



Implementasi algoritma random forest pada web-app sebagai instrumen deteksi dini penyakit diabetes

Habibul Fauzan¹, Elin Haerani^{*2}, Fitra Kurnia³, Novi Yanti⁴

Email: ¹112050112476@students.uin-suska.ac.id, ^{2*}elin.haerani@uin-suska.ac.id, ³fitra.k@uin-suska.ac.id, ⁴novi_yanti@uin-suska.ac.id

¹²³⁴Teknik Informatika, Sains dan Teknologi, Universitas Islam Negeri Sultan Syarif Kasim Riau

Diterima: 01 April 2026 | Direvisi: - | Disetujui: 26 April 2026
©2020 Program Studi Teknik Informatika Fakultas Ilmu Komputer,
Universitas Muhammadiyah Riau, Indonesia

Abstrak

Diabetes adalah penyakit metabolik kronis yang termasuk salah satu penyebab utama kematian di dunia, dengan perkiraan jumlah penderitanya bisa mencapai 1,3 miliar pada tahun 2050. Keterlambatan diagnosis menjadi masalah utama, di mana hampir separuh penderita tidak menyadari kondisinya pada tahap awal, sehingga meningkatkan risiko komplikasi fatal. Pendekatan *data mining* melalui algoritma klasifikasi telah banyak dimanfaatkan untuk skrining awal. Namun, pengembangan model rekam medis sering kali terkendala oleh masalah ketidakseimbangan data (*imbalanced data*), yang menyebabkan model bias terhadap kelas mayoritas dan menurunkan sensitivitas deteksi pada kelas minoritas (pasien sakit). Selain itu, belum banyak penelitian yang mengintegrasikan model prediktif tersebut ke dalam antarmuka aplikasi yang responsif bagi pengguna akhir. Sehingga penelitian ini mengimplementasikan *Random Forest* yang dioptimalkan dengan teknik SMOTE (*Synthetic Minority Over-sampling Technique*) ke dalam aplikasi berbasis *web* agar bisa digunakan sebagai alat deteksi dini yang lebih praktis. *Random Forest* dipilih karena mampu menangani data kompleks dan mengurangi risiko *overfitting*. Tahapan penelitian mencakup pra-pemrosesan data, penyeimbangan data latihan menggunakan SMOTE, penyetelan parameter model melalui *hyperparameter tuning* dengan *Grid Search*, serta pengembangan sistem berarsitektur *client-server* menggunakan *AstroJS* dan *Flask*. Hasil evaluasi menunjukkan bahwa penggunaan SMOTE mampu meningkatkan kemampuan model dalam mengenali kelas minoritas (pasien diabetes) secara nyata. Model mencapai *Recall* 75,0% dengan akurasi 95,8%, sehingga jumlah kesalahan *False Negative* dapat ditekan. Aplikasi yang dikembangkan diuji menggunakan *Black Box Testing* dan dinyatakan berhasil sebagai alat deteksi dini yang responsif serta mudah diakses, baik oleh tenaga kesehatan maupun masyarakat umum.

Kata kunci: *Diabetes, Random Forest, SMOTE, Aplikasi Web, Deteksi Dini*

Implementation of random forest algorithm in a web application as an early detection tool for diabetes

Abstract

Diabetes is a chronic metabolic disease and one of the leading causes of death worldwide, with the number of sufferers projected to reach 1.3 billion by 2050. Delayed diagnosis remains a primary challenge, as nearly half of those affected are unaware of their condition in the early stages, thereby increasing the risk of fatal complications. Data mining approaches using classification algorithms have been widely utilized for early screening. However, the development of medical record models is often hindered by imbalanced data, which causes models to be biased toward the majority class and reduces detection sensitivity for the minority class (patients with diabetes). Furthermore, there is a lack of research integrating these predictive models into responsive application interfaces for end-users. Consequently, this study implements *Random Forest* optimized with the SMOTE (*Synthetic Minority Over-sampling Technique*) into a web-based application to serve as a practical early detection tool. *Random Forest* was selected for its ability to handle complex data and reduce the risk of *overfitting*. The research stages include data preprocessing, balancing training data using SMOTE, model parameter adjustment through hyperparameter tuning with *Grid*

Search, and the development of a client-server architecture using AstroJS and Flask. The evaluation results demonstrate that the use of SMOTE significantly improves the model's ability to identify the minority class. The model achieved a Recall of 75.0% and an overall accuracy of 95.8%, effectively minimizing False Negative errors. The developed application was verified through Black Box Testing and was declared successful as a responsive and accessible early detection tool for both healthcare professionals and the general public.

Keywords: Diabetes, Random Forest, SMOTE, Web Application, Early Detection

1. PENDAHULUAN

Diabetes mellitus adalah gangguan metabolisme kronis yang terjadi ketika tubuh tidak mampu memproduksi insulin yang cukup atau tidak bisa menggunakannya secara efektif untuk mengatur kadar gula darah [1]. Kondisi kadar gula darah tinggi yang tidak terkontrol ini berpotensi memicu kerusakan serius pada berbagai sistem tubuh, khususnya saraf dan pembuluh darah. Komplikasi yang sering muncul meliputi penyakit jantung, stroke, kerusakan saraf (neuropathy), gagal ginjal, kebutaan, hingga risiko amputasi [2]. Intinya, masalah terbesar muncul ketika gula darah dibiarkan tinggi terus menerus dalam jangka panjang. Dengan dampaknya yang luas ke seluruh tubuh, diabetes sekarang jadi salah satu penyebab utama kematian dan kecacatan di dunia. Selain soal kesehatan, penyakit ini juga memberi beban ekonomi dan sosial yang besar bagi negara-negara terdampak [3].

Secara data, jumlah penderita diabetes memang sedang naik dengan kecepatan yang mengkhawatirkan. Pada 2021 tercatat sekitar 529 juta orang hidup dengan diabetes, dan angkanya diperkirakan bisa melonjak menjadi lebih dari 1,3 miliar pada 2050 jika tidak ada intervensi kesehatan yang kuat [4]. Di Asia termasuk Indonesia laju peningkatan kasus termasuk yang paling cepat di dunia [5]. Proyeksi nasional juga menunjukkan hal yang sama, tanpa pencegahan dan deteksi dini yang masif, prevalensi diabetes di Indonesia diperkirakan terus naik tajam dalam beberapa dekade ke depan [6]. Data ini menegaskan bahwa kita butuh instrumen kesehatan yang efektif untuk menahan laju pertumbuhan penyakit diabetes.

Masalah besar dalam penanganan diabetes bukan cuma soal jumlah kasusnya, tapi karena penyakit ini sering “diam diam” di awal. Banyak orang tidak sadar sampai kondisinya sudah memburuk. Faktanya, hampir setengah penderita diabetes saat ini belum terdiagnosis, sehingga risiko komplikasi berat dan kematian dini meningkat [7]. Keterlambatan ini biasanya dipicu oleh rendahnya kesadaran masyarakat tentang faktor risiko, akses layanan kesehatan yang masih terbatas dan mahal, serta sistem deteksi dini di fasilitas kesehatan primer yang belum optimal [8]. Oleh karena itu, diperlukan pendekatan teknologi yang dapat mempermudah proses skrining awal secara akurat dan efisien.

Perkembangan teknologi data mining memberi peluang besar untuk menjawab tantangan ini [9]. Secara sederhana, data mining adalah proses menemukan pola dan informasi penting dari kumpulan data. Ia merupakan bagian dari Knowledge Discovery in Databases (KDD) karena fokusnya pada penggalian pola signifikan dari basis data besar hingga menjadi pengetahuan baru [10]. Salah satu teknik yang paling sering dipakai adalah klasifikasi, yakni mengelompokkan data ke dalam kategori tertentu berdasarkan target yang ingin diprediksi [9], [10]. Berbagai algoritma klasifikasi sudah dicoba untuk memprediksi risiko diabetes. Misalnya, penelitian Silalahi dkk. (2023) menggunakan metode *supervised learning* dengan *K-Nearest Neighbor* (KNN) pada kasus diabetes pada wanita dan memperoleh akurasi 82–84% [11]. Lalu, Putri dkk. (2021) menerapkan *Decision Tree* C4.5 dan mencapai akurasi sekitar 90% [12]. Namun, model pohon keputusan tunggal seringkali rentan terhadap *overfitting* atau terlalu pas pada data latih sehingga kurang akurat pada data baru.

Penelitian Samet dkk. (2022) menunjukkan bahwa Random Forest mampu mencapai akurasi sekitar 92%, lebih tinggi dibanding algoritma tunggal lainnya [13]. Kelebihan ada pada kemampuan menangani data yang kompleks sekaligus menekan varians model lewat bagging dan pemilihan fitur acak. Di sisi lain, Priande & Zamroni (2024) menekankan pentingnya optimasi *hyperparameter*. Mereka menemukan bahwa pengaturan jumlah pohon (*n_estimators*) yang tepat bisa menurunkan error secara signifikan [14]. Sementara itu, Anisa & Kurniawan (2024) sudah membuat aplikasi web prediksi diabetes berbasis *Flask*, tetapi masih terbatas pada implementasi dasar dan belum banyak mengoptimalkan antarmuka pengguna [15].

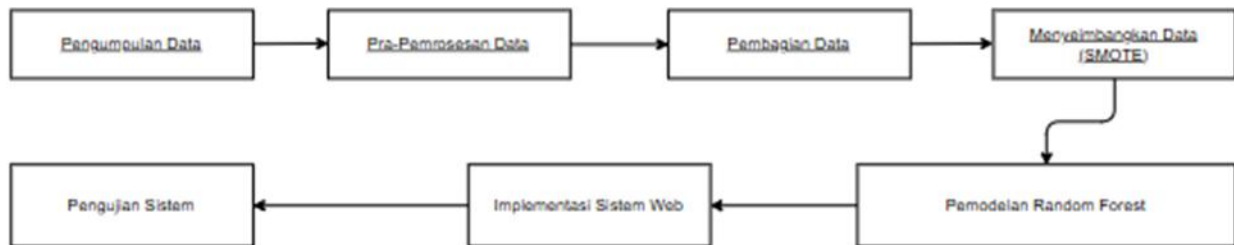
Walaupun sudah banyak penelitian, masih ada gap yang cukup besar. Pertama, dataset medis sering tidak seimbang (*imbalanced*). Data pasien sehat jauh lebih banyak dibanding data pasien sakit. Jika model dilatih langsung dari data mentah, hasilnya cenderung bias ke kelas mayoritas, sehingga kemampuan mendeteksi kelas minoritas (yang justru penting untuk deteksi dini) jadi buruk. Lal & S (2023) juga menekankan bahwa isu bias ini masih perlu dieksplorasi lebih dalam [16]. Kedua, kebanyakan penelitian fokus pada pengembangan algoritma, tanpa implementasi yang benar-benar bisa dipakai pengguna akhir. Solusi yang ada pun sering belum memperhatikan pengalaman pengguna, terutama dalam bentuk aplikasi web yang responsif dan mudah diakses.

Berdasarkan analisis masalah penelitian tersebut, penelitian ini bertujuan untuk mengimplementasikan algoritma *Random Forest* yang diperkuat dengan teknik SMOTE (*Synthetic Minority Over-sampling Technique*) untuk mengatasi masalah ketidakseimbangan data. *Random Forest* dipilih karena mampu menangani data kompleks dan mengurangi risiko *overfitting*, dinilai sangat sesuai untuk membangun model prediksi yang stabil dan akurat [16]. Solusi ini kemudian diimplementasikan ke dalam sebuah *Web-App* modern menggunakan teknologi *AstroJS* pada sisi *frontend* dan *Flask* pada sisi *backend*. Harapan utama

dari penelitian ini adalah tersedianya instrumen deteksi dini yang tidak hanya memiliki sensitivitas tinggi dalam mengenali pasien diabetes, tetapi juga praktis dan mudah diakses untuk mendukung layanan kesehatan di masyarakat.

2. METODE PENELITIAN

Penelitian ini memakai metodologi yang sistematis untuk membangun sistem deteksi dini diabetes. Pendekatannya menggabungkan teknik *data mining* dengan pengembangan perangkat lunak modern. Secara umum, alurnya mencakup pengumpulan data, pra-pemrosesan, penanganan ketidakseimbangan kelas, pemodelan algoritma, implementasi sistem berbasis web, hingga pengujian. Alur metodologi penelitian ditampilkan pada Gambar 1.



Gambar 1. Arsitektur Enkripsi AES

2.1. Pengumpulan Data

Data penelitian ini berupa data sekunder dari *Kaggle*. Dataset berisi 100.000 baris rekam medis pasien dengan beberapa atribut, seperti gender, usia, hipertensi, riwayat penyakit jantung, riwayat merokok, *BMI*, kadar *HbA1c*, dan kadar glukosa darah. Variabel targetnya adalah status diabetes dengan nilai biner, 0 untuk negatif dan 1 untuk positif.

Tabel 1. Tabel Barang

No	Atribut	Deskripsi
1	<i>Gender</i>	Jenis kelamin pasien
2	<i>Age</i>	Umur pasien
3	<i>hypertension</i>	Pasien memiliki riwayat hipertensi atau tidak
4	Heart Disease	Pasien memiliki riwayat penyakit jantung atau tidak
5	Smoking History	Status riwayat merokok
6	BMI	Indeks Massa Tubu, dihitung berdasarkan berat badan (kg) dibagi tinggi badan kuadrat (m ²)
7	HbA1c Level	Level HbA1c Adalah ukuran rata-rata kadar gula darah seseorang selama 2-3 bulan terakhir.
8	Blood Glucose Level	Level glukosa darah merujuk pada jumlah glukosa dalam aliran darah pada suatu waktu tertentu
9	Diabetes	Diabetes adalah variabel target yang diprediksi, dengan nilai 1 menunjukkan adanya diabetes dan 0 menunjukkan tidak adanya diabetes

2.2. Pra-Pemrosesan Data

Tahap pra-pemrosesan data dilakukan untuk meningkatkan kualitas data sebelum digunakan oleh algoritma [17]. Tahapan dalam penelitian ini terdiri dari tiga langkah utama:

1. Pembersihan Data (*Data Cleaning*)
 Dari analisis awal, ditemukan data duplikat dengan nilai identik di semua kolom. Karena itu, 3.854 baris data duplikat dihapus. Penghapusan duplikat merupakan bagian penting dari pembersihan data karena duplikasi dapat menimbulkan bias [18].
2. Rekayasa Fitur (*Feature Engineering*)
Random Forest membutuhkan input numerik [19]. Oleh karena itu, variabel kategorikal seperti *gender* dan *smoking history* dikonversi menjadi angka menggunakan teknik *Label Encoding*. Sebagai contoh, atribut '*Female*' dipetakan menjadi 0 dan '*Male*' menjadi 1.
3. Standarisasi Data
 Fitur numerik seperti usia dan glukosa darah memiliki skala yang jauh berbeda. Untuk mencegah bias skala, dilakukan standarisasi dengan *StandardScaler* [20]. Hasilnya, distribusi setiap fitur memiliki rata-rata 0 dan standar deviasi 1.

2.3. Pembagian Data (*Data Splitting*)

Setelah data dibersihkan, dataset dibagi dengan rasio 80% data latih dan 20% data uji. Pemisahan ini dilakukan agar evaluasi model tetap objektif.

2.4. Penyeimbangan Data

Pada analisis distribusi kelas, terlihat ketidakseimbangan yang cukup besar antara pasien diabetes dan non-diabetes. Kondisi ini bisa membuat model bias ke kelas mayoritas. Untuk mengatasinya, penelitian ini menggunakan SMOTE (*Synthetic Minority Over-sampling Technique*), yaitu teknik yang membuat sampel baru untuk kelas minoritas berdasarkan tetangga terdekatnya, sehingga proporsi kelas menjadi lebih seimbang [21], [22]. Penerapan SMOTE hanya diterapkan pada data latih agar tidak terjadi kebocoran data (*data leakage*) ke data uji [23].

2.5. Pemodelan Algoritma Random Forest

Model klasifikasi dibangun dengan *Random Forest*, yaitu metode *ensemble* yang menggabungkan banyak pohon keputusan. Pada penelitian ini random forest dioptimasi dengan *hyperparameter tuning* menggunakan *Grid Search* dengan *5-fold cross-validation* (CV=5). Parameter yang diuji antara lain jumlah pohon (*n_estimators*), kedalaman maksimum (*max_depth*), dan jumlah sampel minimum untuk pemisahan node.

Tabel 2. Nilai *Hyperparameter*

<i>hyperparameter</i>	Nilai yang diuji	Deskripsi
<i>n_estimators</i>	50, 100, 200	Jumlah pohon dalam <i>tree</i> (hutan).
<i>Max_depth</i>	10, 20, <i>none</i>	Kedalaman maksimum setiap pohon.
<i>Min_sample_split</i>	2, 5	Jumlah minimum sampel untuk memecah simpul.
<i>Min_sample_leaf</i>	1, 4, 8	Jumlah minimum sampel pada daun pohon.

2.6. Evaluasi

Setelah pengujian selesai, dilakukan evaluasi untuk menilai performa model. Ukuran yang dipakai adalah confusion matrix, yaitu tabel klasifikasi yang menunjukkan seberapa sering prediksi model benar atau salah, sehingga kinerjanya bisa dilihat dengan jelas [24]. Confusion matrix menampilkan jumlah data uji yang terklasifikasi benar maupun salah [25]. Beberapa metrik evaluasi yang digunakan sebagai berikut:

1. *Accuracy*

Akurasi Merupakan perbandingan antara jumlah prediksi yang benar (baik positif maupun negatif) dengan total keseluruhan data [26].

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \tag{1}$$

2. *Precision*

Precision atau presisi merupakan perbandingan antara jumlah prediksi benar positif dengan total seluruh hasil yang diprediksi positif [26].

$$Precision = \frac{TP}{TP+FP} \tag{2}$$

3. *Recall*

Recall merupakan perbandingan antara jumlah prediksi benar positif dengan total keseluruhan data yang benar positif [26].

$$Recall = \frac{TP}{TP+FN} \tag{3}$$

4. *F1-score*

F1-Score adalah rata-rata harmonis antara precision dan recall [26].

$$F1 = \frac{2 \cdot Precision \cdot Recall}{Precision + Recall} \tag{4}$$

2.7. Implementasi dan Pengujian Sistem

Sistem *Web-App* dibuat dengan konsep client-server. *Backend* dibangun menggunakan *Python Flask* sebagai *REST API* untuk memproses permintaan prediksi dari model yang sudah diekspor (*serialized*). *Frontend* dibuat dengan *AstroJS* agar antarmuka lebih responsif dan cepat.

Untuk memastikan kualitasnya, sistem diuji menggunakan *Black Box Testing*. Pengujian ini fokus pada validasi fungsi *input-output*, memastikan sistem mampu menerima berbagai jenis masukan, memberi hasil prediksi secara *real-time*, dan menangani kesalahan *input* tanpa membuat aplikasi gagal.

3. HASIL DAN PEMBAHASAN

3.1. Hasil Pra-Pemrosesan Data

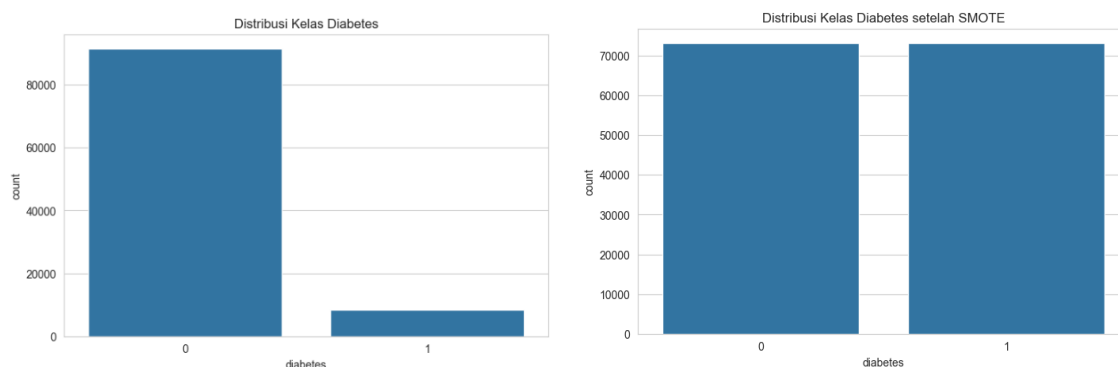
Tahap awal penelitian ini melibatkan pemrosesan 100.000 data rekam medis pasien yang memiliki sembilan atribut klinis utama (rincian pada Tabel 1). Dalam tahapan pengecekan data, ditemukan adanya 3.854 data yang duplikat. Data ganda seperti ini berisiko tinggi membuat model klasifikasi mengalami *overfitting* yaitu model akan lebih banyak menghafal data yang berulang daripada benar-benar mempelajari polanya. Untuk mencegah hal tersebut, seluruh data duplikat dibersihkan sehingga menyisakan 96.146 data unik yang valid untuk diproses lebih lanjut

Setelah proses pembersihan selesai, dataset disesuaikan dengan kebutuhan algoritma *Random Forest* yang mengharuskan semua input berupa angka (numerik) untuk pemisahan *node*. Langkah pertama adalah mengubah data kategorikal menggunakan teknik *Label Encoding*. Contohnya, jenis kelamin perempuan diubah menjadi angka 0 dan laki-laki menjadi 1. Cara yang sama juga diterapkan pada variabel riwayat merokok (*smoking_history*), di mana setiap kategorinya diberi nilai angka yang spesifik.

Selanjutnya, dilakukan standarisasi pada fitur numerik. Hal ini penting karena terdapat perbedaan rentang nilai yang cukup jauh antar variabel; misalnya, umur pasien berada di rentang puluhan (0–80 tahun), sedangkan kadar glukosa darah bisa mencapai ratusan (>200 mg/dL). Perbedaan skala ini berisiko membuat algoritma menjadi bias dengan memprioritaskan fitur yang bernilai besar. Untuk mencegah hal tersebut, metode *StandardScaler* digunakan agar semua fitur numerik memiliki skala yang seragam, yaitu dengan rata-rata (*mean*) 0 dan standar deviasi 1. Rangkaian pra-pemrosesan ini memastikan setiap parameter klinis memiliki bobot yang setara saat model mulai dilatih.

3.2. Hasil Penyeimbangan Data dengan SMOTE

Analisis distribusi variabel target menunjukkan ketimpangan yang besar. Sebagian besar sampel adalah pasien non-diabetes (kelas 0), sedangkan jumlah pasien diabetes (kelas 1) jauh lebih sedikit. Untuk mengatasi masalah ketidakseimbangan kelas, penelitian ini menerapkan teknik SMOTE. SMOTE bekerja dengan cara membangkitkan data sintesis baru.



Gambar 2. Distribusi Kelas Diabetes sebelum dan sesudah SMOTE

Seperti yang ditunjukkan pada Gambar 2, setelah penerapan SMOTE, distribusi kelas pada data latih menjadi seimbang. Keseimbangan ini memberikan kesempatan bagi algoritma *Random Forest* untuk mempelajari karakteristik fitur dari kelas positif diabetes secara lebih mendalam, sehingga diharapkan dapat meningkatkan sensitivitas (*Recall*) model.

3.3. Implementasi *Random Forest*

Pemodelan klasifikasi dilakukan dengan menerapkan algoritma *Random Forest* pada data latih yang telah diseimbangkan melalui teknik SMOTE. Setelah itu, model dioptimasi melalui penyetelan parameter (*hyperparameter tuning*) menggunakan metode *Grid Search*. Pendekatan ini bertujuan untuk memastikan algoritma memiliki kepekaan yang tinggi dalam mempelajari karakteristik kelas minoritas, yaitu pasien yang berisiko diabetes.

Berdasarkan hasil pengujian, model menghasilkan tingkat akurasi keseluruhan sebesar 95,8%. Analisis lebih mendalam pada *Confusion Matrix* menunjukkan efektivitas model dalam mendeteksi kelas positif, yang dibuktikan dengan pencapaian nilai *Recall* sebesar 75%. Model berhasil mengenali 1.271 pasien diabetes secara tepat (*True Positive*) dan secara signifikan menekan angka *False Negative* (pasien diabetes yang terlewat atau gagal terdeteksi) menjadi 425 kasus.

Sensitivitas yang tinggi dalam model ini menyebabkan munculnya 373 kasus *False Positive*, yaitu kondisi ketika pasien yang sebenarnya sehat diprediksi memiliki risiko penyakit. Meskipun demikian, dalam pengembangan sistem deteksi dini (*early detection*), kondisi ini masih dianggap sebagai *trade-off* yang dapat diterima. Pada tahap skrining awal, tujuan utama bukanlah memberikan diagnosis akhir, melainkan memastikan bahwa pasien yang berpotensi berisiko tidak terlewatkan. Oleh karena itu,

menekan jumlah kasus pasien berisiko yang tidak terdeteksi menjadi prioritas utama. Pendekatan ini sejalan dengan prinsip preventif dalam dunia medis. Dalam praktik skrining, lebih aman jika seseorang yang sehat terdeteksi berisiko dan kemudian menjalani pemeriksaan lanjutan, dibandingkan dengan pasien yang sebenarnya sakit tetapi dinyatakan sehat sehingga tidak mendapatkan penanganan sejak dini [27].

Tabel 2. Nilai *Hyperparameter*

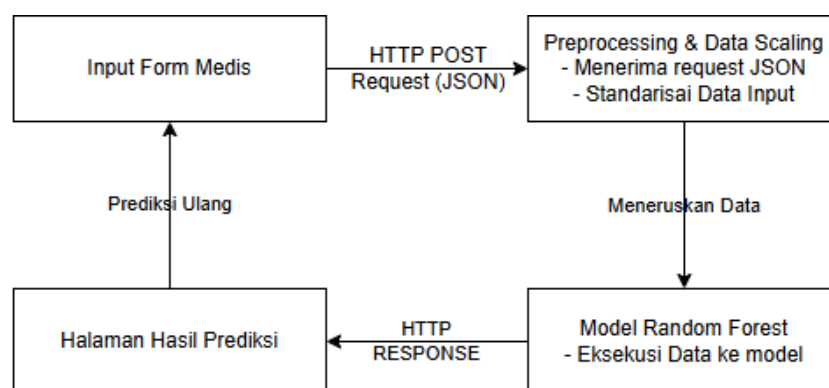
Metrik Evaluasi	Kelas 0	Kelas 1
<i>Accuracy</i>	95,8%	95,8%
Recall	98%	75%
<i>Precision</i>	98%	77%
<i>F1-Score</i>	98%	76%

3.4 Implementasi Berbasis Web

Setelah melalui tahap evaluasi dan terpilihnya model *Random Forest* dengan optimasi SMOTE sebagai model terbaik, langkah selanjutnya adalah mengintegrasikan model tersebut ke dalam sebuah perangkat lunak yang dapat diakses oleh pengguna. Implementasi ini bertujuan untuk menjembatani kesenjangan antara model matematis yang kompleks dengan kebutuhan praktis di lapangan, yaitu menyediakan alat deteksi dini yang mudah digunakan (*user-friendly*).

1. Arsitektur Sistem

Untuk menjamin performa dan skalabilitas aplikasi, sistem dirancang menggunakan arsitektur *Client-Server* terdistribusi yang memisahkan antara antarmuka presentasi (*Frontend*) dan logika pemrosesan kecerdasan buatan (*Backend*). Pemisahan komponen ini sangat krusial agar beban komputasi dari model *Random Forest* tidak membebani memori perangkat pengguna, melainkan dieksekusi secara terpusat di server. Alur komunikasi arsitektur ini divisualisasikan pada Gambar 4

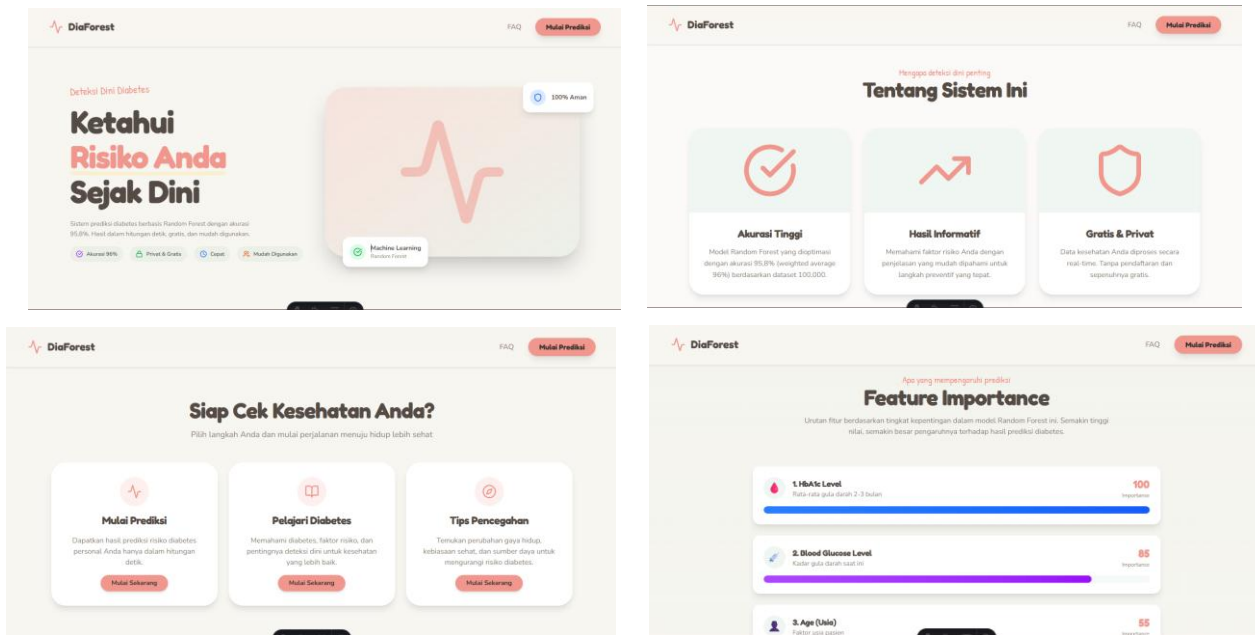


Gambar 3. Diagram Alur Arsitektur

Berdasarkan Gambar 3, aliran data pada sistem ini berlangsung secara asinkron (*asynchronous*) dengan menggunakan pola komunikasi RESTful API. Proses interaksi dimulai ketika pengguna memasukkan parameter input melalui formulir web pada antarmuka AstroJS, yang kemudian melakukan validasi sisi klien (*client-side validation*) untuk memastikan integritas tipe data. Setelah data tervalidasi, *frontend* mengonversi input tersebut ke dalam struktur *JavaScript Object Notation* (JSON) dan mengirimkannya melalui *HTTP POST Request* menuju *endpoint API backend* Flask. Selanjutnya, server Flask menerima *payload* JSON, mengekstrak nilai fiturnya, dan menerapkan fungsi transformasi data berupa normalisasi *StandardScaler* agar rentang nilainya sesuai dengan distribusi data latih model. Data yang telah terstandarisasi tersebut kemudian diproses oleh model *Random Forest*, di mana hasil akhirnya dikemas kembali oleh server ke dalam bentuk respons JSON dan dikirimkan ke *browser* pengguna hanya dalam hitungan detik

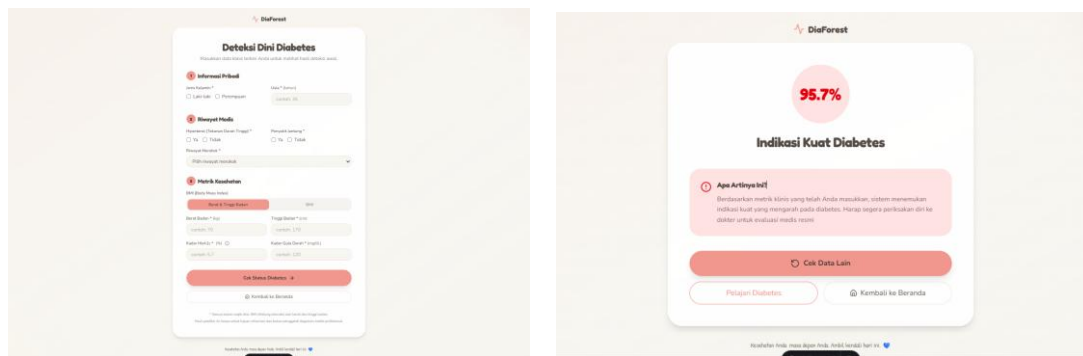
2. User Interface

Desain antarmuka dirancang sederhana namun informatif. Halaman beranda aplikasi menyajikan visualisasi data dan fakta medis yang bersumber dari studi literatur, bertujuan untuk memberikan pemahaman konteks sebelum pengguna melakukan deteksi dini.



Gambar 3. User Interface

Setelah mendapatkan pemahaman konteks melalui halaman beranda, pengguna diarahkan ke halaman prediksi. Halaman ini berisi formulir input data. Hasil prediksi ditampilkan secara komprehensif dan mudah dipahami. Sistem juga menyajikan status risiko lengkap dengan persentase probabilitasnya.



Gambar 3. Halaman Hasil Klasifikasi Deteksi Dini

3.5 Pengujian Sistem

Untuk memastikan keandalan sistem, dilakukan serangkaian pengujian perangkat lunak menggunakan metode *Black Box Testing*. Metode ini berfokus pada pengujian fungsionalitas aplikasi dengan meninjau kesesuaian antara *input* yang diberikan dan *output* yang dihasilkan, tanpa memperhatikan struktur kode internal program. Skenario pengujian dirancang menggunakan teknik *Equivalence Partitioning*, di mana input dibagi menjadi kelas data valid dan tidak valid. Pengujian mencakup validasi formulir, akurasi respons sistem, dan penanganan kesalahan (*error handling*). Hasil pengujian fungsionalitas sistem disajikan secara lengkap pada Tabel 4.

Tabel 2. Hasil Pengujian *BlackBox Testing*

No.	Skenario Pengujian	Data Input	Hasil yang Diharapkan	Hasil Aktual Sistem	Status
1	Navigasi Beranda (<i>Landing Page</i>)	Klik tombol "Cek Sekarang" atau "Mulai Sekarang"	Sistem secara otomatis mengarahkan pengguna ke halaman formulir prediksi.	Berhasil berpindah halaman ke formulir prediksi.	Berhasil
2	Prediksi Indikasi Kuat Diabetes (Positif)	Input data klinis berisiko.	Sistem memproses data dan menampilkan hasil "Indikasi Kuat Diabetes" dengan probabilitas di atas 60%.	Sistem menampilkan status "Indikasi Kuat Diabetes" dengan persen probabilitas.	Berhasil
3	Validasi Input Kosong	Mengosongkan salah satu kolom.	Browser menghentikan pengiriman data dan memunculkan alert atau peringatan bahwa kolom wajib diisi.	Sistem menahan proses dan memunculkan notifikasi form harus diisi.	Berhasil

4	Validasi Batasan Nilai (<i>Constraint</i>)	Memasukkan nilai negatif pada kolom numerik.	Sistem menolak input tersebut karena melanggar batasan nilai.	Sistem memunculkan pesan validasi agar nilai harus lebih besar sama dengan.	Berhasil
5	Reset & Prediksi Baru	Klik tombol "Cek Data Lain" pada halaman hasil.	Sistem mengarahkan kembali ke formulir kosong untuk melakukan pengecekan data lain.	Halaman berhasil di-reset dan siap menerima input data baru.	Berhasil

ada dasarnya disarankan untuk tidak menggunakan *numbering* (1,2,3...,a,b,c dst) dalam pembahasan naskahnya, ubah menjadi dalam bentuk kalimat. Hindari menggunakan *Bullet*/daftar berurut dengan simbol *, √ dan lainnya.

4. KESIMPULAN

Keseluruhan tahapan penelitian ini membuktikan bahwa integrasi algoritma *Random Forest* yang dioptimasi dengan teknik *Synthetic Minority Over-sampling Technique* (SMOTE) dan *Grid Search* telah berhasil diimplementasikan sebagai instrumen deteksi dini penyakit diabetes. Penerapan teknik SMOTE terbukti sangat krusial dalam mengatasi kendala ketidakseimbangan kelas (*imbalanced data*) pada dataset rekam medis. Model klasifikasi akhir yang dikembangkan mampu mencapai tingkat akurasi keseluruhan sebesar 95,8% dengan nilai *Recall* untuk pasien diabetes sebesar 75,0%. Keberhasilan model dalam mendeteksi 1.271 kasus secara tepat (*True Positive*) dan menekan angka *False Negative* menjadi 425 kasus membuktikan bahwa sistem ini memiliki sensitivitas yang memadai untuk mengenali pengguna yang berisiko. Keputusan ini sangat sejalan dengan prinsip preventif medis pada sistem skrining awal, di mana lebih baik pasien berisiko menjalani pemeriksaan klinis lanjutan daripada dibiarkan luput dari diagnosis. Pengembangan kedalam sistem aplikasi web dengan arsitektur *Client-Server* menggunakan *AstroJS* dan *Flask* juga telah berhasil menjembatani kebutuhan antara kecanggihan model komputasi dengan kemudahan akses praktis bagi pengguna. Hasil pengujian *Black Box* menunjukkan bahwa sistem berjalan dengan valid dan stabil dalam menangani berbagai skenario masukan data serta mampu memberikan respons prediksi secara *real-time*. Meskipun model ini memiliki performa yang sangat baik, penelitian ini memiliki keterbatasan pada ketergantungan terhadap data sekunder. Oleh karena itu, saran untuk penelitian selanjutnya adalah melakukan pengujian menggunakan data klinis lokal dari institusi kesehatan di Indonesia untuk meningkatkan validitas model terhadap populasi spesifik. Sistem ini diharapkan dapat terus dikembangkan menjadi alat bantu skrining mandiri yang handal dalam upaya menekan angka komplikasi diabetes di masyarakat.

DAFTAR PUSTAKA

- [1] F. Marwati dan R. Fauzi, "Prediksi Penyakit Diabetes Melitus Menggunakan Jaringan Syaraf Tiruan dengan Metode Backpropagation," *Jurnal Informatika Utama*, vol. 2, no. 1, hlm. 26–34, Mei 2024, doi: 10.55903/jitu.v2i1.163.
- [2] Y. S. M. Silitonga dkk., "PENDEKATAN KEDOKTERAN KELUARGA PADA DIABETES MELLITUS DENGAN KOMPLIKASI KRONIS," vol. 1, 2024.
- [3] X. Lin dkk., "Global, regional, and national burden and trend of diabetes in 195 countries and territories: an analysis from 1990 to 2025," *Scientific Reports*, vol. 10, no. 1, hlm. 14790, Sep 2020, doi: 10.1038/s41598-020-71908-9.
- [4] K. L. Ong dkk., "Global, regional, and national burden of diabetes from 1990 to 2021, with projections of prevalence to 2050: a systematic analysis for the Global Burden of Disease Study 2021," *The Lancet*, vol. 402, no. 10397, hlm. 203–234, Jul 2023, doi: 10.1016/S0140-6736(23)01301-6.
- [5] U. Ahmed dkk., "Prediction of Diabetes Empowered With Fused Machine Learning," *IEEE Access*, vol. 10, hlm. 8529–8538, 2022, doi: 10.1109/ACCESS.2022.3142097.
- [6] M. Wahidin dkk., "Projection of diabetes morbidity and mortality till 2045 in Indonesia based on risk factors and NCD prevention and control programs," *Sci Rep*, vol. 14, no. 1, hlm. 5424, Mar 2024, doi: 10.1038/s41598-024-54563-2.
- [7] D. W. Soeatmadji, R. Rosandi, M. R. Saraswati, R. P. Sibarani, dan W. O. Tarigan, "Clinicodemographic Profile and Outcomes of Type 2 Diabetes Mellitus in the Indonesian Cohort of DISCOVER: A 3-Year Prospective Cohort Study.," *J ASEAN Fed Endocr Soc*, vol. 38, no. 1, hlm. 68–74, 2023, doi: 10.15605/jafes.038.01.10.
- [8] J. Zhang, Z. Zhang, K. Zhang, X. Ge, R. Sun, dan X. Zhai, "Early detection of type 2 diabetes risk: limitations of current diagnostic criteria," *Front Endocrinol (Lausanne)*, vol. 14, hlm. 1260623, Nov 2023, doi: 10.3389/fendo.2023.1260623.
- [9] A. Prastyo, S. Sutikno, dan K. Khadijah, "Improving support vector machine and backpropagation performance for diabetes mellitus classification," *Computer Science and Information Technologies*, vol. 5, no. 2, hlm. 140–149, Jul 2024, doi: 10.11591/csit.v5i2.pp140-149.
- [10] M. Nur, "Data Mining Untuk Memprediksi Kelulusan Mahasiswa Jurusan Teknik Informatika UIN Syarif Hidayatullah Jakarta Menggunakan Metode Klasifikasi C4.5," bachelorThesis, Fakultas Sains dan Teknologi Universitas Islam Negeri Syarif Hidayatullah Jakarta, 2022. Diakses: 14 Juli 2025. [Daring]. Tersedia pada: <https://repository.uinjkt.ac.id/dspace/handle/123456789/65006>
- [11] A. P. Silalahi, H. G. Simanullang, dan M. I. Hutapea, "SUPERVISED LEARNING METODE K-NEAREST NEIGHBOR UNTUK PREDIKSI DIABETES PADA WANITA," *METHOMIKA: Jurnal Manajemen Informatika & Komputerisasi Akuntansi*, vol. 7, no. 1, Art. no. 1, Apr 2023, doi: 10.46880/jmika.Vol7No1.pp144-149.
- [12] S. U. Putri, E. Irawan, dan F. Rizky, *Kesatria : Jurnal Penerapan Sistem Informasi (Komputer dan Manajemen)*, vol. 2, no. 1, Art. no. 1, Jan 2021, doi: 10.30645/kesatria.v2i1.56.
- [13] S. Samet, M. R. Laouar, I. Bendib, dan S. Eom, "Analysis and Prediction of Diabetes Disease Using Machine Learning Methods.," *International Journal of Decision Support System Technology*, vol. 14, no. 1, hlm. 1–19, Jul 2022, doi: 10.4018/IJDSST.303943.
- [14] E. Priande dan G. M. Zamroni, "PENGEMBANGAN SISTEM PREDIKSI HARGA MOBIL BEKAS OLX MENGGUNAKAN ALGORITMA RANDOM FOREST," vol. 12, no. 1, 2024.
- [15] N. Anisa dan A. Kurniawan, "The Diabetes Prediction Using Flask and Decision Tree Classifier with Cross-Validation: Prediksi Diabetes Menggunakan Flask dan Decision Tree Classifier dengan Validasi Silang," *Install: Information System and Technology Journal*, vol. 1, no. 1, hlm. 35–44, Jun 2024, doi: 10.33859/install.v1i1.548.
- [16] D. Lal dan A. V. S., "Exploring the Efficacy of Machine Learning Algorithms for Diabetes Prediction: A Comparative Prediction," *IJRASET*, vol. 11, no. 5, hlm. 2904–2910, Mei 2023, doi: 10.22214/ijraset.2023.51565.
- [17] F. Alghifari dan D. Juardi, "PENERAPAN DATA MINING PADA PENJUALAN MAKANAN DAN MINUMAN MENGGUNAKAN METODE ALGORITMA NAÏVE BAYES: Studi Kasus : Makan Barbeque Sepuasnya," *JURNAL ILMIAH INFORMATIKA*, vol. 9, no. 02, hlm. 75–81, Sep 2021, doi: 10.33884/jif.v9i02.3755.
- [18] Y. Zhao dkk., "On the Impact of Sample Duplication in Machine-Learning-Based Android Malware Detection," *ACM Trans. Softw. Eng. Methodol.*, vol. 30, no. 3, hlm. 1–38, Jul 2021, doi: 10.1145/3446905.
- [19] J. Kuswanto dan L. Hakim, "Penerapan Algoritma Random Forest untuk memprediksi Performa Akademik Mahasiswa," 2025.
- [20] J. Halif, D. Wahiddin, I. Sanjaya, dan S. Faisal, "Model Regresi Linear Berganda untuk Prediksi Tingkat Pengangguran di Provinsi Jawa Barat," *Jurnal Algoritma*, vol. 22, no. 1, hlm. 324–335, Mei 2025, doi: 10.33364/algoritma/v.22-1.2312.
- [21] M. Faisal dan I. B. Santoso, "Algoritma Random Forest dan Synthetic Minority Oversampling Technique (SMOTE) untuk Deteksi Diabetes".
- [22] R. Saputra dan A. J. P. Sibarani, "Implementasi Data Mining Menggunakan Algoritma Apriori Untuk Meningkatkan Pola Penjualan Obat | JATISI," Agu 2020, Diakses: 2 Maret 2026. [Daring]. Tersedia pada: <https://jurnal.mdp.ac.id/index.php/jatisi/article/view/195>
- [23] U. K. J. Dn dan M. Rahardi, "Analysis of SMOTE and Random Search on Machine Learning Algorithms for Stroke Disease Diagnosis," vol. 10, no. 1.
- [24] M. F. Martias, J. Jasril, S. Sanjaya, L. Handayani, dan F. Yanto, "Klasifikasi Citra Daging Sapi dan Daging Babi Menggunakan CNN Arsitektur EfficientNet-B6 dan Augmentasi Data," *Jurnal Sistem Komputer dan Informatika (JSON)*, vol. 4, no. 4, Art. no. 4, Jun 2023, doi: 10.30865/json.v4i4.6195.
- [25] D. Normawati dan S. A. Prayogi, "Implementasi Naïve Bayes Classifier Dan Confusion Matrix Pada Analisis Sentimen Berbasis Teks Pada Twitter," *J-SAKTI (Jurnal Sains Komputer dan Informatika)*, vol. 5, no. 2, Art. no. 2, Sep 2021, doi: 10.30645/j-sakti.v5i2.369.
- [26] E. A. Novia, W. I. Rahayu, dan C. Prianto, *Sistem perbandingan algoritma k-means dan naïve bayes untuk memprediksi prioritas pembayaran tagihan rumah sakit berdasarkan tingkat kepentingan*. Kreatif, 2021. [Daring]. Tersedia pada: <https://books.google.co.id/books?id=MND9DwAAQBAJ>
- [27] V. M. Putri, B. A. Wisesa, I. A. Eddyul, dan S. A. Darma, "Sistem Pakar Deteksi Keterlambatan Bicara Anak Menggunakan Forward Chaining dan Naïve Bayes," no. 2, 2025.