

**LAPORAN AKHIR PENELITIAN
DANA MANDIRI**



**Implementasi *Machine Learning*
Dalam Sistem Pakar Diagnostik**

Ketua : Dra. Sulistyowati, M.Kom 0324056703
Anggota :
1. Sunarto, M Kom 0307066502
2. Umar Khoirul Afif 1152000081

**PROGRAM STUDI TEKNIK INFORMATIKA
INSTITUT TEKNOLOGI INDONESIA
Agustus, 2024**

HALAMAN PENGESAHAN

Judul Penelitian : Implementasi *Machine Learning* Dalam Sistem Pakar Diagnostik

Jenis Penelitian^{a)} : Dana Mandiri

Bidang Penelitian^{b)} : Ilmu Komputer

Tujuan Sosial Ekonomi^{c)} : Aplikasi

Peneliti

a. Nama Lengkap : Dra. Sulistyowati, M.Kom

b. NIDN : 0324056703

c. Jabatan Fungsional : Lektor

d. Program Studi : Teknik Informatika

e. Nomor HP : 08176543515

f. Alamat Surel (e-mail) : liliswiyono2403@gmail.com

Anggota Peneliti 1

a. Nama Lengkap : Sunarto, M Kom

b. NIDN : 0307066502

c. Institusi : Institut Teknologi Indonesia

Anggota Peneliti 2

a. Nama Lengkap : Umar Khoirul Afif

b. NIM : 1152000081

c. Institusi : Institut Teknologi Indonesia

Institusi Sumber Dana^{d)} : Pribadi

Biaya Penelitian : Rp. 10.000.000,

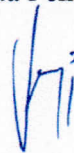
Kota Tangerang Selatan, Agustus 2024

Mengetahui,
Kaprosdi Teknik Informatika



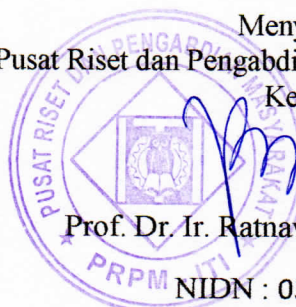
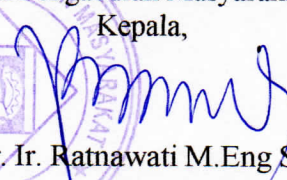
Muhammad Soleh, S Si, M Kom
NIDN : 0302128902

Ketua Peneliti,



Dra. Sulistyowati, M Kom
NIDN : 0324056703

Menyetujui,
Pusat Riset dan Pengabdian Masyarakat (PRPM) ITI
Kepala,



Prof. Dr. Ir. Ratnawati M.Eng Sc. IPM
NIDN : 0301036303



INSTITUT TEKNOLOGI INDONESIA

Jl. Raya Puspiptek, Tangerang Selatan - 15314
(021) 7562757

www.iti.ac.id [institutteknologiindonesia](https://www.instagram.com/institutteknologiindonesia) [@kampusITI](https://www.facebook.com/kampusITI) [Institut Teknologi Indo](https://www.youtube.com/channel/UC...)

SURAT TUGAS

No. : 026/ST-PLT/PRPM-PP/ITI/V/2024

- Pertimbangan** : Bahwa dalam rangka melaksanakan kegiatan Penelitian bagi Dose Program Studi Informatika Institut Teknologi Indonesia, perl dikeluarkan surat tugas.
- Dasar** : 1. Pembebanan Tugas dosen Program Studi Informatika;
2. Surat Permohonan Tanggal 27 Mei 2024;
3. Kepentingan Institut Teknologi Indonesia.

DITUGASKAN

- Kepada** : Dosen Program Studi Informatika – ITI (Terlampir)
- Untuk** : 1. Melaksanakan kegiatan Penelitian pada Semester Genap Tahun Akademik 2023/2024;
2. Melaporkan hasil tugas kepada Kepala PRPM-ITI;
3. Dilaksanakan dengan penuh rasa tanggung jawab.

Tangerang Selatan, 30 Mei 2024
Pusat Riset dan Pengabdian Masyarakat
Kepala,


Prof. Dr. Ir. Ratnawati, M.Eng.Sc., IPM

Tembusan Yth.

1. Wakil Rektor Bid Akademik, Penelitian dan Kemahasiswaan
2. Ka. Biro SDMO
3. Ka. Prodi Informatika
4. Arsip

Lampiran Surat Tugas

No. 026/ST-PLT/PP/RSVA-PP/IT/IV/2024

Tanggal 27 Mei 2024

DAFTAR PENELITIAN DOSEN PROGRAM STUDI INFORMATIKA SEMESTER GENAP TAHUN AKADEMIK 2023/2024

Revisi 12 Agustus 2024

NO	TOPIC PENELITIAN	DISIPLIN ILMU	BLOK/WAKTU	SUMBER DANA	KLIMAH DANA (Rp)	KETERLIBATAN PRODUK/ INSTITUSI LAIN	KETERLIBATAN MAHASISWA
1	Electronic Voting (e-voting) as Decentralized Application on the Blockchain	Engineering and Technology	Ketua: Surjo Bramono, S.T., M.T Anggota: Dra. Endang Rahmawati Djuwakarsangum, M.Kom	Mandiri	10.000.000	Veraniam	Sandiana Ribia Savitri (NRP: 1151900004)
2	Implementasi Sistem Informasi Geografis pada Menjelajah Tulo Sembako	Engineering and Technology	Ir. Samiati Andri, M.Kom	Mandiri	10.000.000	Tidak Ada	Fandi Khalk (NRP: 115200033)
3	Paket dan Penyelesaian Outbreak di Masa Pandemi Covid-19	Engineering and Technology	Ketua: Ir. Yulidia Sri Sulaini, S.T., M.T Anggota: Muhammad Ramli, S.T., M.Kom	Mandiri	10.000.000	Tidak Ada	Tidak Ada
4	Penyusunan Metode User Centered Design pada Pengembangan Front End Platform Pembelajaran Berbasis Website PT. Bina Artificial Indonesia	Engineering and Technology	Melani Indrasari, S.T., M.Kom	Mandiri	10.000.000	PT. Bina Artificial Indonesia	Bino Tri Darmawan (NRP: 1151900008)
5	Pengembangan Model Generasi Link Lupa Otomatis dengan Ulangi Langkah Monev	Engineering and Technology	Dhno Harabna Putra, S.T., M.Kom	Mandiri	10.000.000	Tidak Ada	Tidak Ada
6	Pemeriksaan Kulit Sakak Pondoh (Solozoi zakozoi) sebagai Minuman Kesehatan	Engineering and Technology	Ketua: Ir. Muhamad, M.Si., IPM Anggota: 1. Ir. Danti Nurani, M.Si., IPM 2. Ir. Syahid Makosim, M.Si., IPM 3. Prof. Dr. Ir. Subandari, Sachmoemar, M.Si. 4. Dra. Indro Sukmadi, M.Sc	Institut Teknologi Indonesia (ITI)	10.000.000	Prodi Teknologi Industri Pertanian (TPI) ITI	Tidak Ada
7	Implementasi Machine Learning dalam Membenarkan Sistem Pakar Diagnosis	Engineering and Technology	Ketua: Dra. Subiyowati, M.Kom Anggota: Sunarto, S.Kom, M.Kom	Mandiri	10.000.000	Tidak Ada	Umar Khoirul Afif (NRP: 1152000081)
8	Implementasi Algoritma Convolutional Neural Network untuk Mendeteksi Cytiva X-ray Perut-paru dalam Sistem Aplikasi Penemuan Tuberkulosis	Engineering and Technology	Muhammad Suhel, S.S., M.Kom	Institut Teknologi Indonesia (ITI)	11.000.000	Tidak Ada	Dandi Rifaldi Aidiansyah (NRP: 1151900072)
9	Pendekatan A dan Data Salin dalam Bencana Geo-Hidroметеоролог di Sumatera Utara	Engineering and Technology	Ir. Yulidia Sri Sulaini, S.T., M.T	Mandiri	10.000.000	PSPP & BMKG	Marzuki Simambela (NRP: 2202400118)

Pusat Riset dan Pengabdian Masyarakat
Institut Teknologi Indonesia
Kepala



Prof. Dr. Ir. Ratna wati, M.Eng.Sc., IPM

PRAKATA

Puji Syukur ke hadirat Tuhan yang telah melimpahkan pengetahuan dan hikmat yang diperlukan sehingga laporan penelitian ini dapat diselesaikan. Penelitian dengan judul Implementasi *Machine Learning* Dalam Sistem Pakar Diagnostik ini dilakukan dengan mencermati perkembangan ilmu kecerdasan buatan, khususnya machine learning yang sudah menyentuh berbagai bidang termasuk bidang kesehatan.

Akhir kata, semoga laporan penelitian ini dapat bermanfaat bagi pembacanya.

Tangerang Selatan, Agustus 2024

Penulis

ABSTRAK

Transformasi Digital di bidang kesehatan dapat diketahui salah satunya dengan berkembangnya aplikasi Sistem Pakar Diagnostik. Sistem Pakar Diagnostik bisa menjembatani penderita suatu penyakit dengan seorang pakar. Jadi sebelum bertemu atau berkonsultasi dengan seorang pakar, seseorang bisa terlebih dahulu mengetahui diagnosa awal terkait kemungkinan penyakit yang dideritanya, yaitu dengan menggunakan aplikasi Sistem Pakar. Dalam Penelitian ini dibangun Sistem Pakar Diagnostik untuk mendiagnosa apakah seorang perokok aktif atau pasif mengidap suatu penyakit tertentu terkait dengan aktivitas merokoknya. Aplikasi dibangun dengan memanfaatkan teknologi *machine learning* dan mengimplementasikan model algoritma *randomforest*. Model dilatih dengan menggunakan dataprimer yang dikumpulkan dengan cara melempar kuesioner dan diperoleh 184 respon responden. Adapun poin poin pertanyaan dalam kuesioner divalidasi terlebih dahulu oleh seorang pakar. Hasil evaluasi model menunjukkan pencapaian akurasi sebesar 94%. Selanjutnya dikembangkan aplikasi dalam bentuk aplikasi berbasis web dengan menggunakan *framework Python Flask*.

Kata kunci : *machine learning, Python Flask, Random Forest*, sistem pakar diagnostik, web aplikasi.

BAB 1 PENDAHULUAN

1.1 Latar Belakang

Asap rokok menjadi topik yang penting terutama mereka yang sampai saat ini masih tidak mau meninggalkan kebiasaan merokok. Perlu diperhatikan bahwa asap rokok mengandung beberapa jenis zat kimia terutama karbon monoksida, hidrogen sianida, tar, dan nikotin yang merupakan racun bagi tubuh manusia. Bahkan organisasi kesehatan dunia, *WHO* juga secara khusus mengingatkan bahaya dari asap rokok. (Johan, 2023).

Indonesia tercatat sebagai negara dengan jumlah perokok terbanyak kedua di dunia, dengan sekitar 33,8% atau sekitar 65,7 juta orang yang merokok, setelah Amerika. Data dari Badan Pusat Statistik pada tahun 2021 menunjukkan bahwa jumlah perokok pada usia 13-15 tahun mencapai sekitar 1,44%, dan pada usia 16-18 tahun sekitar 3,69%. Konsumsi tembakau dalam rokok diketahui membunuh sekitar delapan juta jiwa setiap tahunnya. Perserikatan Bangsa-Bangsa (*PBB*) melaporkan bahwa 3,3 juta pengguna rokok akan meninggal karena penyakit yang berhubungan dengan paru-paru serta kanker. (Fathoni et al., 2023).

Rokok memberikan dampak buruk bagi kesehatan karena mengandung *nikotin*, zat berbahaya yang berasal dari *tembakau*. *Nikotin*, meski dalam dosis rendah, hanya mengganggu pernapasan, dalam jumlah banyak, *nikotin* dapat menghambat aliran darah sehingga menimbulkan berbagai penyakit seperti jantung, stroke, kanker, impotensi, dan gangguan kesuburan. Selain *nikotin*, rokok juga mengandung racun lain yang dapat mempercepat penuaan kulit dan merusak gigi. Biasanya seseorang yang mengidap penyakit tersebut baru memeriksakan diri setelah ada di stadium lanjut. Padahal tentu sebelumnya sudah ada beberapa gejala yang dialami tubuh. Selain kurangnya kesadaran akan kesehatan, seringkali seseorang tidak memeriksakan diri juga karena besarnya upaya yang perlu dilakukan saat akan memeriksakan diri.

Perkembangan teknologi dan komputer telah meluas ke berbagai bidang seperti pendidikan, sosial, dan bahkan kesehatan. Hal ini memicu terjadinya transformasi digital

di berbagai sektor, termasuk sektor kesehatan. Salah satu wujud transformasi digital di bidang kesehatan adalah dengan banyaknya aplikasi Sistem Pakar, yaitu suatu aplikasi yang berfungsi seperti dokter atau pakar. Sistem Pakar dibangun berdasarkan literatur dan pengalaman kepakaran seorang pakar. Namun demikian seringkali kondisi faktual tidak seperti yang ada di literatur, sehingga sering ditemui kasus salah diagnosa. Sehingga seseorang dianggap pakar bukan saja karena bisa menerapkan keilmuan secara tepat tapi juga secara bijak.

Machine learning merupakan salah satu cabang *kecerdasan buatan (AI)* yang memungkinkan komputer belajar sendiri tanpa harus diprogram secara eksplisit. *Machine Learning* menggunakan data untuk melatih model yang dapat digunakan untuk membuat prediksi atau mengambil keputusan. Dengan demikian *machine learning* bisa membuat komputer mampu menangkap pola pola dalam mengambil keputusan berdasar data sebelumnya. Saat ini *machine learning* sudah banyak diterapkan dalam membuat suatu aplikasi cerdas.

1.2 Rumusan Masalah

Berdasarkan uraian di atas, rumusan masalah yang dapat diambil adalah bisakah dibuat suatu aplikasi sistem pakar dengan menerapkan *machine learning*.

1.3 Tujuan

Tujuan dari penelitian ini adalah untuk membuat suatu aplikasi sistem pakar pendeteksi dini penyakit pada perokok berbasis web yaitu dengan mendeteksi gejala-gejala yang dirasakan, dengan menerapkan *machine learning*

1.4 Batasan Masalah

Agar pengerjaan tugas akhir ini mendapatkan hasil yang spesifik, maka sistem yang dirancang dan dibuat dibatasi dengan ruang lingkup pembahasan, yaitu:

1. Aplikasi yang dibangun adalah aplikasi berbasis website. Website ini dibuat sebagai media yang akan memberikan gambaran umum tentang sistem pakar untuk mendeteksi dini penyakit akibat merokok.
2. Pembuatan aplikasi ini menggunakan bahasa pemrograman *Python* di kembangkan menggunakan *framework Python Flask* dan menggunakan metode pengumpulan data serta menggunakan *ESDLC* sebagai metode pengembangan sistem.

3. Kecerdasan yang ditanamkan dimodelkan dengan menggunakan data primer dari 184 responden dan menggunakan model *Random Forest*.
4. Jenis penyakit pada perokok yang akan dibahas terbatas sesuai dengan kepakaran narasumber.

BAB 2

TINJAUAN PUSTAKA

2.1. *Artificial Intelligence*

Artificial Intelligence (AI) atau biasa disebut Kecerdasan Buatan didefinisikan sebagai kecerdasan yang ditunjukkan oleh sebuah mesin atau *software*. Menurut Kusumadewi (2003), kecerdasan buatan merupakan salah satu bagian dalam ilmu komputer yang membuat agar mesin (komputer) dapat melakukan pekerjaan seperti dan sebaik yang dilakukan manusia. Kecerdasan diciptakan dan dimasukkan ke dalam suatu mesin (komputer) agar dapat melakukan pekerjaan yang dapat dilakukan manusia. Beberapa macam bidang yang menggunakan kecerdasan buatan antara lain sistem pakar (*expert system*), permainan komputer, logika *fuzzy*, jaringan syaraf tiruan (*artificial neural network*), robotika (*robotics*), pengolahan bahasa alami (*natural language processing*), pengenalan pola (*pattern recognition*), dan pengenalan suara (*speech recognition*) (Simarmata, 2006).

Kecerdasan buatan ditujukan dalam perancangan otomatisasi tingkah laku cerdas dalam sistem kecerdasan komputer. Pengaplikasian kecerdasan buatan terdiri dari 2 bagian utama yang sangat dibutuhkan,, yaitu (Kusumadewi, 2003):

- a. Basis Pengetahuan (*Knowledge Base*), berisi fakta-fakta, teori, pemikiran dan hubungan antara satu dengan lainnya.
- b. Motor Inferensi (*Inference Engine*), yaitu kemampuan untuk menarik kesimpulan berdasarkan pengalaman.

2.2. *Machine Learning*

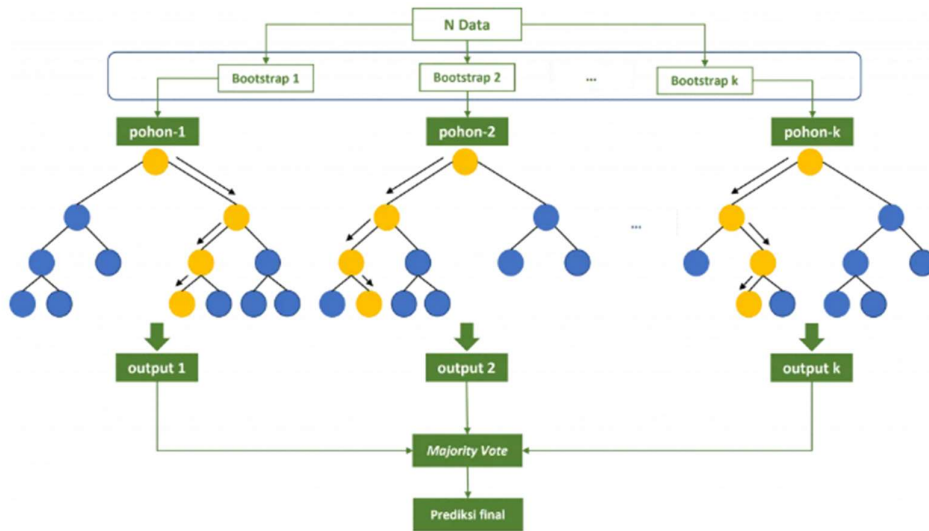
Machine Learning (ML) merupakan serangkaian teknik yang dapat membantu dalam menangani dan memprediksi data yang sangat besar dengan cara merepresentasikan data-data tersebut dengan

algoritma pembelajaran (Danukusumo, 2017). Istilah *machine learning* pertama kali didefinisikan oleh Arthur Samuel pada tahun 1959. Menurutnya, *machine learning* adalah suatu bidang ilmu komputer yang memberikan kemampuan pembelajaran kepada komputer untuk mengetahui sesuatu tanpa pemrograman yang jelas.

Menurut (Mohri et.al, 2012), *machine learning* dapat didefinisikan sebagai metode komputasi berdasarkan pengalaman untuk meningkatkan performa atau membuat prediksi yang akurat. Definisi pengalaman di sini ialah informasi sebelumnya yang telah tersedia dan bisa dijadikan data pembelajar.

2.3 Random Forest

Random Forest adalah model ansambel berbasis pohon yang sangat populer dalam pembelajaran mesin. Diperkenalkan oleh Leo Breiman pada tahun 2001, model ini dapat digunakan untuk pemodelan regresi dan klasifikasi. Dalam regresi, prediksi dihitung sebagai rata-rata hasil yang diperoleh dari setiap pohon keputusan. Sedangkan untuk klasifikasi, prediksi ditentukan berdasarkan suara terbanyak dari seluruh pohon. Misalnya ada model *random forest* dengan 100 pohon dan 72 pohon memprediksi data sebagai kelas A, sedangkan 28 pohon lainnya memprediksinya sebagai kelas B, maka data tersebut akan diprediksi sebagai kelas A. (Cahya Alkahfi, 2024).



Gambar 2.1 *Random Forest*

Agar prediksinya akurat dan stabil, *random forest* menggunakan metode *bagging* (*bootstrap aggregating*). Metode ini menggabungkan beberapa algoritma untuk meningkatkan akurasi dalam *data mining* (*machine learning*). *Bagging* bekerja dengan mengambil sampel data secara acak dari dataset asli. Proses ini diawali dengan pengambilan sampel awal (*raw sampling*), kemudian setiap sampel diganti dengan *bootstrap*, sehingga menghasilkan sampel *bootstrap* yang digunakan untuk membangun pohon keputusan. (Rasya Anwar, 2024). Cara kerja *random forest* dapat dijelaskan melalui langkah-langkah utama berikut:

1. Pengambilan Sampel *Bootstrap*. *Random Forest* dimulai dengan mengambil beberapa sampel acak (dengan penggantian) dari dataset asli untuk membentuk beberapa subset dari data. Proses ini disebut *bootstrap sampling*, dan setiap subset digunakan untuk melatih satu pohon keputusan.

2. Pembentukan Pohon Keputusan. Setiap subset data digunakan untuk membentuk pohon keputusan. Dalam proses ini, hanya subset fitur acak yang dipertimbangkan pada setiap node pembelahan. Hal ini memastikan pohon yang dihasilkan bervariasi dan tidak bergantung satu sama lain.

3. Pembelahan Node. Setiap pohon dibangun dengan memilih fitur dan titik pemisahan yang memaksimalkan pemisahan data menjadi kelas atau nilai target. Proses ini berlanjut hingga pohon mencapai kedalaman maksimum yang ditentukan atau tidak ada lagi pembagian yang dapat dilakukan.

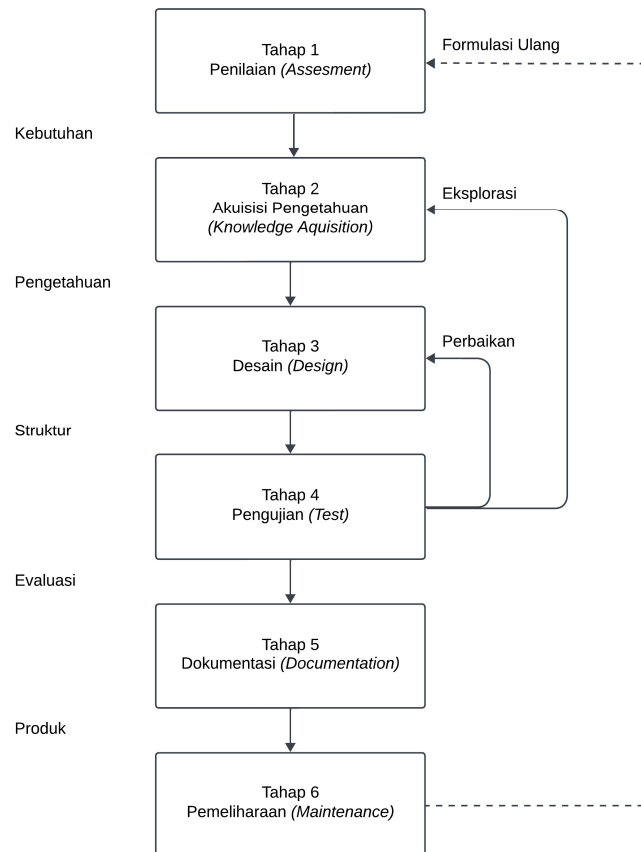
4. Pemungutan suara/Rata-rata. Setelah semua pohon di hutan terbentuk, *random forest* menggabungkan hasil setiap pohon untuk menghasilkan prediksi akhir. Untuk klasifikasi, prediksi diambil berdasarkan suara terbanyak dari hasil prediksi setiap pohon. Untuk regresi, prediksi diambil dengan menghitung rata-rata hasil prediksi dari masing-masing pohon.

5. Penilaian dan Validasi. Model *random forest* divalidasi menggunakan data uji atau metode validasi silang untuk menilai kinerja dan akurasi. Keandalan suatu model diukur berdasarkan kemampuannya dalam memprediksi data yang belum pernah dilihat.

BAB 3

METODOLOGI PENELITIAN

Dalam pengembangan sistem pakar, pendekatan metode *Expert System Development Life Cycle (ESDLC)* telah digunakan, seperti yang disajikan dalam gambar berikut.



Gambar 3.1 *Expert System Development Life Cycle (ESDLC)*

Proses ini mencakup berbagai tahapan yang dirancang untuk memastikan bahwa sistem pakar dapat berfungsi secara optimal dan memberikan hasil yang akurat sesuai dengan kebutuhan pengguna.

Penilaian (Assesment)

Pada tahap ini dilakukan penilaian dan justifikasi permasalahan, diawali dengan analisis sistem yang sedang berjalan yaitu analisis kondisi faktual saat ini sebagaimana

yang telah dipaparkan pada Bab Pendahuluan. Langkah selanjutnya adalah membuat aplikasi deteksi dini penyakit yang diderita oleh perokok dengan mengimplementasikan *machine learning (ML)*. Berdasarkan kondisi di masyarakat, diperoleh beberapa fakta penting, antara lain:

1. Banyak penyakit yang berkaitan dengan aktivitas merokok, baik secara langsung maupun tidak langsung yang berdampak signifikan terhadap kesehatan individu.
2. Tingginya angka perokok di masyarakat, dengan kebiasaan merokok yang bahkan dimulai sejak usia dini menunjukkan tren yang mengkhawatirkan.
3. Sebagian perokok tidak menyadari bahwa dirinya sudah memiliki satu atau beberapa penyakit akibat kebiasaan merokok dan baru memeriksakan diri setelah kondisi kesehatannya cukup parah.

Dari fakta tersebut, terlihat jelas bahwa diperlukan suatu alat deteksi dini penyakit yang diderita oleh perokok. Alat ini diharapkan dapat membantu dalam mengidentifikasi penyakit sejak dini, sehingga memungkinkan penanganan yang lebih cepat dan efektif. Dengan demikian, upaya peningkatan kewaspadaan dan kesehatan masyarakat dapat ditingkatkan melalui teknologi ini.

Akuisisi Pengetahuan (Knowledge Acquisition)

Pada tahap ini, proses identifikasi jenis penyakit yang terkait dengan aktivitas merokok serta gejala yang dirasakan dilakukan. Berdasarkan penelitian yang sebelumnya telah dilakukan oleh Willy Welvi Nanda dan Cahyo Aji Widyasworo, ditemukan 10 jenis penyakit yang terkait dengan merokok serta gejala-gejala yang dapat dirasakan oleh seseorang yang menderita satu atau lebih dari penyakit tersebut. Proses ini melibatkan pengumpulan data dari berbagai sumber baik literatur ilmiah maupun studi kasus klinis.

Setelah data awal diperoleh, langkah berikutnya adalah konfirmasi dan validasi data yang sudah diperoleh. Data tersebut dikonfirmasi oleh seorang pakar di bidang penyakit dalam, yaitu Dr. Irene. Dr. Irene yang memiliki pengalaman luas dalam menangani berbagai penyakit akibat merokok memberikan wawasan yang sangat berharga. Dr. Irene meninjau daftar penyakit dan gejala yang telah dikumpulkan memastikan bahwa

informasi tersebut tidak hanya akurat tetapi juga relevan dengan kondisi klinis yang sering ditemui di lapangan. Dr. Irene juga memberikan masukan berdasarkan pengalaman klinisnya dan memperkaya data dengan referensi dari literatur terbaru. Proses ini membantu dalam menyempurnakan daftar gejala sehingga mencakup semua kemungkinan manifestasi yang mungkin terjadi pada penderita. Hasil dari tahap ini diperoleh data jenis penyakit dan gejala yang di rasakan seperti yang disajikan dalam tabel berikut ini.

Tabel 3.1 Penyakit dan Gejala

No	Penyakit	Gejala
1	Kanker Mulut	Merasakan nyeri saat menelan, mengalami sariawan, mengalami sariawan disertai darah, mengalami bercak merah dan putih di mulut, mengalami benjolan di dalam mulut, merasakan gigi goyang tanpa sebab
2	Kanker Paru-paru Tahap Awal	Mengalami batuk berdarah, merasakan napas terasa sesak, mengalami berat badan menurun, merasakan dada terasa nyeri, mengalami bunyi mengi seperti siulan saat bernapas
3	Kanker Esofagus Tahap Awal	Merasakan nyeri saat menelan, mengalami batuk berdarah, merasakan napas terasa sesak, mengalami berat badan menurun, merasakan tulang belakang nyeri, mengalami BAB berdarah
4	Kanker Lambung	Merasakan nyeri di uluh hati, mengalami naik asam lambung, mengalami cepat kenyang, merasakan tubuh terasa lemas, mengalami perut membengkak
5	Kanker Ginjal Tahap Awal	Mengalami berat badan menurun, merasakan nyeri saat buang air kecil, mengalami pucat dan mudah lelah, mengalami urine berdarah
6	Kanker Serviks	Selalu mengalami keluar urine saat fases, merasakan nyeri saat berhubungan seksual, mengalami pendarahan vagina yang tidak normal

7	Serangan Jantung Tahap Awal	Merasakan napas terasa sesak, mengalami berat badan menurun, mengalami keringat dingin, merasakan napas berat, merasakan detak jantung lebih berdebar
8	Randang Tenggorokan	Merasakan nyeri saat menelan, merasakan tenggorokan sakit dan gatal, mengalami suara serak, merasakan batuk kering/kronis.
9	Ateroklerosis Otak	Merasakan napas berat, merasakan mati rasa hingga lumpuh di bagian wajah atau tungkai, sering sekali merasakan kebingungan, mengalami sulit berbicara, merasakan sakit kepala (pusing)
10	Impotensi	Merasakan penis sulit untuk ereksi, merasakan tidak ada gairah untuk seksual, merasakan tidak bisa untuk ejakulasi, mengalami kesulitan kelimaks saat berhubungan seksual
11	Gas Lambung	Merasakan nyeri saat menelan, mengalami suara serak, merasakan batuk kering/kronis, merasakan dada terasa nyeri, merasakan mual, mengalami muntah, mengalami perut membengkak
12	Hipertensi	Merasakan napas terasa sesak, merasakan mual, mengalami muntah, sering sekali merasakan kebingungan, mengalami sulit berbicara, merasakan sakit kepala (pusing), mengalami pucat dan mudah lelah
13	Asma	Merasakan batuk kering/kronis, merasakan napas terasa sesak, mengalami bunyi mengi seperti siulan saat bernapas, mengalami sulit berbicara, merasakan sakit kepala (pusing), mengalami pucat dan mudah lelah

Tahap selanjutnya adalah pengumpulan data yang dibutuhkan untuk implementasi *Machine Learning*. Proses ini dilakukan dengan menyebarkan kuesioner kepada

responden sasaran menggunakan platform *Google Form*. Link kuesioner yang disebarikan adalah <https://forms.gle/R8eNGHFYMZz4KQ589> , dan dari penyebaran ini berhasil diperoleh 184 responden.

Pengumpulan data ini merupakan tahapan krusial karena data yang diperoleh akan dijadikan dasar untuk membangun model *Machine Learning* yang akurat dan reliabel. Setiap responden mengisi kuesioner yang terdiri dari berbagai pertanyaan yang dirancang untuk mengumpulkan informasi yang relevan dengan tujuan penelitian. Proses penyebaran kuesioner dilakukan secara cermat untuk memastikan bahwa data yang dikumpulkan representatif dan mencakup berbagai aspek yang dibutuhkan. Hasil kuesioner ini kemudian dikumpulkan dalam bentuk data mentah, yang akan melalui tahap pengolahan dan analisis lebih lanjut. Data mentah ini mencakup berbagai variabel yang akan dianalisis untuk mengidentifikasi pola dan tren yang dapat digunakan dalam model ML. Berikut ini adalah data mentah yang berhasil diperoleh dari responden, yang akan menjadi dasar pengembangan model selanjutnya.

The image shows a screenshot of a Microsoft Excel spreadsheet containing a dataset of 184 respondents. The spreadsheet has 48 columns and 184 rows. The columns represent various variables, including demographic information (e.g., gender, age, education) and health-related data (e.g., smoking status, alcohol consumption, chronic conditions like hypertension, diabetes, and cancer). The data is organized in a grid format, with each row representing an individual respondent's answers to the survey questions.

Gambar 3.21 Dataset Responden

Langkah selanjutnya adalah menganalisisnya menggunakan metode *EDA* (*Exploratory Data Analysis*) dan menerapkan model *Machine Learning*.

Dataset

Dataset yang terkumpul memiliki 48 atribut dan 1 target, namun hanya 46 atribut

yang digunakan dalam penelitian ini. Atribut atau fitur dalam penelitian ini dianggap sebagai variabel independen, sedangkan target dianggap sebagai variabel dependen. Metode yang diterapkan bertujuan untuk memanfaatkan variabel independen dalam memprediksi variabel dependen. Lebih spesifiknya, variabel independen dalam *dataset* ini mencakup berbagai informasi tentang responden, seperti usia, kebiasaan merokok, dan lainnya. Variabel dependen merupakan indikator apakah responden menderita suatu penyakit tertentu atau tidak. Beberapa atribut yang termasuk dalam *dataset* dan target yang menjadi fokus analisis penelitian dapat dilihat pada gambar berikut.

#	Column	Non-Null Count	Dtype
0	Jenis Kelamin	184 non-null	object
1	Usia Anda	184 non-null	int64
2	Apakah anda memiliki kebiasaan merokok ?	184 non-null	object
3	Rokok apa yang anda gunakan ?	184 non-null	object
4	Apakah orang di sekitar anda merokok ?	184 non-null	object
5	Sudah berapa lama anda merokok ?	184 non-null	object
6	Apakah anda pengidap penyakit di bawah ini? Jika iya silahkan pilih salah satu penyakit di bawah ini	184 non-null	object
7	Saya merasakan nyeri saat menelan	184 non-null	object
8	Saya merasakan tenggorokan sakit dan gatal	184 non-null	object
9	Saya mengalami suara serak	184 non-null	object
10	Saya merasakan batuk kering/kronis	184 non-null	object
11	Saya mengalami batuk berdarah	184 non-null	object
12	Saya merasakan napas terasa sesak	184 non-null	object
13	Saya mengalami berat badan menurun	184 non-null	object
14	Saya merasakan tulang belakang nyeri	184 non-null	object
15	Saya mengalami BAB berdarah	184 non-null	object
16	Saya mengalami sariawan	184 non-null	object
17	Saya mengalami sariawan disertai darah	184 non-null	object
18	Saya mengalami bercak merah dan putih di mulut	184 non-null	object
19	Saya mengalami benjolan di dalam mulut	184 non-null	object
20	Saya merasakan gigi goyang tanpa sebab	184 non-null	object
21	Saya merasakan dada terasa nyeri	184 non-null	object
22	Saya mengalami bunyi mengi seperti siulan saat bernapas	184 non-null	object
23	Saya mengalami keringat dingin	184 non-null	object
24	Saya merasakan napas berat	184 non-null	object
25	Saya merasakan detak jantung lebih berdebar	184 non-null	object
26	Saya merasakan mual	184 non-null	object
27	Saya mengalami muntah	184 non-null	object
28	Saya merasakan nyeri di uluh hati	184 non-null	object
29	Saya mengalami naik asam lambung	184 non-null	object
30	Saya mengalami cepat kenyang	184 non-null	object
31	Saya merasakan tubuh terasa lemas	184 non-null	object
32	Saya mengalami perut membengkak	184 non-null	object
33	Saya merasakan mati rasa hingga lumpuh di bagian wajah atau tungkai	184 non-null	object
34	Saya sering sekali merasakan kebingungan	184 non-null	object
35	Saya mengalami sulit berbicara	184 non-null	object
36	Saya merasakan sakit kepala (pusing)	184 non-null	object
37	Saya merasakan nyeri saat buang air kecil	184 non-null	object
38	Saya mengalami pucat dan mudah lelah	184 non-null	object
39	Saya mengalami urine berdarah	184 non-null	object
40	Saya merasakan penis sulit untuk ereksi	184 non-null	object
41	Saya merasakan tidak ada gairah untuk seksual	184 non-null	object
42	Saya merasakan tidak bisa untuk ejakulasi	184 non-null	object
43	Saya mengalami kesulitan kelimaks saat berhubungan seksual	184 non-null	object
44	Saya selalu mengalami keluar urine saat fases	184 non-null	object
45	Saya merasakan nyeri saat berhubungan seksual	184 non-null	object
46	Saya mengalami pendarahan vagina yang tidak normal	184 non-null	object

Gambar 3.3 Seluruh Kolom *Dataset*

Data dalam *dataset* ini mencakup informasi dari berbagai kolom yang terdiri dari target dan fitur. Target utama dari data ini adalah kolom "Apakah Anda menderita

penyakit di bawah ini? Jika ya, silakan pilih salah satu penyakit di bawah ini", yang berfungsi untuk mengidentifikasi apakah responden menderita suatu penyakit tertentu. Selain kolom target, terdapat beberapa fitur yang mencakup data demografi, kebiasaan, dan kondisi kesehatan umum responden. Fitur-fitur tersebut meliputi jenis kelamin, usia, kebiasaan merokok, jenis rokok yang digunakan, dan informasi tambahan lainnya. Data dalam fitur ini digunakan untuk memberikan gambaran yang lebih lengkap tentang profil dan kondisi kesehatan responden.

Preprocessing

Dalam melakukan analisis data diperlukan beberapa tahap *preprocessing* untuk memastikan data lebih optimal saat digunakan. Tahap-tahap *preprocessing* ini meliputi :

Menghapus Kolom 1 dan 2

Dua kolom yang dianggap kurang penting untuk analisis dihilangkan. Kolom pertama berisi jawaban responden mengenai kesediaan mereka untuk mengikuti survei, sedangkan kolom kedua meminta nomor telepon untuk tujuan undian berhadiah. Penghapusan dilakukan karena informasi tersebut tidak relevan untuk analisis dan untuk melindungi privasi responden.

```
columns_to_drop = [  
    'Saya bersedia menjadi responden ini',  
    'Mohon cantumkan nomor telepon yang dapat dihubungi  
(dibutuhkan juga untuk undian berupa saldo OVO/Gopay)'  
]  
df2 = df.drop(columns=columns_to_drop)
```

Dengan demikian, data yang diolah akan lebih terfokus pada variabel yang relevan dan analisis akan lebih akurat tanpa adanya gangguan dari data yang tidak diperlukan. Langkah ini juga memastikan bahwa privasi responden tetap terlindungi, sesuai dengan kebijakan perlindungan data yang berlaku.

Mengganti Teks Kolom Indeks 24

Selanjutnya, ditemukan kolom yang terkadang memuat kata 'iya' dengan huruf kecil. Karena menginginkan keseragaman dengan huruf kapital di awal, kata 'iya' diganti menjadi 'Iya' di seluruh kolom. Perubahan ini dilakukan untuk memastikan konsistensi penulisan data, sehingga memudahkan analisis di kemudian hari. Standardisasi ini juga membantu menghindari kesalahan interpretasi data dan meningkatkan akurasi hasil analisis. Dengan data yang konsisten, proses pengolahan dan pelaporan menjadi lebih efisien dan andal.

```
df2.iloc[:, 24] = df2.iloc[:, 24].str.replace('iya', 'Iya')
```

Mempangkas Teks Kolom Indeks 6

Masih ada satu tugas lagi yang harus diselesaikan. Di kolom lain, ada beberapa entri dengan spasi tambahan di awal atau akhir teks. Seperti ada entri yang ditulis sebagai ' Tidak ada' dengan spasi di awal dan akhir. Untuk merapikan data spasi tambahan dihilangkan sehingga teks menjadi 'Tidak ada'. Meski tampak sederhana, langkah ini dapat membuat perbedaan yang signifikan dalam analisis data nanti. Dengan data yang bersih dan konsisten, akurasi dan keandalan hasil analisis akan lebih terjamin.

```
df2.iloc[:, 6] = df2.iloc[:, 6].str.strip()
```

Pengecekan Missing Value

Sebelum memulai pemrosesan model *machine learning*, langkah pertama yang sangat penting adalah melakukan verifikasi terhadap *dataset* yang akan digunakan. Proses ini bertujuan untuk memastikan bahwa *dataset* terbebas dari data kosong atau *missing value*. Data yang tidak lengkap dapat mengakibatkan hasil model menjadi tidak akurat dan bias. Oleh karena itu, setiap kolom dan baris pada *dataset* harus dicek untuk memastikan kelengkapan isinya. Jika ditemukan data kosong, maka harus dilakukan tindakan seperti mengisi *missing value* dengan metode tertentu atau menghapus baris atau kolom yang bermasalah. Dengan melakukan verifikasi tersebut, maka dapat dipastikan bahwa *dataset* dalam

kondisi bersih dan siap digunakan untuk melatih model *machine learning* dengan akurasi yang lebih tinggi.

```
missing_values = df2.isnull().sum()
print(missing_values)
```

```
Jenis Kelamin 0
Usia Anda 0
Apakah anda memiliki kebiasaan merokok ? 0
Rokok apa yang anda gunakan ? 0
Apakah orang di sekitar anda merokok ? 0
Sudah berapa lama anda merokok ? 0
Apakah anda pengidap penyakit di bawah ini? Jika iya silahkan pilih salah satu penyakit di bawah ini 0
Saya merasakan nyeri saat menelan 0
Saya merasakan tenggorokan sakit dan gatal 0
Saya mengalami suara serak 0
Saya merasakan batuk kering/kronis 0
Saya mengalami batuk berdarah 0
Saya merasakan napas terasa sesak 0
Saya mengalami berat badan menurun 0
Saya merasakan tulang belakang nyeri 0
Saya mengalami BAB berdarah 0
Saya mengalami sariawan 0
Saya mengalami sariawan disertai darah 0
Saya mengalami bercak merah dan putih di mulut 0
Saya mengalami benjolan di dalam mulut 0
Saya merasakan gigi goyang tanpa sebab 0
Saya merasakan dada terasa nyeri 0
Saya mengalami bunyi mengi seperti siulan saat bernapas 0
Saya mengalami keringat dingin 0
Saya merasakan napas berat 0
Saya merasakan detak jantung lebih berdebar 0
Saya merasakan mual 0
Saya mengalami muntah 0
Saya merasakan nyeri di uluh hati 0
Saya mengalami naik asam lambung 0
Saya mengalami cepat kenyang 0
Saya merasakan tubuh terasa lemas 0
Saya mengalami perut membengkak 0
Saya merasakan mati rasa hingga lumpuh di bagian wajah atau tungkai 0
Saya sering sekali merasakan kebingungan 0
Saya mengalami sulit berbicara 0
Saya merasakan sakit kepala (pusing) 0
Saya merasakan nyeri saat buang air kecil 0
Saya mengalami pucat dan mudah lelah 0
Saya mengalami urine berdarah 0
Saya merasakan penis sulit untuk ereksi 0
Saya merasakan tidak ada gairah untuk seksual 0
Saya merasakan tidak bisa untuk ejakulasi 0
Saya mengalami kesulitan kelimaks saat berhubungan seksual 0
Saya selalu mengalami keluar urine saat fases 0
Saya merasakan nyeri saat berhubungan seksual 0
Saya mengalami pendarahan vagina yang tidak normal 0
dtype: int64
```

Gambar 3.4 Data *Missing Value*

Menentukan X Fitur dan Y Target

Dataset kemudian dibagi menjadi dua bagian utama yaitu fitur (X) dan target

(y). Fitur terdiri dari variabel independen yang digunakan untuk memprediksi target, sedangkan target merupakan variabel dependen yang menjadi fokus prediksi. Selanjutnya *dataset* dipisahkan menjadi dua bagian kolom 0-6 digunakan sebagai fitur, dan kolom 7 hingga akhir sebagai fitur tambahan. Kolom ke-6 dipisahkan khusus sebagai target. Pembagian ini dilakukan untuk memudahkan proses pelatihan model dan memudahkan analisis hubungan antara variabel independen dan dependen.

```
part1 = df2.iloc[:, 0:6]
part2 = df2.iloc[:, 7:]
X = pd.concat([part1, part2], axis=1)

y = df2.iloc[:, 6]
y = y.to_frame(y.name)
```

	Jenis Kelamin Anda	Usia	Apakah anda memiliki kebiasaan merokok ?	Rokok apa yang anda gunakan ?	Apakah orang di sekitar anda merokok ?	Sudah berapa lama anda merokok ?	Saya merasakan nyeri saat menelan	Saya merasakan tenggorokan sakit dan gatal	Saya mengalami suara serak	Saya merasakan batuk kering/ kronis	...	Saya merasakan nyeri saat buang air kecil	Saya mengalami pucat dan mudah lelah	Saya mengalami urine berdarah	Saya merasakan penis sulit untuk ereksi
0	Perempuan	23	Tidak	Tidak Merokok	Tidak	0 Tahun	Tidak	Tidak	Tidak	Tidak	...	Tidak	Tidak	Tidak	Tidak
1	Laki-laki	35	Tidak	Tidak Merokok	Iya	0 Tahun	Iya	Iya	Iya	Iya	...	Tidak	Iya	Iya	Iya
2	Laki-laki	67	Iya	Tembakau	Iya	> 7 Tahun	Iya	Tidak	Iya	Iya	...	Tidak	Tidak	Tidak	Tidak
3	Laki-laki	21	Iya	Elektrik	Tidak	0 Tahun	Tidak	Tidak	Tidak	Tidak	...	Tidak	Tidak	Tidak	Tidak
4	Laki-laki	50	Iya	Tembakau	Iya	< 7 Tahun	Iya	Tidak	Iya	Iya	...	Tidak	Tidak	Tidak	Tidak

5 rows x 46 columns

Gambar 3.5 Data X

Apakah anda pengidap penyakit di bawah ini? Jika iya silahkan pilih salah satu penyakit di bawah ini	
0	Tidak ada
1	Tidak ada
2	Gas lambung
3	Tidak ada
4	Gas lambung

Gambar 3.6 Data Y

Label Encoding

Agar data dapat diproses oleh algoritma *machine learning*, variabel kategoris

perlu diubah ke dalam format numerik melalui proses pengodean. Pertama, salinan data fitur dan target dibuat untuk dikodekan. Selanjutnya setiap kolom dalam data fitur yang bertipe 'objek' (kategorikal) diubah ke dalam format numerik menggunakan *LabelEncoder*. Demikian pula target diubah ke dalam format numerik. Informasi tentang *encoder* disimpan menggunakan *joblib* sehingga dapat digunakan kembali jika diperlukan. Proses ini memastikan bahwa data siap dianalisis dan diproses oleh algoritma *machine learning*.

```
from sklearn.preprocessing import LabelEncoder
import joblib

X_encode = X.copy()
y_encode = y.copy()

label_encoders_X = {}
for col in X_encode.columns:
    if X_encode[col].dtype == 'object': # Check if dtype is
'object' (non-numeric)
        le = LabelEncoder()
        X_encode[col] = le.fit_transform(X_encode[col])
        label_encoders_X[col] = le

le_y = LabelEncoder()
col = y.columns[0]
y_encode[col] = le_y.fit_transform(y[col])

le = LabelEncoder()
le.fit(y[col])

joblib.dump(label_encoders_X, 'label_encoders_X.pkl')
joblib.dump(le, 'label_encoders_y.pkl')
```

Splitting Data

Pembagian data pada sintaks berikut ini bertujuan untuk memisahkan *dataset*

menjadi dua subset yang berbeda, yaitu *training dataset* dan *testing dataset*. Pembagian dilakukan dengan perbandingan 80:20, di mana 80% data digunakan untuk melatih model (*training dataset*) dan 20% sisanya digunakan untuk menguji model (*testing dataset*).

Proses pembagian data ini dilakukan dengan menggunakan fungsi `train_test_split` dari library `sklearn.model_selection`. Fungsi ini memungkinkan pembagian data secara acak menjadi *training* dan *testing subset*, sehingga performa model dapat diuji pada data yang sebelumnya tidak terlihat.

```
from sklearn.model_selection import train_test_split
y_numpy = y_encode[y.columns[0]].to_numpy()

X_train, X_test, y_train, y_test = train_test_split(X_encode,
y_numpy, test_size=0.2, random_state=46)
```

Modelling

Implementasi Model

Pada tahap ini, model *Random Forest Classifier* diterapkan untuk melakukan prediksi terhadap data uji (`x_test`). Model ini telah melalui proses pelatihan sebelumnya dengan menggunakan data latih yang terdiri dari berbagai fitur yang dianggap relevan. Tujuan penerapan model ini yaitu untuk memprediksi nilai-nilai pada data uji berdasarkan pola-pola yang telah dipelajari dan dipahami oleh model selama proses pelatihan. Prediksi ini membantu dalam memperkirakan hasil-hasil yang mungkin muncul pada data uji yang belum pernah digunakan dalam pelatihan model.

Proses prediksi ini merupakan langkah penting dalam pengujian model, karena memungkinkan adanya evaluasi seberapa baik model dapat menggeneralisasi pengetahuannya ke data-data baru yang belum pernah terlihat. Hasil prediksi ini kemudian disimpan dalam variabel `y_pred`, yang selanjutnya digunakan untuk analisis dan evaluasi lebih lanjut. Analisis ini meliputi pengecekan keakuratan prediksi, identifikasi kesalahan atau anomali, dan

penilaian kinerja model secara keseluruhan.

```
# Modelling dan K-Fold Cross Validation
from sklearn.ensemble import RandomForestClassifier

# Tentukan model dengan parameter yang telah disesuaikan
rfc = RandomForestClassifier(
    n_estimators=100,
    criterion='gini',
    max_depth=None,
    min_samples_split=5,
    min_samples_leaf=1,
    max_features=None,
    random_state=46,
)

# Melatih model
rfc.fit(X_train, y_train)

# Memprediksi model
y_pred = rfc.predict(X_test)
```

Akurasi Model

Setelah model dibangun menggunakan algoritma *Random Forest*, langkah selanjutnya adalah memeriksa keakuratan model untuk mengetahui seberapa baik kinerjanya. Pemeriksaan keakuratan dilakukan dengan membandingkan hasil prediksi model dengan data uji yang telah disiapkan. Dari hasil perhitungan, model *Random Forest* ini berhasil mengklasifikasikan 94,12% data uji dengan benar. Hal ini menunjukkan bahwa model memiliki kinerja yang sangat baik dalam menangani tugas-tugas klasifikasi dan *dataset* yang diberikan. Akurasi model dihitung dengan menggunakan fungsi 'accuracy_score', yang mengambil dua parameter `y_test` dan `y_pred`.

```
from sklearn.metrics import accuracy_score

accuracy = accuracy_score(y_test, y_pred)
print(f"Accuracy: {accuracy}")
```

```
Accuracy: 0.9411764705882353
```

Evaluasi Model

Tahap selanjutnya adalah evaluasi model. Dalam evaluasi model ini, ada beberapa langkah yang harus dilakukan untuk memastikan kinerja model yang akurat. Langkah-langkah yang dimaksud meliputi *K-Fold Cross Validation*, *Confusion Matrix*, dan *Classification Report*. Berikut ini adalah proses tahapannya.

K-Fold Cross Validation

Cross Validation dapat membantu menentukan apakah model *overfitting* atau tidak. Metode *cross validation* yang digunakan dalam kasus ini adalah *K-Fold Cross Validation* dengan $k = 5$.

```
from sklearn.model_selection import KFold, cross_val_score

# Menentukan K-Fold Cross Validation
kf = KFold(n_splits=5, shuffle=True, random_state=42)

# Melakukan K-Fold Cross Validation
scores = cross_val_score(rfc, X_encode, y_numpy, cv=kf)
print("Accuracy with cross-validation: %.2f with standard
deviation %.2f" % (scores.mean(), scores.std()))

print(f"K-Fold Cross Validation Results: {scores}")
print(f"Mean Accuracy: {np.mean(scores)}")
```

```
Accuracy with cross-validation: 0.91 with standard deviation
0.05
K-Fold Cross Validation Results: [0.94117647 0.85294118
0.85294118 0.96969697 0.93939394]
Mean Accuracy: 0.9112299465240643
```

Nilai akurasi dengan *cross validation* sebesar 0,91 dengan simpangan baku 0,05 menunjukkan bahwa model mencapai akurasi yang sangat baik pada data yang digunakan untuk pelatihan model. Nilai akurasi ini merupakan hasil rata-rata dari lima lipatan berbeda dalam *K-Fold Cross Validation*, dengan setiap lipatan memberikan hasil yang sedikit berbeda. Hasil ini menunjukkan bahwa model tidak hanya bekerja dengan baik pada data pelatihan tetapi juga menunjukkan konsistensi di berbagai subset data.

Confusion Matrix

Dalam upaya untuk memahami kinerja model *Random Forest* yang telah dibangun, dilakukan evaluasi dengan menggunakan *confusion matrix*. Fungsi 'confusion_matrix' diimpor dari *library sklearn.metrics* untuk menghitung *confusion matrix*. Data label aktual (*y_test*) dan hasil prediksi model (*y_pred*) digunakan dalam perhitungan ini. Hasil *confusion matrix* yang diperoleh adalah sebagai berikut.

```
# Menentukan akurasi dengan confusion matrix
from sklearn.metrics import confusion_matrix

cm = confusion_matrix(y_test, y_pred)
print(cm)
```

```

[[ 1  0  0  0  0  0  0  0  0]
 [ 0  1  0  0  0  0  0  0  0]
 [ 0  0  1  0  0  0  0  0  0]
 [ 0  0  0  1  0  0  0  0  0]
 [ 0  0  0  0  1  0  0  0  0]
 [ 0  0  0  0  0  4  0  0  0]
 [ 0  0  0  0  0  0  1  0  0]
 [ 0  0  0  0  0  0  0  1  0]
 [ 0  0  0  1  0  0  0  1 21]]

```

Nilai-nilai pada diagonal utama menunjukkan jumlah prediksi yang benar untuk setiap kelas. Dengan contoh kelas pertama memiliki 1 prediksi yang benar, kelas kedua memiliki 1 prediksi yang benar, kelas keempat memiliki 4 prediksi yang benar, dan seterusnya. Hal ini menunjukkan bahwa model cukup baik dalam memprediksi kelas-kelas tertentu. Dan nilai-nilai di luar diagonal utama menunjukkan jumlah prediksi yang salah. Dengan contoh ada satu kasus di mana kelas ketujuh diprediksi secara salah sebagai kelas kedelapan. Hal ini menunjukkan area yang perlu ditingkatkan dalam model.

Confusion matrix ini memberikan wawasan berharga tentang kinerja model *Random Forest* yang telah dibuat. Dari hasil ini, dapat dilihat bahwa beberapa kelas memiliki prediksi yang sangat akurat sementara beberapa kelas lainnya mengalami kesalahan prediksi. Informasi ini sangat berguna untuk langkah selanjutnya dalam meningkatkan akurasi model. Dengan memahami lokasi kesalahan prediksi, penyesuaian dan perbaikan yang diperlukan pada model dapat dilakukan di masa mendatang.

Classification Report

Setelah model *Random Forest* dilatih, langkah selanjutnya yaitu menggunakan *classification report* dari *sklearn.metrics*. Prediksi dilakukan pada data pengujian, dan laporan yang dihasilkan mencakup *presisi*, *recall*, *f1-score*, dan *support* untuk setiap kelas. Hasil *classification report* menunjukkan bahwa akurasi model mencapai 94%, dengan sebagian besar kelas memiliki nilai *presisi*, *recall*, dan *f1-score* yang tinggi. Kelas dengan data yang lebih sedikit seperti kelas

4 dan 9 menunjukkan kinerja yang sedikit lebih rendah, tetapi secara keseluruhan, model tersebut berkinerja sangat baik.

```
from sklearn.metrics import classification_report

report = classification_report(y_test, y_pred,
                              zero_division=0)
print(report)
```

	precision	recall	f1-score	support
0	1.00	1.00	1.00	1
1	1.00	1.00	1.00	1
2	1.00	1.00	1.00	1
4	0.50	1.00	0.67	1
5	1.00	1.00	1.00	1
7	1.00	1.00	1.00	4
8	1.00	1.00	1.00	1
9	0.50	1.00	0.67	1
11	1.00	0.91	0.95	23
accuracy			0.94	34
macro avg	0.89	0.99	0.92	34
weighted avg	0.97	0.94	0.95	34

Dengan *presisi*, *recall*, dan *f1-score* yang sempurna untuk beberapa kelas, serta akurasi keseluruhan yang tinggi, efektivitas model dalam mengklasifikasikan data telah terbukti. Analisis ini mengonfirmasi kesiapan model *Random Forest* yang dilatih untuk digunakan dalam aplikasi nyata, meskipun masih ada ruang untuk perbaikan lebih lanjut dalam menangani kelas dengan data yang tidak seimbang.

Tahap selanjutnya dalam metodologi *ESDLC* adalah tahap Perancangan Aplikasi, Pengujian, Dokumentasi dan Pemeliharaan. Semua tahap ini akan diuraikan dalam BAB selanjutnya, yaitu BAB Analisis dan Perancangan, Pengujian dan Implementasi Sistem.

BAB 4

ANALISIS PERANCANGAN DAN IMPLEMENTASI SISTEM

Analisis Sistem

Pada tahap ini, sistem pakar dianalisis secara menyeluruh untuk mengidentifikasi kebutuhan dan fungsionalitas yang diperlukan oleh pengguna dan bisnis. Proses analisis ini dilakukan untuk memastikan bahwa sistem yang dirancang akan dapat memenuhi semua persyaratan yang telah ditentukan.

Analisis Fungsional

Analisis fungsional dari sistem telah dilakukan untuk memahami kebutuhan dan karakteristik setiap halaman serta fitur yang ada. Setiap elemen dalam sistem ini dirancang dengan tujuan memberikan pengalaman pengguna yang optimal dan memastikan bahwa semua data yang dihasilkan dapat dikelola dengan baik oleh pihak *admin*. Berikut adalah tabel analisis fungsional ada di dalam sistem:

Tabel 4.1 Analisis Fungsional

No	Fitur	Deskripsi Fungsional	Pelaku Kegiatan
1	Daftar	Proses pendaftaran disediakan agar pengguna baru dapat membuat akun dengan mengikuti langkah-langkah yang telah di berikan.	User
2	Login	Fitur login diimplementasikan untuk mengamankan akses ke sistem. Pengguna diminta untuk memasukkan kredensial sebelum mengakses layanan.	User
3	Tentang	Informasi mengenai aplikasi dan pengembang, memungkinkan pengguna untuk mempelajari lebih lanjut tentang fitur aplikasi.	User
4	Beranda	Halaman beranda dirancang untuk	User

	(User)	menyajikan informasi utama kepada pengguna. Akses ke fitur-fitur utama difasilitasi.	
5	Diagnosa (User)	Proses diagnosa disederhanakan untuk memudahkan pengguna dalam mendapatkan hasil diagnosa melalui instruksi yang disediakan.	User
6	Riwayat (User)	Riwayat diagnosa disimpan dan dapat diakses oleh pengguna kapan saja, dengan data disusun secara kronologis.	User
7	Home (Admin)	Halaman utama untuk admin dirancang untuk memberikan akses cepat ke berbagai fungsi administrasi dengan rangkuman data yang relevan.	Admin
8	Data Pengguna (Admin)	Data pengguna diatur dan dikelola dalam tabel yang terstruktur untuk memudahkan pengelolaan dan pemantauan.	Admin
9	Data Riwayat Diagnosa (Admin)	Riwayat diagnosa disusun dalam format yang mudah diakses oleh admin.	Admin
10	Random Forest (Admin)	Algoritma Random Forest diimplementasikan untuk analisis data, digunakan oleh admin untuk menghasilkan prediksi yang akurat.	Admin
11	Ubah Password (Admin)	Fitur untuk mengubah kata sandi disediakan agar admin dapat menjaga keamanan akun mereka sesuai kebutuhan.	Admin

Tabel ini menyajikan rincian fungsional dari setiap halaman dan fitur yang ada, termasuk deskripsi kegiatan dan siapa yang terlibat dalam pelaksanaannya.

Analisis Kebutuhan

Kebutuhan sistem dianalisis untuk memastikan bahwa semua persyaratan pengguna dan bisnis telah diidentifikasi dengan jelas dan terdokumentasi dengan baik. Kebutuhan ini mencakup kebutuhan fungsional seperti kemampuan sistem untuk memberikan rekomendasi berdasarkan data yang diinput oleh pengguna, serta kebutuhan non-fungsional seperti kecepatan respon, keamanan data, dan antarmuka yang ramah pengguna. Berikut adalah kebutuhan sistem selama penelitian ini dilakukan.

1. *Dektop dengan spesifikasi 12th Gen Intel® Core™ i712700F 4.90 GHz, Memori RAM 32 GB, VGA RTX 3070 Super, Hardisk 2TB, Sistem Operasi Windows 11*
2. *Microsoft Office*
3. *Google Form*
4. *Python*
5. *Jupyter Notebook*
6. *Visual Studio Code*
7. *Browser (seperti Google Chrome, Mozilla Firefox, dll)*

Perancangan Sistem

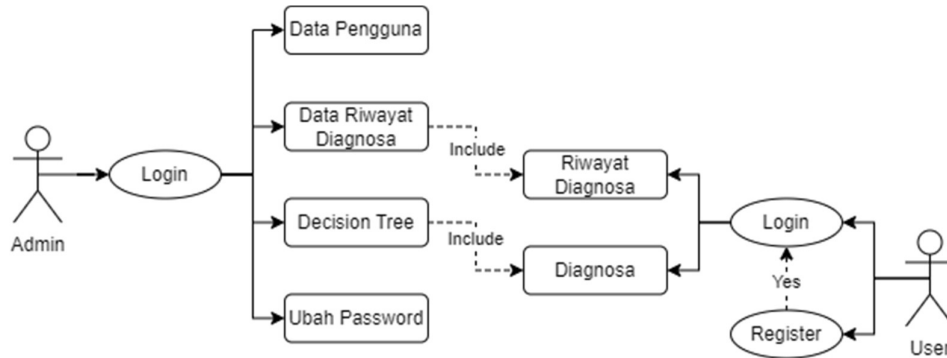
Pada tahap ini, sistem pakar dirancang untuk memenuhi kebutuhan yang telah diidentifikasi pada tahap analisis. Perancangan ini melibatkan pembuatan *blueprint* atau kerangka kerja untuk sistem yang akan dikembangkan, termasuk desain arsitektur sistem dan komponen-komponen yang akan digunakan. Berikut adalah tahapan yang dilakukan.

Alur Sistem Flowchart

Alur sistem divisualisasikan menggunakan *flowchart* untuk menggambarkan proses dan alur kerja sistem yang diusulkan. *Flowchart* ini membantu dalam memvisualisasikan bagaimana data akan mengalir melalui sistem pakar dan bagaimana berbagai komponen sistem akan berinteraksi satu sama lain. Berikut adalah diagram-diagram yang digunakan selama proses penelitian.

Use Case Diagram

Seperti yang terlihat pada gambar berikut, *admin* melakukan login aplikasi untuk masuk kedalam sistem, setelah berhasil login *admin* dapat mengelola data identifikasi, mengelola data diagnosa, mengelola data aturan, mengelola data konsultasi, dan mengelola data pengguna. Sedangkan pengguna hanya bisa melakukan konsultasi.

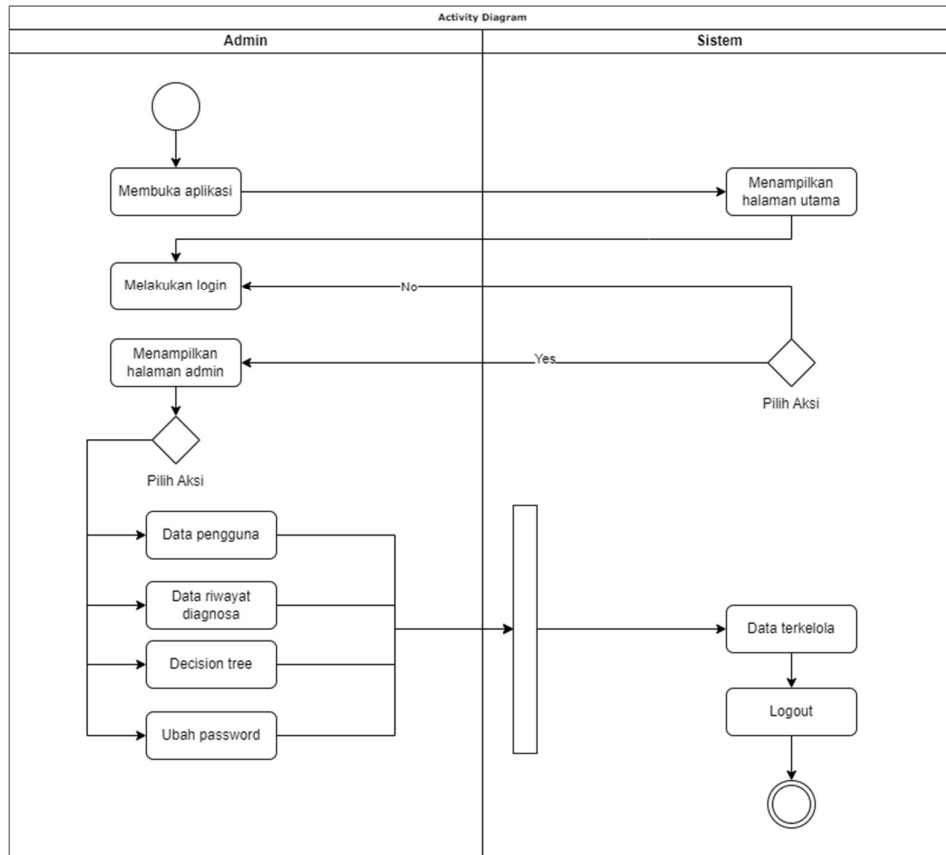


Gambar 4.1 Use Case Diagram

Activity Diagram

1. Activity Diagram Admin

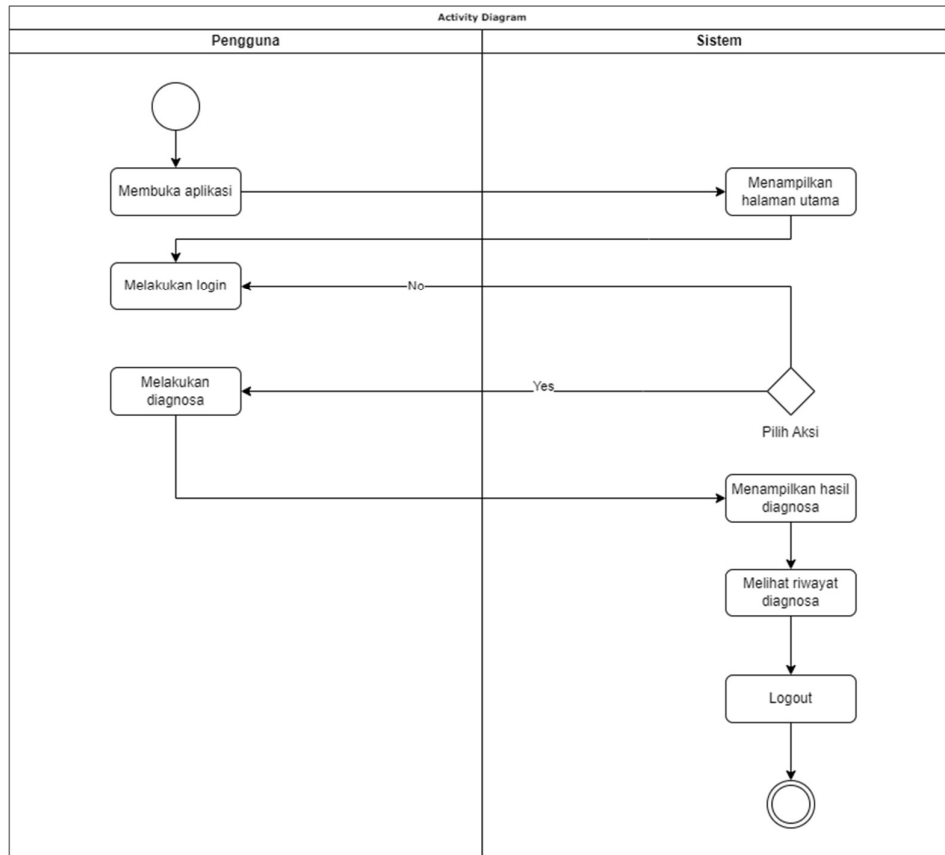
Seperti yang terlihat pada gambar berikut, *admin* melakukan proses login, jika data login benar maka *admin* dapat masuk kedalam sistem dan dipindahkan kehalaman utama *admin* jika salah maka *admin* harus login ulang, setelah *admin* berhasil login maka admin bisa mengelola data yang ada.



Gambar 4.2 Activity Diagram Admin

2. Activity Diagram Pengguna

Seperti yang terlihat pada gambar berikut, pengguna melakukan konsultasi dengan memilih data-data yang sesuai.

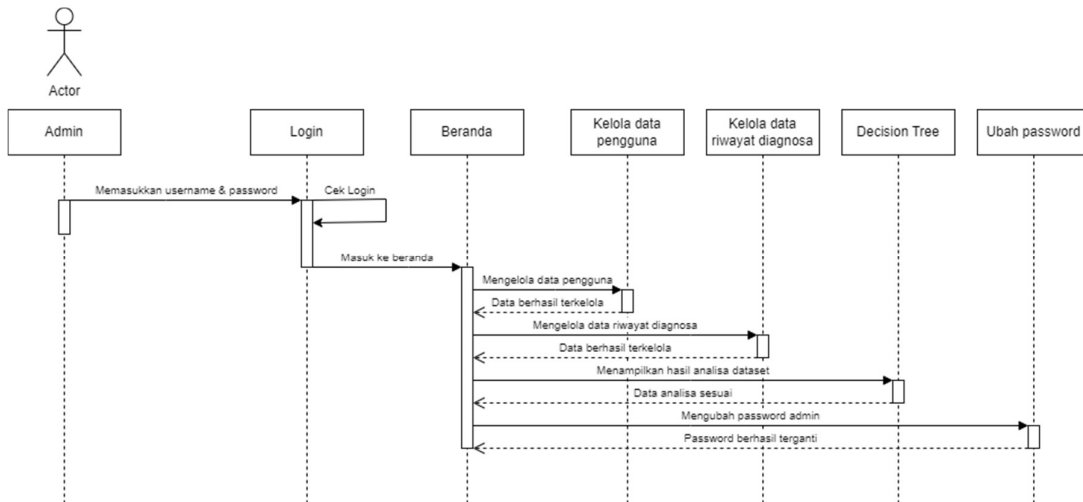


Gambar 4.3 Activity Diagram pengguna

Sequence Diagram

1. Sequence Diagram Admin

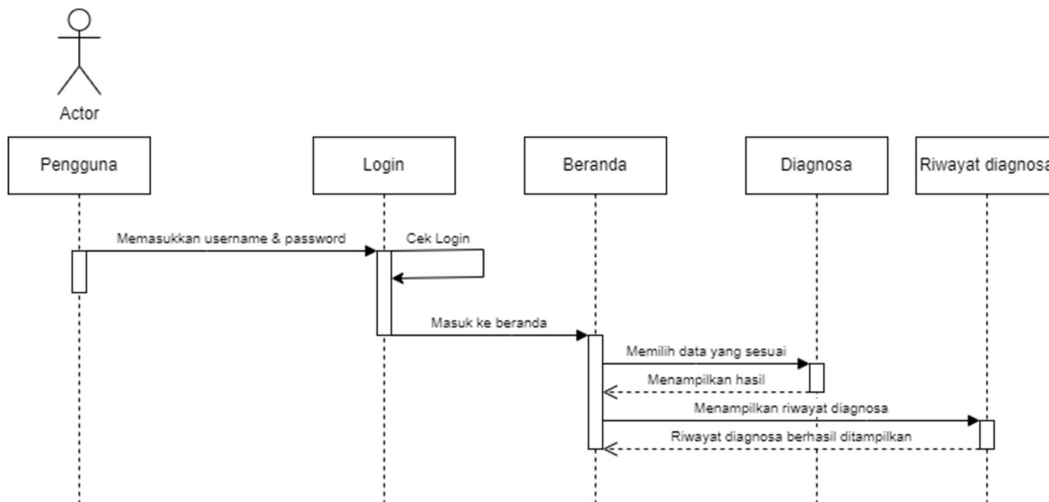
Pada gambar berikut, *admin* melakukan login terlebih dahulu sebelum mengakses menu-menu yang ada di dalam sistem. *Admin* dapat menambah, merubah dan menghapus data-data yang ada.



Gambar 4.4 Sequence Diagram Admin

2. Sequence Diagram Pengguna

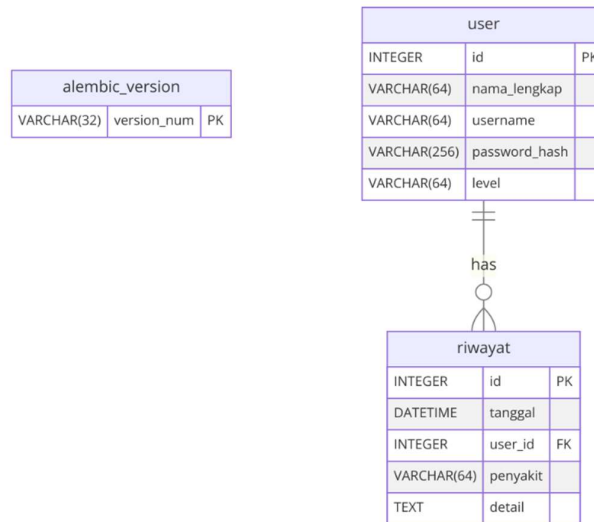
Pada gambar berikut, pengguna melakukan proses konsultasi, setelah konsultasi maka akan muncul hasilnya.



Gambar 4.5 Sequence Diagram Pengguna

Class Diagram

Seperti pada gambar berikut, merupakan *class diagram* dimana setiap table memiliki hubungan dengan table yang lain yang saling berkaitan untuk sistem nantinya.

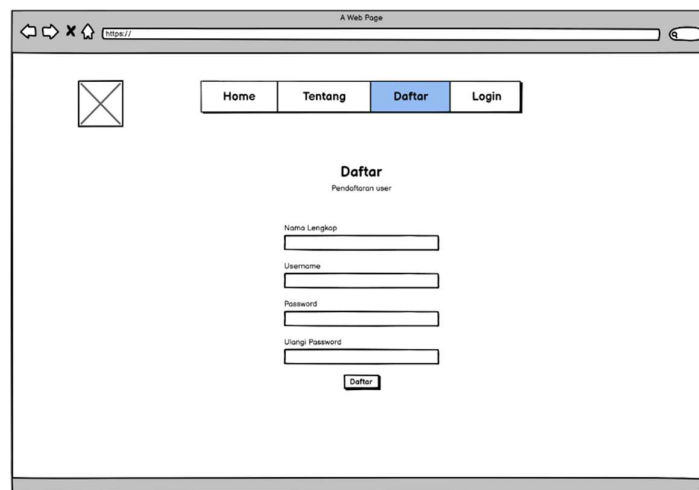


Gambar 4.6 *Class Diagram*

Perancangan UI

Tampilan Daftar

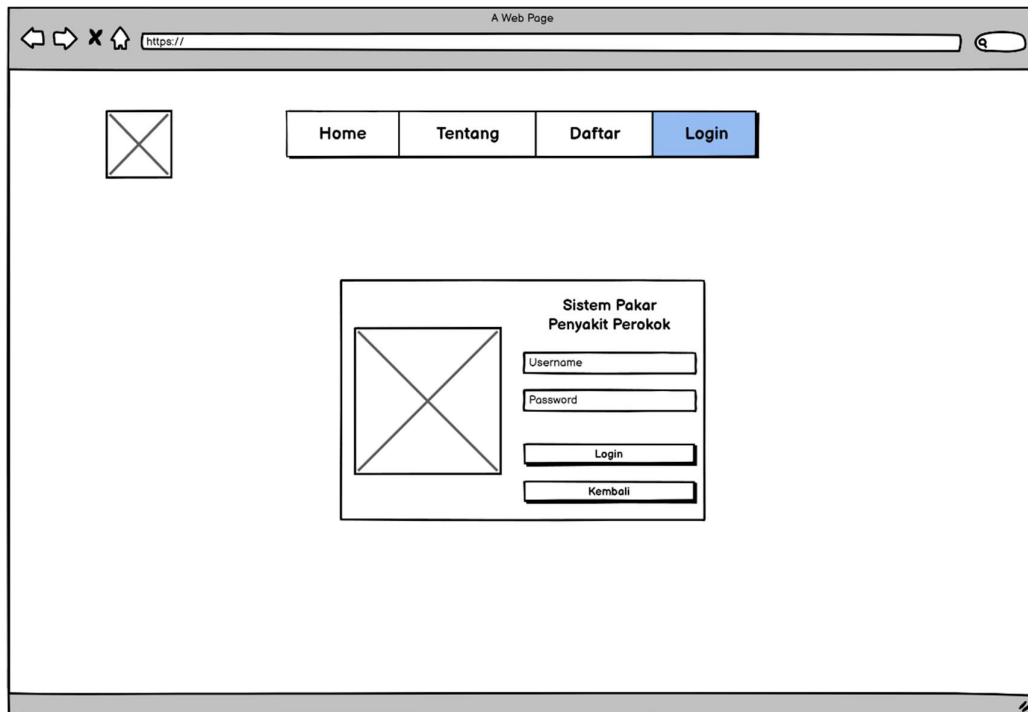
Pada gambar berikut, ditampilkan rancangan tampilan daftar yang akan dibuat dan diimplementasikan dalam sistem.



Gambar 4.7 Tampilan Daftar

Tampilan Login

Pada gambar berikut, ditampilkan rancangan tampilan login yang akan dibuat dan diimplementasikan dalam sistem. Rancangan ini dirancang untuk memastikan bahwa proses autentikasi pengguna dilakukan dengan aman dan efisien, sehingga pengguna dapat mengakses sistem dengan mudah.

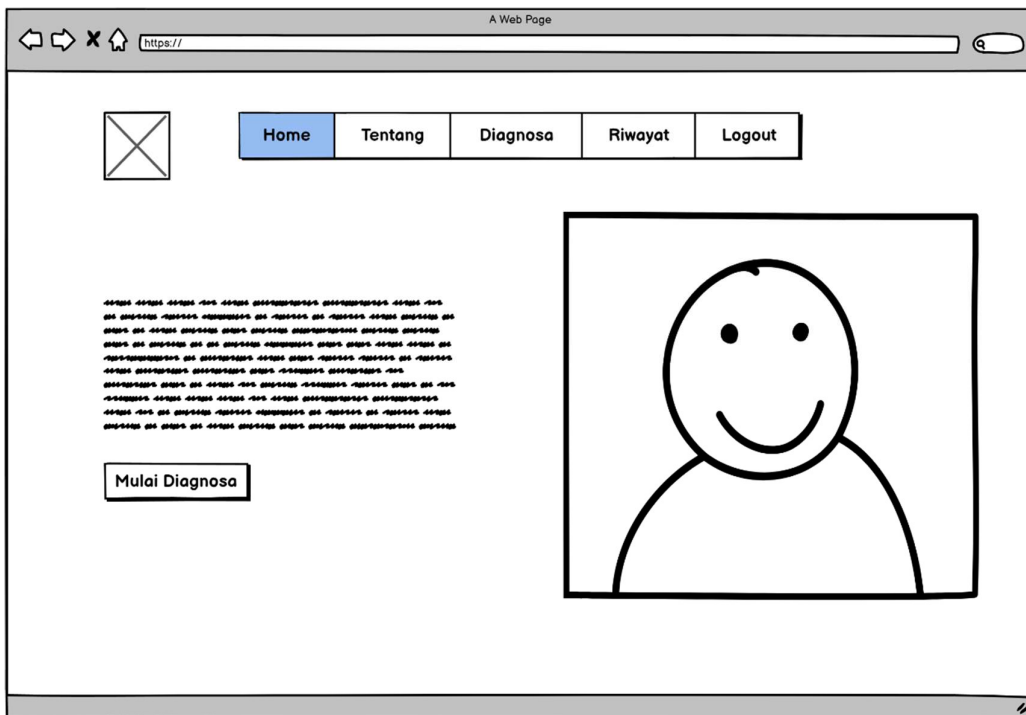


The image shows a wireframe of a web browser window. The browser's address bar contains 'https://'. The page layout includes a navigation menu with four buttons: 'Home', 'Tentang', 'Daftar', and 'Login'. The 'Login' button is highlighted in blue. Below the navigation menu is a central login form titled 'Sistem Pakar Penyakit Perokok'. The form contains a placeholder image (a square with an 'X'), a 'Username' input field, a 'Password' input field, a 'Login' button, and a 'Kembali' button.

Gambar 4.8 Tampilan *Login*

Tampilan Halaman Utama Pengguna

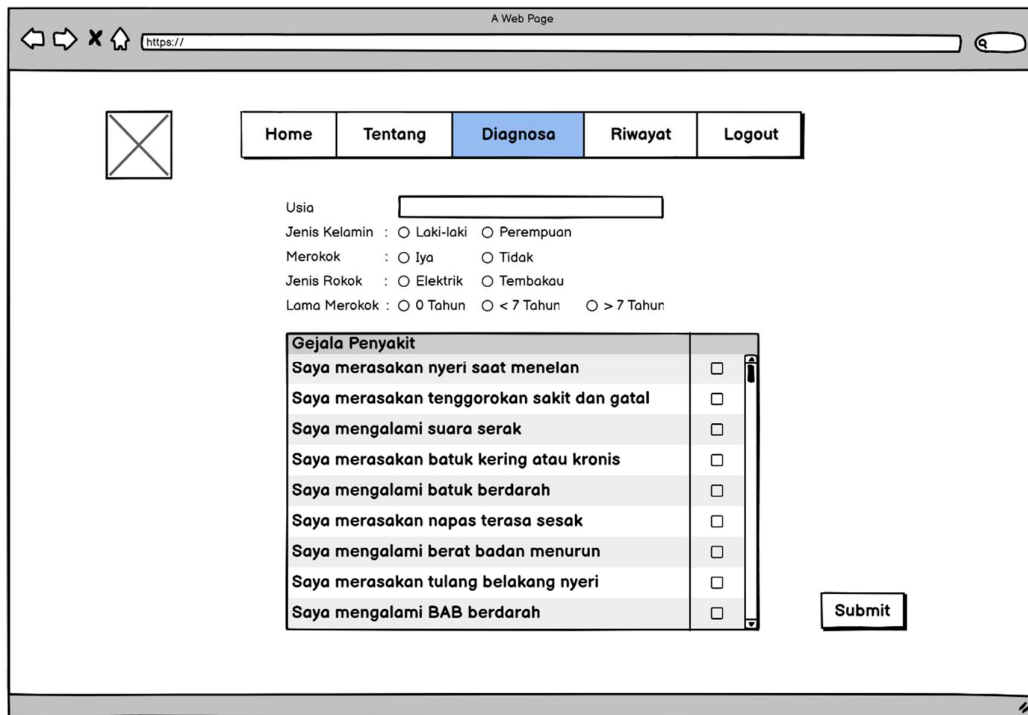
Pada gambar berikut, ditampilkan rancangan tampilan utama pengguna yang akan dibuat dan diimplementasikan dalam sistem. Rancangan ini disusun untuk memastikan bahwa pengguna dapat mengakses fitur-fitur utama dengan mudah dan navigasi yang intuitif. Setiap elemen pada tampilan utama diatur sedemikian rupa agar informasi yang dibutuhkan oleh pengguna dapat disajikan dengan jelas dan efisien. Tata letak dan desain visual telah dipertimbangkan secara matang untuk memberikan pengalaman pengguna yang optimal.



Gambar 4.9 Tampilan Utama Pengguna

Tampilan Halaman Diagnosa Pengguna

Pada gambar berikut, ditampilkan rancangan tampilan diagnosa untuk pengguna yang akan dibuat dan diimplementasikan dalam sistem. Rancangan ini dirancang untuk memudahkan pengguna dalam mengakses layanan diagnosa dengan cara yang intuitif dan *user-friendly*. Setiap komponen pada tampilan diagnosa telah diatur secara strategis untuk memastikan alur diagnosa berjalan lancar dan informasi yang diberikan mudah dipahami oleh pengguna. Desain visual dan fungsionalitas telah dipertimbangkan dengan seksama untuk memberikan pengalaman konsultasi yang efektif dan efisien.



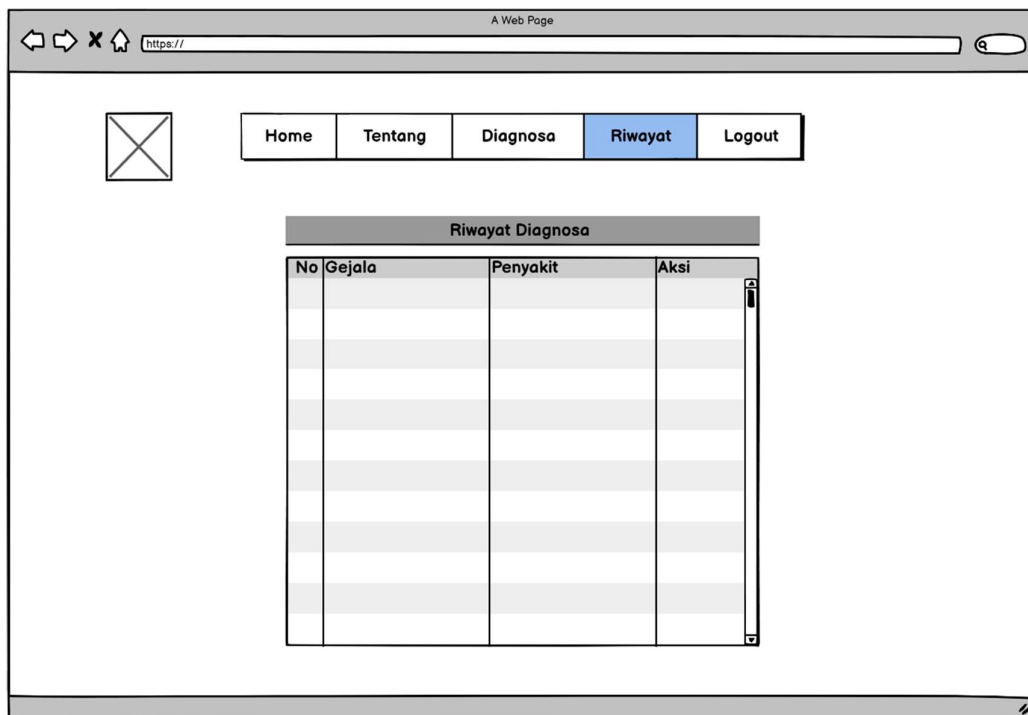
The screenshot shows a web browser window with a navigation menu containing 'Home', 'Tentang', 'Diagnosa' (highlighted), 'Riwayat', and 'Logout'. The form includes input fields for 'Usia', radio buttons for 'Jenis Kelamin' (Laki-laki, Perempuan), 'Merokok' (Iya, Tidak), 'Jenis Rokok' (Elektrik, Tembakau), and 'Lama Merokok' (0 Tahun, < 7 Tahun, > 7 Tahun). A table lists symptoms with checkboxes, and a 'Submit' button is located at the bottom right.

Gejala Penyakit	
Saya merasakan nyeri saat menelan	<input type="checkbox"/>
Saya merasakan tenggorokan sakit dan gatal	<input type="checkbox"/>
Saya mengalami suara serak	<input type="checkbox"/>
Saya merasakan batuk kering atau kronis	<input type="checkbox"/>
Saya mengalami batuk berdarah	<input type="checkbox"/>
Saya merasakan napas terasa sesak	<input type="checkbox"/>
Saya mengalami berat badan menurun	<input type="checkbox"/>
Saya merasakan tulang belakang nyeri	<input type="checkbox"/>
Saya mengalami BAB berdarah	<input type="checkbox"/>

Gambar 4.10 Tampilan Konsultasi Pengguna

Tampilan Halaman Riwayat Diagnosa Pengguna

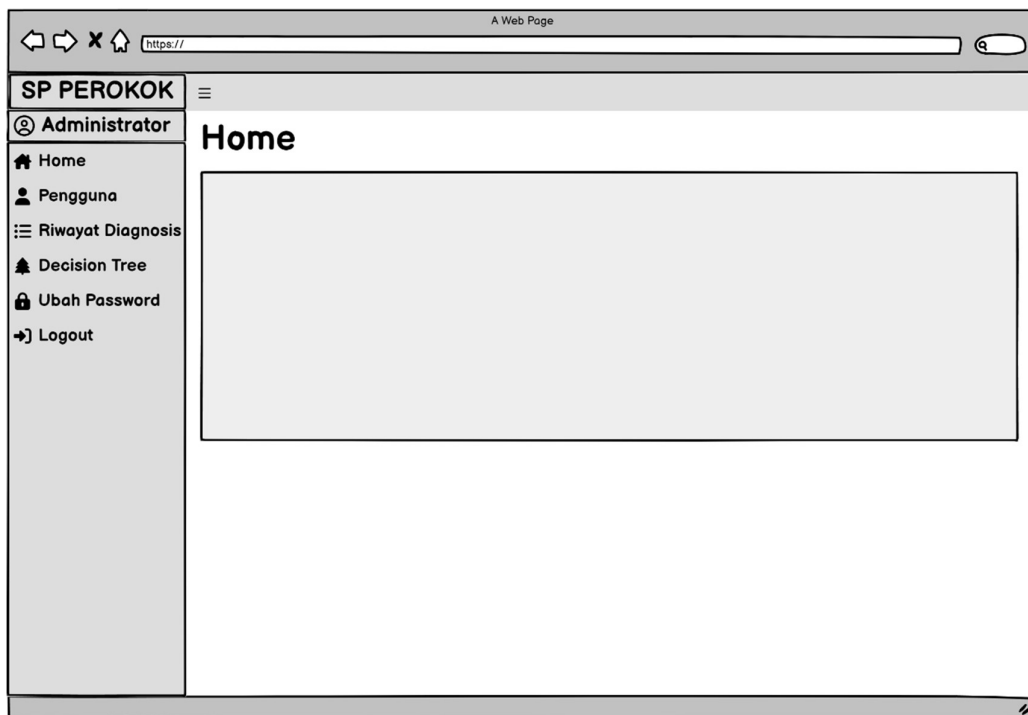
Pada gambar tersebut, ditampilkan rancangan tampilan riwayat diagnosa untuk pengguna yang akan dibuat dan diimplementasikan dalam sistem. Setiap elemen pada tampilan riwayat diagnosa telah diatur untuk memastikan bahwa informasi disampaikan dengan cara yang mudah diakses dan dipahami oleh pengguna. Desain ini juga telah dipertimbangkan agar riwayat diagnosa dapat diinterpretasikan dengan akurat dan cepat.



Gambar 4.11 Tampilan Hasil Konsultasi Pengguna

Tampilan Halaman Utama Admin

Pada gambar tersebut, ditampilkan rancangan tampilan utama untuk *admin* yang akan dibuat dan diimplementasikan dalam sistem. Rancangan ini disusun untuk memastikan bahwa *admin* dapat mengelola dan memantau seluruh fungsi sistem dengan efisien. Setiap elemen pada tampilan utama *admin* diatur dengan cermat agar informasi penting dapat diakses dengan cepat dan pengelolaan sistem dapat dilakukan secara efektif. Desain ini juga mempertimbangkan kebutuhan *admin* untuk navigasi yang mudah dan akses ke fitur-fitur yang diperlukan dalam pengelolaan sistem sehari-hari.

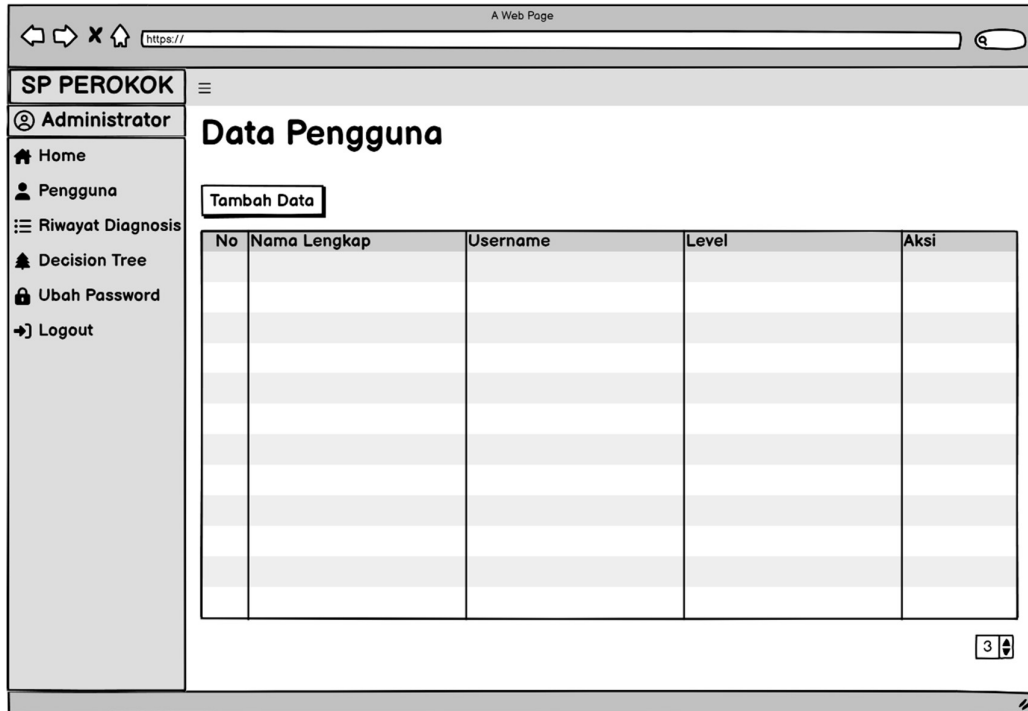


Gambar 4.12 Tampilan Utama Admin

Tampilan Halaman Data Pengguna

Pada gambar tersebut, ditampilkan rancangan tampilan data pengguna di bagian *admin* yang akan dibuat dan diimplementasikan dalam sistem. Rancangan

ini disusun untuk memungkinkan *admin* mengakses dan mengelola data pengguna dengan mudah dan efisien. Setiap komponen pada tampilan data pengguna diatur secara strategis untuk memastikan bahwa informasi penting dapat ditemukan dan diubah dengan cepat. Desain ini juga dirancang untuk meminimalkan kesalahan dalam pengelolaan data dan meningkatkan akurasi serta keamanan dalam pengelolaan informasi pengguna.

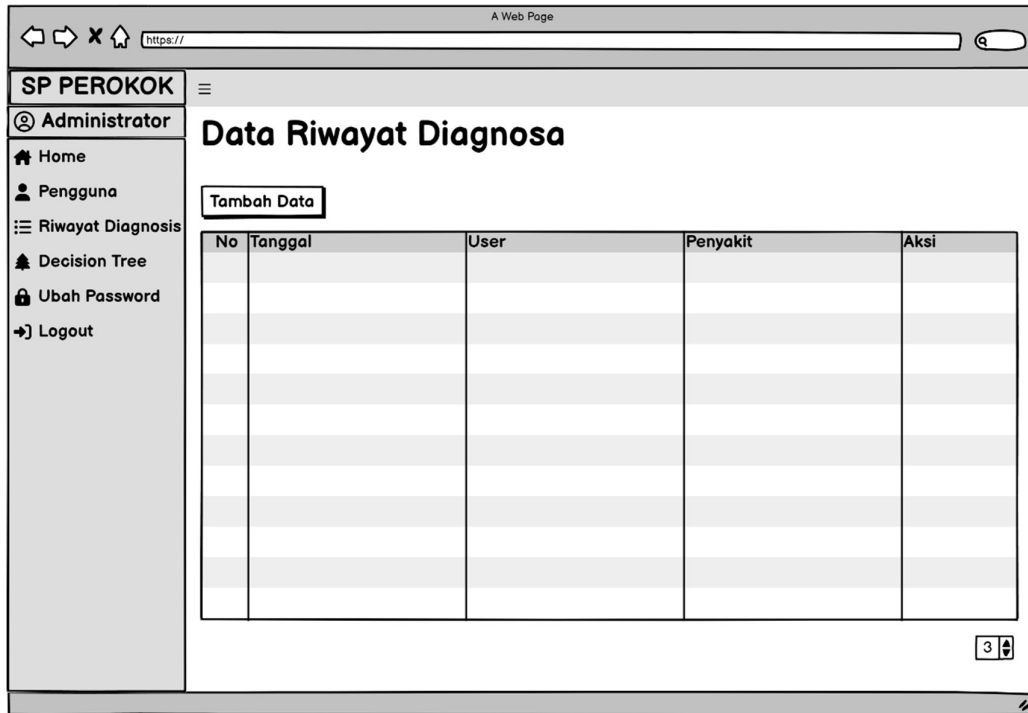


Gambar 4.13 Tampilan Data Pengguna

Tampilan Data Riwayat Diagnosa

Pada gambar tersebut, ditampilkan rancangan tampilan data riwayat diagnosa di bagian *admin* yang akan dibuat dan diimplementasikan dalam sistem. Rancangan ini disusun untuk memudahkan *admin* dalam mengakses dan mengelola riwayat diagnosa pengguna secara efisien. Setiap elemen pada tampilan ini diatur dengan cermat agar informasi riwayat diagnosa dapat disajikan secara jelas dan terstruktur, sehingga *admin* dapat dengan mudah menelusuri dan menganalisis data yang diperlukan. Desain ini juga dipertimbangkan agar pengelolaan riwayat diagnosa dilakukan dengan akurat dan aman, mendukung

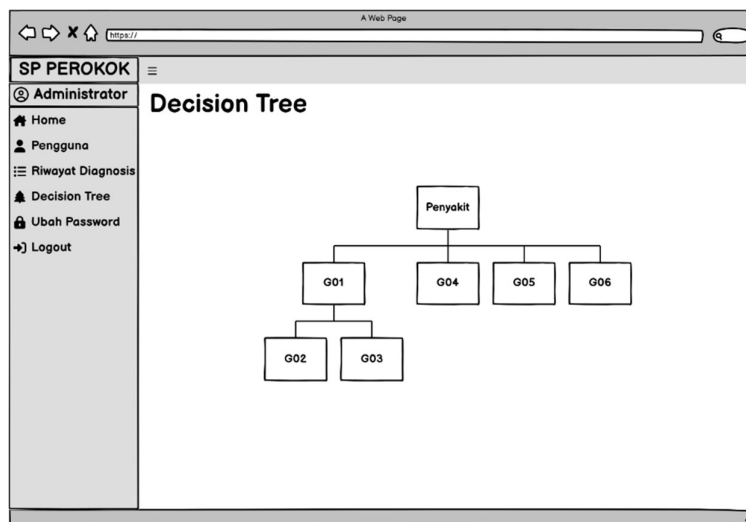
tugas *admin* dalam menjaga integritas dan kerahasiaan data.



Gambar 4.14 Tampilan Riwayat Diagnosa

Tampilan Halaman Visualisasi Analisis

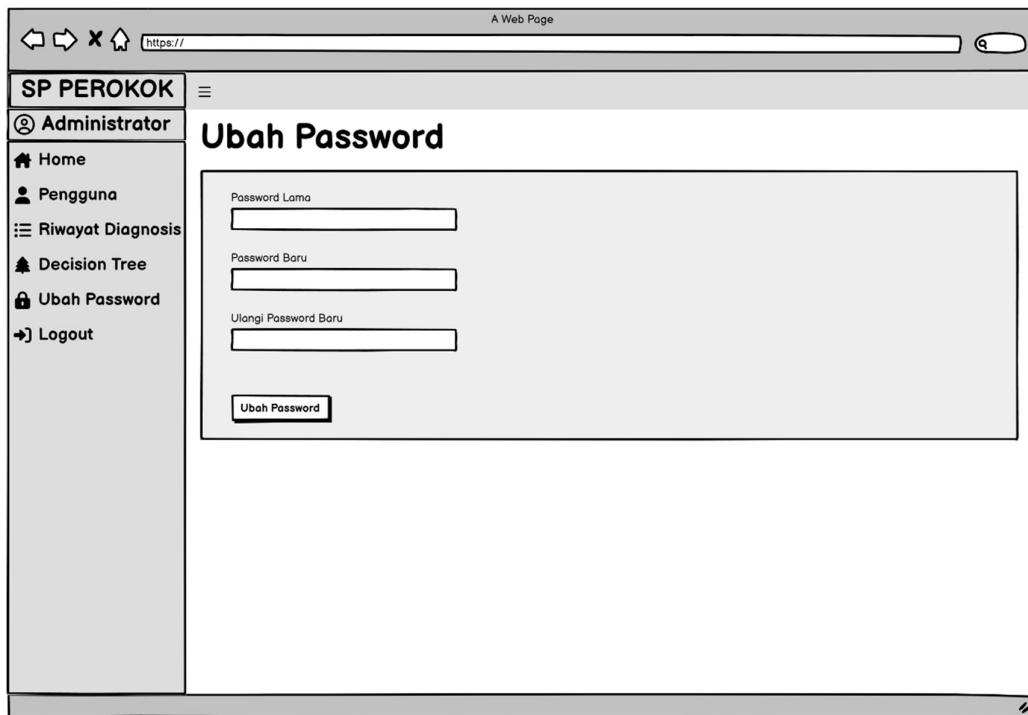
Pada gambar merupakan rancangan tampilan visualisasi analisis di bagian *admin* yang akan dibuat.



Gambar 4.15 Tampilan Visualisasi Analisis

Tampilan Ubah Password Admin

Pada gambar tersebut, ditampilkan rancangan tampilan ubah *password admin* yang akan dibuat dan diimplementasikan dalam sistem. Rancangan ini disusun untuk memudahkan *admin* dalam memperbarui informasi keamanan mereka secara aman dan efisien. Setiap elemen pada tampilan ini diatur dengan cermat untuk memastikan bahwa proses penggantian *password* dapat dilakukan dengan mudah dan cepat, sambil tetap menjaga tingkat keamanan yang tinggi. Desain ini juga dirancang untuk meminimalkan kesalahan dalam pengisian informasi dan memberikan petunjuk yang jelas bagi admin selama proses perubahan *password*.



The image shows a web browser window with the address bar displaying "https://". The page title is "A Web Page". The main content area is titled "Ubah Password" and is accessed by an "Administrator". The sidebar on the left contains the following navigation items: Home, Pengguna, Riwayat Diagnosis, Decision Tree, Ubah Password, and Logout. The main content area contains three input fields for "Password Lama", "Password Baru", and "Ulangi Password Baru", followed by a "Ubah Password" button.

Gambar 4.16 Tampilan Ubah *Password Admin*

Pengujian Sistem

Setelah sistem selesai dibangun, langkah berikutnya adalah dilakukan pengujian terhadap sistem tersebut untuk memastikan bahwa semua fungsionalitas bekerja sesuai dengan yang diharapkan. Pengujian ini dilakukan menggunakan metode *blackbox testing*, di mana pengujian difokuskan pada *output* yang dihasilkan oleh sistem berdasarkan *input* yang diberikan, tanpa melihat struktur internal atau kode sumber sistem. Dengan metode ini, diharapkan semua kemungkinan kesalahan atau *bug* yang mungkin ada dapat diidentifikasi dan diperbaiki sebelum sistem diimplementasikan secara penuh. Berikut adalah tabel pengujian *black box testing*.

Tabel 4.2 Hasil *Black Box Testing*

No	Fitur yang Diuji	Input yang Diberikan	Hasil yang Diharapkan	Hasil yang Diperoleh	Status
1	Form Registrasi	Nama, email, dan password valid	Pengguna berhasil mendaftarkan dan diarahkan ke halaman login	Hasil sesuai harapan	Lulus
2	Form Login	Email dan password valid	Pengguna berhasil masuk ke dashboard	Hasil sesuai harapan	Lulus
3	Validasi Login	Email atau password salah	Peringatan bahwa email atau password salah	Hasil sesuai harapan	Lulus
4	Tampilan Beranda User	Akses dengan login	Beranda user ditampilkan dengan informasi utama	Hasil sesuai harapan	Lulus
5	Form Diagnosa	Gejala yang dialami pengguna	Hasil diagnosa ditampilkan	Hasil sesuai harapan	Lulus
6	Tampilan Riwayat	Akses riwayat diagnosa	Riwayat diagnosa ditampilkan dengan benar	Hasil sesuai harapan	Lulus
7	Tampilan Dashboard	Login dengan akun admin	Dashboard admin dengan informasi	Hasil sesuai harapan	Lulus

	Admin		utama ditampilkan		
8	Manajemen Data Pengguna	Hapus pengguna	Pengguna dapat dihapus dari sistem	Hasil sesuai harapan	Lulus
9	Tampilan Riwayat Diagnosa	Akses data riwayat diagnosa pengguna	Data riwayat diagnosa ditampilkan dengan lengkap	Hasil sesuai harapan	Lulus
10	Visualisasi Proses Analisis	Akses visualisasi	Visualisasi proses analisis dapat dilihat	Hasil sesuai harapan	Lulus
11	Form Ubah Password Admin	Password lama, password baru	Password berhasil diubah dan disimpan	Hasil sesuai harapan	Lulus

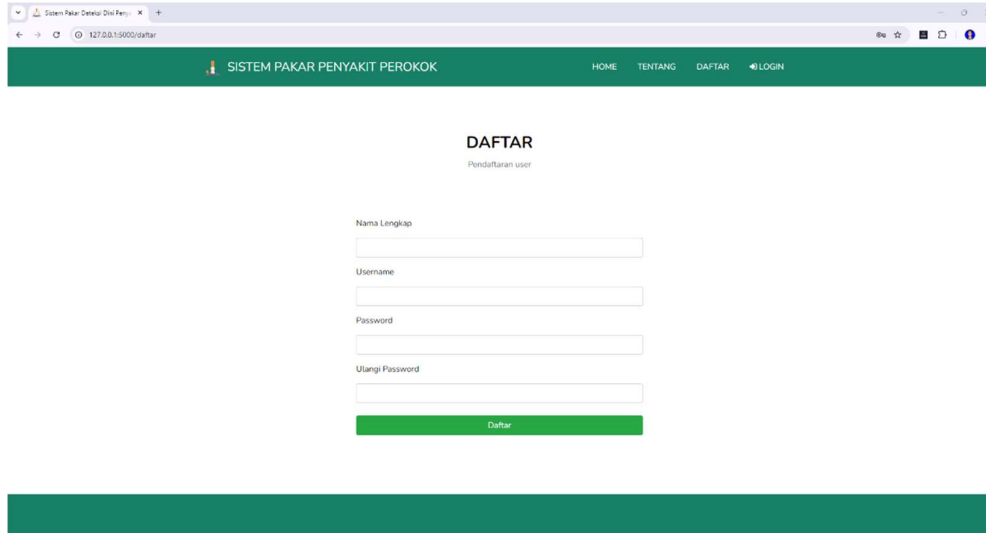
Tabel ini mencakup pengujian untuk berbagai fitur utama dari website sistem pakar yang telah dibuat. Setiap fitur diuji dengan menggunakan metode *black box testing* untuk memastikan bahwa fungsionalitas bekerja seperti yang diharapkan. Jika ada hasil yang tidak sesuai, maka perlu dilakukan *debugging* dan pengujian ulang.

Implementasi Sistem

Pada aplikasi ini terdapat dua *role user*, yaitu *admin* dan pengguna. Setiap *role user* memiliki tampilan masing-masing. Berikut adalah hasil implementasi Rancang Bangun Sistem Pakar Deteksi Dini Penyakit Pada Perokok Dengan Pendekatan *Machine Learning*.

Implementasi Halaman Daftar

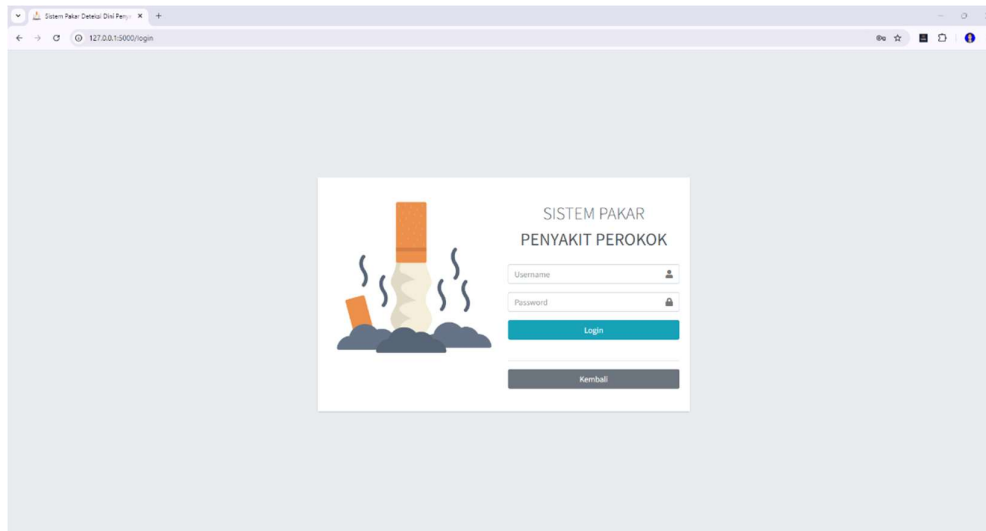
Pada halaman ini, formulir pendaftaran untuk pengguna baru disediakan. Nama lengkap, *username*, *password*, dan konfirmasi *password* harus dimasukkan oleh pengguna sebelum pendaftaran dapat diproses. Semua informasi yang diminta harus diisi dengan benar agar proses pendaftaran berhasil. Setelah data dimasukkan, tombol "Daftar" harus diklik untuk menyimpan informasi pengguna baru di sistem. Informasi yang dimasukkan akan diverifikasi untuk memastikan keaslian dan keakuratan data. Jika terdapat kesalahan dalam pengisian, pengguna akan diminta untuk memperbaiki data sebelum pendaftaran dapat diselesaikan. Setelah pendaftaran berhasil, pengguna akan diarahkan ke halaman *login* untuk masuk ke dalam sistem.



Gambar 4.17 Implementasi Halaman Daftar

Implementasi Halaman Login

Pada halaman ini, formulir *login* untuk pengguna ditampilkan. Pengguna diminta untuk memasukkan *username* dan *password* mereka. Setelah informasi ini dimasukkan, tombol "Login" harus diklik untuk mengakses sistem. Jika informasi yang dimasukkan benar, pengguna akan diarahkan ke halaman beranda mereka. Jika tidak, mereka akan tetap berada di halaman *login* dengan pesan kesalahan yang ditampilkan.



Gambar 4.18 Implementasi Halaman *Login*

Implementasi Halaman Utama Pengguna

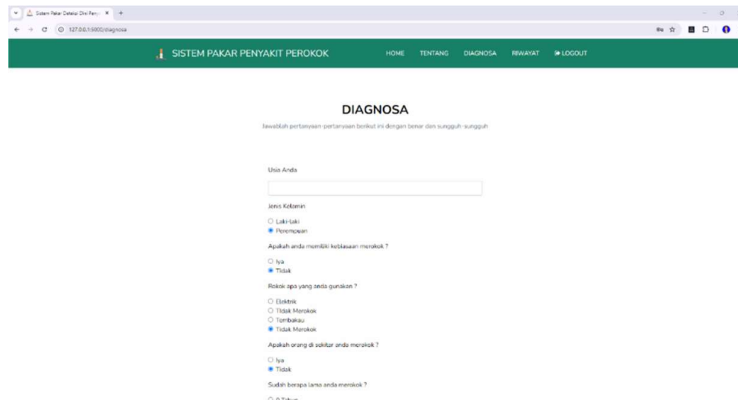
Setelah *login* berhasil dilakukan, halaman beranda pengguna akan ditampilkan. Pada halaman ini, berbagai opsi menu seperti *Home*, *Tentang*, *Diagnosa*, *Riwayat*, dan *logout* disediakan pada *navbar* untuk navigasi. Halaman ini juga menampilkan pesan selamat datang dan informasi umum tentang sistem pakar deteksi dini penyakit pada perokok. Pengguna dapat mengklik tombol "Mulai Diagnosa" untuk memulai proses diagnosa.



Gambar 4.19 Implementasi Halaman Utama Pengguna

Implementasi Halaman Diagnosa Pengguna

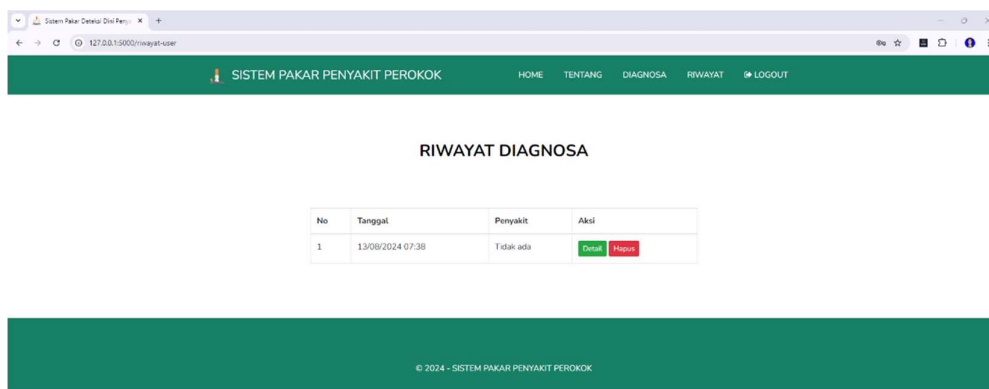
Pada halaman ini, formulir diagnosa yang berisi beberapa pertanyaan tentang kebiasaan merokok dan gejala yang dirasakan disediakan. Pengguna diminta untuk mengisi semua pertanyaan dengan benar sebelum diagnosa dapat dilakukan. Informasi yang dikumpulkan melalui formulir ini akan digunakan untuk menentukan kemungkinan penyakit yang diderita oleh pengguna. Setelah semua pertanyaan dijawab, tombol "Submit" harus diklik untuk memproses diagnosa. Hasil diagnosa akan ditampilkan kepada pengguna setelah proses analisis selesai dilakukan.



Gambar 4.20 Implementasi Halaman Diagnosa Pengguna

Implementasi Halaman Riwayat Diagnosa Pengguna

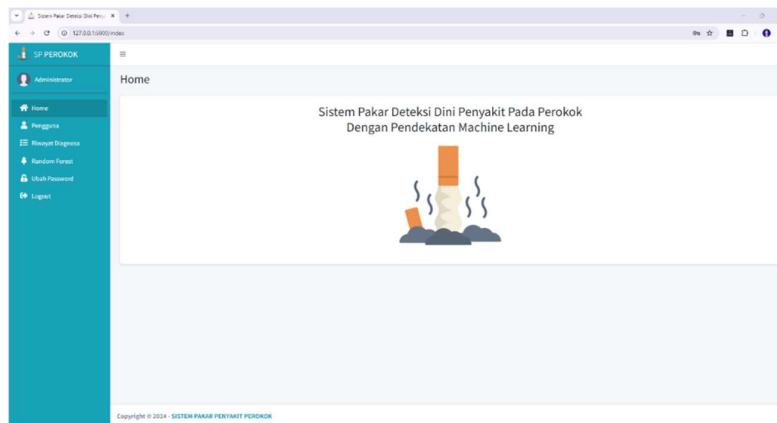
Riwayat diagnosa pengguna akan ditampilkan dalam bentuk tabel pada halaman ini. Tabel ini mencakup nomor, tanggal, penyakit, dan aksi yang dapat dilakukan oleh pengguna, seperti melihat detail diagnosa. Informasi ini memungkinkan pengguna untuk melacak dan mengelola riwayat kesehatan mereka berdasarkan hasil diagnosa yang telah dilakukan sebelumnya. Setiap entri dalam tabel dapat diklik untuk melihat detail lebih lanjut tentang diagnosa yang telah dilakukan. Informasi detail ini mencakup gejala yang dilaporkan, dan hasil analisis.



Gambar 4.21 Implementasi Halaman Riwayat Diagnosa

Implementasi Halaman Utama Admin

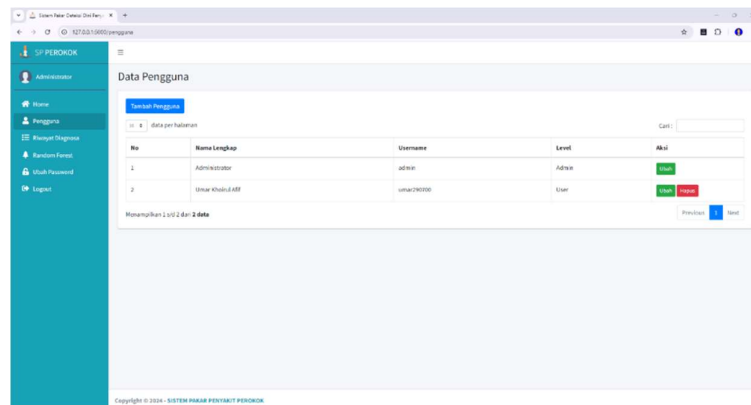
Pada halaman ini, beranda untuk administrator ditampilkan. Sistem Pakar Deteksi Dini Penyakit Pada Perokok dengan Pendekatan *Machine Learning* dijelaskan secara ringkas di sini. Berbagai opsi menu yang terletak di sidebar kiri disediakan untuk navigasi, seperti *Home*, *Pengguna*, *Riwayat Diagnosa*, *Random Forest*, *Ubah Password*, dan *Logout*. Informasi umum tentang sistem juga ditampilkan untuk memberikan pemahaman dasar kepada pengguna.



Gambar 4.22 Implementasi Halaman Utama Admin

Implementasi Halaman Data Pengguna

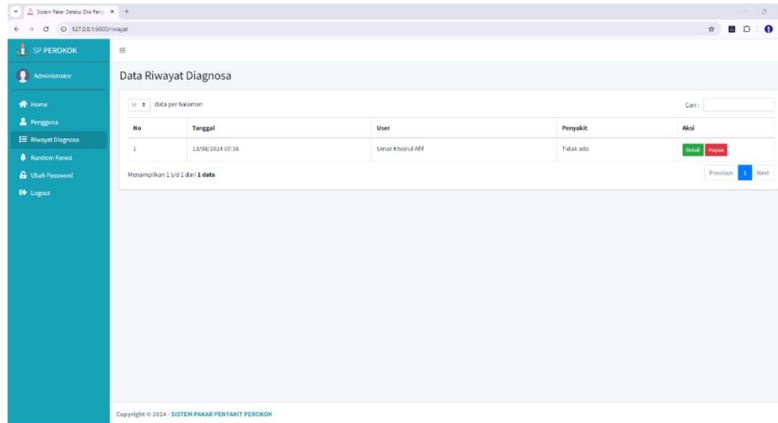
Pada halaman ini, data pengguna sistem ditampilkan dalam bentuk tabel. Nama lengkap, *username*, dan level akses pengguna disajikan dalam kolom-kolom tabel. Tindakan yang dapat diambil terhadap setiap pengguna, seperti mengubah atau menghapus data, disediakan melalui tombol yang relevan.



Gambar 4.23 Implementasi Halaman Data Pengguna

Implementasi Data Riwayat Diagnosa

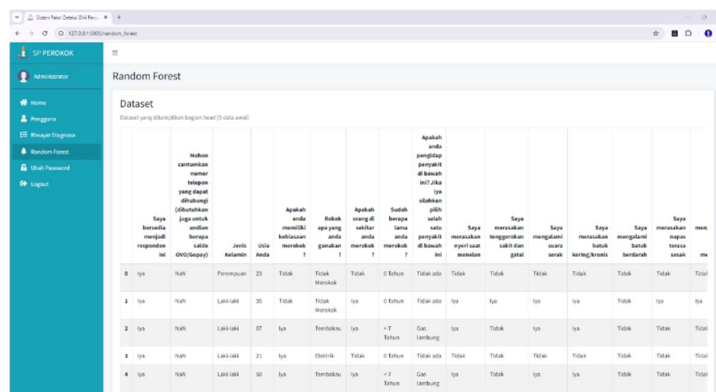
Riwayat diagnosa pengguna ditampilkan pada halaman ini. Data diagnosa ditampilkan dalam bentuk tabel yang mencakup nomor, tanggal, *user*, penyakit, dan aksi yang dapat diambil. *Admin* dapat melihat detail setiap diagnosa yang telah dilakukan oleh pengguna melalui tabel ini. Opsi pencarian juga disediakan untuk mempermudah pencarian riwayat diagnosa tertentu.



Gambar 4.24 Implementasi Data Riwayat Diagnosa

Implementasi Halaman Visualisasi Analisis

Dataset yang digunakan dalam analisis *Random Forest* ditampilkan pada halaman ini. Data yang ditampilkan mencakup informasi demografis dan gejala yang dialami oleh pengguna. Informasi ini digunakan untuk membangun model prediktif menggunakan algoritma *Random Forest*.



Gambar 4.25 Implementasi Halaman Visualisasi Analisis

Implementasi Ubah Password Admin

Pada halaman ini, *admin* dapat mengubah *password*. *Admin* diminta untuk memasukkan *password* lama, *password* baru, dan konfirmasi *password* baru sebelum perubahan dapat disimpan. Setelah semua informasi dimasukkan, tombol "Ubah Password" harus diklik untuk menyimpan perubahan.

The screenshot displays a web application interface for changing an administrator's password. On the left, a teal sidebar contains a menu with the following items: 'Administrator' (with a user icon), 'Home' (with a house icon), 'Pengguna' (with a person icon), 'Riwayat Diagnosa' (with a list icon), 'Random Forest' (with a tree icon), 'Ubah Password' (with a lock icon and highlighted in dark teal), and 'Logout' (with a door icon). The main content area is titled 'Ubah Password' and contains three text input fields: 'Password Lama', 'Password Baru', and 'Ulangi Password Baru'. Below these fields is a green button labeled 'Ubah Password'. The browser's address bar shows the URL '127.0.0.1:5000/password'. At the bottom of the page, a small copyright notice reads 'Copyright © 2024 - SISTEM PAKAR PENYAKIT PEROKOK'.

Gambar 4.26 Implementasi Ubah *Password Admin*

BAB 5

KESIMPULAN DAN SARAN

Kesimpulan

Penelitian ini telah berhasil merancang dan membangun aplikasi website untuk deteksi dini penyakit pada perokok dengan pendekatan *machine learning*. Aplikasi ini dirancang untuk membantu pengguna dalam mengidentifikasi risiko penyakit yang mungkin dihadapi oleh perokok berdasarkan data gejala yang diinput. Pendekatan *machine learning* yang diterapkan dalam aplikasi ini memungkinkan deteksi dini yang lebih akurat dan cepat, memberikan nilai tambah dalam upaya peningkatan kesehatan masyarakat secara keseluruhan.

Manfaat dari penelitian ini sangat signifikan bagi masyarakat, terutama bagi perokok yang ingin mengetahui kondisi kesehatannya lebih awal. Bagi pengguna, aplikasi ini membantu dalam melakukan *self-assessment* terhadap kesehatan mereka tanpa harus segera pergi ke fasilitas kesehatan.

Saran

Adapun saran pada penelitian ini adalah sebagai berikut:

1. Aplikasi deteksi dini penyakit pada perokok dengan pendekatan *machine learning* ini sebaiknya dikembangkan menjadi aplikasi mobile untuk meningkatkan aksesibilitas dan kemudahan penggunaan oleh masyarakat luas.
2. Disarankan untuk menambahkan fitur notifikasi kepada pengguna yang memberikan peringatan atau saran tindak lanjut berdasarkan hasil deteksi yang diperoleh, sehingga pengguna dapat segera mengambil langkah-langkah pencegahan yang diperlukan.
3. Perlu dipertimbangkan untuk memperluas cakupan data yang digunakan dalam model *machine learning* dengan menambahkan data dari berbagai sumber yang lebih luas dan terkini, sehingga akurasi prediksi dapat lebih ditingkatkan.
4. Integrasi dengan sistem kesehatan atau penyedia layanan medis pihak ketiga sebaiknya dipertimbangkan agar pengguna dapat dengan mudah mengakses layanan konsultasi medis setelah menerima hasil deteksi dari aplikasi ini.