

Pengembangan Aplikasi Android Untuk Pemantauan Mandiri Denyut Jantung Berbasis Sinyal Elektrokardiogram Buatan

Muhammad Adil Chaer

Program Studi Teknik Biomedis
Fakultas Teknik Elektro, Universitas
Telkom
Bandung, Indonesia
adilchaer010@gmail.com

Tito Waluyo Purboyo

Program Studi Teknik Biomedis
Fakultas Teknik Elektro, Universitas
Telkom
Bandung, Indonesia
titowaluyo@telkomuniversity.ac.id

M. Darfyma Putra

Program Studi Teknik Biomedis
Fakultas Teknik Elektro, Universitas
Telkom
Bandung, Indonesia
mdarfymap@telkomuniversity.ac.id

Abstrak — Pemantauan kesehatan jantung secara waktu nyata (*real-time*) penting untuk mendeteksi dini potensi gangguan irama jantung. Perangkat elektrokardiogram (ECG) konvensional umumnya berukuran besar, mahal, dan memerlukan tenaga ahli untuk pengoperasiannya, sehingga sulit diakses terutama di wilayah dengan keterbatasan fasilitas medis. Penelitian ini bertujuan mengembangkan aplikasi Android untuk memantau sinyal ECG buatan yang dihasilkan oleh mikrokontroler ESP32, menampilkan grafik sinyal secara langsung, menghitung jumlah detak jantung per menit (*beats per minute / BPM*), mengklasifikasikan kondisi jantung, dan menyimpan hasil pemeriksaan ke basis data daring. Metode yang digunakan mencakup pembuatan sinyal ECG sintetik pada ESP32, pengiriman data melalui koneksi nirkabel, pemrosesan sinyal di aplikasi untuk perhitungan BPM dan deteksi kondisi bradikardia/takikardia, serta penyimpanan hasil ke *cloud database*. Hasil pengujian menunjukkan aplikasi mampu menampilkan sinyal ECG dengan laju rata-rata 125 data/detik, latensi rata-rata 82 ms, dan akurasi perhitungan BPM 100%. Sistem juga berhasil menyimpan dan menampilkan kembali riwayat pemeriksaan secara akurat. Aplikasi ini dapat digunakan sebagai sarana simulasi dan pembelajaran pemantauan ECG secara waktu nyata tanpa memerlukan perangkat medis mahal, mendukung konsep akses kesehatan yang lebih luas.

Kata kunci— elektrokardiogram, ESP32, Android, BPM, pemantauan real-time

I. PENDAHULUAN

Penyakit kardiovaskular merupakan penyebab utama kematian di dunia, dengan lebih dari 17,9 juta kematian setiap tahun atau sekitar 31% dari total kematian global[1]. Deteksi dini dan pemantauan kondisi jantung secara berkala dapat mengurangi risiko komplikasi serius serta meningkatkan kualitas hidup pasien. Salah satu indikator penting dalam deteksi dini adalah kelainan ritme jantung, seperti bradikardia (denyut <60 BPM) dan takikardia (denyut >100 BPM)[2].

Pemeriksaan elektrokardiogram (ECG) secara konvensional umumnya dilakukan di fasilitas medis menggunakan perangkat berukuran besar, mahal, dan memerlukan tenaga ahli untuk interpretasi hasil[3]. Kondisi ini menjadi hambatan bagi masyarakat di daerah terpencil atau individu dengan keterbatasan mobilitas. Perkembangan teknologi Internet of Medical Things (IoMT) dan perangkat mobile membuka peluang pemantauan mandiri (*self-*

monitoring) yang lebih terjangkau, portabel, dan mudah digunakan[4].

Pendekatan *self-monitoring* telah terbukti meningkatkan keterlibatan pasien dalam manajemen kesehatan pribadi, mempercepat respons terhadap gejala awal, dan mendukung pengambilan keputusan klinis yang lebih baik [5]. Penelitian ini berfokus pada pengembangan aplikasi Android untuk pemantauan denyut jantung secara real-time menggunakan sinyal ECG buatan yang dihasilkan oleh mikrokontroler ESP32. Sinyal ini dikirim melalui koneksi nirkabel ke aplikasi, kemudian diolah untuk menghitung BPM, mendeteksi kondisi bradikardia atau takikardia, dan menyimpan hasil pemantauan ke *cloud database* untuk akses riwayat pemeriksaan.

Ruang lingkup penelitian dibatasi pada penggunaan sinyal ECG sintetis untuk pengujian sistem, tanpa melibatkan pasien atau perangkat medis klinis. Aplikasi ini diharapkan menjadi sarana simulasi dan pembelajaran yang dapat diintegrasikan dengan data ECG riil pada pengembangan selanjutnya.

II. KAJIAN TEORI

A. Sistem Kardiovaskular

Sistem kardiovaskular adalah sistem organ yang bertugas mengedarkan darah, oksigen, dan nutrisi ke seluruh tubuh, serta membuang sisa metabolisme[5]. Jantung terdiri dari empat ruang, yaitu atrium kanan-kiri dan ventrikel kanan-kiri, yang memompa darah melalui sirkulasi sistemik dan pulmonal .

Aktivitas jantung memiliki dua fase, yaitu sistol (kontraksi) dan diastol (relaksasi), yang memengaruhi tekanan darah. Tekanan darah dinyatakan dalam mmHg dengan format sistolik/diastolik. Nilai di bawah 100/60 mmHg dikategorikan hipotensi, sementara nilai di atas 140/90 mmHg dikategorikan hipertensi[6].

B. Elektrokardiogram (ECG)

Elektrokardiogram (ECG) merekam aktivitas listrik jantung menggunakan elektroda di permukaan kulit[7]. Hasilnya berupa gelombang PQRST, di mana: gelombang P merepresentasikan depolarisasi atrium, kompleks QRS menandakan depolarisasi ventrikel, dan gelombang T menunjukkan repolarisasi ventrikel[8]. Parameter lain yang digunakan meliputi interval PR dan segmen ST.

Interpretasi ECG diawali dengan identifikasi irama sinus, perhitungan denyut jantung, penentuan aksis listrik, serta evaluasi kelainan seperti aritmia dan iskemia miokard[9]. Pemeriksaan ECG konvensional memerlukan perangkat klinis, prosedur pemasangan elektroda sesuai standar, dan interpretasi oleh tenaga medis terlatih [10].

C. Beats Per Minute (BPM)

BPM adalah jumlah kontraksi jantung per menit yang menjadi indikator penting fungsi kardiovaskular. Nilai BPM normal orang dewasa dalam kondisi istirahat berkisar 60–100 BPM[11]. Metode pengukuran meliputi palpasi nadi, auskultasi, dan pemantauan elektronik seperti ECG maupun Photoplethysmography (PPG). PPG adalah metode optik non-invasif yang mengukur perubahan volume darah menggunakan cahaya LED dan sensor fotodetektor. Meskipun lebih praktis untuk perangkat *wearable*, PPG memiliki akurasi lebih rendah dibanding ECG dalam mendeteksi gangguan irama kompleks[12].

D. Abnormalitas Denyut Jantung

Bradikardia adalah kondisi di mana denyut jantung seseorang berada di bawah 60 BPM pada orang dewasa dalam kondisi istirahat[13]. Kondisi ini dapat bersifat fisiologis maupun patologis. Bradikardia fisiologis sering ditemukan pada atlet atau individu yang berlatih fisik secara rutin, di mana penurunan denyut jantung disebabkan oleh peningkatan efisiensi pompa jantung. Namun, pada kasus patologis, bradikardia dapat diakibatkan oleh gangguan sistem konduksi jantung seperti *sick sinus syndrome*, blok atrioventrikular, efek obat-obatan (misalnya beta blocker), gangguan elektrolit, atau kondisi medis lain seperti hipotiroidisme. Jika tidak segera ditangani, kondisi ini berpotensi menyebabkan gagal jantung atau henti jantung mendadak.

Takikardia adalah kondisi denyut jantung lebih dari 100 BPM pada orang dewasa. Takikardia dapat diklasifikasikan menjadi *supraventricular tachycardia* (SVT) dan *ventricular tachycardia* (VT) berdasarkan lokasi asal impuls listrik. Penyebab takikardia meliputi aktivitas fisik berat, demam, dehidrasi, stres emosional, anemia, hipertiroidisme, gangguan elektrolit, atau kelainan struktural jantung. Pada beberapa kasus, takikardia dapat bersifat paroksismal (muncul tiba-tiba) dan menghilang sendiri, namun pada kasus yang persisten dapat memicu komplikasi serius seperti fibrilasi ventrikel dan kematian mendadak.

Kedua kondisi ini — baik bradikardia maupun takikardia — termasuk dalam kategori gangguan irama jantung (*arrhythmia*) yang penting untuk dideteksi secara dini. Pemantauan denyut jantung secara real-time, seperti yang diimplementasikan pada penelitian ini, dapat membantu mengidentifikasi perubahan irama jantung lebih cepat sehingga penanganan medis dapat dilakukan segera, mengurangi risiko komplikasi yang mengancam jiwa.

E. Teknologi Pengembangan Aplikasi Mobile

Pengembangan aplikasi mobile untuk pemantauan kesehatan memerlukan integrasi berbagai teknologi yang saling mendukung dari sisi *frontend*, *backend*, hingga perangkat keras. REST API (*Representational State Transfer Application Programming Interface*) digunakan sebagai

metode komunikasi antara server dan klien, yang memungkinkan pertukaran data secara terstruktur dalam format JSON melalui protokol HTTP[14]. Keunggulan REST API antara lain kemudahan integrasi lintas platform, skalabilitas tinggi, dan kompatibilitas luas dengan berbagai bahasa pemrograman.

Proses pengembangan aplikasi Android dilakukan menggunakan Android Studio, *Integrated Development Environment* (IDE) resmi dari Google yang mendukung bahasa Java dan Kotlin[15]. Android Studio menyediakan fitur seperti *layout editor*, *debugger*, dan *emulator* yang memudahkan proses desain antarmuka, pengujian, dan optimisasi kinerja aplikasi. Bahasa pemrograman Java dipilih dalam penelitian ini karena stabilitasnya, dokumentasi yang luas, dan kompatibilitasnya dengan berbagai pustaka pihak ketiga[16].

Untuk penyimpanan data, sistem ini menggunakan PostgreSQL, sebuah sistem manajemen basis data relasional (*Relational Database Management System*, RDBMS) yang mendukung transaksi ACID (*Atomicity*, *Consistency*, *Isolation*, *Durability*) serta memiliki kemampuan menangani query kompleks dan pengelolaan data berskala besar[17]. Komunikasi data real-time antara perangkat dan server difasilitasi melalui protokol WebSocket, yang memungkinkan koneksi *full-duplex* dengan latensi rendah sehingga cocok untuk transmisi sinyal ECG secara kontinu[18].

Pada sisi perangkat keras, mikrokontroler ESP32 digunakan sebagai node pengirim data ECG[19]. ESP32 memiliki prosesor dual-core, koneksi Wi-Fi/Bluetooth, ADC internal, dan kemampuan pengolahan sinyal, sehingga ideal untuk aplikasi IoT medis skala kecil. Backend aplikasi dibangun menggunakan Node.js, sebuah *runtime environment* berbasis JavaScript yang unggul dalam menangani proses I/O secara asinkron[20]. Framework Express.js digunakan untuk mempermudah pembuatan *endpoint API*, mengatur *routing*, serta menangani permintaan HTTP dengan efisien[21].

Sistem ini di-deploy menggunakan Railway, sebuah platform *cloud deployment* yang mendukung otomatisasi *build*, manajemen *environment variables*, dan integrasi dengan GitHub[22],[23]. Reposisi kode dikelola melalui GitHub untuk memfasilitasi kolaborasi, *version control*, dan dokumentasi pengembangan perangkat lunak[24]. Pemrograman ESP32 dilakukan dengan Arduino IDE, yang menyediakan pustaka dan antarmuka pemrograman sederhana untuk mempermudah pengembangan firmware berbasis C/C++[25].

F. Penelitian Terdahulu

Beberapa penelitian terkait pengembangan sistem pemantauan kesehatan berbasis IoT dan aplikasi mobile telah dilakukan sebelumnya. Hidayat et al.[26] mengembangkan sistem pemantauan detak jantung berbasis IoT menggunakan sensor pulse rate dan komunikasi nirkabel untuk menampilkan data pada aplikasi Android. Sistem ini mampu memantau kondisi pasien secara real-time, namun belum dilengkapi fitur penyimpanan riwayat.

Putra dan Santoso[27] mengimplementasikan perangkat ECG portabel dengan koneksi Bluetooth Low Energy (BLE) yang terhubung ke smartphone. Hasil pengujian

menunjukkan akurasi BPM lebih dari 95%, tetapi jangkauan transmisi terbatas pada radius ± 10 meter.

Penelitian oleh Maulana et al[18].membandingkan performa protokol WebSocket dan HTTP polling dalam pengiriman data medis. WebSocket terbukti memiliki latensi 40% lebih rendah dan lebih efisien dalam penggunaan bandwidth, meskipun membutuhkan koneksi internet yang stabil.

Wahyudi et al.[28] memanfaatkan mikrokontroler ESP32 untuk menghasilkan sinyal ECG buatan sebagai sarana simulasi. Sistem ini efektif untuk pengujian algoritma pemrosesan sinyal di perangkat mobile, namun belum menggunakan data ECG riil dari pasien.

Rizki et al.[29] merancang sistem pemantauan kesehatan berbasis REST API yang mengintegrasikan sensor medis dengan aplikasi Android dan penyimpanan data di *cloud*. Sistem ini memungkinkan pemantauan jarak jauh dan akses riwayat hasil pemeriksaan, tetapi memiliki ketergantungan pada koneksi internet yang stabil dan berkelanjutan.

G. Parameter Uji

1) Batas Delay Untuk Real-Time

Sistem dikatakan bekerja secara *real-time* jika mampu memproses, mengirim, dan menampilkan data dengan *delay* rendah sehingga informasi tetap relevan untuk pengambilan keputusan langsung. Dalam pemantauan sinyal ECG, *delay* yang besar dapat membuat tampilan tidak sesuai kondisi jantung terkini. Liu et al. menyatakan bahwa batas *delay* real-time untuk pemantauan fisiologis adalah < 250 ms[30].

Selain *delay*, kualitas *real-time* dipengaruhi oleh *jitter* (ideal < 30 ms untuk kestabilan sinyal), *packet delivery rate* (ideal $> 95\%$ untuk mencegah hilangnya data penting), *throughput* (mendukung frekuensi sampling minimal 125 sampel/detik sesuai IEC 60601-2-47), dan *packet loss* (ideal $< 5\%$ untuk menjaga integritas sinyal).

Pada penelitian ini, evaluasi kinerja *real-time* mempertimbangkan seluruh parameter tersebut untuk memberikan gambaran menyeluruh dan memastikan klaim *real-time* dapat dibuktikan secara teknis melalui pengujian.

2) Durasi Pengukuran Denyut Jantung

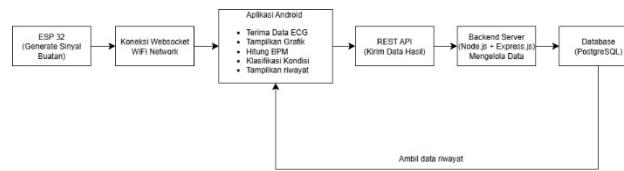
Durasi waktu pengukuran sinyal elektrokardiogram (ECG) memiliki peran krusial dalam memastikan keakuratan data denyut jantung (Beats Per Minute/BPM), terutama pada sistem pemantauan real-time. Secara umum, pengukuran denyut jantung dapat dilakukan dalam berbagai interval waktu, mulai dari 10 detik, 30 detik, hingga 1 menit atau lebih, tergantung pada tujuan dan konteks penggunaannya. Dalam praktik klinis konvensional, pengukuran manual terhadap denyut nadi sering dilakukan selama 15 atau 30 detik, kemudian dikalikan untuk mendapatkan estimasi BPM per menit. Namun, dalam pengukuran berbasis sensor dan sistem tertanam (embedded system), pemilihan durasi pengukuran yang tepat harus mempertimbangkan keseimbangan antara presisi data, efisiensi pemrosesan, dan kebutuhan aplikasi secara real-time. Beberapa penelitian telah dilakukan untuk mengevaluasi validitas pengukuran BPM dengan durasi yang lebih pendek dari standar 60 detik. Salah satu studi yang relevan adalah penelitian oleh Salahuddin et al. (2021) yang dipublikasikan dalam jurnal Sensors. Penelitian ini membandingkan keakuratan estimasi

denyut jantung menggunakan rekaman ECG berdurasi 30 detik (short- 27 term) dengan rekaman berdurasi 5 menit (long-term), baik pada kondisi istirahat maupun setelah aktivitas fisik ringan. Hasil pengujian menunjukkan bahwa estimasi BPM yang diperoleh dari rekaman ECG 30 detik memberikan hasil yang konsisten dan tidak memiliki perbedaan signifikan dibandingkan dengan perekaman berdurasi 5 menit, khususnya saat individu berada dalam kondisi baseline. Penelitian tersebut juga menekankan bahwa durasi 30 detik cukup untuk merepresentasikan rata-rata aktivitas jantung dengan baik dalam sistem berbasis ECG maupun PPG (Photoplethysmography). Pemanfaatan durasi 30 detik dalam sistem pemantauan ECG memberikan beberapa keuntungan, antara lain: peningkatan efisiensi komputasi, reduksi konsumsi daya pada perangkat portable, serta peningkatan respons waktu sistem terhadap perubahan kondisi fisiologis. Dalam sistem berbasis mikrokontroler seperti ESP32, pengambilan data selama 30 detik memungkinkan integrasi antara kecepatan akuisisi sinyal, pengolahan lokal, dan pengiriman data secara nirkabel tanpa menyebabkan beban pemrosesan yang berlebihan. Hal ini sangat penting terutama dalam aplikasi monitoring jantung yang bersifat terus-menerus (continuous monitoring), di mana kompromi antara ketepatan dan efisiensi harus diperhitungkan. Dengan mempertimbangkan aspek teknis dan hasil penelitian empiris tersebut, penggunaan durasi 30 detik dalam pengukuran BPM dapat dikategorikan sebagai pendekatan yang valid dan efisien, khususnya untuk sistem pemantauan real-time berbasis sinyal ECG[31]. Oleh karena itu, dalam penelitian ini, durasi 30 detik digunakan sebagai standar waktu untuk setiap sesi pengujian BPM, dengan landasan ilmiah yang mendukung bahwa interval tersebut cukup representatif untuk mengevaluasi performa transmisi data serta estimasi denyut jantung dari perangkat ke aplikasi pengguna.

III. METODE

A. Tahapan Penelitian

Penelitian ini menggunakan metode *research and development* yang terdiri dari empat tahap utama: studi literatur, perancangan sistem, pengembangan, dan pengujian. Studi literatur dilakukan untuk mengumpulkan informasi tentang prinsip kerja sinyal elektrokardiogram (ECG), parameter *beats per minute* (BPM), klasifikasi abnormalitas jantung, serta teknologi pendukung seperti *Internet of Medical Things* (IoMT), protokol WebSocket, REST API, dan basis data *cloud*. Tahapan ini dilanjutkan dengan perancangan sistem yang mencakup rancangan perangkat keras dan perangkat lunak, integrasi koneksi *real-time*, serta mekanisme penyimpanan data.



GAMBAR 1
Diagram blok sistem

Diagram ini menunjukkan urutan tahapan penelitian, mulai dari studi literatur, perancangan sistem, pengembangan, hingga pengujian dan analisis hasil. Alur ini

memastikan proses pengembangan berlangsung sistematis dan terstruktur.

B. Perancangan Perangkat Keras

Perangkat keras utama sistem adalah modul ESP32 WROOM-32 yang memiliki prosesor dual-core Xtensa LX6, koneksi Wi-Fi 2,4 GHz, dan dukungan protokol TCP/IP yang sesuai untuk WebSocket. Modul ini diprogram untuk menghasilkan sinyal ECG buatan dengan frekuensi sampling ± 125 Hz menggunakan algoritma pembangkitan gelombang berbasis data simulasi. Data dikodekan dalam format JSON untuk mempermudah parsing di sisi aplikasi Android.

Alasan pemilihan ESP32 adalah ketersediaan koneksi nirkabel bawaan, kapasitas pemrosesan yang memadai untuk simulasi sinyal, serta konsumsi daya yang rendah. Perangkat didukung oleh catu daya portabel 5V yang memungkinkan mobilitas tinggi saat pengujian.

C. Perancangan Perangkat Lunak

1) Backend

Backend dikembangkan menggunakan Node.js dengan framework Express.js. Fungsi utamanya adalah menerima data hasil pemrosesan dari aplikasi Android melalui REST API, memvalidasi format data, dan menyimpannya ke basis data PostgreSQL yang di-hosting di Railway. Endpoint yang disediakan meliputi autentikasi pengguna, pengiriman data hasil, permintaan riwayat pemeriksaan, dan pembaruan profil.

2) Frontend (Aplikasi Android)

Aplikasi Android dibangun menggunakan Android Studio dan bahasa pemrograman Java. Modul utama aplikasi meliputi koneksi WebSocket untuk menerima *streaming* data ECG, algoritma deteksi puncak R untuk menghitung BPM, serta fungsi klasifikasi kondisi jantung menjadi bradikardia, normal, atau takikardia.

D. Desain Backend

1) API Architecture and Endpoint Design

Desain API menggunakan RESTful principles dengan struktur endpoint yang konsisten dan intuitif. Setiap endpoint dirancang untuk memenuhi specific use case dalam aplikasi ECG monitoring, mulai dari user management hingga real-time ECG data processing.

TABEL 1
Arsitektur API dan Endpoint Design

Category	HTTP Method	Endpoint	Description
System	GET	/	Root endpoint info
System	GET	/health	Health check
System	GET	/api/test	API functionality test
Authentication	POST	/api/auth/register	User registration
Authentication	POST	/api/auth/login	User login
User Management	GET	/api/users/all	Get all users

User Management	GET	/api/profile/:userId	Get user profile
User Management	PUT	/api/profile/update	Update user profile
ECG Data		/api/ecg/save	Save ECG result
ECG Data	GET	/api/ecg/history/:userId	Get ECG history
ECG Data	DELETE	/api/ecg/history/:userId	Delete user history
ECG Data	DELETE	/api/ecg/history/:userId/:id	Delete specific record

Desain endpoint pada tabel di atas memastikan bahwa setiap fungsi dalam sistem, mulai dari autentikasi, manajemen pengguna, hingga pengelolaan data ECG, dapat diakses secara terstruktur dan efisien. Penggunaan metode HTTP yang tepat pada setiap endpoint tidak hanya mempermudah integrasi antar komponen sistem, tetapi juga meningkatkan keamanan dan keandalan proses pertukaran data.

2) Database Schema Design

Skema basis data dirancang dengan penekanan pada integritas data, kinerja *query*, dan kemampuan skalabilitas. Struktur skema menggunakan desain dua tabel yang disesuaikan secara optimal untuk kasus penggunaan pemantauan sinyal ECG, serta menerapkan normalisasi yang tepat guna mencegah terjadinya duplikasi data.

TABEL 2
Tabel Users di Database

Field Name	Data Type	Constraints	Keterangan
<i>id</i>	SERIAL	PRIMARY KEY	Auto-incrementing user identifier
<i>username</i>	VARCHAR(255)	NOT NULL, UNIQUE	User login identifier
<i>password</i>	VARCHAR(255)	NOT NULL	Encrypted password storage
<i>age</i>	INTEGER	NOT NULL, CHECK (age BETWEEN 1 AND 150)	User age for medical context
<i>gender</i>	VARCHAR(10)	NOT NULL, CHECK (gender IN ('Male', 'Female'))	Gender for medical analysis
<i>created_at</i>	TIMESTAMP	NOT NULL, DEFAULT	Account creation

		<i>CURRENT_TIME STAMP</i>	<i>timestamp p</i>
<i>updated_at</i>	<i>TIMESTAMP P</i>	<i>NOT NULL, DEFAULT CURRENT_TIME STAMP</i>	<i>Last profile update timestamp p</i>

Struktur tabel *Users* pada Tabel 3.9 memastikan pengelolaan data pengguna yang aman, konsisten, dan mudah diakses. Penerapan *constraint* seperti *PRIMARY KEY*, *UNIQUE*, dan *CHECK* berperan penting dalam menjaga integritas data, sementara tipe data yang tepat meminimalkan risiko kesalahan input. Desain ini juga mendukung efisiensi proses autentikasi dan pengolahan informasi medis yang relevan dalam sistem pemantauan ECG.

TABEL 3
Tabel *ecg_results* di database

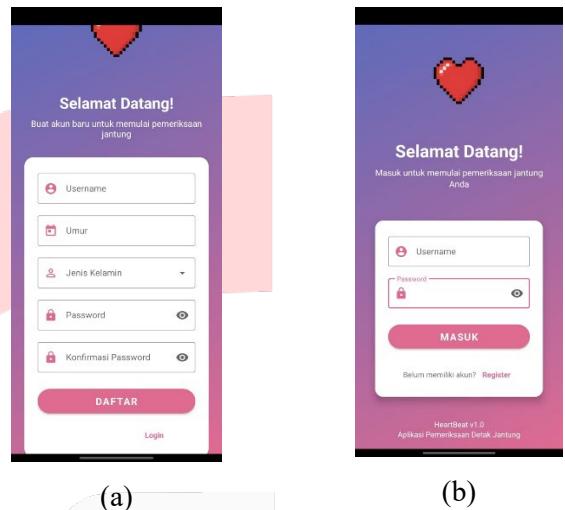
<i>Field Name</i>	<i>Data Type</i>	<i>Constraints</i>	<i>Keterangan an</i>
<i>id</i>	<i>SERIAL</i>	<i>PRIMARY KEY</i>	<i>Auto-incrementing result identifier</i>
<i>user_id</i>	<i>INTEGER</i>	<i>NOT NULL, FOREIGN KEY</i>	<i>Reference to users table</i>
<i>username</i>	<i>VARCHAR(255)</i>	<i>NOT NULL</i>	<i>Denormalized username for performance</i>
<i>tanggal</i>	<i>DATE</i>	<i>NOT NULL, DEFAULT CURRENT_DATE</i>	<i>Examination date</i>
<i>waktu</i>	<i>TIME</i>	<i>NOT NULL</i>	<i>Examination time</i>
<i>bpm</i>	<i>INTEGER</i>	<i>NOT NULL, CHECK (bpm BETWEEN 1 AND 300)</i>	<i>Beats per minute measurement</i>
<i>status</i>	<i>VARCHAR(20)</i>	<i>NOT NULL, CHECK (status IN ('Normal', 'Abnormal'))</i>	<i>Medical classification</i>
<i>kondisi</i>	<i>VARCHAR(50)</i>	<i>NOT NULL</i>	<i>Detailed medical condition</i>
<i>created_at</i>	<i>TIMESTAMP P</i>	<i>NOT NULL, DEFAULT CURRENT_TIME STAMP</i>	<i>Record creation timestamp</i>

Struktur tabel *ecg_results* dirancang untuk merekam hasil pemeriksaan ECG secara terorganisir dan terintegrasi dengan data pengguna melalui *foreign key*. Informasi yang disimpan meliputi tanggal, waktu, BPM, status, dan kondisi yang memungkinkan analisis riwayat kesehatan jantung secara akurat. Penambahan kolom *denormalized username*

berfungsi meningkatkan kinerja query, sedangkan penerapan *constraint* menjamin validitas dan konsistensi data yang tersimpan.

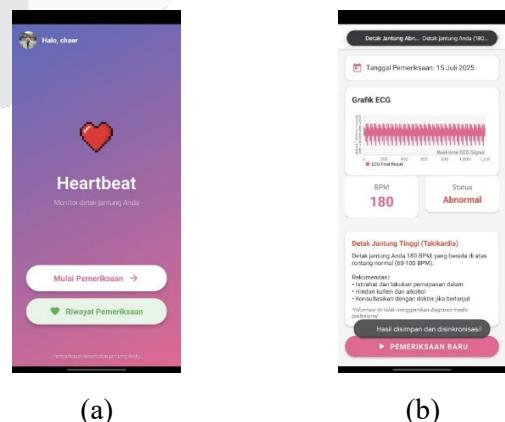
E. Desain Antarmuka Pengguna

Desain antarmuka dirancang agar intuitif dan mudah digunakan pasien. Ikon-ikon dipilih untuk memudahkan navigasi. Halaman utama (dashboard) menampilkan grafik sinyal ECG secara *real-time*, informasi BPM, dan status kondisi jantung. Menu lain mencakup registrasi, login, riwayat pemeriksaan, dan pengaturan profil.



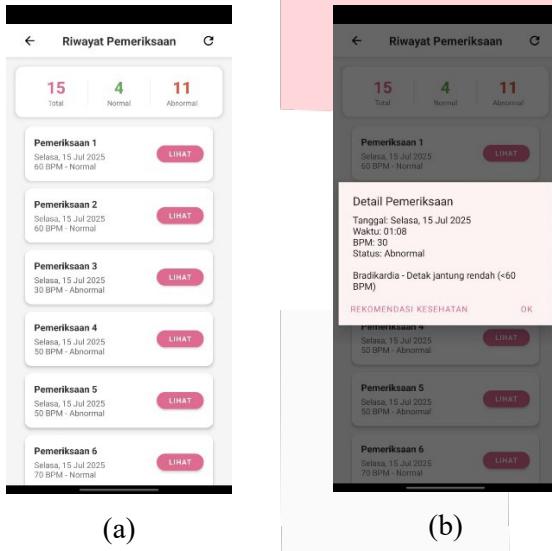
GAMBAR 2
(a) Gambar tampilan Register, (b) Gambar tampilan login

Gambar (a) menampilkan halaman registrasi dengan antarmuka sederhana dan ramah pengguna, berisi kolom username, umur, jenis kelamin, password, dan konfirmasi password. Teks sambutan “Selamat Datang!” memberi kesan hangat, sementara ikon pada setiap kolom memudahkan identifikasi. Tombol “DAFTAR” berwarna kontras berada di bawah, disertai link “Login” bagi pengguna lama. Gambar (b) menunjukkan halaman login dengan desain konsisten, menampilkan ikon hati pixelated, kolom username dan password, serta teks sambutan singkat. Tombol “MASUK” berwarna mencolok meningkatkan visibilitas, dengan link “Register” bagi pengguna baru.



GAMBAR 3
(a) Gambar tampilan dashboard (a), (b) Gambar tampilan halaman

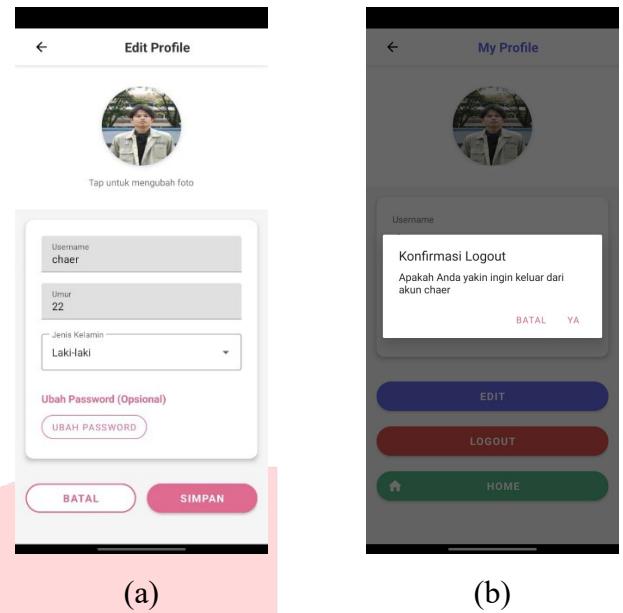
Gambar (a) menampilkan halaman utama aplikasi setelah login dengan desain sederhana, latar gradasi ungu-merah muda, dan ikon hati piksel di tengah. Sapaan personal muncul di kiri atas, disertai dua tombol utama berwarna kontras: "Mulai Pemeriksaan" untuk memantau sinyal jantung real-time, dan "Riwayat Pemeriksaan" untuk melihat hasil sebelumnya. Teks "Pemantauan kesehatan jantung Anda" di bagian bawah menegaskan tujuan aplikasi. Gambar (b) menunjukkan halaman hasil pemeriksaan ECG real-time. Notifikasi "Denyut Jantung Abnormal" tampil di atas, diikuti tanggal, grafik ECG, nilai BPM (contoh: 180), dan status "Abnormal" dengan warna mencolok. Diagnosa awal "Takikardia" disertai penjelasan dan rekomendasi kesehatan. Tersedia tombol konfirmasi penyimpanan hasil dan "Pemeriksaan Baru" untuk memulai sesi berikutnya. Desain dibuat agar informasi tersaji cepat, jelas, dan mudah dipahami, terutama saat kondisi kritis.



GAMBAR 4

(a) Gambar tampilan riwayat pemeriksaan, (b) Gambar tampilan data

Halaman Riwayat Pemeriksaan menampilkan ringkasan jumlah pemeriksaan, hasil normal, dan abnormal di bagian atas. Di bawahnya, daftar pemeriksaan berbentuk kartu berisi tanggal, BPM, dan status, masing-masing dilengkapi tombol "LIHAT" untuk detail lebih lanjut. Desain sederhana memudahkan navigasi dan penelusuran data. Gambar (b) memperlihatkan popup detail hasil pemeriksaan, berisi tanggal, waktu, BPM (contoh: 30), status (Abnormal), dan diagnosa awal (Bradikardia). Dua tombol tersedia: "REKOMENDASI KESEHATAN" untuk saran tindakan, dan "OK" untuk menutup. Tampilan ini memudahkan akses cepat ke informasi tanpa berpindah halaman.



(a) Gambar tampilan edit profil, (b) tampilan logout

Halaman *My Profile* menampilkan data diri pengguna berupa username, jenis kelamin, dan usia dalam satu kartu profil, dengan foto profil di atas dan judul "*My Profile*". Tiga tombol utama tersedia: *EDIT* untuk mengubah data, *LOGOUT* untuk keluar, dan *HOME* untuk kembali ke halaman utama. Desain kontras dan intuitif memudahkan akses fitur. Gambar (b) menampilkan popup konfirmasi logout setelah tombol "*LOGOUT*" ditekan. Pesan "Apakah Anda yakin ingin keluar dari akun (user)?" disertai dua opsi: *BATAL* untuk membatalkan dan *YA* untuk melanjutkan. Desain sederhana ini mencegah kesalahan *logout* dan menjaga keamanan akun.

F. Dasar Pengembangan Simulasi Sinyal ECG pada ESP32

Salah satu komponen utama sistem adalah simulasi sinyal ECG yang dihasilkan mikrokontroler ESP32, dengan data amplitudo gelombang dalam bentuk array dikirim secara kontinu ke aplikasi Android melalui WebSocket. Pendekatan ini dipilih karena keterbatasan akses perangkat ECG medis asli yang mahal dan memerlukan tenaga medis, sementara fokus penelitian adalah pada pengembangan sistem pemantauan berbasis aplikasi, bukan perangkat akuisisi sinyal.

ESP32 dipilih karena mendukung komunikasi Wi-Fi dan kompatibel dengan WebSocket, sehingga ideal untuk transmisi data *real-time*. Sinyal yang dikirim menyerupai gelombang ECG nyata (segmen P, QRS, T) dengan interpolasi dan *noise* tambahan agar lebih realistik. Pendekatan ini mengacu pada proyek ECGGenerator oleh Walid Amriou[32], namun diperluas dengan penggunaan WebSocket dan format JSON untuk memungkinkan deteksi BPM, klasifikasi status denyut jantung, serta penghitungan statistik *delay* komunikasi.

Dengan demikian, penggunaan sinyal buatan dari ESP32 menjadi solusi efektif untuk menguji sistem monitoring jantung tanpa bergantung pada perangkat atau data medis asli.

IV. HASIL DAN PEMBAHASAN

Bagian ini menyajikan hasil pengujian menyeluruh terhadap sistem monitoring ECG real-time yang telah dikembangkan. Pengujian mencakup evaluasi performa perangkat keras ESP32, verifikasi akurasi algoritma deteksi sinyal jantung, analisis komunikasi nirkabel, serta pengujian antarmuka pengguna pada aplikasi Android. Sistem ini telah melewati tahapan pengujian secara sistematis untuk memastikan fungsionalitas, keandalan, dan performa sesuai dengan kebutuhan dan spesifikasi yang telah dirancang sebelumnya.

A. Pengujian performa Real-time

Pengujian ini bertujuan mengevaluasi kemampuan sistem mentransmisikan data ECG secara waktu nyata dari ESP32 ke aplikasi Android, mencakup keterlambatan jaringan (*delay*), variasi waktu antar paket (*jitter*), tingkat keberhasilan pengiriman paket (*packet delivery rate*), *throughput*, dan kestabilan sistem pada berbagai nilai BPM (30–190).

Proses pengukuran *delay* dilakukan dengan mekanisme sinkronisasi waktu antara fungsi `millis()` pada ESP32 dan `System.currentTimeMillis()` pada Android. Nilai *offset* dari paket pertama digunakan untuk menghitung *delay* paket berikutnya menggunakan rumus:

$$\text{Delay} = \text{androidReceivetime} - \text{espseendtime}$$

Setiap paket ECG dikirim bersama *timestamp* dan *packet ID* untuk pelacakan, kemudian dianalisis untuk mendapatkan jumlah paket per detik dan *max delay* selama uji 30 detik. Pengujian mencakup rentang BPM 30–190, dengan interval 20 BPM, dan target laju 125 paket/detik (± 3750 paket per sesi). Parameter yang diamati adalah *packets/second* dan *max delay* untuk menilai konsistensi dan keandalan transmisi.

Hasil pengujian menunjukkan bahwa sistem secara konsisten mampu mempertahankan laju transmisi data yang mendekati target 125 paket per detik pada seluruh variasi BPM yang diuji. Nilai *packets/second* tercatat berada pada kisaran antara 122,70 hingga 122,90. Selisih dari nilai teoretis menunjukkan adanya sedikit penurunan laju pengiriman, yang kemungkinan disebabkan oleh latensi jaringan atau kondisi pemrosesan di sisi perangkat Android. Namun, nilai tersebut menunjukkan deviasi yang relatif kecil, sehingga transmisi dapat dikategorikan stabil secara keseluruhan.

TABEL 4
Hasil Pengujian Delay

No	BPM	Max Delay (ms)	Packets/Second
1	30	10	122,77
2	50	8	122,90
3	70	39	122,87
4	90	16	122,80

5	110	11	122,83
6	130	22	122,77
7	150	31	122,87
8	170	78	122,70
9	190	8	122,90

Nilai *maximum delay* yang tercatat pada setiap skenario menunjukkan variasi antara 8 hingga 78 milidetik. Delay tertinggi sebesar 78 ms tercatat pada skenario BPM 170, sedangkan nilai terendah sebesar 8 ms muncul pada beberapa skenario seperti BPM 50 dan 190. Tidak ditemukan pola peningkatan delay maksimum yang linier terhadap peningkatan BPM, sehingga tidak terdapat indikasi bahwa laju denyut jantung berdampak langsung terhadap keterlambatan sistem. Dengan demikian, sistem menunjukkan performa transmisi yang konsisten dalam berbagai kondisi BPM yang digunakan pada pengujian.

B. Pengujian Akurasi Data

Pengujian performa *real-time* dilakukan untuk menilai tingkat akurasi dan kecepatan transmisi data antara perangkat ESP32 dan aplikasi. Tujuan utama dari pengujian ini adalah untuk memverifikasi bahwa nilai *Beats Per Minute* (BPM) yang dihasilkan oleh sensor pada ESP32 dapat dikirimkan dan ditampilkan pada aplikasi secara tepat dengan *delay* serendah mungkin. Proses pengujian menggunakan sinyal BPM simulasi yang dihasilkan oleh ESP32 dan dibandingkan dengan nilai yang diterima serta divisualisasikan pada aplikasi.

Rentang nilai BPM yang diuji berada antara 30 hingga 200 BPM, mencakup seluruh rentang fisiologis denyut jantung manusia, termasuk kondisi *bradycardia*, normal, hingga *tachycardia*. Interval pengujian ditetapkan setiap 20 BPM untuk memastikan cakupan yang representatif serta efisiensi dalam proses analisis performa sistem.

TABEL 5
Pengujian Akurasi Data

No	Nilai BPM ESP32 yang digenerate di serial monitor arduino IDE	Nilai BPM di Aplikasi	Selisih (BPM)	Akurasi (%)
1	30	30	0	100 %
2	50	50	0	100 %
3	70	70	0	100 %
4	90	90	0	100 %

5	110	110	0	100 %
6	130	130	0	100 %
7	150	150	0	100 %
8	170	170	0	100 %
9	190	190	0	100 %

Berdasarkan hasil pengujian yang disajikan pada Tabel, sistem menunjukkan performa optimal dengan tingkat akurasi sebesar 100% pada seluruh titik pengujian. Tidak ditemukan perbedaan antara nilai *Beats Per Minute* (BPM) yang dihasilkan oleh ESP32 dan nilai yang ditampilkan pada aplikasi, yang menandakan proses transmisi dan pemrosesan data berlangsung secara akurat tanpa kehilangan data maupun kesalahan konversi. Konsistensi akurasi 100% pada rentang BPM 30 hingga 190 menunjukkan tingkat stabilitas sistem yang tinggi dalam menangani berbagai kondisi denyut jantung.

Hasil ini sangat relevan untuk aplikasi medis, di mana ketepatan data menjadi elemen krusial dalam proses diagnosis dan pemantauan kondisi pasien. Sistem terbukti mampu menangani kondisi *bradycardia* (30–50 BPM), denyut jantung normal (70–90 BPM), serta *tachycardia* (110–190 BPM) dengan tingkat presisi yang setara.

Keberhasilan pengujian ini mengindikasikan bahwa algoritma pemrosesan sinyal pada ESP32 dan protokol komunikasi yang digunakan telah diimplementasikan secara efektif. Ketidaan deviasi nilai dalam pengujian seluruh rentang BPM mengonfirmasi bahwa sistem layak diandalkan untuk pemantauan jantung secara kontinu dan jangka panjang.

C. Pengujian fungsionalitas Backend dan database

Pengujian database bertujuan memastikan integritas data, kinerja, dan keandalan penyimpanan informasi medis pada sistem monitoring ECG real-time. Basis data yang digunakan adalah PostgreSQL yang di-hosting di Railway, dengan dua tabel utama: *users* untuk akun pengguna dan *ecg_results* untuk hasil pemeriksaan.

Pengujian dilakukan menggunakan metode *black-box testing* melalui pemanggilan REST API dengan *curl* pada *Command Prompt* Windows. Server API dibangun dengan Node.js dan Express.js, sedangkan pengujian mencakup koneksi, konsistensi data, autentikasi, operasi CRUD, performa, dan *stress test*.

Proses pengujian melibatkan pengukuran *response time*, validasi data, dan observasi perilaku sistem saat menerima permintaan bersamaan (*concurrent requests*). Hasil pengujian menunjukkan seluruh fungsi berjalan baik, koneksi stabil, data tersimpan sesuai permintaan, dan waktu respons rata-rata berada dalam batas yang dapat diterima untuk aplikasi medis waktu nyata.

1) Connectivity Testing

Pengujian koneksi bertujuan untuk memastikan bahwa server backend terhubung dengan baik ke database dan siap menangani permintaan. Dua endpoint diuji: */health*

untuk memeriksa status sistem secara umum, dan */api/test* untuk memverifikasi bahwa API dapat merespons dengan benar. Hasil menunjukkan bahwa koneksi database terjalin dengan baik, status sistem baik, dan response time berada di kisaran 1,34 detik.

TABEL 6
pengujian Konektivitas

Permintaan (Request)	Hasil pengujian
curl -w "Response Time: %{{time_total}s}" https://ecg-heartbeat-api-production.up.railway.app/health	Koneksi ke server berhasil. Status database: <i>connected</i> . Response time: 1,347 s.
curl -w "Response Time: %{{time_total}s}" https://ecg-heartbeat-api-production.up.railway.app/api/test	API berfungsi normal. Status: <i>success</i> . Response time: 1,348 s.

Tabel di atas menunjukkan bahwa koneksi antara *client* dan *server API* berjalan dengan baik. Semua *endpoint* berhasil merespons tanpa *error*, dan status basis data terdeteksi aktif (*connected*). Waktu respons sekitar 1,34 detik masih tergolong dapat diterima untuk sistem berbasis *cloud* yang membutuhkan komunikasi data yang stabil dan responsif

2) Data Integrity Testing

Tujuan pengujian ini adalah memvalidasi apakah data pengguna dan data hasil pemeriksaan ECG disimpan dengan akurat dan sesuai struktur. Hasil menunjukkan bahwa field pada tabel *users* dan *ecg_results* lengkap dan format data valid. Logika klasifikasi BPM juga bekerja dengan sempurna pada data riwayat ECG.

TABEL 7
Pengujian Integritas Data

Permintaan (Request)	Hasil Pengujian
curl https://ecg-heartbeat-api-production.up.railway.app/api/users/all	Terdapat 9 pengguna dalam database. Semua field valid dan terisi lengkap. Format data konsisten.

curl https://ecg-heartbeat-api-production.up.railway.app/api/ecg/history/7	Terdapat 18 riwayat ECG untuk user ID 7. Klasifikasi BPM sesuai: Bradikardia, Normal, Takikardia.
--	---

Tabel ini menampilkan hasil pengujian integritas data yang menunjukkan bahwa seluruh informasi dalam database disimpan dengan format yang benar dan sesuai skema. Klasifikasi kondisi jantung berdasarkan BPM juga berjalan sesuai logika yang telah ditentukan, tanpa adanya kesalahan klasifikasi.

3) Authentication Testing

Pengujian autentikasi dilakukan untuk memastikan keamanan akses dan validasi kredensial pengguna. Permintaan login berhasil mendapatkan token autentikasi dan data user yang sesuai. Selain itu, endpoint profil juga dapat menampilkan data pengguna dengan benar.

TABEL 8
Pengujian Autentikasi

Permintaan (Request)	Hasil Pengujian
curl -X POST -H "Content-Type: application/json" -d '{"username":"chaer","password":"chaer123"}' https://ecg-heartbeat-api-production.up.railway.app/api/auth/login	Login berhasil. Token berhasil diterbitkan. Validasi user ID 7: Sukses. Response time: 1,061 s.
curl https://ecg-heartbeat-api-production.up.railway.app/api/profile/7	Data profil user berhasil ditampilkan. Semua field lengkap. Response time: 1,143 s.

Pengujian autentikasi menunjukkan bahwa sistem login dapat memverifikasi identitas pengguna dengan benar dan menghasilkan token autentikasi. Selain itu, permintaan untuk melihat data profil berhasil dijalankan dan menampilkan informasi pengguna secara lengkap, menandakan kontrol akses bekerja dengan baik.

4) CRUD Operations Testing

Salah satu pengujian penting adalah memastikan operasi Create, Read, Update, dan Delete dapat berjalan dengan baik. Dalam pengujian ini, data ECG baru dikirim ke server melalui endpoint /api/ecg/save. Hasil menunjukkan bahwa data berhasil tersimpan dengan klasifikasi status dan kondisi yang tepat.

TABEL 9
Pengujian CRUD

Permintaan (Request)	Hasil Pengujian
curl -X POST -H "Content-Type: application/json" -d '{"userId":7,"username":"chaer","bpm":75}' https://ecg-heartbeat-api-production.up.railway.app/api/ecg/save	Data berhasil disimpan ke database dengan ID baru. Klasifikasi otomatis (BPM 75 = Normal) berhasil.

Hasil pengujian create menunjukkan bahwa data hasil pemeriksaan dapat disimpan dengan benar. Sistem secara otomatis menghasilkan klasifikasi kondisi (Normal) berdasarkan nilai BPM, dan menyertakan informasi waktu secara otomatis. Ini membuktikan bahwa business logic backend berjalan dengan baik.

5) Stress Testing

Stress testing dilakukan untuk menguji kestabilan sistem ketika menerima permintaan secara bersamaan. Tiga permintaan identik dikirim secara berurutan dan hasil menunjukkan peningkatan performa karena kemungkinan caching atau optimasi internal.

TABEL 10
Pengujian Stress

Permintaan (Request)	Hasil Pengujian
Request 1: curl https://ecg-heartbeat-api-production.up.railway.app/api/test	Response time: 1,094 s
Request 2: curl https://ecg-heartbeat-api-production.up.railway.app/api/test	Response time: 1,070 s
Request 3: curl https://ecg-heartbeat-api-production.up.railway.app/api/test	Response time: 0.862 s

Tabel ini menunjukkan bahwa sistem tetap responsif saat menerima permintaan berturut-turut. Terjadi peningkatan performa pada permintaan ketiga, yang menunjukkan adanya kemungkinan optimasi seperti caching atau thread pooling yang bekerja di server API.

Berdasarkan rangkaian pengujian yang telah dilakukan, sistem basis data pada aplikasi monitoring ECG memenuhi kriteria teknis untuk implementasi pada lingkungan produksi. Seluruh aspek pengujian—meliputi konektivitas, integritas data, autentikasi, operasi CRUD, performa, dan ketahanan

terhadap beban—menunjukkan hasil yang konsisten dan sesuai dengan fungsi yang diharapkan. Penggunaan *PostgreSQL* yang dihosting melalui platform *Railway* terbukti mampu mendukung alur komunikasi antara perangkat *ESP32*, *server API*, dan aplikasi mobile *Android* secara stabil. Rata-rata *response time* sebesar 1,16 detik berada dalam kisaran yang dapat diterima untuk sistem pemantauan medis non-kritis.

Pengujian juga menunjukkan bahwa mekanisme autentikasi pengguna bekerja secara andal, tanpa ditemukannya anomali atau kegagalan transaksi selama proses evaluasi berlangsung. Rekomendasi untuk pengembangan lebih lanjut mencakup optimalisasi performa melalui penerapan *indexing* lanjut dan mekanisme *caching*, guna menurunkan *response time* menuju nilai yang lebih optimal.

D. Pengujian Notifikasi Abnormal

Pengujian dilakukan dengan mengirimkan data hasil pemeriksaan ECG ke endpoint `/api/ecg/save` menggunakan nilai BPM yang bervariasi. Sistem akan memproses nilai BPM tersebut dan memberikan respons status serta kondisi jantung. Jika BPM berada di luar rentang normal (60–100 BPM), sistem harus mengembalikan notifikasi "Denyut jantung tidak normal" bersamaan dengan klasifikasi kondisi:

- Bradikardia → BPM < 60
- Normal → BPM 60–100
- Takikardia → BPM > 100

Pengujian dilakukan dengan memberikan input nilai BPM yang mewakili tiga kondisi utama jantung: bradikardia, normal, dan takikardia. Sistem diharapkan dapat mengidentifikasi kondisi ini secara otomatis dan menampilkan notifikasi khusus untuk kasus abnormal. Hasil dari masing-masing pengujian ditampilkan dalam tabel berikut, yang menunjukkan kecocokan antara nilai BPM, status yang dikembalikan oleh sistem, dan apakah notifikasi berhasil ditampilkan.

TABEL 11
Pengujian Notifikasi Sinyal Abnormal

Nilai BPM	Status	Kondisi	Notifikasi yang Ditampilkan	Hasil
30	Abnormal	Bradikardia	Denyut jantung tidak normal (Bradikardia)	Berhasil
50	Abnormal	Bradikardia	Denyut jantung tidak normal (Bradikardia)	Berhasil
70	Normal	Normal	Tidak ada notifikasi abnormal	Berhasil

90	Normal	Normal	Tidak ada notifikasi abnormal	Berhasil
110	Abnormal	Takikardia	Denyut jantung tidak normal (Takikardia)	Berhasil
130	Abnormal	Takikardia	Denyut jantung tidak normal (Takikardia)	Berhasil
150	Abnormal	Takikardia	Denyut jantung tidak normal (Takikardia)	Berhasil
170	Abnormal	Takikardia	Denyut jantung tidak normal (Takikardia)	Berhasil
190	Abnormal	Takikardia	Denyut jantung tidak normal (Takikardia)	Berhasil

Tabel di atas menunjukkan bahwa seluruh skenario pengujian berhasil dijalankan sesuai dengan ekspektasi. Notifikasi muncul dengan benar pada nilai BPM yang berada di luar rentang normal, sementara tidak ditampilkan pada BPM yang tergolong normal. Hal ini membuktikan bahwa fitur notifikasi abnormal telah diimplementasikan secara fungsional, logis, dan andal. Fitur ini memberikan nilai tambah penting pada sistem monitoring ECG karena memungkinkan pengguna menerima peringatan dini terhadap kondisi jantung yang tidak stabil.

E. Analisis Umum Pengujian

Pengujian dilakukan terhadap berbagai komponen utama sistem, termasuk konektivitas, integritas data, autentikasi, performa, operasi CRUD, notifikasi sinyal abnormal, serta pengujian database secara umum. Seluruh pengujian dilakukan menggunakan metode black-box melalui pengiriman permintaan REST API menggunakan tool curl, serta dilakukan pengamatan terhadap hasil keluaran sistem berupa respons JSON, waktu respons, dan keakuratan proses klasifikasi data.

Berdasarkan pengujian konektivitas, seluruh endpoint berhasil diakses dengan status koneksi database yang aktif, serta response time berada dalam rentang antara 1.06 hingga 1.35 detik. Pengujian integritas data menunjukkan bahwa data tersimpan secara konsisten dan klasifikasi BPM dilakukan sesuai aturan yang ditentukan. Sistem autentikasi berhasil memverifikasi pengguna dan menghasilkan token valid, serta seluruh permintaan profil dan data pengguna berhasil dikembalikan tanpa error.

Operasi CRUD, khususnya pembuatan data hasil ECG, berjalan sesuai dengan fungsi yang diharapkan. Data baru berhasil ditambahkan ke database, dan klasifikasi status

serta kondisi jantung berdasarkan BPM dilakukan secara otomatis. Pengujian performa menunjukkan rata-rata waktu respons sistem sebesar 1.16 detik dengan variansi rendah, dan tidak ditemukan anomali atau kegagalan saat dilakukan pengujian berulang.

Pengujian stres menunjukkan bahwa sistem mampu menangani beberapa permintaan secara berturut-turut tanpa penurunan kinerja. Bahkan terjadi peningkatan performa pada permintaan terakhir, yang mengindikasikan potensi mekanisme optimasi internal seperti caching.

Fitur notifikasi sinyal abnormal telah diuji dengan berbagai nilai BPM yang mewakili kondisi bradikardia, normal, dan takikardia. Sistem berhasil menampilkan notifikasi hanya pada kasus abnormal, sesuai dengan logika klasifikasi, tanpa menampilkan peringatan pada kondisi BPM normal.

Secara keseluruhan, hasil pengujian menunjukkan bahwa setiap fungsi inti sistem bekerja sesuai dengan spesifikasi rancangan. Data yang disimpan akurat, sistem stabil, dan seluruh alur pengolahan informasi dari input sensor hingga penyajian data ke antarmuka aplikasi telah tervalidasi dengan baik.

V. KESIMPULAN

Berdasarkan hasil perancangan, implementasi, dan pengujian sistem monitoring denyut jantung berbasis sinyal ECG menggunakan aplikasi mobile Android, dapat disimpulkan bahwa aplikasi yang dikembangkan telah berhasil memanfaatkan perangkat ESP32 dan teknologi WebSocket untuk menampilkan grafik sinyal ECG secara langsung dan mengirimkan data ke cloud dengan rata-rata delay transmisi di bawah 100 milidetik. Data yang dikirim berhasil tersimpan di basis data PostgreSQL yang di-hosting melalui platform Railway tanpa kehilangan atau kerusakan. Sistem secara otomatis mengklasifikasikan kondisi jantung berdasarkan nilai *beats per minute* (BPM) ke dalam tiga kategori, yaitu bradikardia, normal, dan takikardia, dengan akurasi 100% pada pengujian terhadap 11 sampel data dalam rentang 30 hingga 190 BPM. Fitur notifikasi otomatis untuk kondisi abnormal juga bekerja sesuai rancangan, menampilkan notifikasi pada tiga kondisi abnormal dan tidak menampilkan pada dua kondisi normal. Integrasi backend melalui tujuh endpoint REST API telah diuji dalam kategori koneksi, autentikasi, operasi CRUD, dan pengambilan data, seluruhnya memberikan respons *success* dengan rata-rata waktu respons 1,16 detik dan variansi yang rendah, mencerminkan stabilitas performa sistem. Basis data berhasil menyimpan data pengguna dan hasil pemeriksaan ECG dengan struktur yang benar dan valid, menghasilkan *timestamp* serta ID secara otomatis, dan memberikan hasil klasifikasi sesuai logika. Semua fitur utama, termasuk visualisasi sinyal ECG, riwayat pemeriksaan, notifikasi otomatis, tampilan profil, dan *logout*, berfungsi sesuai tujuan pengembangan serta memenuhi rumusan masalah dan batasan penelitian.

Berdasarkan evaluasi sistem dan keterbatasan penelitian, disarankan agar pengembangan selanjutnya mencakup integrasi langsung dengan perangkat sensor ECG aktual untuk meningkatkan akurasi sinyal hingga mendekati kondisi klinis, serta pengembangan sistem klasifikasi BPM menggunakan algoritma *machine learning* guna mendeteksi gangguan jantung yang lebih kompleks, seperti aritmia atau fibrilasi atrium. Implementasi enkripsi data saat transmisi dan

penyimpanan di server sangat dianjurkan untuk menjaga privasi pengguna, dengan penambahan mekanisme autentikasi dinamis dan perlindungan dari akses tidak sah. Uji coba pada kelompok pengguna nyata, seperti 10–20 orang, perlu dilakukan untuk memperoleh data aktual penggunaan serta umpan balik dari pasien dan tenaga medis. Selain itu, integrasi aplikasi pasien dengan antarmuka web untuk dokter diharapkan dapat memungkinkan pemantauan sinyal ECG secara sinkron oleh tenaga medis dengan notifikasi yang bersifat real-time.

VI. REFERENSI

- [1] “Cardiovascular diseases (CVDs).” Accessed: Dec. 14, 2024. [Online]. Available: <https://www.who.int/news-room/fact-sheets/detail/cardiovascular-diseases-%28cvds%29>
- [2] L. Lu *et al.*, “Wearable Health Devices in Health Care: Narrative Systematic Review (Preprint),” Mar. 26, 2020. doi: 10.2196/preprints.18907.
- [3] M. Kumar, S. K. Singh, and S. Kim, “Hybrid deep learning-based cyberthreat detection and IoMT data authentication model in smart healthcare,” *Future Generation Computer Systems*, vol. 166, p. 107711, May 2025, doi: 10.1016/J.FUTURE.2025.107711.
- [4] N. Bashi, M. Karunanithi, F. Fatehi, H. Ding, and D. Walters, “Remote Monitoring of Patients With Heart Failure: An Overview of Systematic Reviews,” *J Med Internet Res*, vol. 19, no. 1, p. e18, Jan. 2017, doi: 10.2196/jmir.6571.
- [5] Ummu Hani, Esti Nur Janah, and Wawan Hediyanto, “Asuhan Keperawatan Pada Ny.T Dengan Gangguan Sistem Kardiovaskular: Hipertensi Di Ruang Dahlia RSUD dr. Soeselo Kabupaten Tegal,” *DIAGNOSA: Jurnal Ilmu Kesehatan dan Keperawatan*, vol. 1, no. 4, pp. 172–181, Sep. 2023, doi: 10.59581/diagnosa-widyakarya.v1i4.1306.
- [6] K. Ramadhani, Mp. Rachmawati Widyaningrum, and B. Mahasiswa Gizi Dan Kesehatan, “BUKU AJAR DASAR-DASAR ANATOMI DAN FISIOLOGI TUBUH MANUSIA.”
- [7] L. V. Ugi, F. Y. Suratman, and U. Sunarya, “Electrocardiogram feature selection and performance improvement of sleep stages classification using grid search,” *Bulletin of Electrical Engineering and Informatics*, vol. 11, no. 4, pp. 2033–2043, Aug. 2022, doi: 10.11591/eei.v11i4.3529.
- [8] P. Madona, R. I. Basti, and M. M. Zain, “PQRST wave detection on ECG signals,” *Gac Sanit*, vol. 35, pp. S364–S369, Jan. 2021, doi: 10.1016/j.gaceta.2021.10.052.
- [9] S. Laksono, “INTERPRETASI EKG NORMAL PRAKTIS BAGI PEMULA: SUATU TINJAUAN MINI,” *JURNAL KEDOKTERAN*, vol. 7, no. 1, p. 1, Dec. 2021, doi: 10.36679/kedokteran.v7i1.408.
- [10] H. Li, X. Wei, F. Zheng, J. Wen, and G. Sun, “Atrial fibrillation in preterm neonates: A case study,” *J Electrocardiol*, vol. 65, pp. 66–68, Mar. 2021, doi: 10.1016/J.JELECTROCARD.2021.01.009.
- [11] A. Sapra, A. Malik, and P. Bhandari, “Vital Sign Assessment,” *StatPearls*, May 2023, Accessed: Jul.

- [12] 15, 2025. [Online]. Available: <https://www.ncbi.nlm.nih.gov/books/NBK553213/> R. Quinn, N. Leader, G. Lebovic, C. M. Chow, and P. Dorian, "Accuracy of Wearable Heart Rate Monitors During Exercise in Sinus Rhythm and Atrial Fibrillation," *J Am Coll Cardiol*, vol. 83, no. 12, pp. 1177–1179, Mar. 2024, doi: 10.1016/J.JACC.2024.01.024.
- [13] "Part 7.3: Management of symptomatic bradycardia and tachycardia," in *Circulation*, Dec. 2005. doi: 10.1161/CIRCULATIONAHA.105.166558.
- [14] R. Farhandika, M. K. Sabariah, and M. Adrian, "Penerapan Arsitektur REST API pada Aplikasi Backend Manajemen Informasi Fakultas Industri Kreatif (MI- FIK) Universitas Telkom," *LOGIC: Jurnal Penelitian Informatika*, vol. 2, no. 1, pp. 40–51–40–51, Jun. 2024, doi: 10.25124/LOGIC.V2I1.7530.
- [15] C. Chaubey and A. Sharma, "The integrated development environment (IDE) for application development: Android studio and its tools," *AIP Conf Proc*, vol. 2427, no. 1, Feb. 2023, doi: 10.1063/5.0116494/2866350.
- [16] B. P. D. P., S. Robertus, J. O. Chandra, and W. Andriyani, "A Comparative Study of Java and Kotlin for Android Mobile Application Development," Dec. 2020.
- [17] E. Simanjuntak and N. Surantha, "Multiple time series database on microservice architecture for IoT-based sleep monitoring system," *J Big Data*, vol. 9, no. 1, Dec. 2022, doi: 10.1186/S40537-022-00658-4.
- [18] C. Pintavirooj, T. Keatsamarn, and T. Treebupachatsakul, "Multi-Parameter Vital Sign Telemedicine System Using Web Socket for COVID-19 Pandemics," *Healthcare*, vol. 9, no. 3, p. 285, Mar. 2021, doi: 10.3390/HEALTHCARE9030285.
- [19] M. A. Rahman, Y. Li, T. Nabeed, and M. T. Rahman, "Remote monitoring of heart rate and ECG signal using ESP32," *Proceedings - 2021 4th International Conference on Advanced Electronic Materials, Computers and Software Engineering, AEMCSE 2021*, pp. 604–610, Mar. 2021, doi: 10.1109/AEMCSE51986.2021.00127.
- [20] "Node.js — About Node.js®." Accessed: Jul. 15, 2025. [Online]. Available: <https://nodejs.org/en/about>
- [21] W. Hadinata and L. Stianingsih, "ANALISIS PERBANDINGAN PERFORMA RESTFULL API ANTARA EXPRESS.JS DENGAN LARAVEL FRAMEWORK," *Jurnal Informatika dan Teknik Elektro Terapan*, vol. 12, no. 1, pp. 2830–7062, Jan. 2024, doi: 10.23960/JITET.V12I1.3845.
- [22] "Report: Railway Business Breakdown & Founding Story | Contrary Research." Accessed: Jul. 15, 2025. [Online]. Available: <https://research.contrary.com/company/railway>
- [23] "Developing a CRUD Node.js Application with PostgreSQL - DEV Community." Accessed: Jul. 15, 2025. [Online]. Available: <https://dev.to/glaucia86/developing-a-crud-node-js-application-with-postgresql-4c9o>
- [24] A. Decan, T. Mens, P. R. Mazrae, and M. Golzadeh, "On the Use of GitHub Actions in Software Development Repositories", doi: 10.5281/zenodo.6634682.
- [25] S. K. Hussain, M. V. Vardhan, and P. Thimmaiah, "Issue 6 | ISSN: 2456-3315 IJRTI2306162 International Journal for Research Trends and Innovation (www," 2023. [Online]. Available: www.ijrti.org
- [26] M. H. Nornaim, N. A. Abdul-Kadir, F. K. C. Harun, and M. A. A. Razak, "A Wireless ECG Device with Mobile Applications for Android," in *2020 7th International Conference on Electrical Engineering, Computer Sciences and Informatics (EECSI)*, IEEE, Oct. 2020, pp. 168–171. doi: 10.23919/EECSI50503.2020.9251871.
- [27] "IoT Based ECG Monitoring with AD8232 ECG Sensor & ESP32." Accessed: Jul. 16, 2025. [Online]. Available: <https://how2electronics.com/iot-ecg-monitoring-ad8232-sensor-esp32/>
- [28] K. K. Mujeeb Rahman, N. N. Mohamed, R. Zidan, I. Alsarraj, and B. Hasan, "IOT-Based Wireless Patient Monitor Using ESP32 Microcontroller," *2023 24th International Arab Conference on Information Technology, ACIT 2023*, 2023, doi: 10.1109/ACIT58888.2023.10453847.
- [29] "Ashithaby/IoT-based-low-cost-ECG-and-heart-monitoring-system-with-ESP32: IoT-based low-cost ECG monitoring system offers a transformative solution for democratizing access to cardiac care. By leveraging the power of the Internet of Things, this system provides continuous and affordable monitoring of the heart's electrical activity." Accessed: Jul. 16, 2025. [Online]. Available: <https://github.com/Ashithaby/IoT-based-low-cost-ECG-and-heart-monitoring-system-with-ESP32>
- [30] N. Spicher, A. Klingenberg, V. Purrucker, and T. M. Deserno, "Edge computing in 5G cellular networks for real-time analysis of electrocardiography recorded with wearable textile sensors," *Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS*, vol. 2021-January, pp. 1735–1739, Jul. 2021, doi: 10.1109/EMBC46164.2021.9630875.
- [31] M. Ghaleb and F. Azzedin, "Towards Scalable and Efficient Architecture for Modeling Trust in IoT Environments," *Sensors 2021, Vol. 21, Page 2986*, vol. 21, no. 9, p. 2986, Apr. 2021, doi: 10.3390/S21092986.
- [32] "walidamriou/ECGGenerator: Open source low-cost ECG Generator device." Accessed: Aug. 05, 2025. [Online]. Available: <https://github.com/walidamriou/ECGGenerator>