

# **IMPLEMENTASI ALGORITMA NEURAL NETWORK UNTUK PREDIKSI PENYAKIT DIABETES**



**NAMA : RIZKYANDY WIBOWO**

**NIM : 191011402578**

**PROGRAM STUDI TEKNIK INFORMATIKA  
FAKULTAS ILMU KOMPUTER  
UNIVERSITAS PAMULANG  
KOTA TANGERANG SELATAN  
2022/2023**

## Daftar Isi

<b>BAB I</b>	4
1.1 Latar Belakang	4
1.2 Batasan Masalah	6
1.3 Rumusan Masalah	6
1.4 Tujuan Penelitian	7
1.5 Manfaat Penelitian	7
1.6 Metodologi Penelitian	7
1.7 Sistematika Penulisan	8
<b>BAB II</b>	9
2.1 Landasan Teori	9
2.2 Landasan Teori	12
2.2.1 <i>Data Mining</i>	12
2.2.2 <i>Machine Learning</i>	14
2.2.3 <i>Python</i>	14
2.2.4 <i>Library Python</i>	16
2.2.5 Neural Network	18
2.2.6 Pengujian Model	20
2.2.7 Framework	22
2.2.8 Aplikasi Pendukung	22
2.2.9 Kerangka Pemikiran	23
2.3 Tinjauan Objek	24
2.3.1 Diabetes Melitus	24
<b>BAB III</b>	25
3.1 Analisa Kebutuhan	25
3.1.1 Pengumpulan Data(Dataset)	29
3.1.2 Pengolahan Data Awal	31
3.2 Pearancangan Penelitian	32
<b>BAB IV</b>	38
<b>IMPLEMENTASI DAN HASIL PENGUJIAN</b>	38
4.1 Hasil	38
4.1.1 Persiapan Data	38
4.2 Pembahasan	38
4.2.1 Pengolahan Data	38
4.2.2 Eksplorasi Data	38

4.2.3 Pembersihan Data .....	<b>Error! Bookmark not defined.</b>
4.2.4 Membangun Model.....	43
4.2.5 Pengujian Model.....	44
4.3 Evaluasi dan Validasi Hasil Model .....	<b>Error! Bookmark not defined.</b>
4.4 Membangun Web App .....	45
4.4.1 Desain <i>User Interface (UI/UX)</i> .....	46
4.4.2 Implementasi Aplikasi Web .....	48
4.4.3 Pengujian Aplikasi.....	49
4.4.4 Analisis dan Evaluasi.....	49
<b>BAB V .....</b>	<b>51</b>
<b>KESIMPULAN DAN SARAN .....</b>	<b>51</b>
5.1 Kesimpulan.....	51
5.2 Saran.....	51
<b>DAFTAR PUSTAKA.....</b>	<b>Error! Bookmark not defined.</b>
Hasan, K., Alam, A., Hussein, D., & Hasan, M. (2020). Diabetes Prediction Using Ensembling of Different Machine Learning Classifiers. <i>IEEE Access</i> 8, 76516-76531. <b>Error!</b> <b>Bookmark not defined.</b>	
Daftar Riwayat Hidup .....	56

# **BAB I**

## **PENDAHULUAN**

### **1.1 Latar Belakang**

Diabetes merupakan suatu penyakit tidak menular yang cukup serius di mana pankreas tidak dapat memproduksi insulin secara maksimal. Diabetes dapat menyerang siapa saja tanpa mengenal usia baik lansia, orang dewasa, maupun anak-anak yang ditandai dengan meningkatnya kadar gula (glukosa) darah dalam tubuh manusia. Diabetes dapat disebabkan oleh banyak faktor seperti tekanan darah tinggi, kadar gula berlebih, berat badan, Riwayat keturunan diabetes, usia, jumlah kehamilan seseorang, ketebalan lipatan kulit, jumlah kadar insulin dalam tubuh, kurangnya aktivitas fisik dan pola hidup, serta diet tidak sehat. Faktor-faktor tersebut merupakan variabel yang digunakan dalam penelitian ini untuk membuat sistem cerdas yang dapat memprediksi penyakit diabetes (Cahyani & dkk, 2022)

Strategi klasifikasi digunakan secara luas di bidang medis untuk mengklasifikasikan data ke dalam kelas yang berbeda menurut beberapa kendala yang secara komparatif merupakan pengklasifikasi individu. Diabetes mellitus (DM), menurut definisi World Health Organization (WHO), adalah penyakit degeneratif kronis yang disebabkan oleh produksi insulin yang tidak mencukupi di pankreas atau oleh ketidakmampuan tubuh untuk secara efektif menggunakan insulin yang diproduksi, mengambil hyperglycemia (peningkatan glukosa darah) sebagai indikator utama. Karena gejalanya yang mirip dengan kondisi sakit biasa, banyak orang yang tidak menyadari bahwa mereka mengidap penyakit diabetes dan bahkan sudah mengarah pada komplikasi. Untuk memastikan bahwa seseorang apakah mengidap diabetes atau tidak maka perlu diagnosis dokter melalui cek darah. Bagi orang awam, setidaknya harus mengenal beberapa gejala yang biasanya mengiringi penyakit diabetes ini seperti, sering buang air kecil, mudah merasa haus, mudah merasa lapar, turunnya berat badan secara drastis, kulit kering, penyembuhan luka relatif lama, dan adanya gangguan penglihatan. Hampir setengah dari semua penderita diabetes memiliki faktor keturunan, yang merupakan salah satu ciri terpenting DM. Salah satu machine learning yang dapat membantu dalam mendeteksi resiko diabetes adalah neural network. Neural Network adalah sebuah cabang dari kecerdasan buatan (artificial

intelligence) yang cara kerjanya meniru cara kerja syaraf-syaraf otak manusia. Dengan cara ini, Neural Network memberikan program komputer sebuah kemampuan untuk bisa mengenali pola dan menyelesaikan berbagai masalah (Aprilia & dkk, 2021).

*Decision Tree* adalah algoritma dalam pembelajaran mesin yang digunakan untuk pengambilan keputusan dan prediksi. Pohon (*tree*) adalah sebuah struktur data yang terdiri dari simpul (*node*) dan rusuk (*edge*). Simpul pada sebuah pohon dibedakan menjadi tiga, yaitu simpul akar (*root node*), simpul percabangan/ internal (*branch/ internal node*) dan simpul daun (*leaf node*) (Eska, 2016). Kekurangan algoritma *Decision Tree* cenderung mempelajari pola yang sangat spesifik pada data pelatihan, yang dapat mengakibatkan overfitting pada dataset tersebut. Ini berarti model akan berkinerja buruk pada data baru yang belum pernah dilihat sebelumnya. *Decision Tree* sensitif terhadap perubahan kecil dalam data pelatihan, sedikit perubahan dalam data dapat menghasilkan struktur pohon yang berbeda, yang pada gilirannya dapat menghasilkan hasil yang berbeda.

*Naive Bayes* merupakan salah satu algoritma metode pengklasifikasian suatu probabilitas dan statistik yang diperoleh Thomas Bayes seorang ilmuwan Inggris dengan cara melakukan prediksi peluang di masa depan berdasarkan pengalaman pada masa sebelumnya (Bustami, 2014). Kelemahan pada algoritma ini yaitu sensitivitas terhadap data latih, jika jumlah data latih sangat terbatas atau terdapat *outlier* yang signifikan, dapat membuat estimasi probabilitas dapat menjadi tidak akurat. Algoritma *Naive Bayes* cenderung hanya memerhatikan hubungan statistik antara fitur-fitur dalam data latih, tanpa memperhatikan konteks atau makna di balik data tersebut. Ini dapat mengurangi kemampuan algoritma untuk memahami aspek kontekstual dari data.

*Support Vector Machine* (SVM) adalah sistem pembelajaran yang menggunakan ruang hipotesis berupa fungsi-fungsi linier dalam sebuah ruang fitur (*feature space*) berdimensi tinggi, dilatih dengan algoritma pembelajaran yang didasarkan pada teori optimasi dengan mengimplementasikan *learning bias* yang berasal dari teori pembelajaran statistik (Munawarah, Soesanto, & Faisal, 2016). Kekurangan dari metode SVM adalah sulit diaplikasikan untuk dataset dengan jumlah dimensi yang sangat besar (Aulia, Aprianti, Supriyanto, & Rozikin, 2022).

*K-Nearest Neighbor* (KNN) merupakan metode klasifikasi terhadap sekumpulan data berdasarkan pembelajaran data yang sudah terklasifikasi sebelumnya (Admojo & Ahsanawati, 2020). Kekurangan pada metode ini yaitu KNN harus menyimpan semua data

*training* dalam memori dan melakukan perhitungan jarak dengan semua data tersebut, algoritma ini mungkin menjadi kurang efisien pada dataset yang sangat besar.

*Neural Network* adalah sebuah model komputasi yang terinspirasi oleh struktur dan fungsi jaringan saraf biologis dalam otak manusia, *neural network* mengandung elemen pemrosesan dan pembobotan yang saling terhubung. Setiap lapisan dalam jaringan berisi oleh kelompok elemen pemrosesan (Hadianto, Novitasari, & Rahmawati, 2019), kelebihan dari metode ini yaitu mampu mengenali dan memahami pola-pola yang sangat kompleks dan abstrak dalam data, melalui lapisan-lapisan dan hubungan antara unit-unit pemrosesan, *neural network* dapat memahami representasi data dalam berbagai tingkat abstraksi dan hirarki. Ini berarti mereka dapat menangkap fitur-fitur yang semakin kompleks dan informasi yang semakin berguna pada setiap lapisan

## **1.2 Batasan Masalah**

Untuk memperjelas arah permasalahan yang akan dibahas dari rumusan masalah diatas, maka batasan masalah pada penelitian ini hanya membahas sebagai berikut:

- a) Pada penelitian ini data yang digunakan merupakan data dari penelitian sebelumnya, yaitu dataset yang diambil pada situs website (<https://www.kaggle.com/datasets/iammustafatz/diabetes-prediction-dataset>) dan tidak membahas menggunakan data real.
- b) Penelitian ini berfokus pada pembuatan model machine learning yang digunakan untuk melakukan deteksi diabetes, dan juga membahas tentang implementasi model ke dalam sebuah web app pendeteksi.
- c) Dalam penelitian ini hanya membahas deteksi Diabetes dengan menggunakan algoritma neural network tidak membahas tentang penggunaan algoritma atau metode lainnya.

Dengan batasan-batasan ini, penelitian akan lebih fokus pada pengembangan sistem deteksi diabetes yang dapat membantu mengidentifikasi ciri ciri, memberikan respons yang cepat, dan mengurangi dampak yang dihasilkan oleh serangan ransomware.

## **1.3 Rumusan Masalah**

- a. Apakah dengan mengimplementasikan algoritma Neural Network cukup akurat untuk mendeteksi diabetes melitus?

- b. Apa saja faktor-faktor yang berkontribusi terhadap keberhasilan penggunaan Neural Network dalam mendeteksi diabetes?

#### **1.4 Tujuan Penelitian**

- a) Mengembangkan model deteksi diabetes menggunakan algoritma Neural Network.
- b) Membersihkan, dan menganalisis data klinis untuk melatih dan menguji model deteksi diabetes.
- c) Mempelajari faktor-faktor yang memengaruhi kinerja Neural Network dalam konteks deteksi diabetes.

#### **1.5 Manfaat Penelitian**

Adap beberapa manfaat penelitian yang dibuat oleh peneliti antara lain :

- a) Untuk Universitas:

Dengan adanya Skripsi yang mengangkat tema machine learning dapat digunakan sebagai referensi untuk mahasiswa generasi berikutnya supaya menjadi bahan pembuatan penelitan yang akan dibuat terutama dibidang machine learning dan Kesehatan.

- b) Untuk Peneliti :

Agar dapat mempraktikan dan menguji ilmu yang dipelajari selama kuliah khususnya dibidang machine learning dan data science.

- c) Untuk Masyarakat :

Menambah wawasan Masyarakat dibidang kesehatan tentang seberapa berbahayanya diabetes melitus dan dampak yang ditimbulkan dan juga pengetahuan tentang machine learning.

#### **1.6 Metodologi Penelitian**

Metodologi yang digunakan pada penelitian ini melalui beberapa tahapan ,yaitu :

- a) Observasi

Observasi adalah pengamatan atau pencatatan fenomena, peristiwa, atau objek dalam dunia nyata untuk tujuan pengumpulan data atau pemahaman lebih lanjut. Mengamati

dataset yang dikumpulkan untuk mengidentifikasi dan mengetahui data klinis antara orang yang tidak terkena diabetes dan yang terkena diabetes. Metode observasi yang digunakan yaitu observasi non-partisipan dengan mengambil data langsung melalui dataset yang diambil dari (link website dataset).

b) Studi Pustaka

Tahap ini dilakukan dengan cara mengkaji dan mempelajari literatur dan referensi berupa naskah ilmiah, buku konsep machine learning, serta cara kerja algoritma Neural Network sehingga dapat menunjang metodologi yang akan diterapkan pada penelitian.

## **1.7 Sistematika Penulisan**

Untuk memudahkan dalam proses penyusunan tugas akhir, penulis menyusun ke dalam lima bab. Setiap bab tersebut secara keseluruhan saling berkaitan satu sama lain, dimana diawali dengan bab pendahuluan dan diakhiri dengan bab penutup yang berupa kesimpulan dan saran. Maka dibuat suatu sistematika penulisan sebagai berikut :

### **BAB I           PENDAHULUAN**

Bab ini berisi penjelasan secara sistematis mengenai landasan topik penelitian yang meliputi latar belakang, identifikasi masalah, rumusan masalah dan batasan masalah, tujuan penelitian, manfaat penelitian, metodologi penelitian, dan yang terakhir sistematika penulisan.

### **BAB II          TINJAUAN PUSTAKA**

Bab ini menguraikan tentang penelitian sebelumnya yang pernah dilakukan dan landasan teori penelitian mengenai data *mining*, *Machine Learning*, dan metode Algoritma *Neural Network* yang berkaitan langsung dengan penelitian ini.

### **BAB III        METODE PENELITIAN**

Bab ini menjelaskan secara sistematis, bagaimana proses penelitian dilakukan. Penjelasan pada bab ini meliputi Analisa kebutuhan sistem dan perancangan penelitian.

### **BAB IV        IMPLEMENTASI DAN PENGUJIAN**

Bab ini menjelaskan hasil pengujian yang dilakukan serta analisis dari data yang diperoleh dari hasil pengujian yang dilakukan.

### **BAB V          KESIMPULAN DAN SARAN**

Bab ini menjelaskan hasil akhir dari penelitian ini dan saran untuk penelitian yang akan datang.



## BAB II

### 2.1 Landasan Teori

Penelitian yang dilakukan oleh Kamrul hasan dkk. pada tahun 2020 dengan judul penelitian “*Diabetes Prediction Using Ensembling of Different Machine Learning Classifiers*” metode yang digunakan pada penelitian ini adalah *k-nearest neighbors*. Sedangkan masalah yang dihadapi pada penelitian ini adalah meningkatkan prediksi diabetes di mana bobot diperkirakan dari Kurva Area Di Bawah ROC (AUC) model ML yang sesuai. Adapun tujuan penelitian ini adalah dapat menghasilkan kinerja yang lebih baik dalam prediksi diabetes. adapun hasil dari penelitian ini didapatkan hasil tingkat akurasi sebesar 95% (Hasan, Alam, Hussein, & Hasan, 2020).

Pada penelitian yang dibuat oleh Mitushi soni pada tahun 2020 dengan judul penelitian “*Diabetes Prediction using Machine Learning Techniques*” metode yang dipakai pada penelitian ini adalah *Random Forest*. Sedangkan masalah yang dihadapi pada penelitian ini adalah mencari model *machine learning* manakah yang paling akurat untuk system deteksi diabetes. Adapun tujuan penelitian ini adalah untuk menyelidiki model yang dapat memprediksi diabetes dengan akurasi yang lebih baik. Pendekatan yang diusulkan menggunakan berbagai metode pembelajaran klasifikasi dan ansambel yang menggunakan pengklasifikasi SVM, Knn, Random Forest, Decision Tree, Logistic Regression dan Gradient Boosting. Dan akurasi klasifikasi telah tercapai sebesar 77% (Soni & Varma, 2020).

Penelitian yang dikerjakan oleh P. Nagaraj dan P. Deepalakshmi pada tahun 2021 dengan judul “*Diabetes Prediction Using Enhanced SVM and Deep Neural Network Learning Techniques: An Algorithmic Approach for Early Screening of Diabetes*”. metode yang digunakan pada penelitian ini adalah *SVM* dan *Deep Neural network*. Sedangkan masalah yang dialami pada penelitian tersebut adalah meningkatkan efektifitas yang lebih baik dengan *mengupgrade vector deep learning*. untuk mencapai akurasi tinggi dan waktu pemrosesan yang lebih singkat dengan kumpulan data yang sangat besar. Kemudian tujuan dari penelitian ini adalah untuk mencapai akurasi tinggi dan waktu pemrosesan yang lebih singkat dengan kumpulan data yang sangat besar. Adapun hasil yang didapat dari

eksperimen adalah akurasi klasifikasi maksimum yang diperoleh melalui ESVM dan DNN sebesar 98,45% (Nagaraj & Deepalakshmi, 2021)

Penelitian yang dikerjakan oleh Arianna Dagliati dkk pada tahun 2018 yang berjudul “*Machine Learning Methods to Predict Diabetes Complications*” metode yang digunakan pada penelitian ini adalah *Random Forest* dan *Logistic Regression*. sedangkan masalah yang dihadapi pada penelitian berikut adalah memprediksi komplikasi diabetes dengan menggunakan metode *Random forest* dan *Logistic Regression* untuk menangani ketidak seimbangan pada class. Mengenai tujuan dari penelitian ini adalah menggabungkan data mining dan machine learning untuk memprediksi komplikasi diabetes tipe 2 dengan lebih efektif. kemudian hasil yang diperoleh dari penelitian ini adalah untuk model LR retinopati didapatkan prediksi sebesar 99% sedangkan untuk LR neuropati sebesar 95% (Dagliati, Marini, & dkk, 2018)

Penelitian yang dilakukan oleh Radhanath Patra dan Bonomali khuntia pada tahun 2021 dengan judul “*Analysis and Prediction Of Pima Indian Diabetes Dataset Using SDKNN Classifier Technique*”.metode yang digunakan dalam penelitian ini adalah *standard deviation K-nearest neighbour*.Sedangkan masalah yang dialami pada penelitian tersebut adalah menggunakan rumus perhitungan jarak terbaru untuk menemukan nearest neighbour pada KNN. Kemudian tujuan dari penelitian ini adalah untuk mencapai akurasi yang lebih tinggi dengan menggunakan metode KNN untuk memprediksi diabetes.Mengenai hasil yang didapat dari penelitian ini adalah akurasi klasifikasi rata-rata memberikan hasil sebesar 83,2% sebuah peningkatan besar dibandingkan Teknik konvensional lainnya (Patra & Khuntia, 2021).

Penelitian yang dilakukan oleh Madhusmita Rout dan Amandeep Kaur pada tahun 2020 dengan judul “*Evaluating Machine Learning Methods for Predicting Diabetes among Female Patients in Bangladesh*”.lalu metode yang digunakan dalam penelitian ini adalah *Naïve Bayes*.Sedangkan masalah yang dialami pada penelitian tersebut adalah mengatasi perbedaan dataset antara negara yang berbeda. Kemudian tujuan dari penelitian ini adalah untuk mendeteksi diabetes pada pasien menggunakan Machine Learning.adapun hasil yang diperoleh dari penelitian tersebut adalah nilai akurasi tertinggi sebesar 81,2% dengan menggunakan metode KNN (Kaur & Rout, 2020)

Peneliti	Tahun	Topik Penelitian	Metode	Hasil Penelitian
Kamrul hasan dkk	2020	<i>Diabetes Prediction Using Ensembling of Different Machine Learning Classifiers</i>	<i>K-Nearest Neighbour, Decission Tree, Random Forest, Xgboost dan Naive Bayes</i>	hasil dari penelitian ini didapatkan hasil tingkat akurasi sebesar 95% dari model Xgboost
Mitushi soni	2020	<i>Diabetes Prediction using Machine Learning Techniques</i>	<i>Support Vector Machine, K-Nearest Neighbour, Decission Tree, Logistic Regression, Random Forest dan Gradient Boost</i>	Hasil perbandingan tertinggi dari model penelitian ini adalah Forest dengan akurasi sebesar 77%
P. Nagaraj dan P. Deepalakshmi	2021	<i>Diabetes Prediction Using Enhanced SVM and Deep Neural Network Learning Techniques: An Algorithmic Approach for Early Screening of Diabetes</i>	<i>Decision Tree, Neural Networks, Logistic Regression, Naive Bayes dan Support Vector Machine</i>	Hasil dari penelitian ini diperoleh nilai akurasi klasifikasi maksimum yang diperoleh melalui ESVM dan DNN sebesar 98,45%
Arianna Dagliati dkk	2018	<i>Machine Learning Methods to Predict Diabetes Complications</i>	<i>Support Vector Machine, Random Forest, Naive Bayes dan Logistic Regression</i>	hasil yang diperoleh dari penelitian ini adalah untuk model LR retinopati didapatkan prediksi sebesar 99% sedangkan untuk LR neuropati sebesar 95%
Radhanath Patra dan Bonomali khuntia	2021	<i>Analysis and Prediction Of Pima Indian Diabetes Dataset Using SDKNN</i>	<i>K-Nearest Neighbour</i>	hasil yang didapat dari penelitian ini adalah akurasi klasifikasi rata-rata memberikan hasil sebesar 83,2% sebuah

		<i>Classifier Technique</i>		peningkatan besar dibandingkan Teknik konvensional lainnya
Madhusmita Rout dan Amandeep Kaur	2020	<i>Evaluating Machine Learning Methods for Predicting Diabetes among Female Patients in Bangladesh</i>	<i>K-Nearest Neighbour, Decision Tree, Random Forest dan Naive Bayes</i>	hasil yang diperoleh dari penelitian tersebut adalah nilai akurasi tertinggi sebesar 81,2% dengan menggunakan metode KNN

## 2.2 Landasan Teori

### 2.2.1 Data Mining

*Data mining* adalah analisa terhadap data untuk menemukan hubungan yang jelas serta menyimpulkannya yang belum diketahui sebelumnya dengan cara terkini dipahami dan berguna bagi pemilik data tersebut. Data mining adalah metoda yang digunakan untuk mengekstraksi informasi prediktif tersembunyi pada *database*, ini adalah teknologi yang sangat potensial bagi perusahaan yang sangat potensial bagi perusahaan dalam memberdayakan data warehouse (Noviyanto, 2020).

*Data Mining* adalah suatu istilah yang digunakan untuk menguraikan penemuan pengetahuan didalam database, dapat diartikan bahwa *Data Mining* digunakan untuk ekstraksi dari informasi penting yang tersembunyi dari database yang besar. Dalam beberapa tahun belakangan ini, kemajuan dalam beberapa bidang ilmu pengetahuan seperti *science*, *business* dan lain-lain, dampaknya adalah peningkatan koleksi database sehingga kumpulan data yang demikian banyak dapat didayagunakan untuk pengambilan keputusan. (Jollyta, Ramdhan, & Zarlis, 2020)

#### 2.2.1.1 Tahap-tahap Data Mining :

Ada Tujuh Tahapan dari *data mining* sebagai berikut (Romadhon & Kodar, 2020);

- Pembersihan Data, Proses untuk membuang data yang tidak valid dan yang tidak sesuai untuk digunakan.
- Integrasi Data, Merupakan proses penggabungan sumber-sumber data.
- Seleksi Data, Proses pengambilan data-data yang sesuai untuk analisis.

- d. Transformasi Data, Pengubahan data menjadi bentuk yang tepat sesuai data mining.
- e. Proses Mining, Proses awal pengkajian metode.
- f. Evaluasi Data, Proses untuk mengidentifikasi pola yang menarik untuk mewakili pengetahuan dari data yang tersembunyi.
- g. Presentasi pengetahuan, merupakan proses presentasi pengetahuan dan Teknik visualisasi dapat membantu memberitahukan hasil dari data mining.

#### **2.2.1.2 Metode Data Mining**

Metode data mining merupakan suatu proses utama yang digunakan saat metode diterapkan untuk menemukan pengetahuan berharga dari data, ada beberapa teknik dan sifat analisa yang dapat digolongkan dalam *data mining* (Muslim, et al., 2019) yaitu;

##### **a. Classification**

Klasifikasi merupakan teknik yang digunakan untuk menemukan model agar dapat menjelaskan atau membedakan konsep atau kelas data, dengan tujuan untuk dapat memperkirakan kelas dari suatu objek yang labelnya tidak diketahui. Metode klasifikasi yang sering digunakan yaitu, *Support Vector Machine, Multilayer Perceptron, Naive bayes, ID3, Ensemble Methode, dll.*

##### **b. Association**

*Association rule mining* adalah teknik data mining yang berfokus untuk menemukan aturan kesamaan dalam suatu kejadian. Contoh aturan asosiasi yang sering dijumpai adalah proses pembelian barang dagangan pada pusat perbelanjaan. Metode asosiasi yang umum digunakan adalah FP-Growth, *Coefficient of Correlation*, Chi Square, A Priori, dll.

##### **c. Clustering**

*Clustering* merupakan teknik dengan cara mengelompokkan data secara otomatis tanpa diberitahukan label kelasnya. *Clustering* dapat digunakan untuk memberikan label pada kelas data yang belum diketahui, karena *clustering* sering digolongkan sebagai metode *unsupervised learning*. Metode *clustering* yang sering digunakan yaitu, *K-Medoids, K-Means, Fuzzy C-Means, Self-Organizing Map (SOM)*, dll.

##### **d. Prediction**

memperkirakan suatu nilai di masa mendatang, misalnya memprediksi stok barang tiga tahun ke depan. Yang termasuk fungsi ini antara lain metode Neural Network, Decision Tree, dan k –Nearest Neighbor.

e. *Regression*

Digunakan untuk mencari pola dan menentukan nilai numerik. Metode yang sering digunakan adalah linear regression, logistic regression, support vector regression, dll.

f. *Description*

memberi gambaran secara ringkas terhadap sejumlah data yang bersakala besar dan memiliki banyak jenis. Termasuk di dalamnya metode *Decision Tree*, *Exploratory Data Analysis* dan *Neural Network*.

### **2.2.2 Machine Learning**

*Machine Learning* merupakan salah satu cabang ilmu Kecerdasan Buatan, khususnya yang mempelajari tentang bagaimana computer mampu belajar dari data untuk meningkatkan kecerdasannya. *Machine Learning* memiliki fokus pada pengembangan sebuah sistem yang mampu belajar sendiri untuk memutuskan sesuatu, tanpa harus berulang kali di program oleh manusia. Dengan metode tersebut, mesin tidak hanya menemukan aturan untuk perilaku optimal dalam pengambilan keputusan, namun dapat juga beradaptasi dengan perubahan yang terjadi. (Wahyono, 2018).

*Machine Learning* (ML) atau Mesin Pembelajaran adalah cabang dari AI yang fokus belajar dari data (*learn from data*), yaitu fokus pada pengembangan sistem yang mampu belajar secara “mandiri” tanpa harus berulang kali diprogram manusia. ML membutuhkan Data yang valid sebagai bahan belajar (ketika proses *training*) sebelum digunakan ketika *testing* untuk hasil *output* yang optimal (Cholissodin, Sutrisno, Soebroto, Hasanah, & Febiola, 2020).

### **2.2.3 Python**

*Python* merupakan bahasa pemrograman tingkat tinggi (*haight level language*) yang dikembangkan Oleh Guldo van Rossum pada tahun 1989 dan diperkenalkan untuk pertama kalinya pada tahun 1991. *Python* lahir atas dasar keinginan untuk mempermudah seorang programmer dalam menyelesaikan tugas-tugasnya dengan cepat. *Python* dirancang untuk

memberikan kemudahan yang sangat luar biasa kepada programmer baik dari segi efisiensi waktu, maupun kemudahan dalam pengembangan program khususnya dalam hal kompatibiliras dengan sistem (Wadi, 2015)

*Python* adalah bahasa pemrograman tingkat tinggi yang sering digunakan dalam berbagai bidang, termasuk pengembangan perangkat lunak, analisis data, kecerdasan buatan, pemrosesan bahasa alami, pengembangan web, dan sebagainya. *Python* memiliki sintaks yang mudah dipahami dan digunakan, serta berbagai *library* dan *framework* yang kuat yang membuatnya populer di kalangan pengembang. beberapa fitur dan kegunaan *Python* Menurut (Nelli, 2015) sebagai berikut :

- a. Sintaks yang Mudah: *Python* dirancang dengan sintaks yang mudah dibaca dan dipahami, mirip dengan bahasa Inggris. Ini membuatnya menjadi bahasa yang cocok untuk pemula dan mengurangi waktu yang dibutuhkan untuk mengembangkan kode.
- b. *Platform-Independent*: *Python* dapat dijalankan di berbagai platform, termasuk *Windows*, *macOS*, *Linux*, dan banyak lagi. Ini memungkinkan pengembang untuk membuat aplikasi yang dapat berjalan di berbagai lingkungan.
- c. Kaya *Library*: *Python* memiliki ekosistem yang kaya dengan banyak *library* dan modul yang tersedia, seperti *NumPy* untuk komputasi numerik, *pandas* untuk analisis data, *TensorFlow* dan *PyTorch* untuk kecerdasan buatan, *Django* dan *Flask* untuk pengembangan web, dan banyak lagi. *Library-library* ini membantu memperluas kemampuan *Python* dan mempercepat proses pengembangan.
- d. Analisis Data: *Python* sangat populer dalam analisis data dan ilmu data. Dengan *library* seperti *pandas*, *NumPy*, dan *matplotlib*, *Python* memungkinkan pengguna untuk membaca, memanipulasi, dan menganalisis data dengan mudah. Ini menjadikannya alat yang kuat untuk melakukan eksplorasi data, visualisasi, dan pembuatan model prediktif.
- e. Kecerdasan Buatan: *Python* adalah pilihan yang populer untuk pengembangan kecerdasan buatan. *Library* seperti *TensorFlow*, *Keras*, *PyTorch*, dan *scikit-learn* menyediakan alat yang kuat untuk melatih dan menerapkan model pembelajaran mesin dan jaringan saraf.
- f. Pengembangan Web: *Python* digunakan secara luas dalam pengembangan web. *Framework* populer seperti *Django* dan *Flask* menyediakan alat untuk membangun aplikasi web yang efisien dan aman. *Python* juga dapat digunakan untuk mengembangkan API, *backend*, dan bahkan *frontend* menggunakan *framework* seperti *Django REST framework* dan *Flask-RESTful*.

- g. Automasi Tugas: *Python* dapat digunakan untuk otomatisasi tugas sehari-hari. Dengan *Python*, Anda dapat menulis skrip untuk mengotomatisasi tugas rutin seperti pengolahan file, pengambilan data dari web, pemrosesan teks, dan sebagainya.

*Python* adalah bahasa yang serbaguna dan kuat, dengan komunitas yang besar dan aktif. Kelebihan-kelebihan di atas menjadikan *Python* pilihan yang baik untuk berbagai kebutuhan pemrograman, dari pengembangan perangkat lunak hingga analisis data dan kecerdasan buatan.

## **2.2.4 Library Python**

### **2.2.4.1 Pandas**

Panda bertujuan untuk menjadi blok bangunan tingkat tinggi mendasar untuk melakukan praktik, analisis data dunia nyata dengan *Python*. Selain itu, ia memiliki tujuan yang lebih luas untuk menjadi yang paling kuat dan fleksibel alat analisis / manipulasi data sumber terbuka tersedia dalam bahasa apa pun. (PyData, 2023).

### **2.2.4.2 Matplotlib**

*Matplotlib* adalah *library plot Python* 2D yang menghasilkan kualitas publikasi angka dalam berbagai format *hardcopy* dan lingkungan interaktif di seluruh *Platform*, *Matplotlib* adalah *library komprehensif* untuk membuat statis, animasi, dan visualisasi interaktif dengan *Python*. *Matplotlib* membuat segalanya menjadi mudah hal-hal mudah dan sulit mungkin (Hunter, Dale, Firing, & Droettboom, 2023). *Matplotlib* adalah *library Python* yang fokus pada visualisasi data seperti membuat plot grafik. *Matplotlib* pertama kali diciptakan oleh John D. Hunter dan sekarang telah dikelola oleh tim *developer* yang besar. Awalnya *matplotlib* dirancang untuk menghasilkan plot grafik yang sesuai pada publikasi jurnal atau artikel ilmiah. *Matplotlib* dapat digunakan dalam skrip *Ipython shell-server* aplikasi web, dan beberapa *tollkit graphical user interface* (GUI) lainnya (Rohman, 2019)

### **2.2.4.3 NumPy**

*NumPy* adalah proyek *open source* yang memungkinkan komputasi numerik dengan *Python*. Itu dibuat pada tahun 2005 membangun karya awal *library Numerik* dan *Numarray*. *NumPy* adalah paket dasar untuk komputasi ilmiah dengan *Python*. Ini adalah *Python library* yang menyediakan objek *array* multidimensi, berbagai turunan objek (seperti *array* dan matriks bertopeng), dan bermacam-macam rutinitas untuk operasi cepat pada *array*, termasuk matematika, logis, manipulasi bentuk, menyortir, memilih, I / O, transformasi *Fourier diskrit*,



*aljabar* linier dasar, operasi statistik dasar, simulasi acak dan banyak lagi. (R, Harris, Millman, & Vander, 2020).

#### **2.2.4.4 Seaborn**

*Seaborn* adalah sebuah pustaka *Python* yang digunakan untuk visualisasi data berbasis statistik. Pustaka ini dibangun di atas pustaka *matplotlib* dan dirancang khusus untuk membuat visualisasi data yang lebih menarik dan informatif dengan lebih sedikit kode, menurut (Waskom, 2021) *Seaborn* adalah *library* visualisasi data *Python* berdasarkan *matplotlib*. Ini menyediakan antarmuka tingkat tinggi untuk menggambar grafik statistik yang menarik dan informatif.

*Seaborn* menyediakan berbagai jenis plot yang telah diatur dengan baik dan memiliki gaya yang estetik secara default. Hal ini memungkinkan pengguna untuk dengan mudah membuat plot seperti scatter plot, line plot, bar plot, histogram, heatmap, dan banyak jenis plot lainnya.

#### **2.2.4.5 Scikit-Learn**

*Scikit-Learn* adalah modul *Python* yang mengintegrasikan banyak algoritma *machine learning*. Pustaka ini awalnya dikembangkan oleh Cournapeu pada tahun 2007, namun rilis pertama yang sebenarnya terjadi pada tahun 2010 (Nelli, Python Data Analytics, 2015). Terdapat beberapa pustaka *Python* yang menyediakan implementasi yang solid dari berbagai algoritma *machine learning*. Salah satu yang paling terkenal adalah *Scikit-Learn*, sebuah paket yang menyediakan versi efisien dari sejumlah besar algoritma umum. *Scikit-Learn* ditandai dengan API yang bersih, seragam, dan efisien, serta dokumentasi *online* yang sangat berguna dan lengkap. Keuntungan dari keseragaman ini adalah setelah Anda memahami penggunaan dasar dan sintaksis *Scikit-Learn* untuk satu jenis model, beralih ke model atau algoritma baru akan sangat mudah (VanderPlas, 2017).

#### **2.2.4.6 Keras**

Keras adalah pustaka pembelajaran mesin (*machine learning*) yang populer dan kuat yang awalnya ditulis dalam bahasa pemrograman *Python*. Pustaka ini dirancang untuk memudahkan pembuatan, pelatihan, dan evaluasi model jaringan saraf tiruan (*neural networks*) serta model pembelajaran mesin lainnya.

Keras dapat menyederhanakan *script TensorFlow* dan memberikan solusi yang jelas untuk permasalahan dalam mengembangkan model *deep learning* (Syarovy & Sutiarso,

2023). Keras banyak digunakan oleh berbagai organisasi ilmiah di seluruh dunia, termasuk CERN, NASA, dan NIH (Gulli & Pal, 2017).

#### **2.2.4.7 Tensorflow**

TensorFlow adalah platform open source end-to-end diperuntukan machine learning. Yang memiliki circle ekosistem alat, pustaka, dan sumber daya komunitas yang sangat fleksibel dan komprehensif yang memungkinkan seorang ahli ataupun pemula untuk mengembangkan dan membangun aplikasi dengan menerapkan model machine learning di platform TensorFlow. Peneliti mendorong machine learning mutakhir dan pengembang dengan mudah membangun dan menerapkan aplikasi yang didukung machine learning. TensorFlow selalu menyediakan jalur langsung ke produksi. Baik di server, perangkat edge, atau web, TensorFlow memungkinkan Anda melatih dan menerapkan model dengan mudah, apapun bahasa yang Anda gunakan (tensorflow, 2021).

#### **2.2.5 Neural Network**

*Neural Network* adalah model matematika yang terinspirasi dari cara kerja jaringan saraf dalam otak manusia. Ini adalah salah satu komponen inti dalam *machine learning* dan *artificial intelligence*. *Neural Network* merupakan salah satu metode dalam data mining yang memiliki keunggulan tingkat akurasi yang lebih baik atau optimal dibandingkan dengan metode lainnya (Handayani, Riandini, & Situmorang, 2022)

*Neural Network* adalah sebuah metode yang terinspirasi oleh struktur jaringan syaraf otak manusia, yang dirancang untuk meniru cara otak manusia melakukan pemrosesan dan penyimpanan informasi (Ramdhan, 2019). *Neural Network* terdiri dari *node* yang menggabungkan inputnya, yang bisa berupa variabel dari database atau output dari node lainnya. *Node-node* ini dapat dikelompokkan menjadi tiga lapisan yang sederhana, yaitu lapisan input, lapisan output, dan lapisan tersembunyi (Bhakti, 2019). Berikut adalah gambar jaringan syaraf sederhana dengan fungsi aktivasi F.

##### **2.2.5.1 Arsitektur Jaringan**

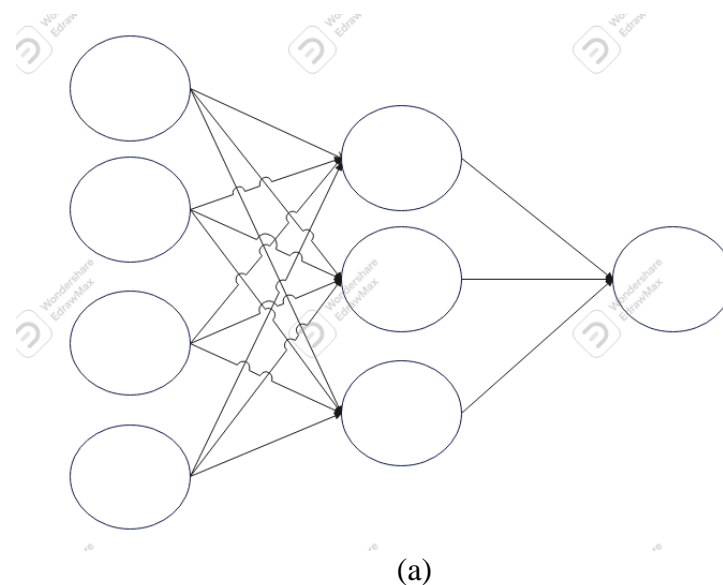
Menurut (Ataline Jeanethe Maya Hukubun, 2022) struktur Neural Network Neural Network terdiri dari 3 lapisan, yaitu :

- a. Lapisan input atau masukan (buffer) memiliki beberapa neuron input (perceptron), jumlahnya tergantung dataset yang dibutuhkan dalam proses pembelajaran. Neuron lapisan input tidak memiliki fungsi transfer, tapi ada faktor skala di setiap input untuk menormalkan sinyalnya dengan

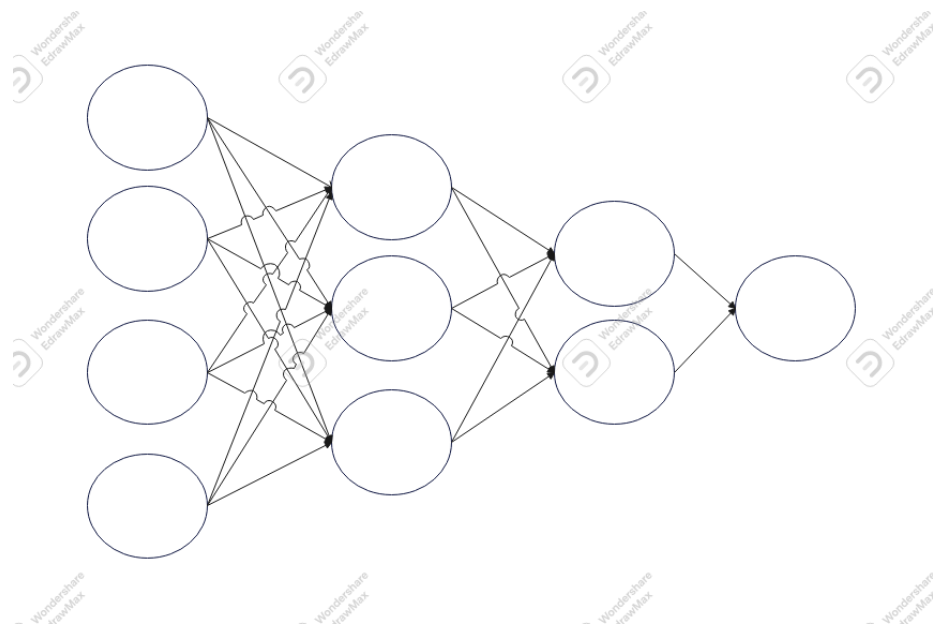
mentransmisikan sinyal yang dihitung ke lapisan tersembunyi lalu ke lapisan output.

- b. Lapisan tersembunyi berfungsi untuk menghubungkan lapisan input dan output, dengan melewati bobot yang dihitung ke lapisan keluaran. Kesalahan disajikan ke lapisan input melalui propagasi balik (umpan balik). Lapisan tersembunyi juga terdiri dari beberapa neuron tersembunyi.
- c. Lapisan output atau keluaran (buffer) berisi beberapa neuron keluaran (atau satu neuron), yang mewakili sebuah kelas dari kumpulan data keluaran.

Neural network bisa berisi satu atau lebih lapisan tersembunyi, seperti yang ditunjukkan pada gambar 1. Semua neuron di lapisan input terhubung ke neuron lapisan tersembunyi. Pemrosesan komputasi data di lapisan tersembunyi sangat kompleks dan tidak kita ketahui.



Gambar 1 Struktur jaringan syaraf tiruan : (a) berisi satu lapisan masukan, satu lapisan tersembunyi, dan satu lapisan keluaran. (b) satu lapisan masukan, dua lapisan tersembunyi, dan satu lapisan keluaran.



(b)

#### 2.2.5.2 Artificial Neural Network

*Artificial Neural Network Artificial* (ANN) atau Jaringan Syaraf Tiruan merupakan sebuah teknik atau pendekatan pengolahan informasi yang terinspirasi oleh cara kerja sistem saraf biologis, khususnya pada sel otak manusia dalam memproses informasi. Elemen kunci dari teknik ini adalah struktur sistem pengolahan informasi yang bersifat unik dan beragam untuk tiap aplikasi. Neural Network terdiri dari sejumlah besar elemen pemrosesan informasi (neuron) yang saling terhubung dan bekerja bersama-sama untuk menyelesaikan sebuah masalah tertentu, yang pada umumnya adalah masalah klasifikasi ataupun prediksi (Widiputra, 2016).

#### 2.2.5.3 Deep Neural Network

*Deep Neural Networks* adalah salah satu algoritma berbasis jaringan saraf yang dapat digunakan untuk pengambilan keputusan. Contoh yang dibahas kali ini adalah mengenai penentuan penerimaan pengajuan kredit sepeda motor baru berdasarkan kelompok data yang sudah ada (Pip Tools, 2024).

### 2.2.6 Pengujian Model

#### 2.2.6.1 F1-Score

*F1-score* adalah salah satu metrik evaluasi yang umum digunakan dalam bidang klasifikasi dalam pembelajaran mesin untuk memberikan gambaran yang lebih komprehensif tentang kinerja model dalam memprediksi kelas yang benar .

*F1-score* digunakan untuk mengukur kombinasi hasil *precision* dan *recall*, sehingga menjadi satu nilai pengukuran (Istighfarizky, et al., 2022). *F-1 score* membantu mengetahui kemampuan model dalam mengenali *false negative* dan *false positive* (Abdurrohman, Dini, & Muharram, 2018).

$$F1 = 2 \times \frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}}$$

#### 2.2.6.2 Recall

*Recall* adalah adalah metrik evaluasi yang digunakan dalam bidang klasifikasi untuk mengukur sejauh mana model dapat mengidentifikasi semua instance positif yang sebenarnya. *recall* menunjukkan berapa persen data kategori positif yang terklasifikasikan dengan benar oleh *system* (Istighfarizky, et al., 2022), Dalam kata lain, *recall* mengukur kemampuan model untuk menemukan atau "mengingat" semua contoh yang benar dari kelas positif. Persamaan *recall* perbandingan antara *true positive* terhadap total contoh yang benar-benar *positive* (Nasution & Hayaty, 2019):

$$\text{Recall} = \frac{TP}{TP+FN}$$

Dimana :

- a) TP yaitu *True Positive*
- b) FP yaitu *False Positive*
- c) FN yaitu *False Negative*

#### 2.2.6.3 Akurasi

Akurasi menggambarkan seberapa akurat system dapat mengklasifikasikan data secara benar. Dengan kata lain, nilai akurasi merupakan perbandingan antara data yang terklasifikasi benar dengan keseluruhan data (Istighfarizky, et al., 2022)

$$\text{Akurasi} = \frac{TP+TN}{TP+TN+FP+FN}$$

#### 2.2.6.4 Confussion Matrix

*Confusion matrix* adalah alat yang digunakan untuk menganalisis kemampuan model klasifikasi dalam mengenali berbagai *tuple* data yang berbeda. (Nasution & Hayaty, 2019). *Confusion Matrix* adalah tabel dengan 4 kombinasi berbeda dari nilai prediksi dan nilai aktual. Ada empat istilah yang merupakan representasi hasil proses klasifikasi pada

confusion matrix yaitu *True Positif*, *True Negatif*, *False Positif*, dan *False Negatif* (University, 2023).

		Actual Values	
		Positive (1)	Negative (0)
Predicted Values	Positive (1)	TP	FP
	Negative (0)	FN	TN

**Gambar 2. 1 Confusion Matrix**

Dimana :

- a) TP (*True Positive*) Hasil memprediksi positif dan itu benar
- b) TN (*True Negative*) Hasil memprediksi negatif dan itu benar
- c) FP (*False Positive*) Hasil memprediksi positif dan itu salah
- d) FN (*False Negative*) Hasil memprediksi negatif dan itu salah

## **2.2.7 Framework**

### **2.2.7.1 Streamlit**

Streamlit adalah sebuah framework berbasis Python dan bersifat open-source yang dibuat untuk memudahkan dalam membangun aplikasi web di bidang sains data dan machine learning yang interaktif . Salah satu hal menarik dari framework ini adalah kita tidak perlu mengetahui banyak hal tentang teknologi web development. Kita tidak perlu dipusingkan tentang bagaimana mengatur tampilan website dengan CSS, HTML, atau Javascript. Untuk menggunakan Streamlit, kita cukup memiliki modal dasar mengetahui bahasa Python saja(Universitas Islam Indonesia,2022).

## **2.2.8 Aplikasi Pendukung**

### **2.2.8.1 Visual Studio Code**

Visual Studio Code , juga biasa disebut VS Code , [12] adalah editor kode sumber yang dibuat oleh Microsoft dengan Electron Framework , untuk Windows , Linux dan macOS . [13] Fitur-fiturnya mencakup dukungan untuk debugging , penyorotan sintaksis ,

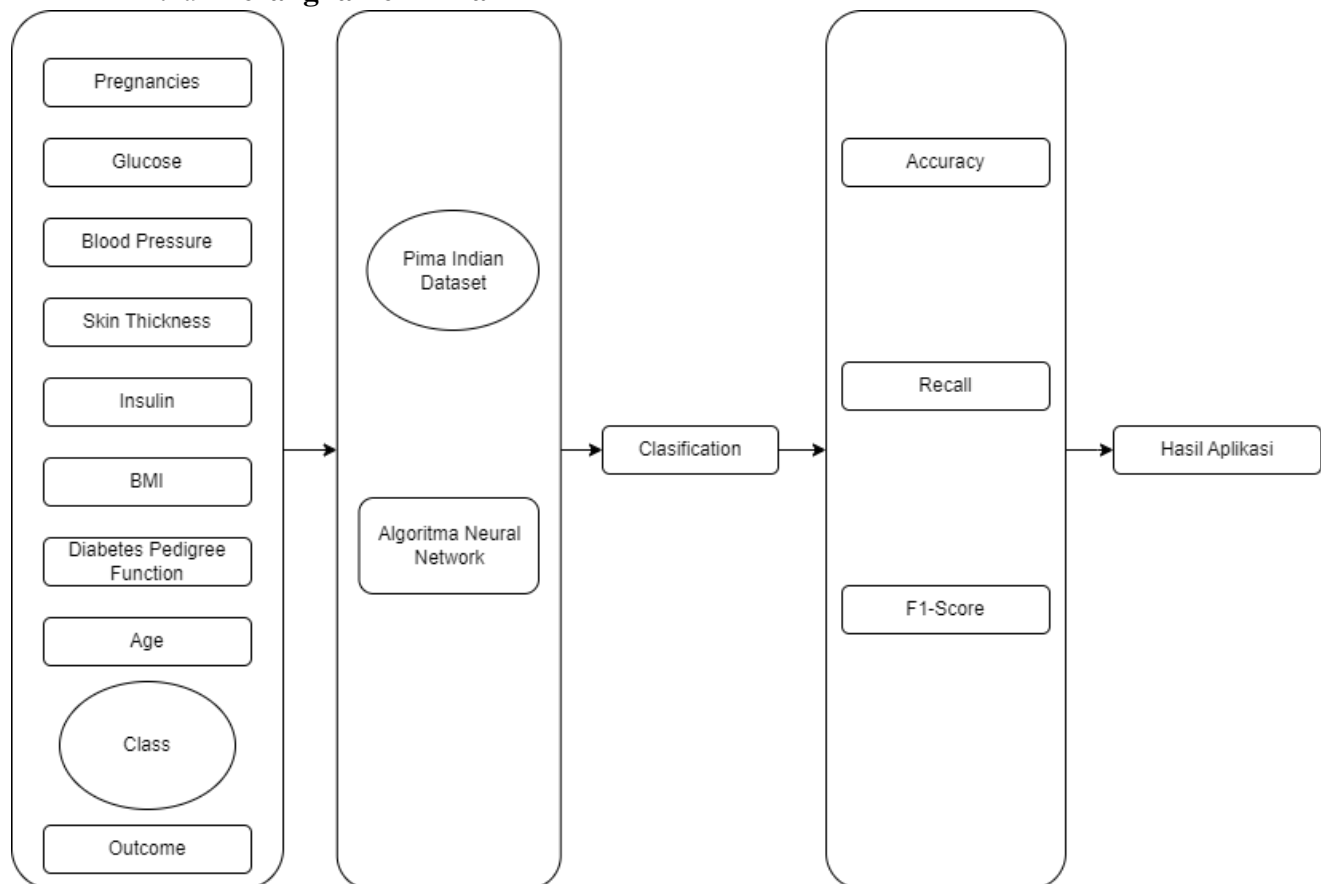
penyelesaian kode cerdas , cuplikan , pemfaktoran ulang kode , dan Git yang tertanam . Pengguna dapat mengubah tema , pintasan keyboard , preferensi, dan pemasangan ekstensi yang menambah fungsionalitas.

Dalam Survei Pengembang Stack Overflow 2023, Visual Studio Code menduduki peringkat alat lingkungan pengembang paling populer di antara 86.544 responden, dengan 73,71% melaporkan bahwa mereka menggunakannya. Hal ini meningkatkan penggunaannya di kalangan mereka yang belajar coding dibandingkan dengan mereka yang sedang mengembangkan profesi(Wikipedia,2023).

#### 2.2.8.2 Jupyter Notebook

*Jupyter Notebook* biasa juga di sebut *jupyter* adalah dokumen yang dapat dibagikan yang menggabungkan kode komputer, bahasa sederhana deskripsi, data, visualisasi kaya seperti model 3D, bagan, grafik, dan angka, dan kontrol interaktif. Buku catatan, bersama dengan editor (seperti *JupyterLab*), menyediakan lingkungan interaktif yang cepat untuk pembuatan prototipe dan menjelaskan kode, mengeksplorasi dan memvisualisasikan data, dan berbagi ide dengan Lain (Collonovall, Dafna, Ivanov, & Eric, 2023).

#### 2.2.9 Kerangka Pemikiran



**Kerangka Pemikiran**

Dalam kerangka pemikiran diatas dijelaskan bahwa penelitian ini ditunjukan untuk mengetahui seberapa tinggi tingkat akurasi *Neural Network* dalam memprediksi orang yang beresiko terkena Diabetes melitus dan menghasilkan nilai akurasi , *F1-score*, dan *Recall* serta pengimplementasian model yang telah dibuat kedalam sebuah aplikasi.

## **2.3 Tinjauan Objek**

### **2.3.1 Diabetes Melitus**

Menurut Kemenkes RI (2020), menjelaskan bahwa diabetes mellitus (DM) adalah penyakit kronis atau menahun berupa gangguan metabolik yang ditandai dengan peningkatan kadar glukosa darah diatas normal. Diabetes mellitus adalah penyakit kronis yang kompleks yang membutuhkan perawatan medis berkelanjutan dengan strategi pengurangan risiko multifaktor di luar kendali glikemik (American Diabetes Association, 2018).

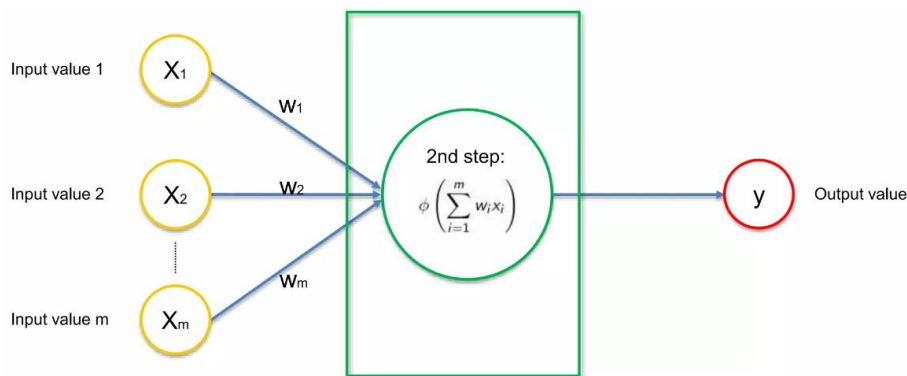
Menurut P2PTM Kemenkes RI (2020), diabetes mellitus merupakan suatu penyakit menahun yang ditandai oleh kadar glukosa darah yang melebihi nilai normal. Dimana nilai normal gula darah sewaktu (GDS) / tanpa puasa adalah  $< 200$  mg/dl sedangkan gula darah puasa (GDP)  $< 126$  mg/dl. Diabetes mellitus disebabkan oleh kekurangan hormon insulin yang dihasilkan oleh pankreas untuk menurunkan kadar gula darah.



## BAB III

### 3.1 Metode

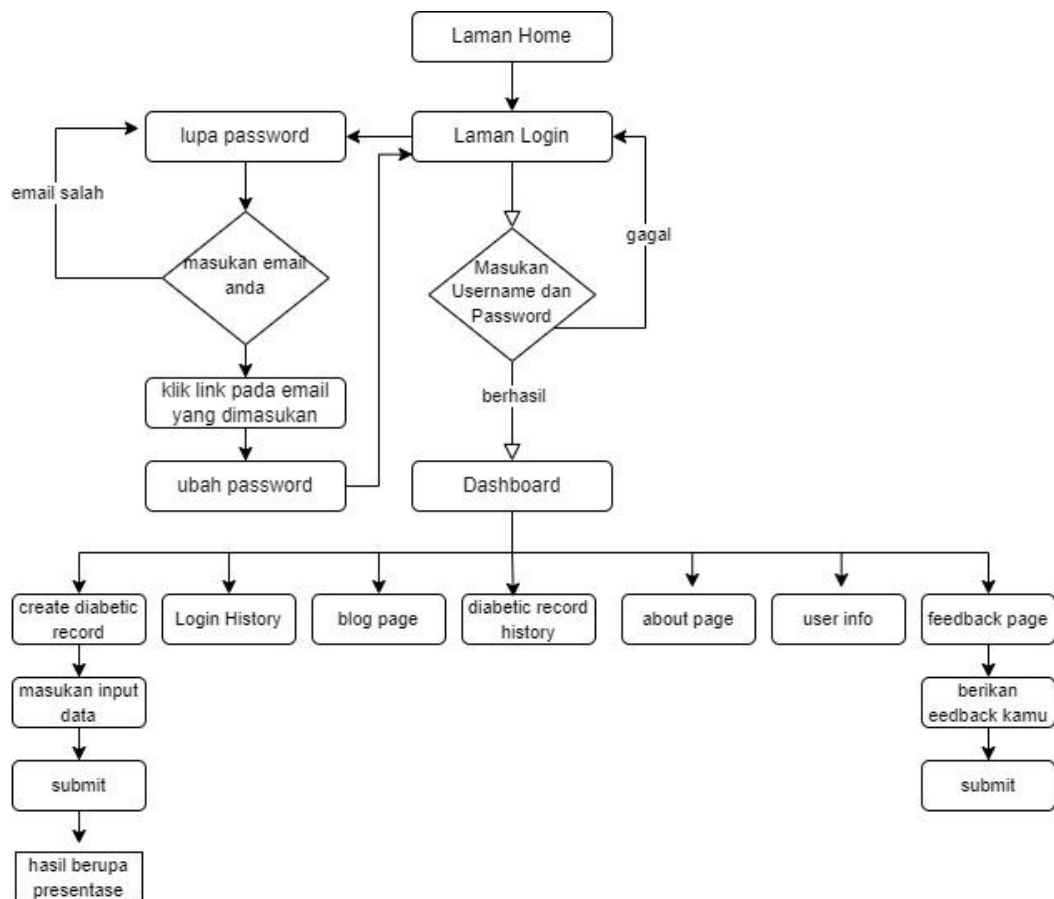
Pada penelitian ini saya menggunakan metode *Artificial Neural Network* Metode neural network prediksi, atau *Artificial Neural Network* (ANN), adalah sebuah model yang meniru jaringan saraf biologis untuk mempelajari pola dan membuat prediksi. Metode ini bekerja dengan cara yang mirip seperti neuron biologis yang saling berinteraksi untuk mengidentifikasi pola, menganalisis opsi, dan sampai pada kesimpulan. ANN sangat berguna dalam menangani masalah yang kompleks dan nonlinear, karena kemampuannya untuk belajar dari data dan menyesuaikan diri dengan perubahan pola. rumus metode *Artificial Neural Network* :



**gambar 3.1 Rumus Neural Network**

### 3.2 Flowchart Sistem

*Flowchart* adalah diagram yang digunakan untuk menggambarkan alur kerja atau proses dalam sebuah aplikasi. Diagram ini menggunakan simbol-simbol grafis untuk merepresentasikan berbagai langkah atau tahapan yang terjadi dalam sistem, serta bagaimana aliran data atau proses berpindah dari satu langkah ke langkah lainnya.



**gambar 3.2 Flowchart Sistem**

### 3.3 Analisa Kebutuhan

Tahap analisis kebutuhan merupakan langkah penting dalam pengembangan sistem yang bertujuan untuk memahami dan mendefinisikan kebutuhan dari sistem yang akan dikembangkan. Analisis ini membantu memastikan bahwa sistem yang dibangun sesuai dengan harapan pengguna dan memenuhi tujuan utama penelitian. Kebutuhan sistem dibagi menjadi dua bagian utama, yaitu kebutuhan fungsional dan non-fungsional.

#### 3.3.1 Kebutuhan Fungsional

Kebutuhan fungsional adalah kebutuhan yang terkait langsung dengan fungsi dan fitur yang harus dimiliki oleh sistem agar dapat berjalan sesuai dengan tujuannya. Berikut adalah beberapa kebutuhan fungsional yang diidentifikasi untuk sistem deteksi diabetes menggunakan machine learning:

- a) Penerimaan Input Data:

Sistem harus dapat menerima input data dari dataset yang telah ditentukan. Dataset ini akan digunakan sebagai basis untuk pelatihan dan pengujian model prediksi. Data yang diterima harus mencakup semua variabel yang relevan seperti usia, jenis kelamin, BMI, kadar glukosa, tekanan darah, insulin, dan faktor risiko lainnya.

b) Pengolahan Data Awal:

Sistem harus mampu melakukan pengolahan data awal yang mencakup pembersihan data, normalisasi data, dan pembagian dataset menjadi data pelatihan dan data pengujian. Pengolahan data awal ini penting untuk memastikan bahwa data yang digunakan dalam pelatihan model adalah data yang berkualitas dan representatif.

c) Pembangunan Model Prediksi:

Sistem harus mampu membangun model prediksi diabetes menggunakan algoritma Neural Network. Model ini akan dilatih menggunakan data pelatihan yang telah diproses sebelumnya dan harus mampu mengenali pola-pola dalam data yang menunjukkan adanya risiko diabetes.

d) Pengujian dan Evaluasi Model:

Sistem harus dapat menguji dan mengevaluasi model yang telah dibuat. Pengujian dilakukan dengan menggunakan data pengujian untuk mengukur kinerja model dalam memprediksi diabetes. Evaluasi kinerja model dilakukan dengan menggunakan metrik evaluasi seperti akurasi, presisi, recall, dan F1-score.

e) Antarmuka Pengguna (UI):

Sistem harus memiliki antarmuka pengguna (UI) yang memudahkan interaksi pengguna. Antarmuka ini harus intuitif dan user-friendly sehingga pengguna,

termasuk mereka yang tidak memiliki latar belakang teknis, dapat dengan mudah mengoperasikan sistem dan memahami hasil prediksi yang diberikan

f) **Pengelolaan Koneksi Frontend dan Backend:**

Sistem harus dapat mengelola koneksi antara frontend dan backend dengan baik. Frontend, yang dibangun menggunakan React.js, harus dapat berkomunikasi dengan backend yang menggunakan Express.js. Backend ini kemudian akan berinteraksi dengan server machine learning yang menggunakan Flask untuk melakukan prediksi.

### **3.3.2 Analisa Kebutuhan Non-Fungsional**

Kebutuhan non-fungsional adalah kebutuhan yang berkaitan dengan kualitas sistem dan bagaimana sistem tersebut beroperasi. Berikut adalah beberapa kebutuhan non-fungsional yang diidentifikasi untuk sistem ini:

a) **Waktu Respons yang Cepat:**

Sistem harus memiliki waktu respons yang cepat dalam memproses data. Waktu respons yang cepat penting agar pengguna tidak mengalami keterlambatan dalam mendapatkan hasil prediksi.

b) **Kemudahan Penggunaan:**

Sistem harus mudah digunakan dan dioperasikan oleh pengguna awam. Ini berarti bahwa antarmuka pengguna harus dirancang sedemikian rupa sehingga tidak memerlukan pelatihan khusus untuk menggunakannya. Instruksi yang jelas dan navigasi yang mudah akan membantu pengguna dalam berinteraksi dengan sistem.

Software	Spesifikasi
Sistem Operasi	Windows 10 atau setara
Aplikasi Simulator	Pemrograman dan Browser

Library Python	Pandas, Pycaret, Scikit-learn, Numpy, Tensorflow, Matplotlib, Seaborn, Keras, Tailwindcss
Library Javascript	Cors, Axios, Nodemailer, JSONwebtoken, Nodemon, Crypto, Bcrypt, MySQL2,
Framework Aplikasi	Vite React Js, Node Js, Flask, Express Js
Database	MySQL

### 3.3.3 Pengumpulan Data(Dataset)

Pada penelitian Dataset yang digunakan adalah dataset yang tersedia secara public di situs Kaggle.com dan telah banyak digunakan oleh peneliti untuk mengevaluasi kinerja deteksi kemungkinan diabetes , yang memiliki atribut kolom sebagai berikut :

No	Nama Atribut	Keterangan
1	Pregnancies	Jumlah kehamilan yang pernah dialami oleh pasien.
2	Glucose	Kadar glukosa (gula) dalam darah pasien, yang diukur dalam miligram per desiliter (mg/dL).
3	BloodPressure	Tekanan darah sistolik pasien (tekanan saat jantung berkontraksi) dalam milimeter raksa (mm Hg).
4	SkinThickness	Ketebalan lipatan kulit pada trisep (lengan atas) pasien dalam milimeter.
5	Insulin	Kadar insulin dalam darah pasien.

6	BMI	Indeks massa tubuh pasien,. Ini adalah indikator umum untuk mengukur apakah seseorang memiliki berat badan yang sehat atau tidak.
7	DiabetesPedigreeFunction	Fungsi yang menggambarkan seberapa besar riwayat diabetes ada dalam keluarga pasien. Ini sering digunakan untuk memperkirakan risiko genetik diabetes.
8	Age	Usia pasien dalam satuan tahun.
9	Outcome	Hasil dari tes diabetes , yang mana hasilnya adalah angka 0 dan 1 jika angkanya 0 maka pasien tersebut bebas diabetes dan sebaliknya.

### 3.3.4 Pengolahan Data Awal

Dalam dataset ini mencakup sejumlah fitur atau atribut yang digunakan untuk memprediksi apakah seseorang mengidap diabetes (Outcome). Dataset ini digunakan untuk mengembangkan model prediksi diabetes berdasarkan fitur-fitur ini.

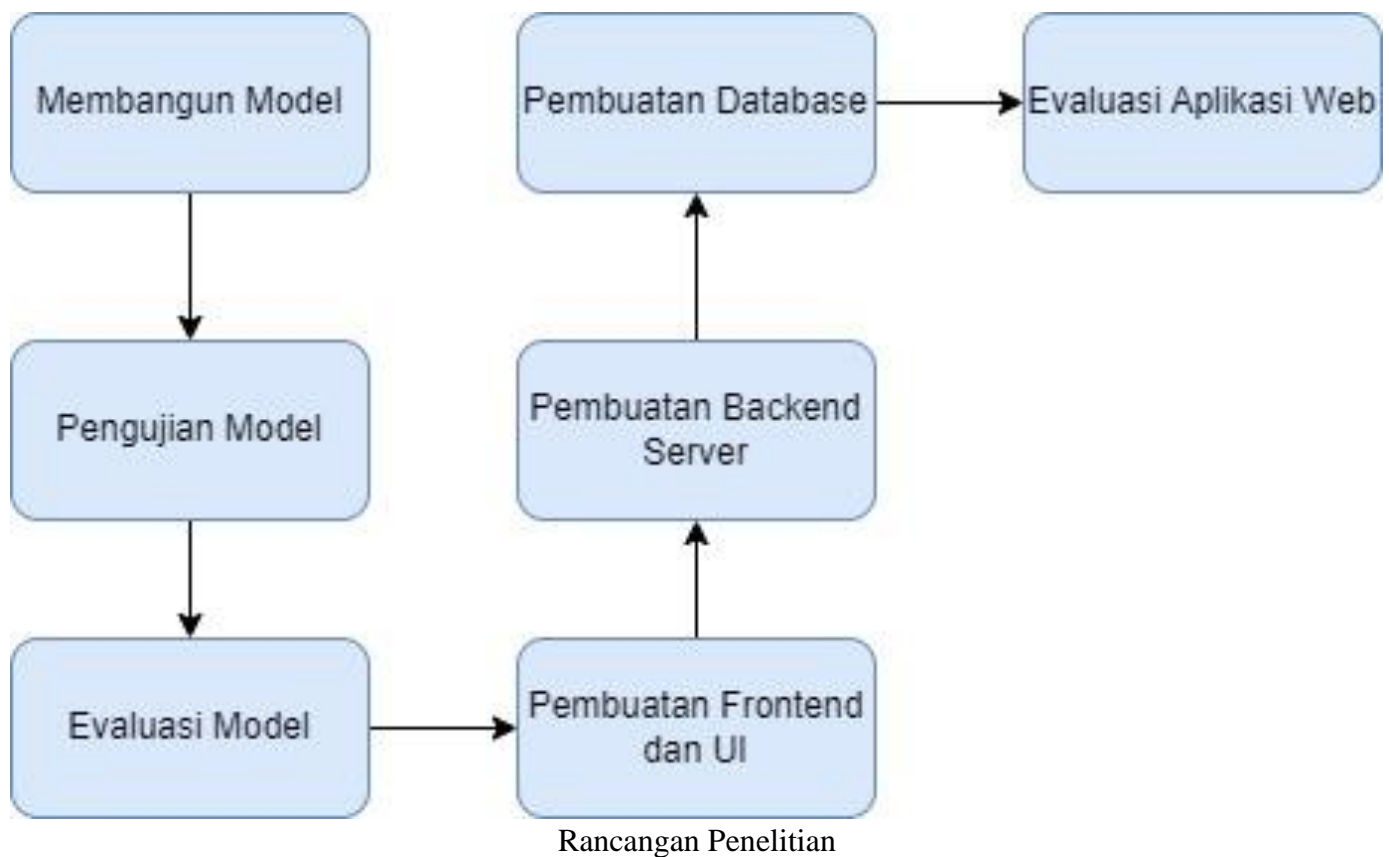
Yang kemudian, data ini akan dibagi menjadi dua bagian: satu untuk melatih model (training data) dan yang lainnya untuk menguji model (testing data). Tujuan akhirnya adalah untuk membangun model yang dapat memprediksi apakah seseorang memiliki risiko diabetes berdasarkan atribut-atribut ini.

No.	Nama Atribut	Tipe Data
1	Pregnancies	Int
2	Glucose	Int
3	BloodPressure	Int
4	SkinThickness	Int
5	Insulin	Int
6	BMI	Float
7	DiabetesPedigreeFunction	Float
8	Age	Int
9	Outcome	Int

### 3.4 Pearancangan Penelitian

Pada penelitian ini menggunakan metode Neural Network dalam mendeteksi siapa yang lebih beresiko terkena diabetes, yang dimana data yang digunakan merupakan dataset yang tersedia secara public di situs web (Kaggle .com)

Tahapan tahapan yang perlu dilakukan dalam penelitian ini adalah : karena data yang diambil sudah bersih dan semua datanya sudah sesuai kita bisa langsung Membangun model, lalu pengujian model dan terakhir evaluasi & validasi hasil Pembuatan Aplikasi Web dan Evaluasi Aplikasi seperti gambar berikut ini :





Berdasarkan gambar diatas , dapat dijelaskan bahwa tahapan dalam penelitian ini adalah sebagai berikut :

**a. Membangun Model**

Selanjutnya yaitu membangun model dengan algoritma Neural Network. Ini diimplementasikan dengan menggunakan library scikit-learn, Keras, Tensorflow, Seaborn, Matplotlib. Scikit-learn atau Sklearn adalah library berbasis Python untuk membangun model pembelajaran mesin.

**b. Pengujian Model**

Pengujian model merupakan langkah penting dalam mengevaluasi performa dan efektivitas model yang telah dibangun dalam pembelajaran mesin atau statistik. Tujuan dari pengujian model adalah untuk memberikan pemahaman yang jelas mengenai kemampuan model dalam melakukan prediksi atau klasifikasi data yang belum pernah dilihat sebelumnya dengan akurasi dan keandalan yang tinggi. Dalam pengujian model, terdapat beragam metrik dan teknik yang digunakan, termasuk:

1. Akurasi (*Accuracy*): Metrik ini mengukur sejauh mana model mampu mengklasifikasikan data dengan benar. Akurasi dihitung dengan membandingkan prediksi model dengan nilai sebenarnya dari data pengujian.
2. Recall (*Recall*): Juga dikenal sebagai sensitivitas, metrik ini menunjukkan sejauh mana model dapat mengidentifikasi data positif dengan baik. Ini mengukur proporsi data positif yang berhasil diklasifikasikan dengan benar dari keseluruhan data positif yang ada.
3. F1-Skor (*F1-Score*): F1-Skor merupakan ukuran yang menggabungkan presisi (*precision*) dan Recall. Metrik ini memberikan keseimbangan antara presisi dan Recall, dan seringkali digunakan ketika terdapat ketidakseimbangan antara kelas-kelas dalam dataset.

4. Matriks Confusion (*Confusion Matrix*): Matriks ini digunakan untuk menggambarkan klasifikasi yang benar dan salah yang dilakukan oleh model. Matriks ini terdiri dari empat istilah: True Positive (*TP*), True Negative (*TN*), False Positive (*FP*), dan False Negative (*FN*).

Dengan menggunakan berbagai metrik ini, pengujian model memungkinkan para peneliti dan praktisi untuk mendapatkan pemahaman yang lebih dalam tentang seberapa baik model dapat digunakan untuk memecahkan masalah yang ada.

#### c. Evaluasi & Validasi Hasil

Evaluasi hasil dalam penelitian ini adalah proses pengujian model machine learning untuk deteksi diabetes yang telah dibuat dengan menggunakan pola data baru yaitu data uji 154 baris data 8 kolom. Dimana pada saat pembuatan proses model machine learning menggunakan data latih sebanyak 614 baris data 8 kolom.

#### Evaluasi & validasi hasil

<i>Dataset</i>	Jumlah
Data Latih	614 baris data 8 kolom
Data Uji	154 baris data 8 kolom

#### d. Pembuatan Database

Tahap ini melibatkan pembuatan dan konfigurasi database yang akan digunakan untuk menyimpan data pengguna, hasil prediksi, dan informasi lainnya yang relevan. MySQL digunakan sebagai sistem manajemen basis data.

#### e. Pembuatan Frontend UI

Frontend dikembangkan menggunakan React.js untuk menyediakan antarmuka pengguna yang interaktif dan responsif. Antarmuka ini memungkinkan pengguna untuk memasukkan data medis, melihat hasil prediksi, dan mengakses fitur-fitur lainnya seperti riwayat login, feedback, dan resep sehat.

## Halaman Utama:

Halaman utama berisi Menu Registrasi dan Login yang mana jika user ingin menggunakan Aplikasi ini Harus memiliki akun terlebih dahulu kemudian melakukan Login untuk masuk ke Halaman Dashboard Page yang bersisi Menu yang lainnya antara lain:

### Halaman Login

Halaman Login adalah halaman awal yang akan diakses oleh pengguna sebelum mereka dapat mengakses fitur-fitur lain dalam aplikasi web. Halaman ini dirancang untuk memastikan bahwa hanya pengguna yang sah yang dapat masuk dan menggunakan sistem.

### Halaman Dashboard

Halaman Dashboard adalah pusat kontrol utama dalam aplikasi web di mana pengguna dapat mengakses berbagai fitur dan fungsi yang disediakan oleh sistem. Halaman ini dirancang untuk memberikan pengguna akses yang cepat dan mudah ke berbagai menu utama dan informasi penting.

### Halaman Add Record atau Record Page

Halaman Record Prediksi adalah halaman yang dirancang untuk memungkinkan pengguna memasukkan data medis, melakukan prediksi diabetes, dan menyimpan hasil prediksi ke dalam database berdasarkan user ID. Halaman ini memainkan peran penting dalam pengumpulan data dan analisis prediksi. Dan Field yang harus diisi antara lain :

- Jumlah Kehamilan
- Glukosa Darah
- Tekanan Darah
- Ketebalan Kulit
- Insulin
- Indeks Massa Tubuh (BMI)
- Fungsi Pewarisan Diabetes
- Usia

Tombol "Prediksi" untuk memulai proses prediksi yang dilakukan oleh model yang telah dibuat yang nanti akan menampilkan hasil pada bagian hasil prediksi.

### Halaman History Record

Halaman History Record adalah halaman yang memberikan pengguna akses untuk melihat hasil rekaman prediksi diabetes yang pernah mereka lakukan sebelumnya. Halaman ini dirancang untuk membantu pengguna melacak riwayat prediksi dan memahami perkembangan kondisi kesehatan mereka berdasarkan data yang telah dimasukkan dan juga berisi tanggal dan waktu serta nama orang yang diprediksi agar si pengguna mengetahui dataprediksi milik siapa dan kapan ia pernah melakukan prediksi ini.

### Halaman History Login

Halaman History Login adalah halaman yang memberikan pengguna akses untuk melihat riwayat login mereka. Halaman ini dirancang untuk meningkatkan keamanan akun pengguna dengan memungkinkan mereka memonitor aktivitas login. Pengguna dapat melihat waktu dan tanggal setiap kali mereka login ke sistem. Fitur ini membantu pengguna mendeteksi aktivitas yang mencurigakan dan mengambil tindakan cepat, seperti mengubah password, jika mereka menduga akun mereka telah diretas. Jika pengguna melihat aktivitas login yang tidak dikenali, mereka dapat dengan mudah mengakses link "Lupa Password" pada halaman login untuk mengganti kata sandi mereka dan mengamankan akun.

### Halaman Feedback

Halaman Feedback adalah halaman yang memungkinkan pengguna untuk memberikan penilaian dan berbagi pengalaman mereka setelah menggunakan aplikasi web ini. Halaman ini berfungsi sebagai sarana komunikasi antara pengguna dan pengembang, di mana pengguna dapat memberikan saran untuk peningkatan aplikasi sesuai dengan kebutuhan mereka. Selain itu, halaman ini juga berfungsi sebagai platform untuk melaporkan error atau bug yang ditemukan selama penggunaan aplikasi. Dengan feedback yang konstruktif, pengembang dapat terus memperbaiki dan mengembangkan aplikasi untuk memberikan layanan yang lebih baik kepada pengguna.

### Halaman Healthy Recipe

Halaman Healthy Recipe adalah halaman di mana pengguna dapat menemukan berbagai resep masakan sehat yang dirancang untuk membantu mereka menghindari diabetes. Resep-resep yang ditampilkan fokus pada makanan yang rendah gula dan kaya nutrisi. Setiap kali pengguna mengklik menu resep yang tersedia, mereka akan diarahkan ke halaman resep yang diambil dari API data web masakan sehat seperti <https://spoonacular.com/food-api>. Halaman ini memberikan variasi resep yang diperbarui

secara berkala, memastikan pengguna selalu memiliki pilihan baru dan menarik untuk dicoba. Dengan menyediakan informasi tentang makanan sehat, halaman ini membantu pengguna dalam membuat keputusan yang lebih baik terkait pola makan mereka.

#### Halaman About

Halaman About adalah halaman yang memberikan informasi detail tentang aplikasi web ini, tujuan pengembangannya, dan tim yang bekerja di baliknya. Halaman ini bertujuan untuk memberikan transparansi kepada pengguna mengenai misi dan visi aplikasi, serta teknologi yang digunakan dalam pembuatannya.

#### **f. Pembuatan Server Backend**

Backend server dikembangkan menggunakan Express.js untuk menangani permintaan dari frontend dan berinteraksi dengan database. Backend ini juga bertanggung jawab untuk mengirimkan data medis ke server machine learning dan menerima data yang akan prediksi menggunakan Flask framework untuk memuat model dan melakukan perhitungan .

#### **g. Evaluasi Aplikasi Web**

Setelah semua komponen sistem dikembangkan, dilakukan evaluasi keseluruhan aplikasi web. Evaluasi ini mencakup:

- Pengujian Fungsionalitas: Memastikan semua fitur aplikasi berjalan sesuai dengan yang diharapkan.
- Pengujian Kinerja: Menilai responsivitas dan kecepatan aplikasi dalam memproses data.
- Uji Pengguna: Melibatkan pengguna untuk mendapatkan feedback tentang pengalaman mereka dalam menggunakan aplikasi.

## BAB IV

### IMPLEMENTASI DAN HASIL PENGUJIAN

#### 4.1 Hasil

Tujuan dari penelitian ini adalah untuk menerapkan Algoritma *Neural Network* untuk mendeteksi Diabetes Melitus. Yang dilakukan menggunakan *PIMA Indian Dataset*.

##### 4.1.1 Persiapan Data

Pada penelitian ini memakai *PIMA Indian Dataset* yang tersedia secara publik pada Website [kaggle.com](https://www.kaggle.com). Gambaran dari Datasetnya seperti pada tabel dibawah ini :

**Tabel 4. 1 Persiapan Data**

No	Nama	Record	Feature	Class
1	<i>PIMA Indian Dataset</i>	768	8	1

Dataset ini telah banyak digunakan oleh peneliti sebelumnya yaitu (Md Kamrul Hasan, Md Ashraful Alam, Dola Das, Eklas Hossain, Mahmudul Hasan, 2020)

#### 4.2 Pembahasan

##### 4.2.1 Pengolahan Data

Pada tahap ini dilakukan pengolahan data dengan cara melihat apakah ada data non-numerik jika ada nantinya akan diubah menjadi format yang dapat dipahami oleh model atau algoritma *Neural Network* biasanya diubah menjadi numerik.

##### 4.2.2 Eksplorasi Data

Pada tahap ini akan dilakukan pengecekan data yaitu berupa pencarian isi, jumlah dan tipe data pada setiap atribut, mencari apa terdapat nilai kosong dan mencari data yang duplikat

##### 4.2.2.1 Analisa Data

Pada fase penelitian ini dilakukan pembersihan data yang terdapat dalam kumpulan data untuk memastikan kualitas dan kevalidan data yang akan digunakan dengan membersihkan data yang duplikat, outlier, dan missing value.

No.	Nama Atribut	Tipe Data
1	Pregnancies	Int
2	Glucose	Int
3	BloodPressure	Int
4	SkinThickness	Int
5	Insulin	Int
6	BMI	Float
7	DiabetesPedigreeFunction	Float
8	Age	Int
9	Outcome	Int

Pada dataset ini, semua kolom sudah bertipe integer dan float sehingga dapat langsung dilakukan pemrosesan.

Kemudian dilakukan pengecekan terhadap sampel data yaitu *Class* atau label.

a. Class atau Label

*Class* atau label pada dataset ini adalah *class* yang akan berfungsi sebagai output dari model mesin *learning* yang dibuat, hasil analisa pada kolom *class* dapat dilihat pada tabel dibawah ini.

Keterangan	Nilai
Jumlah Baris	768 Baris Data
Keunikan setiap data	2 Baris Data, 1 = <i>orang tersebut terkena diabetes</i> 0 = <i>orang tersebut bebas diabetes</i>
Frekuensi kemunculan	500
Data yang sering muncul	0

b. *Feature* atau Atribut

Pengecekan data dari 8 sampel atribut atau feature (*Pregnancies, Glucose Blood Pressure, Skin Thickness, Insulin, BMI, Diabetes Pedigree Function, Age*).

1. *Pregnancies*

Pada kolom *Pregnancies* terdapat 768 baris data yang dimana data tersebut memiliki nilai rata-rata 3.845052 dengan nilai standar deviasi sebesar 3.369578 dengan nilai minimal 0 dan nilai maksimal 17.

Keterangan	Nilai
Jumlah Baris	768 Baris Data
Rata-rata	3.845052
Standar Deviasi	3.369578

2. *Glucose*

Pada kolom *Pregnancies* terdapat 768 baris data yang dimana data tersebut memiliki nilai rata-rata 120.894531 dengan nilai standar deviasi sebesar 31.972618 dengan nilai minimal 0 dan nilai maksimal 199.

Keterangan	Nilai
Jumlah Baris	768 Baris Data
Rata-rata	120.894531
Standar Deviasi	31.972618

3. *Blood Preasure*

Pada kolom *Pregnancies* terdapat 768 baris data yang dimana data tersebut memiliki nilai rata-rata 69.105469 dengan nilai standar deviasi sebesar 19.355807 dengan nilai minimal 0 dan nilai maksimal 122.

Keterangan	Nilai
Jumlah Baris	768 Baris Data
Rata-rata	69.105469
Standar Deviasi	19.355807



#### 4. *Skin Thickness*

Pada kolom *Pregnancies* terdapat 768 baris data yang dimana data tersebut memiliki nilai rata-rata 20.536458 dengan nilai standar deviasi sebesar 15.952218 dengan nilai minimal 0 dan nilai maksimal 99.

Keterangan	Nilai
Jumlah Baris	768 Baris Data
Rata-rata	20.536458
Standar Deviasi	15.952218

#### 5. *Insulin*

Pada kolom *Pregnancies* terdapat 768 baris data yang dimana data tersebut memiliki nilai rata-rata 79.799479 dengan nilai standar deviasi sebesar 115.244002 dengan nilai minimal 0 dan nilai maksimal 846.

Keterangan	Nilai
Jumlah Baris	768 Baris Data
Rata-rata	79.799479
Standar Deviasi	115.244002

#### 6. *BMI*

Pada kolom *Pregnancies* terdapat 768 baris data yang dimana data tersebut memiliki nilai rata-rata 31.992578 dengan nilai standar deviasi sebesar 7.884160 dengan nilai minimal 0 dan nilai maksimal 67.1.

Keterangan	Nilai
Jumlah Baris	768 Baris Data
Rata-rata	31.992578
Standar Deviasi	7.884160

#### 7. *Diabetes Pedigree Function*

Pada kolom *Pregnancies* terdapat 768 baris data yang dimana data tersebut memiliki nilai rata-rata 0.471876 dengan nilai standar deviasi sebesar 0.331329 dengan nilai minimal 0 dan nilai maksimal 2.420.

Keterangan	Nilai
Jumlah Baris	768 Baris Data
Rata-rata	0.471876
Standar Deviasi	0.331329

#### 8. Age

Pada kolom *Pregnancies* terdapat 768 baris data yang dimana data tersebut memiliki nilai rata-rata 33.240885 dengan nilai standar deviasi sebesar 11.760232 dengan nilai minimal 0 dan nilai maksimal 81.

Keterangan	Nilai
Jumlah Baris	768 Baris Data
Rata-rata	33.240885
Standar Deviasi	11.760232

#### 4.2.2.2 Statistik Data

Pada proses ini dilakukan proses penggalian informasi terkait nilai-nilai statistik pada dataset, khususnya pada kolom-kolom yg bertipe numerik.

	Pregna ncies	Glucos e	BloodPr essure	SkinThi ckness	Insulin	BMI	DiabetesPedigr eeFunction	Age	Outco me
co un t	768.00 0000	768.00 0000	768.000 000	768.00 0000	768.00 0000	768.00 0000	768.000000	768.00 0000	768.00 0000
me an	3.8450 52	120.89 4531	69.1054 69	20.536 458	79.799 479	31.992 578	0.471876	33.240 885	0.3489 58
std	3.3695 78	31.972 618	19.3558 07	15.952 218	115.24 4002	7.8841 60	0.331329	11.760 232	0.4769 51
mi n	0.0000 00	0.0000 00	0.00000 0	0.0000 00	0.0000 00	0.0000 00	0.078000	21.000 000	0.0000 00
25 %	1.0000 00	99.000 000	62.0000 00	0.0000 00	0.0000 00	27.300 000	0.243750	24.000 000	0.0000 00

50 %	3.0000 00	117.00 0000	72.0000 00	23.000 000	30.500 000	32.000 000	0.372500	29.000 000	0.0000 00
75 %	6.0000 00	140.25 0000	80.0000 00	32.000 000	127.25 0000	36.600 000	0.626250	41.000 000	1.0000 00
ma x	17.000 000	199.00 0000	122.000 000	99.000 000	846.00 0000	67.100 000	2.420000	81.000 000	1.0000 00

#### 4.2.2.3 Visualisasi Data

Pada tahap ini akan dilakukan eksplorasi data menggunakan grafik atau diagram untuk memberikan gambaran terhadap sebuah data pada tiap kolom.

##### a. *Class* atau label

Dari total 768 data *Class*, terdapat sebanyak 5540 (63%) data yang bukan merupakan malware dan sebanyak 1719 (37%) data yang merupakan *malware*.

#### 4.2.3 Membangun Model

Membangun model merupakan langkah krusial dalam analisis data, di mana kita menciptakan suatu representasi matematis yang dapat menggambarkan hubungan antara variabel dalam *Dataset*. Pada penelitian ini saya menggunakan metode Neural Network untuk membangun model *machine learning* ini.

Dataset	Jumlah
<i>PIMA Indian Dataset</i>	768 baris dan 9 kolom

##### a. Split Data

Pada tahap ini, data akan dibagi menjadi data latih dan data uji menggunakan `train_test_split`. Adapun besaran dari data latih yaitu 80% dan data uji yaitu 20%. Seperti pada tabel berikut:

Dataset	Jumlah
<i>Data Train</i>	614 baris dan 8 kolom
<i>Data Test</i>	154 baris dan 8 kolom

b. Training Model

Model yang digunakan yaitu klasifikasi, yang bertujuan untuk mendeteksi penderita diabetes melitus berdasarkan berbagai fitur atau atribut yang ada dalam dataset.

Dataset	Nilai
<i>ephocs</i>	50
<i>Laerning Rate</i>	0.002
<i>Layer</i>	1 layer input dengan 8 unit neuron 1 hidden layer dengan 30 unit neuron 1 hidden layer dengan 10 unit neuron 1 output layer dengan 1 neuron

Berdasarkan gambar dibawah ini, terdapat 8 neuron pada layer input, 30 neuron dan 10 neuron pada 2 hidden layer, dan 2 neuron pada layer output. Cara kerja ketiga layer tersebut adalah input layer menerima masukan nilai awal dari data yang akan diproses, nantinya dilewatkan ke hidden layer lalu akan keluar melalui output layer dalam bentuk bobot pula.

#### 4.2.4 Pengujian Model

Pada tahap ini dilakukan pengujian performa model dengan menggunakan metrik evaluasi yang digunakan dalam metode *Neural Network* (NN). Metrik evaluasi ini akan memberikan pemahaman tentang sejauh mana model dapat mendeteksi *malware* pada sistem andorid dengan akurat. Beberapa metrik evaluasi yang digunakan dalam pengujian model ini meliputi akurasi, *recall*, *F1-Score*, dan *Confusion matrix*.

Hasil dari model ANN

Model	Hasil
-------	-------

Akurasi	76.3%
<i>Recall</i>	76.1
<i>F1-Score</i>	76.1

Setelah kedua model tersebut dievaluasi dan dilakukan validasi hasil kemudian kita lakukan penyimpanan kedua model tersebut kedalam format H5. Format file H5 (HDF5) adalah format penyimpanan yang efisien dan dapat digunakan untuk menyimpan berbagai jenis data, termasuk model neural network. Pustaka seperti Keras dapat menyimpan model mereka dalam format H5, yang kemudian dapat diimpor kembali untuk digunakan atau disaring lebih lanjut.

### 4.3 Implementasi Sistem

Implementasi sistem adalah proses menerapkan dan menjalankan sistem baru atau yang telah diperbarui di lingkungan nyata. Tujuan utamanya adalah memastikan bahwa sistem tersebut berfungsi dengan baik dan memenuhi kebutuhan pengguna. Proses ini mencakup pemasangan perangkat lunak dan perangkat keras, pengujian untuk memastikan semuanya berjalan lancar, pelatihan pengguna agar mereka dapat menggunakan sistem dengan efektif, dan pemeliharaan serta dukungan berkelanjutan untuk memastikan sistem tetap beroperasi dengan baik. Berikut ini adalah Struktur Tabel yang telah dibangun :

#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra	Tindakan
<input type="checkbox"/> 1	<b>id</b> 🔑🔑	varchar(5)	utf8mb4_general_ci	Tidak	Tidak	Tidak ada			✎ Ubah ➡ Hapus Lainnya
<input type="checkbox"/> 2	<b>username</b> 🔑	varchar(50)	utf8mb4_general_ci	Tidak	Tidak	Tidak ada			✎ Ubah ➡ Hapus Lainnya
<input type="checkbox"/> 3	<b>email</b>	varchar(33)	utf8mb4_general_ci	Tidak	Tidak	Tidak ada			✎ Ubah ➡ Hapus Lainnya
<input type="checkbox"/> 4	<b>password</b>	varchar(70)	utf8mb4_general_ci	Tidak	Tidak	Tidak ada			✎ Ubah ➡ Hapus Lainnya
<input type="checkbox"/> 5	<b>retype_password</b>	varchar(30)	utf8mb4_general_ci	Tidak	Tidak	Tidak ada			✎ Ubah ➡ Hapus Lainnya
<input type="checkbox"/> 6	<b>created_time</b>	datetime		Tidak	current_timestamp()				✎ Ubah ➡ Hapus Lainnya
<input type="checkbox"/> 7	<b>reset_password_token</b>	varchar(255)	utf8mb4_general_ci	Ya	NULL				✎ Ubah ➡ Hapus Lainnya
<input type="checkbox"/> 8	<b>reset_password_expires</b>	bigint(20)		Ya	NULL				✎ Ubah ➡ Hapus Lainnya

**Gambar 4.3.1 Tabel User**

#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra	Tindakan
<input type="checkbox"/>	1 id_record 🗝️	int(8)			Tidak	Tidak ada		AUTO_INCREMENT	Ubah Hapus Lainnya
<input type="checkbox"/>	2 user_id 🗝️	varchar(5)	utf8mb4_general_ci		Tidak	Tidak ada			Ubah Hapus Lainnya
<input type="checkbox"/>	3 name	varchar(50)	utf8mb4_general_ci		Tidak	Tidak ada			Ubah Hapus Lainnya
<input type="checkbox"/>	4 record_time	datetime			Tidak	current_timestamp()			Ubah Hapus Lainnya
<input type="checkbox"/>	5 pregnancies	int(3)			Tidak	Tidak ada			Ubah Hapus Lainnya
<input type="checkbox"/>	6 glucose	int(4)			Tidak	Tidak ada			Ubah Hapus Lainnya
<input type="checkbox"/>	7 blood_preasure	int(3)			Tidak	Tidak ada			Ubah Hapus Lainnya
<input type="checkbox"/>	8 skin_thickness	int(3)			Tidak	Tidak ada			Ubah Hapus Lainnya
<input type="checkbox"/>	9 insulin	int(4)			Tidak	Tidak ada			Ubah Hapus Lainnya
<input type="checkbox"/>	10 bmi	float			Tidak	Tidak ada			Ubah Hapus Lainnya
<input type="checkbox"/>	11 diabetes_pedigree_function	float			Tidak	Tidak ada			Ubah Hapus Lainnya
<input type="checkbox"/>	12 age	int(3)			Tidak	Tidak ada			Ubah Hapus Lainnya
<input type="checkbox"/>	13 outcome	text	utf8mb4_general_ci		Tidak	Tidak ada			Ubah Hapus Lainnya

**Gambar 4.3.2 Tabel Record**

#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra	Tindakan
<input type="checkbox"/>	1 id 🗝️	int(11)			Tidak	Tidak ada		AUTO_INCREMENT	Ubah Hapus Lainnya
<input type="checkbox"/>	2 username 🗝️	varchar(50)	utf8mb4_general_ci		Tidak	Tidak ada			Ubah Hapus Lainnya
<input type="checkbox"/>	3 login_time	timestamp			Tidak	current_timestamp()			Ubah Hapus Lainnya
<input type="checkbox"/>	4 logout_time	timestamp		Ya		NULL			Ubah Hapus Lainnya

**Gambar 4.3.3 Tabel User\_session**

#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra	Tindakan
<input type="checkbox"/>	1 id 🗝️	int(5)			Tidak	Tidak ada		AUTO_INCREMENT	Ubah Hapus Lainnya
<input type="checkbox"/>	2 username 🗝️	varchar(80)	utf8mb4_general_ci		Tidak	Tidak ada			Ubah Hapus Lainnya
<input type="checkbox"/>	3 content	text	utf8mb4_general_ci		Tidak	Tidak ada			Ubah Hapus Lainnya

**Gambar 4.3.1 Tabel Feedback**

## 4.4 Membangun Web App

Pada tahap ini penulis akan mengimplementasikan model yang telah dibuat ke dalam sebuah web app sederhana yang dapat memprediksi seseorang terkena diabetes melitus dengan memasukkan beberapa nilai pada variabel seperti *Pregnancies*, *BMI*, *Bloodpressure*, *Skin thickness*, *DiabetesPedigreeFunction*, *Glucose*, dan *Insulin*. dan user dapat langsung mengetahui hasilnya berupa presentase dari hasil kalkulasi yang dilakukan oleh model Neural Network..

### 4.4.1 Desain User Interface (UI/UX)

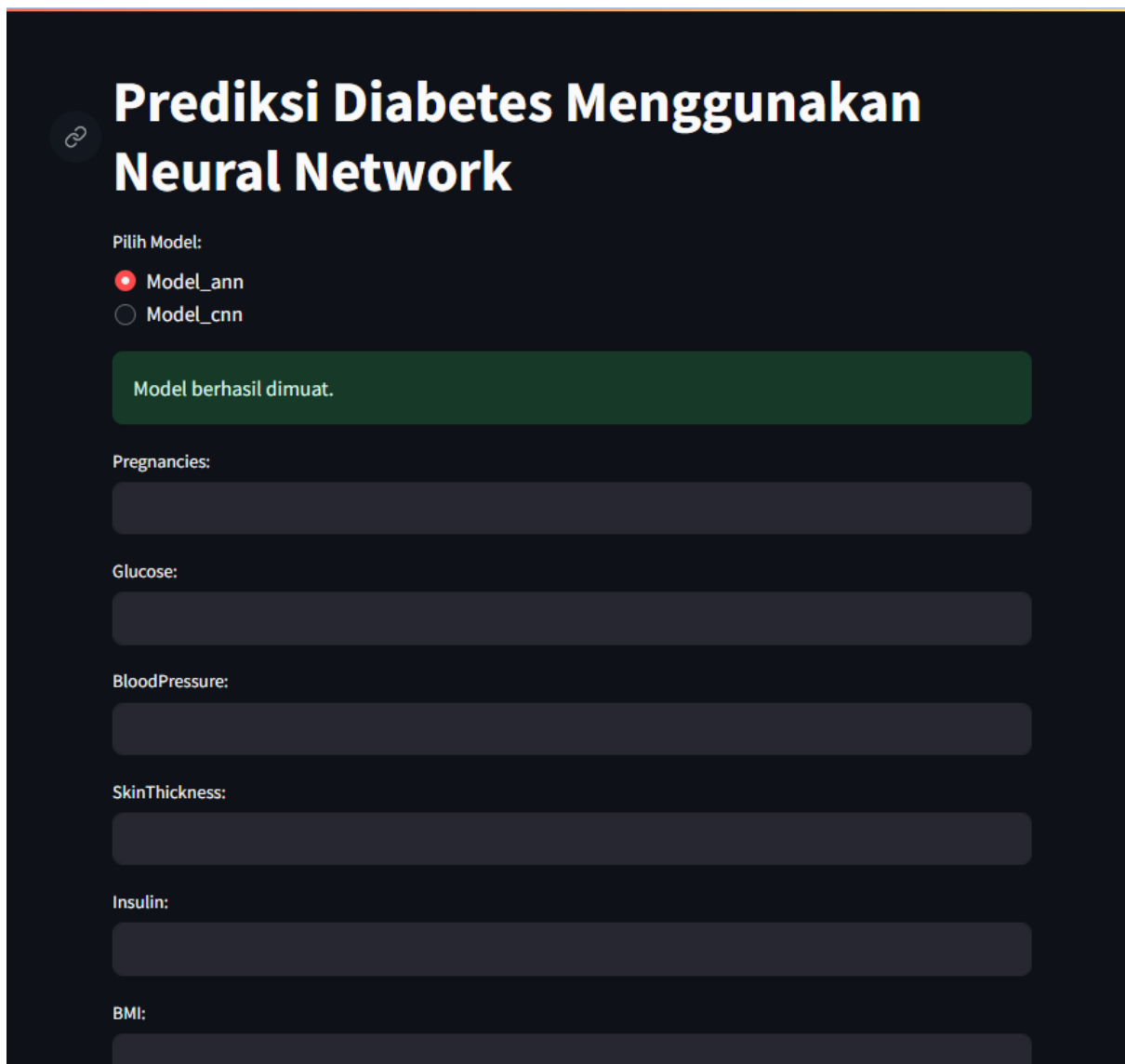
Desain *User Interface (UI/UX)* aplikasi web prediksi diabetes dibuat dengan mempertimbangkan kebutuhan pengguna serta memberikan pengalaman pengguna yang sederhana dan informatif. Beberapa elemen UI yang disertakan dalam aplikasi web ini antara lain:

#### 4.4.1.1 Tombol Pilihan Jenis Neural Network

Sebelum user memasukan data variabel user diperbolehkan untuk memilih jenis Neural Network mana yang diinginkan apakah jenis ANN atau CNN lalu untuk tipe tombol yang digunakan adalah tipe *Radio Button*.

#### 4.4.1.2 Formulir Input

Pengguna dapat memasukkan data yang diperlukan untuk melakukan prediksi diabetes melalui formulir input. Formulir input mencakup variabel-variabel seperti jumlah *Pregnancies*, *BMI*, *Bloodpressure*, *Skin thickness*, *DiabetesPedigreeFunction*, *Glucose*, dan *Insulin*..



Tampilan UI *input* dan pilihan *Neural Network*

#### 4.4.1.3 Tombol Prediksi

Setelah pengguna mengisi formulir input, aplikasi menyediakan tombol prediksi yang akan mengaktifkan model *Neural Network* untuk melakukan prediksi berdasarkan data yang dimasukkan.

#### 4.4.1.4 Tampilan Prediksi

Hasil prediksi ditampilkan dengan jelas kepada pengguna. Informasi tersebut dapat mencakup hasil prediksi (positif/negatif diabetes) dan probabilitas prediksi.



Tampilan tombol Prediksi beserta hasil

#### 4.4.2 Implementasi Aplikasi Web

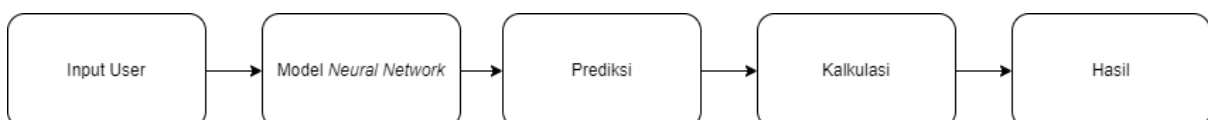
Implementasi aplikasi web prediksi diabetes dilakukan menggunakan framework Streamlit sebagai platform utama pengembangan web app. Berikut adalah langkah-langkah implementasi :

##### 4.4.2.1 Integrasi Model *Neural Network*

Model *Neural Network* yang telah dilatih sebelumnya diintegrasikan ke dalam aplikasi web menggunakan Streamlit lalu menghubungkannya dengan model yang telah dibuat beserta bobotnya. Hal ini memungkinkan aplikasi untuk melakukan prediksi berdasarkan input pengguna.

##### 4.4.2.2 Penanganan Input dan Prediksi

Aplikasi web dibangun untuk menangani input dari pengguna dan menggunakan model *Neural Network* untuk menghasilkan prediksi. Alur kerja ini diimplementasikan secara bersih dan efisien.





## Diagram Pemrosesan

### 4.4.3 Pengujian Aplikasi

Pada tahap ini, aplikasi web prediksi diabetes diuji coba untuk memastikan kinerjanya sesuai dengan yang diharapkan. Pengujian mencakup uji fungsionalitas, responsivitas, dan keakuratan prediksi.

- a) Fungsionalitas pada tahap ini aplikasi dites semua tombolnya apakah semua tombol tersebut berfungsi dengan baik ataukah tidak dan juga tidak lupa pada bagian input pada bagian input jika kita memasukan nilai selain angka maka akan terjadi error.
- b) Responsivitas pada tahap ini WebApp dites secepat apa responya atau jeda waktunya setelah kita melakukan perintah apakah jeda tersebut terlalu lama ataukah tidak. Dan ternyata jedaanya tidak terlalu lama yang berarti WebApp ini cukup responsif.
- c) Keakuratan prediksi pada tahap ini adalah tahapan paling penting karena jika prediksinya terlalu banyak meleset maka Web App ini dapat dinyatakan tidak efektif untuk memprediksi diabetes. Disini akan dilakukan penginputan data dari dataset sebelumnya, dan setelah dimasukan hasilnya banyak yang meleset dan harus dievaluasi ulang pada Web Appnya.

### 4.4.4 Analisis dan Evaluasi

Hasil pengujian dievaluasi dan dianalisis untuk memastikan bahwa aplikasi web dapat memberikan prediksi diabetes dengan akurat dan memberikan pengalaman pengguna yang baik.

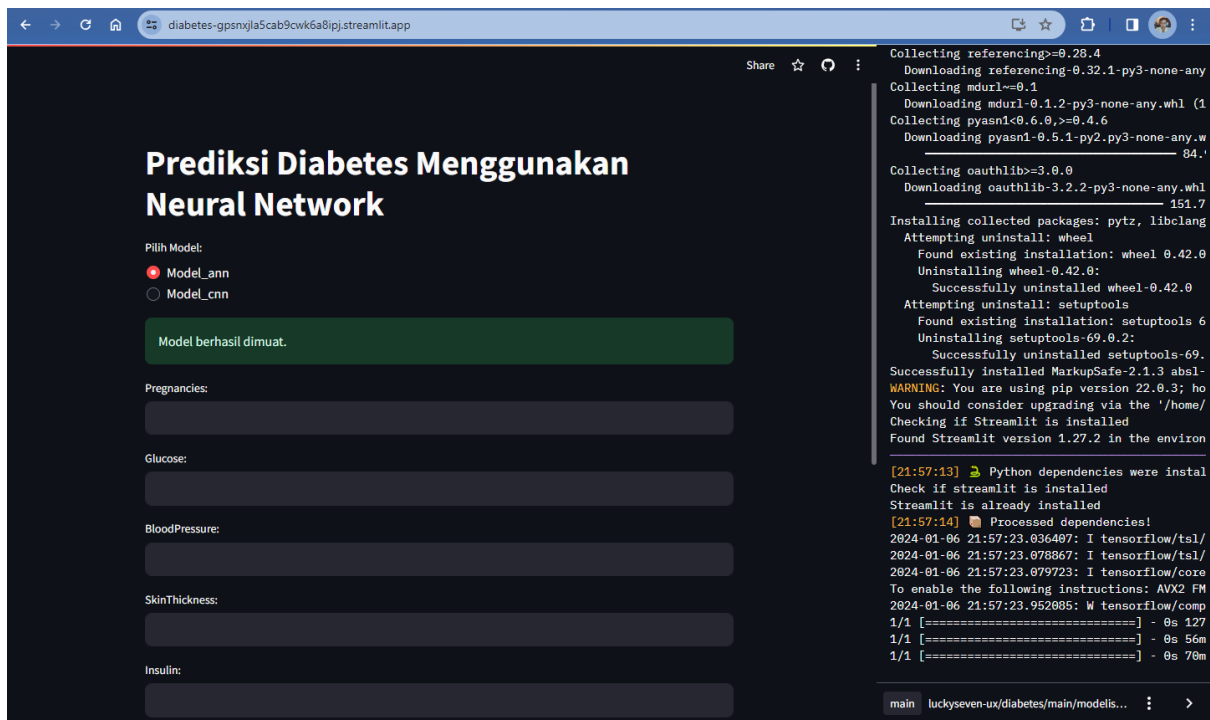
Pada tahap ini dilakukan evaluasi pada Web App dengan cara mengamati nilai prediksi dari pengidap diabetes dan yang tidak terkena diabetes dari dataset dengan cara melakukan input pada Web App dari dataset yang digunakan pada model *Neural network*.

Lalu disini didapatkan pola jika nilai prediksinya kurang dari 0.7 maka orang tersebut terkena diabetes lalu jika nilai prediksinya lebih dari 0.7 maka orang tersebut bebas diabetes setelah itu kita lakukan evaluasi pada bagian prediksinya dengan memberi aturan jika nilai prediksinya lebih dari 0.7 maka orang tersebut bebas diabetes.

Kemudian dilakukan lagi pengujian dan hasilnya cukup akurat untuk memprediksi diabetes dari dataset tadi.

#### 4.4.5 Penguploadan Aplikasi Web ke Hosting

Setelah semua proses tadi selesai Aplikasi web ini diupload pada hosting Streamlit Cloud agar aplikasi Web Machine learning ini dapat digunakan kapan pun dan dimanapun hanya dengan koneksi internet. Dan bisa anda akses pada link berikut : <https://diabetes-gpsnxjla5cab9cwk6a8ipj.streamlit.app/>



Tampilan Web App Setelah di unggah ke Hosting

## **BAB V**

### **KESIMPULAN DAN SARAN**

#### **5.1 Kesimpulan**

Berdasarkan implementasi dan pengujian yang telah dilakukan sebelumnya, diperoleh hasil akurasi, f1-score, dan recall. Hasil-hasil tersebut membuktikan bahwa:

a) Aplikasi Deteksi Diabetes Berbasis Neural Network Efektif:

Aplikasi deteksi diabetes menggunakan algoritma neural network pada sistem ini terbukti efektif dalam mendiagnosis kondisi diabetes. Hasil akurasi yang tinggi menunjukkan potensi aplikasi untuk membantu pengguna meminimalisir risiko terkena penyakit tersebut.

b) Identifikasi Cepat:

Penerapan algoritma neural network pada aplikasi deteksi diabetes mempermudah identifikasi berbagai faktor dan gejala yang terkait dengan kondisi diabetes. Hal ini memungkinkan pengguna untuk lebih cepat mengenali potensi risiko diabetes, memungkinkan tindakan pencegahan yang lebih efektif.

c) Pentingnya Penerapan Teknologi dalam Kesehatan:

Keberhasilan aplikasi ini menegaskan pentingnya penerapan teknologi, khususnya algoritma neural network, dalam bidang kesehatan. Teknologi dapat menjadi alat yang efektif untuk mendukung diagnosis dan pencegahan penyakit.

#### **5.2 Saran**

Adapun saran yang ingin disampaikan oleh penulis untuk penelitian berikutnya antara lain :

a) Uji Klinis dan Validasi Medis:

Melibatkan uji klinis dan validasi medis lebih lanjut akan meningkatkan kepercayaan dan penerimaan aplikasi dalam komunitas medis. Hal ini dapat membuka pintu bagi integrasi lebih lanjut dalam praktik klinis.

b) Evaluasi Aspek Keamanan dan Privasi:

Penelitian mendetail terkait aspek keamanan dan privasi dalam penggunaan aplikasi kesehatan sangat penting. Evaluasi menyeluruh terhadap kebijakan privasi dan langkah-langkah keamanan harus diintegrasikan.

c) Pengembangan Versi Aplikasi yang Dapat Diakses Lebih Luas:

Agar manfaat aplikasi dapat dirasakan oleh lebih banyak orang, pengembangan versi yang dapat diakses secara lebih luas, seperti aplikasi seluler, dapat dijelajahi.

d) Penelitian Lanjutan Mengenai Model Neural Network yang Optimal:

Penelitian lebih lanjut mengenai jenis dan konfigurasi model neural network yang optimal untuk deteksi diabetes dapat meningkatkan kinerja aplikasi.

e) Kolaborasi dengan Pihak Kesehatan dan Institusi Medis:

Kolaborasi dengan pihak kesehatan dan institusi medis dapat membuka peluang untuk penerapan aplikasi dalam lingkungan klinis yang lebih luas.

## DAFTAR PUSTAKA

- Abdurrohman, H., Dini, R., & Muharram, A. P. (2018). Evaluasi Performa metode Deep Learning untuk Klasifikasi Citra Lesi Kulit The HAM10000. *Seminar Nasional Instrumentasi, Kontrol dan Otomasi (SNIKO)*.
- Admojo, F. T., & Ahsanawati. (2020). Klasifikasi Aroma Alkohol Menggunakan Metode KNN. *Indonesian Journal of Data and Science*, 34-38.
- Aprilia, W., & dkk. (2021). Prediksi Kemungkinan Diabetes pada Tahap Awal. *Sistemasi: Jurnal Sistem Informasi*, 163-171.
- Ataline Jeanethe Maya Hukubun. (2022).
- Aulia, A., Aprianti, B., Supriyanto, Y., & Rozikin, C. (2022). prediksi harga emas dengan menggunakan algoritma support vector regression (Svr) dan linear regression (lr). *jurnal ilmiah wahana pendidikan*, 84-88.
- Bhakti, H. D. (2019). Aplikasi Artificial Neural Network (ANN) untuk Memprediksi Masa Studi Mahasiswa Program Studi Teknik Informatika Universitas Muhammadiyah Gresik. *Jurnal Eksplora Informatika*, 88-95.
- Bustami. (2014). Penerapan Algoritma Naive Bayes Untuk Mengklasifikasi Data Nasabah Asuransi. *Jurnal Informatika* , 884-898.
- Cahyani, Q., & dkk. (2022). Prediksi Risiko Penyakit Diabetes menggunakan Algoritma Regresi Logistik. *Journal of Machine Learning and Artificial Intelligence*, 107-114.
- Collonov, F., Dafna, I., Ivanov, P., & Eric, C. (2023, July 4). *Project Jupyter's origins and governance*. Retrieved from Jupyter: <https://jupyter.org/>
- Dagliati, A., Marini, S., & dkk. (2018). Machine Learning Methods to Predict Diabetes. *Journal of diabetes science and technology*, 295-302.
- Eska, J. (2016). Penerapan Data Mining Untuk Prediksi Penjualan Wallpaper Menggunakan Algoritma C4.5. *JURTEKSI (Jurnal Teknologi dan Sistem Informasi)*, 9-13.
- Gulli, A., & Pal, S. (2017). *Deep Learning With Keras*. Brimingham: Packt Publishing.
- Hadianto, N., Novitasari, H. B., & Rahmawati, A. (2019). Klasifikasi Peminjaman Nasabah Bank Menggunakan Metode Neural Network. *PILAR Nusa Mandiri*, 163-170.

- Handayani, M., Riandini, M., & Situmorang, Z. (2022). Perbandingan Fungsi Optimasi Neural Network Dalam Klasifikasi Kelayakan Calon Suami. *Jurnal Informatika*, 78-84.
- Hasan, K., Alam, A., Hussein, D., & Hasan, M. (2020). Diabetes Prediction Using Ensembling of Different Machine Learning Classifiers. *IEEE Access* 8, 76516-76531.
- Hunter, J., Dale, D., Firing, E., & Droettboom, M. (2023, february 13). *Library matplotlib*. Retrieved from matplotlib: [https://matplotlib.org/stable/users/release\\_notes.html](https://matplotlib.org/stable/users/release_notes.html)
- Istighfarizky, F., ER, N. A., Widiartha, I. M., Astutia, L. G., Putra, I. G., & Suhartana, I. K. (2022). Klasifikasi Jurnal menggunakan Metode KNN dengan Mengimplementasikan Perbandingan Seleksi Fitur. *Jurnal Elektronik Ilmu Komputer Udayana*, 167-176.
- Kaur, A., & Rout, M. (2020). Evaluating Machine Learning Methods for Predicting Diabetes among Female Patients in Bangladesh. *International Conference on Intelligent Engineering and Management (ICIEM)*, 1-20.
- Munawarah, R., Soesanto, O., & Faisal, M. R. (2016). Penerapan Metode Support Vector Machine Pada Diagnosa Hepatitis. *Kumpulan Jurnal Ilmu Komputer (KLIK)*, 103-113.
- Nagaraj, P., & Deepalakshmi, P. (2021). Diabetes Prediction Using Diabetes Prediction Using Enhanced SVM and Deep Neural Network Learning Techniques: An Algorithmic Approach for Early Screening of Diabetes. *International Journal of Healthcare Information Systems and Informatics (IJHISI)*, 1-20.
- Nasution, M. R., & Hayaty, M. (2019). Perbandingan Akurasi dan Waktu Proses Algoritma K-NN dan SVM dalam Analisis Sentimen Twitter. *JURNAL INFORMATIKA*, 6(2), 212-218.
- Naz, H., & Ahuja, S. (2020). Deep learning approach for diabetes prediction using PIMA Indian dataset. *Journal of Diabetes & Metabolic Disorders*, 391-403.
- Nelli, F. (2015). *Data Analysis and Science Using Pandas, matplotlib, and the Python Programming Language*.
- Nelli, F. (2015). *Python Data Analytics*. New York: Apress Media.
- Noviyanto. (2020). Penerapan Data Mining dalam Mengelompokkan Jumlah Kematian. *Jurnal Informatika dan Komputer*, 183-188.

- Patra, R., & Khuntia, B. (2021). Analysis and Prediction Of Pima Indian Diabetes Dataset Using SDKNN Classifier Technique. *IOP Conference Series: Materials Science and Engineering*, 1-14.
- R, C., Harris, Millman, K. J., & Vander, J. W. (2020, sep). Array programming with NumPy. *Nature*, 585, 357-362. Retrieved from <https://doi.org/10.1038/s41586-020-2649-2>
- Ramadhan, N. A. (2019). Penerapan Metode Neural Network Untuk Prediksi Nilai Ujian Nasional (Study Kasus Di Smk Muhammadiyah Slawi). *Jurnal Ilmiah Indonesia*, 118-130.
- Rohman, y. A. (2019, 12 8). *Pengenalan NumPy, Pandas, Matplotlib*. Retrieved from medium.com: <https://medium.com/@yasirabd/pengenalan-numpy-pandas-matplotlib-b90bafd36c0>
- Soni, M., & Varma, S. (2020). Diabetes Prediction using Machine Learning Techniques. *International Journal of Engineering Research & Technology (ijert)*, 921-925.
- Syarovy, M., & Sutiarto, A. P. (2023). Pemanfaatan Model Neural Network Dalam Generasi Baru Pertanian Presisi di Perkebunan Kelapa Sawit. *Warta PPKS*, 39-54.
- VanderPlas, J. (2017). *Python Data Science Handbook*. United States of America: O'Reilly Media, Inc.,.
- Wadi, H. (2015). *Pemrograman Python : untuk pelajar dan mahasiswa*. Bandung: TR Publisher.
- Waskom, M. L. (2021). seaborn: statistical data visualization. *Journal of Open Source Software*, 6, 3021. Retrieved from <https://doi.org/10.21105/joss.03021>
- Widiputra, H. D. (2016).

## Daftar Riwayat Hidup