

Rancang Bangun Dashboard LeadsUp Berbasis *Predictive Lead Scoring* untuk Penentuan Target Nasabah

Sendi Ainur Rokhimah¹, Astri Charolina^{*2}, Firdhaus Hari Saputro Al Haris³

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

Email: ¹ainurrsendi@gmail.com, ^{*2}astricharolina@usahidsolo.ac.id, ³edoz2003@gmail.com

(Naskah masuk: 1 Juni 2026, diterima untuk diterbitkan: 2 Juli 2023)

Abstrak: Latar belakang masalah sektor perbankan masih menggunakan sistem pemasaran konvensional yang bersifat acak sehingga menyebabkan rendahnya konversi deposito berjangka dan tingginya biaya operasional tim sales. Tujuan penelitian merancang dan membangun dashboard LeadsUp yang mengintegrasikan hasil predictive lead scoring untuk menentukan target nasabah potensial. Metode penelitian dilaksanakan melalui tiga tahapan, yaitu identifikasi masalah untuk mengurai ketidakefektifan outbound call perbankan, pengumpulan data dan literatur, menggunakan Bank Marketing Dataset dari UCI Repository serta pengembangan sistem dengan menerapkan metode waterfall. Hasil penelitian berupa dashboard yang dibangun menggunakan stack teknologi React.js, Express.js, dan Supabase, yang menyediakan fitur dashboard admin (manajemen sales, distribusi leads dan validasi kredit) serta dashboard sales (pemantauan kerja individu, daftar leads dan riwayat kredit). Sistem divalidasi melalui 3 metode pengujian, yaitu Black Box Testing untuk verifikasi fungsionalitas antarmuka, Postman Testing untuk menjamin stabilitas integrasi API, serta Pengujian Model berbasis algoritma Random Forest yang menghasilkan tingkat akurasi klasifikasi prospek nasabah sebesar 87%. Dengan demikian, LeadsUp mampu membantu perbankan dalam meningkatkan strategi pemasaran melalui penentuan target nasabah berbasis data.

Kata Kunci – Dashboard; Deposito Berjangka; Predictive Lead Scoring; Random Forest; Waterfall

Design and Development of LeadsUp Dashboard Based on Predictive Lead Scoring for Customer Targeting

Abstract: The background of the problem is that the banking sector still uses a conventional, random marketing system, resulting in low time deposit conversions and high operational costs for the sales team. The purpose of this research is to design and build a LeadsUp dashboard that integrates predictive lead scoring results to determine potential customer targets. The research method was implemented through three stages: problem identification to unravel the ineffectiveness of banking outbound calls, data and literature collection, using the Bank Marketing Dataset from the UCI Repository, and system development by applying the waterfall method. The research results are a dashboard built using the React.js, Express.js, and Supabase technology stack, which provides admin dashboard features (sales management, lead distribution, and credit validation) and sales dashboards (individual work monitoring, lead lists, and credit history). The system was validated through three testing methods: Black Box Testing to verify interface functionality, Postman Testing to ensure the stability of API integration, and Random Forest algorithm-based Model Testing, which resulted in an 87% accuracy rate in customer prospect classification. Thus, LeadsUp is able to help banks improve their marketing strategies through data-based customer targeting.

Keywords – Dashboard; Time Deposit; Predictive Lead Scoring; Random Forest; Waterfall

1. PENDAHULUAN

Sektor perbankan adalah bagian penting dalam mengembangkan ekonomi modern yang memainkan fungsi yang sangat penting dalam memajukan ekonomi suatu negara, sehingga perbankan perlu secara konsisten melakukan inovasi dengan meningkatkan kualitas dan kuantitasnya [1]. Perkembangan teknologi digital menuntut perbankan untuk meninggalkan metode pemasaran konvensional yang bersifat satu arah dan beralih ke pendekatan yang lebih personal dan berbasis data [2].

Tim penjualan sering menemui kesulitan dalam mengutamakan prospek karena metode penawaran yang manual tanpa bantuan sistem analitik. Proses komunikasi dalam aktivitas penjualan membutuhkan pengeluaran yang tinggi, baik dari aspek waktu maupun materi. Untuk itu, perbankan perlu memusatkan fokus pada pengembangan prospek yang paling cocok dan menunjukkan tingkat ketertarikan yang tinggi demi mempertahankan tingkat pengembalian investasi yang menguntungkan [3]. Salah satu pendekatan yang saat ini banyak digunakan adalah implementasi sistem rekomendasi yang berbasis kecerdasan buatan untuk menyesuaikan tawaran produk perbankan, termasuk deposito berjangka [4]. Deposito berjangka didefinisikan sebagai simpanan yang penarikannya hanya dapat dilakukan pada waktu tertentu berdasarkan perjanjian antara nasabah bank, sesuai dengan Undang-Undang Perbankan Nomor 10 Tahun 1998 [5].

Sebagai solusi atas permasalahan tersebut, pendekatan berbasis data melalui *predictive lead scoring* hadir untuk mengidentifikasi peluang konversi nasabah secara akurat dengan memanfaatkan data historis. *Predictive lead scoring* adalah metodologi yang memanfaatkan *machine learning* untuk menganalisis data sehingga memberikan nilai numerik kepada *leads* berdasarkan kemungkinan mereka untuk berkonversi menjadi pelanggan. Sistem ini mengidentifikasi pola-pola dari data historis yang mungkin terlewatkan oleh analisis manusia [6].

Pada penelitian yang telah dilakukan sebelumnya, penerapan *predictive lead scoring* terbukti mampu meningkatkan efisiensi konversi prospek dalam bidang penjualan B2B. Kemudian dari penelitian tersebut diimplementasikan ke studi kasus ini dalam konteks perbankan untuk produk deposito berjangka [7]. Selanjutnya penerapan *machine learning* dapat merubah cara penilaian *lead* yang sebelumnya bersifat subjektif menjadi keputusan berbasis data. Penelitian ini menyoroti bahwa integrasi AI dalam sistem manajemen hubungan pelanggan (CRM) dapat mengenali pola perilaku pelanggan yang rumit yang sering kali tidak terdeteksi oleh analisis manual [8]. Selain itu, penerapan *machine learning* dapat mengatasi keterbatasan metode pemasaran konvensional dalam memprediksi *telemarketing* untuk produk deposito berjangka. Penelitian ini menggunakan lima algoritma *machine learning*, yaitu *Logistic Regression*, *Decision Tree*, *Random Forest*, *SVM*, dan *Gaussian Naive Bayes* dan hasilnya *Logistic Regression* adalah model terbaik di antara kelima algoritma tersebut [9]. Walaupun demikian, sebagian besar penelitian tersebut masih terfokus pada pengembangan model prediksi dan belum mengimplementasikannya dalam bentuk sistem operasional yang siap pakai oleh tim lapangan.

Penelitian ini bertujuan untuk menutup celah tersebut dengan merancang dan membangun *dashboard* dengan menggabungkan hasil prediksi skor prioritas secara langsung menggunakan algoritma *Random Forest*. *Dashboard* adalah tampilan visual informasi penting yang hanya berisi data yang diperlukan untuk mencapai tujuan tertentu [10]. *Dashboard* ini dikembangkan dengan menggunakan *stack* React.js, Express.js, dan Supabase sebagai pengelola basis data. Inovasi sistem ini terletak pada kemampuannya yang memungkinkan tim *sales* untuk menyaring nasabah menurut kategori prioritas dan langsung menghubungi mereka melalui fitur *WhatsApp* yang terintegrasi. Dengan rancang bangun ini, diharapkan LeadsUp mampu berfungsi sebagai alat bantu yang efektif untuk sektor perbankan dalam mengidentifikasi target nasabah potensial secara tepat dan efisien.

2. METODE PENELITIAN

Metode Penelitian ini menjelaskan langkah-langkah dalam merancang dan membangun *dashboard* LeadsUp. Penelitian ini dibagi menjadi tiga tahapan untuk menyelesaikan hambatan operasional bank yang ada. Pertama, tahapan identifikasi masalah untuk penentuan arah penelitian. Kedua, pengumpulan data dan literatur untuk mengumpulkan data mentah yang akan dilatih dan mencari referensi pendukung sistem. Ketiga, tahapan pengembangan sistem menggunakan metode *waterfall* untuk membangun sistem secara berurutan.

2.1. Tahapan Identifikasi Masalah

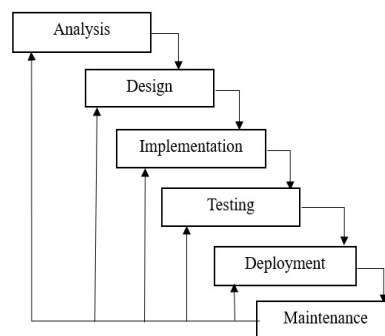
Penelitian ini dimulai dari adanya ketidakefektifan dalam *outbound call* perbankan yang masih melakukan seleksi nasabah secara acak. Situasi ini mengakibatkan rendahnya angka konversi dan tingginya biaya operasional karena tim penjualan menghabiskan waktu pada nasabah yang tidak memiliki prospek. Nilai strategis yang ditawarkan oleh penelitian ini adalah penerapan *predictive lead scoring* menggunakan algoritma *Random Forest* untuk memprediksi potensi nasabah. Solusi ini mengubah proses pengambilan keputusan yang dulu bersifat subjektif menjadi berbasis data.

2.2. Tahapan Pengumpulan Data dan Literatur

Tahap ini dilakukan untuk menghimpun dataset serta referensi yang diperlukan yang mendukung keberlanjutan penelitian. *Dataset* yang digunakan adalah *Bank Marketing Dataset* yang diperoleh dari *UCI Machine Learning Repository*. Data tersebut berperan sebagai basis pelatihan model untuk menghasilkan skor prediksi prioritas. Setelah dataset diperoleh, dilakukan studi literatur dilakukan dengan mengkaji penelitian terdahulu mengenai algoritma *machine learning* seperti random forest serta mempelajari arsitektur teknologi pendukung seperti React.js, Express.js, dan manajemen basis data pada supabase guna menjamin kesesuaian pengembangan sistem.

2.3. Tahapan Pengembangan Sistem

Metode yang digunakan dalam pengembangan sistem ini menggunakan model *waterfall*. Metode ini mengharuskan pengerjaannya dilakukan secara berurutan dan sekuensial [11]. Pendekatan *waterfall* yang sistematis dan bertahap akan mempermudah proses pengembangan mulai dari analisis kebutuhan, *design*, implementasi, *testing*, *deployment* dan yang terakhir *maintenance*. Metode *waterfall* dapat dilihat pada Gambar 1.



Gambar 1. Metode Waterfall [12]

2.3.1. Tahap Analysis

Pada tahap ini, dilakukan identifikasi mendalam mengenai kebutuhan sistem baik dari sisi fungsional maupun non-fungsional. Kebutuhan fungsional difokuskan pada kemampuan *dashboard* dalam menyajikan skor prediksi nasabah. Sementara itu, kebutuhan non fungsional mencakup penentuan arsitektur teknologi serta pemilihan layanan *cloud* untuk memastikan sistem dapat beroperasi dengan stabil dan responsif.

2.3.2. Tahap Design

Tahap ini difokuskan untuk menerjemahkan kebutuhan analisis ke dalam desain teknis sebelum diimplementasikan dalam kode program. Rancangan fungsionalitas dan interaksi pengguna menggunakan *usecase diagram* yang berbasis *Unified Modeling Language* (UML).

2.3.3. Tahap Implementation

Tahap ini merupakan realisasi dari desain ke dalam kode program. Pembangunan sistem mencakup empat komponen. Pertama, konfigurasi basis data menggunakan Supabase. Kedua,

pembuatan alur pemrosesan data pada sisi *backend* dengan Express.js dan node.js. Ketiga, pembangunan antarmuka *frontend* menggunakan React.js dan tailwind CSS. Keempat, *machine learning* memanfaatkan algoritma *Random Forest* untuk memprediksi konversi dan mengidentifikasi label prioritas, dengan tujuan menetapkan target nasabah yang paling potensial.

2.3.4. Tahap Testing

Tahap ini bertujuan memastikan bahwa semua komponen sistem memenuhi spesifikasi dan standar kualitas. Proses verifikasi dilakukan dengan tiga metode pengujian yang berbeda. Pertama, pengujian fungsionalitas antarmuka menggunakan *Black Box Testing*. Kedua, pengujian integrasi data dan *endpoint* API dengan Postman. Ketiga, pengujian performa model data prediktif dengan algoritma *Random Forest*.

2.3.5. Tahap Deployment

Tahap ini mencakup proses *deployment* sistem ke server produksi guna memastikan aplikasi dapat diakses secara publik melalui jaringan internet. Seluruh operasional didukung oleh layanan vercel untuk *frontend* serta railway untuk *backend* dan *machine learning*.

2.3.6. Tahap Maintenance

Tahap akhir ini difokuskan pada pemeliharaan sistem pasca implementasi melalui pemantauan performa API, perbaikan bug, serta penyesuaian fitur secara berkala guna menjamin keberlanjutan operasional *dashboard* sesuai kebutuhan pengguna.

3. HASIL DAN PEMBAHASAN

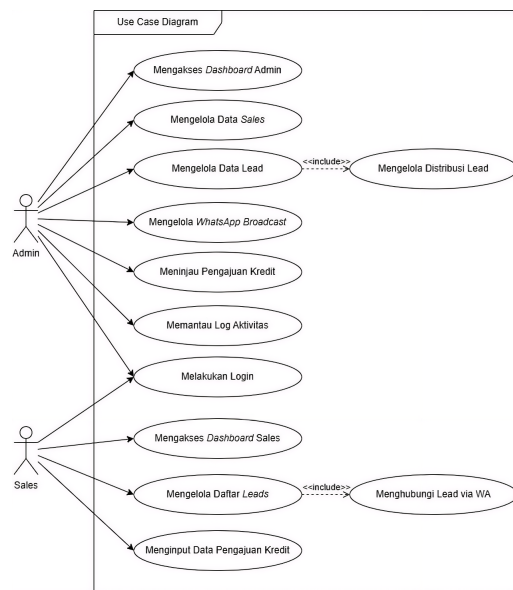
Bab ini menjelaskan hasil pengembangan *dashboard* LeadsUp yang terdiri dari tiga bagian utama. Pertama, implementasi sistem yang menjelaskan realisasi arsitektur teknologi. Kedua, *random forest classifier* yang membahas model prediksi. Ketiga, pengujian sistem yang menawarkan validasi kinerja fitur dan pengujian akurasi model data.

3.1. Implementasi Sistem

Pembahasan tentang hasil penelitian rancang bangun *dashboard* LeadsUp ditampilkan melalui penjelasan teknis mengenai pelaksanaan perangkat lunak. Proses implementasi sistem dibagi menjadi tiga. Pertama, visualisasi fungsi sistem dengan pemetaan interaksi antara aktor dalam *usecase diagram*. Kedua, struktur penyimpanan data pada basis data supabase. Ketiga, realisasi fisik tampilan *dashboard* yang membedakan hak akses serta fitur fungsional untuk pengguna admin dan *sales*. Adapun rincian dari setiap tahapan implementasi tersebut akan dijelaskan sebagai berikut.

3.1.1. Usecase diagram

Rancangan fungsionalitas sistem LeadsUp digambarkan melalui *usecase diagram* untuk mendefinisikan interaksi antara aktor dengan fitur-fitur utama. Berdasarkan rancangan pada Gambar 2, terdapat dua aktor utama yaitu admin dan *sales*.



Gambar 2. Use Case Diagram

Aktor admin memiliki tanggung jawab penuh dalam mengelola operasional dan memantau sistem. Tugas utama admin meliputi pengaturan data *sales*, mengupload data lead melalui file CSV untuk proses *scoring* otomatis, menjalankan pesan massal *WhatsApp* serta memvalidasi pengajuan kredit. Di sisi lain, aktor *sales* dapat mengawasi indikator kinerja utama serta menyaring daftar *leads* berdasarkan prioritas dari hasil prediksi. Keefektifan komunikasi diperkuat oleh integrasi fitur *WhatsApp Direct* untuk langsung menghubungi *leads*. *Sales* juga bertanggung jawab dalam pengiriman formulir aplikasi kredit dari *leads* yang tertarik.

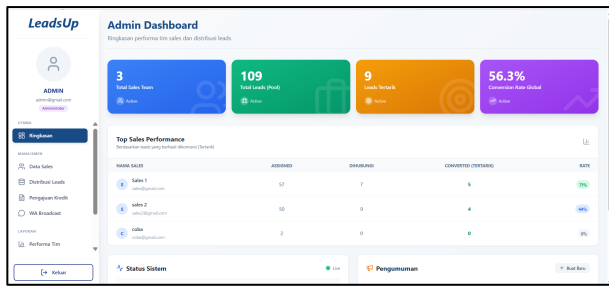
3.1.2. Implementasi Database (Supabase)

Implementasi basis data pada sistem LeadsUp menggunakan supabase sebagai infrastruktur *Cloud PostgreSQL* struktur basis data dibuat untuk menyimpan data penting yang mendukung fungsionalitas *predictive scoring* dan manajemen tugas *sales*. Inti dari *database* ini terletak pada tabel *leads* yang berisi informasi strategis mengenai pelanggan. Operasional sistem ini didukung oleh tabel *user* untuk pengelolaan wewenang, tabel *credits* untuk merekam spesifikasi pengajuan pelanggan secara rinci, serta tabel-tabel pendukung lainnya seperti *audit logs*, *announcements*, dan *campaigns* yang berfungsi untuk memantau keamanan serta distribusi informasi internal.

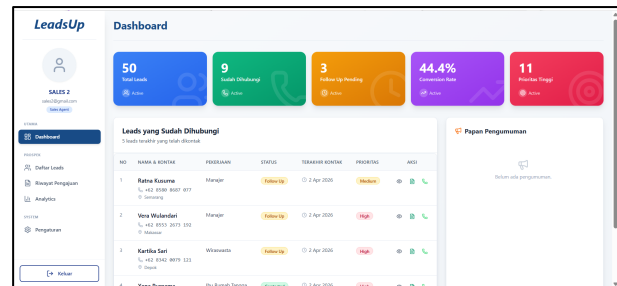
3.1.3. Implementasi Antarmuka Dashboard

Bagian ini menyajikan tampilan antarmuka dari *dashboard* LeadsUp yang telah terintegrasi dengan model prediksi. Tampilan sistem dibagi menjadi dua berdasarkan peran pengguna yaitu tampilan *dashboard* admin dan *sales*. *Dashboard* admin adalah pusat kontrol yang dirancang untuk memberikan visibilitas penuh bagi admin dalam memantau kinerja pemasaran secara langsung. tampilan ini mengintegrasikan data dari supabase dan menampilkannya dalam bentuk panel statistik yang informatif, mencakup jumlah tim *sales*, total *leads* yang tersedia, hingga persentase konversi *global*. Tampilan *dashboard* admin disajikan pada Gambar 3.

Dashboard sales dirancang untuk meningkatkan efisiensi tim lapangan dalam menjalankan strategi pemasaran. Tampilan ini menyajikan informasi mengenai *sales*, termasuk total tugas yang diterima dan jumlah *leads* yang telah dihubungi. Fitur utama dari halaman ini menampilkan daftar *leads* terakhir dihubungi yang terintegrasi langsung dengan hasil prediksi *random forest* dalam label prioritas. Tampilan *dashboard* sales disajikan pada Gambar 4.



Gambar 3. Dashboard Admin



Gambar 4. Dashboard Sales

3.2. Random Forest Classifier

Random Forest adalah *ensemble learning approach* yang populer karena *adaptabilitas* dan akurasi [13]. Algoritma ini menggunakan konsep *decision trees*, membangun kumpulan *decision trees* dan menggabungkan hasilnya untuk menghasilkan prediksi akhir [14]. *Decision Tree* merupakan suatu metode yang memecah kumpulan data menjadi beberapa bagian seperti cabang [15]. Hasil prediksi dari *Random Forest* diperoleh dari hasil terbanyak dari setiap pohon (*majority voting*) untuk klasifikasi, atau rata-rata untuk regresi [16].

Berikut merupakan rumus untuk mencari hasil prediksi *Random Forest* dapat dilihat pada Persamaan (1) [17]:

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

Keterangan :

$h_i(x)$: Hasil prediksi dari pohon ke- i terhadap data input x

n : Jumlah pohon yang digunakan dalam model.

$\hat{y}$: Nilai prediksi akhir yang ditentukan dari kelas yang paling banyak muncul (*majority voting*)

Probabilitas dalam *Random Forest Classifier* menggambarkan sejauh mana model memiliki kepercayaan terhadap suatu prediksi berdasarkan distribusi suara (*voting*) dari seluruh pohon keputusan. Angka probabilitas ini memiliki peranan penting di dalam sistem LeadsUp karena menjadi acuan utama untuk menetapkan label prioritas bagi nasabah. Probabilitas "Yes" adalah perbandingan antara banyaknya pohon keputusan yang memprediksi bahwa seorang nasabah akan melakukan konversi dibandingkan dengan total keseluruhan pohon dalam model. Secara matematis, rumus untuk menghitung probabilitas "Yes" dapat dituliskan pada Persamaan (2) [18]:

$$P(\text{yes}) = \frac{\sum_{i=1}^n [Tree_i = "yes"]}{n} \quad (2)$$

Keterangan :

$P(\text{yes})$: Nilai probabilitas nasabah akan mengambil produk.

n : Total jumlah pohon (*trees*) yang digunakan dalam model.

$[Tree_i = "yes"]$: Jumlah pohon yang memberikan prediksi "Yes".

Nilai probabilitas yang diperoleh dari model akan dipetakan ke dalam beberapa kategori prioritas. Tujuannya adalah untuk membantu tim lapangan dalam menetapkan seberapa mendesak tindakan yang perlu diambil terhadap masing-masing nasabah. Kriteria batas (*threshold*) dari kategori prioritas dalam sistem LeadsUp disajikan pada Tabel 1.

Tabel 1. Kategori Probabilitas

Range Probabilitas	Label Prioritas	Tindakan Tim Sales
0,70 – 1,00	<i>High Priority</i>	Fokus utama untuk segera dihubungi
0,40 – 0,69	<i>Medium Priority</i>	Dihubungi setelah daftar <i>high priority</i> selesai
0,00 – 0,39	<i>Low priority</i>	Nasabah dengan potensi rendah

Untuk memberikan gambaran tentang cara kerjanya, Tabel 2. Menyajikan simulasi perhitungan pada 10 sample data nasabah dengan memanfaatkan model *Random Forest* yang dikonfigurasi dengan 3 pohon keputusan.

Tabel 2. Perhitungan *Random Forest*

age	previous	campaign	Y	Tree 1	Tree 2	Tree 3	Prediksi Random Forest	Benar/Tidak	Probabilitas	Kategori probabilitas
56	0	1	No	no	no	yes	No	yes	33%	low
57	0	1	No	no	no	yes	No	yes	33%	low
37	0	1	No	yes	no	yes	Yes	no	67%	medium
40	0	1	No	yes	no	yes	Yes	no	67%	medium
56	0	1	No	no	no	yes	No	yes	33%	low
59	0	1	No	no	no	yes	No	yes	33%	low
41	0	1	No	no	no	yes	No	yes	33%	low
24	0	1	No	yes	no	yes	Yes	no	67%	medium
25	0	1	No	yes	no	yes	Yes	no	67%	medium
41	0	1	No	no	no	yes	No	yes	33%	low

Berdasarkan Tabel 2 di atas, dapat dijelaskan alur penentuan prioritas sebagai berikut:

Pada baris pertama, *Tree 1* dan *Tree 2* memprediksi 'no', sedangkan *Tree 3* memprediksi 'yes'. Karena label 'no' memiliki jumlah suara terbanyak, maka hasil akhir prediksi sistem adalah 'no'. Probabilitas dihitung berdasarkan rasio pohon yang memprediksi 'yes'. sehingga nilai probabilitasnya adalah = 33 %

$$P(\text{yes}) = \frac{1}{3} = 0.33 = 0.33 \cdot 100\% = 33\%$$

Berdasarkan aturan pada Tabel 1. nilai probabilitas 33% masuk ke dalam rentang 0,00 – 0.39, sehingga nasabah tersebut secara otomatis dikategorikan sebagai *Low Priority*.

Selanjutnya pada baris ketiga, *Tree 1* dan *Tree 3* memprediksi 'yes', sedangkan *Tree 2* memprediksi 'no'. Karena label 'yes' memiliki jumlah suara terbanyak, maka hasil akhir prediksi sistem adalah 'yes'. Probabilitas dihitung berdasarkan rasio pohon yang memprediksi 'yes'. sehingga nilai probabilitasnya adalah = 67 %

$$P(\text{yes}) = \frac{2}{3} = 0.67 = 0.67 \cdot 100\% = 67\%$$

Berdasarkan aturan pada Tabel 1. nilai probabilitas 67% masuk ke dalam rentang 0,40 – 0.69, sehingga nasabah tersebut secara otomatis dikategorikan sebagai *Medium Priority*.

3.3. Pengujian Sistem

Pengujian ini dilakukan melalui tiga metode utama, yaitu pengujian *Black Box*, pengujian API menggunakan Postman, dan pengujian performa model *Random Forest*. Rangkaian pengujian ini untuk memastikan bahwa sistem dashboard LeadsUp berjalan sesuai dengan spesifikasi fungsional yang telah ditetapkan.

3.3.1. Metode Black box Testing

Black box testing merupakan metode pengujian yang berfokus pada fungsionalitas perangkat lunak tanpa perlu memahami detail kode sumber didalamnya [19]. Metode ini akan menitikberatkan pada pengujian antarmuka pengguna, input yang diberikan, output dihasilkan oleh sistem [20]. Hasil pengujian menunjukkan bahwa semua elemen pada antarmuka berfungsi sesuai dengan skenario penggunaan yang diharapkan. Untuk memastikan bahwa sistem berfungsi

dengan baik, beberapa skenario pengujian yang mencakup semua fitur utama pada *dashboard* admin dan *sales*. Detail skenario pengujian beserta hasil validasinya disajikan dalam Tabel 3.

Tabel 3. Hasil *BlackBox Testing*

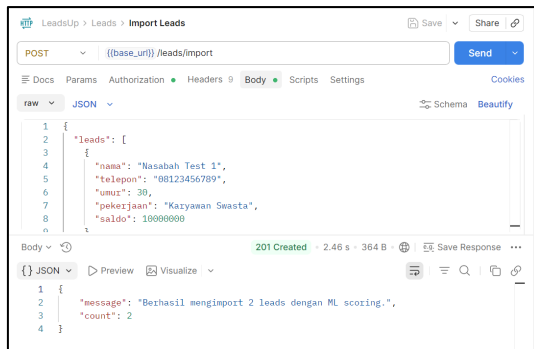
No.	Scenario Pengujian	Role	Langkah Pengujian	Hasil Yang Diharapkan	Hasil Pengujian	Kesimpulan
1.	Login Berhasil	admin /sales	Masukkan email dan <i>password</i> yang terdaftar dengan benar.	Berhasil masuk dan diarahkan ke <i>dashboard</i> sesuai <i>role</i> .	Sesuai	Berhasil
2.	Tambah Sales Baru	admin	Klik tombol "Tambah Sales Baru", isi form data sales, lalu simpan.	Sales baru muncul di tabel	Sesuai	Berhasil
3.	Distribusi Manual ke Sales	admin	Centang satu atau beberapa nasabah, pilih nama sales di <i>dropdown</i> lalu klik "Tugaskan".	Muncul notifikasi sukses	Sesuai	Berhasil
4.	Pengajuan Kredit	sales	Klik tombol hijau muda "Kredit".	Sistem mengarahkan user ke formulir pengajuan kredit nasabah tersebut.	Sesuai	Berhasil
5.	Hubungi Nasabah (WA)	sales	Klik ikon Telepon/ <i>WhatsApp</i> pada kolom Aksi.	Sistem membuka aplikasi <i>WhatsApp</i> dengan nomor nasabah yang sudah terisi otomatis.	Sesuai	Berhasil

3.3.2. Postman Testing

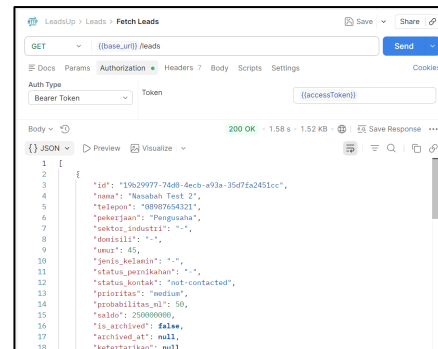
Postman merupakan aplikasi klien API untuk pengembang yang mempermudah pembuatan, distribusi, pengujian, dan dokumentasi API [21]. Postman digunakan untuk memastikan bahwa setiap *endpoint* API beroperasi sebagaimana mestinya dan memberikan respons yang diharapkan pada tahap pengembangan. Platform ini bertugas mengirimkan permintaan HTTP ke server Express.js untuk menguji *endpoint* API, serta memeriksa data respons yang diberikan oleh server [22]. Proses pengujian ini menjamin bahwa jalur komunikasi data antara *database* Supabase, model prediksi, dan antarmuka *dashboard* berfungsi dengan baik.

Pengujian dilakukan dengan mengirimkan informasi pelanggan dalam format JSON yang memuat parameter demografi dan finansial. Ketika permintaan diterima, *backend* secara otomatis mengalirkan data tersebut ke *ML service* yang mengoperasikan model *Random Forest*. Hasil pengujian menunjukkan status 201 Created dengan pesan 'Berhasil mengimport 2 leads dengan *ML scoring*'. Ini menunjukkan bahwa proses integrasi antara *backend* Node.js dan API ML yang berbasis Python telah berjalan dengan baik. Hasil pengujian beserta hasil validasinya disajikan dalam Gambar 5.

Pengujian dilanjutkan dengan mengirim permintaan menarik data dari *database*. Dari respon yang ditunjukkan nasabah yang baru saja diimpor kini sudah memiliki nilai pada kolom *prioritas* dan *probabilitas_ml*. Karena kolom ini tidak diisi secara manual selama proses import, kemunculan nilai tersebut menunjukkan bahwa sistem *predictive lead scoring* berhasil melakukan perhitungan otomatis terkait tingkat potensi nasabah sebelum data ditampilkan di *dashboard*. Hasil pengujian beserta hasil validasinya disajikan dalam Gambar 6.



Gambar 5. Pengujian Import leads



Gambar 6. Pengujian Fetch Leads

3.3.3. Pengujian Model

Pengujian ini bertujuan untuk mengukur performa model *Random Forest* dalam mengklasifikasikan prospek nasabah. Model *Random Forest* yang telah dilatih dan dievaluasi menggunakan data uji dan akan dievaluasi menggunakan data uji untuk mengetahui tingkat akurasi dari model *Random Forest*. Parameter utama yang digunakan dalam pengembangan model *Random Forest* meliputi jumlah pohon keputusan (*number of trees*) dan kedalaman maksimal pohon (*maximum depth*). Pengujian akurasi dilakukan menggunakan *confusion matrix* untuk menilai seberapa baik sebuah model dapat memprediksi target dari sebuah data fitur [23]. *Confusion matrix* disajikan pada Tabel 4.

Tabel 4. *Confusion Matrix*

Aktual	Prediksi	
	Negatif (1)	Positif (1)
Negatif (0)	TN	FP
Positif (1)	FN	TP

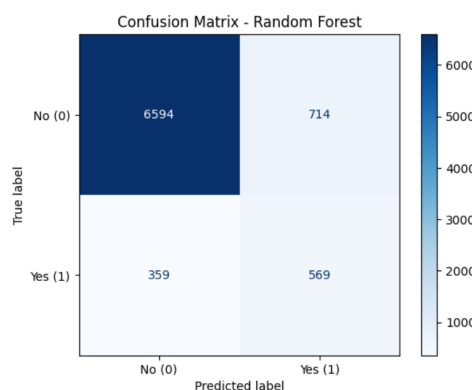
Penjelasan dari TN, FN, FP dan TP sebagai berikut :

1. *True Negative (TN)* : Data negatif yang diprediksi benar sebagai negatif.
2. *False Positive (FP)* : Data negatif yang salah diprediksi sebagai positif.
3. *False Negative (FN)* : Data positif yang salah diprediksi sebagai negatif.
4. *True Positive (TP)* : Data positif yang diprediksi benar sebagai positif.

Penilaian performa model dilaksanakan menggunakan metrik akurasi untuk menghitung persentase total prediksi yang tepat terhadap seluruh data yang diuji. Proses ini bertujuan untuk menentukan seberapa baik model yang dihasilkan dalam memprediksi data [24]. Rumus akurasi dituliskan pada Persamaan (3) [25]:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \cdot 100\% \quad (3)$$

Berikut adalah hasil *confusion matrix* yang diperoleh dari evaluasi model dengan menggunakan data uji disajikan pada Gambar 7.



Gambar 7. Confusion Matrix

Perhitungan akurasi berdasarkan nilai yang tertera pada *confusion matrix* di atas adalah :

$$\begin{aligned} \text{Accuracy} &= \frac{TP + TN}{TP + TN + FP + FN} \cdot 100\% \\ \text{Accuracy} &= \frac{569 + 6594}{569 + 659 + 714 + 359} \cdot 100\% \\ \text{Accuracy} &= \frac{7163}{8236} \cdot 100\% \\ \text{Accuracy} &= 0,869 \cdot 100\% \\ \text{Accuracy} &\approx 0,87 \cdot 100\% \\ \text{Accuracy} &\approx 87\% \end{aligned}$$

Hasil evaluasi menunjukkan akurasi sebesar 87%, yang menunjukkan bahwa model *Random Forest* memiliki kinerja yang baik dalam mengklasifikasikan data nasabah. Untuk analisis yang lebih mendalam, hasil evaluasi model *Random Forest* juga disajikan melalui *classification report* yang mencakup metrik *precision*, *recall*, dan *f1-score* disajikan pada Gambar 8.

=== Detail Classification Report ===				
	precision	recall	f1-score	support
0	0.95	0.90	0.92	7308
1	0.44	0.61	0.51	928
accuracy			0.87	8236
macro avg	0.70	0.76	0.72	8236
weighted avg	0.89	0.87	0.88	8236

Gambar 8. Clasification Report

4. KESIMPULAN

Berdasarkan hasil rancang bangun dan pengujian yang telah dilakukan pada sistem *dashboard LeadsUp*, dapat ditarik beberapa simpulan sebagai berikut :

1. Penelitian ini berhasil merancang dan membangun *dashboard LeadsUp* menggunakan *tech stack* React.js, Express.js, dan Supabase yang terintegrasi dengan model *predictive lead scoring*.
2. Sistem ini berhasil mengubah cara pemasaran deposito berjangka dari metode konvensional menjadi berbasis prioritas (*High, Medium, Low*) sehingga membantu dalam menentukan target nasabah.
3. Berdasarkan hasil pengujian *Black Box*, semua fungsi berjalan dengan baik skenario yang ditetapkan. Hasil pengujian Postman juga menunjukkan adanya kestabilan dalam pertukaran data. Kemudian pada pengujian model algoritma *Random Forest* dalam melakukan klasifikasi potensi nasabah dengan tingkat akurasi mencapai 87% menjadikannya layak digunakan sebagai landasan dalam pengambilan keputusan pemasaran.
4. Peneliti selanjutnya dianjurkan untuk memperkuat prosedur keamanan data melalui penerapan enkripsi terkini yang setara dengan standar perbankan, demi menjamin bahwa sistem memenuhi ketentuan privasi data keuangan yang berlaku dengan ketat.

UCAPAN TERIMA KASIH

Penulis menyampaikan terima kasih yang sebesar-besarnya kepada seluruh dosen Program Studi Informatika, Fakultas Sains, Teknologi dan Kesehatan Universitas Sahid Surakarta yang telah memberikan bimbingan, arahan dan memfasilitasi penulis selama masa perkuliahan untuk terus berkembang. Terima kasih juga ditujukan kepada rekan-rekan seperjuangan atas motivasi serta diskusi mendalam mengenai logika algoritma *Random Forest* selama penyusunan jurnal ini.

DAFTAR PUSTAKA

- [1] R. A. R. Muhammad, W. N. Aghniya, and H. Tantyoko, "Penerapan Algoritma Machine Learning Untuk Memprediksi Term Deposit Nasabah Perbankan," *LEDGER J. Inform. Inf.*

- Technol.*, vol. 2, no. 3, pp. 145–156, 2023.
- [2] D. A. Fauziah, A. Muliawan, I. Sabilirrasyad, and B. Wahyu, “Smart Campaign for Smart Business Using Machine Learning on Improving Product Marketing Strategy,” vol. 6, pp. 616–622, 2025.
- [3] M. Hamim, “Toward a Smart Lead Scoring,” vol. 13, no. 2, pp. 433–443, 2022.
- [4] L. Hiryanto, “Sistem Rekomendasi Deposito Berjangka Menggunakan Metode Random Forest: Studi Kasus Pada Bank Marketing Uci,” *Comput. J. Comput. Sci. Inf. Syst.*, vol. 9, no. 1, pp. 1–8, 2025.
- [5] D. Pratiwi AR, A. Fabanyo, K. N. Salam, A. Asmawiyah, and Herenal Daeng Toto, “Faktor-faktor yang Mempengaruhi Nasabah Memilih Tabungan Simpedes pada PT. Bank Rakyat Indonesia (Persero), Tbk. Unit Borong Raya,” *J. Manaj. Perbank. Keuang. Nitro*, vol. 5, no. 2, pp. 119–133, 2022.
- [6] V. Panguluri, “AI-Powered Predictive Lead Scoring: Transforming Sales Prioritization through Data-Driven Insights,” *SARC Publ.*, vol. 5, no. 8, pp. 175–181, 2025.
- [7] M. Wu, P. Andreev, and M. Benyoucef, “Association for Information Systems Association for Information Systems Smart Sales: Amplifying the Power of Predictive Lead Scoring in Smart Sales: Amplifying the Power of Predictive Lead Scoring in B2B Sales B2B Sales,” no. July, 2024.
- [8] V. K. Tarra and A. K. Mittapelly, “AI-Driven Lead Scoring in Salesforce: Using Machine Learning Models to Prioritize High-Value Leads and Optimize Conversion Rates,” *Int. J. Emerg. Trends Comput. Sci. Inf. Technol.*, vol. 5, no. 2, pp. 63–72, 2024.
- [9] P. Borugadda, P. Nandru, C. Madhavaiah, and S. Theresa, “Predicting the Success of Bank Telemarketing for Selling Long-term Deposits: An Application of Machine Learning Algorithms,” *Theresa J. Humanit. Soc. Sci.*, vol. 7, no. 1, pp. 91–108, 2021.
- [10] Y. Mardi, A. Wahyuni, and F. Gusnil, “Pembuatan dan Pemanfaatan Dashboard Data Rekam Medis Sebagai Alat Bantu Sistem Pendukung Keputusan,” *Arter. J. Ilmu Kesehat.*, vol. 4, no. 3, pp. 155–161, 2023.
- [11] J. A. Ramadhan, D. T. Haniva, and A. Suharso, “Systematic Literature Review Penggunaan Metodologi Pengembangan Sistem Informasi Waterfall , Agile , dan Hybrid,” vol. 07, pp. 36–42, 2023.
- [12] A. Sutikno, “Jurnal Publikasi Ilmu Komputer dan Sistem Informasi Kearsipan Dinamis Berbasis Web di Lingkungan Pemerintah Desa Sukaresik,” pp. 0–10, 2025.
- [13] X. Y. Tan, S. M. Lim, T. M. Lim, C. W. Tan, and N. A. Husaini, “Leveraging Random Forest Algorithm for Enhanced Lead Conversion and Customer Retention,” *Int. J. Adv. Sci. Eng. Inf. Technol.*, vol. 14, no. 6, pp. 2141–2148, 2024.
- [14] H. A. Salman, A. Kalakech, and A. Steiti, “Random Forest Algorithm Overview,” vol. 2024, pp. 69–79, 2024.
- [15] A. F. Azmi And A. Voutama, “Komputa : Jurnal Ilmiah Komputer Dan Informatika Prediksi Churn Nasabah Bank Menggunakan Klasifikasi Random Forest Dan Decision Tree Dengan Evaluasi Confusion Matrix Komputa : Jurnal Ilmiah Komputer Dan Informatika,” Vol. 13, No. 1, 2024.
- [16] E. Helmud, E. Helmud, and P. Romadiana, “Classification Comparison Performance of Supervised Machine Learning Random Forest and Decision Tree Algorithms Using Confusion Matrix,” vol. 13, pp. 92–97, 2024.
- [17] R. Saputra, A. Purno, W. Wibowo, J. C. No, K. C. Kidul, and K. Bandung, “Analisis Komparasi Algoritma SVM , Random Forest dan MLP-NN,” vol. 11, no. 1, pp. 211–224, 2026.
- [18] M. S. Efendi and A. K. Zyen, “Penerapan Algoritma Random Forest Untuk Prediksi Penjualan Dan Sistem Persediaan Produk,” vol. 5, no. 1, pp. 12–20, 2024.
- [19] I. S. Handayanto and I. Nuryasin, “Pengujian Blackbox Decision Table pada Sistem Aplikasi Mobile Sharing Story App Blackbox Decision Table Testing on the Mobile Sharing Story App Application System,” vol. 13, no. 105, 2024.
- [20] I. Imam, R. Khoirri, and S. T. Informatika, “Rancang Bangun Aplikasi Antrian Komunitas Game Arcade Maimai Berbasis Framework Laravel Design and Development of Maimai Arcade Game Community Queue Application Based on Laravel Framework,” vol. 15, no. 295,

pp. 341–351, 2026.

- [21] P. P. Kore, M. J. Lohar, M. T. Surve, and S. Jadhav, "API Testing Using Postman Tool," no. December, 2022.
- [22] S. D. Sri, M. A. S, and S. V. R, "Automating REST API Postman Test Cases Using LLM," 2024.
- [23] G. Ramadhan, "Analisis Sistem Kendali Kursi Roda Berbasis Sinyal Electrooculography Dengan Metode Random Forest," 2025.
- [24] R. Andrianto, D. Retnoningsih, F. Fitriyadi, And A. Charolina, "Prediksi Kelulusan Mahasiswa Universitas Sahid Surakarta Angkatan 2021 Menggunakan K-Nearest Neighbor," Vol. 9, No. 8, Pp. 39–48, 2025.
- [25] S. Amaliah and M. Nusrang, "Penerapan Metode Random Forest Untuk Klasifikasi Varian Minuman Kopi Di Kedai Kopi Konijiwa Bantaeng," vol. 4, no. 2, pp. 121–127, 2022.