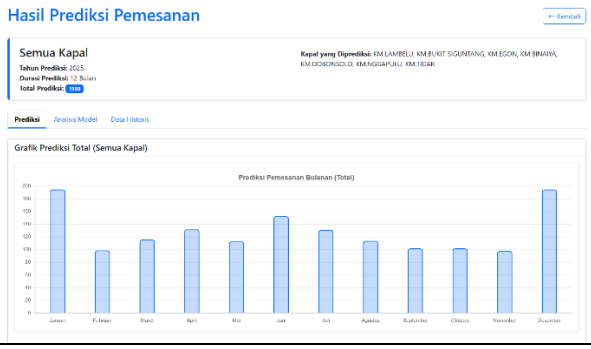
Tabel 4. 2 Pengujian *Blackbox* Halaman Utama

Test Faktor	Hasil	Keterangan
User pertama kali mengakses website	✓	Sukses, tampilkan halaman utama
Screenshot		
		

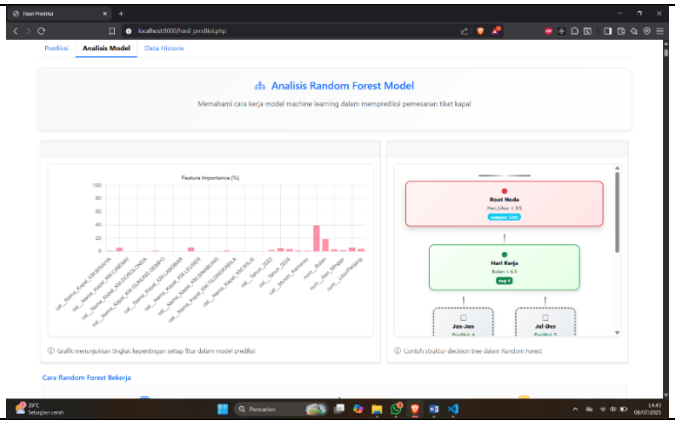
Tabel 4. 3 Pengujian *Blackbox* Upload File

Test Faktor	Hasil	Keterangan
User meng-upload file excel	✓	Sukses, file excel berhasil terinput
Screenshot		
		

Tabel 4. 4 Pengujian *Blackbox* Halaman Prediksi

Test Faktor	Hasil	Keterangan
User memulai prediksi	✓	Sukses, grafik prediksi berhasil muncul
Screenshot		
		

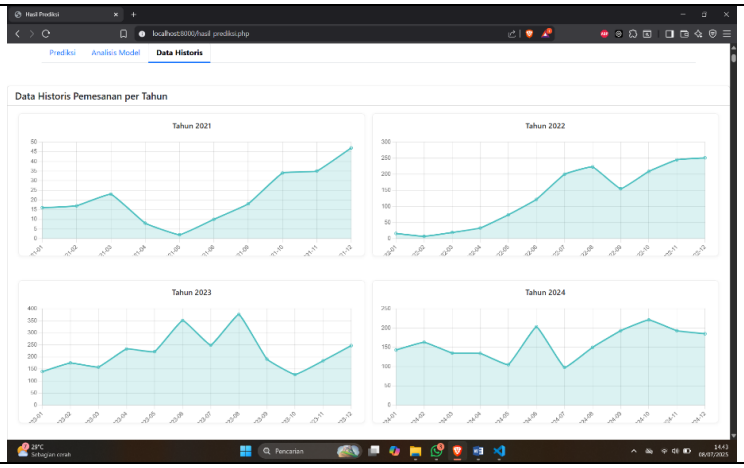
Tabel 4. 5 Pengujian *Blackbox* Analisis Model

Test Faktor	Hasil	Keterangan
User melihat analisis model	✓	Sukses, analisis model berhasil muncul
Screenshot		
		

Tabel 4. 6 Pengujian *Blackbox* Data Historis

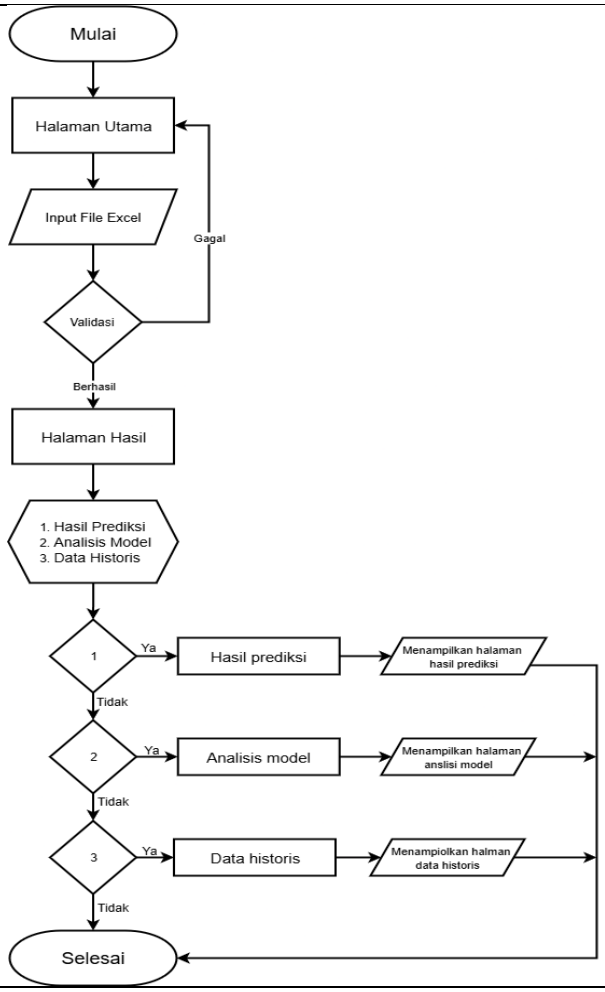
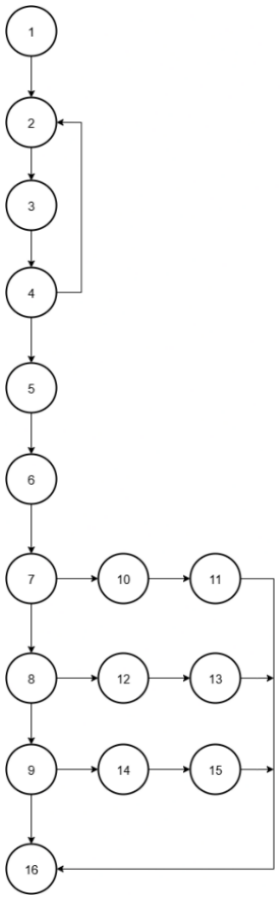
Test Faktor	Hasil	Keterangan
User melihat data historis	✓	Sukses, grafik data historis berhasil muncul

Lanjutan Tabel 4.6

Screenshot


2. Pengujian *White Box*

Tabel 4. 7 Pengujian *Whitebox* Prediksi

Flowchart	Flowgraph
	

Berdasarkan gambar diatas, dilakukan perhitungan sebagai berikut

a) Menghitung *Cyclomatic Complexity* V (G) = E – N + 2

   N (node) = 16

   E (edge) = 19

   P (predikat) = 4

   Penyelesaian V (G) = E – N + 2

   = 19 – 16 + 2

   = 5

   Predikat Node (N) = P + 1

   = 4 + 1

   = 5

b) Berdasarkan perhitungan *Cyclomatic Complexity* dari *flowgraph* diatas memiliki *Region* = 5

c) *Independent path* pada flowgraph diatas adalah:

   Path 1 = 1-2-3-4-5-6-7-8-9-16

   Path 2 = 1-2-3-4-2-3-4-5-6-7-8-9-16

   Path 3 = 1-2-3-4-5-6-7-10-11-16

   Path 4 = 1-2-3-4-5-6-7-8-12-13-16

   Path 5 = 1-2-3-4-5-6-7-8-9-14-15-16

d) Grafik Matriks

Tabel 4. 8 Grafik matriks prediksi

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	E - 1
1		1															1 - 1 = 0
2			1	1													2 - 1 = 1
3				1													1 - 1 = 0
4					1												1 - 1 = 0
5						1											1 - 1 = 0
6							1										1 - 1 = 0
7								1		1							2 - 1 = 1
8									1			1					2 - 1 = 1
9														1		1	2 - 1 = 1
10											1						1 - 1 = 0
11																1	1 - 1 = 0
12													1				1 - 1 = 0
13																1	1 - 1 = 0
14															1		1 - 1 = 0
15																1	1 - 1 = 0
16																	0
SUM (E + 1)																	4 + 1 = 5

3. *Mean Absolute Percentage Error*

Sebelum memulai perhitungan *Mean Absolute Percentage Error* (MAPE), langkah pertama yang perlu dilakukan adalah menentukan proses perhitungan manual untuk memperoleh hasil prediksi per-kapal. Sebagai contoh, untuk mendapatkan hasil prediksi kapal KM.BUKIT SIGUNTANG dalam bulan januari hingga mei. Adapun urutan untuk mendapatkan hasil prediksi yaitu

1) Prediksi Total Bulanan (dari *Random Forest*)

   Prediksi bulanan berdasarkan model *Random Forest* yang telah dilatih dengan data historis 2021-2024.

2) Data Historis Kapal

   Data pemesanan kapal historis untuk menentukan bobot global untuk kapal tertentu, seperti Bukit Siguntang. Cara menentukan bobot global yaitu :

$$Bobot_{Global} = Total_{Kapal_{BS}} \div Total_{Semua_{Kapal}} \quad \text{.....(3)}$$

$$Bobot_{Global} = 2{,}847 \div 6{,}441 = 0.442 = 44.2\% \quad \text{.....(4)}$$

3) Hitung Proporsi Bulanan Historis

Proporsi ini penting untuk memahami bagaimana distribusi pemesanan bulanan dalam periode tertentu. Cara menentukan hitung proporsi bulanan historis yaitu :

$$Monthly_{Share} = Total_{Historis_{BS}} \div Total_{Historis_{Kapal}} \text{.....(5)}$$

$$Monthly_{ShareJun} = 1{,}247 \div 2{,}873 = 0.434 = 43.4\% \text{.....(6)}$$

$$Monthly_{ShareFeb} = 712 \div 1{,}565 = 0.455 = 45.5\% \text{ .....(7)}$$

$$Monthly_{ShareMar} = 600 \div 2{,}317 = 0.259 = 25.9\% \text{.....(8)}$$

$$Monthly_{ShareApr} = 706 \div 1{,}792 = 0.394 = 39.4\% \text{ .....(9)}$$

$$Monthly_{ShareMei} = 664 \div 1{,}785 = 0.372 = 37.2\% \text{.....(10)}$$

4) Hitung Blended Share

Blended Share ini penting untuk menggabungkan pola musiman dengan konsistensi global untuk mendapatkan hasil prediksi kapal yang akurat dan realistis. Cara menentukan blended share yaitu :

$$Blended_{Share} = \alpha \times Monthly_{Share} + (1 - \alpha) \times Bobot_{Global} \text{.....(11)}$$

$$Blended_{ShareJan} = 0.90 \times 43.4\% + 0.10 \times 44.2\% = 43.48\% \text{ .....(12)}$$

$$Blended_{ShareFeb} = 0.90 \times 45.5\% + 0.10 \times 44.2\% = 45.37\% \text{ .....(13)}$$

$$Blended_{ShareMar} = 0.90 \times 25.9\% + 0.10 \times 44.2\% = 27.73\% \text{ .....(14)}$$

$$Blended_{ShareApr} = 0.90 \times 39.4\% + 0.10 \times 44.2\% = 39.88\% \text{ .....(15)}$$

$$Blended_{ShareMei} = 0.90 \times 37.2\% + 0.10 \times 44.2\% = 37.90\% \text{ .....(16)}$$

5) Hitung Final Prediksi

Adapun cara untuk mendapatkan hasil prediksi dengan menggunakan rumus

$$Final_{Prediksi} = Total_{Bulanan} \times Blended_{Share} \text{.....(17)}$$

$Final_{PrediksiJan} = 198 \times 43.48\% = 86.09$ .....(18)

$Final_{PrediksiFeb} = 99 \times 45.37\% = 44.92$ .....(19)

$Final_{PrediksiMar} = 116 \times 27.73\% = 32.17$ .....(20)

$Final_{PrediksiApr} = 132 \times 39.88\% = 52.64$ .....(21)

$Final_{PrediksiMei} = 113 \times 37.90\% = 42.83$ .....(22)

Berikutnya dilakukan pengujian sistem untuk menganalisis kinerja model prediksi sejauh mana akurasi prediksi dapat tercapai yang diterapkan pada per-kapal dari bulan januari hingga mei

a. KM. BUKIT SIGUNTANG

Tabel 4. 9 Data Kapal KM. BUKIT SIGUNTANG tahun 2025

Bulan	Data Asli	Data Prediksi
Januari	94	86
Febuari	30	45
Maret	39	30
April	41	52
Mei	24	42

1) Menghitung Selisih Absolut Persentase untuk setiap titik data:

$MAPE_1 = \left|\frac{94-86}{94}\right| \times 100\% = 8.51\%$ .....(23)

$MAPE_2 = \left|\frac{30-45}{30}\right| \times 100\% = 50.00\%$ .....(24)

$MAPE_3 = \left|\frac{39-30}{39}\right| \times 100\% = 23.08\%$ .....(25)

$MAPE_4 = \left|\frac{41-52}{41}\right| \times 100\% = 26.83\%$ .....(26)

$MAPE_5 = \left|\frac{24-42}{24}\right| \times 100\% = 75.00\%$ .....(27)

2) Menghitung MAPE :

$MAPE = \frac{8.51+50.00+23.08+26.83+75.00}{5} = \frac{183.42}{5} = 36.684$ .....(28)

Ini berarti rata-rata kesalahan prediksi dibandingkan dengan data asli adalah sekitar 36.68%.

b. KM. LAMBELU

Tabel 4. 10 Data Kapal KM. LAMBELU tahun 2025

Bulan	Data Asli	Data Prediksi
Januari	64	55
Febuari	30	28
Maret	26	52
April	60	49
Mei	27	44

1) Menghitung Selisih Absolut Persentase untuk setiap titik data:

$$MAPE_1 = \left|\frac{64-55}{64}\right| \times 100\% = 16.06\% \quad \text{.....(29)}$$

$$MAPE_2 = \left|\frac{30-28}{30}\right| \times 100\% = 06.67\% \quad \text{.....(30)}$$

$$MAPE_3 = \left|\frac{26-52}{26}\right| \times 100\% = 100.00\% \quad \text{.....(31)}$$

$$MAPE_4 = \left|\frac{60-49}{60}\right| \times 100\% = 18.33\% \quad \text{.....(32)}$$

$$MAPE_5 = \left|\frac{27-44}{27}\right| \times 100\% = 62.96\% \quad \text{.....(33)}$$

2) Menghitung MAPE :

$$MAPE = \frac{14.06+6.67+100.00+18.33+62.96}{5} = \frac{1202.02}{5} = 40.404 \quad \text{.....(34)}$$

Ini berarti rata-rata kesalahan prediksi dibandingkan dengan data asli adalah sekitar 40.40%.

c. KM. EGON

Tabel 4. 11 Data Kapal KM. EGON tahun 2025

Bulan	Data Asli	Data Prediksi
Januari	19	30
Febuari	13	14
Maret	25	17
April	12	16
Mei	36	16

1) Menghitung Selisih Absolut Persentase untuk setiap titik data:

$$MAPE_1 = \left|\frac{19-30}{19}\right| \times 100\% = 57.89\% \quad \text{.....(35)}$$

$$MAPE_2 = \left|\frac{13-14}{13}\right| \times 100\% = 07.69\% \quad \text{.....(36)}$$

$$MAPE_3 = \left|\frac{25-17}{25}\right| \times 100\% = 32.33\% \quad \text{.....(37)}$$

$$MAPE_4 = \left|\frac{16-04}{16}\right| \times 100\% = 33.33\% \quad \text{.....(38)}$$

$$MAPE_5 = \left|\frac{16-20}{16}\right| \times 100\% = 55.56\% \quad \text{.....(39)}$$

2) Menghitung MAPE :

$$MAPE = \frac{57.89+7.69+32.00+33.33+55.56}{5} = \frac{186.47}{5} = 37.294 \quad \text{.....(40)}$$

Ini berarti rata-rata kesalahan prediksi dibandingkan dengan data asli adalah sekitar 37.29%

d. KM. BINAIYA

Tabel 4. 12 Data Kapal KM. BINAIYA tahun 2025

Bulan	Data Asli	Data Prediksi
Januari	21	16
Febuari	4	7
Maret	24	9
April	11	11
Mei	1	9

1) Menghitung Selisih Absolut Persentase untuk setiap titik data:

$$MAPE_1 = \left|\frac{21-16}{21}\right| \times 100\% = 23.81\% \quad \text{.....(41)}$$

$$MAPE_2 = \left|\frac{04-07}{04}\right| \times 100\% = 75.00\% \quad \text{.....(42)}$$

$$MAPE_3 = \left|\frac{24-09}{24}\right| \times 100\% = 62.50\% \quad \text{.....(43)}$$

$$MAPE_4 = \left|\frac{11-11}{11}\right| \times 100\% = 00.00\% \quad \text{.....(44)}$$

$$MAPE_5 = \left|\frac{01-09}{01}\right| \times 100\% = 00.08\% \text{ .....(45)}$$

2) Menghitung MAPE :

$$MAPE = \frac{23.81+75.00+62.50+00.00+00.08}{5} = \frac{162.13}{5} = 32.42 \text{ .....(46)}$$

Ini berarti rata-rata kesalahan prediksi dibandingkan dengan data asli adalah sekitar 32.42%.

e. KM. TIDAR

Tabel 4. 13 Data Kapal KM. TIDAR tahun 2025

Bulan	Data Asli	Data Prediksi
Januari	0	3
Febuari	4	1
Maret	2	10
April	1	3
Mei	0	1

1) Menghitung Selisih Absolut Persentase untuk setiap titik data:

$$MAPE_1 = \left|\frac{0-03}{0}\right| \times 100\% = 10.64\% \text{ .....(47)}$$

$$MAPE_2 = \left|\frac{04-01}{04}\right| \times 100\% = 75.00\% \text{ .....(48)}$$

$$MAPE_3 = \left|\frac{02-10}{02}\right| \times 100\% = 40.00\% \text{ .....(49)}$$

$$MAPE_4 = \left|\frac{01-03}{01}\right| \times 100\% = 02.00\% \text{ .....(50)}$$

$$MAPE_5 = \left|\frac{0-1}{0}\right| \times 100\% = 79.17\% \text{ .....(51)}$$

2) Menghitung MAPE :

$$MAPE = \frac{03.00+75.00+40.00+02.00+01.00}{5} = \frac{24.2}{5} = 04.84 \text{ .....(52)}$$

Ini berarti rata-rata kesalahan prediksi dibandingkan dengan data asli adalah sekitar 04.84%.

4. *Usability* Testing

Tabel 4. 14 Pertanyaan

Pertanyaan
Seberapa puas Anda dengan akurasi prediksi yang diberikan oleh sistem pemesanan tiket ini?
Seberapa puas Anda dengan kecepatan sistem dalam memberikan hasil prediksi?
Seberapa mudah Anda merasa dalam menggunakan sistem pemesanan tiket ini?
Secara keseluruhan, seberapa puas Anda dengan sistem prediksi pemesanan tiket ini?

Keempat pertanyaan tersebut mempunyai indikator nilai yang di sajikan dalam tabel 4.15

Tabel 4. 15 Indikator Penilaian Kuesioner

Nilai	Keteranagan
4	Sangat Setuju
3	Setuju
2	Cukup
1	Tidak Setuju

Selanjutnya hasil pengisian kuisioner tersebut dihitung, dan hasilnya diukur dengan menggunakan indikator kepuasaan pengguna

Tabel 4. 16 Indikator Kepuasan Pengguna

Presentasi	Keteranagan
81 – 100%	Sangat Puas
61 – 80%	Puas
41 – 60%	Cukup Puas
21 – 40%	Kurang Puas
1 – 20%	Tidak Puas

Hasil dari 2 orang responden ditampilkan dalam tabel berikut

Tabel 4. 17 Dokumentasi pengguna

Nama	Jabatan	Uraian Kegiatan	Dokumentasi
Basri	Direktur	Pengguna mencoba sistem prediksi	

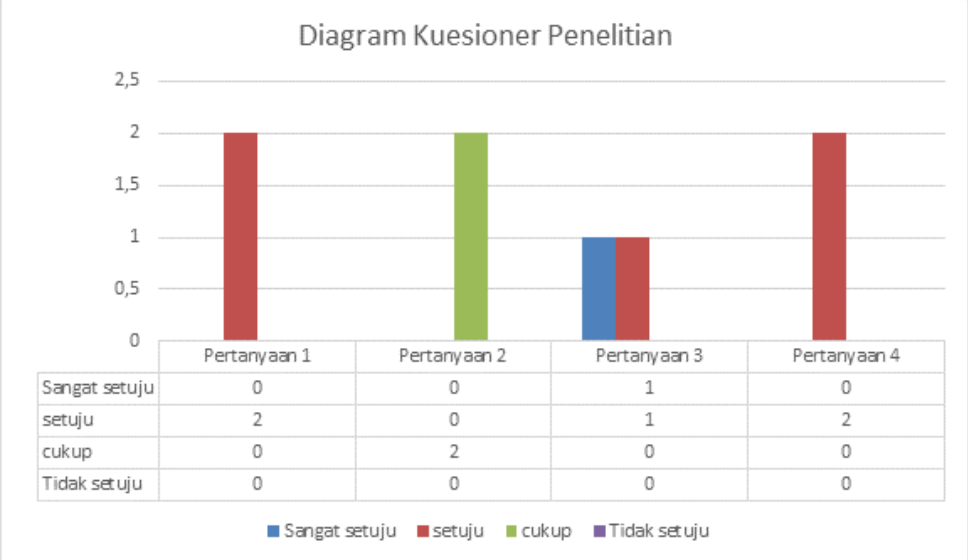
Lanjutan Tabel 4.17

Nama	Jabatan	Uraian Kegiatan	Dokumentasi
Hesti Rahmawati	Karyawan	Pengguna mencoba melihat data historis	

Tabel 4. 18 Hasil Pengisian Kuesioner

Nama	Jabatan	Daftar Pertanyaan				Total Nilai
		1	2	3	4	
Basri	Direktur	3	2	3	3	11
Hesti Rahmawati	Karyawan	3	2	4	3	12

Tabel di atas menggambarkan hasil survei mengenai persepsi responden terhadap pertanyaan yang diajukan. Untuk memudahkan pemahaman, diagram berikut menunjukkan distribusi data tersebut secara visual



Gambar 4. 9 Diagram Kuesioner Penelitian

Jumlah nilai maksimal = Nilai tertinggi indikator penilaian × Jumlah pertanyaan

= 4 × 4 = 16

Jumlah responden = 2 orang

Jumlah kepuasan maksima = 16 × 2 = 32

Skor Kepuasan Pengguna = $\frac{Total\ Nilai}{Jumlah\ Kepuasan\ maksima} \times 100\%$

$= \frac{23}{32} \times 100\%$

$= 71{,}8\%$

Dari hasil tabel tersebut didapatkan skor 71,8% yang diketahui bahwa pengguna puas dengan sistem prediksi pemesana tiket kapal yang digunkan

BAB V

KESIMPULAN

A. Kesimpulan

Berdasarkan pengujian yang telah dilakukan, diperoleh MAPE untuk data tahun 2025, dapat disimpulkan bahwa terdapat perbedaan antara nilai prediksi dan nilai asli untuk setiap kapal yang dianalisis. Nilai kesalahan rata-rata prediksi (MAPE) menunjukkan tingkat ketepatan model prediksi pada setiap kapal sebagai berikut:

1. KM. Bukit Siguntang menunjukkan bahwa kesalahan prediksi (MAPE) rata-rata adalah sekitar 36.68%, yang mengindikasikan adanya perbedaan signifikan antara data asli dan data prediksi, terutama pada bulan Januari yang mengalami penurunan signifikan.
2. KM. Lambelu memiliki nilai MAPE sebesar 40.40%, yang berarti kesalahan prediksi untuk kapal ini lebih tinggi daripada KM. Bukit Siguntang. Meskipun data pada Februari dan Maret cenderung lebih akurat, ketidaktepatan lebih terlihat pada bulan lainnya, seperti Januari dan April.
3. KM. Egon memiliki nilai MAPE sebesar 37.29%. Perbedaan antara data asli dan prediksi pada kapal ini juga menunjukkan kesalahan yang cukup besar, meskipun perbedaan pada bulan Februari dan Maret lebih kecil.
4. KM. Binaiya tercatat dengan MAPE sebesar 32.42%, yang sedikit lebih rendah dibandingkan kapal lainnya, namun tetap menunjukkan adanya

kesalahan dalam prediksi yang cukup signifikan, khususnya pada bulan Januari dan Maret.

5. KM. Tidar memiliki MAPE yang sangat rendah, yaitu 4.84%,

Meskipun model prediksi yang digunakan dalam analisis ini menunjukkan hasil yang bervariasi, beberapa kapal, seperti KM. LAMBELU dan KM. BUKIT SIGUNTANG, memiliki MAPE yang cukup tinggi, yaitu di atas 30%. Hal ini menunjukkan bahwa model prediksi tidak sepenuhnya akurat dan bisa menghasilkan kesalahan yang signifikan.

Selain itu, meskipun KM. TIDAR memiliki hasil yang sangat baik dengan MAPE 4.84%, model ini masih tidak sepenuhnya dapat diandalkan untuk keseluruhan data, terutama untuk kapal lainnya yang memiliki kesalahan prediksi yang lebih tinggi. Hal ini menunjukkan bahwa model ini mungkin kurang cocok untuk data yang digunakan, dan diperlukan pendekatan lain yang lebih sesuai untuk meningkatkan akurasi prediksi di masa depan.

B. Saran

Untuk pengembangan selanjutnya, penulis mengusulkan agar sistem prediksi ini dapat mengintegrasikan algoritma lain, seperti *Support Vector Machine* (SVM), ARIMA, dan *Regresi Linear*. Dengan menerapkan berbagai algoritma ini, diharapkan dapat meningkatkan keberagaman dalam pemodelan dan menghasilkan prediksi yang lebih tepat.

DAFTAR PUSTAKA

Candra, A, Faizah, AN, & Sulaiman, H (2025). Digital Literacy and Training Ms. Office Excel. *Jurnal Visi Pengabdian …*, ejournal.uhn.ac.id,

Deni, DK, & Ferida, FY (2023). Usability Testing Penggunaan Menu Kartu Hasil Studi Di Website Sistem Informasi Akademik Universitas Teknologi Yogyakarta. *Jurnal Teknologi dan Manajemen Industri*

Dewi, BES, Haikal, S, Sulistyowati, HS, & ... (2024). Penerapan Machine Learning Menggunakan Algoritma Random Forest Untuk Prediksi Harga Mobil Bekas. *Jurnal Teknologi …*, Banisalehjurnal.Ubs.Ac.Id,

Dewi, SP, Nurwati, N, & Rahayu, E (2022). Penerapan Data Mining Untuk Prediksi Penjualan Produk Terlaris Menggunakan Metode K-Nearest Neighbor.

Enterprise, J (2023). *HTML, PHP, dan MySQL untuk Pemula (Update Version)*., books.google.com,

Faisal, MR (2017). Seri Belajar ASP. NET: ASP. NET Core MVC &MySQL dengan Visual Studio Code.

Fauzi, K (2024). Sistem Prediksi Penjualan Produk: Analisis Cerdas dengan Algoritma FP-Growth.

Ginantra, N. L. W. S. R. & Anandita, I. B. G., 2019. Penerapan Metode Single Exponential Smoothing Dalam Peramalan Penjualan Barang. Jurnal Sains Komputer dan Informatika (J SAKTI),

Haryono, W, Thoyyibah, T, Puspitasari, T, Maulida, R, & ... (2021). Pelatihan Pembuatan dan Pengelolaan Website sebagai Sarana Informasi pada Madrasah Tsanawiyah Al-Fatah Mandiri Jakarta. JAMAIKA: Jurnal Abdi

Haviluddin. (2011). Memahami Penggunaan UML (Unified Modelling Language). Memahami Penggunaan UML (Unified Modelling Language), 6(1), 1–15.

Hirzi, RH, Hidayaturrohman, U, Kertanah, K, & ... (2023). Prediksi Jumlah Wisatawan Menggunakan Metode Random Forest, Single Exponential Smoothing dan Double Exponential Smoothing di Provinsi NTB.

Ibna, AZ, & Nasution, MIP (2024). Implikasi Penggunaan Basis Data dalam Era Big Data. *Journal Sains Student …*, ejurnal.kampusakademik.co.id,

Jamaluddin, M (2021). Algoritma dan Pemrograman Komputer dengan Python.,

Julianto, S., & Setiawan, S. (2019). Perancangan Sistem Informasi Pemesanan Tiket Bus Pada Po. Handoyo Berbasis Online. Simatupang, Julianto Sianturi, Setiawan, 3(2), 11–25.

Kamal, IM, & Ilyas, R (2017). Prediksi penjualan buku menggunakan data mining di pt. niaga swadaya.

Khalim, KA, Hayati, U, & Bahtiar, A (2023). Perbandingan Prediksi Penyakit Hipertensi Menggunakan Metode Random Forest Dan Naïve Bayes.

Listiono, E. D., Surahman, A., & Sintaro, S. (2021). Ensiklopedia Istilah Geografi Menggunakan Metode Sequential Search Berbasis Android Studi Kasus: Sma Teladan Way Jepara Lampung Timur. Jurnal Teknologi Dan Sistem Informasi, 2(1), 35–42.

Mardiani, E, Rahmansyah, N, Wahyudi, NM, Wijaya, YF, & ... (2021). *Kumpulan Latihan PHP*., books.google.com,

Maulana, Rizki Ali (2021) Aplikasi Prediksi Kebutuhan Bahan Baku Barang Cetakan Menggunakan Metode Arima Berbasis Web.

N. Noviyanto, "Penerapan Data Mining dalam Mengelompokkan Jumlah Kematian Penderita COVID-19 Berdasarkan Negara di Benua Asia," Paradig.

Noviantoro, A, Silviana, AB, Fitriani, RR, & ... (2022). Rancangan Dan Implementasi Aplikasi Sewa Lapangan Badminton Wilayah Depok Berbasis Web.

Okta, OS (2022). Klasifikasi Pada Literasi Membaca Buku Oleh Mahasiswa Menggunakan Algoritma Random Forest Di Perguruan Tinggi Lampung.

Pramesti, D, & Baihaqi, WM (2023). Perbandingan Prediksi Jumlah Transaksi Ojek Online Menggunakan Regresi Linier dan Random Forest.

Praniffa, A. C., Syahri, A., Sandes, F., Fariha, U., Giansyah, Q. A., & Hamzah, M. (2023). Pengujian Sistem Informasi Parkir Berbasis Web Pada UIN SUSKA RIAU Menggunakan White Box dan Black Box Testing. *Jurnal Testing dan Implementasi Sistem Informasi*, *1*(1), 1-16.

Putra, N. S., Ritchi, H., & Alfian, A. (2023). Hubungan Big Data Analytics terhadap Kualitas Audit: Penerapan pada Instansi Pemerintah. Jurnal Riset Akuntansi Dan Keuangan, 11(1), 57–72.

RAKHMAT, GA, & MUTOHAR, W (2023). Prakiraan Hujan menggunakan Metode Random Forest dan Cross Validation.

Rianto, M, & Yunis, R (2021). Analisis Runtun Waktu Untuk Memprediksi Jumlah Mahasiswa Baru Dengan Model Random Forest.

Ridlo, IA (2017). Panduan pembuatan flowchart. Fakultas Kesehatan Masyarakat, osf.io

Rizaldi, MF, Akrom, AA, & ... (2023). Pengenalan Website Sebagai Pengembangan Profil Pondok Pesantren MaQis Al-Hamidy 4. KARYA: Jurnal …, jurnalfkip.samawa-university.ac.id,

Roza, R, Fauzan, MN, & Rahayu, WI (2020). Tutorial sistem informasi prediksi jumlah pelanggan menggunakan metode regresi linier berganda berbasis web menggunakan framework codeigniter.

Santoso, RE, Prawiyogi, AG, Rahardja, U, & ... (2022). Penggunaan dan manfaat big data dalam konten digital. *ADI Bisnis Digital …*, adi-journal.org,

Shobori, I, Khairil, K, & Suryana, E (2024). Prediksi Penjualan Barang Menggunakan Metode Single Exponential Smoothing Pada Duo Sister Hijab Bengkulu.

Siburian, VW, & Mulyana, IE (2019). Prediksi Harga Ponsel Menggunakan Metode Random Forest. Annual Research Seminar (ARS), core.ac.uk,

Sudrajat, D., Purnamasari, A. I., Dikananda, A. R., Kurnia, D. A., & Bahtiar, A. (2022). Klasifikasi Mutu Pembelajaran Hybrid berdasarkan Algoritma C.45, Random Forest dan Naïve Bayes dengan Optimasi Bootsrap Areggating (Bagging) pada masa COVID-19. JURIKOM (Jurnal Riset Komputer), 9(6), 2227.

Syahrudin, AN, & Kurniawan, T (2018). Input dan output pada bahasa pemrograman python. Jurnal Dasar Pemograman Python

Wantono, S (2014). Prediksi Penyelesaian Studi Mahasiswa Baru dengan Metode Fuzzy Tsukamoto (Study Kasus di Universitas Muhammadiyah Gresik).

Wintana, D., Pribadi, D., & Nurhadi, M. Y. (2022). Analisis Perbandingan Efektifitas White-Box Testing dan Black-Box Testing. *Jurnal Larik Ladang Artikel Ilmu Komputer*, *2*(1), 8-16.