

Rancang Bangun Aplikasi Rekomendasi Produk Telekomunikasi Berbasis WEB Berdasarkan Perilaku Pelanggan

Dippo Ardy Wibowo¹, Hardika Khusnuliawati^{*2}, Farid Fitriyadi³

^{1,2,3}Program Studi Informatika, Fakultas Sains Teknologi dan Kesehatan,
Universitas Sahid Surakarta

Email: ¹dippoardy83@gmail.com, ^{*2}khusnuliawati@usahidsolo.ac.id, ³farid@usahidsolo.ac.id

(Naskah masuk: 2 Juni 2026, diterima untuk diterbitkan: 2 Juli 2026)

Abstrak: Banyaknya variasi paket telekomunikasi dapat menyulitkan pelanggan dalam memilih layanan yang sesuai dengan kebutuhan penggunaan mereka. Penelitian ini bertujuan merancang dan membangun aplikasi rekomendasi produk telekomunikasi berbasis web berdasarkan perilaku pelanggan. Sistem dikembangkan menggunakan React.js pada sisi frontend, FastAPI pada sisi backend, serta PostgreSQL sebagai database. Model rekomendasi menggunakan algoritma Random Forest yang dilatih pada dataset Telco Customer Churn sebanyak 7.043 data pelanggan. Dataset diperkaya melalui pembentukan fitur perilaku dan label rekomendasi berbasis aturan bisnis, serta didukung data transaksi dan event sintetis untuk kebutuhan pengembangan aplikasi. Model menggunakan 21 fitur input untuk mengklasifikasikan pelanggan ke dalam enam kelas rekomendasi paket. Hasil pengujian terhadap 1.409 data uji menunjukkan akurasi sebesar 86,8%. Sistem juga menyediakan dashboard pengguna untuk menampilkan rekomendasi dan riwayat pembelian, serta dashboard admin untuk mengelola produk serta memantau rekomendasi pengguna. Hasil penelitian menunjukkan bahwa aplikasi mampu mengintegrasikan proses klasifikasi rekomendasi berbasis perilaku dengan penyajian paket secara terstruktur pada aplikasi web.

Kata Kunci – sistem rekomendasi; telekomunikasi; Random Forest; perilaku pelanggan; aplikasi web

Design and Development of a Web-Based Telecommunication Product Recommendation Application Based on Customer Behavior

Abstract: The wide variety of telecommunication packages can make it difficult for customers to select services that suit their usage needs. This study aims to design and develop a web-based telecommunication product recommendation application based on customer behavior. The system was developed using React.js for the frontend, FastAPI for the backend, and PostgreSQL as the database. The recommendation model applies the Random Forest algorithm trained on the Telco Customer Churn dataset consisting of 7,043 customer records. The dataset was enriched through behavioral feature construction and rule-based recommendation labels, supported by synthetic transaction and event data for application development purposes. The model uses 21 input features to classify customers into six package recommendation classes. Testing on 1,409 test records resulted in an accuracy of 86.8%. The system also provides a user dashboard for displaying recommendations and purchase history, as well as an admin dashboard for managing products and monitoring user recommendations. The results show that the application is able to integrate behavior-based recommendation classification with structured package presentation in a web application.

Keywords – recommendation system; telecommunication; Random Forest; customer behavior; web application

1. PENDAHULUAN

Perkembangan teknologi informasi dan komunikasi telah mendorong transformasi digital pada berbagai sektor industri, termasuk industri telekomunikasi [1]. Perusahaan telekomunikasi saat ini mulai memanfaatkan data pelanggan untuk memahami pola penggunaan layanan serta mendukung strategi pemasaran berbasis data [2]. Banyaknya variasi paket internet, telepon, dan layanan kombinasi yang tersedia sering kali menyebabkan pelanggan mengalami kesulitan dalam menentukan paket yang sesuai dengan kebutuhan penggunaan mereka [3].

Sistem rekomendasi merupakan salah satu pendekatan yang dapat digunakan untuk membantu pengguna memperoleh layanan yang sesuai berdasarkan data historis dan perilaku pengguna [4]. Penerapan sistem rekomendasi pada layanan digital dinilai mampu meningkatkan personalisasi layanan dan membantu proses pengambilan keputusan pengguna secara lebih efektif [1]. Pada sektor telekomunikasi, sistem rekomendasi dapat dimanfaatkan untuk memberikan rekomendasi paket layanan berdasarkan pola penggunaan pelanggan sehingga penawaran produk menjadi lebih tepat sasaran [2][3].

Penggunaan *Machine Learning* dalam sistem rekomendasi saat ini semakin banyak diterapkan karena mampu mempelajari pola penggunaan pelanggan dari data historis dan menghasilkan rekomendasi yang sesuai dengan kebutuhan pengguna [5]. Dalam penelitian ini digunakan algoritma Random Forest karena memiliki kemampuan klasifikasi yang baik, mampu menangani data dengan jumlah fitur yang cukup banyak, serta dapat mengurangi risiko *overfitting* melalui pendekatan *ensemble learning* [6][7]. Selain itu, *Random Forest* mampu menghasilkan performa prediksi yang stabil pada data dengan karakteristik kompleks sehingga sesuai digunakan dalam pengolahan data perilaku pelanggan telekomunikasi [6].

Selain pemilihan algoritma, proses *preprocessing* dan *feature engineering* juga memiliki peran penting dalam meningkatkan performa model *Machine Learning* [5]. Dengan pengolahan data yang tepat, sistem rekomendasi mampu menghasilkan rekomendasi paket telekomunikasi yang lebih relevan dan personal sesuai karakteristik masing-masing pengguna [7].

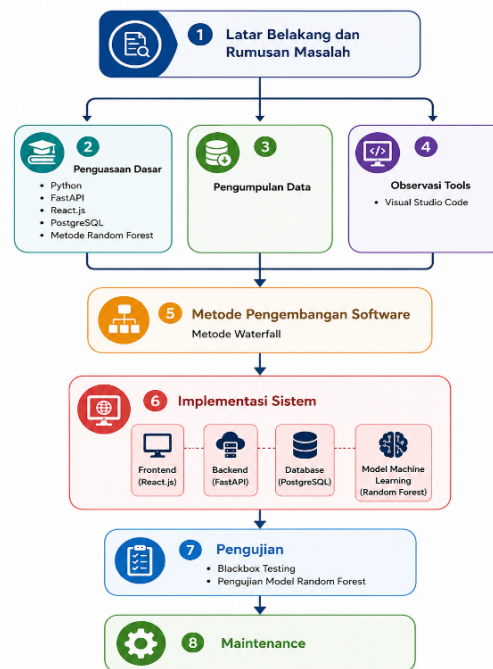
Meskipun penelitian terkait sistem rekomendasi telekomunikasi telah dilakukan, sebagian besar penelitian sebelumnya masih berfokus pada metode rekomendasi dan pengujian model, sedangkan implementasi sistem berbasis web yang terintegrasi langsung dengan *Machine Learning* secara *end-to-end* masih belum banyak dibahas secara menyeluruh [2][3].

Berdasarkan permasalahan tersebut, penelitian ini mengembangkan aplikasi rekomendasi produk telekomunikasi berbasis web menggunakan algoritma *Random Forest*. Sistem dikembangkan menggunakan React.js sebagai *frontend* [8], FastAPI sebagai *backend*, serta PostgreSQL sebagai *database* [9]. Model *Random Forest* digunakan untuk menganalisis perilaku pelanggan dan menghasilkan rekomendasi paket telekomunikasi berdasarkan pola penggunaan layanan pelanggan. Selain itu, sistem menerapkan pendekatan *top-k recommendation* sehingga pengguna dapat memperoleh beberapa rekomendasi berdasarkan hasil prediksi model [10]. Sistem ini diharapkan dapat membantu proses personalisasi layanan serta mendukung penawaran paket telekomunikasi secara lebih efektif dan tepat sasaran.

2. METODE PENELITIAN

2.1. Kerangka Pemikiran

Kerangka pemikiran penelitian ini diawali dari latar belakang mengenai banyaknya variasi paket telekomunikasi yang menyulitkan pelanggan dalam memilih paket sesuai kebutuhan dan pola penggunaan. Berdasarkan permasalahan tersebut, dirumuskan rancangan sistem rekomendasi produk telekomunikasi berbasis web menggunakan algoritma Random Forest. Penelitian ini didukung oleh penguasaan dasar teknologi seperti Python, FastAPI, React.js, PostgreSQL, serta pemahaman metode *Random Forest*, pengumpulan data, dan observasi tools pengembangan menggunakan *Visual Studio Code*. Selanjutnya, sistem dikembangkan menggunakan metode *Waterfall*, kemudian diimplementasikan dalam bentuk aplikasi web yang terdiri dari *frontend*, *backend*, *database*, dan model *machine learning*. Tahap akhir dilakukan melalui pengujian sistem menggunakan *blackbox testing* serta pengujian model *Random Forest* untuk mengetahui kinerja rekomendasi yang dihasilkan. Alur kerangka pemikiran ditunjukkan pada Gambar 1.



Gambar 1. Kerangka Pemikiran

2.2. Observasi

Observasi dilakukan dengan mempelajari kebutuhan sistem rekomendasi paket telekomunikasi berdasarkan perilaku pelanggan. Data yang diamati meliputi karakteristik penggunaan layanan, seperti jenis paket, merek perangkat, rata-rata penggunaan data, persentase penggunaan video, durasi panggilan, frekuensi SMS, pengeluaran bulanan, frekuensi top-up, skor perjalanan, serta atribut turunan lain yang dapat menggambarkan perilaku pelanggan.

Dataset yang digunakan berasal dari *Telco Customer Churn* yang kemudian diperkaya dengan target rekomendasi. Selain itu, sistem juga menggunakan data transaksi sintesis dan data *event* sintesis untuk mendukung kebutuhan pengembangan dan evaluasi. Proses observasi ini menjadi dasar dalam tahap *preprocessing*, *feature engineering*, pelatihan model, dan perancangan alur rekomendasi pada aplikasi web.

2.3. Studi Literatur

Studi literatur dilakukan dengan mempelajari konsep sistem rekomendasi, algoritma *Random Forest*, pengembangan aplikasi web, serta pengujian perangkat lunak. Literatur juga digunakan untuk memperkuat pemahaman mengenai teknologi yang digunakan dalam sistem, yaitu React.js sebagai *frontend*, FastAPI sebagai *backend*, PostgreSQL sebagai *database*. Hasil studi literatur digunakan sebagai landasan dalam menentukan metode penelitian, desain sistem, serta evaluasi model.

2.4. Metode Pengembangan Aplikasi

Metode pengembangan sistem yang digunakan adalah *Software Development Life Cycle* (SDLC) dengan model *Waterfall*. Model *Waterfall* merupakan salah satu model pengembangan perangkat lunak yang dilakukan secara terstruktur dan berurutan. Tahapan yang dilakukan meliputi analisis kebutuhan, desain sistem, implementasi, pengujian, *deployment*, dan *maintenance* [11].

2.4.1. Analisis Kebutuhan Sistem

Tahap analisis kebutuhan dilakukan berdasarkan permasalahan dalam proses penawaran paket telekomunikasi yang belum sesuai dengan perilaku dan kebutuhan pelanggan. Kebutuhan fungsional meliputi pengelolaan data pelanggan, pengelolaan data produk telekomunikasi, proses

rekomendasi paket menggunakan *Random Forest*, serta penyajian hasil rekomendasi pada *dashboard* aplikasi.

Sementara itu, kebutuhan *non-fungsional* mencakup keamanan sistem melalui autentikasi pengguna, performa aplikasi yang stabil, tampilan antarmuka yang responsif, serta kemudahan akses melalui berbagai perangkat.

2.4.2. Desain Sistem

Tahap desain dilakukan dengan merancang arsitektur sistem yang terdiri dari *frontend*, *backend*, *database*, dan modul *machine learning*. *Frontend* berfungsi sebagai antarmuka pengguna, *backend* menyediakan REST API dan logika bisnis, PostgreSQL menyimpan data pelanggan dan produk, sedangkan modul *Random Forest* bertugas menghasilkan rekomendasi berdasarkan fitur pelanggan.

2.4.3. Implementasi

Pada tahap implementasi, sistem dibangun berdasarkan rancangan yang telah dibuat sebelumnya. Pengembangan antarmuka aplikasi dilakukan menggunakan React.js dan Tailwind CSS agar tampilan sistem dapat berjalan secara responsif dan mudah digunakan oleh pengguna. Sementara itu, sisi backend dikembangkan menggunakan FastAPI yang berfungsi untuk mengelola proses bisnis, komunikasi API, serta integrasi dengan *database* PostgreSQL.

Model *Machine Learning* yang digunakan adalah *Random Forest* yang sebelumnya telah melewati proses *preprocessing* data dan pembentukan fitur perilaku pelanggan. Model tersebut digunakan untuk memproses data pelanggan dan menghasilkan rekomendasi paket telekomunikasi yang sesuai dengan pola penggunaan masing-masing pengguna. Hasil rekomendasi kemudian ditampilkan pada dashboard aplikasi secara otomatis melalui integrasi antara *backend* dan *frontend*.

2.4.4. Deployment

Tahap *deployment* dilakukan untuk mempublikasikan sistem agar dapat diakses secara online. *Frontend* aplikasi yang dikembangkan menggunakan React.js diimplementasikan menggunakan platform Vercel, sedangkan *backend* FastAPI diimplementasikan menggunakan platform Railway. Pemilihan platform ini didasarkan pada kemampuannya dalam mendukung pengembangan aplikasi web modern serta memudahkan integrasi antara *frontend*, *backend*, *database*, dan model *Machine Learning*.

2.4.5. Pengujian

Tahap pengujian dilakukan untuk memastikan sistem rekomendasi produk telekomunikasi berjalan sesuai dengan kebutuhan yang telah ditentukan. Pengujian fungsional dilakukan pada fitur utama seperti autentikasi pengguna, pengelolaan data pelanggan, pengambilan data produk, proses rekomendasi paket, serta tampilan dashboard aplikasi.

Evaluasi model *Machine Learning* dilakukan menggunakan *confusion matrix* dengan metrik *accuracy* untuk mengukur performa algoritma *Random Forest* dalam menghasilkan rekomendasi paket telekomunikasi berdasarkan perilaku pelanggan. Hasil evaluasi tersebut digunakan untuk mengetahui tingkat ketepatan model dalam memberikan rekomendasi sehingga sistem yang dikembangkan dapat digunakan secara optimal sebagai pendukung personalisasi layanan telekomunikasi.

2.4.6. Maintenance (Pemeliharaan)

Tahap *maintenance* (Pemeliharaan) dilakukan setelah sistem berhasil diimplementasikan dengan melakukan pengecekan dan perbaikan terhadap error yang ditemukan pada aplikasi. Selain itu, dilakukan pengoptimalan performa sistem serta penyesuaian tampilan aplikasi agar tetap

nyaman digunakan. Proses monitoring juga dilakukan secara berkala untuk memastikan sistem rekomendasi dapat berjalan dengan baik dan tetap menghasilkan rekomendasi paket telekomunikasi secara optimal.

3. HASIL DAN PEMBAHASAN

Bab ini menyajikan hasil pengembangan sistem serta evaluasi model *Machine Learning* yang digunakan dalam penelitian. Pembahasan difokuskan pada proses pengolahan data pelanggan, penerapan algoritma *Random Forest*, serta analisis performa model dalam menghasilkan rekomendasi paket telekomunikasi berdasarkan perilaku pelanggan. Selain itu, ditampilkan pula implementasi sistem berbasis web sebagai media penyajian hasil rekomendasi paket telekomunikasi untuk mendukung personalisasi layanan dan membantu proses penawaran produk secara lebih tepat sasaran.

3.1. Dataset dan Pre-Processing Data

Dataset yang digunakan berasal dari Telco Customer Churn (Kaggle) yang diperkaya dengan data transaksi dan *event* sintetis. Dataset hasil pengolahan terdiri dari 7.043 data pelanggan, 7.318 data transaksi sintetis, dan 136.686 data event sintetis. Ringkasan dataset ditunjukkan pada Tabel 1.

Tabel 1. Ringkasan Dataset

Keterangan	Nilai
Data Pelanggan	7.043
Data Transaksi Sintetis	7.318
Data event Sintetis	136.686
Jumlah Fitur Input Model	21
Jumlah Kelas Target	6
Rasio data latih : uji	80% : 20%

Preprocessing meliputi seleksi kolom, pembersihan nilai kosong menggunakan nilai median untuk kolom numerik, penghapusan duplikasi berdasarkan identitas pelanggan, dan konversi tipe data. Karena dataset asli tidak menyediakan label rekomendasi produk dan beberapa atribut perilaku penggunaan layanan secara langsung, penelitian ini melakukan *feature engineering* untuk membentuk fitur yang merepresentasikan pola penggunaan data, komunikasi, nilai pelanggan, dan risiko *churn*.

3.2. Feature Engineering

Feature engineering dilakukan untuk membentuk fitur yang merepresentasikan perilaku pelanggan berdasarkan penggunaan layanan telekomunikasi. Proses ini meliputi pembentukan fitur perilaku pelanggan serta fitur turunan (*interaction features*) yang digunakan sebagai input model *machine learning*. Selain itu, lima kolom kategorikal dikodekan menggunakan *Label Encoding* sehingga total fitur input model berjumlah 21 fitur. Fitur perilaku pelanggan ditunjukkan pada Tabel 2, sedangkan fitur turunan utama ditunjukkan pada Tabel 3.

Berdasarkan Tabel 2, fitur perilaku digunakan untuk menggambarkan kebiasaan utama pelanggan dalam menggunakan layanan telekomunikasi. Penggunaan data dan *streaming* dapat menunjukkan kebutuhan paket internet, durasi panggilan dapat menunjukkan kebutuhan paket telepon, sedangkan potensi keluhan dapat menjadi indikator kebutuhan penanganan pelanggan melalui paket retensi. Dengan adanya fitur tersebut, model dapat mempelajari perbedaan karakteristik pelanggan sebelum menentukan kelas rekomendasi paket.

Tabel 2. Fitur perilaku pelanggan

Fitur	Keterangan
pct_video_usage	Merepresentasikan tingkat penggunaan layanan streaming pelanggan
avg_data_usage_gb	Menggambarkan estimasi intensitas penggunaan data internet
avg_call_duration	Merepresentasikan intensitas komunikasi pelanggan
device_brand	Menggambarkan segmentasi perangkat pelanggan berdasarkan tingkat penggunaan layanan
complaint_count	Merepresentasikan tingkat potensi keluhan pelanggan terhadap layanan

Berdasarkan Tabel 3, fitur turunan digunakan untuk memperkuat representasi perilaku pelanggan melalui hubungan beberapa atribut dasar, seperti penggunaan layanan, pengeluaran, aktivitas komunikasi, lama berlangganan, dan risiko *churn*. Informasi tersebut membantu model membedakan karakteristik pelanggan secara lebih menyeluruh sehingga rekomendasi paket dapat disesuaikan dengan profil penggunaan masing-masing pelanggan.

Tabel 3. Fitur Turunan dan Interaction Features

Fitur	Keterangan
recency	Mengukur kedekatan pelanggan terhadap aktivitas keluhan layanan
data_intensity	Menggambarkan tingkat konsumsi data terhadap pengeluaran pelanggan
communication_intensity	Merepresentasikan intensitas aktivitas komunikasi pelanggan
freq_x_monetary	Menggambarkan hubungan antara frekuensi penggunaan dan nilai pengeluaran pelanggan
loyalty_score	Mengukur tingkat loyalitas pelanggan berdasarkan tenure, CLTV, dan churn score

3.3. Pembentukan Target Rekomendasi

Label target ditentukan menggunakan pendekatan *rule-based* karena dataset asli tidak menyediakan label rekomendasi produk secara langsung. Penentuan target dilakukan berdasarkan beberapa karakteristik perilaku pelanggan, seperti tingkat risiko *churn*, pengeluaran bulanan, penggunaan layanan *streaming*, jenis layanan, dan lama berlangganan (*tenure*). Tabel 4 menunjukkan kelas target rekomendasi beserta karakteristik utamanya.

Tabel 4. Kelas Target Rekomendasi

No	Kelas	Kondisi Utama
1.	Paket Retensi	Pelanggan dengan risiko churn tinggi
2.	Paket Data Premium	Pelanggan dengan penggunaan data dan pengeluaran tinggi
3.	Paket Keluarga/Kombo	Pelanggan dengan layanan keluarga dan penggunaan multi layanan
4.	Paket Telepon	Pelanggan yang dominan menggunakan layanan telepon
5.	Paket Kuota Besar	Pelanggan dengan aktivitas internet dan streaming tinggi
6.	Paket Pemula	Pelanggan baru dengan tingkat penggunaan layanan rendah

3.4. Pelatihan Model Random Forest

Random Forest membangun sejumlah pohon keputusan secara paralel dan menentukan prediksi akhir melalui mekanisme *majority voting*. Secara sistematis, prediksi akhir dapat dirumuskan pada Persamaan (1):

$$\hat{y} = \text{mode}(h_1(x), h_2(x), \dots, h_n(x)) \quad (1)$$

Di mana:

- $h_i(x)$ merupakan hasil prediksi dari pohon ke- i terhadap data input x
- n adalah jumlah pohon yang digunakan dalam model.

- Fungsi *mode* digunakan untuk menentukan kelas yang paling sering muncul sebagai hasil akhir prediksi

Selain prediksi kelas, model menghasilkan probabilitas per kelas yang digunakan untuk menyusun rekomendasi *Top-K*. *Probabilitas* suatu kelas dihitung berdasarkan proporsi pohon yang memprediksi kelas tersebut dan dihitung dengan persamaan (2):

$$P(kelas) = \frac{\text{jumlah pohon yang memprediksi kelas}}{\text{jumlah seluruh pohon}} \quad (2)$$

Sebagai contoh, jika dari 3 pohon diperoleh prediksi (Paket Retensi, Paket Retensi, Paket Kuota Besar), maka prediksi akhir adalah Paket Retensi dengan probabilitas $2/3 = 0,67$ (67%). Pada model ini yang menggunakan 300 pohon, probabilitas dihitung dari proporsi seluruh pohon. Hasil prediksi model pada enam data uji ditunjukkan pada Tabel 5.

Tabel 5. Hasil Prediksi Model Random Forest pada Data Uji

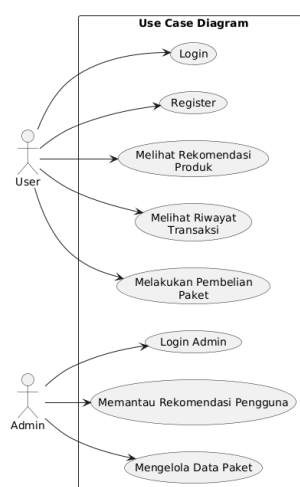
Data	Label Aktual	Prediksi model	Probabilitas
Pelanggan 1	Paket Data Premium	Paket Data Premium	99,2%
Pelanggan 2	Paket Keluarga/Kombo	Paket Keluarga/Kombo	85.97%
Pelanggan 3	Paket Kuota Besar	Paket Kuota Besar	99.91%
Pelanggan 4	Paket Pemula	Paket Pemula	100.00%
Pelanggan 5	Paket Retensi	Paket Retensi	100.00%
Pelanggan 6	Paket Telepon	Paket Telepon	100.00%

3.5. Implementasi Dashboard

Tahap implementasi *dashboard* dilakukan untuk menyajikan hasil rekomendasi model *Random Forest* dalam bentuk antarmuka web yang dapat diakses oleh pengguna. Sistem dikembangkan menggunakan React.js pada sisi *frontend*, FastAPI pada sisi *backend*, PostgreSQL sebagai *database*. Model *Random Forest* disimpan dalam format .pkl dan dimuat langsung oleh *backend* saat aplikasi dijalankan.

3.5.1. Use Case Diagram

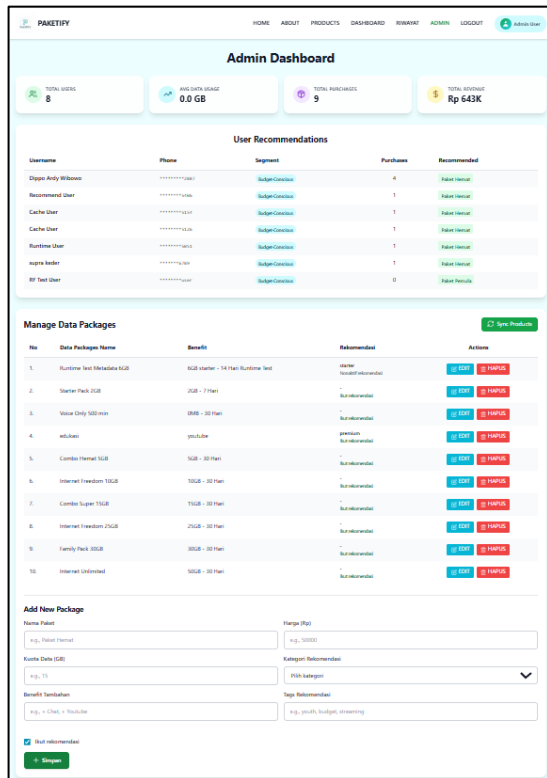
Alur interaksi pengguna dengan sistem ditunjukkan pada Gambar 2 *use case diagram*. Sistem memiliki dua peran pengguna yaitu *admin* dan *user*. Admin memiliki akses untuk mengelola data produk, melihat user, serta memantau hasil rekomendasi. *User* dapat melakukan login, melihat rekomendasi paket telekomunikasi yang dipersonalisasi, serta melihat riwayat pembelian.



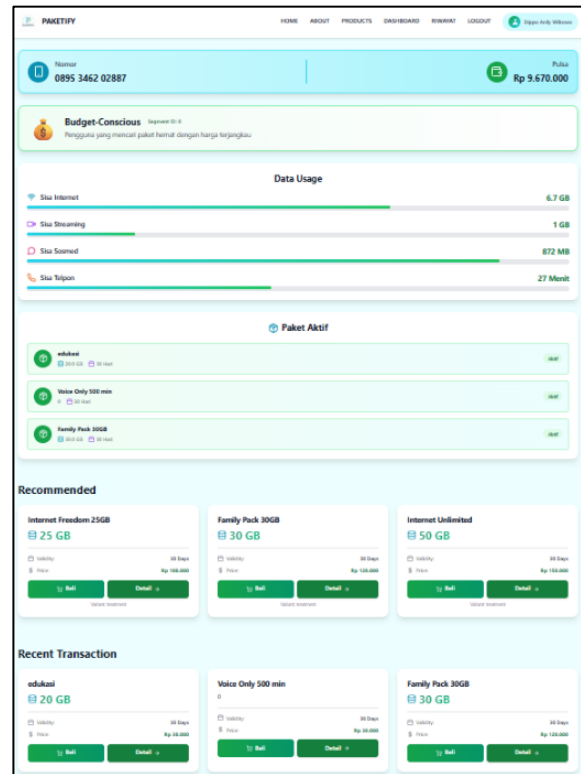
Gambar 2. Use Case Diagram

3.5.2. Dashboard Admin

Dashboard admin yang ditunjukkan pada Gambar 3. berfungsi untuk melihat ringkasan data sistem, statistik pengguna, data produk, serta hasil rekomendasi pengguna. Halaman ini membantu admin memantau keluaran sistem rekomendasi dan memastikan data produk yang ditampilkan kepada pengguna tetap sesuai dengan katalog yang tersedia.



Gambar 3. Dashboard Admin



Gambar 4. Dashboard Rekomendasi

3.5.3. Dashboard Rekomendasi

Dashboard rekomendasi yang ditunjukkan pada Gambar 4. untuk *user* yang berfungsi untuk menampilkan paket telekomunikasi yang dipersonalisasi berdasarkan profil dan perilaku pelanggan. Halaman ini menampilkan daftar paket rekomendasi beserta informasi utama seperti nama paket, harga, kuota. Berbeda dengan dashboard admin yang berfokus pada pemantauan data sistem, dashboard rekomendasi *user* berfokus pada penyajian paket yang relevan agar pengguna dapat memilih layanan yang sesuai dengan kebutuhannya.

3.6. Pengujian Sistem

Tahap pengujian sistem dilakukan untuk memastikan rekomendasi produk telekomunikasi dapat berjalan sesuai dengan kebutuhan yang telah dirancang. Pengujian pada penelitian ini meliputi evaluasi model *Machine Learning* menggunakan metrik akurasi untuk mengukur kinerja rekomendasi yang dihasilkan. Selain itu, dilakukan pengujian fungsional pada fitur utama aplikasi seperti proses login, pengambilan data pelanggan, proses prediksi menggunakan model *Random Forest*, serta tampilan hasil rekomendasi paket telekomunikasi pada dashboard aplikasi.

3.6.1. Pengujian Model

Pengujian dilakukan pada 1.409 data uji. Metrik yang digunakan adalah metrik *accuracy*. Karena penelitian ini menggunakan klasifikasi multi-kelas dengan enam kelas rekomendasi, nilai *accuracy* dihitung menggunakan Persamaan (3):

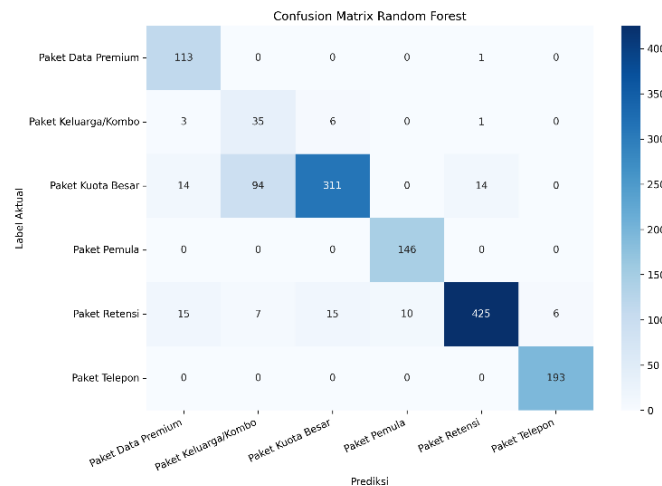
$$Accuracy = \frac{\text{Jumlah Prediksi Benar}}{\text{Jumlah Seluruh Data Uji}} \quad (3)$$

Maka nilai *accuracy* dapat dihitung sebagai berikut :

$$Accuracy = \frac{1.223}{1.409} = 0,86799 = 86,8\%$$

Hasil pengujian menunjukkan bahwa model memperoleh *accuracy* sebesar 86,8%. Nilai ini menunjukkan bahwa sebagian besar kelas rekomendasi berhasil diprediksi dengan benar pada data uji.

Berdasarkan *confusion matrix* pada Gambar 5, kesalahan prediksi paling menonjol terjadi pada kelas Paket Kuota Besar yang diprediksi sebagai Paket Keluarga/Kombo sebanyak 94 data. Hal ini diduga terjadi karena kedua kelas memiliki karakteristik yang saling beririsan, terutama pada pelanggan yang aktif menggunakan layanan internet, memiliki aktivitas streaming, dan nilai pengeluaran yang relatif tinggi. Selain itu, label target pada penelitian ini dibentuk menggunakan aturan *rule-based*, sehingga beberapa kondisi pembentukan kelas dapat memiliki kemiripan antar label. Kondisi tersebut dapat menyebabkan model lebih sulit membedakan batas antara Paket Kuota Besar dan Paket Keluarga/Kombo, serta berdampak pada rendahnya *precision* kelas Paket Keluarga/Kombo pada *classification report*. *Precision* merupakan metrik yang menunjukkan ketepatan prediksi model pada suatu kelas, yaitu seberapa banyak data yang benar-benar sesuai dari seluruh data yang diprediksi sebagai kelas tersebut.



Gambar 5. Confusion Matrix Model Random Forest

Tabel 6. Classification Report

Kelas	Precision	Recall	F1 - Score	Support
Paket Data Premium	0,78	0,99	0,87	114
Paket Keluarga/Kombo	0,26	0,78	0,39	45
Paket Kuota Besar	0,94	0,72	0,81	433
Paket Pemula	0,94	1,00	0,97	146
Paket Retensi	0,96	0,89	0,92	478
Paket Telepon	0,97	1,00	0,98	193
Accuracy			0,87	1409
Macro	0,81	0,90	0,82	1409
Weighted avg	0,92	0,87	0,88	1409

Selain itu, hasil evaluasi model juga ditunjukkan melalui *classification report* pada Tabel 6. Nilai *F1-score* digunakan untuk melihat keseimbangan antara ketepatan prediksi model dan kemampuannya dalam mengenali data pada masing-masing kelas rekomendasi. Berdasarkan hasil tersebut, model menunjukkan performa yang cukup baik pada sebagian besar kelas rekomendasi. Kelas Paket Telepon memperoleh hasil terbaik dengan *F1-score* sebesar 0,98, diikuti Paket Pemula

sebesar 0,97 dan Paket Retensi sebesar 0,92. Sementara itu, kelas Paket Keluarga/Kombo memiliki performa paling rendah dengan *F1-score* sebesar 0,39 karena jumlah data uji pada kelas tersebut hanya 45 data. Secara keseluruhan, hasil ini menunjukkan bahwa model telah mampu mengklasifikasikan rekomendasi paket dengan baik, meskipun performa pada kelas dengan jumlah data lebih sedikit masih perlu ditingkatkan.

3.6.2. Pengujian Black Box

Pengujian *blackbox* dilakukan untuk memastikan setiap fitur berjalan sesuai kebutuhan berdasarkan kesesuaian *input* dan *output*. Hasil pengujian ditunjukkan pada Tabel 7.

Tabel 7. Hasil Black Box Testing

No	Fitur/Halaman yang Diuji	Role	Skenario Pengujian	Hasil yang Diharapkan	Hasil Aktual	Status
1.	Login	User	Input nomor hp dan password yang valid	Berhasil masuk ke dashboard user	Sesuai	Berhasil
2.	Dashboard	User	Mengakses halaman dashboard rekomendasi	Rekomendasi paket telekomunikasi tampil	Sesuai	Berhasil
3.	Detail Paket	User	Memilih salah satu paket rekomendasi	Detail paket seperti nama, harga, dan kuota tampil	Sesuai	Berhasil
4.	Login	Admin	Input nomor hp dan password yang valid	Berhasil masuk ke dashboard admin	Sesuai	Berhasil
5	Monitoring Rekomendasi Pengguna	Admin	Mengakses dashboard admin	Ringkasan pengguna, jumlah pembelian, dan produk rekomendasi tampil	Sesuai	Berhasil
6.	Data Paket	Admin	Mengakses halaman data paket	Data paket telekomunikasi tampil dan dapat dikelola	Sesuai	Berhasil

Berdasarkan hasil pengujian *blackbox* fitur utama pada *role user* dan admin berjalan sesuai dengan hasil yang diharapkan. Pengujian menunjukkan bahwa sistem memenuhi kebutuhan fungsional yang ditentukan dan dapat digunakan dengan baik oleh pengguna.

4. KESIMPULAN

Kesimpulan berdasarkan hasil penelitian dan pembahasan, dapat disimpulkan bahwa:

- 1) Sistem rekomendasi produk telekomunikasi berbasis web berhasil dikembangkan untuk membantu personalisasi rekomendasi paket berdasarkan perilaku pelanggan.
- 2) Algoritma *Random Forest* mampu menghasilkan performa yang baik dengan tingkat *accuracy* sebesar 86,8% pada 1.409 data uji.
- 3) Implementasi dashboard berbasis React.js mampu menampilkan hasil rekomendasi paket telekomunikasi kepada *user* secara informatif.
- 4) Hasil pengujian sistem menunjukkan bahwa fitur utama berjalan dengan baik, dibuktikan melalui pengujian fungsional (*black box*).
- 5) Penelitian selanjutnya disarankan untuk menggunakan data pelanggan nyata serta membandingkan *Random Forest* dengan algoritma rekomendasi lainnya.

DAFTAR PUSTAKA

- [1] E. Jlassi, A. Chaabouni, and M. Triki, "Impact of recommendation systems on AI-enabled customer experience: Mediator role of perceived usefulness and perceived trust," *Journal of Telecommunications and the Digital Economy*, vol. 13, no. 1, pp. 160–191, Mar. 2025, doi: 10.18080/jtde.v13n1.1137.

- [2] I. Sulistyowati, D. Setyawan, and N. Zuhdi Arzaqi, "Sistem Rekomendasi Pemilihan Layanan Data Dan Internet Segmen Bges Dengan Metode Knowledge Base Berbasis Web Pada Pt. Telkom Surakarta," 2024.
- [3] F. Lestari, D. Hartanti, and H. Hasanah, "Pemodelan Sistem Rekomendasi Pemilihan Paket Internet IndiHome Menggunakan Metode Knowledge," 2025.
- [4] Y. H. Alfaifi, "Recommender Systems Applications: Data Sources, Features, and Challenges," Oct. 01, 2024, *Multidisciplinary Digital Publishing Institute (MDPI)*. doi: 10.3390/info15100660.
- [5] N. Guan, "Machine learning algorithms in the development of recommender systems," *Applied and Computational Engineering*, vol. 51, no. 1, pp. 119–128, Mar. 2024, doi: 10.54254/2755-2721/51/20241189.
- [6] T. Agustina, M. Masrizal, and I. Irmayanti, "Performance Analysis of Random Forest Algorithm for Network Anomaly Detection using Feature Selection," *sinkron*, vol. 8, no. 2, Apr. 2024, doi: 10.33395/sinkron.v8i2.13625.
- [7] S. Amutha and R. Vikram Surya, "An Improved Product Recommender System Using Collaborative Filtering and a Comparative Study of ML Algorithms," *Cybernetics and Information Technologies*, vol. 23, no. 4, pp. 51–62, Nov. 2023, doi: 10.2478/cait-2023-0035.
- [8] R. Jonathan and Suprihadi, "Development of Front-End Web Applications Utilizing Single Page Application Framework and React.js Library," *International Journal Software Engineering and Computer Science (IJSECS)*, vol. 3, no. 3, pp. 529–536, Dec. 2023, doi: 10.35870/ijsecs.v3i3.1943.
- [9] S. V. Salunke and A. Ouda, "A Performance Benchmark for the PostgreSQL and MySQL Databases," Oct. 01, 2024, *Multidisciplinary Digital Publishing Institute (MDPI)*. doi: 10.3390/fi16100382.
- [10] D. D. Le and H. W. Lauw, "Efficient Retrieval of Matrix Factorization-Based Top-k Recommendations: A Survey of Recent Approaches," 2021.
- [11] A. Nurseptaji and Y. Ramdhani, "Penerapan Metodologi Waterfall Pada Rancangan Sistem Informasi Perpustakaan," *Jurnal Device*, vol. 11, pp. 1–12, 2021.