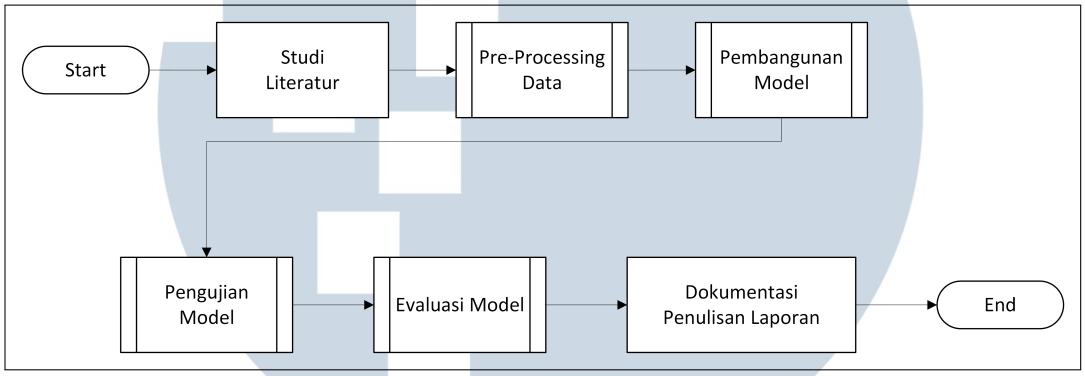
BAB 3
METODOLOGI PENELITIAN

3.1 Tahapan Penelitian

Diagram alur dari proses alur kerja atau tahap penelitian secara umum dapat dilihat melalui Gambar 3.1.



Gambar 3.1. Diagram Alur Kerja Penelitian

Gambar 3.1 merupakan tahapan penelitian yang dilakukan dan tahapan tersebut dijabarkan sebagai berikut.

1. Studi Literatur

   Telaah literatur menjadi langkah awal dalam penelitian dan penyusunan laporan sebagai acuan teori ataupun pengetahuan mengenai topik yang diangkat. Telaah literatur bertujuan untuk menggali informasi yang menjadi fokus penelitian. Informasi pengetahuan didapatkan dari membaca dan mengkaji sumber-sumber yang terpercaya seperti jurnal dan buku literatur. Telaah literatur yang dibahas terdiri dari beberapa teori yang digunakan dalam penelitian antara lain, Penyakit Jantung, *Machine Learning*, Algoritma *Random Forest*, *Supervised Learning*, *Confusion Matrix*, dan SMOTE.

2. Pengumpulan Data

   Proses penelitian dilakukan dengan diawali dengan tahap mencari *dataset* dari sumber yang relevan. *Dataset* yang digunakan adalah *Framingham* yang dapat ditemukan dari *website Kaggle* [33]. *Dataset* ini berformat CSV yang terdiri dari 15 fitur dan 1 target dengan 4.240 sampel data. Data terdiri dari beberapa fitur seperti: *male, age, education, currentSmoker,*

cigsPerDay, BPMeds, prevalentStroke, prevalentHyp, diabetes, totChol, sysBP, diaBP, BMI, heartRate, glucose, TenYearCHD. *Dataset* diambil dengan memfokuskan isi data yang berkaitan dengan penyakit jantung koroner.

3. Pre-Processing Data

   Pada tahap ini dilakukan *pre-processing data* untuk *data cleaning*, mengkonversi data yang memiliki tipe data *non-integer* menjadi numerik, dan pemilihan fitur yang memiliki hubungan kuat dengan variabel *target*. Kemudian, data untuk *validation set* diambil terlebih dahulu sebesar 10% dari *dataset*. Setelah itu, sisa *dataset* digunakan untuk *training* dan *testing*.

4. Pembangunan Model

   Pada tahap ini, pembangunan model dilakukan menggunakan algoritma *Random Forest* untuk memprediksi kemungkinan seseorang mengalami penyakit jantung koroner berdasarkan laporan kesehatan. *Dataset* yang digunakan berisi beberapa fitur seperti: *male, age, education, currentSmoker, cigsPerDay, BPMeds, prevalentStroke, prevalentHyp, diabetes, totChol, sysBP, diaBP, BMI, heartRate, glucose, TenYearCHD*. Oleh karena itu, proses pembangunan model berfokus pada pelatihan algoritma untuk mempelajari pola dalam data dan kategori pasien pada kelas dalam target tersebut.

5. Pengujian Model

   Pada tahap ini, model yang telah dilatih menggunakan *training set* diuji menggunakan *testing set* dan *validation set* untuk mengukur performa. Dilakukan beberapa uji coba yaitu menggunakan sejumlah 10%, 15%, 20%, 25%, 30%, 35% dari *dataset* digunakan untuk *testing set* dalam pemilihan dan penyetelan *hyperparameter*, supaya lebih optimal dan terhindar dari *overfitting*. Pengujian ini dilakukan untuk mengukur performa model setelah pelatihan. Jika model memberikan performa yang baik saat *training set* dan kurang baik pada *testing set*, maka indikasi menunjukkan *overfitting* yang dapat dilakukan perbaikan dengan *tuning hyperparameter* dan SMOTE.

6. Evaluasi

   Pada tahap ini, evaluasi dilakukan dengan tujuan untuk memeriksa dari penilaian kinerja algoritma *Random Forest*. Keberhasilan algoritma diukur dengan menggunakan *confusion matrix* untuk menghitung metrik evaluasi

seperti akurasi, *precision*, *recall*, dan *f1-Score*. Selain itu, dihitung juga akurasi menggunakan *validation set* untuk menilai sejauh mana model mampu melakukan generalisasi terhadap data baru. Hasil evaluasi memberikan gambaran tentang seberapa efektif model algoritma *Random Forest* dalam mendeteksi penyakit jantung. Hasil dari evaluasi menjadi dasar dalam menentukan efektivitas model untuk membuat prediksi yang akurat.

7. Dokumentasi Penulisan Laporan

   Dokumentasi dalam penelitian ini bertujuan untuk merekam setiap tahap dan perkembangan yang dilakukan selama proses penelitian dan memberi informasi untuk penelitian ke depan. Tahap ini mencakup *progress* setiap langkah dari penelitian. Dokumentasi dilakukan dengan menulis setiap pekerjaan yang dilakukan melalui penyusunan laporan. Selain itu, dilakukan konsultasi dengan dosen pembimbing dengan mengkonsultasikan pekerjaan yang dilakukan untuk diberi *feedback* dan dapat dilakukan peningkatan dalam pengerjaan. Oleh karena itu, tahap ini dilakukan selama penelitian berlangsung.

3.2 Pre-Processing Data

Pada tahap ini, dilakukan beberapa tahap untuk mempersiapkan proses data sebelum digunakan dalam pembangunan model. *Dataset* yang digunakan pada penelitian ini adalah *dataset* Framingham yang diunduh dari *Kaggle* [33] memiliki 15 fitur dan 1 target dengan 4.240 sampel data. Berdasarkan hasil eksplorasi data, ditemukan bahwa beberapa fitur memiliki nilai yang hilang, kemudian dilakukan penghapusan baris pada data yang memiliki nilai hilang, sehingga jumlah sampel *dataset* berkurang dari 4.240 menjadi 3.658. Dari jumlah data pasien sebanyak 3.658, klasifikasi kelas positif atau yang terkena penyakit jantung yaitu sebanyak 557 data pasien, dan kelas negatif atau pasien tidak terkena penyakit jantung yaitu sebanyak 3.101 data pasien. Klasifikasi kelas dalam *dataset* yang digunakan mengalami ketidakseimbangan kelas atau *imbalanced data* dengan kelas mayoritas yaitu pasien tidak terkena penyakit jantung (negatif) dan minoritas yaitu pasien terkena penyakit jantung (positif). Setelah dilakukan eksplorasi data dan konsultasi dengan dokter spesialis jantung dan pembuluh darah bersama Dr. Hamzah Muhammad Zein Sp.JP, FIHA , diputuskan untuk menghapus beberapa fitur yang dianggap kurang relevan dalam menentukan kemungkinan seseorang

mengalami penyakit jantung koroner. Fitur-fitur tersebut adalah *education, BPMeds, cigsPerDay*, dan *heartRate*, dengan penghapusan keempat fitur tersebut, jumlah fitur yang digunakan dalam pembangunan model berkurang dari 15 fitur menjadi 11 fitur. Langkah ini dilakukan untuk memastikan model lebih fokus dengan faktor yang lebih relevan terhadap penyakit jantung koroner khususnya yang berkaitan dengan kondisi pembuluh darah menurut pendapat medis. Detail dari setiap fiturnya dapat dilihat seperti pada Tabel 3.1.

Tabel 3.1. Tabel Fitur dan Tipe Data

No	Fitur	Tipe Data
1	Male	Integer
2	Age	Integer
3	currentSmoker	Integer
4	prevalentStroke	Integer
5	prevalentHyp	Integer
6	diabetes	Integer
7	totChol	Float
8	sysBP	Float
9	diaBP	Float
10	BMI	Float
11	glucose	Float
12	TenYearCHD	Integer

Detail dari setiap fitur dapat dijabarkan sebagai berikut:

1. Male

   Fitur *Male* pada *dataset* memiliki tipe data *Integer* dan menunjukkan jenis kelamin pasien, 1 jika laki-laki dan 0 jika perempuan.

2. Age

   Fitur *Age* pada *dataset* memiliki tipe data *Integer* dan menunjukkan usia pasien dalam tahun.

3. currentSmoker

Fitur *currentSmoker* pada *dataset* memiliki tipe data *Integer* dan menunjukkan kondisi pasien perokok aktif atau tidak.

4. prevalentStroke

   Fitur *prevalentStroke* pada *dataset* memiliki tipe data *Integer* dan menunjukkan riwayat pasien pernah mengalami penyakit stroke atau tidak.

5. prevalentHyp

   Fitur *prevalentHyp* pada *dataset* memiliki tipe data *Integer* dan menunjukkan riwayat pasien pernah mengalami penyakit hipertensi atau tidak.

6. diabetes

   Fitur *diabetes* pada *dataset* memiliki tipe data *Integer* dan menunjukkan kondisi pasien menderita diabetes atau tidak.

7. totChol

   Fitur *totChol* pada *dataset* memiliki tipe data *Float* dan menunjukkan total kolestrol dalam mg/dL.

8. sysBP

   Fitur *sysBP* pada *dataset* memiliki tipe data *Float* dan menunjukkan tekanan darah sistolik yaitu saat jantung berkontraksi dalam mmHg.

9. diaBP

   Fitur *diaBP* pada *dataset* memiliki tipe data *Float* dan menunjukkan tekanan darah diastolik yaitu saat jantung beristirahat di antara detak dalam mmHg.

10. BMI

    Fitur BMI pada *dataset* memiliki tipe data *Float* dan menunjukkan indeks massa tubuh dengan mengukur kesehatan berat badan seseorang dalam kg/m^2.

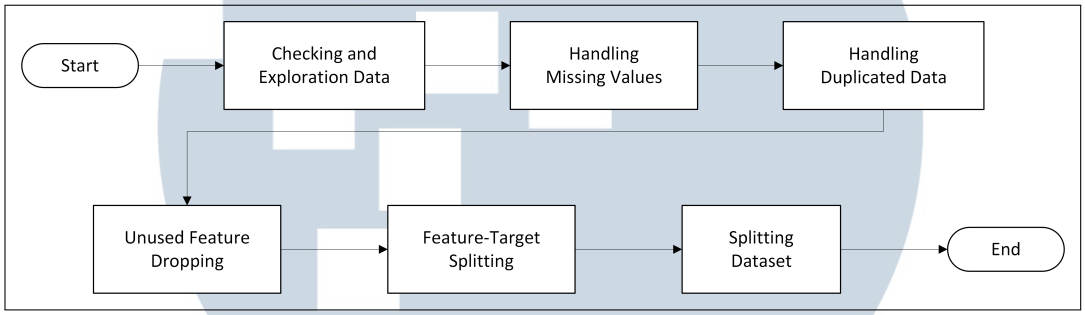
11. glucose

    Fitur *glucose* pada *dataset* memiliki tipe data *Float* dan menunjukkan kadar glukosa dalam darah dengan satuan mg/dL.

12. TenYearCHD

    Fitur *TenYearCHD* pada *dataset* memiliki tipe data *Integer* dan menunjukkan target variabel kondisi pasien mengalami penyakit jantung koroner dalam 10 tahun ke depan.

Diagram alur dari proses *Pre-Processing Data* dapat dilihat melalui Gambar 3.2.



Gambar 3.2. Diagram Alur Pre-Processing Data

Dari Gambar 3.2 dapat dijabarkan tahap alur *Pre-Processing Data* terdiri dari:

1. *Checking and Exploration Data*

   Pada tahap ini, dilakukan eksplorasi awal terhadap *dataset* untuk memahami struktur data, tipe data, jumlah sampel, jumlah fitur, serta distribusi nilai dalam setiap fitur. Selain itu, dilakukan pengecekan untuk data yang terduplikasi dan nilai yang hilang.

2. *Handling Missing Values*

   Pada tahap ini, dilakukan penanganan terhadap data yang memiliki nilai *missing* agar tidak menyebabkan masalah dalam analisis dan pembangunan model. Metode yang digunakan yaitu dengan menghapus baris dengan *missing values*.

3. *Handling Duplicated Data*

   Pada tahap ini, dilakukan pengecekan kondisi data terduplikasi. Jika terdapat data terduplikasi, data yang sama dapat dihapus untuk menghindari bias dalam pelatihan model.

4. *Unused Feature Dropping*

   Pada tahap ini, dilakukan penghapusan kolom untuk fitur yang kurang relevan terhadap target yang ingin dianalisis untuk diprediksi berdasarkan pendapat dari narasumber atau ahli yaitu dokter spesialis jantung dan pembuluh darah yaitu Dr. Hamzah Muhammad Zein Sp.JP, FIHA.

5. *Feature-Target Splitting*

   Pada tahap ini, *dataset* dibagi menjadi dua bagian utama yaitu fitur atau X sebagai variabel independen yang digunakan sebagai prediktor dan target atau Y sebagai label yang diprediksi oleh model.

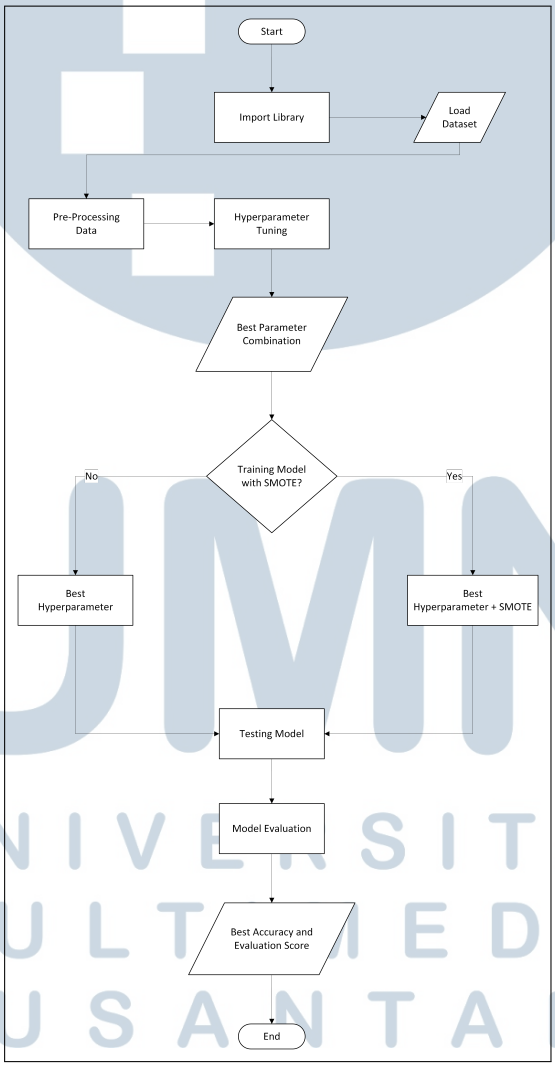
6. *Splitting Dataset*

   Pada tahap ini, data yang telah diproses dibagi menjadi beberapa bagian yaitu *training set*, *testing set*, dan *validation set*. *Training set* digunakan untuk melatih model, *testing set* digunakan untuk mengevaluasi performa model, dan *validation set* digunakan untuk mengukur performa kebaikan kerja model. Jumlah data atau pasien setelah melalui proses pembersihan menjadi 3.658 data atau pasien. Dari 3.658 data pasien tersebut dipisahkan terlebih dahulu sebesar 10% untuk menjadi data validasi atau sekitar 366 data pasien. Kemudian, sisa data 3.292 digunakan sebagai data *training* dan *testing* dengan beberapa skenario *splitting* rasio dataset *training* dan *testing* yaitu 90:10, 85:15, 80:20, 75:25, 70:30, dan 65:35.

3.3 Pembangunan Model hingga Evaluasi

Tahap pembangunan model dilakukan untuk membangun model klasifikasi prediksi pasien mengalami penyakit jantung dan keefektifan algoritma dalam membaca pelatihan pola data. Pembangunan model dilakukan dengan menggunakan data yang telah dikumpulkan yaitu *dataset framingham* berisi fitur kesehatan yang terfokus pada faktor penyakit jantung. Pada tahap awal, *dataset* telah dipisahkan sebesar 10% untuk data validasi, kemudian sisa data digunakan untuk *training* dan *testing*. Penggunaan beberapa skenario uji coba, *dataset* untuk *training* dan *testing* dibagi menjadi 90%, 85%, 80%, 75%, 70%, dan 65% dari dataset digunakan untuk melatih performa model yang dibangun dan 10%, 15%, 20%, 25%, 30%, dan 35% dari dataset untuk menguji model. Pembagian ini bertujuan untuk mengevaluasi seberapa baik model dapat mengeneralisasi

data yang belum pernah dieksekusi sebelumnya. Pada tahap pembangunan model, algoritma *Random Forest* diterapkan dengan konfigurasi *hyperparameter* yang disesuaikan untuk meningkatkan performa model. Setelah model dilatih menggunakan data *training*, dilakukan pengujian menggunakan data *testing* yang telah dipisahkan sebelumnya. Evaluasi model dilakukan menggunakan *confusion matrix* dengan menghitung mentrik evaluasi yang mencakup akurasi, *precision*, *recall*, dan f1-*score*. Tahap ini bertujuan untuk memahami performa model, mengidentifikasi kemungkinan *overfitting* atau *underfitting*, dan menilai sejauh mana model memprediksi dengan akurat. Diagram alur dari proses pembangunan model dapat dilihat melalui Gambar 3.3.



Gambar 3.3. Diagram Alur Pembangunan Model

Gambar 3.3 merupakan diagram alur tahap pembangunan model yang

dijelaskan dengan uraian sebagai berikut:

1. *Import Library*

   Pada tahap ini, *library* yang diperlukan diimpor untuk memproses data, membangun model, dan melakukan evaluasi.

2. *Load Dataset*

   Pada tahap ini, *dataset* yang berisi pasien dengan fitur kesehatan dimuat dengan menggunakan *library*. *Dataset* ini berasal dari sumber eksternal seperti *Kaggle*.

3. *Pre-Processing Data*

   Pada tahap ini, data dibersihkan dan dipersiapkan untuk digunakan di dalam model. Proses ini mencakup *handling missing and duplicated values*, *data splitting* menjadi *training*, *testing*, dan *validation set*.

4. *Hyperparameter Tuning*

   Pada tahap ini, dilakukan proses untuk mencari kombinasi parameter terbaik dari *Random Forest* menggunakan teknik *RandomizedSearchCV*.

5. *Best Parameter Combination*

   Pada tahap ini, setelah melakukan *hyperparameter tuning* diperoleh kombinasi parameter yang terbaik dengan memberikan performa yang optimal kepada model.

6. *Training Model*

   Pada tahap ini, model *Random Forest* dilatih menggunakan *dataset training* dengan parameter terbaik yang telah ditemukan dari *hyperparameter tuning*.

7. *Training Model with SMOTE*

   Pada tahap ini, terdapat skenario uji coba untuk menyeimbangkan jumlah data antara kelas mayoritas dan minoritas dengan digunakan MSOTE.

8. *Testing Model*

   Pada tahap ini, model diuji dengan *dataset testing* yang belum pernah digunakan sebelumnya untuk mengukur performa.

9. *Model Evaluation*

   Pada tahap ini, dilakukan uji performa menggunakan *validation set*. Hal ini dilakukan untuk mengukur kemungkinan *overfitting* atau *underfitting* pada data yang belum pernah digunakan. Selain itu, model dievaluasi menggunakan metrik seperti *accuracy, precision, recall*, f1-*score*, dan *confusion matrix* untuk menilai seberapa baik model dalam mendeteksi penyakit jantung koroner atau mengukur keakuratan dalam prediksi, dan mengukur kesiapan model untuk digunakan pada *real case/world*.

10. *Best Accuracy and Evaluation Score*

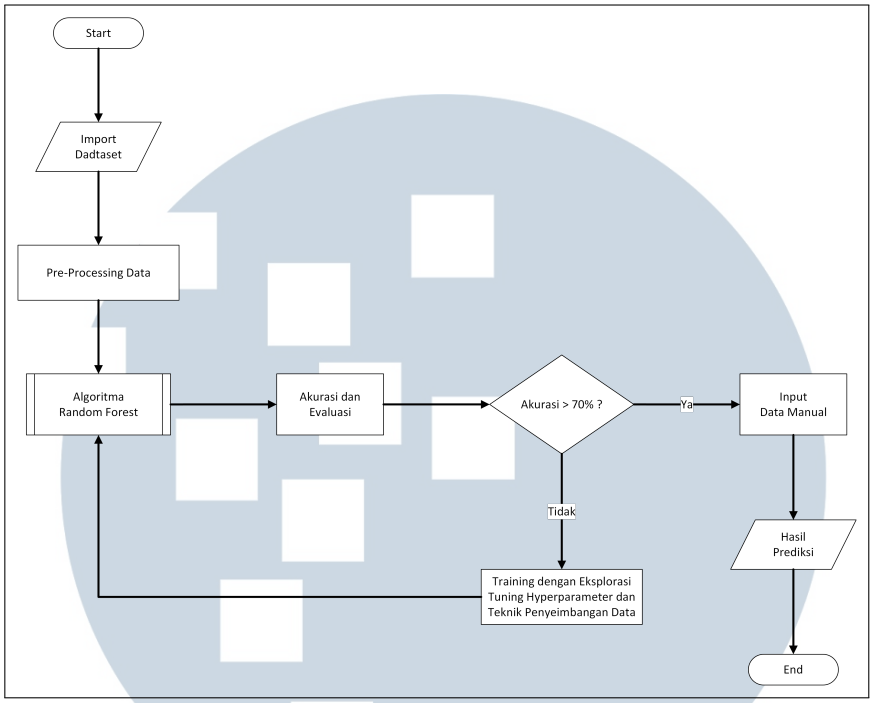
    Pada tahap ini, dilakukan penentuan untuk kombinasi terbaik berdasarkan akurasi dan skor evaluasi tertinggi, dan membandingkan dengan model sebelum serta sesudah penerapan *hyperparameter tuning* maupun SMOTE untuk meninjau terjadi peningkatan performa atau tidak.

3.4 Perancangan

Pada tahap ini, dilakukan perancangan alur program untuk memudahkan proses implementasi algoritma *Random Forest* pada deteksi penyakit jantung. Tahap ini dimulai dengan membuat *flowchart* sebagai alur program secara umum hingga detil. Kemudian, program diimplementasikan berdasarkan *flowchart* yang telah dirancang untuk memberikan gambaran alur program.

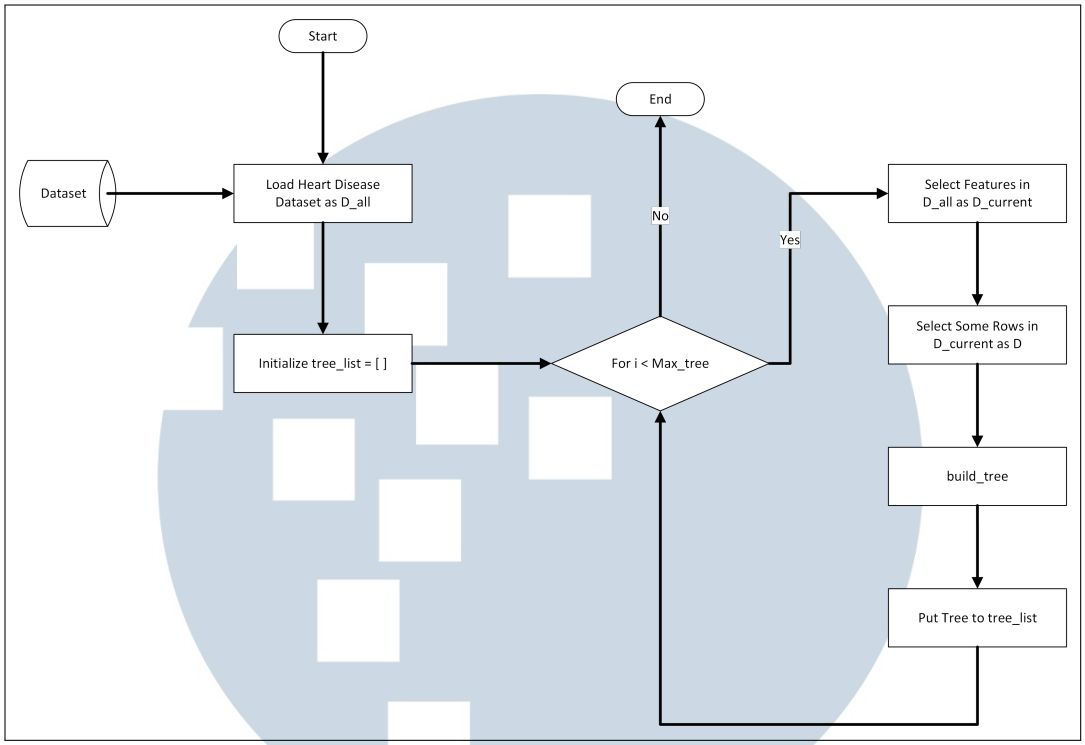
3.4.1 Flowchart

Flowchart yaitu diagram yang digunakan untuk merepresentasikan alur atau proses dalam suatu sistem dengan menggunakan simbol-simbol tertentu. Diagram ini membantu untuk memberikan gambaran langkah-langkah suatu proses dengan visual supaya lebih mudah untuk dipahami.



Gambar 3.4. Diagram Alur Program

Flowchart pada Gambar 3.4 menunjukkan diagram alur program deteksi penyakit jantung menggunakan algoritma *Random Forest*. Proses dimulai dengan *import dataset* yang dibutuhkan dalam program untuk dapat diolah pada tahap pra-pemrosesan data. Kemudian dilakukan pelatihan *dataset* dengan menggunakan modul Algoritma *Random Forest*. Program menggunakan algoritma *Random Forest* untuk melatih *dataset* dan dilakukan uji coba. Setelah itu, dapat menghitung akurasi dari *dataset* dan melakukan evaluasi performa model melalui *validation set*, *precision, f1-score,* dan *recall*. Setelah itu, jika akurasi lebih dari 70% maka model layak untuk melakukan *input* data baru guna dilakukan pengujian dan prediksi probabilitas terkena penyakit jantung berdasarkan kebiasaan dari hasil latih *dataset*. Jika tidak, maka dilakukan *training* ulang dengan eksplorasi metode atau teknik untuk dapat meningkatkan nilai akurasi.



Gambar 3.5. Diagram Alur Modul Algoritma Random Forest

Flowchart pada Gambar 3.5 menunjukkan diagram alur modul Algoritma *Random Forest* [34]. Proses dimulai dengan memuat *dataset* penyakit jantung ke dalam variabel bernama $D_{\_all}$. *Dataset* ini berisi data kesehatan pasien yang digunakan sebagai dasar dalam membangun model prediksi. Setelah *dataset* berhasil dimuat, proses dilanjutkan dengan inisialisasi list kosong dengan nama *tree_list*. List ini berfungsi dalam menyimpan seluruh pohon keputusan yang akan dibuat selama proses pembentukan model *Random Forest* berlangsung.

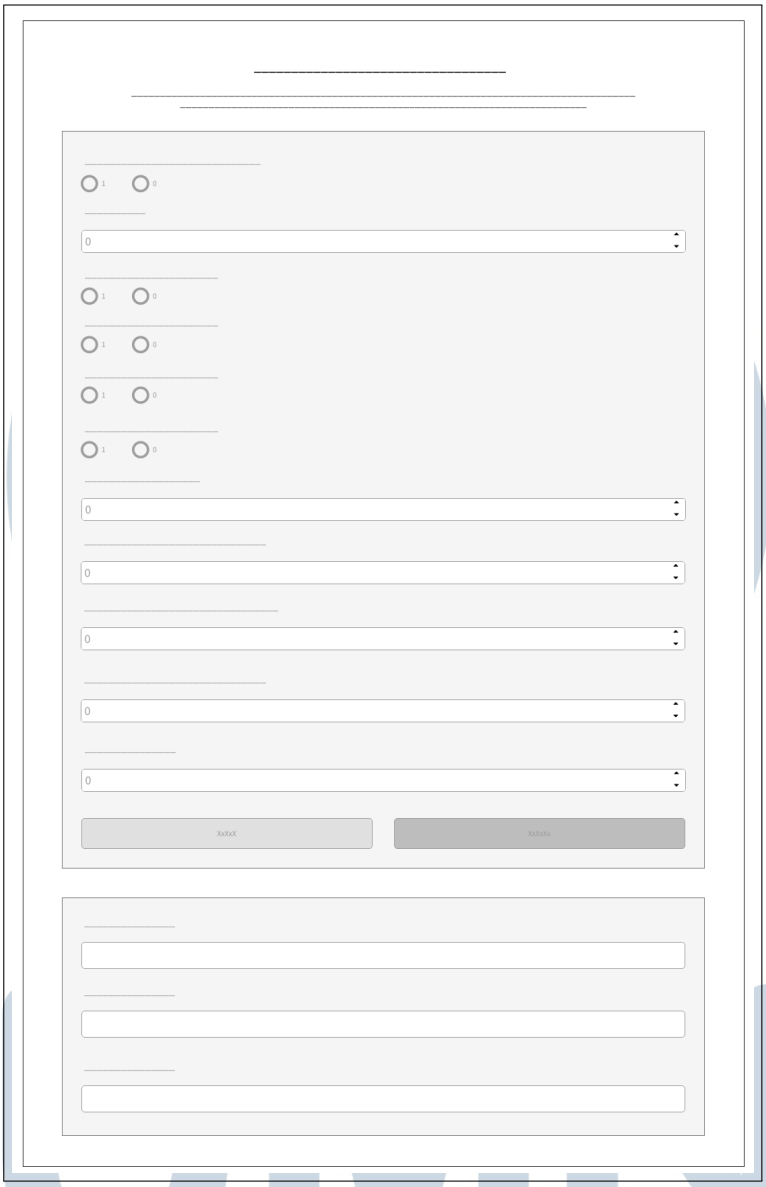
Setelah dilakukan inisialisasi, program akan memasuki proses iterasi sebanyak jumlah maksimum pohon yang dibentuk, disebut dengan *Max_tree*. Dalam proses iterasi, akan dilakukan pemilihan fitur secara acak dari *dataset* utama D*_all* dan akan disimpan ke dalam variabel D*_current*. Proses ini dilakukan agar setiap pohon dilatih dengan kombinasi fitur yang berbeda, sehingga memperbanyak keberagaman pohon dalam hutan keputusan.

Dari fitur yang telah diseleksi, akan diambil beberapa baris data secara acak untuk dijadikan *dataset* pelatihan bagi satu pohon keputusan, dan data akan disimpan ke dalam variabel D. Selanjutnya, pohon keputusan akan dibangun berdasarkan dalam D, melalui proses pemisahan data sesuai dengan nilai fitur hingga mencapai kondisi akhir untuk menentukan klasifikasi.

Pohon yang telah terbentuk dimasukkan ke dalam *tree_list*. Setelah satu iterasi selesai, sistem akan memeriksan jumlah pohon yang dibuat telah mencapai nilai *Max_tree*. Jika belum mencapai maka proses akan kembali ke awal untuk membentuk pohon berikutnya. Jika sudah, maka proses berakhir.

3.4.2 Wireframe

Wireframe yaitu rancangan visual sederhana yang memberikan gambaran dari tata letak (*layout*) dan struktur dasar dari sebuah halaman *website*, tanpa memperhatikan warna, *font*, atau elemen visual lainnya. *Wireframe* untuk *input* data manual dapat dilihat melalui Gambar 3.6.



Gambar 3.6. Wireframe Input Data Manual

Wireframe pada Gambar 3.6 menunjukkan gambaran kasar dari tampilan sederhana berupa *website* untuk *input* data secara manual. Penempatan elemen-elemen pada *wireframe* disusun secara terstruktur, dimulai dari judul halaman, deskripsi singkat, kolom isian data pengguna, hingga tombol aksi seperti *Submit* dan *Clear*, hingga kolom hasil *output*. Tujuannya adalah untuk memberikan pengalaman pengguna (*user experience*) yang sederhana namun informatif dalam proses pengisian data kesehatan secara manual dan hasil prediksi penyakit jantung.