

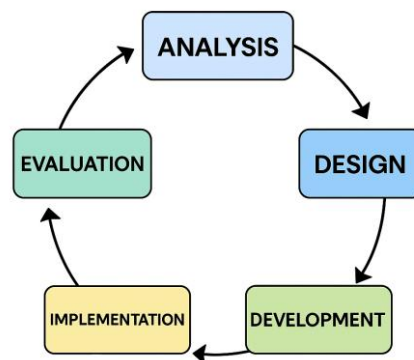
BAB III

METODE PENELITIAN

Bab ini menjelaskan pendekatan dan langkah-langkah yang digunakan dalam pelaksanaan penelitian.

3.1 Jenis Penelitian

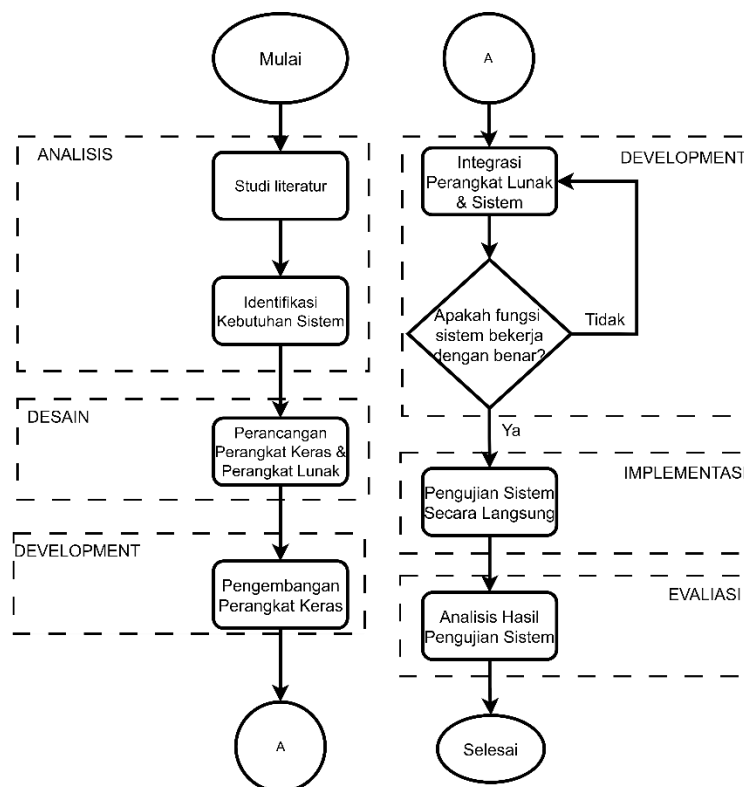
Jenis penelitian ini menggunakan metode *Research and Development* (R&D) yang dipilih karena sesuai dengan yang dibutuhkan untuk merancang, mengembangkan, dan mengevaluasi sistem pemantauan kesehatan jantung berbasis sinyal ECG dan PPG menggunakan *machine learning* dan deep learning, dengan fokus pada deteksi aritmia. Metode ini dipilih karena bersifat iteratif dan sistematis, sehingga sesuai untuk menghasilkan produk inovatif yang dapat diterapkan di dunia medis. R&D merupakan proses logis dan terstruktur yang mencakup identifikasi permasalahan, pembuatan prototipe, hingga penyebaran hasil kepada pengguna. Sejalan dengan pendapat [48], penelitian ini mengacu pada pengembangan model ADDIE yang terdiri dari lima tahap: Analisis, Perancangan, Pengembangan, Implementasi, dan Evaluasi, karena model ini memberikan struktur kerja yang jelas dan berorientasi pada hasil. [53]. Pada Gambar 3.1 menunjukkan siklus ADDIE yang terdiri dari lima tahap utama yang saling terhubung.



Gambar 3. 1 Desain Penelitain

3.2 Alur Penelitian

Alur penelitian berfungsi sebagai panduan utama bagi peneliti agar setiap tahapan yang dilakukan berjalan terarah dan sesuai tujuan. Dalam penelitian ini, alur disusun berdasarkan model pengembangan ADDIE yang bersifat sistematis dan saling berkesinambungan antar tahapannya. Rangkaian proses tersebut divisualisasikan melalui diagram alir pada Gambar 3.2, yang menunjukkan hubungan logis antar langkah dalam pelaksanaan penelitian.



Gambar 3. 2 Alur Penelitian

3.2.1 Analysis (Analisis Kebutuhan)

Tahap awal penelitian ini adalah analisis kebutuhan untuk membangun sistem deteksi aritmia. Proses ini dimulai dengan melakukan studi literatur secara mendalam untuk memahami teknologi-teknologi kunci seperti *Photoplethysmography* (PPG), *Electrocardiogram* (ECG), *Internet of Things* (IoT), dan *Machine learning* (ML). Selanjutnya, dilakukan identifikasi spesifikasi sistem yang esensial, mencakup kebutuhan perangkat keras dan perangkat lunak agar

Muhammad Wildan Alfarizy, 2025

DETEKSI ARITMIA JANTUNG BERBASIS INTEGRASI SINYAL ECG DAN PPG DENGAN PENDEKATAN MACHINE LEARNIGN DAN DEEP LEARNING DALAM SISTEM IOT

Universitas Pendidikan Indonesia | repository.upi.edu | perpustakaan.upi.edu

mampu mendeteksi aritmia secara akurat. Terakhir, dilakukan analisis dan seleksi dataset publik yang relevan dan berkualitas untuk digunakan dalam melatih model deteksi yang akan dikembangkan.

3.2.2 Design (Perancangan Sistem)

Setelah kebutuhan sistem dianalisis, tahap selanjutnya adalah perancangan sistem. Pada fase ini, dilakukan perancangan arsitektur sistem IoT secara menyeluruh, yang mengintegrasikan tiga komponen utama: perangkat keras (*hardware*) untuk akuisisi data, infrastruktur server untuk pemrosesan dan penyimpanan data, serta aplikasi pengguna. Selanjutnya, dilakukan perancangan pendekatan algoritma *machine learning* dengan mengevaluasi dan memilih model yang paling sesuai, seperti *Convolutional Neural Network* (CNN), *Support Vector Machine* (SVM), atau LightGBM, untuk klasifikasi sinyal jantung. Terakhir, dibuat desain antarmuka pengguna (*UI/UX*) yang intuitif dan fungsional untuk aplikasi *monitoring*, guna memastikan pengguna dapat memantau kondisi jantung dengan mudah.

3.2.3 Development (Pengembangan Sistem)

Pada tahap pengembangan, realisasi desain sistem dimulai. Langkah pertama adalah melakukan integrasi antara sensor PPG dan ECG dengan platform IoT, dalam hal ini menggunakan mikrokontroler ESP32 untuk akuisisi dan pengiriman data. Selanjutnya, dilakukan pelatihan ketiga model *machine learning*, yaitu *Convolutional Neural Network* (CNN), *Support Vector Machine* (SVM), dan LightGBM. Proses ini menggunakan dataset yang telah dipilih dan diproses pada tahap analisis kebutuhan. Terakhir, dilakukan pengembangan aplikasi berbasis *web* yang berfungsi sebagai *dashboard* untuk visualisasi hasil deteksi aritmia, sehingga data dapat ditampilkan secara informatif kepada pengguna.

3.2.4 Implementation (Implementasi dan Pengujian)

Tahap terakhir adalah implementasi dan pengujian untuk mengevaluasi kinerja sistem secara keseluruhan. Langkah pertama adalah menguji sistem yang telah dikembangkan secara fungsional menggunakan data uji yang telah dipisahkan

Muhammad Wildan Alfarizy, 2025

DETEKSI ARITMIA JANTUNG BERBASIS INTEGRASI SINYAL ECG DAN PPG DENGAN PENDEKATAN MACHINE LEARNING DAN DEEP LEARNING DALAM SISTEM IOT

Universitas Pendidikan Indonesia | repository.upi.edu | perpustakaan.upi.edu

dari dataset. Selanjutnya, dilakukan perbandingan performa dari ketiga model *machine learning* (CNN, SVM, dan LightGBM) untuk menentukan model mana yang memiliki akurasi, presisi, dan *recall* terbaik dalam mendeteksi aritmia. Sebagai langkah final, dilakukan validasi hasil deteksi sistem dengan cara membandingkannya terhadap label atau diagnosis asli yang tersedia pada dataset untuk memastikan keandalan dan akurasi sistem.

3.3.5 *Evaluation* (Evaluasi dan Analisis Hasil)

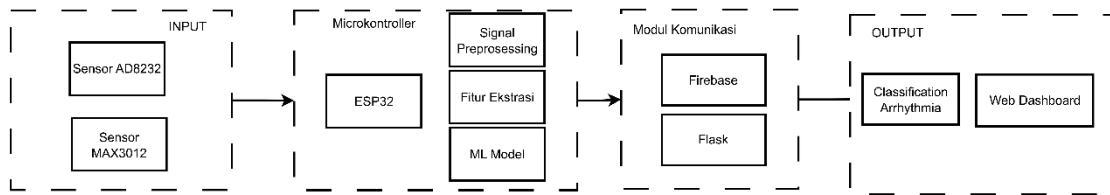
Pada tahap evaluasi, dilakukan analisis mendalam terhadap hasil pengujian. Proses ini diawali dengan menilai kinerja sistem secara kuantitatif berdasarkan metrik evaluasi yang telah ditetapkan sebelumnya, seperti matriks konfusi (*confusion matrix*). Selanjutnya, dilakukan penghitungan nilai akurasi, sensitivitas, dan spesifisitas secara spesifik untuk setiap model guna mengukur kemampuannya dalam mengklasifikasi data dengan benar. Berdasarkan hasil perbandingan metrik-metrik tersebut, langkah terakhir adalah menentukan model terbaik yang menunjukkan performa paling unggul untuk deteksi aritmia.

3.3 Perancangan Sistem

Perancangan sistem merupakan tahapan krusial dalam pengembangan penelitian ini, yang bertujuan untuk memastikan seluruh komponen dapat terintegrasi dan berfungsi sesuai dengan tujuan penelitian. Proses perancangan ini mencakup aspek arsitektur sistem secara keseluruhan, perancangan *hardware* (alat), dan perancangan *software*.

3.3.1 Arsitektur Sistem

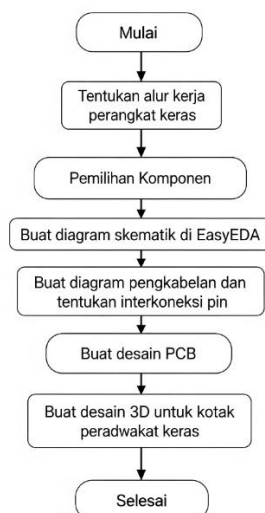
Arsitektur sistem secara keseluruhan dirancang untuk memberikan gambaran komprehensif mengenai komponen-komponen utama dan interaksi antar-komponen dalam sistem ini. Arsitektur sistem disajikan pada Gambar 3.3.



Gambar 3. 3 *Arsitektur diagram sistem deteksi aritmia*

Gambar 3.3 mengilustrasikan komponen utama sistem yang terdiri dari unit pengukuran (AD8232 dan MAX30102), unit kendali (ESP32), platform *cloud* (Firebase), dan *web dashboard*. Masing-masing komponen memiliki peran spesifik dan saling berinteraksi dalam alur data yang terstruktur. Pada sistem ini, sensor AD8232 dan MAX30102 digunakan untuk mendeteksi detak jantung, dengan ESP32 sebagai pusat pemrosesan dan pengirim data. Hasil pembacaan sensor dikirim secara *real-time* ke Firebase, yang berfungsi sebagai basis data berbasis *cloud*. Dari Firebase, data diteruskan ke Google Sheet untuk pencatatan otomatis dan ke backend Flask untuk analisis lanjutan, sebelum akhirnya divisualisasikan melalui *Web Dashboard*. *Dashboard* ini memfasilitasi pemantauan kondisi pasien secara langsung oleh tenaga medis maupun pengguna terkait.

3.3.2 Perancangan *Hardware*



Gambar 3. 4 *Flowchart Perancangan Hardware*

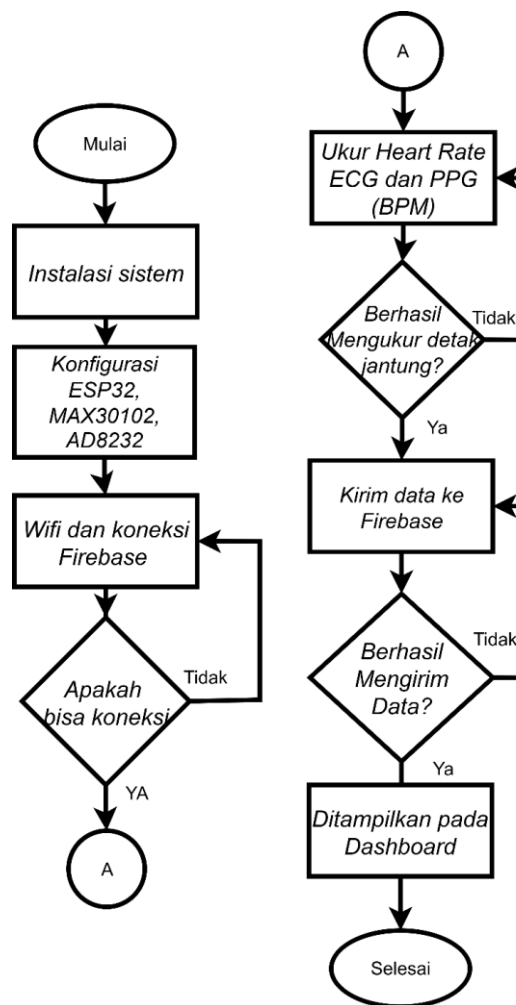
Muhammad Wildan Alfarizy, 2025

DETEKSI ARITMIA JANTUNG BERBASIS INTEGRASI SINYAL ECG DAN PPG DENGAN PENDEKATAN MACHINE LEARNIGN DAN DEEP LEARNING DALAM SISTEM IOT

Universitas Pendidikan Indonesia | repository.upi.edu | perpustakaan.upi.edu

Gambar 3.4 menyajikan *Flowchart* perancangan *hardware* yang dirancang secara sistematis dalam konteks penelitian ini. Diagram alir tersebut memberikan gambaran bahwa setiap langkah dalam proses perancangan dilakukan secara berurutan dan terstruktur, dengan tujuan memastikan integrasi antar komponen serta fungsionalitas sistem berjalan secara optimal.

3.3.2.1 Flowchart Hardware system



Gambar 3. 5 Alur kerja sistem deteksi aritmia

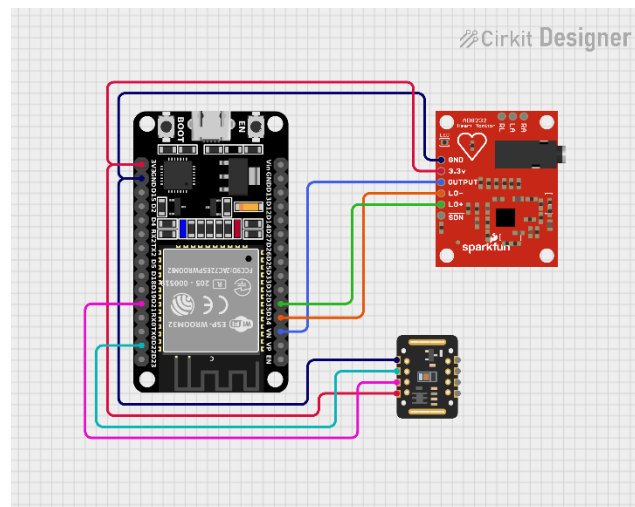
Gambar 3.5 memperlihatkan alur kerja sistem deteksi aritmia berbasis ESP32 yang mengintegrasikan sensor MAX30102 untuk detak jantung dan SpO₂, sensor AD8232 untuk sinyal ECG, serta mekanisme pengiriman data ke *Firestore Realtime Database*. Proses dimulai dari *System Initialization* dan konfigurasi perangkat

Muhammad Wildan Alfarizy, 2025
DETEKSI ARITMIA JANTUNG BERBASIS INTEGRASI SINYAL ECG DAN PPG DENGAN PENDEKATAN MACHINE LEARNING DAN DEEP LEARNING DALAM SISTEM IOT

Universitas Pendidikan Indonesia | repository.upi.edu | perpustakaan.upi.edu

keras, dilanjutkan dengan koneksi ke jaringan Wi-Fi dan Firebase. Setelah koneksi berhasil, sistem melakukan akuisisi data PPG dan ECG, kemudian mengirimkannya ke Firebase serta menampilkannya pada layar OLED sebagai umpan balik lokal. Flowchart ini menegaskan bagaimana seluruh proses berlangsung secara berurutan, otomatis, dan *real-time* sehingga mendukung pemantauan kesehatan pasien secara nirkabel dan terintegrasi.

3.3.2.2 Skema Rangkaian dan *Wiring Diagram*



Gambar 3. 6 *Wiring Diagram* sistem deteksi aritmia

Gambar 3.6 menunjukkan rangkaian sistem deteksi aritmia berbasis ESP32 yang menggabungkan sensor MAX30102 untuk detak jantung dan SpO2, serta sensor AD8232 untuk sinyal ECG. Sensor MAX30102 terhubung ke ESP32 melalui antarmuka I2C (GPIO21 dan GPIO22), sedangkan AD8232 terhubung melalui pin analog dan digital. ESP32 berfungsi sebagai mikrokontroler utama yang memproses data dari sensor, lalu mengirimkannya secara *real-time* ke *Firestore Realtime Database* melalui koneksi *Wi-Fi*. Dengan demikian, data kesehatan pasien dapat dipantau secara jarak jauh melalui *platform cloud*.

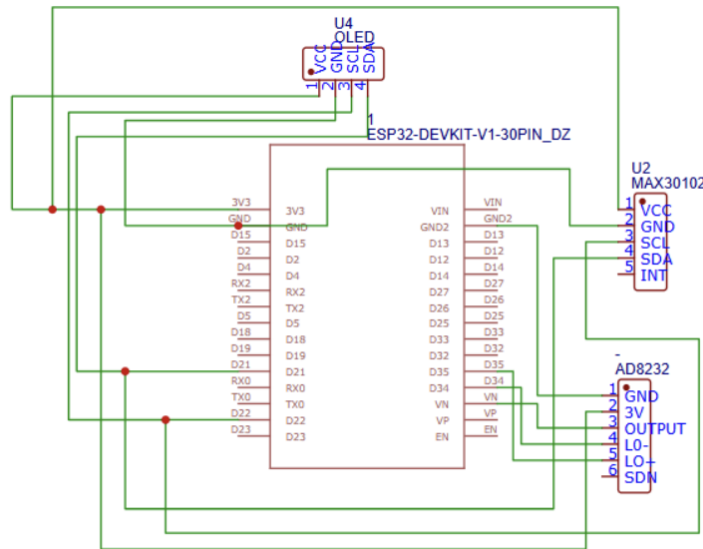
Untuk mendukung visualisasi lokal secara langsung, sistem ini juga dilengkapi dengan modul OLED *display* yang menampilkan hasil pembacaan detak jantung dan kadar oksigen (SpO2) secara *real-time* pada perangkat. Integrasi OLED

Muhammad Wildan Alfarizy, 2025

DETEKSI ARITMIA JANTUNG BERBASIS INTEGRASI SINYAL ECG DAN PPG DENGAN PENDEKATAN MACHINE LEARNING DAN DEEP LEARNING DALAM SISTEM IOT

Universitas Pendidikan Indonesia | repository.upi.edu | perpustakaan.upi.edu

display ini memungkinkan pengguna melihat informasi tanpa perlu akses ke *cloud* secara terus-menerus.



Gambar 3. 7 Skematik Rangkaian

Melengkapi gambar sebelumnya, Gambar 3.7 menunjukkan skematik rangkaian sistem secara detail, termasuk koneksi antar pin pada ESP32, sensor MAX30102, sensor AD8232, dan OLED. Dalam skematik ini terlihat bahwa sensor MAX30102 dan OLED menggunakan jalur komunikasi I2C yang sama (SDA pada GPIO21 dan SCL pada GPIO22), sedangkan sinyal ECG dari AD8232 dibaca melalui pin analog GPIO36 (VP). Pin LO+ dan LO- dari AD8232 terhubung ke pin digital ESP32 untuk mendeteksi kondisi kabel elektroda.

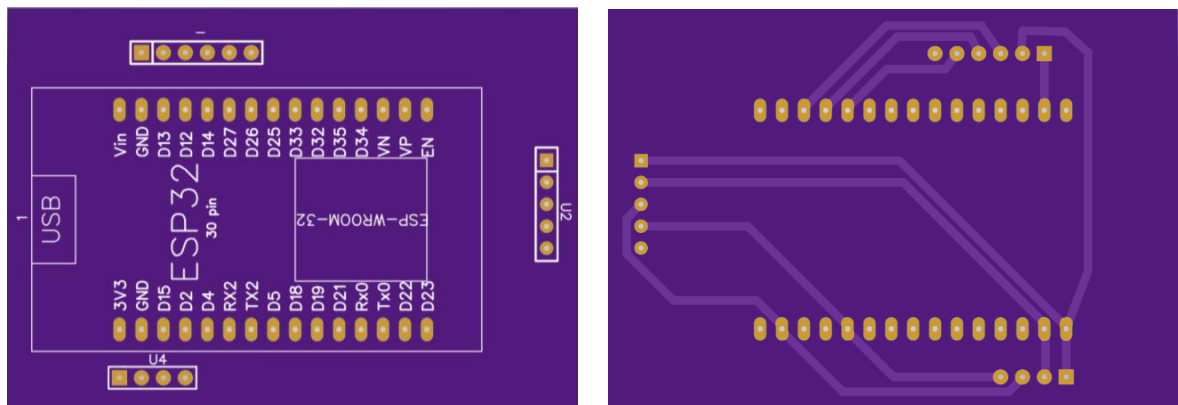
3.3.2.3 Desain PCB dan Desain Box

Setelah seluruh komponen berhasil berjalan sesuai perancangan, rangkaian dialihkan ke dalam bentuk *Printed Circuit Board* (PCB) untuk memastikan keandalan dan kestabilan koneksi, serta untuk memperbaiki tata letak fisik komponen agar lebih terorganisir. Proses ini juga bertujuan untuk mengurangi potensi gangguan yang timbul akibat koneksi kabel longgar atau *noise* antarkabel.

Muhammad Wildan Alfarizy, 2025

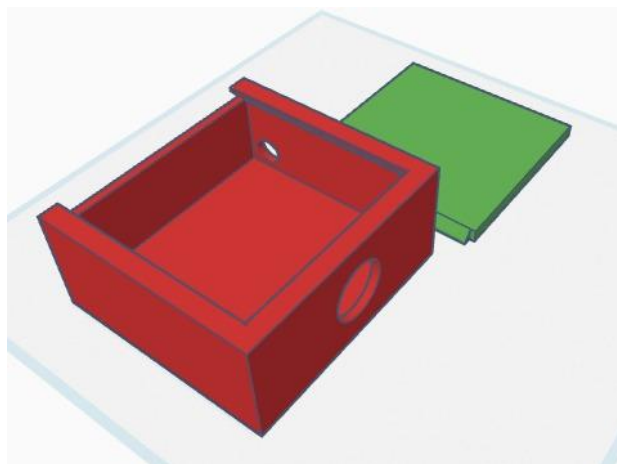
DETEKSI ARITMIA JANTUNG BERBASIS INTEGRASI SINYAL ECG DAN PPG DENGAN PENDEKATAN MACHINE LEARNIGN DAN DEEP LEARNING DALAM SISTEM IOT

Universitas Pendidikan Indonesia | repository.upi.edu | perpustakaan.upi.edu



Gambar 3. 8 *Desain PCB*

Desain PCB pada Gambar 3.8 merupakan pengembangan dari *wiring diagram* sebelumnya, dengan tujuan merapikan koneksi antar komponen dan meningkatkan kestabilan sistem. Layout PCB disusun berdasarkan konfigurasi pin yang telah diuji pada tahap perakitan *breadboard*, sehingga memastikan kompatibilitas dan fungsi setiap komponen. Sebagai kelanjutan, dibuat pula *desain box 3D* untuk menampung mikrokontroler, sensor, dan modul display, yang berfungsi melindungi rangkaian sekaligus meningkatkan portabilitas serta kerapihan perangkat.



Gambar 3. 9 *Desain Box 3D*

Gambar 3.9 menampilkan *desain* tiga dimensi (*3D design*) dari *enclosure* perangkat *monitoring* kesehatan, yang berfungsi menampung serta melindungi komponen inti seperti ESP32, sensor MAX30102, dan AD8232. Desain ini dibuat

Muhammad Wildan Alfarizy, 2025

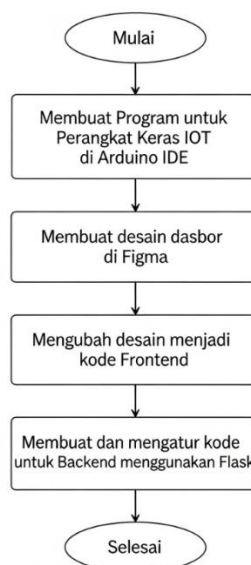
DETEKSI ARITMIA JANTUNG BERBASIS INTEGRASI SINYAL ECG DAN PPG DENGAN PENDEKATAN MACHINE LEARNING DAN DEEP LEARNING DALAM SISTEM IOT

Universitas Pendidikan Indonesia | repository.upi.edu | perpustakaan.upi.edu

setelah validasi fungsional melalui breadboard dan wiring diagram, dengan tujuan menjaga kinerja sistem sekaligus memudahkan pemasangan serta perawatan. Box terdiri dari badan utama dan penutup yang dapat dibuka-tutup tanpa alat tambahan, dilengkapi bukaan untuk koneksi sensor, ventilasi, dan port daya sesuai kebutuhan IoT. Selain proteksi, desain kompak ini juga meningkatkan estetika dan portabilitas, sehingga perangkat lebih layak dipakai dalam pemantauan kesehatan real-time, baik di klinis maupun penggunaan mandiri.

3.3.3 Perancangan *Software*

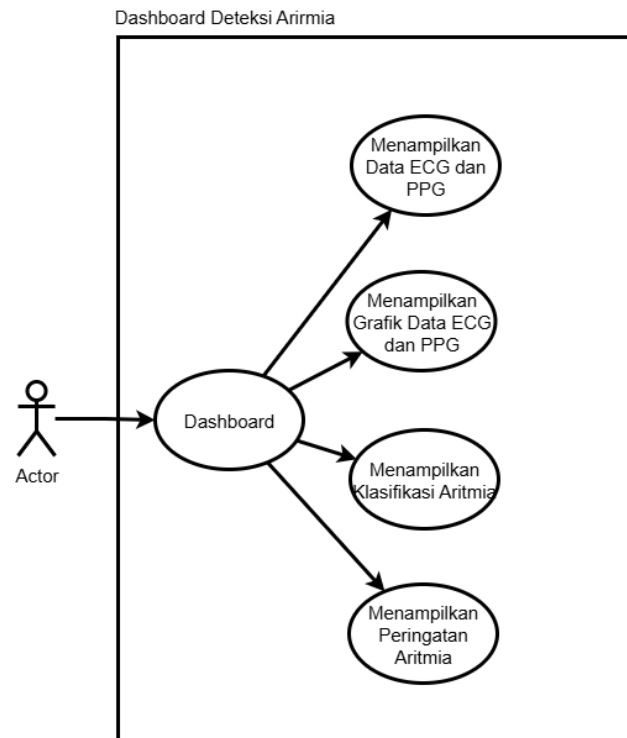
Perancangan *software* bertujuan untuk memastikan bahwa proses pengambilan data, pemrosesan, penyimpanan, dan penyajian informasi dapat berlangsung secara terintegrasi dan efisien dalam satu sistem yang saling terhubung. Tahapan ini menjadi krusial untuk menjembatani antara perangkat keras yang mengumpulkan data biometrik dan sistem cloud yang menampilkannya kepada pengguna akhir.



Gambar 3. 10 *Perancangan Software*

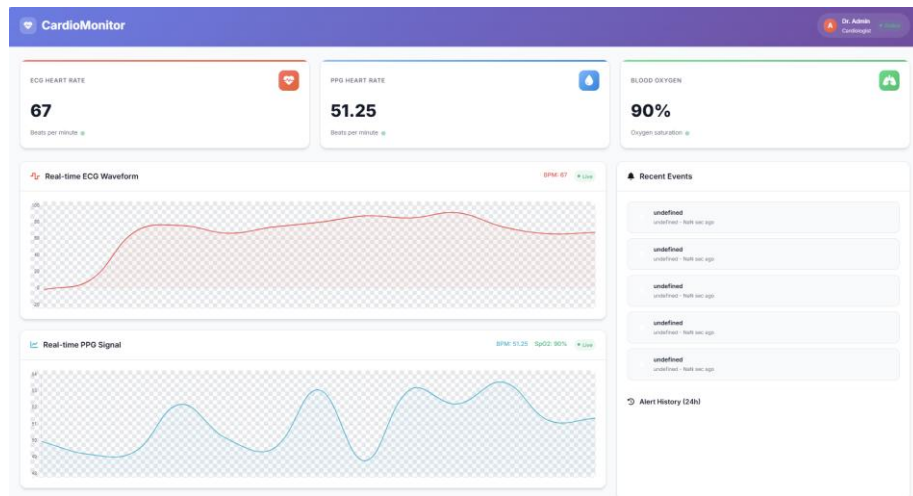
Sebagaimana ditunjukkan pada Gambar 3.9 , perancangan perangkat lunak dibagi ke dalam beberapa komponen utama. Tahap pertama adalah perancangan

program utama (*main code*) pada mikrokontroler ESP32, yang bertanggung jawab untuk membaca data dari sensor (MAX30102 dan AD8232), memprosesnya, serta mengirimkan data secara *real-time* ke Firebase melalui koneksi Wi-Fi.



Gambar 3. 11 Use case diagram

Use case diagram pada Gambar 3.11 menggambarkan interaksi antara pengguna, dan sistem. Aktor utama seperti pasien dan dokter dapat melakukan fungsi seperti melihat data kesehatan, memantau riwayat pengukuran, dan menerima notifikasi anomali. Sementara itu, sistem bertugas mengolah data dari sensor, menyimpannya di Firebase, dan menyajikannya dalam bentuk visualisasi pada *dashboard*.



Gambar 3. 12 Desain Dashboard

Selanjutnya dilakukan perancangan antarmuka pengguna (*user interface*) menggunakan *platform desain Figma*. Desain ini bertujuan untuk menciptakan tampilan *dashboard* yang informatif dan mudah dipahami oleh pengguna, dengan fokus pada visualisasi detak jantung, SpO2, dan sinyal ECG/PPG. Setelah desain antarmuka selesai, dilakukan proses *slicing*, yaitu tahap pengubahan desain visual dari Figma menjadi format kode HTML, CSS, dan JavaScript agar dapat diintegrasikan ke dalam *web dashboard*. Proses ini memastikan bahwa elemen visual dapat diterjemahkan ke dalam bentuk antarmuka fungsional.

Tahap terakhir adalah integrasi *frontend* dengan *backend* menggunakan *framework Flask*. Flask berfungsi sebagai *server backend* yang mengelola pengambilan data dari Firebase, melakukan pengolahan jika diperlukan, serta mengirimkan data ke antarmuka *web* secara dinamis. Dengan demikian, seluruh sistem *software* mampu bekerja secara utuh dan responsif dalam mendukung pemantauan kesehatan secara *real-time*.

3.3.4 Perancangan Database

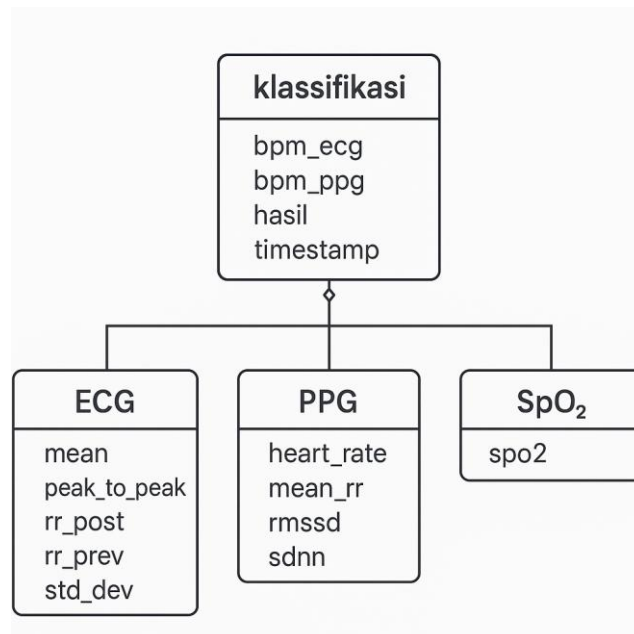
Tahap perancangan basis data merupakan fase krusial yang memastikan seluruh data hasil pengukuran dari perangkat keras dapat diorganisasi secara sistematis, terlindungi dengan baik, serta mudah diakses oleh komponen sistem

Muhammad Wildan Alfarizy, 2025

DETEKSI ARITMIA JANTUNG BERBASIS INTEGRASI SINYAL ECG DAN PPG DENGAN PENDEKATAN MACHINE LEARNING DAN DEEP LEARNING DALAM SISTEM IOT

Universitas Pendidikan Indonesia | repository.upi.edu | perpustakaan.upi.edu

lainnya, seperti modul *backend* maupun *dashboard* visualisasi. Dalam penelitian ini, sistem memanfaatkan *Firebase Realtime Database* yang menggunakan model penyimpanan NoSQL dengan struktur JSON tree. Pemilihan basis data berbasis NoSQL didasari oleh beberapa alasan mendasar, di antaranya fleksibilitasnya dalam menangani data semi-terstruktur, kemampuan melakukan pembaruan data secara *real-time*, serta kemudahan integrasinya dengan perangkat *Internet of Things* yang menjadi bagian integral dari sistem penelitian ini. Pertimbangan tersebut memastikan bahwa arsitektur basis data mampu mendukung kebutuhan sistem yang dinamis dan responsif terhadap perubahan data.



Gambar 3. 13 bentuk Entity Relationship Diagram

Desain struktur basis data yang digunakan dalam penelitian ini terdiri atas dua entitas utama, yakni *live_data* dan klasifikasi, sebagaimana digambarkan pada Gambar 3.13 dalam bentuk *Entity Relationship Diagram* (ERD). Entitas *live_data* berfungsi untuk menyimpan informasi yang bersumber dari sensor, yang mencakup parameter data ECG seperti *mean*, *peak_to_peak*, *rr_post*, *rr_prev*, *rr_ratio*, dan *std_dev*. Sementara itu, untuk data PPG disimpan parameter seperti *heart_rate*, *mean_rr*, *rmssd*, dan *sdn*. Adapun entitas klasifikasi dirancang untuk merekam

Muhammad Wildan Alfarizy, 2025

DETEKSI ARITMIA JANTUNG BERBASIS INTEGRASI SINYAL ECG DAN PPG DENGAN PENDEKATAN MACHINE LEARNIGN DAN DEEP LEARNING DALAM SISTEM IOT

Universitas Pendidikan Indonesia | repository.upi.edu | perpustakaan.upi.edu

hasil analisis klasifikasi, termasuk nilai *beats per minute* (BPM) dari masing-masing sensor, keluaran klasifikasi aritmia, serta *timestamp* yang menandai waktu pengambilan data. Pemisahan fungsi antara kedua entitas ini bertujuan untuk menjaga keteraturan dan konsistensi struktur data, sekaligus memudahkan pengelolaan dan pengembangan sistem di masa mendatang

```
{
  "klasifikasi": {
    "bpm_ecg": 60,
    "bpm_ppg": 75,
    "hasil": "Normal",
    "timestamp": 1754329618.999643
  },
  "live_data": {
    "ecg": {
      "mean": 0,
      "peak_to_peak": 0,
      "rr_post": 320,
      "rr_prev": 320,
      "rr_ratio": 1,
      "std_dev": 0
    },
    "ppg": {
      "heart_rate": 90,
      "mean_rr": 666.66669,
      "rmssd": 33,
      "sdnn": 40
    },
    "spo2": 0
  }
}
```

Gambar 3. 14 *struktur JSON Tree*

Struktur penyimpanan yang diimplementasikan pada Firebase divisualisasikan dalam Gambar 3.14, yang menunjukkan representasi JSON tree di mana setiap catatan hasil pengukuran memiliki keterkaitan langsung antara data sensor dan hasil klasifikasinya. Pendekatan ini tidak hanya menjamin integrasi data yang efisien, tetapi juga memberikan kemudahan dalam proses analisis historis serta pemantauan kondisi kesehatan secara berkelanjutan. Dengan rancangan seperti ini, sistem dapat memberikan informasi yang cepat dan akurat kepada pengguna, sekaligus memperkuat reliabilitas hasil penelitian yang dilakukan.

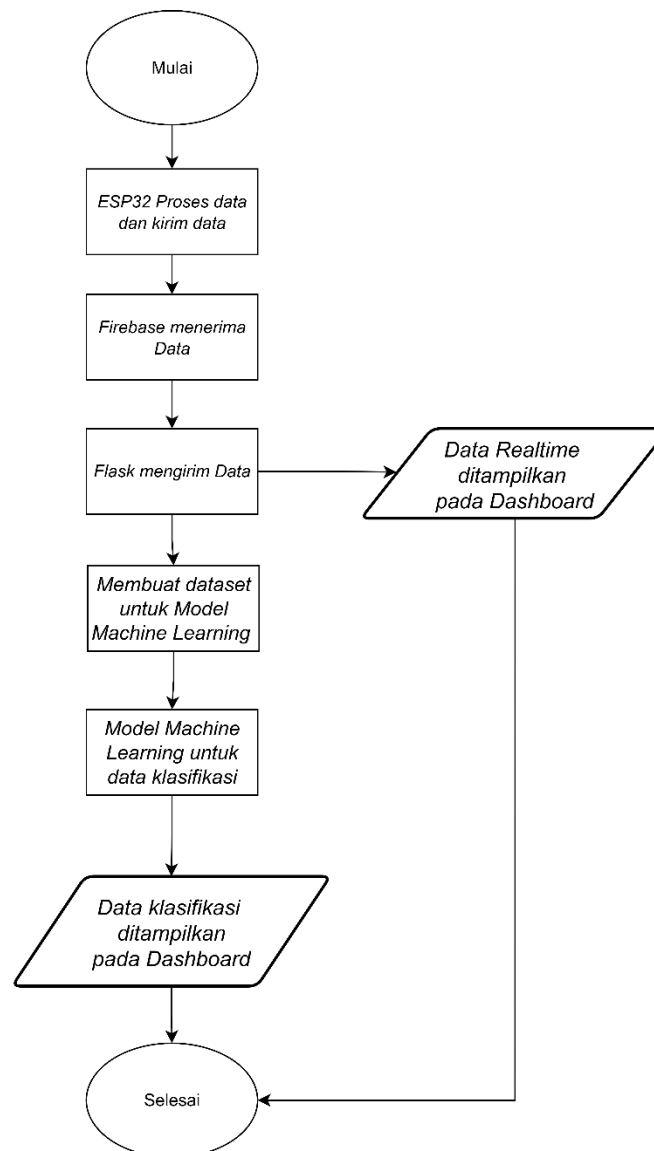
Muhammad Wildan Alfarizy, 2025

DETEKSI ARITMIA JANTUNG BERBASIS INTEGRASI SINYAL ECG DAN PPG DENGAN PENDEKATAN MACHINE LEARNIGN DAN DEEP LEARNING DALAM SISTEM IOT

Universitas Pendidikan Indonesia | repository.upi.edu | perpustakaan.upi.edu

3.3.5 Perancangan *Backend*

Backend dikembangkan menggunakan *framework Flask (Python)* yang berfungsi sebagai perantara antara *Firebase* dengan model klasifikasi dan tampilan *web dashboard*. Fungsi utama *server backend* seperti pada Gambar 3.12



Gambar 3. 15 Perancangan *Backend*

Seperti pada Gambar 3.12, Flask akan menarik data dari *Firebase* untuk ditampilkan ke *dashboard* dan untuk diproses ke model *machine learning*. Sebelum model bisa menghasilkan data klasifikasi, *flask* akan memproses terlebih dahulu

Muhammad Wildan Alfarizy, 2025

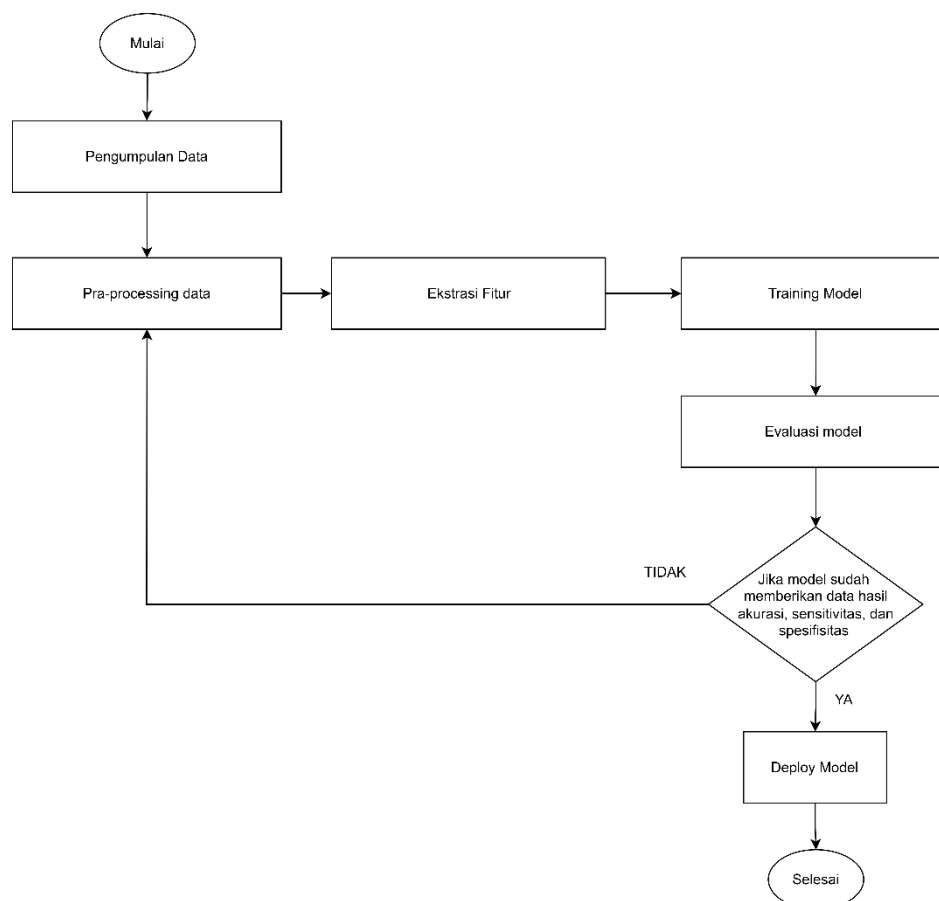
DETEKSI ARITMIA JANTUNG BERBASIS INTEGRASI SINYAL ECG DAN PPG DENGAN PENDEKATAN MACHINE LEARNIGN DAN DEEP LEARNING DALAM SISTEM IOT

Universitas Pendidikan Indonesia | repository.upi.edu | perpustakaan.upi.edu

data dari *Firestore* untuk diproses menjadi sebuah data *frame* untuk dijadikan *input* ke dalam model *machine learning*. Setelah itu, flask akan mengembalikan hasil klasifikasi untuk ditampilkan ke *dashboard*

3.4 Metode *Machine learning*

Penelitian ini memanfaatkan metode *machine learning* untuk membangun sistem klasifikasi otomatis yang mampu mendeteksi aritmia berdasarkan sinyal jantung. Pengembangan model dilakukan secara terstruktur, dimulai dari pengumpulan data ECG dan PPG, tahap *preprocessing*, ekstraksi fitur, hingga pelatihan dan evaluasi model klasifikasi seperti CNN, SVM, dan LightGBM. Seluruh tahapan ini divisualisasikan dalam Gambar 3.6, yang menggambarkan alur sistem dari input sinyal hingga output deteksi aritmia secara otomatis.



Gambar 3. 16 Diagram Alir Tahapan Pengembangan Model Klasifikasi

Muhammad Wildan Alfarizy, 2025

DETEKSI ARITMIA JANTUNG BERBASIS INTEGRASI SINYAL ECG DAN PPG DENGAN PENDEKATAN MACHINE LEARNIGN DAN DEEP LEARNING DALAM SISTEM IOT

Universitas Pendidikan Indonesia | repository.upi.edu | perpustakaan.upi.edu

3.4.1 Pemilihan Algoritma

Dalam penelitian ini, tiga algoritma *machine learning* dipilih untuk dievaluasi, yaitu *Convolutional Neural Network* (CNN), *Support Vector Machine* (SVM), dan *Light Gradient Boosting Machine* (LightGBM). Pemilihan ketiga algoritma ini didasarkan pada karakteristik unik dan keunggulan masing-masing dalam menangani data dan permasalahan klasifikasi yang kompleks.

3.4.1.1 CNN

Convolutional Neural Networks (CNN) telah muncul sebagai alat yang efektif dalam analisis sinyal *Electrocardiogram* (ECG) karena kemampuannya dalam mengekstraksi fitur secara otomatis dari data sekuensial. Kemampuan ini sangat penting dalam aplikasi medis, di mana analisis mendetail terhadap karakteristik sinyal diperlukan untuk diagnosis yang akurat. Arsitektur CNN, terutama ketika menggunakan konfigurasi seperti lapisan *konvolusi*, *batch normalization*, *max pooling*, dan *dropout*, memfasilitasi pengenalan pola spasial maupun temporal. Hal ini menjadikan CNN sangat mumpuni dalam membedakan antara sinyal ECG normal dan aritmia tanpa memerlukan proses ekstraksi fitur manual yang ekstensif [54], [55], [56]

Dalam penelitian ini, CNN diimplementasikan menggunakan *Conv1D* untuk mengolah data sinyal satu dimensi. Model terdiri dari dua blok *Conv1D-BatchNormalization-MaxPooling1D-Dropout* untuk mengekstraksi dan menstabilkan fitur, diikuti lapisan *Dense* untuk proses klasifikasi akhir. Untuk mengatasi ketidakseimbangan kelas, digunakan *class weighting* yang dihitung berdasarkan distribusi kelas pada data latih. Proses pelatihan dilakukan selama 15 *epoch* dengan *batch size* 128, serta memanfaatkan *validation data* dari data uji untuk memantau kinerja model selama pelatihan. Pendekatan ini memungkinkan CNN untuk secara langsung mempelajari representasi fitur dari sinyal ECG, memaksimalkan akurasi deteksi aritmia sekaligus meminimalkan risiko kehilangan informasi penting yang sering terjadi pada proses ekstraksi fitur manual.

3.4.1.2 SVM

Support Vector Machine (SVM) adalah algoritma yang diakui secara luas karena kemampuannya yang sangat efektif dalam mengatasi data berdimensi tinggi dan menemukan hyperplane optimal yang memisahkan kelas dengan margin maksimum. Sifat ini menjadikan SVM cukup robust terhadap *overfitting*, terutama dalam situasi di mana jumlah fitur melebihi jumlah data [57], [58]. Dengan kemampuan untuk menggeneralisasi, SVM dapat dipakai dalam berbagai domain, termasuk analisis medis, di mana struktur data sering kali kompleks dan tinggi dimensi dengan informasi relevan.

Pada implementasinya, untuk mempercepat proses komputasi *kernel* RBF, digunakan pendekatan *Nystroem kernel approximation* dengan 300 komponen. Hasil aproksimasi kernel ini kemudian diproses menggunakan LinearSVC dengan bobot kelas yang diseimbangkan (*class_weight='balanced'*) untuk menangani kemungkinan ketidakseimbangan data.

3.4.1.3 LightGBM

Light Gradient Boosting Machine (LightGBM) merupakan algoritma berbasis gradient boosting yang dipilih dalam penelitian ini karena efisiensinya dalam proses pelatihan, terutama pada dataset berskala besar. LightGBM menunjukkan performa yang lebih baik dibandingkan dengan algoritma *boosting* konvensional seperti XGBoost, dengan waktu pelatihan yang lebih singkat dan konsumsi memori yang lebih rendah, tanpa mengorbankan akurasi model [59]. Efisiensi ini dicapai melalui teknik pembelajaran berbasis *leaf-wise growth*, yang membangun pohon keputusan dengan cara lebih mendalam daripada metode tradisional yang lebih bersifat *level-wise*. Selain itu, LightGBM juga memanfaatkan histogram-based decision tree, yang mempercepat proses pembuatan model dengan mengurangi kompleksitas komputasi yang diperlukan untuk membagi data.

Untuk mengatasi ketidakseimbangan kelas pada data, digunakan parameter *scale_pos_weight* yang dihitung berdasarkan rasio jumlah sampel kelas mayoritas terhadap kelas minoritas. Selain itu, proses pelatihan juga memanfaatkan parameter

Muhammad Wildan Alfarizy, 2025

DETEKSI ARITMIA JANTUNG BERBASIS INTEGRASI SINYAL ECG DAN PPG DENGAN PENDEKATAN MACHINE LEARNING DAN DEEP LEARNING DALAM SISTEM IOT

Universitas Pendidikan Indonesia | repository.upi.edu | perpustakaan.upi.edu

eval_set untuk memantau metrik *logloss* pada data uji secara berkala, yang membantu dalam memantau kinerja model selama proses pelatihan. Dengan pendekatan ini, LightGBM mampu memanfaatkan bobot kelas (*class weighting*) untuk memperbaiki sensitivitas model terhadap kelas minoritas, sekaligus mempertahankan kecepatan pelatihan dan efisiensi memori yang menjadi ciri khasnya.

3.4.2 Sumber dan Pra-pemrosesan Data

Bagian ini menguraikan secara rinci mengenai asal-usul data yang dijadikan objek dalam penelitian, termasuk jenis dan sumber datanya. Selain itu, dijelaskan pula langkah-langkah sistematis yang ditempuh untuk melakukan praproses data, guna memastikan bahwa data tersebut berada dalam kondisi yang optimal sebelum digunakan dalam tahapan pelatihan dan pengujian algoritma *machine learning*. Setiap proses persiapan data ini memiliki peran penting dalam menjamin validitas hasil serta meningkatkan performa model yang dikembangkan.

3.4.2.1 Sumber Data

Penelitian ini memanfaatkan sinyal ECG dari *MIT-BIH Arrhythmia Database* yang tersedia di PhysioNet. Dataset ini dipilih karena validitasnya tinggi, telah diverifikasi oleh tenaga medis, dan banyak digunakan dalam studi kardiologi. MIT-BIH berisi lebih dari 48 rekaman ECG dua kanal dari 47 subjek, masing-masing berdurasi ± 30 menit dengan resolusi 360 Hz, sehingga representatif untuk pengembangan model machine learning[60].

Selain itu, penelitian juga menggunakan sinyal PPG sebagai sumber tambahan, mengingat sifatnya non-invasif dan kompatibel dengan perangkat *wearable IoT*. Data eksperimen diperoleh dari sensor ECG (AD8232) dan sensor PPG (MAX30102) yang terhubung dengan ESP32, lalu dikirim ke *Firestore Realtime Database* sebagai input model klasifikasi. Pendekatan ini membuktikan kemampuan sistem dalam mendeteksi aritmia secara *real-time* dan berkesinambungan sesuai konsep **IoT** untuk pemantauan kesehatan.

3.4.2.2 Pra-pemrosesan Sinyal

Pra-pemrosesan sinyal merupakan tahap krusial dalam pengolahan data biosinyal, yang bertujuan untuk meningkatkan kualitas sinyal mentah sebelum dimasukkan ke dalam model pembelajaran mesin. Pada penelitian ini, sinyal *Electrocardiogram* (ECG) dan *photoplethysmogram* (PPG) yang dihasilkan oleh sensor AD8232 dan MAX30102 terlebih dahulu diproses melalui serangkaian tahapan untuk mengurangi gangguan (*noise*) serta menyeragamkan skala data. Prosedur ini tidak hanya penting dalam menjaga keakuratan model, tetapi juga mendukung konsistensi dan kestabilan selama proses pelatihan. Tahapan-tahapan pra-pemrosesan yang dilakukan dijelaskan secara rinci pada subbagian berikut:

3.4.2.3 Filtering

Langkah pertama dalam pra-pemrosesan adalah penyaringan sinyal untuk menghilangkan komponen frekuensi yang tidak diinginkan. Pada sinyal ECG, digunakan band-pass filter dengan rentang frekuensi antara 0.5 Hz hingga 40 Hz, yang dirancang untuk menyingkirkan berbagai bentuk gangguan seperti baseline wander, artefak gerakan (*motion artifacts*), dan interferensi dari jaringan listrik. Rentang ini dipilih sesuai dengan frekuensi fisiologis aktivitas jantung yang relevan, sehingga informasi utama tetap terjaga.

Sementara itu, untuk sinyal PPG yang diperoleh dari sensor MAX30102, digunakan *butter_bandpass_filter* atau Filter *Butterworth Bandpass* dengan frekuensi cutoff di 4 Hz. Filter ini dirancang untuk mempertahankan komponen utama dari gelombang denyut jantung, sekaligus mereduksi noise berfrekuensi tinggi yang dapat mengganggu ketepatan identifikasi puncak sinyal.

3.4.2.4 Segmentasi

Setelah proses penyaringan selesai, sinyal dibagi menjadi potongan-potongan berdurasi tetap, yakni 5 detik untuk setiap segmen. Segmentasi ini dilakukan untuk menjamin bahwa seluruh data memiliki panjang waktu dan struktur yang konsisten, sehingga dapat diproses secara seragam oleh model *machine learning*. Konsistensi ini sangat penting terutama dalam konteks model berbasis *Convolutional Neural*

Muhammad Wildan Alfarizy, 2025

DETEKSI ARITMIA JANTUNG BERBASIS INTEGRASI SINYAL ECG DAN PPG DENGAN PENDEKATAN MACHINE LEARNING DAN DEEP LEARNING DALAM SISTEM IOT

Universitas Pendidikan Indonesia | repository.upi.edu | perpustakaan.upi.edu

Network (CNN), yang membutuhkan dimensi input tetap agar dapat melakukan konvolusi dengan efektif. Dengan pendekatan segmentasi ini, setiap unit data yang dianalisis berada dalam rentang waktu yang representatif terhadap satu siklus denyut jantung.

3.4.2.5 Normalisasi

Langkah berikutnya adalah normalisasi, yang bertujuan untuk menyeragamkan skala nilai pada setiap segmen sinyal. Proses ini dilakukan menggunakan teknik Z-score normalization, yaitu mengubah setiap nilai dalam segmen menjadi distribusi standar dengan rata-rata nol ($\mu = 0$) dan deviasi standar satu ($\sigma = 1$). Normalisasi sangat penting dalam konteks pembelajaran mesin karena membantu menghindari bias akibat perbedaan skala antar fitur, serta meningkatkan stabilitas dan kecepatan konvergensi selama proses pelatihan model.

3.4.2.6 Ekstraksi Fitur

Pada tahap ekstraksi fitur, setelah sinyal selesai melalui proses filtering, segmentasi, dan normalisasi, dilakukan pengambilan sejumlah fitur penting yang memiliki nilai diagnostik. Fitur-fitur ini dipilih karena mampu merepresentasikan kondisi fisiologis subjek sekaligus memberikan parameter numerik yang relevan sebagai dasar dalam melakukan klasifikasi ritme jantung.

3.4.3 Pelatihan Model

Proses pelatihan model dalam penelitian ini dilakukan secara sistematis dengan beberapa tahapan penting. Pertama, dataset dibagi menjadi tiga bagian, yakni 70% untuk data latih, 15% untuk data validasi, dan 15% sisanya untuk data uji. Pembagian ini bertujuan untuk memastikan bahwa model dapat belajar secara efektif serta dievaluasi secara objektif pada data yang belum pernah dilihat sebelumnya.

Selanjutnya, dilakukan pendekatan pelatihan untuk masing-masing algoritma. Pada model *Convolutional Neural Network* (CNN), digunakan arsitektur jaringan yang terdiri dari beberapa lapisan konvolusi dan pooling yang dilanjutkan dengan lapisan

fully connected. Struktur ini dirancang untuk secara bertahap mengekstraksi dan mengabstraksi fitur dari data input.

Untuk model *Support Vector Machine* (SVM), digunakan kernel *Radial Basis Function* (RBF) yang dikenal efektif dalam menangani hubungan *non-linear* antar fitur, sehingga meningkatkan kemampuan klasifikasi pada data yang kompleks. Sementara itu, pada model *Light Gradient Boosting Machine* (LightGBM), dilakukan penyesuaian (*tuning*) terhadap beberapa parameter penting seperti *n_estimators*, *learning_rate*, dan *num_leaves* guna memperoleh performa model yang optimal. Dalam hal optimisasi dan fungsi kehilangan (*loss function*), model CNN menggunakan algoritma optimisasi Adam yang terkenal akan efisiensi dan kestabilannya dalam konvergensi. Adapun fungsi loss yang digunakan adalah *Binary Cross-Entropy*, yang sesuai untuk menangani permasalahan klasifikasi biner secara efektif.

3.4.4 Metrik Evaluasi

Untuk mengevaluasi kinerja dari setiap model klasifikasi yang digunakan dalam penelitian ini, diterapkan sejumlah metrik evaluasi standar yang berbasis pada komponen dari *confusion matrix*, yaitu: *True Positive* (TP), *True Negative* (TN), *False Positive* (FP), dan *False Negative* (FN). Metrik-metrik ini memberikan gambaran menyeluruh mengenai seberapa baik model dalam melakukan prediksi terhadap data uji. Selain akurasi, sensitivitas, dan spesifisitas, penelitian ini juga menggunakan AUC (*Area Under the ROC Curve*) untuk mengukur performa model secara keseluruhan. AUC dipilih karena mampu mengevaluasi *trade-off* antara True Positive Rate dan False Positive Rate pada seluruh threshold.

3.4.2.1 Akurasi

Akurasi digunakan untuk mengukur proporsi total prediksi yang benar terhadap keseluruhan prediksi yang dilakukan. Metrik ini menunjukkan seberapa sering model membuat prediksi yang tepat, baik dalam mengklasifikasikan kelas positif maupun negatif.

Rumus:

Muhammad Wildan Alfarizy, 2025

DETEKSI ARITMIA JANTUNG BERBASIS INTEGRASI SINYAL ECG DAN PPG DENGAN PENDEKATAN MACHINE LEARNING DAN DEEP LEARNING DALAM SISTEM IOT

Universitas Pendidikan Indonesia | repository.upi.edu | perpustakaan.upi.edu

$$Akurasi = \frac{TP+TN}{TP+TN+FP+FN} \quad (3.1)$$

3.4.2.2 Sensitivitas

Sensitivitas, atau yang juga dikenal sebagai *recall*, mengukur kemampuan model dalam mendeteksi atau mengidentifikasi semua kasus positif yang sebenarnya. Metrik ini sangat penting dalam konteks medis atau sistem deteksi dini, di mana kesalahan dalam melewatkan kasus positif bisa berakibat serius.

Rumus:

$$Sensitivitas = \frac{TP}{TP+FN} \quad (3.2)$$

3.4.2.3 Spesifisitas

Spesifisitas menilai kemampuan model dalam mengidentifikasi kasus negatif dengan benar. Metrik ini penting untuk mengetahui seberapa baik model dalam menghindari kesalahan mendeteksi kasus negatif sebagai positif.

Rumus:

$$Spesifisitas = \frac{TN}{TP+TN+FP+FN} \quad (3.3)$$

3.4.2.4 Precision

Precision digunakan untuk mengukur seberapa akurat prediksi positif yang dibuat oleh model. Dalam situasi di mana kesalahan positif palsu (false positive) harus diminimalkan, seperti dalam diagnosis penyakit serius, precision menjadi indikator.

Rumus:

$$Precision = \frac{TP}{TP+FP} \quad (3.4)$$

3.4.2.5 F1-Score

F1-Score merupakan rata-rata harmonik dari precision dan sensitivitas. Metrik ini memberikan keseimbangan antara keduanya, terutama berguna ketika terdapat ketidakseimbangan kelas (class imbalance) dalam dataset. F1-Score memberikan gambaran yang lebih adil terhadap performa model dibandingkan hanya mengandalkan precision atau sensitivitas secara terpisah.

Muhammad Wildan Alfarizy, 2025

DETEKSI ARITMIA JANTUNG BERBASIS INTEGRASI SINYAL ECG DAN PPG DENGAN PENDEKATAN MACHINE LEARNING DAN DEEP LEARNING DALAM SISTEM IOT

Universitas Pendidikan Indonesia | repository.upi.edu | perpustakaan.upi.edu

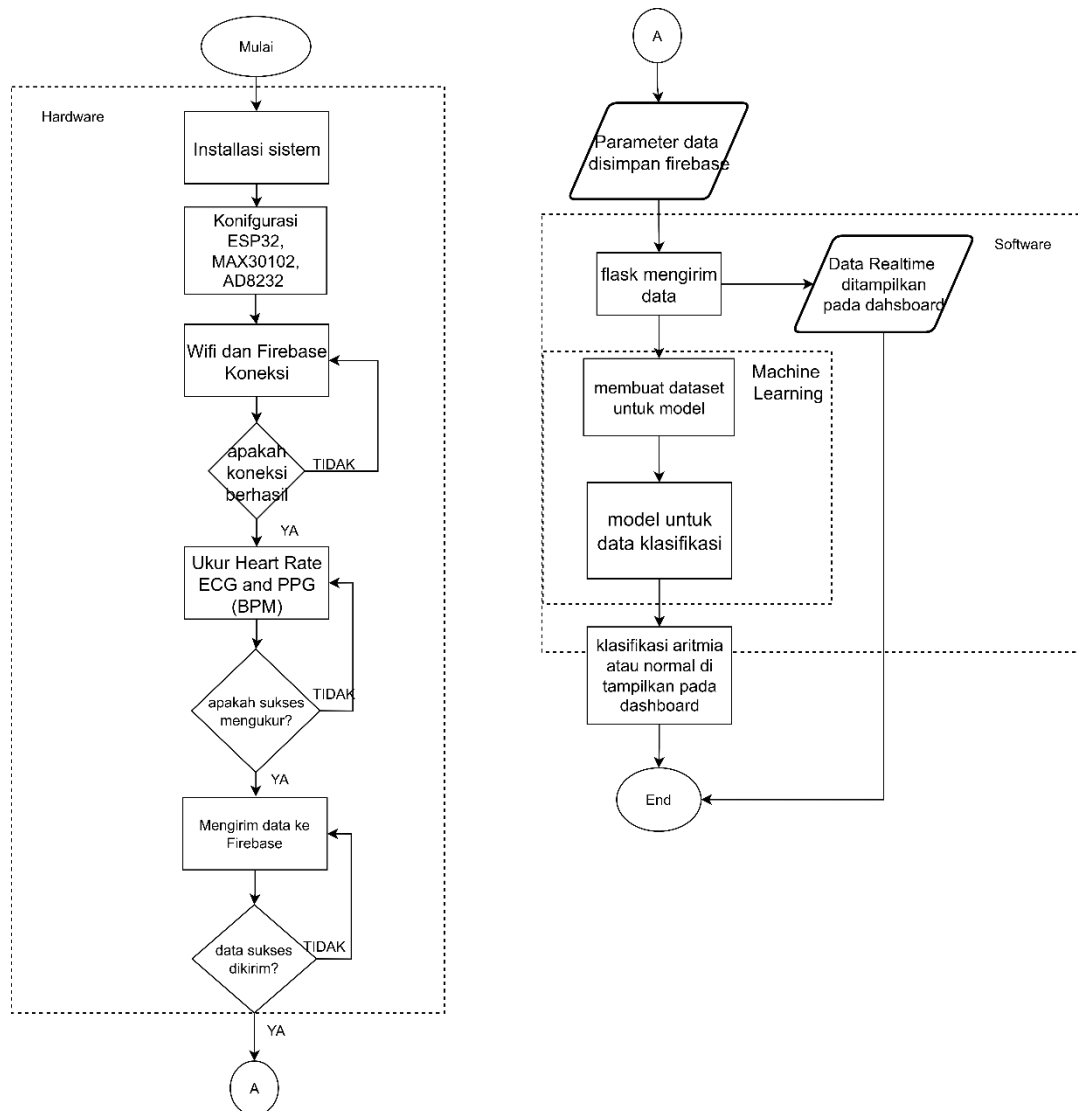
Rumus:

$$F1 - Score = \frac{2 \times Precision \times Sensitivitas}{Precision + Sensitivitas} \quad (3.5)$$

3.4 Data Hasil Pengambilan Langsung

Pengujian lapangan dilakukan pada lima partisipan sehat dalam kondisi relaksasi (duduk tenang tanpa aktivitas fisik) untuk meminimalkan *motion artifacts* dan memastikan sinyal merepresentasikan aktivitas jantung normal. Seluruh partisipan tidak memiliki riwayat kardiovaskular maupun indikasi aritmia, sehingga layak dijadikan sampel uji sinyal normal. Perekaman dilakukan selama 10 menit menggunakan sensor ECG dan PPG secara bersamaan agar data biometrik terintegrasi. Data mentah kemudian diproses melalui tahapan filtering (mengurangi noise), segmentasi (memisahkan bagian relevan), dan normalisasi (menyeragamkan skala). Selanjutnya, dilakukan ekstraksi fitur dengan metode yang sama seperti dataset pelatihan untuk menjaga konsistensi dengan model. Fitur yang dihasilkan dimasukkan ke model klasifikasi terbaik hasil pelatihan sebelumnya untuk menentukan kategori normal atau aritmia serta menganalisis potensi false positive pada sistem..

3.5 Integrasi Keseluruhan Sistem



Gambar 3. 17 Flowchart keseluruhan system deteksi aritmia

Diagram alur penelitian pada Gambar 3.1 menggambarkan tahapan operasional sistem yang dirancang, yang terdiri atas dua komponen utama, yakni *hardware* dan *software*. Proses dimulai dari tahap inialisasi sistem hingga hasil klasifikasi kondisi jantung dapat ditampilkan secara langsung melalui *dashboard* aplikasi.

Pada sisi *hardware*, inialisasi melibatkan konfigurasi awal mikrokontroler ESP32 beserta dua sensor utama: MAX30102 untuk sinyal *Photoplethysmography*

Muhammad Wildan Alfarizy, 2025

DETEKSI ARITMIA JANTUNG BERBASIS INTEGRASI SINYAL ECG DAN PPG DENGAN PENDEKATAN MACHINE LEARNING DAN DEEP LEARNING DALAM SISTEM IOT

Universitas Pendidikan Indonesia | repository.upi.edu | perpustakaan.upi.edu

(PPG) dan AD8232 untuk sinyal *Electrocardiogram* (ECG). Setelah konfigurasi selesai, ESP32 secara otomatis mencoba menghubungkan diri ke jaringan WiFi dan membangun koneksi dengan *Firebase Realtime Database* sebagai media penyimpanan berbasis cloud. Jika koneksi berhasil, sistem mulai melakukan akuisisi data berupa detak jantung dari kedua sensor. Nilai BPM (*beats per minute*) dihitung dari data tersebut, dan apabila hasilnya valid, maka parameter ini dikirim ke Firebase. Proses ini menjadi tahap akhir dari sisi hardware sekaligus titik transisi menuju pengolahan data pada *software*.

Pada sisi *software*, data yang telah tersimpan di Firebase diakses oleh backend aplikasi berbasis Flask. Data ini diproses dengan model *machine learning* dan *deep learning* yang sebelumnya dilatih menggunakan dataset relevan. Model bertugas mengklasifikasikan kondisi jantung menjadi dua kategori utama, yaitu Normal dan Aritmia, berdasarkan parameter utama seperti BPM dan variabilitas detak jantung (HRV). Hasil klasifikasi ini kemudian ditampilkan secara *real-time* melalui *dashboard web* interaktif. Selain klasifikasi, dashboard juga menyajikan data sinyal mentah dan parameter pendukung lain, sehingga pengguna dapat memperoleh gambaran menyeluruh mengenai kondisi jantung.

Secara keseluruhan, alur ini memperlihatkan sinergi antara akuisisi data fisiologis melalui hardware, pemrosesan berbasis cloud oleh software, serta visualisasi hasil klasifikasi melalui antarmuka *web*. Integrasi tersebut menghasilkan sistem deteksi aritmia yang *real-time*, akurat, dan fleksibel, serta dapat diakses dari berbagai lokasi dengan tingkat keandalan yang tinggi.