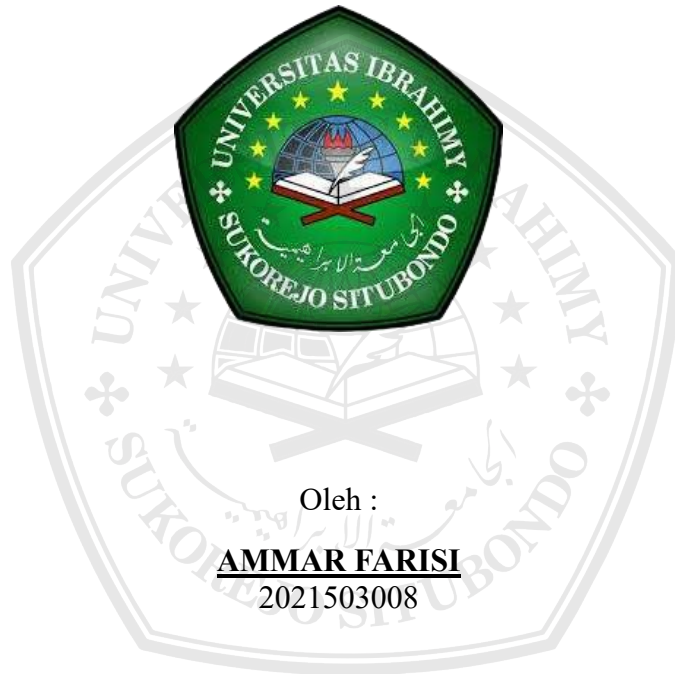


**PREDIKSI PENYAKIT DIABETES MENGGUNAKAN ALGORITMA
SUPPORT VECTOR MACHINE (SVM)**

SKRIPSI



Oleh :

AMMAR FARISI

2021503008

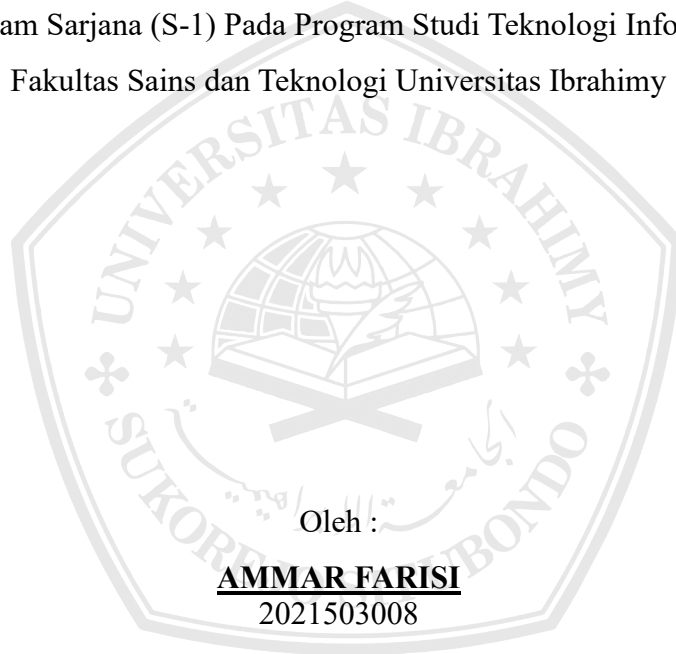
**PROGRAM STUDI TEKNOLOGI INFORMASI
FAKULTAS SAINS DAN TEKNOLOGI UNIVERSITAS IBRAHIMY
SITUBONDO**

2025

**PREDIKSI PENYAKIT DIABETES MENGGUNAKAN ALGORITMA
SUPPORT VECTOR MACHINE (SVM)**

SKRIPSI

Diajukan Untuk Memenuhi Salah Satu Persyaratan dalam Menyelesaikan
Program Sarjana (S-1) Pada Program Studi Teknologi Informasi
Fakultas Sains dan Teknologi Universitas Ibrahimy



Oleh :

AMMAR FARISI
2021503008

**PROGRAM STUDI TEKNOLOGI INFORMASI
FAKULTAS SAINS DAN TEKNOLOGI UNIVERSITAS IBRAHIMY
SITUBONDO**

2025

PERSETUJUAN PEMBIMBING

Nama : Ammar Farisi
NPM/NIM : 2021503008
Judul : Prediksi Penyakit Diabetes Menggunakan Algoritma Support
Vector Machine (SVM)

Telah disetujui oleh :

Pembimbing I


Ahmad Homaidi, M.Kom
NIDN : 0705078901

Pembimbing II


Hermanto, M.Kom
NIDN : 0708087807

PENGESAHAN

SKRIPSI

**PREDIKSI PENYAKIT DIABETES MENGGUNAKAN ALGORITMA
SUPPORT VECTOR MACHINE (SVM)**

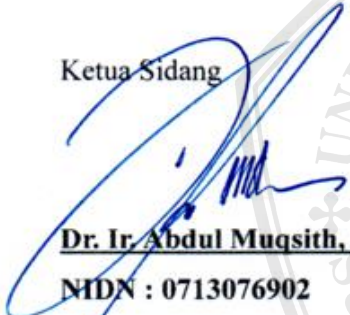
AMMAR FARISI

2021503008

Telah dipertahankan di depan dewan penguji Sidang Munaqasyah Skripsi pada hari Kamis, Tanggal 31 Juli 2025 sebagai salah satu syarat memperoleh gelar Sarjana (S.Kom) pada Fakultas Sains dan Teknologi Universitas Ibrahimiy

Tim Penguji,

Ketua Sidang



Dr. Ir. Abdul Muqsith, M.Ling

NIDN : 0713076902

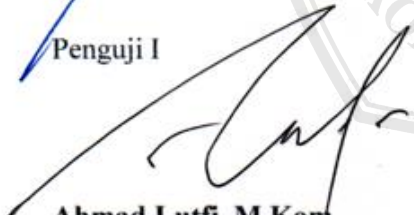
Sekretaris Sidang



Ach. Syaiful Jazil, S.Kom

NIDN : -

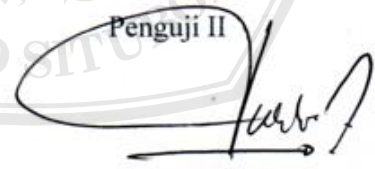
Penguji I



Ahmad Lutfi, M.Kom

NIDN : 0714108803

Penguji II



Akhlis Munazilin, S.Kom., M.T

NIDN : 0712098601

Mengetahui

Dekan,



Abd. Chofur, M.Kom

NIDN : 0711088303

MOTTO

إِنَّ مَعَ الْعُسْرِ يُسْرًا

“Sesungguhnya bersama kesulitan ada kemudahan”

(QS. Al-Insyirah ayat: 6)



KATA PENGANTAR

Segala puji syukur peneliti sampaikan kepada Allah SWT, karena atas Rahmat dan Hidayah-Nya, perencanaan, pelaksanaan dan penyelesaian tugas akhir/skripsi dengan judul “Prediksi Penyakit Diabetes Menggunakan Algoritma *Support Vector Machine (SVM)*” sebagai salah satu syarat penyelesaian program diploma/sarjana dapat terselesaikan dengan baik dan lancar, oleh karena itu kami mengucapkan terima kasih kepada :

1. KHR. Ahmad Azaim Ibrahimi, S.Sy., M.H. selaku Pengasuh Pondok Pesantren Salafiyah Syafi'iyah Sukorejo.
2. Bapak KH. Ahmad Fadloil, S.H., M.H. selaku Rektor Universitas Ibrahimi Situbondo.
3. Bapak Abd. Ghofur, M. Kom., selaku Dekan Fakultas Sains dan Teknologi Universitas Ibrahimi
4. Bapak Dr. Ach. Khumaidi, M.P., selaku Wakil Dekan I Fakultas Sains dan Teknologi Universitas Ibrahimi.
5. Bapak Abd. Wafi, M.P., selaku Wakil Dekan II Fakultas Sains dan Teknologi Universitas Ibrahimi.
6. Bapak Ahmad Lutfi, M. Kom., selaku Wakil Dekan III Fakultas Sains dan Teknologi Universitas Ibrahimi.
7. Bapak Firman Santoso, M. Kom., selaku Ketua Program Studi Teknologi Informasi.
8. Bapak Ahmad Homaidi, M. Kom., selaku Dosen Pembimbing I yang telah membimbing saya untuk penyelesaian skripsi ini.
9. Bapak Hermanto, M. Kom., selaku Dosen Pembimbing II yang telah membimbing saya untuk penyelesaian skripsi ini.
10. Seluruh Dosen Fakultas Sains dan Teknologi Universitas Ibrahimi yang telah memberikan ilmu sehingga saya dapat menyelesaikan Tugas Akhir pada tahun ini.

11. Seluruh Civitas Akademika Fakultas Sains dan Teknologi Universitas
Ibrahimi Situbondo.

Situbondo, 29 Juli 2025
Peneliti

Ammar Farisi



PERSEMBAHAN

Saya persembahkan laporan ini kepada orang-orang yang telah membantu saya dalam menyelesaikan tugas akhir skripsi saya ini :

1. KHR. Ahmad Azaim Ibrahimi, S.Sy., M.H. selaku pengasuh pondok pesantren salafiyah syafi'iyah sukorejo.
2. Kedua Orang tua saya Bpk. Sabihi Dan Ibu Nurul Hikmah yang tak hentinya berdo'a dan berjuang serta mensupport demi masa depan saya yang lebih baik.
3. Saudari-saudari saya MbK Salsabila Adek Yuha Ilmalana dan Adek Fariza Khansa Mecca yang paling saya cintai dan banggakan.
4. Pembimbing I Bpk. Ahmad Homaidi, M.Kom dan Pembimbing II Bpk. Hermanto, M.Kom yang selalu membimbing dan mengarahkan hingga laporan Skripsi ini selesai.
5. Seluruh asrama daerah sunan ampel wabil khusus asrama A.25
6. Sahabat-sahabat saya yang telah memberikan dukungan dalam suka maupun duka.
7. Semua orang yang tidak bisa saya sebut satu persatu yang telah mendukung saya sepenuhnya sehingga bisa menyelesaikan laporan Skripsi ini dengan baik.

DAFTAR ISI

PERSETUJUAN PEMBIMBING	ii
PENGESAHAN	iii
MOTTO	iv
KATA PENGANTAR.....	v
PERSEMBAHAN.....	vii
DAFTAR ISI	viii
DAFTAR GAMBAR	ix
DAFTAR TABEL	x
DAFTAR SEGMENT PROGRAM.....	xii
DAFTAR LAMPIRAN	xiii
ABSTRAK	xiv
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Identifikasi Masalah	3
1.3 Rumusan Masalah	3
1.4 Batasan Masalah	3
1.5 Tujuan Penelitian	4
1.6 Manfaat Penelitian.....	4
1.7 Metode Penelitian.....	5
1.7.1 Jenis Penelitian.....	5
1.7.2 Teknik Pengumpulan Data.....	5
1.7.3 Metode Pengembangan Sistem	6
1.8 Sistematika Pembahasan	8
BAB II TINJAUAN PUSTAKA	10
2.1 Penelitian Terdahulu.....	10
2.2 Landasan Teori.....	13
2.3 Perangkat Lunak yang Digunakan	16
BAB III ANALISIS PERANCANGAN SISTEM.....	20

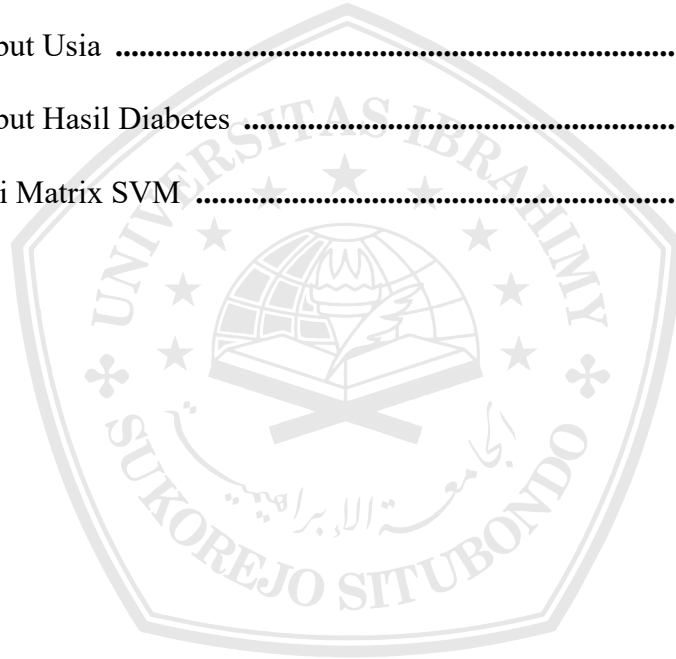
3.1	Analisis Sistem.....	20
3.1.1	Analisis Masalah	20
3.1.2	Analisis Kebutuhan Sistem	21
3.1.3	Analisis Data.....	22
3.2	Perancangan Sistem.....	25
3.2.1	Arsitektur Aplikasi.....	25
3.2.2	Perancangan Model Machine Learning.....	28
3.3	Implementasi Sistem	28
3.3.1	<i>Library</i> Yang Digunakan.....	28
3.3.2	Implementasi <i>Streamlit</i>	31
3.3.2	Uji Coba Sistem oleh Pengguna (User Validator)	34
BAB IV HASIL & PEMBAHASAN		36
4.1	Deskripsi Dataset	36
4.2	Preprocessing.....	41
4.2.1	Missing Value.....	41
4.2.2	Drop Data.....	42
4.2.3	Splitting Dataset	43
4.3	Implementasi Algoritma	44
4.3.1	Support vector machine.....	44
4.4	Implementasi <i>Framework Streamlit</i>	47
4.4.1	Tampilan Antarmuka <i>Web</i>	49
4.4.2	Analisis Hasil Model	50
BAB V PENUTUP		53
5.1	Kesimpulan.....	53
5.2	Saran	54
DAFTAR PUSTAKA.....		55
CURICULUM VITAE		58
LAMPIRAN-LAMPIRAN		59

DAFTAR GAMBAR

Gambar 1. 1 Knowledge Discovery In Database	8
Gambar 3. 1 Arsitektur Aplikasi	26
Gambar 3. 2 Algoritma Support Vector Machine (SVM)	28
Gambar 3. 3 Aplikasi Streamlit.....	32
Gambar 3. 4 Perancangan Desain Prediksi Diabetes	34
Gambar 3. 5 Xampp Control Panel.....	35
Gambar 4. 1 Dataset Diabetes.CSV	38
Gambar 4. 2 Grafik Data Diabetes.....	41
Gambar 4. 3 Nilai Confusion Matrix SVM.....	46
Gambar 4. 4 Tampilan Antarmuka Web.....	49
Gambar 4. 5 Tampilan Database PHP/MyAdmin.....	52

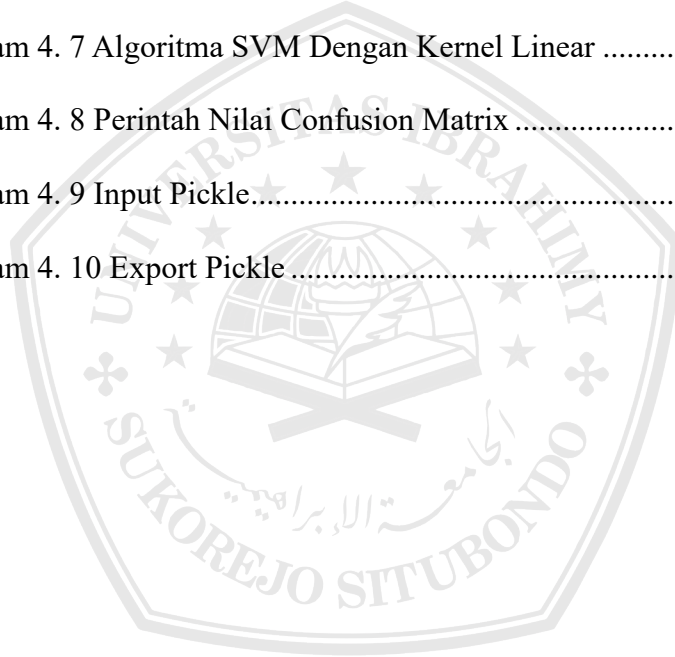
DAFTAR TABEL

Tabel 3. 1 Atribut Kehamilan	22
Tabel 3. 2 Atribut Gula Darah	23
Tabel 3. 3 Atribut Tekanan Darah	23
Tabel 3. 4 Atribut Ketebalan Kulit	23
Tabel 3. 5 Atribut Insulin	24
Tabel 3. 6 Atribut BMI	24
Tabel 3. 7 Atribut Fungsi Selisih Diabetes	24
Tabel 3. 8 Atribut Usia	25
Tabel 3. 9 Atribut Hasil Diabetes	25
Tabel 4. 1 Nilai Matrix SVM	47



DAFTAR SEGMENT PROGRAM

Segmen Program 4. 1 Read Dataset.....	39
Segmen Program 4. 2 Menampilkan Chart Data Diabetes	40
Segmen Program 4. 3 Shape Data.....	42
Segmen Program 4. 4 Drop Data	43
Segmen Program 4. 5 Splitting Data.....	43
Segmen Program 4. 6 Training data Algoritma SVM.....	45
Segmen Program 4. 7 Algoritma SVM Dengan Kernel Linear	45
Segmen Program 4. 8 Perintah Nilai Confusion Matrix	46
Segmen Program 4. 9 Input Pickle.....	48
Segmen Program 4. 10 Export Pickle	48



DAFTAR LAMPIRAN

Lampiran 1 Dataset Diabetes	59
Lampiran 2 Kartu Bimbingan	60
Lampiran 3 SKHCP	61
Lampiran 4 LOA (Letter Of Acceptance)	62
Lampiran 5 Segmen Program Streamlit.....	67



ABSTRAK

Ammar Farisi. 2025. **Prediksi Penyakit Diabetes Menggunakan Algoritma *Support Vector Machine (SVM)***. Skripsi, Program Studi Teknologi Informasi Fakultas Sains dan Teknologi Universitas Ibrahimy.
Pembimbing (I) Ahmad Homaidi, M.Kom., (II) Hermanto, M.Kom

Diabetes melitus merupakan penyakit metabolik kronis yang berpotensi menimbulkan komplikasi serius apabila tidak segera terdeteksi. Penelitian ini bertujuan untuk merancang sistem prediksi diabetes berbasis algoritma *Support Vector Machine (SVM)* yang diimplementasikan dalam bentuk aplikasi *web* menggunakan framework *Streamlit*. Data yang digunakan berasal dari Pima Indians Diabetes Dataset yang mencakup 768 entri pasien dengan delapan atribut medis utama, seperti kadar glukosa, tekanan darah, dan indeks massa tubuh (BMI). Proses penelitian meliputi tahapan pra-pemrosesan data, pelatihan model menggunakan *kernel linear*, serta evaluasi performa model dengan metrik akurasi, presisi, dan recall. Berdasarkan hasil pengujian, model SVM mampu mencapai akurasi sebesar 72%, presisi 73%, dan recall 50%. Aplikasi *web* yang dikembangkan memungkinkan pengguna untuk menginput data kesehatan secara mandiri dan memperoleh prediksi risiko diabetes secara instan. Dengan demikian, sistem ini diharapkan dapat menjadi alat deteksi dini yang praktis dan efisien, serta mendukung langkah *preventif* dalam bidang kesehatan masyarakat.

Kata Kunci : Diabetes, *Machine Learning*, *Support Vector Machine*, Prediksi Penyakit, *Streamlit*.

ABSTRACT

Ammar Farisi. 2025. **Prediksi Penyakit Diabetes Menggunakan Algoritma Support Vektor Machine (SVM)**. *Thesis, Information Technology Study Program, Faculty of Science and Technology, Ibrahimi University.*
Supervisor (I) Ahmad Homaidi, M.Kom., (II) Hermanto, M.Kom.

Diabetes mellitus is a chronic metabolic disorder that can lead to serious complications if not detected early. This study aims to develop a diabetes prediction system using the Support Vector Machine (SVM) algorithm, implemented as a web application through the Streamlit framework. The dataset used is the Pima Indians Diabetes Dataset, consisting of 768 patient records with eight key medical attributes, such as glucose level, blood pressure, and body mass index (BMI). The research process includes data preprocessing, model training using a linear kernel, and performance evaluation using accuracy, precision, and recall metrics. The evaluation results show that the SVM model achieved 72% accuracy, 73% precision, and 50% recall. The developed web application enables users to input their health data independently and receive real-time predictions of their diabetes risk. Thus, this system is expected to serve as a practical and efficient early detection tool and contribute to preventive efforts in public health.

Keywords : *Diabetes, Machine Learning, Support Vector Machine, Disease Prediction, Streamlit*

BAB I PENDAHULUAN

1.1 Latar Belakang

Diabetes melitus adalah penyakit yang umum namun serius di seluruh dunia. Diabetes melitus adalah kondisi yang memengaruhi cara tubuh memproses gula. Kondisi ini terjadi ketika tubuh tidak memproduksi cukup insulin atau tidak menggunakan insulin dengan benar.[1]. Insulin adalah hormon penting dalam tubuh yang membantu mengelola gula darah, yaitu gula yang digunakan tubuh untuk energi. Jika insulin tidak bekerja dengan baik atau jumlahnya tidak mencukupi, kadar gula darah dapat meningkat drastis, yang menyebabkan masalah yang disebut *hiperglikemia*. [2].

Hiperglikemia yang berlangsung dalam jangka panjang dapat memicu berbagai komplikasi serius pada tubuh. Organ vital seperti mata, ginjal, jantung, sistem saraf, dan pembuluh darah berisiko mengalami kerusakan berat. Dalam beberapa kasus, kondisi ini dapat berujung pada kebutaan, gagal ginjal, serangan jantung, bahkan amputasi. Sayangnya, banyak penderita diabetes tidak menyadari keberadaan penyakit ini karena gejalanya cenderung berkembang secara perlahan dan tidak terlalu kentara pada tahap awal.[3]. Akibatnya, banyak penderita baru terdiagnosis ketika penyakit sudah memasuki *fase* komplikasi. Menurut data dari *International Diabetes Federation* (IDF), pada tahun 2019 tercatat sekitar 433 juta orang di seluruh dunia hidup dengan diabetes. Jumlah ini sangat signifikan dan diperkirakan akan terus bertambah, mencapai 578 juta pada tahun 2030, dan berpotensi menembus 700 juta kasus pada tahun 2045. Fakta ini menjadikan diabetes sebagai salah satu isu kesehatan global yang paling mengkhawatirkan. [4].

Indonesia menempati posisi sepuluh besar negara dengan jumlah penderita diabetes tertinggi di dunia. Tingginya angka ini dipengaruhi oleh berbagai faktor, antara lain gaya hidup yang kurang sehat, konsumsi makanan tinggi gula dan lemak, minimnya aktivitas fisik, serta rendahnya kesadaran masyarakat untuk melakukan pemeriksaan kesehatan secara rutin. Selain itu, akses pelayanan kesehatan yang belum merata dan tingginya biaya pemeriksaan medis turut menjadi hambatan dalam penanganan penyakit ini.[5]. Tidak semua lapisan masyarakat, khususnya mereka yang tinggal di daerah pedesaan atau pinggiran kota, memiliki kemudahan akses untuk melakukan pemeriksaan kadar gula darah secara rutin. Oleh karena itu, dibutuhkan suatu sistem deteksi dini yang mampu membantu masyarakat dalam mengidentifikasi risiko diabetes secara mandiri dan cepat, tanpa harus mengunjungi fasilitas layanan kesehatan. Mengingat pentingnya deteksi sejak dini, diperlukan sebuah aplikasi prediktif yang dapat digunakan secara mandiri oleh masyarakat.[6].

Kemajuan teknologi yang pesat telah menghadirkan solusi yang cukup potensial di bidang medis, salah satunya adalah penerapan *machine learning*. Di antara berbagai algoritma yang tersedia, *Support Vector Machine* (SVM) merupakan salah satu yang terbukti efektif dalam melakukan proses klasifikasi dan prediksi.[7]. Algoritma ini berfungsi dengan menentukan *hyperplane* optimal yang mampu memisahkan data ke dalam dua kategori, seperti membedakan antara individu yang berisiko terkena diabetes dan yang tidak. Dengan mengombinasikan algoritma ini bersama dataset diabetes yang tersedia secara publik di *platform* seperti *Kaggle*, memungkinkan pengembangan model prediksi yang memiliki tingkat akurasi tinggi.[8]. Model prediksi ini dapat diintegrasikan ke dalam aplikasi

web sederhana menggunakan *Streamlit*, sehingga masyarakat dapat mengakses dan menggunakannya dengan mudah. Pengguna hanya perlu menginput sejumlah data dasar seperti usia, kadar glukosa, tekanan darah, dan indeks massa tubuh. Setelah itu, sistem akan mengolah informasi tersebut dan memberikan hasil prediksi mengenai kemungkinan seseorang berisiko menderita diabetes atau tidak.

1.2 Identifikasi Masalah

Berdasarkan uraian latar belakang dan identifikasi masalah tersebut, maka rumusan masalah dalam penelitian ini adalah :

1. Kesulitan dalam mengklasifikasi penyakit diabetes secara dini, karena penyakit ini berisiko tinggi dan sulit terdeteksi di tahap awal.
2. Belum tersedia implementasi model prediksi diabetes menggunakan algoritma *Support Vector Machine (SVM)* dalam bentuk aplikasi *web* yang praktis dan mudah digunakan oleh masyarakat umum.

1.3 Rumusan Masalah

Berdasarkan uraian latar belakang dan identifikasi masalah tersebut, maka rumusan masalah dalam penelitian ini adalah :

Bagaimana menerapkan algoritma *Support Vector Machine (SVM)* untuk klasifikasi penyakit diabetes menggunakan bahasa pemrograman *python*?

1.4 Batasan Masalah

Berdasarkan perumusan diatas maka Batasan masalah dari penelitian ini adalah :

1. Dataset yang digunakan dalam penelitian ini adalah Pima Indians Diabetes Dataset yang terdiri dari 8 atribut input dan 2 label kelas yaitu “Yes” dan “No”. Atribut input tersebut meliputi: jumlah kehamilan (*Pregnancies*), kadar

glukosa (*Glucose*), tekanan darah (*BloodPressure*), ketebalan kulit (*SkinThickness*), kadar insulin (*Insulin*), indeks massa tubuh (*BMI*), riwayat keturunan diabetes (*Diabetes Pedigree Function*), dan usia (*Age*).

2. Penelitian ini menggunakan algoritma *Support Vector Machine (SVM)*, dengan evaluasi performa berdasarkan nilai akurasi, presisi, dan recall, Aplikasi dibangun menggunakan *framework Streamlit* dengan antarmuka berbasis web.

1.5 Tujuan Penelitian

Berdasarkan perumusan masalah yang telah dipaparkan maka tujuan dari penelitian ini adalah :

Untuk mengklasifikasi apakah seseorang menderita diabetes atau tidak, berdasarkan parameter medis seperti kadar glukosa, tekanan darah, dan indeks massa tubuh (*BMI*).

1.6 Manfaat Penelitian

Berdasarkan penjelasan diatas, maka didapatkan manfaat dari penelitian ini yaitu :

1. Memberikan solusi alternatif dalam deteksi dini penyakit diabetes secara mandiri melalui teknologi *machine learning* yang akurat dan efisien.
2. Mendukung pengembangan teknologi informasi di bidang kesehatan, khususnya dalam penerapan algoritma *SVM* untuk klasifikasi penyakit berbasis data.

1.7 Metode Penelitian

Metode penelitian adalah pendekatan sistematis yang digunakan untuk mengumpulkan, menganalisis, dan mengolah data guna mencapai tujuan penelitian secara ilmiah. Pada penelitian ini, metode yang diterapkan bertujuan untuk membangun model klasifikasi diabetes yang akurat dan efisien, serta mengimplementasikannya ke dalam aplikasi *web*. Seluruh prosedur dilakukan secara terstruktur dan terencana guna memperoleh informasi yang relevan sebagai jawaban atas permasalahan yang telah ditentukan.[4].

1.7.1 Jenis Penelitian

Penelitian ini tergolong ke dalam jenis penelitian kuantitatif, yang bertujuan menganalisis data numerik secara objektif dengan dukungan metode statistik. Dalam studi ini, data pasien diabetes diolah menggunakan algoritma *machine learning*, yakni *Support Vector Machine* (SVM), guna membangun model klasifikasi yang mampu mengidentifikasi potensi diabetes pada individu berdasarkan sejumlah atribut tertentu. Hasil analisis tersebut kemudian dimanfaatkan untuk merancang sistem deteksi dini yang dapat diakses melalui *platform web*. [9].

1.7.2 Teknik Pengumpulan Data

Untuk memperoleh hasil penelitian, tahapan pengumpulan data dilakukan melalui langkah-langkah berikut :

1. Pengumpulan Data

Data yang digunakan dalam penelitian ini bersumber dari dataset publik yang tersedia di situs *Kaggle.com*, yaitu Pima Indians Diabetes

Dataset. Dataset ini telah umum digunakan dalam berbagai penelitian di bidang kesehatan dan memuat informasi penting seperti jumlah kehamilan, kadar glukosa, tekanan darah, kadar insulin, indeks massa tubuh (BMI), serta beberapa atribut lainnya.[10].

2. Studi Pustaka

Peneliti melakukan kajian pustaka melalui berbagai sumber seperti jurnal ilmiah, artikel, dan buku guna memperdalam pemahaman terkait penyakit diabetes, teknik klasifikasi data, serta algoritma *Support Vector Machine* (SVM). Kajian ini menjadi landasan teoritis dalam perancangan model prediksi dan pengembangan sistem aplikasi.[11].

1.7.3 Metode Pengembangan Sistem

Penelitian ini menggunakan pendekatan *Knowledge Discovery in Databases (KDD)* sebagai kerangka kerja dalam proses pengembangan sistem klasifikasi untuk mendeteksi penyakit diabetes. [12]. *Knowledge Discovery in Databases (KDD)* merupakan rangkaian proses sistematis yang bertujuan untuk mengekstraksi informasi atau pengetahuan baru dari sekumpulan data melalui beberapa langkah berikut :

1. Pemahaman Masalah dan Tujuan

Langkah pertama dimulai dengan memahami pentingnya deteksi dini penyakit diabetes dan bagaimana penerapan algoritma pembelajaran mesin dapat memberikan kontribusi dalam mengatasi masalah tersebut.

2. Seleksi dan Pengumpulan Data

Data penelitian diperoleh dari Pima Indians Diabetes Dataset yang tersedia di Kaggle.com, terdiri dari 768 data pasien dengan sejumlah atribut yang berhubungan dengan kondisi medis.

3. Preprocessing Data

Di tahap ini dilakukan pembersihan data yang berisi nilai kosong atau inkonsisten, perubahan data menjadi bentuk angka, dan proses normalisasi supaya data dapat digunakan dengan efektif dalam melatih model.

4. Transformasi dan Seleksi Fitur

Penentuan fitur didasarkan pada tingkat signifikansinya dalam deteksi diabetes, dengan menitikberatkan pada variabel seperti konsentrasi glukosa, tekanan darah, usia, dan indeks massa tubuh (BMI).

5. Pemodelan (*Data Mining*)

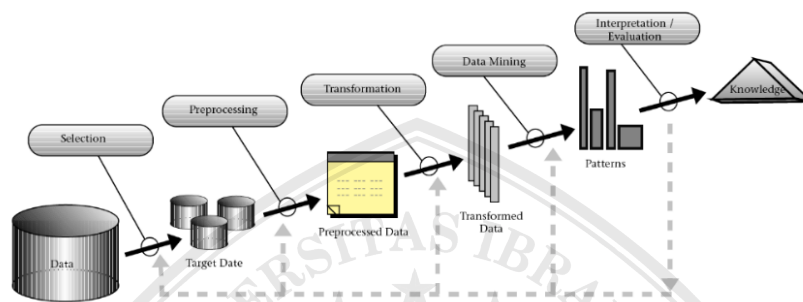
Dalam pengembangan model klasifikasi, algoritma *Support Vector Machine* (SVM) digunakan sebagai metode inti, di mana kernel *linear* dipilih sebagai parameter utama dalam tahap pelatihan model.

6. Evaluasi Model

Untuk menilai performa model dalam memprediksi risiko diabetes, digunakan sejumlah metrik evaluasi seperti akurasi, presisi, dan recall. [13].

7. Implementasi Aplikasi

Setelah melalui tahapan pelatihan dan evaluasi, model kemudian diintegrasikan ke dalam aplikasi *web* berbasis *Streamlit*, dengan tujuan agar masyarakat dapat menggunakannya secara mudah dan independen.



Gambar 1. 1 Knowledge Discovery In Database

1.8 Sistematika Pembahasan

BAB I PENDAHULUAN

Bab ini membahas latar belakang pentingnya deteksi dini penyakit diabetes dan bagaimana teknologi *machine learning*, khususnya algoritma *SVM*, dapat menjadi solusi. Selain itu, bab ini juga memuat identifikasi masalah, rumusan masalah, batasan masalah, tujuan dan manfaat penelitian, metode, serta sistematika pembahasan.

BAB II TINJAUAN PUSTAKA

Berisi kajian teori dan konsep-konsep yang relevan dengan penelitian, seperti definisi penyakit diabetes, klasifikasi dalam data mining, algoritma *Support Vector Machine (SVM)*, serta metode evaluasi performa model.

BAB III METODOLOGI PENELITIAN

Bab ini menjelaskan tahapan penelitian yang mengikuti pendekatan *KDD*. Tahapan tersebut meliputi pengumpulan dan pemilihan data, preprocessing data, pemodelan menggunakan algoritma *SVM*, evaluasi performa model, hingga proses implementasi ke dalam aplikasi berbasis *web*.

BAB IV HASIL DAN PEMBAHASAN

Bab ini menyajikan hasil dari proses preprocessing data, hasil pelatihan dan pengujian model *SVM*, serta analisis performa model berdasarkan metrik evaluasi. Juga dibahas implementasi akhir dalam bentuk aplikasi prediksi berbasis *web*.

BAB V PENUTUP

Berisi kesimpulan dari hasil penelitian yang telah dilakukan serta saran untuk pengembangan lebih lanjut, khususnya dalam peningkatan akurasi model dan pengembangan fitur-fitur aplikasi.

BAB II

TINJAUAN PUSTAKA

2.1 Penelitian Terdahulu

“Sistem Informasi Model Prediksi Penyakit Diabetes Menggunakan Algoritma *Random Forest* di *Fasyankes*”[14]

Berdasarkan penelitian yang dilakukan oleh Muhadi dan Amir Ali (2024), Pengembangan sistem informasi untuk prediksi penyakit diabetes dilakukan dengan memanfaatkan algoritma *Random Forest*. Tujuan dari penelitian ini adalah membangun model klasifikasi guna memprediksi individu yang berisiko mengidap diabetes, berdasarkan dataset Pima Indians Diabetes yang diambil dari situs *Kaggle*. Dalam proses pembagian data, digunakan proporsi 80% untuk data pelatihan dan 20% untuk data pengujian, dari total keseluruhan 768 data.

Model klasifikasi yang dikembangkan menghasilkan akurasi sebesar 76%, dengan nilai presisi 69%, recall 59%, dan F1-score 63%. Evaluasi juga dilakukan menggunakan *confusion matrix*, yang menunjukkan bahwa model cukup efektif dalam mengidentifikasi pasien diabetes, meskipun masih terjadi beberapa kesalahan dalam membedakan data positif dan negatif. Nilai *mean square error (MSE)* yang diperoleh sebesar 0,24, mengindikasikan tingkat kesalahan prediksi yang masih berada dalam batas yang dapat diterima.

Hasil klasifikasi ini membuktikan bahwa algoritma *Random Forest* dapat diterapkan secara efektif dalam memprediksi risiko diabetes berdasarkan data medis pasien. Penelitian ini juga menghadirkan penerapan sistem informasi berbasis *web* sebagai sarana pendukung deteksi dini aplikasi entri data, yang

diharapkan dapat memudahkan tenaga medis dalam mengambil keputusan klinis berbasis data.

“Klasifikasi Tingkat Risiko Diabetes Menggunakan Algoritma *Random Forest*”[15]

Menurut penelitian yang dilakukan oleh Setiawan et al. (2024), algoritma *Random Forest* dimanfaatkan untuk merancang model klasifikasi dalam mengidentifikasi risiko penyakit diabetes dengan menggunakan data yang diperoleh dari situs *Kaggle*. Dataset yang digunakan mencakup 520 data pasien, terdiri dari 16 variabel gejala dan satu label kelas. Tujuan dari penelitian ini adalah mendukung upaya pencegahan dan pengelolaan diabetes melalui pendekatan klasifikasi dalam *data mining*.

Tahapan yang dilakukan dalam penelitian ini mencakup proses preprocessing data, seperti transformasi dan pembersihan, dilanjutkan dengan pembagian data menggunakan rasio 80:20. Model kemudian dilatih menggunakan algoritma *RandomForest Classifier* dari library *sklearn*, dan dievaluasi melalui *Confusion Matrix* serta *ROC Curve*. Hasil klasifikasi menunjukkan bahwa model *Random Forest* mampu mengidentifikasi risiko diabetes dengan akurasi sebesar 98%, presisi mencapai 100%, dan recall sebesar 98%. Selain itu, nilai AUC (*Area Under Curve*) yang diperoleh sebesar 100% mengindikasikan kualitas klasifikasi yang sangat tinggi dan masuk dalam kategori '*Excellent Classification*'

Penelitian ini juga mengungkap hasil seleksi variabel yang paling berpengaruh terhadap risiko diabetes, dengan lima faktor utama yakni : *Polyuria*, *Polydipsia*, Usia, Jenis Kelamin, dan Penurunan Berat Badan Secara Tiba-tiba.

Dengan tingkat akurasi dan kemampuan klasifikasi yang tinggi, penelitian ini menyimpulkan bahwa algoritma *Random Forest* dapat dimanfaatkan secara efektif untuk memprediksi risiko diabetes pada pasien berdasarkan gejala yang telah ditentukan. Model yang dihasilkan berpotensi diterapkan dalam sistem informasi klinis guna membantu tenaga medis dalam pengambilan keputusan berbasis data.

“Deteksi Dini Penyakit Diabetes dengan Menggunakan Algoritma *Random Forest*”[16]

Berdasarkan penelitian yang dilakukan oleh Buani (2024), deteksi dini penyakit diabetes dilakukan dengan pendekatan *Knowledge Discovery in Database (KDD)* dan menggunakan sembilan algoritma klasifikasi, di antaranya *Random Forest*, *SVM*, *Logistic Regression*, *Naïve Bayes*, dan lain-lain. Penelitian ini bertujuan untuk membandingkan performa beberapa algoritma dalam melakukan klasifikasi terhadap risiko diabetes dan menentukan model terbaik untuk diterapkan.

Penelitian ini menggunakan dataset diabetes berjumlah 520 data yang berisi atribut gejala dan faktor risiko, seperti usia, jenis kelamin, polyuria, polydipsia, kelemahan, dan lain-lain. Hasil eksperimen menunjukkan bahwa algoritma *Random Forest* memiliki performa terbaik dengan akurasi sebesar 98,72%, precision 0,97, recall 1,00, f1-score 0,98, dan nilai AUC sebesar 0,98. Penilaian dilakukan dengan menggunakan *Confusion Matrix* dan *ROC Curve*. Dari keseluruhan algoritma yang diuji, *SVM* justru memperoleh akurasi terendah yaitu 60,26% dengan nilai AUC 0,50, yang dikategorikan sebagai klasifikasi gagal (*failure*). Hal ini menunjukkan

bahwa pemilihan algoritma yang tepat sangat menentukan keberhasilan klasifikasi dalam sistem prediksi penyakit diabetes.

Penelitian ini menyimpulkan bahwa *Random Forest* sangat unggul dalam melakukan klasifikasi terhadap risiko diabetes. Model yang dikembangkan melalui tahapan *KDD* ini diharapkan dapat digunakan sebagai dasar pengembangan sistem deteksi dini yang lebih praktis dan akurat untuk mendukung upaya pencegahan diabetes di Indonesia.

2.2 Landasan Teori

2.2.1 Algoritma *Support vector machine* (SVM)

Support Vector Machine (SVM) merupakan metode *machine learning* yang tangguh dan fleksibel, serta banyak digunakan dalam berbagai tugas klasifikasi dan prediksi. Sebagai algoritma *supervised learning*, SVM dilatih melalui pendekatan berbasis optimasi matematika, dengan menerapkan hipotesis yang menggunakan fungsi *linear* pada ruang fitur berdimensi tinggi.[17]. Tujuan utama dari algoritma SVM adalah menentukan *hyperplane* optimal yang dapat memisahkan data ke dalam kelas-kelas yang berbeda secara efektif, dengan memaksimalkan margin, yaitu jarak antara *hyperplane* dan titik data terdekat dari masing-masing kelas. Salah satu keunggulan utama SVM terletak pada kemampuannya dalam mengelola data berdimensi tinggi secara efisien. Melalui penerapan teknik kernel, SVM dapat secara implisit memetakan data input ke dalam ruang fitur berdimensi lebih tinggi, sehingga memungkinkan pemisahan kelas secara *linear* menjadi lebih mudah.[18].

Sebagai metode pembelajaran yang dilatih melalui algoritma berbasis optimasi dan menggunakan hipotesis dengan fungsi *linear* dalam ruang fitur berdimensi tinggi, *Support Vector Machine* (SVM) merupakan algoritma yang digunakan untuk melakukan klasifikasi dan prediksi dengan cara mencari batas pemisah yang paling optimal di antara berbagai kelas dalam sebuah dataset. Proses klasifikasi dilakukan dengan melatih model menggunakan data yang tersedia, dan model yang telah terbentuk kemudian digunakan untuk memprediksi kelas dari data baru yang belum pernah dilihat sebelumnya, proses ini dikenal sebagai pengujian data.

$$f(x_d) = \sum_{i=1}^{ns} a_i y_i \vec{x}_i \vec{x}_d + b \quad (2.1)$$

Keterangan:

$$ns = \text{Jumlah support vector} \quad (2.2)$$

$$a_i = \text{Nilai bobot setiap titik data} \quad (2.3)$$

$$y_i = \text{Data kelas} \quad (2.4)$$

$$\vec{x}_i = \text{Variabel support vector} \quad (2.5)$$

$$\vec{x}_d = \text{Data yang akan diklasifikasi} \quad (2.6)$$

$$b = \text{Nilai bias atau error} \quad (2.7)$$

2.2.2 Prediksi Penyakit

Prediksi penyakit merupakan proses untuk memperkirakan potensi seseorang mengalami atau akan mengalami suatu penyakit berdasarkan data serta informasi kesehatan yang dimiliki. Tujuan utamanya adalah mengidentifikasi risiko penyakit sejak dini, sehingga langkah pencegahan

atau intervensi dapat dilakukan lebih awal guna menghindari terjadinya komplikasi yang lebih parah di masa mendatang.[19].

Dalam praktiknya, prediksi penyakit dilakukan dengan menganalisis berbagai data yang berkaitan dengan kondisi pasien, seperti usia, jenis kelamin, riwayat medis, gejala fisik, hasil laboratorium, dan faktor risiko lainnya[4]. Data tersebut kemudian diproses menggunakan pendekatan statistik ataupun teknologi berbasis *data mining* dan *machine learning* guna menemukan pola atau tren yang tidak terlihat secara kasat mata. Hasil dari analisis ini akan digunakan untuk memprediksi apakah seseorang termasuk dalam kategori berisiko tinggi, sedang, atau rendah terhadap suatu penyakit[20].

Prediksi penyakit tidak hanya bermanfaat dalam aspek klinis, tetapi juga memiliki nilai strategis dalam sistem kesehatan masyarakat. Dengan prediksi yang akurat, tenaga medis dapat mengambil keputusan berbasis data (*data-driven decision making*) dan memberikan penanganan yang lebih tepat sasaran. Selain itu, prediksi penyakit juga mendukung implementasi layanan kesehatan preventif yang lebih efisien dan proaktif[21].

2.2.3 Diabetes

Diabetes melitus merupakan suatu gangguan metabolik yang ditandai oleh ketidakmampuan tubuh dalam memproduksi insulin atau meresponsnya dengan baik, sehingga menyebabkan kadar glukosa dalam darah tetap tinggi secara kronis (*hiperglikemia*).[22]. Insulin, Merupakan hormon yang disekresikan oleh pankreas dan berfungsi penting dalam mengatur proses

metabolisme glukosa dengan memfasilitasi penyerapan glukosa dari aliran darah. Masuk ke dalam sel tubuh untuk dimanfaatkan sebagai sumber energi. Pada individu dengan diabetes melitus, gangguan dalam produksi insulin, aksi insulin, atau keduanya menyebabkan akumulasi glukosa dalam darah, mengarah pada *hiperglikemia* kronis[23].

Hiperglikemia berkepanjangan yang terkait dengan diabetes melitus dapat menyebabkan berbagai komplikasi kesehatan yang serius dan berpotensi mengancam jiwa. Tingginya kadar glukosa darah dapat merusak pembuluh darah kecil (*mikrovaskuler*) dan besar (*makrovaskuler*), mengarah ke komplikasi seperti retinopati diabetik, nefropati, neuropati, dan penyakit kardiovaskular. Komplikasi mikrovaskuler dapat memengaruhi mata, ginjal, dan saraf, menyebabkan gangguan penglihatan, gagal ginjal, dan hilangnya sensasi atau nyeri pada ekstremitas. Komplikasi makrovaskuler meliputi penyakit jantung koroner, stroke, dan penyakit arteri perifer, yang dapat meningkatkan risiko serangan jantung, stroke, dan masalah sirkulasi[24].

2.3 Perangkat Lunak yang Digunakan

2.3.1 Python Notebook

Python telah muncul sebagai salah satu bahasa pemrograman paling populer dan serbaguna di dunia, dengan aplikasi yang luas di berbagai domain, termasuk *data mining*, *machine learning*, pengembangan *web*, dan komputasi ilmiah. Kesederhanaan sintaksis, keterbacaan, dan ekosistem yang kaya dari pustaka dan kerangka kerja pihak ketiga menjadikan *Python* sebagai pilihan utama bagi banyak pengembang dan ilmuwan data[13]. Dalam

konteks *data mining* dan *machine learning*, *Python* menawarkan berbagai pustaka yang kuat seperti *scikit-learn*, *pandas*, dan *TensorFlow*, yang menyediakan alat dan algoritma canggih untuk pra-pemrosesan data, pemodelan prediktif, dan visualisasi[25].

Jupyter Notebook, sebelumnya dikenal sebagai *IPython Notebook*, adalah lingkungan pengembangan interaktif berbasis web yang telah merevolusi cara ilmuwan data, peneliti, dan pengembang berinteraksi dengan kode *Python*. *Jupyter Notebook* menggabungkan eksekusi kode langsung, visualisasi data yang kaya, dan teks naratif dalam satu dokumen, memungkinkan pengguna untuk membuat dan berbagi dokumen yang dapat dieksekusi dan dengan mudah direproduksi yang menggabungkan kode, output, penjelasan, dan multimedia. Antarmuka *notebook* berbasis sel memfasilitasi eksplorasi data iteratif, eksperimen, dan kolaborasi, menjadikannya alat yang tak ternilai untuk *prototipe*, *debugging*, dan komunikasi hasil[26].

2.3.2 *Cursor Editor*

Cursor code editor merupakan inovasi terbaru dalam dunia pengembangan perangkat lunak yang menggabungkan fungsionalitas editor kode tradisional dengan kecerdasan buatan (AI). Dikembangkan oleh tim yang dipimpin oleh mantan insinyur OpenAI, *Cursor* dirancang untuk meningkatkan produktivitas dan efisiensi pengembang dalam menulis, memahami, dan mendebug kode. Fitur utama *Cursor* meliputi *AI-Assisted Coding* yang memberikan saran dan melengkapi kode secara otomatis,

Intelligent Code Navigation untuk navigasi cepat dalam proyek besar, dan *Enhanced Code Understanding* yang membantu pengembang memahami kode kompleks[27]. Cursor juga menawarkan *Automated Refactoring*, *Real-time Error Detection and Fixing*, serta fitur kolaboratif untuk kerja tim. Keunggulan Cursor terletak pada kemampuannya meningkatkan produktivitas, mengurangi kesalahan, mempercepat kurva pembelajaran, dan menjaga konsistensi kode[28].

2.3.3 *Framework Streamlit*

Streamlit telah muncul sebagai *framework open-source* yang populer dan kuat untuk membangun aplikasi *web* yang elegan dan interaktif di bidang *data science* dan *machine learning* menggunakan *Python*. Dirancang dengan fokus pada kesederhanaan dan kemudahan penggunaan, *Streamlit* memungkinkan pengembang untuk dengan cepat membuat antarmuka pengguna yang menarik dan responsif untuk proyek data mereka tanpa perlu memiliki pengetahuan mendalam tentang pengembangan *front-end* atau teknologi *web*. Dengan mengabstraksi kerumitan yang terlibat dalam pengembangan aplikasi *web* tradisional, *Streamlit* memberdayakan ilmuwan data dan peneliti untuk berkonsentrasi pada analisis dan pemodelan data sambil menghasilkan dashboard dan demo yang mengesankan secara visual[29].

Salah satu fitur utama *Streamlit* adalah kemampuannya untuk membuat elemen antarmuka pengguna yang kaya dan interaktif dengan hanya beberapa baris kode *Python*. *Framework* ini menyediakan *API* deklaratif yang

intuitif yang memungkinkan pengembang untuk dengan mudah menambahkan komponen seperti input teks, slider, tombol, dan bagan ke aplikasi mereka. Komponen ini dapat terhubung dengan skrip *Python* yang mendasarinya, memungkinkan pembaruan *real-time* dan interaksi dengan model dan visualisasi data. Pendekatan deklaratif ini menyederhanakan proses pengembangan secara signifikan, menghilangkan kebutuhan untuk secara manual memanipulasi *DOM* atau menangani permintaan *HTTP*[30].



BAB III

ANALISIS PERANCANGAN SISTEM

3.1 Analisis Sistem

Analisis sistem merupakan langkah awal yang krusial dalam proses pengembangan sistem prediksi penyakit diabetes dengan menggunakan algoritma *Support Vector Machine* (SVM). Tahapan ini bertujuan untuk mengidentifikasi permasalahan utama, seperti kurangnya deteksi dini terhadap penyakit diabetes dan keterbatasan akses masyarakat terhadap layanan kesehatan. Algoritma SVM dipilih karena kemampuannya dalam mengklasifikasikan data secara akurat melalui pendekatan *machine learning*. Analisis ini meliputi identifikasi kebutuhan sistem, pengolahan data, serta perancangan model prediksi yang akan diintegrasikan ke dalam aplikasi *web* berbasis Streamlit agar dapat digunakan masyarakat secara praktis dan mandiri.[31].

3.1.1 Analisis Masalah

Masalah pokok dalam penelitian ini adalah lonjakan jumlah penderita diabetes secara global yang terus meningkat, menunjukkan pentingnya pengembangan sistem deteksi dini yang tepat dan dapat diakses oleh masyarakat luas. Menurut data dari *International Diabetes Federation*, pada tahun 2019 terdapat 433 juta kasus diabetes, dan angka ini diproyeksikan akan melonjak menjadi 700 juta pada tahun 2045.[32]. Diabetes merupakan penyakit metabolik kronis yang jika tidak dideteksi dan ditangani sejak dini dapat menimbulkan komplikasi serius, bahkan kematian. Namun, akses masyarakat terhadap deteksi dini masih terbatas, baik karena minimnya fasilitas kesehatan maupun kurangnya

kesadaran individu. Melihat permasalahan tersebut, dibutuhkan solusi berbasis teknologi yang mampu membantu masyarakat dalam memprediksi risiko diabetes secara mandiri. Pemanfaatan *data mining* dan *machine learning*, khususnya algoritma *Support Vector Machine (SVM)*, menjadi pendekatan yang potensial karena mampu mengenali pola dari data pasien dan mengklasifikasikan risiko dengan tingkat akurasi tinggi[7].

3.1.2 Analisis Kebutuhan Sistem

Analisis kebutuhan sistem merupakan tahapan krusial dalam pengembangan sistem deteksi penyakit diabetes menggunakan algoritma *Support Vector Machine (SVM)*

a. Kebutuhan Perangkat Lunak

Untuk mendukung implementasi sistem, berikut adalah spesifikasi perangkat lunak dan keras yang direkomendasikan. Kebutuhan Perangkat Lunak :

- 1) Sistem Operasi: *Windows 10/11*, *macOS*, atau *Linux*
- 2) *Python 3.8* atau versi lebih baru
- 3) *Jupyter Notebook* untuk pengembangan dan analisis data
- 4) *Scikit-learn* untuk implementasi *SVM* dan *Random Forest*
- 5) *Pandas* dan *NumPy* untuk manipulasi data
- 6) *Matplotlib* dan *Seaborn* untuk visualisasi
- 7) *Streamlit* untuk pengembangan antarmuka *web*
- 8) *Cursor Code Editor*

b. Kebutuhan Perangkat Keras

- 1) Prosesor Intel Core i5 generasi ke-8 atau setara (AMD Ryzen 5)
- 2) RAM Minimal 8 GB, direkomendasikan 16 GB
- 3) Penyimpanan SSD 256 GB
- 4) Koneksi internet yang stabil untuk akses ke dataset

3.1.3 Analisis Data

Analisis data pada penelitian ini menggunakan dataset yang diambil dari situs *kaggle.com*, dataset diabetes pima indians ini mencakup 768 entri, terdiri dari 500 data dengan hasil negatif diabetes dan 268 data dengan hasil positif. Distribusi ini mengindikasikan adanya ketidakseimbangan kelas (*class imbalance*), yang merupakan aspek krusial dalam proses pengembangan model klasifikasi.

a. Pregnancies

Kehamilan merupakan atribut yang ada pada dataset Tabel 3.1 ini, meliputi beberapa nilai berikut :

Tabel 3.1 Atribut Kehamilan

No	Kehamilan	Jumlah	No	Kehamilan	Jumlah
1	0	111	10	10	24
2	1	135	11	9	28
3	2	103	12	11	11
4	3	75	13	13	10
5	4	68	14	12	9
6	5	57	15	14	2
7	6	50	16	15	1
8	7	45	17	17	1
9	8	38			

b. Glucose

Gula darah (*glukosa*) yang terdapat pada dataset Tabel 3.2 ini, memiliki beberapa nilai, sebagai berikut :

Tabel 3. 2 Atribut Gula Darah

No	Glukosa	Jumlah
1	0 - 101	223 data
2	102 - 141	358 data
3	142 – 181	151 data
4	182 - 199	36 data

c. Blood Pressure

Atribut ini blood pressure atau tekanan darah yang diperiksa dengan memeriksa tensi darah, pada dataset Tabel 3.3 meliputi :

Tabel 3. 3 Atribut Tekanan Darah

No	Tekanan Darah	Jumlah
1	0 - 60	158 data
2	61 - 84	498 data
3	85 – 122	data

d. Skin Thickness

Atribut skin thickness atau ketebalan kulit menjadi beberapa nilai, pada dataset Tabel 3.4 ini meliputi:

Tabel 3. 4 Atribut Ketebalan Kulit

No	Ketebalan Kulit	Jumlah
1	0 – 21	361 data
2	22 – 36	276 data
3	37 – 99	data

e. Insulin

Atribut insulin menjadi dataset yang penting pada diagnosa penyakit diabetes, pada Tabel 3.5 sebagai berikut :

Tabel 3. 5 Atribut Insulin

No	Insulin	Jumlah
1	0– 52	416 data
2	53 – 79	61 data
3	80 – 115	80 data
4	116 – 165	73 data
5	166 – 846	data

f. BMI

Atribut BMI atau Body Mass Index menjadi salah satu variabel untuk diagnosa penyakit diabetes, yang dijelaskan pada Tabel 3.6 berikut :

Tabel 3. 6 Atribut BMI

No	Fungsi Selisih Diabetes	Jumlah
1	0 – 29,3	161 data
2	29,4 – 38,2	207 data
3	38,3 – 67,1	data

g. Diabetes Pedigree Function

Atribut Diabetes *Pedigree Function* atau fungsil selisih diabetes merupakan bagian dari dataset pada Tabel 3.7 ini, meliputi :

Tabel 3. 7 Atribut Fungsi Selisih Diabetes

No	Fungsi Selisih Diabetes	Jumlah
1	0 – 0,227	161 data
2	0,228 – 0,355	207 data
3	0,356 – 525	138 data
4	0,526 – 0,732	135 data
5	0,737 – 2,42	data

h. Age

Usia penderita menjadi faktor penting dataset ini, pada bagian Tabel

3.8 ini akan dijelaskan sebagai berikut :

Tabel 3. 8 Atribut Usia

No	Umur	Jumlah
1	21– 33	474 data
2	34 – 54	240 data
3	55 – 81	54 data

i. Outcome

Atribut pada Tabel 3.9 ini merupakan hasil diabetes “yes” yang dirubah menjadi 1 dan “no” menjadi 0, yang terdiri dari :

Tabel 3. 9 Atribut Hasil Diabetes

No	Hasil Diabetes	Jumlah	Nilai
1	Yes	268 data	1
2	No	500 data	0

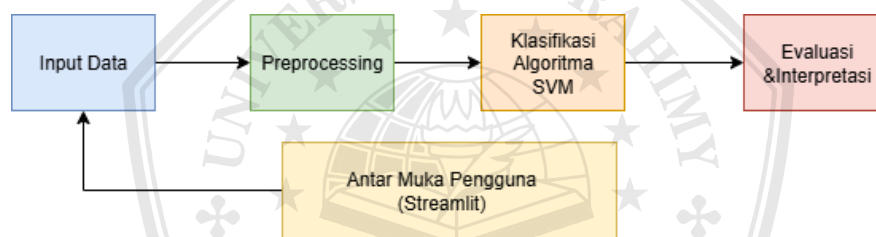
3.2 Perancangan Sistem

Tahap perancangan sistem memegang peranan penting dalam proses pengembangan aplikasi prediksi penyakit diabetes yang menggunakan algoritma *Support Vector Machine* (SVM). Tahapan ini bertujuan untuk memvisualisasikan alur proses data dalam sistem, mulai dari saat pengguna memasukkan data hingga sistem menghasilkan prediksi yang ditampilkan melalui antarmuka aplikasi *web*. [7].

3.2.1 Arsitektur Aplikasi

Arsitektur sistem untuk mendeteksi penyakit diabetes dengan algoritma *Support Vector Machine* (SVM) dirancang dengan fokus pada efisiensi, skalabilitas,

dan kemudahan dalam penggunaannya. Sistem ini terbagi ke dalam beberapa komponen utama yang terintegrasi untuk menghasilkan deteksi yang akurat dan dapat dipercaya. Komponen pertama adalah modul input data, yang berfungsi menerima serta memverifikasi data pasien. Informasi yang dikumpulkan meliputi variabel-variabel seperti jumlah kehamilan, tingkat glukosa, tekanan darah, ketebalan kulit, kadar insulin, indeks massa tubuh (BMI), riwayat genetik diabetes, serta usia. Modul ini juga menangani proses pra-pemrosesan untuk menjamin kualitas serta konsistensi data yang akan digunakan.



Gambar 3. 1 Arsitektur Aplikasi

Diagram alur sistem untuk prediksi penyakit diabetes menggunakan algoritma *Support Vector Machine* (SVM) telah diintegrasikan ke dalam aplikasi *web* berbasis *Streamlit*. Berikut ini merupakan uraian dari masing-masing komponen yang terdapat dalam diagram tersebut :

1. Input Data

- a. Tahap awal di mana pengguna memasukkan data medis (misalnya : kadar glukosa, tekanan darah, BMI, usia, dll).
- b. Data ini dapat dimasukkan melalui antarmuka pengguna (*Streamlit*).

2. Preprocessing

- a. Tahap praproses data bertujuan untuk membersihkan dan mempersiapkan data agar siap digunakan oleh model.
- b. Proses ini bisa mencakup :
 - 1) Normalisasi atau standarisasi data
 - 2) Penanganan nilai kosong
 - 3) Transformasi fitur jika diperlukan

3. Klasifikasi Algoritma SVM

- a. Data yang telah diproses kemudian diklasifikasikan menggunakan algoritma *Support Vector Machine*.
- b. Algoritma ini akan memprediksi apakah pasien berisiko terkena diabetes atau tidak, berdasarkan data yang diberikan.

4. Evaluasi & Interpretasi

- a. Hasil dari prediksi akan dievaluasi menggunakan metrik evaluasi seperti:
 - 1) Akurasi
 - 2) Presisi
 - 3) Recall
 - 4) F1-Score
- b. Interpretasi hasil ini berguna untuk menilai kinerja model serta membantu pengguna memahami hasil prediksi.

5. Antar Muka Pengguna (Streamlit)

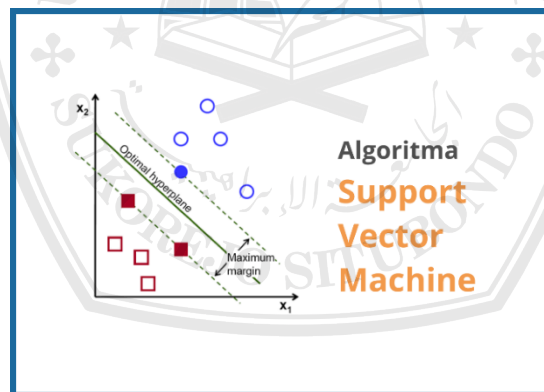
- a. *Streamlit* adalah *framework Python* yang digunakan untuk membangun antarmuka pengguna secara cepat dan interaktif.

b. Fungsinya dalam diagram ini:

- 1) Menerima input data dari pengguna
- 2) Menampilkan hasil prediksi
- 3) Menyediakan tampilan yang mudah digunakan untuk menjalankan seluruh proses dari input hingga evaluasi

3.2.2 Perancangan Model Machine Learning

Algoritma *Support Vector Machine* (SVM) merupakan metode yang dirancang untuk menentukan *hyperplane* terbaik, yaitu sebuah fungsi yang mampu membedakan dua kelas data. Dalam penerapannya, SVM berupaya untuk memperluas margin atau jarak antara data pelatihan dengan garis batas pemisah (*decision boundary*). [33]



Gambar 3. 2 Algoritma Support Vector Machine (SVM)

3.3 Implementasi Sistem

3.3.1 Library Yang Digunakan

1. NumPy

NumPy merupakan pustaka dasar yang sangat penting untuk melakukan komputasi numerik di *Python*. Dalam proyek ini, *NumPy* dimanfaatkan untuk memanipulasi array dan menjalankan berbagai operasi matematika secara efisien.

Library ini menyediakan struktur data berupa array multi-dimensi yang sangat kuat, sehingga sangat berguna untuk merepresentasikan dan mengolah data numerik dalam skala besar. Selain itu, *NumPy* juga dilengkapi dengan beragam fungsi matematika dan statistik yang dapat diterapkan langsung pada array, memungkinkan pelaksanaan operasi vektor dan matriks secara cepat dan efisien.

2. Pandas

Pandas merupakan *library* yang krusial dalam analisis dan manipulasi data menggunakan *Python*. Pada proyek ini, Pandas digunakan untuk memuat, mengelola, dan memodifikasi dataset diabetes. *Library* ini menyediakan struktur data seperti *DataFrame* dan *Series* yang memungkinkan pengolahan data terstruktur secara efisien. Dengan berbagai fungsi yang dimilikinya, Pandas mempermudah proses pembacaan file CSV, manipulasi data, penanganan data yang hilang (*missing values*), hingga agregasi data. Hal ini sangat berguna terutama dalam tahap pra-pemrosesan data sebelum dilakukan analisis lebih lanjut.

3. Scikit-learn

Scikit-learn merupakan pustaka *machine learning* yang lengkap dan banyak digunakan dalam bahasa *Python*. Dalam proyek ini, *scikit-learn* dimanfaatkan untuk berbagai keperluan *machine learning*, seperti pra-pemrosesan data, pembagian dataset, pembuatan model, hingga evaluasi performa model. *Library* ini menyediakan beragam algoritma *machine learning*, termasuk *Support Vector Machine* (SVM) yang digunakan dalam proyek Anda. Selain itu, *scikit-learn* juga menyediakan alat evaluasi model seperti confusion matrix, akurasi, presisi, recall,

dan F1-score. Dengan *scikit-learn*, proses membangun, melatih, dan mengevaluasi model prediksi diabetes dapat dilakukan secara praktis dan efisien.

4. *Matplotlib*

Matplotlib merupakan *library* visualisasi data yang kuat dan fleksibel dalam ekosistem *Python*. Dalam proyek ini, *Matplotlib* digunakan untuk membuat berbagai visualisasi grafik, seperti histogram distribusi kelas dan plot *confusion matrix*. *Library* ini mendukung pembuatan beragam jenis grafik, mulai dari yang sederhana hingga yang lebih kompleks. *Matplotlib* juga menyediakan kontrol penuh terhadap elemen-elemen visualisasi, sehingga Anda dapat menyesuaikan tampilan grafik sesuai dengan kebutuhan analisis dan presentasi data.

5. *Seaborn*

Seaborn merupakan *library* visualisasi statistik yang dikembangkan di atas *Matplotlib*. Dalam proyek ini, *Seaborn* dimanfaatkan untuk membuat heatmap dari *confusion matrix*. *Library* ini menawarkan antarmuka tingkat tinggi yang memudahkan pembuatan grafik statistik yang informatif dan menarik secara visual. *Seaborn* sangat berguna untuk menyajikan visualisasi data yang lebih kompleks dan estetik, berkat fitur-fitur seperti palet warna yang telah disesuaikan serta pengaturan tema yang intuitif dan mudah diterapkan.

6. *Pickle*

Pickle merupakan modul bawaan *Python* yang digunakan untuk melakukan serialisasi dan deserialisasi objek. Dalam proyek ini, *Pickle* dimanfaatkan untuk menyimpan model *machine learning* yang telah dilatih ke dalam sebuah file. Dengan demikian, model tersebut dapat disimpan dan dimuat kembali tanpa harus

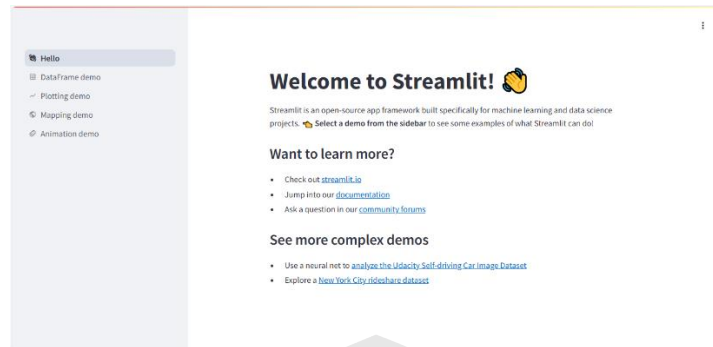
menjalani proses pelatihan ulang. Penggunaan *Pickle* sangat membantu dalam tahap *deployment* model, karena memungkinkan model yang sudah siap pakai digunakan untuk melakukan prediksi di masa mendatang tanpa perlu melalui proses pelatihan yang memakan waktu.

Pembersihan Data (*Data Cleaning*) Pembersihan data merupakan proses penting dalam pra-pemrosesan data, yang bertujuan untuk menghapus atau memperbaiki data yang tidak lengkap, salah, atau tidak konsisten. Langkah ini memastikan bahwa data yang digunakan untuk pelatihan dan evaluasi model bersih dan berkualitas, sehingga hasil analisis atau prediksi menjadi lebih akurat dan dapat diandalkan.

3.3.2 Implementasi *Streamlit*

Penelitian ini memanfaatkan *Streamlit*, sebuah *framework open-source* berbasis *Python*, untuk menampilkan hasil klasifikasi penyakit diabetes dalam bentuk aplikasi *web* yang interaktif dan *user-friendly*. Proses implementasi *Streamlit* diawali dengan menginstal framework menggunakan perintah “*pip install streamlit*” melalui terminal. Setelah semua komponen tersedia, aplikasi dapat dijalankan dengan perintah “*streamlit run diabetes.py.*” *Streamlit* kemudian akan membuka aplikasi secara otomatis di browser default, memungkinkan pengguna untuk memasukkan data dan memperoleh hasil prediksi secara langsung dan *real-time*.

Berikut merupakan tampilan antarmuka *framework streamlit* yang akan digunakan dalam penelitian ini :



Gambar 3. 3 Aplikasi Streamlit

a. Input Data Pengguna

Pengguna diminta untuk mengisi sejumlah parameter yang relevan dengan risiko diabetes. Parameter ini diadaptasi dari *Pima Indians Diabetes Dataset* dan

terdiri atas 8 fitur utama, yaitu:

- 1) Jumlah kehamilan (*Pregnancies*)
- 2) Kadar glukosa (*Glucose*)
- 3) Tekanan darah (*Blood Pressure*)
- 4) Ketebalan kulit (*Skin Thickness*)
- 5) Kadar insulin (*Insulin*)
- 6) Indeks massa tubuh (*BMI*)
- 7) Riwayat keturunan diabetes (*Diabetes Pedigree Function*)
- 8) Usia (*Age*)

b. Proses Klasifikasi oleh Model SVM

Setelah pengguna mengisi semua data yang diminta, informasi tersebut akan dianalisis oleh model klasifikasi yang sebelumnya telah dilatih menggunakan algoritma *Support Vector Machine* (SVM). Model ini akan menentukan apakah data yang diberikan termasuk dalam kategori "berisiko diabetes" atau "tidak berisiko", berdasarkan pola-pola yang telah dikenali dari data pelatihan. Untuk memastikan keakuratan input dan meminimalkan kesalahan pengisian, sistem dilengkapi dengan mekanisme validasi nilai pada setiap parameter, seperti menetapkan batas minimum dan maksimum input, serta memberikan peringatan apabila pengguna memasukkan nilai yang tidak logis (misalnya: BMI = 0 atau usia = 200 tahun).

Tahapan ini mencakup:

- 1) Normalisasi nilai input (jika diperlukan)
- 2) Pemrosesan input ke dalam bentuk array
- 3) Pemanggilan model `.predict()` untuk menghasilkan klasifikasi

c. Output: Hasil Prediksi

Setelah pemrosesan selesai, hasil prediksi akan ditampilkan secara langsung di halaman aplikasi. Output ditampilkan dalam bentuk:

- 1) Notifikasi status (misalnya: **Anda berisiko diabetes** atau **Anda tidak berisiko diabetes**)
- 2) Pesan saran (misalnya: “*Disarankan untuk konsultasi ke dokter jika memiliki gejala tertentu*”)

Visual feedback seperti balon animasi (`st.balloons()`) juga ditambahkan untuk membuat antarmuka lebih interaktif.



The image shows a web interface for a diabetes prediction application. The title is "Prediksi Penyakit Diabetes" with a small icon to the right. Below the title, there are eight input fields arranged in two columns. The left column contains: "Pregnancies", "Blood Pressure", "Insulin", and "Diabetes Pedigree Function". The right column contains: "Glucose", "Skin Thickness", "BMI", and "Age". At the bottom of the form, there are two buttons: "Test Prediksi Diabetes" and "Lihat Data dari Database".

Gambar 3. 4 Perancangan Desain Prediksi Diabetes

3.3.2 Uji Coba Sistem oleh Pengguna (User Validator)

Untuk memastikan bahwa sistem prediksi yang dibangun dapat digunakan secara efektif dan mudah dipahami oleh pengguna umum, dilakukan proses uji coba awal (*user validation*) kepada beberapa mahasiswa sebagai representasi pengguna non-teknis. Tujuan dari tahap ini adalah untuk mengetahui apakah sistem sudah berjalan dengan baik, mudah digunakan, serta memberikan hasil yang dapat dimengerti oleh pengguna awam. Responden merupakan 5 mahasiswa dari program studi Teknologi Informasi, yang belum pernah menggunakan *Streamlit* atau sistem prediksi serupa sebelumnya. Adapun proses uji coba sebagai berikut:

- a. Mahasiswa diberikan akses ke aplikasi prediksi berbasis *web*.
- b. Mereka diminta untuk mengisi data secara mandiri sesuai dengan parameter yang tersedia.

c. Setelah mendapatkan hasil prediksi, mereka mengisi form tanggapan yang berisi:

- 1) Apakah form mudah diisi?
- 2) Apakah hasil prediksi bisa dimengerti?
- 3) Apakah tampilannya menarik dan responsif?
- 4) Saran perbaikan dari pengguna.

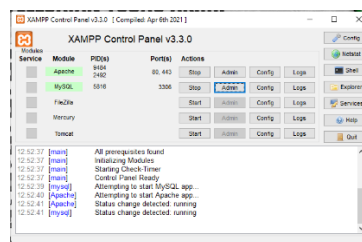
d. Adapun tujuan validasi ini yaitu :

- 1) Mengetahui apakah alur penggunaan aplikasi sudah cukup jelas.
- 2) Memastikan hasil prediksi muncul dengan cepat dan informatif.
- 3) Mengidentifikasi kekurangan dalam hal *UX (User Experience)* atau pesan sistem.

Hasil dari validasi ini akan dijadikan sebagai masukan untuk pengembangan sistem selanjutnya, termasuk penambahan fitur seperti peringatan input tidak wajar, deskripsi risiko, maupun tombol bantuan.

3.3.3 Control Panel Xampp

Gambar ini menampilkan XAMPP Control Panel versi 3.3.0, di mana modul Apache dan MySQL terlihat telah aktif (status: running). Panel ini berfungsi untuk mengelola layanan server lokal yang biasa digunakan dalam proses pengembangan aplikasi web berbasis PHP dan MySQL.



Gambar 3. 5 Xampp Control Panel

BAB IV

HASIL & PEMBAHASAN

4.1 Deskripsi Dataset

Dataset yang digunakan dalam penelitian ini berasal dari *platform Kaggle.com*, sebuah situs penyedia data terbuka yang sangat populer dan banyak digunakan oleh komunitas data *science* di seluruh dunia. Penelitian ini memanfaatkan data sekunder yang telah tersedia di *Kaggle*, sehingga proses pengumpulan data menjadi lebih efisien dan praktis. Dengan menggunakan dataset dari sumber terpercaya ini, penelitian dapat fokus pada analisis dan pengolahan data secara optimal untuk mencapai hasil yang akurat dan relevan.

[\(<https://www.kaggle.com/code/nancyalaswad90/diabetesdatabase/input?select=diabetes.csv>\)](https://www.kaggle.com/code/nancyalaswad90/diabetesdatabase/input?select=diabetes.csv). *Kaggle* adalah sebuah platform daring yang sangat populer di kalangan praktisi data *science*, peneliti, maupun pengembang machine learning ataupun *deep learning*. Platform ini menyediakan berbagai macam dataset yang dapat diakses secara gratis oleh siapa saja yang ingin belajar, melakukan eksperimen, atau mengembangkan model analisis data. Selain sebagai sumber data, *Kaggle* juga menjadi wadah bagi komunitas global untuk berbagi pengetahuan, mengikuti kompetisi, serta berdiskusi mengenai berbagai topik terkait data *science* dan *machine learning*. Banyak perusahaan dan institusi ternama juga memanfaatkan *Kaggle* untuk mengadakan lomba pemecahan masalah nyata berbasis data, sehingga *platform* ini menjadi sangat relevan bagi siapa saja yang ingin mengasah kemampuan analisis data dan algoritma prediksi.

Dalam penelitian ini, data yang digunakan diambil dari salah satu dataset yang tersedia di *Kaggle*, yaitu dataset mengenai penyakit diabetes. Dataset ini sangat sering digunakan dalam penelitian dan pengembangan model prediksi penyakit, karena memiliki struktur data yang jelas dan jumlah sampel yang memadai. Secara spesifik, dataset diabetes yang digunakan terdiri dari 768 baris data, di mana setiap baris mewakili satu individu atau pasien. Setiap sampel memiliki sembilan kolom fitur, yang terdiri dari delapan variabel prediktor dan satu variabel target. Delapan fitur prediktor tersebut meliputi sejumlah parameter klinis dan demografis yang berkaitan dengan risiko diabetes, seperti usia, indeks massa tubuh, tekanan darah, kadar glukosa, dan lain-lain. Sementara itu, fitur target atau label pada dataset ini menunjukkan status diabetes pada masing-masing individu, yaitu apakah pasien tersebut terdiagnosis diabetes atau tidak. Dengan struktur data seperti ini, dataset diabetes dari *Kaggle* sangat cocok digunakan untuk membangun dan menguji model klasifikasi dalam bidang kesehatan, khususnya dalam upaya deteksi dini penyakit diabetes menggunakan metode *machine learning*.

Dataset ini memuat 768 data individu yang masing-masing memiliki 8 fitur penting yang digunakan dalam proses klasifikasi diabetes. Fitur-fitur tersebut antara lain: jumlah kehamilan (*Pregnancies*), konsentrasi glukosa plasma (*Glucose*), tekanan darah diastolik (*Blood Pressure*), ketebalan lipatan kulit triceps (*Skin Thickness*), kadar insulin dua jam (*Insulin*), indeks massa tubuh (*BMI*), fungsi silsilah diabetes (*Diabetes Pedigree Function*), usia (*Age*), dan status diagnosis diabetes sebagai label kelas. Seluruh atribut ini diambil dari Pima Indians Diabetes Database, yang kerap digunakan sebagai acuan dalam penelitian klasifikasi

penyakit diabetes. Keberagaman fitur yang tersedia memungkinkan model prediksi untuk menghasilkan akurasi yang tinggi dalam mengklasifikasikan status diabetes pada individu, sekaligus memberikan wawasan penting terkait pola kesehatan masyarakat yang diteliti untuk lebih jelasnya data atau fitur yang akan peneliti lakukan seperti di bawah ini terkait data diabetes :

- a. *Pregnancies* : Jumlah kehamilan
- b. *Glucose* : Kadar glukosa dalam darah
- c. *BloodPressure* : Tekanan darah diastolik (mm Hg)
- d. *SkinThickness* : Ketebalan lipatan kulit trisep (mm)
- e. *Insulin* : Kadar insulin serum 2 jam (mu U/ml)
- f. *BMI* : Indeks Massa Tubuh (berat dalam kg/(tinggi dalam m)²)
- g. *DiabetesPedigreeFunction* : Fungsi silsilah diabetes
- h. *Age* : Usia (tahun)
- i. *Outcome* atau Hasil : Variabel kelas (0 atau 1) yang menunjukkan ada atau tidaknya diabetes.

Tahap pertama dalam penelitian ini adalah melakukan preprocessing data, yaitu proses menyiapkan dataset agar siap untuk dianalisis secara lebih mendalam. Dataset yang digunakan berasal dari data diabetes yang diambil dari file CSV bernama 'diabetes.csv'. Untuk memuat file CSV tersebut ke dalam program, digunakan *library* pandas pada bahasa pemrograman *Python*. *Library* pandas.

	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFunction	Age	Outcome
0	6	148	72	35	0	33.6	0.627	50	1
1	1	85	66	29	0	26.6	0.351	31	0
2	8	183	64	0	0	23.3	0.672	32	1
3	1	89	66	23	94	28.1	0.167	21	0
4	0	137	40	35	168	43.1	2.288	33	1

Gambar 4. 1 Dataset Diabetes.CSV

menyediakan fungsi `read_csv()` yang sangat berguna untuk membaca dan mengimpor data dari file CSV ke dalam struktur data frame, sehingga memudahkan pengolahan data selanjutnya.

Setelah dataset berhasil dimuat, tahap berikutnya adalah melakukan eksplorasi untuk memahami struktur dan isi data secara menyeluruh. Proses ini meliputi pemeriksaan tipe variabel, distribusi nilai, serta identifikasi nilai yang hilang atau anomali. Eksplorasi data membantu mengenali pola, hubungan antar variabel, dan potensi masalah dalam dataset sehingga analisis selanjutnya dapat dilakukan dengan lebih akurat dan efektif. Dengan memahami karakteristik data secara mendalam, kita dapat menentukan metode pemrosesan dan teknik analisis yang paling sesuai untuk mencapai hasil yang optimal. Tahapan eksplorasi ini sangat penting sebagai fondasi awal dalam proses analisis data yang sistematis dan terstruktur.

```
import pandas as pd
data = pd.read_csv('/content/drive/MyDrive/Colab
Notebooks/Diabetes.csv')
print(data)
```

Segmen Program 4.1 Read Dataset

dari dataset. Fungsi `head()` dari DataFrame digunakan untuk menampilkan beberapa baris awal dari dataset. Hal ini memungkinkan untuk melihat sekilas fitur-fitur yang ada dalam dataset dan tipe datanya. Dari output yang ditampilkan, terlihat bahwa dataset diabetes terdiri dari beberapa fitur seperti jumlah kehamilan, kadar glukosa darah, tekanan darah, ketebalan kulit, kadar insulin, index massa tubuh (BMI), fungsi silsilah diabetes, usia, dan hasil diagnosis diabetes.

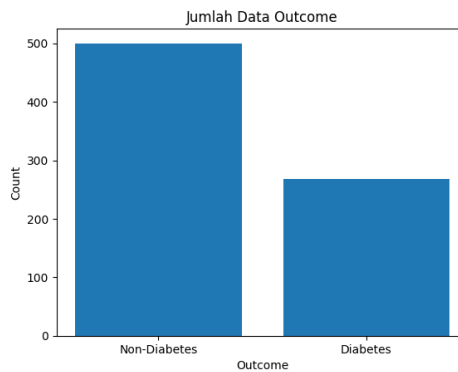
Setelah melakukan eksplorasi awal terhadap dataset diabetes, langkah selanjutnya adalah melakukan visualisasi distribusi data untuk fitur "*Outcome*". Fitur "*Outcome*" merupakan variabel target yang menunjukkan apakah seseorang didiagnosis diabetes (nilai 1) atau tidak (nilai 0).

Untuk memvisualisasikan distribusi data "*Outcome*", digunakan *library Matplotlib* dan *PyPlot* dalam *Python*. Pertama, dilakukan penghitungan jumlah data untuk setiap nilai unik dalam fitur "*Outcome*" menggunakan fungsi *value_counts()* dari *pandas*.

```
import matplotlib.pyplot as plt
# Hitung jumlah data "Outcome"
outcome_counts = diabetes_dataset['Outcome'].value_counts()
# Buat grafik batang
plt.bar(outcome_counts.index, outcome_counts.values)
plt.xlabel('Outcome')
plt.ylabel('Count')
plt.title('Jumlah Data Outcome')
plt.xticks(outcome_counts.index, ['Non-Diabetes', 'Diabetes'])
plt.show()
```

Segmen Program 4. 2 Menampilkan Chart Data Diabetes

Hasilnya disimpan dalam variabel *outcome_counts*. Grafik ditampilkan menggunakan fungsi *plt.show()*. Dari grafik yang dihasilkan, terlihat bahwa jumlah data dengan nilai "*Outcome*" 0 (Non-Diabetes) lebih banyak dibandingkan dengan jumlah data dengan nilai "*Outcome*" 1 (Diabetes).



Gambar 4. 2 Grafik Data Diabetes

Dataset ini menyajikan campuran variabel numerik dan kategorik, dengan "Outcome" sebagai variabel target biner yang menunjukkan diagnosis diabetes. Sebelum melakukan analisis lebih lanjut, penting untuk memahami karakteristik dan distribusi setiap fitur dalam dataset. Untuk fitur kategorik "Outcome", distribusinya adalah sebagai berikut:

1. Kelas 0 (Tidak Diabetes): 500 sampel (65,1%)
2. Kelas 1 (Diabetes): 268 sampel (34,9%)

4.2 Preprocessing

4.2.1 Missing Value

Dilakukan pengecekan terhadap distribusi data untuk fitur "Outcome" menggunakan fungsi `value_counts()` dari `pandas`. Fitur "Outcome" merupakan variabel target yang menunjukkan apakah seseorang didiagnosis diabetes (nilai 1) atau tidak (nilai 0). Hasil dari `value_counts()` menunjukkan bahwa terdapat 500 sampel dengan nilai "Outcome" 0 (Non-Diabetes) dan 268 sampel dengan nilai "Outcome" 1 (Diabetes).

Untuk memvisualisasikan distribusi data "Outcome" secara lebih jelas, digunakan fungsi *drop()* dari pandas untuk memisahkan fitur "Outcome" menjadi DataFrame terpisah (x) dan series (y). Kemudian, dilakukan pembuatan bar plot menggunakan fungsi *bar()* dari *Matplotlib*, dengan memberikan parameter *columns='Outcome'* dan *axis=1* untuk menampilkan distribusi data secara horizontal.

```
diabetes_dataset.shape
```

Segmen Program 4. 3 Shape Data

4.2.2 Drop Data

Tahap preprocessing dengan melakukan drop data merupakan langkah krusial dalam pengolahan data sebelum memasuki proses pemodelan. Pada analisis deteksi penyakit diabetes, dataset yang digunakan memiliki fitur 'Outcome' yang berfungsi sebagai variabel target atau label, yaitu penanda apakah seseorang terdiagnosis diabetes atau tidak. Pada tahap ini, fitur 'Outcome' dipisahkan dari dataset menggunakan fungsi *drop* yang tersedia pada library pandas di *Python*. Pemisahan ini bertujuan untuk memisahkan antara fitur-fitur prediktor (variabel independen) dengan variabel target (variabel dependen) yang nantinya akan digunakan dalam proses pelatihan dan evaluasi model. Setelah fitur 'Outcome' dihilangkan dari dataset, fitur-fitur prediktor disimpan dalam variabel X, sedangkan nilai dari fitur 'Outcome' ditempatkan dalam variabel Y.

```
## pemisahan data dan tabel

x = data.drop(columns=["Hasil"])

y = data["Hasil"]

print("x : ", x.shape)

print("y : ", y.shape)
```

Segmen Program 4. 4 Drop Data

4.2.3 Splitting Dataset

Setelah proses preprocessing dan penghapusan data selesai dilakukan, tahap selanjutnya adalah pembagian data atau splitting data. Tahap ini bertujuan untuk memisahkan dataset menjadi dua bagian utama, yaitu data latih (training data) dan data uji (testing data). Data latih berfungsi untuk melatih model *machine learning* agar dapat mengenali pola serta hubungan antara fitur-fitur prediktor dan variabel target. Sementara itu, data uji digunakan untuk menguji kinerja model yang telah dilatih dalam memprediksi nilai variabel target pada data yang sebelumnya belum pernah digunakan dalam proses pelatihan.

```
## data trainingnya ini

from sklearn.model_selection import train_test_split
x_train, x_test, y_train, y_test =
train_test_split(x, y, test_size=0.2, stratify=y,
random_state=42)
```

Segmen Program 4. 5 Splitting Data

Pada kode yang tersedia, proses pemisahan data dilakukan menggunakan fungsi `train_test_split` dari model *sklearn*. Fungsi ini bertugas membagi dataset X dan Y menjadi dua bagian, yaitu data pelatihan (X_train dan Y_train) serta data

pengujian (X_{test} dan Y_{test}), sesuai dengan proporsi yang telah ditentukan. Dalam contoh ini, data pengujian mengambil porsi 20% dari keseluruhan data ($test_size=0.2$), sehingga sisanya, yaitu 80%, digunakan sebagai data pelatihan. Penambahan parameter $stratify=Y$ bertujuan untuk menjaga keseimbangan proporsi kelas (diabetes dan non-diabetes) pada kedua subset data, sehingga distribusi kelas pada data pelatihan dan pengujian tetap sama seperti pada dataset awal. Dengan membagi data seperti ini, model dapat dilatih menggunakan data pelatihan dan diuji performanya pada data pengujian, sehingga kemampuan model dalam mendeteksi penyakit diabetes dapat dievaluasi secara objektif.

Pra-pemrosesan data merupakan tahap krusial dalam penelitian ini guna menjamin keandalan serta ketepatan data yang digunakan. Dataset yang dipakai berasal dari platform open-source kaggle.com, berisi 768 data terkait diabetes. Setelah melewati proses pra-pemrosesan, dataset tersebut siap untuk dikembangkan menjadi model klasifikasi menggunakan algoritma *Support Vector Machine* (SVM) dengan *kernel linear*. Dataset yang telah dibersihkan dan divalidasi ini menjadi fondasi yang kokoh untuk analisis lanjutan serta pemodelan prediktif dalam upaya menghasilkan klasifikasi penyakit diabetes yang akurat.

4.3 Implementasi Algoritma

4.3.1 Support vector machine

Dalam penelitian ini, model *Support Vector Machine* (SVM) diterapkan dengan menggunakan bahasa pemrograman *Python* yang dijalankan melalui platform *Google Colab*. Pemilihan SVM didasarkan pada kemampuannya yang

efektif dalam mengelola data medis yang kompleks, khususnya untuk membantu proses diagnosis penyakit diabetes secara akurat dan efisien.

```
# metode yang digunakan svm kernel linear
clf = SVC(kernel="linear")
clf.fit(x_train, y_train)
y_pred = clf.predict(x_test)
CLF_acc = accuracy_score(y_pred, y_test)
```

Segmen Program 4. 6 Training data Algoritma SVM

Model SVM dilatih menggunakan data pelatihan yang sudah disiapkan, kemudian diterapkan pada data pengujian. Kernel yang dipilih adalah *kernel linear* karena mengasumsikan kelas-kelas dapat dipisahkan secara *linear*. *Kernel linear* ini dipilih karena efisiensi komputasinya yang tinggi, terutama untuk banyak dataset dunia nyata yang bersifat linier. Kernel linear memproyeksikan data ke ruang fitur yang sama tanpa transformasi kompleks, sehingga mempercepat proses pelatihan dan prediksi. Pendekatan ini cocok untuk data yang memang dapat dipisahkan dengan garis atau hyperplane lurus, sehingga model dapat bekerja secara optimal dengan sumber daya komputasi yang lebih rendah. Kernel linear juga menghindari kebutuhan tuning parameter kernel yang rumit.

```
SVC(kernel='linear')
```

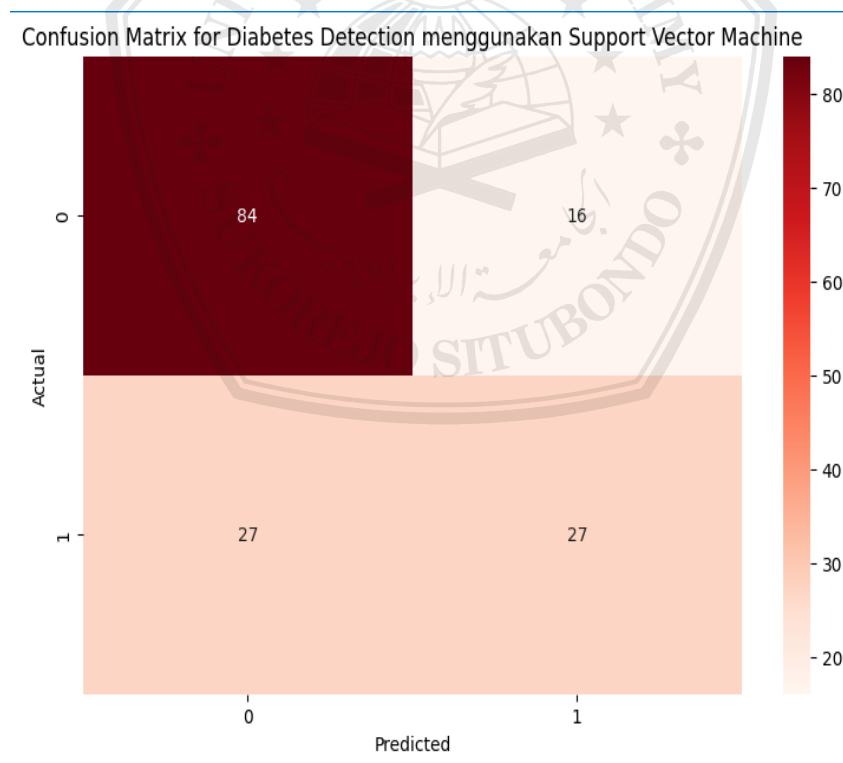
Segmen Program 4. 7 Algoritma SVM Dengan Kernel Linear

Setelah proses pelatihan model SVM rampung, tahap evaluasi kinerja dilakukan dengan menggunakan matriks konfusi. *Matriks* ini memberikan gambaran menyeluruh mengenai kemampuan prediksi model. Selain itu, *matriks*

konfusi juga menjadi dasar penting untuk menghitung berbagai metrik evaluasi, seperti akurasi, presisi, dan recall, yang menilai kualitas model secara keseluruhan.

```
Precision = precision_score(y_test, y_pred,  
average='binary')  
print("Precision:", Precision)  
Recall = recall_score(y_test, y_pred,  
average='binary')  
print("Recall:", Recall)  
Accuracy = accuracy_score(y_test, y_pred)  
print("Accuracy:", Accuracy)
```

Segmen Program 4. 8 Perintah Nilai Confusion Matrix



Gambar 4. 3 Nilai Confusion Matrix SVM

Hasil evaluasi model SVM ditampilkan dalam Tabel 4.1 di bawah ini :

Tabel 4. 1 Nilai Matrix SVM

Matrix	Nilai
Akurasi	72%
Presisi	73%
Recall	50%

Model SVM berhasil mencapai tingkat akurasi sebesar 72%, yang berarti model ini mampu mengklasifikasikan kasus diabetes dengan benar secara keseluruhan dalam sebagian besar data yang diuji. Selain itu, nilai presisi yang diperoleh adalah 73%, yang menunjukkan bahwa dari semua prediksi positif yang dibuat oleh model, 73% di antaranya benar-benar merupakan kasus diabetes. Namun, recall model ini hanya mencapai 50%, yang mengindikasikan bahwa model mampu mengidentifikasi setengah dari seluruh kasus diabetes yang sebenarnya ada dalam data.

4.4 Implementasi *Framework Streamlit*

Fase implementasi dalam penelitian ini menitikberatkan pada perubahan model *Support Vector Machine* (SVM) yang telah dikembangkan menjadi sebuah aplikasi yang praktis dan mudah digunakan. Setelah model selesai dibuat dan divalidasi, model tersebut diserialisasi menggunakan modul *pickle* di *Python*, yaitu proses yang mempertahankan struktur dan parameter model dalam format biner yang ringkas. Proses serialisasi ini sangat penting untuk memastikan deployment model yang efisien, sehingga memungkinkan pemuatan dan eksekusi model secara cepat di berbagai lingkungan tanpa perlu melatih ulang.

Dalam pengembangan aplikasi berbasis *web* menggunakan *framework Streamlit*, kode yang ditunjukkan merupakan bagian dari proses memuat dan menggunakan model *machine learning* yang telah dilatih sebelumnya. Tujuannya adalah untuk mengintegrasikan model tersebut ke dalam aplikasi *web* interaktif yang dapat menerima input dari pengguna dan memberikan prediksi atau klasifikasi berdasarkan model yang telah dilatih.

Baris kode `import pickle` menunjukkan bahwa *library pickle* digunakan untuk memuat model yang telah disimpan dalam format *pickle*. *Pickle* adalah modul *Python* yang digunakan untuk serialisasi dan deserialisasi objek *Python*, termasuk model *machine learning*.

```
import pickle
```

Segmen Program 4. 9 Input Pickle

```
filename = '/content/drive/MyDrive/Colab  
Notebooks/svm_model.pkl'  
pickle.dump(clf, open(filename, 'wb'))
```

Segmen Program 4. 10 Export Pickle

Selanjutnya, baris kode `filename = 'svm_model.pkl'` menetapkan nama file tempat model akan disimpan atau dimuat. Dalam contoh ini, file tersebut bernama `svm_model.pkl`, yang berisi model *Support Vector Machine* yang sudah dilatih untuk klasifikasi penyakit diabetes. Baris `pickle.dump(classifier, open(filename, 'wb'))` digunakan untuk menyimpan objek *classifier* ke dalam file dengan mode penulisan biner ('wb'). Mode ini memungkinkan model disimpan dalam format

pickle agar dapat digunakan kembali di kemudian hari. Setelah model berhasil dimuat menggunakan *pickle*, tahap berikutnya dalam pengembangan aplikasi *Streamlit* adalah membuat antarmuka pengguna yang interaktif.

4.4.1 Tampilan Antarmuka *Web*

Antarmuka aplikasi *web* yang dikembangkan dalam penelitian ini dirancang dengan tujuan untuk memberikan pengalaman pengguna yang intuitif dan ramah.

Di bagian atas tampilan terdapat judul aplikasi, yaitu "*Diabetes Prediction using Support Vector Machine*," yang menjelaskan secara singkat tujuan dan metode yang digunakan. Judul ini memberikan gambaran jelas kepada pengguna mengenai fungsi utama aplikasi, yaitu memprediksi penyakit diabetes dengan menggunakan algoritma *Support Vector Machine*.



Prediksi Penyakit Diabetes 

Pregnancies	Glucose
<input type="text"/>	<input type="text"/>
Blood Pressure	Skin Thickness
<input type="text"/>	<input type="text"/>
Insulin	BMI
<input type="text"/>	<input type="text"/>
Diabetes Pedigree Function	Age
<input type="text"/>	<input type="text"/>

Gambar 4. 4 Tampilan Antarmuka *Web*

Di bawah judul, tersedia beberapa kolom input yang mengumpulkan data penting dari pengguna untuk keperluan prediksi. Kolom-kolom tersebut meliputi "*Pregnancies*," "*Glucose*," "*Blood Pressure*," "*Skin Thickness*," "*Insulin*," "*BMI*," "*Diabetes Pedigree Function*," dan "*Age*." Setiap kolom dilengkapi dengan deskripsi singkat yang menjelaskan jenis data yang harus dimasukkan, seperti "*Number of Pregnancies*" atau "*Glucose level in blood*," sehingga memudahkan pengguna memahami informasi yang diperlukan.

Di bagian bawah kolom input, terdapat tombol "*Predict*" yang berfungsi untuk memulai proses prediksi setelah semua data diisi lengkap oleh pengguna.

4.4.2 Analisis Hasil Model

Hasil analisis terhadap model *Support Vector Machine* (SVM) dalam mendeteksi penyakit diabetes memperlihatkan *performa* yang cukup baik, meskipun masih terdapat peluang untuk perbaikan. Model ini berhasil meraih tingkat akurasi sebesar 72%, yang menandakan kemampuannya dalam membedakan kasus diabetes dan non-diabetes secara cukup efektif. Persentase akurasi tersebut menunjukkan bahwa SVM dapat dimanfaatkan sebagai alat bantu dalam *skrining* awal diabetes, khususnya di lingkungan dengan keterbatasan sumber daya medis. Meski demikian, masih terdapat sekitar 28% kasus yang terklasifikasi secara keliru atau salah, sehingga penegakan diagnosis akhir tetap memerlukan konfirmasi dari tenaga medis profesional.

Tingkat presisi model yang mencapai 73% merupakan pencapaian yang cukup memuaskan dari hasil pengujian ini. Nilai presisi yang tinggi menandakan bahwa prediksi positif diabetes yang dihasilkan oleh model memiliki tingkat

keakuratan yang baik. Dengan demikian, kemungkinan terjadinya false positive dapat diminimalkan, sehingga pasien terhindar dari kecemasan yang tidak perlu dan pemeriksaan lanjutan yang sebenarnya tidak dibutuhkan. Selain itu, presisi yang optimal ini menunjukkan bahwa model mampu secara efektif mengidentifikasi individu yang benar-benar berisiko tinggi terkena diabetes, sehingga intervensi dini dapat dilakukan secara lebih tepat dan efisien.

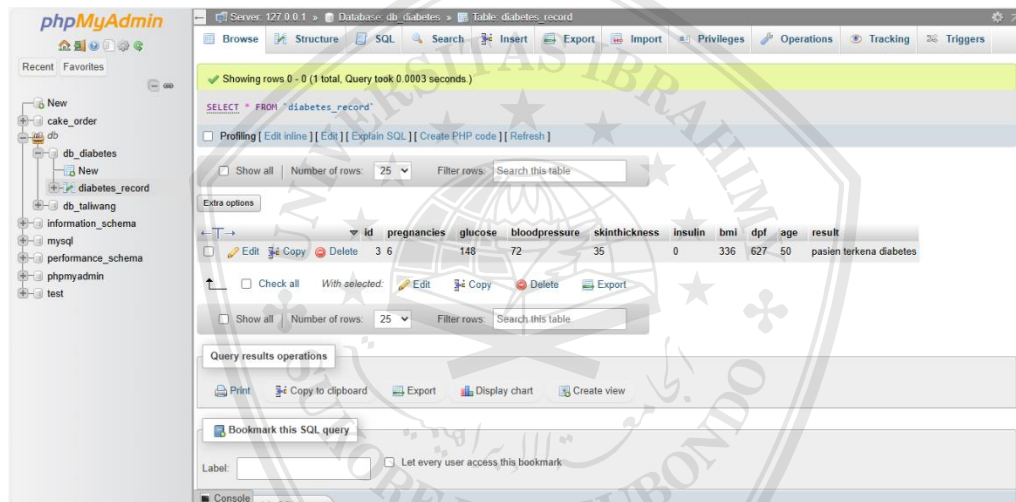
Walaupun demikian, nilai recall model yang tergolong rendah, yaitu sebesar 50%, menjadi perhatian penting dalam evaluasi ini. Recall yang tidak optimal ini mengindikasikan bahwa model hanya mampu mendeteksi kurang lebih separuh dari keseluruhan kasus diabetes yang sebenarnya ada. Hal tersebut berarti terdapat risiko tinggi bahwa sejumlah besar kasus diabetes yang sesungguhnya tidak teridentifikasi (*false negatives*). Situasi ini menandakan bahwa model tersebut mungkin kurang andal jika digunakan sebagai satu-satunya alat skrining, karena potensi besar kasus yang terlewatkan. Oleh karena itu, peningkatan nilai recall menjadi prioritas utama dalam pengembangan model selanjutnya..

Keunggulan utama model SVM dalam kasus ini terletak pada kemampuannya yang efektif dalam membedakan antara penderita diabetes dan non-diabetes, terutama ditunjukkan oleh tingkat akurasi dan presisi yang cukup tinggi. Model ini sangat sesuai digunakan untuk skrining awal karena mampu mengidentifikasi individu dengan risiko tinggi secara cukup tepat, sehingga membantu menentukan prioritas pemeriksaan lanjutan dan memaksimalkan pemanfaatan sumber daya medis yang terbatas. Meski demikian, model ini memiliki keterbatasan, terutama pada nilai recall yang rendah, yang dapat

mengurangi kemampuannya dalam menangkap kompleksitas berbagai faktor penyebab diabetes.

4.4.3 Tampilan Database php/MyAdmin

Gambar ini menampilkan hasil prediksi atau diagnosis penyakit diabetes pada seorang pasien berdasarkan sejumlah parameter medis. Melalui tampilan phpMyAdmin ini, pengguna dapat melakukan berbagai tindakan seperti mengubah, menghapus, atau mengekspor data yang tersimpan di dalam tabel.



Gambar 4. 5 Tampilan Database PHP/MyAdmin

BAB V

PENUTUP

5.1 Kesimpulan

Berdasarkan hasil penelitian tentang penerapan algoritma *Support Vector Machine* (SVM) untuk klasifikasi penyakit diabetes, dapat disimpulkan beberapa temuan kunci. Penelitian ini memanfaatkan dataset dari *platform Kaggle* yang terdiri dari 768 sampel data, dilengkapi 8 fitur prediktor (seperti kadar glukosa dan tekanan darah) serta 2 fitur target sebagai indikator diabetes.

Kinerja algoritma SVM menunjukkan hasil yang mengesankan dengan akurasi sebesar 72%, presisi 73%, dan recall 50%. Angka ini mengindikasikan bahwa model cukup handal dalam mengidentifikasi kasus diabetes secara tepat, meskipun masih ada ruang untuk peningkatan dalam mendeteksi semua kasus positif (*recall*). Proses preprocessing data yang cermat menjadi faktor penentu keberhasilan model. Tahapan ini meliputi penanganan nilai hilang, normalisasi fitur, dan pemilihan atribut relevan untuk memastikan kualitas input. Implementasi model ke dalam aplikasi *web* juga berhasil dilakukan, memungkinkan penggunaan praktis oleh tenaga medis atau masyarakat umum. Potensi utama penelitian terletak pada deteksi dini diabetes, di mana model ini dapat menjadi alat *skrining* awal yang terjangkau. Dengan pengembangan lebih lanjut—seperti penambahan data pelatihan, optimasi *hyperparameter*, atau integrasi teknik ensemble learning—akurasi dan efisiensi model dapat ditingkatkan secara signifikan.

Dampak sosial yang diharapkan adalah terciptanya solusi berbasis AI yang dapat diakses luas, mendukung pencegahan diabetes melalui diagnosis lebih cepat.

Kolaborasi antara peneliti, praktisi kesehatan, dan pengembang teknologi akan menjadi kunci dalam menyempurnakan alat ini menjadi asisten medis yang andal di masa depan.

5.2 Saran

Penelitian ini berhasil menerapkan algoritma *Support Vector Machine* (SVM) untuk mengklasifikasikan penyakit diabetes dengan hasil yang cukup memuaskan. Model yang dikembangkan menunjukkan performa yang baik dalam membedakan antara pasien yang mengidap diabetes dan yang tidak. Meskipun demikian, terdapat beberapa aspek yang dapat diperbaiki dan dikembangkan pada penelitian selanjutnya agar hasilnya lebih optimal. Pertama, peneliti disarankan untuk mengeksplorasi teknik penyeimbangan data seperti *oversampling* atau *undersampling*. Teknik-teknik ini sangat berguna untuk mengatasi masalah ketidakseimbangan kelas dalam dataset, dimana jumlah data pada kelas diabetes dan non-diabetes tidak seimbang. Dengan demikian, model dapat lebih akurat dalam mengklasifikasikan kedua kelas tersebut. Kedua, optimasi *hyperparameter* menggunakan metode *Grid Search* atau *Random Search* juga sangat dianjurkan. Langkah ini bertujuan untuk menemukan kombinasi parameter terbaik pada algoritma SVM sehingga dapat meningkatkan akurasi dan kemampuan generalisasi model ketika diaplikasikan pada data baru. Dengan menerapkan saran-saran ini, diharapkan penelitian berikutnya dapat menghasilkan model yang lebih handal dan efektif dalam deteksi dini diabetes.

DAFTAR PUSTAKA

- [1] N. Novianti, M. Zarlis, and P. Sihombing, "Penerapan Algoritma Adaboost Untuk Peningkatan Kinerja Klasifikasi Data Mining Pada Imbalance Dataset Diabetes," *J. Media Inform. Budidarma*, vol. 6, no. 2, p. 1200, 2022, doi: 10.30865/mib.v6i2.4017.
- [2] Rumiris Simatupang, "Penyuluhan tentang diabetes melitus pada lansia penderita DM," vol. 2, no. 3, pp. 849–858, 2023.
- [3] D. H. Bima-ntb, "Edukasi 5 Pilar Diabetes Mellitus Dalam Upaya Pencegahan Hiperglikemia meningkat pertahunnya . Kondisi ini terkait dengan perilaku pasien DM yang masih kurang tercapai yang selanjutnya dapat meminimalkan – mencegah terjadinya gangguan fungsi," vol. 1, no. 2, pp. 67–75, 2022.
- [4] A. Fauzi and A. H. Yunial, "Optimasi Algoritma Klasifikasi Naive Bayes, Decision Tree, K – Nearest Neighbor, dan Random Forest menggunakan Algoritma Particle Swarm Optimization pada Diabetes Dataset," *J. Edukasi dan Penelit. Inform.*, vol. 8, no. 3, p. 470, 2022, doi: 10.26418/jp.v8i3.56656.
- [5] G. Gunawan, A. Rahmawati, S. Suhada, T. Hidayatulloh, and D. Wintana, "Optimasi Linear Sampling dan Information Gain pada Algoritma Decision Tree Untuk Diagnosis Penyakit Diabetes," *Multinetics*, vol. 7, no. 2, pp. 124–131, 2022, doi: 10.32722/multinetics.v7i2.3796.
- [6] G. J. Azmi, "Analisis Faktor yang Memengaruhi Tingkat Diabetes Melitus pada Masyarakat di Kota/Kabupaten di Provinsi Jawa Timur Tahun 2018," vol. 5, no. 5, pp. 623–632, 2023.
- [7] H. Alrasyid, A. Homaidi, M. Kom, Z. Fatah, and M. Kom, "Comparison Support Vector Machine and Random Forest Algorithms in Detect Diabetes," vol. 1, no. 1, pp. 447–453, 2024.
- [8] A. Jalil, A. Homaidi, and Z. Fatah, "Implementasi Algoritma Support Vector Machine Untuk Klasifikasi Status Stunting Pada Balita," *G-Tech J. Teknol. Terap.*, vol. 8, no. 3, pp. 2070–2079, 2024, doi: 10.33379/gtech.v8i3.4811.
- [9] D. Trisnawati, M. Windarti, I. Sulistyowati, and F. B. Hartono, "SISTEM PAKAR PENDIAGNOSA PENYAKIT DIABETES," vol. 2, no. 1, pp. 14–19, 2022, doi: 10.54840/jcstech.v2i1.32.
- [10] B. R. Asmoro, A. Wibowo, and A. F. Aryadi, "Penggunaan Algoritma K-Means Untuk Menganalisa Penjualan Di Bigmart," *Jmari*, vol. 3, no. 2, pp. 189–199, 2022, doi: 10.33050/jmari.v3i2.2427.
- [11] U. Kusnia and F. Kurniawan, "Analisis Sentimen Review Aplikasi Media Berita Online Pada Google Play menggunakan Metode Algoritma Support Vector Machines (SVM) Dan Naive Bayes," vol. 5, no. 36, pp. 24–28, 2022.
- [12] F. Zafira, B. Irawan, and A. Bahtiar, "Penerapan Data Mining Untuk

- Estimasi Stok Barang Dengan Metode K-Means Clustering,” *JATI (Jurnal Mhs. Tek. Inform.*, vol. 8, no. 1, pp. 156–161, 2024, doi: 10.36040/jati.v8i1.8319.
- [13] H. S. D. Suparwito, R. (Universitas S. D. Gunawan, I. (Universitas S. D. Binanto, R. (Universitas S. D. Arum Kumalasanti, and W. (Universitas S. D. Widyastuti, *Pengantar Pembelajaran Mesin Menggunakan Bahasa Pemrograman Python*. 2023.
- [14] A. Ali, “Sistem Informasi Model Prediksi Penyakit Diabetes Menggunakan Algoritma Random Forest di Fasyankes”.
- [15] A. Setiawan, Z. H. Nst, Z. Khairi, and L. Efrizoni, “KLASIFIKASI TINGKAT RISIKO DIABETES MENGGUNAKAN ALGORITMA,” vol. 7, no. 2, pp. 263–271, 2024.
- [16] D. Cahya, P. Buani, and U. N. Mandiri, “Deteksi Dini Penyakit Diabetes dengan Menggunakan Algoritma Random,” vol. 12, no. 1, pp. 1–8, 2024.
- [17] T. Tukino and F. Fifi, “Penerapan Support Vector Machine Untuk Analisis Sentimen Pada Layanan Ojek Online,” *J. Desain Dan Anal. Teknol.*, vol. 3, no. 2, pp. 104–113, 2024, doi: 10.58520/jddat.v3i2.59.
- [18] E. Tohidi, R. Perdana Herdiansyah, E. Wahyudin, and K. Kaslani, “Analisa Sentimen Komentar Video Youtube Di Channel Tvonenews Tentang Calon Presiden Prabowo Subianto Menggunakan Support Vector Machine,” *JATI (Jurnal Mhs. Tek. Inform.*, vol. 8, no. 1, pp. 660–667, 2024, doi: 10.36040/jati.v8i1.8560.
- [19] F. Sains and U. Muhammadiyah, “Sistem Prediksi Penyakit Jantung Berbasis Web Menggunakan Metode SVM dan Framework Streamlit,” vol. 4, no. 2, pp. 442–452, 2023.
- [20] Ismai, *Data Mining: Pengolahan Data Menjadi Informasi dengan RapidMiner*. 2017.
- [21] A. Az, R. Septian, M. A. Saktiawan, and R. A. Saputra, “PREDIKSI PENYAKIT HIPERTENSI MENGGUNAKAN MACHINE LEARNING DENGAN ALGORITMA REGRESI LOGISTIK,” vol. 8, no. 6, pp. 12062–12068, 2024.
- [22] D. Anggita, S. Fauzan, E. L. Neri, and M. A. Maulana, “AKTIVITAS FISIK DAN PENINGKATAN BASAL METABOLISME RATE PADA PASIEN DIABETES MELITUS,” vol. 15, no. 02, pp. 80–85, 2024, doi: 10.54630/jk2.v15i1.320.
- [23] M Setiawan, *Sistem endokrin dan diabetes mellitus*. 2021.
- [24] G. Indriati and U. Riau, “GAMBARAN KOMPLIKASI DIABETES MELITUS PADA,” vol. 11, 2023.
- [25] M. Nasta’in, A. Munazilin, and A. Susanto, “Predict The Trend Of

- Indonesian People's Interest In Bitcoin Facing The 2024 Bitcoin Halving Using A Forecasting Algorithm," *J. Ibrahi*, vol. 1, no. 1, pp. 463–468, 2024.
- [26] A. Pajankar, P. Python, and D. Visualization, "Exploring Jupyter Notebook," pp. 17–29.
- [27] P. Voinov and M. Rigger, "Forest : Structural Code Editing with Multiple Cursors," pp. 137–152, 2022, doi: 10.1145/3563835.3567663.
- [28] F. Svendsen, "Collaborative code editing Tool : Design and Evaluation," no. June, 2022.
- [29] M. Fikri, N. Syahbani, and N. G. Ramadhan, "Klasifikasi Gerakan Yoga dengan Model Convolutional Neural Network Menggunakan Framework Streamlit," vol. 7, pp. 509–519, 2023, doi: 10.30865/mib.v7i1.5520.
- [30] R. Hanson, "Stmol : A component for building interactive molecular visualizations within streamlit web-applications," no. September, pp. 1–10, 2022, doi: 10.3389/fmolb.2022.990846.
- [31] D. S. Utami and A. Erfina, "ANALISIS SENTIMEN PINJAMAN ONLINE DI TWITTER MENGGUNAKAN ALGORITMA SUPPORT VECTOR MACHINE (SVM)," pp. 299–305, 2021.
- [32] Z. S. R Azhari, "Hubungan Aktivitas Fisik Dengan Kadar Glukosa Darah Pada Penyandang Diabetes Melitus Tipe 2 Di Wilayah Perumahan Bugel Mas Indah RW 009," vol. 2, no. 7, pp. 86–90, 2022.
- [33] F. Abdusyukur, "Penerapan Algoritma Support Vector Machine (Svm) Untuk Klasifikasi Pencemaran Nama Baik Di Media Sosial Twitter," *Komputa J. Ilm. Komput. dan Inform.*, vol. 12, no. 1, pp. 73–82, 2023, doi: 10.34010/komputa.v12i1.9418.

CURICULUM VITAE**Contact Me**

Email
ammarfarisi689@gmail.com

IG
[@ammarfrsii_](https://www.instagram.com/ammarfrsii_)

Facebook
ammar farisi

IDENTITAS

NAMA : Ammar Farisi

PRODI : Teknologi Informasi

FAKULTAS : Sains dan Teknologi

ALAMAT : Candikuning, Tabanan, Bali

ORANG TUA

AYAH : Sabihi

IBU : Nurul Hikmah

RIWAYAT PENDIDIKAN

TK/RA : RA Al-Hidayah Candikuning

SD/MI : MI Al-Hidayah Candikuning

SMP/MTs : MTs Al-Hidayah Candikuning

SMA/MA : SMA Ibrahimi Sukorejo

KEAHLIAN

- Web Developer (Front-End)
- Software House

LAMPIRAN-LAMPIRAN

A. Lampiran Dataset

	A	B	C	D	E	F	G	H	I
1	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFunction	Age	Outcome
2	6	148	72	35	0	336	627	50	1
3	1	85	66	29	0	266	351	31	0
4	8	183	64	0	0	233	672	32	1
5	1	89	66	23	94	281	167	21	0
6	0	137	40	35	168	431	2288	33	1
7	5	116	74	0	0	256	201	30	0
8	3	78	50	32	88	31	248	26	1
9	10	115	0	0	0	353	134	29	0
10	2	197	70	45	543	305	158	53	1
11	8	125	96	0	0	0	232	54	1
12	4	110	92	0	0	376	191	30	0
13	10	168	74	0	0	38	537	34	1
14	10	139	80	0	0	271	1441	57	0
15	1	189	60	23	846	301	398	59	1
16	5	166	72	19	175	258	587	51	1
17	7	100	0	0	0	30	484	32	1
18	0	118	84	47	230	458	551	31	1
19	7	107	74	0	0	296	254	31	1
20	1	103	30	38	83	433	183	33	0
21	1	115	70	30	96	346	529	32	1
22	3	126	88	41	235	393	704	27	0
23	0	99	81	0	0	254	288	50	0


Lampiran 1 Dataset Diabetes

Link Dataset :

(<https://www.kaggle.com/code/vincentlugat/pima-indians-diabetes-eda-prediction-0-906>)

B. Lampiran Kartu Bimbingan

KARTU BIMBINGAN TUGAS AKHIR / SKRIPSI
FAKULTAS SAINS & TEKNOLOGI
UNIVERSITAS IBRAHIMY
TAHUN AKADEMIK 2024/2025



NPM : 2021503008
 Nama : Ammar Faris
 Program Studi : Teknologi Informasi
 Judul TA / Skripsi : Prediksi Penyakit Diabetes menggunakan Algoritma Support Vector machine (SVM)

= CATATAN =

1. Dalam penyusunan Laporan TA / Skripsi, mahasiswa harus berkonsultasi dengan pembimbingnya secara bertahap.
2. Pada setiap konsultasi, kartu bimbingan harus dibawa dan diisi oleh pembimbing
3. Mahasiswa wajib Konsultasi selama penyusunan Laporan TA / Skripsi ke pembimbing Minimal 6 x
4. Waktu bimbingan dimulai sejak tahapan proposal sampai laporan kegiatan
5. Skedul TA / Skripsi dapat dilihat pada buku panduan penyusunan Laporan Kegiatan.

Pembimbing I : Ahmad Hamdi, M.Kom

NO	TANGGAL	CATATAN	PARAF
1	01 Juni 2025	Perbaikan Rumusan masalah dan tujuan Penelitian, landasan teori	[Signature]
2	08 Juni 2025	ACC Bab 1,2	[Signature]
3	05 Juni 2025	Penambahan bab 3 dan 4	[Signature]
4	07 Juni 2025	ACC bab 3,4	[Signature]
5	15 Juni 2025	Perbaikan bab 5 dan KTI	[Signature]
6	20 Juni 2025	ACC bab 5 dan KTI	[Signature]
7	02 Juli 2025	ACC / Publish jurnal	[Signature]

Pembimbing II : Hermanto, M.Kom

NO	TANGGAL	CATATAN	PARAF
1	01 Juni 2025	Revisi bab 1,2,3	[Signature]
2	04 Juni 2025	ACC bab 1,2,3	[Signature]
3	10 Juni 2025	Perbaikan Program (akurasi model)	[Signature]
4	12 Juni 2025	Revisi bab 4,5	[Signature]
5	18 Juni 2025	ACC bab 4,5	[Signature]
6	01 Juli 2025	Revisi jurnal	[Signature]
7	05 Juli 2025	ACC jurnal / Publish	[Signature]

Lampiran 2 Kartu Bimbingan

C. Lampiran SKHCP



PONDOK PESANTREN SALAFIYAH SYAFI'YAH SUKOREJO
UNIVERSITAS IBRAHIMY
PERPUSTAKAAN IBRAHIMY
NPP. 3512142F2006567
Jl. KHR. Syaiful Arifin No. 1-2 PO. Box. 2 Kode Pos. 68374 Phone (0338) 452666 Fax. (0338) 453068
SUMBEREJO BANYUPUTIH SITUBONDO JAWA TIMUR



SURAT KETERANGAN HASIL PEMERIKSAAN PLAGIASI

Yang bertanda tangan di bawah ini

Nama : Muhammad Ali Ridla, M.Kom.
Jabatan : Kepala Perpustakaan

Menyatakan dengan sebenarnya bahwa:

NPM : 2021503008
Nama : AMMAR FARISI
Fakultas : Sains dan Teknologi
Prodi : Teknologi Informasi
Kecamatan : BATURITI
Kabupaten : KAB. TABANAN
Provinsi : Bali
Judul Skripsi : PREDIKSI PENYAKIT DIABETES
MENGUNAKAN ALGORITMA SUPPORT
VECTOR MACHINE (SVM)

Dengan dosen Pembimbing :

1. Ahmad Homaidi, M.Kom.
2. Hermanto, M.Kom

Telah dilakukan cek plagiasi di Perpustakaan Universitas Ibrahimy dengan persentase plagiasi terakhir sebesar 17% .

Demikian Surat Keterangan ini dibuat untuk dipergunakan sebagaimana mestinya.

Sukorejo, 9 Juli 2025
Kepala Perpustakaan,



Dokumen ini telah ditandatangani secara elektronik.

Muhammad Ali Ridla, M.Kom.



UU ITE No 11 Tahun 2008 Pasal 5 Ayat 1
"Informasi Elektronik dan/atau Dokumen Elektronik dan/atau hasil cetaknya merupakan alat bukti yang sah."

Lampiran 3 SKHCP

D. Lampiran LOA (Letter Of Acceptance)



Letter of Acceptance

Nomor: 786 /JTMIT/LoA/2025

Kepada Yth:

Ammar Farisi, Ahmad Homaidi, Hermanto

Jurusan Teknologi Informasi, Fakultas Sains dan Teknologi, Universitas Ibrahimy Situbondo

Terima kasih atas partisipasi Bapak/Ibu dalam mengirimkan artikel ilmiah ke **Jurnal Teknologi dan Manajemen Industri Terapan (JTMIT) e-ISSN: 2829-0038** dengan judul;

"Prediksi Penyakit Diabetes Menggunakan Algoritma Support Vector Machine (SVM)"

Berdasarkan hasil evaluasi dan penelaahan oleh tim reviewer, dengan ini kami menyatakan bahwa artikel tersebut **DITERIMA** untuk diterbitkan pada Jurnal Teknologi dan Manajemen Industri Terapan (JTMIT) **Volume 4, Nomor 3, Edisi September 2025** dalam versi elektronik.

Demikian informasi ini disampaikan. Atas perhatian dan kontribusi Bapak/Ibu, kami ucapkan terima kasih.

Pekanbaru, 24 Juni 2025,

Editor In Chief



Jurnal Teknologi dan
Manajemen Industri Terapan

Muhammad Rizki, MT., MBA, IPM., ASEAN Eng



Lampiran 4 LOA (Letter Of Acceptance)

E. Lampiran Segmen Program Streamlit

```
import pickle
import streamlit as st
import mysql.connector

# Koneksi ke MySQL (XAMPP)
conn = mysql.connector.connect(
    host="localhost",
    user="root",
    password="",
    database="db_diabetes"
)
c = conn.cursor()

# Fungsi simpan data ke MySQL
def insert_data(pregnancies, glucose,
bloodpressure, skinthickness, insulin, bmi,
dpf, age, result):
    sql = '''INSERT INTO diabetes_record (
        pregnancies, glucose,
        bloodpressure, skinthickness,
        insulin, bmi, dpf, age, result
        ) VALUES (%s, %s, %s, %s, %s, %s, %s,
        %s, %s)'''
    val = (pregnancies, glucose, bloodpressure,
skinthickness, insulin, bmi, dpf, age, result)
    c.execute(sql, val)
    conn.commit()
```

```
# Fungsi hapus data
def delete_data(record_id):
    c.execute("DELETE FROM diabetes_record
WHERE id = %s", (record_id,))
    conn.commit()

# Fungsi update data
def update_data(record_id, new_result):
    c.execute("UPDATE diabetes_record SET
result = %s WHERE id = %s", (new_result,
record_id))
    conn.commit()

# Load model
diabetes_model =
pickle.load(open('diabetes_model.sav', 'rb'))

# Streamlit
st.title('Prediksi Penyakit Diabetes')

col1, col2 = st.columns(2)
with col1:
    Pregnancies = st.text_input('Pregnancies',
placeholder="")
    BloodPressure = st.text_input('Blood
Pressure', placeholder="")
    Insulin = st.text_input('Insulin',
placeholder="")
    DiabetesPedigreeFunction =
st.text_input('Diabetes Pedigree Function',
placeholder="")
```

```
with col2:
    Glucose = st.text_input('Glucose',
placeholder="")
    SkinThickness = st.text_input('Skin
Thickness', placeholder="")
    BMI = st.text_input('BMI', placeholder="")
    Age = st.text_input('Age', placeholder="")

# Tombol prediksi
if st.button('Test Prediksi Diabetes'):
    try:
        inputs = [Pregnancies, Glucose,
BloodPressure, SkinThickness,
        Insulin, BMI,
DiabetesPedigreeFunction, Age]
        if '' in inputs:
            st.warning("Semua kolom wajib
diisi.")
        else:
            values = [float(i) for i in inputs]
            prediction =
diabetes_model.predict([values])
            diagnosis = 'pasien terkena
diabetes' if prediction[0] == 1 else 'pasien
tidak terkena diabetes'
            st.success(f"Hasil Prediksi:
{diagnosis}")
            insert_data(*inputs, diagnosis)
    except ValueError:
```

```
        st.error("Input harus berupa angka
valid.")
    except Exception as e:
        st.error(f"Terjadi kesalahan: {e}")

# Tampilkan data + edit / hapus
if st.button('Lihat Data dari Database'):
    try:
        c.execute('SELECT * FROM
diabetes_record ORDER BY id DESC')
        records = c.fetchall()
        if records:
            st.subheader("Data Tersimpan")
            for row in records:
                st.markdown("---")
                st.write(f"ID: {row[0]}")
                st.write(f"Pregnancies:
{row[1]}, Glucose: {row[2]}, BloodPressure:
{row[3]}, "
                    f"BMI: {row[6]}, Age:
{row[8]}")
                st.write(f"Hasil:
**{row[9]**")

            col1, col2 = st.columns(2)
            with col1:
                new_result =
st.text_input(f'Edit hasil ID {row[0]}',
value=row[9], key=f'edit_{row[0]}')
                if st.button(f'Update ID
{row[0]}', key=f'update_{row[0]}'):
```

```
update_data(row[0],
new_result)

st.success(f'Data ID
{row[0]} berhasil diperbarui.')

st.experimental_rerun()

with col2:
    if st.button(f'Hapus ID
{row[0]}', key=f'delete_{row[0]}'):
        delete_data(row[0])
        st.warning(f'Data ID
{row[0]} dihapus.')
        st.experimental_rerun()

    else:
        st.info("Belum ada data yang
tersimpan.")
except Exception as e:
    st.error(f"Gagal menampilkan data:
{e}")ss
```

Lampiran 5 Segmen Program Streamlit