



**TUGAS AKHIR - EC184801**

**APLIKASI TELEKARDIOLOGI UNTUK MOBILE  
ANDROID DILENGKAPI SENSOR ECG**

**Robby Aldriyanto Raffly**  
**NRP 0721 17 4000 0011**

**Dosen Pembimbing**  
**Arief Kurniawan, ST., MT.**  
**Dr. I Ketut Eddy Purnama, ST., MT.**

**DEPARTEMEN TEKNIK KOMPUTER**  
**Fakultas Teknologi ELEKTRO DAN INFORMATIKA CERDAS**  
**Institut Teknologi Sepuluh Nopember**  
**Surabaya 2021**



**TUGAS AKHIR - EC184801**

**APLIKASI TELEKARDIOLOGI UNTUK MOBILE  
ANDROID DILENGKAPI SENSOR ECG**

**Robby Aldriyanto Raffly  
NRP 0721 17 4000 0011**

**Dosen Pembimbing  
Arief Kurniawan, ST., MT.  
Dr. I Ketut Eddy Purnama, ST., MT.**

**DEPARTEMEN TEKNIK KOMPUTER  
Fakultas Teknologi ELEKTRO DAN INFORMATIKA CERDAS  
Institut Teknologi Sepuluh Nopember  
Surabaya 2021**

## **PERNYATAAN KEASLIAN TUGAS AKHIR**

Dengan ini saya menyatakan bahwa isi sebagian maupun keseluruhan Tugas Akhir sada dengan judul **”Aplikasi Telekardiologi untuk Mobile Android Dilengkapi Sensor ECG”** adalah benar-benar hasil karya intelektual mandiri, diselesaikan tanpa menggunakan bahan-bahan yang tidak diijinkan da bukan karya pi-hak lain yang saya akui sebagai karya sendiri.

Semua referensi yang dikutip maupun dirujuk telah ditulis secara lengkap pada daftar pustaka.

Apabila ternyata pernyataan ini tidak benar, saya bersedia menerima sanksi sesuai peraturan yang berlaku.

Surabaya, 14 Juni 2021

Robby Aldriyanto Raffly  
0721 17 4000 0011

*Halaman ini sengaja dikosongkan*

## **ABSTRAK**

Nama Mahasiswa : Robby Aldriyanto Raffly  
Judul Tugas Akhir : Aplikasi Telekardiologi untuk Mobile Android Dilengkapi Sensor ECG  
Pembimbing : 1. Arief Kurniawan, ST., MT.  
                  2. Dr. I Ketut Eddy Purnama, ST., MT.

Jantung merupakan organ yang sangat penting bagi tubuh. Kematian karena penyakit jantung merupakan salah satu kematian terbanyak di dunia. Penyakit jantung dapat diketahui sejak dini dengan cara memonitoring sinyal detak jantung. Pada Tugas Akhir ini kami mengajukan judul penelitian tentang implementasi sistem telekardiologi. Implementasi sistem telekardiologi yang kami ajukan adalah membuat aplikasi android yang dapat digunakan untuk mengambil data sinyal ECG dari arduino melalui modul bluetooth HC-05. Sinyal ECG tersebut didapatkan arduino dengan menggunakan sensor AD8232. Aplikasi juga memiliki fitur untuk mengirimkan sinyal ECG kepada dokter spesialis jantung sehingga dokter dapat men-diagnosa sinyal ECG pasien tanpa harus mengecek pasien dengan bertemu secara langsung. Aplikasi juga memiliki fitur chat dengan dokter agar pasien dapat berkonsultasi jarak jauh. Hasil yang diharapkan dari penelitian ini adalah aplikasi dapat mengambil data melalui komunikasi bluetooth, mengirimkan data sinyal ECG kepada dokter, dan chatting.

Kata Kunci : Telekardiologi, ECG, Pengolahan Sinyal, Aplikasi Mobile

*Halaman ini sengaja dikosongkan*

# **ABSTRACT**

*Name : Robby Aldriyanto Raffly  
Title : Telecardiology App for Android Mobile  
Equipped with ECG Sensor  
Advisors : 1. Arief Kurniawan, ST., MT.  
2. Dr. I Ketut Eddy Purnama, ST., MT.*

*The heart is a very important organ for the body. Death from heart disease is one of the most common deaths in the world. Heart disease can be detected early by monitoring the heartbeat signal. In this final project, we propose a research title on the implementation of telecardiology systems. The implementation of the telecardiology system that we propose is to create an android application that can be used to retrieve ECG signal data from Arduino via the HC-05 bluetooth module. The ECG signal is obtained by Arduino using the AD8232 sensor. The application also has a feature to send an ECG signal to a cardiologist so that doctors can diagnose a patient's ECG signal without having to check the patient in person. The application also has a chat feature with doctors so that patients can consult remotely. The expected result of this research is that the application can retrieve data via bluetooth communication, send ECG signal data to doctors, and chat.*

*Keywords : Telecardiology, ECG, Signal Processing, Mobile Apps*

*Halaman ini sengaja dikosongkan*

## KATA PENGANTAR

Puji dan syukur kehadirat Tuhan Yang Maha Esa atas segala karunia-Nya, penulis dapat menyelesaikan penelitian ini dengan judul **Aplikasi Telekardiologi untuk Mobile Android Dilengkapi Sensor ECG**.

Penelitian ini disusun dalam rangka pemenuhan bidang riset di Departemen Teknik Komputer ITS, Bidang Studi Telematika, serta digunakan sebagai persyaratan menyelesaikan pendidikan S1. Oleh karena itu, penulis mengucapkan terima kasih kepada:

1. Keluarga, Ibu, Ayah, dan Saudara tercinta yang telah memberikan dorongan spiritual dan material dalam penyelesaian buku penelitian ini.
2. Bapak Dr. Supeno Mardi Susiki Nugroho, ST., MT. selaku Kepala Departemen Teknik Komputer, Fakultas Teknik Elektro dan Informatika Cerdas, Institut Teknologi Sepuluh Nopember.
3. Bapak Arief Kurniawan, ST., MT. selaku dosen pembimbing I dan Bapak Dr. I Ketut Eddy Purnama, ST., MT. selaku dosen pembimbing II yang selalu memberikan arahan selama mengerjakan penelitian tugas akhir ini.
4. Bapak-ibu dosen pengajar Departemen Teknik Komputer, atas pengajaran, bimbingan, serta perhatian yang diberikan kepada penulis selama ini.
5. Seluruh teman-teman dari angkatan e57, Teknik Komputer, Laboratorium B401, dan B201 Teknik Komputer ITS.

Kesempurnaan hanya milik Allah SWT, untuk itu penulis memohon segenap kritik dan saran yang membangun. Semoga penelitian ini dapat memberikan manfaat bagi kita semua. Amin.

Surabaya, Maret 2021

Robby Aldriyanto Raffly

*Halaman ini sengaja dikosongkan*

# **DAFTAR ISI**

<b>Abstrak</b>	i
<b>Abstract</b>	iii
<b>KATA PENGANTAR</b>	v
<b>DAFTAR ISI</b>	vii
<b>DAFTAR GAMBAR</b>	xii
<b>DAFTAR TABEL</b>	xiii
<b>NOMENKLATUR</b>	xv
<b>1 PENDAHULUAN</b>	1
1.1 Latar belakang . . . . .	1
1.2 Perumusan Masalah . . . . .	2
1.3 Tujuan . . . . .	2
1.4 Batasan masalah . . . . .	2
1.5 Sistematika Penulisan . . . . .	3
<b>2 TINJAUAN PUSTAKA</b>	5
2.1 Dasar Teori . . . . .	5
2.1.1 Kardiologi . . . . .	5
2.1.2 ECG atau EKG . . . . .	7
2.1.3 Jenis ECG . . . . .	7
2.1.4 Sinyal ECG . . . . .	9
2.1.5 AD8232 . . . . .	10

2.1.6	AD8232 Placement . . . . .	10
2.1.7	Modul Bluetooth HC-05 . . . . .	11
2.1.8	Arduino Nano . . . . .	13
2.2	Penelitian Terkait . . . . .	14
2.2.1	<i>IoT based Real Time ECG Monitoring System using Cypress WICED</i> . . . . .	14
2.2.2	<i>An IoT-cloud Based Wearable ECG Monitoring System for Smart Healthcare</i> . . . . .	14
<b>3</b>	<b>DESAIN DAN IMPLEMENTASI</b>	<b>15</b>
3.1	Desain Sistem . . . . .	15
3.2	Desain Perangkat . . . . .	16
3.3	Desain UI Aplikasi . . . . .	17
3.4	Desain Database . . . . .	21
3.5	Akuisisi Data . . . . .	23
3.6	Perekaman Data . . . . .	24
3.7	Penyimpanan Data . . . . .	26
3.8	Menampilkan Data . . . . .	27
3.9	Diagnosa Data . . . . .	30
3.10	Fitur Chat . . . . .	31
<b>4</b>	<b>HASIL DAN PENGUJIAN</b>	<b>33</b>
4.1	Pengujian Rekaman Data Selama 15 Menit . . . . .	33
4.2	Pengujian Persentase Data Hilang . . . . .	36
4.3	Pengujian Waktu Pengiriman Data Menuju Database	38
4.4	Pengujian kesesuaian data dari arduino dan data di aplikasi . . . . .	41
4.5	Pengujian Kesesuaian Grafik ECG . . . . .	53
<b>5</b>	<b>PENUTUP</b>	<b>57</b>

5.1	Kesimpulan . . . . .	57
5.2	Saran . . . . .	57
<b>DAFTAR PUSTAKA</b>		<b>59</b>
<b>Biografi Penulis</b>		<b>63</b>

*Halaman ini sengaja dikosongkan*

# DAFTAR GAMBAR

2.1	Gelombang ECG. . . . .	9
2.2	Sensor AD8232. . . . .	10
2.3	Penempatan AD8232 1 (kiri), Penempatan AD8232 2 (kanan) . . . . .	11
3.1	Arsitektur (a) dan Blok Diagram (b) Alur Kerja Sistem	15
3.2	Desain Perangkat ECG . . . . .	16
3.3	Desain UI pada aplikasi dokter. . . . .	17
3.4	Desain UI pada aplikasi pasien. . . . .	19
3.5	Desain database. . . . .	21
3.6	Perangkat ECG. . . . .	23
3.7	Diagram mekanisme akuisisi data. . . . .	24
3.8	Implementasi UI Rekam Sinyal ECG. . . . .	25
3.9	Tabel data_ecg. . . . .	27
3.10	Implementasi UI Grafik ECG dokter. . . . .	28
3.11	Implementasi UI Grafik ECG pasien. . . . .	29
3.12	Implementasi UI Riwayat Diagnosa. . . . .	30
3.13	Implementasi UI Chatscreen Pasien. . . . .	31
3.14	Implementasi UI Chatscreen Dokter. . . . .	32
4.1	Percobaan 1 : (a)Data awal yang dikirim dari arduino, (b)Data akhir yang dikirim arduino. . . . .	42
4.2	Percobaan 1: (kiri) Data awal yang diterima aplikasi, (kanan) Data akhir yang diterima aplikasi. . . . .	43
4.3	Percobaan 2 : (a)Data awal yang dikirim dari arduino, (b)Data akhir yang dikirim arduino. . . . .	45

4.4	Percobaan 2: (kiri) Data awal yang diterima aplikasi, (kanan) Data akhir yang diterima aplikasi. . . . .	46
4.5	Percobaan 3 : (a)Data awal yang dikirim dari arduino, (b)Data akhir yang dikirim arduino. . . . .	47
4.6	Percobaan 3: (kiri) Data awal yang diterima aplikasi, (kanan) Data akhir yang diterima aplikasi. . . . .	48
4.7	Percobaan 4 : (a)Data awal yang dikirim dari arduino, (b)Data akhir yang dikirim arduino. . . . .	49
4.8	Percobaan 4: (kiri) Data awal yang diterima aplikasi, (kanan) Data akhir yang diterima aplikasi. . . . .	50
4.9	Percobaan 5 : (a)Data awal yang dikirim dari arduino, (b)Data akhir yang dikirim arduino. . . . .	51
4.10	Percobaan 5: (kiri) Data awal yang diterima aplikasi, (kanan) Data akhir yang diterima aplikasi. . . . .	52
4.11	Grafik ECG yang ditampilkan pada aplikasi. . . . .	54
4.12	Grafik ECG yang ditampilkan pada serial ploter Arduino IDE. . . . .	55

## DAFTAR TABEL

2.1	Spesifikasi Arduino Nano [2] . . . . .	13
4.1	Spesifikasi <i>Smartphone</i> yang Digunakan . . . . .	33
4.2	Spesifikasi <i>Smartphone</i> yang Digunakan . . . . .	33
4.3	Tabel hasil percobaan rekaman data dengan target 15 menit . . . . .	34
4.4	Tabel hasil percobaan persentase data hilang . . . . .	37
4.5	Tabel hasil percobaan durasi upload . . . . .	39

*Halaman ini sengaja dikosongkan*

# NUMENKLATUR

*TP* : *True Positive*

*TN* : *True Negative*

*FP* : *False Positive*

*FN* : *False Negative*

*Halaman ini sengaja dikosongkan*

# BAB 1

## PENDAHULUAN

Penelitian ini dilatarbelakangi oleh berbagai kondisi yang menjadi acuan. Selain itu juga terdapat beberapa permasalahan yang akan dijawab sebagai luaran dari penelitian.

### 1.1 Latar belakang

Telekardiologi berasal dari kata tele yang berarti jarak jauh dan kardiologi yang merupakan suatu cabang kedokteran yang berhubungan dengan studi dan perawatan kelainan-kelainan di sistem kardiovaskular, yaitu jantung, pembuluh darah, dan pembuluh nadi[3]. Penyakit jantung merupakan penyakit yang menjadi penyebab utama kematian secara global dalam 15 tahun terakhir. Dari 56,9 juta kematian di seluruh dunia pada tahun 2016, lebih dari setengah (54%) disebabkan oleh 10 penyebab teratas. Penyakit jantung iskemik dan stroke adalah pembunuh terbesar di dunia, yang apabila digabungkan jumlahnya menyebabkan 15,2 juta kematian pada tahun 2016 [4]. Penyakit kardiovaskular juga menjadi penyebab kematian nomor satu di Indonesia. Data dari Institute for Health Metrics and Evaluation, lembaga statistik kesehatan asal Amerika Serikat menyebutkan jumlah kematian akibat penyakit ini mencapai 36,3 persen dari total kematian di Indonesia pada 2016. Selanjutnya, kanker dan diabetes menjadi penyakit yang juga menimbulkan banyak kematian. Indonesia sendiri mendapat peringkat ke-3 di ASEAN setelah Laos dan Filipina dalam hal jumlah kematian yang disebabkan penyakit kardiovaskular pada tahun 2016 [5]. Salah satu cara untuk mengurangi resiko kematian yang disebabkan penyakit kardiovaskular adalah melakukan pemeriksaan jantung sejak dini. Pemeriksaan kesehatan jantung dapat menggunakan Electrocardiogram (ECG).

Electrocardiogram adalah tes yang menggunakan alat elektrokardiograf yang dapat menunjukkan seberapa cepat jantung seorang berdetak, dimana denyut jantung tersebut berupa impulse

listrik yang diukur oleh sensor. ECG digunakan untuk menampilkan aktivitas pada jantung, sehingga tenaga medis dapat mendiagnosis apakah detak jantung pada pasien tersebut normal atau tidak. Pemeriksaan rekaman ECG biasanya hanya dapat dilakukan dilakukan di rumah sakit dengan fasilitas lengkap. Hal tersebut membuat pasien jantung menjadi malas untuk memeriksakan kesehatan jantungnya ke Rumah Sakit. Terlebih lagi disaat kondisi pandemi COVID 19. Rumah Sakit merupakan salah satu tempat yang beresiko menyebabkan penyebaran virus COVID 19. Maka dari itu dibutuhkan sebuah sistem telekardiologi menggunakan aplikasi android agar pasien dan dokter dapat melakukan pemeriksaan secara online tanpa harus pergi ke Rumah Sakit.

## 1.2 Perumusan Masalah

Pemeriksaan rekaman ECG masih dilakukan secara offline di Rumah Sakit. Kondisi pandemi COVID 19 menyebabkan Rumah Sakit menjadi tempat yang beresiko penyebaran virus. Oleh karena itu, dibutuhkan sistem telekardiologi dengan menggunakan aplikasi android yang dapat digunakan pasien untuk mengambil sinyal ECG dan mengirim data sinyal ECG kepada dokter spesialis sehingga pasien dapat berkonsultasi dirumah pribadi.

## 1.3 Tujuan

Adapun tujuan dari penelitian Tugas Akhir ini adalah membuat aplikasi android untuk mengambil data ECG melalui bluetooth dan mengirimkan data ECG tersebut kepada dokter spesialis jantung.

## 1.4 Batasan masalah

Batasan masalah yang timbul dari permasalahan Tugas Akhir ini adalah:

1. Tujuan utama penelitian ini adalah membuat aplikasi yang dapat merekam data ECG, mengirim data ECG, menampilkan grafik ECG, dan *chatting*.
2. Kegiatan uji adalah menampilkan gambar sinyal pada aplikasi

android dan melakukan chatting antara pasien dengan dokter.

3. Data ECG diambil dari jantung penulis dan alat simulator.

## 1.5 Sistematika Penulisan

Laporan penelitian Tugas akhir ini tersusun dalam sistematika dan terstruktur sehingga mudah dipahami dan dipelajari oleh pembaca maupun seseorang yang ingin melanjutkan penelitian ini. Alur sistematika penulisan laporan penelitian ini yaitu:

1. BAB I Pendahuluan

Bab ini berisi uraian tentang latar belakang permasalahan, penegasan dan alasan pemilihan judul, sistematika laporan, tujuan dan metodologi penelitian.

2. BAB II Dasar Teori

Pada bab ini berisi tentang uraian secara sistematis teori-teori yang berhubungan dengan permasalahan yang dibahas pada penelitian ini. Teori-teori ini digunakan sebagai dasar dalam penelitian, yaitu informasi singkat mengenai sinyal ECG, *Mobile Programming*, aritmia, dan elektrokardiogram.

3. BAB III Perancangan Sistem dan Impementasi

Bab ini berisi tentang penjelasan-penjelasan terkait eksperimen yang akan dilakukan dan langkah-langkah data diolah hingga menghasilkan visualisasi. Guna mendukung eksperimen pada penelitian ini, digunakanlah blok diagram atau *work flow* agar sistem yang akan dibuat dapat terlihat dan mudah dibaca untuk implementasi pada pelaksaan tugas akhir.

4. BAB IV Pengujian dan Analisa

Bab ini menjelaskan tentang pengujian eksperimen yang dilakukan terhadap data dan analisanya. Beberapa teknik visualisasi akan ditunjukkan hasilnya pada bab ini dan dilakukan analisa terhadap hasil visualisasi dan informasi yang didapat dari hasil mengamati visualisasi yang tersaji.

5. BAB V Penutup

Bab ini merupakan penutup yang berisi kesimpulan yang di-

ambil dari penelitian dan pengujian yang telah dilakukan. Saran dan kritik yang membangun untuk pengembangkan lebih lanjut juga dituliskan pada bab ini.

# BAB 2

## TINJAUAN PUSTAKA

Demi mendukung penelitian ini, dibutuhkan beberapa teori penunjang sebagai bahan acuan dan referensi. Dengan demikian penelitian ini menjadi lebih terarah.

### 2.1 Dasar Teori

#### 2.1.1 Kardiologi

Kardiologi merupakan cabang ilmu kedokteran yang mana dikeluarkan mempelajari tentang gangguan jantung dan pembuluh darah. Dokter spesialis yang telah mempelajari kardiologi biasa disebut kardiolog. Akan tetapi kita harus membedakan antara kardiolog dengan ahli bedah jantung, *Cardiothoracic* dan *cardiovascular*, serta ahli bedah yang dapat melakukan bedah jantung menggunakan cara sternotomi. Apabila dibandingkan dengan kardiolog, ahli bedah jantung dapat melakukan operasi dengan cara membuka rongga dada kemudian melaksanakan bedah jantung, sedangkan kardiolog tidak melakukan operasi bedah, namun melakukan tes dan prosedur lain seperti angioplasti. Kardiologi kemudian dikembangkan lagi menjadi beberapa bagian, yaitu:

1. Ekokardiografi, merupakan metode pemeriksaan jantung yang menggunakan ultrasound (gelombang suara berfrekuensi tinggi) yang dapat menangkap gambaran struktur organ jantung. Ekokardiografi biasanya dibantu dengan teknologi Doppler yang dapat mengukur kecepatan dan arah aliran darah,
2. Elektrofisiologi Kardiak, merupakan bidang yang mempelajari sifat listrik dan kondisi penyakit jantung, serta pengobatan kelainan irama jantung (aritmia),
3. Kardiologi Intervensional, merupakan metode perawatan jantung non-bedah yang menggunakan tabung fleksibel kecil atau kateter untuk memperbaiki struktur jantung yang rusak atau tersumbat. Beberapa prosedur yang dapat dilakukan pada

- jantung dengan cara kateterisasi antara lain: Angioplasti, intervensi koroner perkutan, dan valvuloplasti,
4. Kardiologi Nuklir, merupakan metode pemeriksaan jantung menggunakan alat nuklir yang dapat memvisualkan isotop jantung dengan menggunakan radioaktif.

Dokter spesialis (kardiolog) bertugas menangani bermacam-macam gangguan terkait jantung dan sistem pembuluh darahnya. Spesifik mengenai kardiologi, jantung memiliki beberapa bagian anatomi (antara lain; atrium, ventrikel, katup) dan beberapa fitur fisiologis (antara lain; sistol, suara) yang mana telah didokumentasikan selama ratusan tahun lalu. Gangguan pada jantung dapat juga berarti mengalami sakit jantung dan sakit kardiovaskuler. Penyakit jantung dan pembuluh darah dapat mengakibatkan jumlah kematian yang cukup signifikan. Penyakit kardiovaskuler telah menyebabkan sekitar 30%. Persentase tersebut merupakan angka yang cukup tinggi, karena jantung memiliki peran yang amat penting bagi tubuh yaitu memompa darah ke seluruh tubuh. Hal tersebut berarti keadaan jantung terhubung dan juga mempengaruhi seluruh keadaan tubuh. Apabila jantung memiliki masalah atau gangguan, maka tubuh juga akan terpengaruh.

Kardiologi telah berkembang sejak ratusan tahun lalu. Beberapa catatan kuno telah menjelaskan sedikit bahasan mengenai jantung. Pada 1241, Ibnu Sina dikomentari oleh Ibn al-Naïis yang mana menyebut tentang sirkulasi darah saat berkomentar. Kemudian tahun 1628, dokter William Harvey dari Inggris adalah orang yang dapat menggambarkan secara tepat bagaimana sistem sirkulasi dan kandungan darah yang dipompa ke seluruh tubuh oleh jantung. Pada 1706, Profesor anatomi Raymond de Vieussens dari Prancis dapat menggambarkan struktur ruang serta pembuluh yang ada pada jantung. Beliau merupakan dokter pertama yang dapat menggambarkan bilik kiri jantung secara akurat. Kemudian pada 1733, Biarawan Stephen Hales, beliau merupakan orang pertama yang dapat mengukur tekanan darah. Perkembangan selanjutnya pada 1816, dokter René-Tnéophile-Hyacinthe Laennec berasal Prancis merupakan orang yang berhasil membuat stetoskop untuk mendiagnosis sejumlah infeksi pada rongga dada. Pada awal abad ke-19 berkem-

bang sedikit moderen, dokter Willem Einthoven dari Belanda pada 1903 mengembangkan elektrokardiograf (ECG atau EKG) sehingga beliau berhasil meraih penghargaan Nobel Kedokteran pada 1924. Dokter James Bryan Herrick dari Amerika, pada 1912, beliau merupakan orang yang dapat menggambarkan gejala penyakit jantung [6].

### **2.1.2 ECG atau EKG**

Elektrokardiogram atau EKG merupakan tes yang bertujuan untuk mengukur serta merekam sinyal listrik pada jantung. Sedangkan alat yang digunakan untuk mendeteksi sinyal listrik yang dihasilkan jantung dinamakan elektrokardiograf. Elektrokardiograf dapat membaca sinyal listrik dan kemudian menggambarkan menjadi grafik yang dapat ditampilkan pada layar monitor. Metode elektrokardiogram merupakan metode yang aman dan tidak menyakiti karena dilakukan tanpa memberi aliran listrik secara langsung dan tidak memberi sayatan (noninvasif), serta metode ini juga tidak memakan waktu yang lama [8].

### **2.1.3 Jenis ECG**

Berikut merupakan beberapa jenis tes ECG yang biasa dilakukan:

- 1. Cardiopulmonary exercise test (CPET)*

CPET merupakan tes yang biasa dilakukan untuk mendeteksi penyakit jantung atau paru-paru. Saat tes CPET sedang berlangsung, pasien melakukan tes sambil berolahraga ringan dengan sepeda tegak dan bernafas melalui corong. Setiap napas pasien juga akan diukur untuk dinilai seberapa baik kinerja pada tubuh. Kekuatan dan kapasitas paru-paru juga diukur dan direkam terlebih dahulu sebelum berolahraga, selama berolahraga, dan sesudah berolahraga. Secara total, lama waktu yang diperlukan untuk tes CPET adalah sekitar 40 menit. Untuk waktu berolahraga yang dihabiskan pasien kurang lebih adalah 10 menit. Tes CPET membutuhkan upaya maksimal yang dilakukan pasien agar dapat memperoleh data yang paling bagus atau benar [9].

## *2. Exercise electrocardiogram (stress test)*

Sama seperti tes CPET, tes stress juga dilakukan pasien sambil berolahraga dengan mengayuh sepeda statis atau dengan berjalan diatas treadmill. Tes Stress memiliki tujuan untuk memantau kondisi selama keadaan stres. Tes stress biasanya dilakukan setelah pasien mengalami serangan jantung, operasi jantung, dan juga saat terdeteksi mengalami penyakit arteri koroner [9].

## *3. Monitor Holter*

Holter merupakan jenis perangkat yang dapat dipakai untuk mendapat impuls listrik dari jantung yang bisa digunakan untuk memantau keadaan jantung selama 24 jam atau lebih. Elektroda ditempatkan sesuai tempat tertentu pada dada, lengan, dan kaki. Elektroda tersebut terhubung dengan mesin elektrokardiograf melalui kabel timah, sehingga aktivitas jantung dapat dideteksi, divisualisasikan dan kemudian dicetak untuk diinformasikan kepada dokter [9].

## *4. Resting 12-lead EKG*

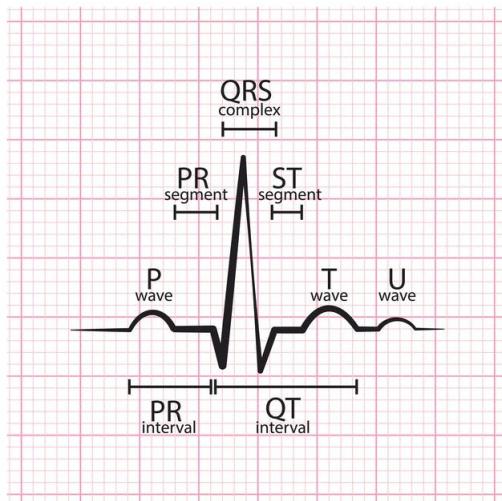
Resting 12-lead EKG merupakan tes yang ideal mengetahui sinyal listrik dari keadaan jantung pasien. Tes ini dilakukan saat keadaan rest atau tidak beraktivitas yang mana pasien menjalani tes ini sambil diam berbaring, setelah itu perangkat 12-lead EKG akan mengukur aktivitas kelistrikan jantung melalui 12 elektroda yg dipasang pada dada, lengan, dan kaki pasien secara bersamaan. Tes ini merupakan jenis yang dapat dipakai secara rutin untuk memantau kondisi jantung sebelum gejala berkembang [9].

## *5. Signal-averaged electrocardiogram (SAECG)*

Tes SAECG merupakan jenis dengan teknik elektrokardiografi khusus, di mana beberapa sinyal listrik dari jantung dirata-ratakan untuk menghilangkan interferensi dan mengungkapkan variasi kecil pada kompleks QRS, biasanya disebut "potensial akhir". Ini mungkin merupakan kecenderungan terhadap *ventricular tachyarrhythmias* yang berpotensi berbahaya. Tes

SAECG dilakukan selama kurang lebih 20 menit untuk mendeteksi adanya aritmia dalam durasi pendek [9].

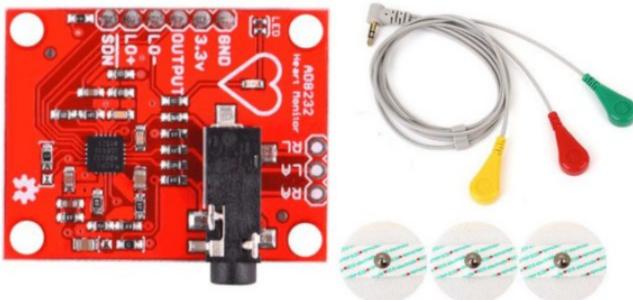
#### 2.1.4 Sinyal ECG



**Gambar 2.1:** Gelombang ECG.

Sinyal atau gelombang ECG merupakan gambaran atau visualisasi data dari aktivitas kelistrikan pada jantung yang telah diukur menggunakan sensor ECG. Gelombang ECG normal terdiri dari 3 bagian dasar. Yang pertama adalah gelombang P (sebelum lonjakan) yang mana menggambarkan aktivitas pada atrium yang memeras darah ke bawah menuju ventrikel. Kedua adalah QRS kompleks yang tampak seperti lonjakan menggambarkan mewakili dua ventrikel yang memeras darah ke tubuh dan paru-paru. Dan yang ketiga adalah gelombang T pada posisi terakhir mencerminkan pemulihan ventrikel saat menjadi rileks untuk menerima darah lagi[10].

### 2.1.5 AD8232

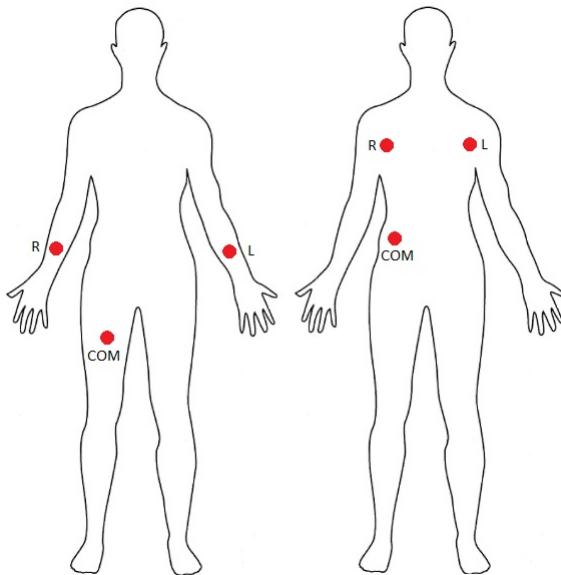


**Gambar 2.2:** Sensor AD8232.

AD8232 adalah sensor yang dapat mendeteksi sinyal terintegrasi yang bisa dipakai untuk tes pengukuran detak jantung dan aplikasi pengukuran biopotensial lainnya. AD8232 menggunakan IC pada yang terpasang pada boardnya yaitu modul SEN-12650, modul tersebut memiliki jack 3.5 female yang berfungsi untuk menancapkan kabel elektroda. AD8232 dirancang agar dapat mengekstrak, memperkuat, dan memfilter sinyal biopotensial kecil di hadapan kondisi bising, seperti suara yang dibuat oleh gerakan tubuh. Modul AD8232 memiliki sembilan pin, yaitu SDN, LO +, LO -, OUTPUT, 3.3V, GND, dan juga disediakan pin RA (Lengan Kanan), LA (Lengan Kiri), dan RL (Kaki Kanan) yang dapat dihubungkan dengan tambahan sensor AD8232 yang lain untuk membuat custom jumlah leadnya. Selain itu, modul AD8232 memiliki lampu indikator LED yang akan berkedip mengikuti irama detak jantung[11].

### 2.1.6 AD8232 *Placement*

Dengan modul AD8232, elektroda dihubungkan melalui jack audio. Output buffer dan filter tersedia melalui output analog yang dapat dibaca oleh input analog pada mikrokontroler. Berikut ini merupakan 2 macam penempatan electrode yang ditunjukkan pada Gambar 2.3.



**Gambar 2.3:** Penempatan AD8232 1 (kiri), Penempatan AD8232 2 (kanan)

Berdasarkan gambar tersebut, terdapat 2 macam penempatan electrode. Perbedaan penempatan tersebut tidak mempengaruhi data yang dihasilkan sensor sehingga pengguna bebas memilih salah satu penempatan electrode berdasarkan gambar tersebut. Elektroda mendeteksi perubahan listrik yang terjadi saat jantung melewati proses detak jantung. Elektroda mendeteksi sinyal listrik kecil yang berada dalam kisaran beberapa ratus mikrovolt ke monitor EKG yang memproses sinyal. Pemrosesan melibatkan amplifikasi dan penyaringan dan kemudian data disediakan dalam format yang dapat digunakan untuk memperbarui tampilan dari beberapa jenis [12].

### 2.1.7 Modul Bluetooth HC-05

Modul Bluetooth HC-05 merupakan konverter Bluetooth ke serial yang dapat digunakan untuk menghubungkan antara mikro-

kontroler (seperti Arduino) dengan perangkat berkemampuan Bluetooth lainnya. Pinout dan deskripsi HC-05 adalah seperti berikut:

1. **KEY/En**, Pin ini digunakan untuk membawa modul Bluetooth dalam mode perintah AT. Secara default, pin ini beroperasi dalam mode data. Pin Kunci/EN harus tinggi untuk mengoperasikan Bluetooth dalam mode perintah. Di HC-05, kecepatan baud default dalam mode perintah adalah 38400bps dan 9600 dalam mode data,
2. **VCC**, Digunakan untuk memberi daya pada modul Bluetooth. Berikan 5V / 3,3 V ke Pin ini,
3. **GND**, Pin ground dari modul,
4. **TXD**, Hubungkan pin ini dengan pin RXD Mikrokontroler. Pin ini mentransmisikan data Serial (sinyal nirkabel yang diterima oleh modul Bluetooth diubah oleh modul dan ditransmisikan secara serial pada pin ini),
5. **RXD**, Hubungkan pin ini ke pin TXD Mikrokontroler. Modul Bluetooth HC-05 menerima data dari pin ini dan kemudian mentransmisikannya secara nirkabel,
6. **State**, Ini digunakan untuk memeriksa apakah modul terhubung atau tidak. Ini bertindak sebagai indikator status.

Modul Bluetooth HC-05 dapat digunakan dalam dua mode operasi: Mode Perintah dan Mode Data. Modus Perintah, dalam Mode Perintah kita dapat berkomunikasi dengan modul Bluetooth melalui AT Commands untuk mengkonfigurasi berbagai pengaturan dan parameter Modul. Ini termasuk informasi firmware, mengubah Baud Rate, mengubah nama modul, dll. Kita juga dapat menggunakan untuk mengatur HC-05 sebagai master atau slave. Untuk memilih salah satu mode, kita perlu mengaktifkan Mode Perintah dan mengirimkan Perintah AT yang benar. Baud rate defaultnya adalah 38400bps dalam mode perintah .

Modus Data, dalam mode ini modul digunakan untuk berkomunikasi dengan perangkat Bluetooth lain, yaitu transfer data terjadi dalam mode ini. Pertukaran data antar perangkat. Baud rate defaultnya adalah 9600bps dalam mode data [13].

<b>Spesification</b>	<b>Values</b>
Microcontroller	Atmel ATmega168 for Arduino Nano 2.x, Atmel ATmega328 for Arduino Nano 3.x
Operating voltage	5 Volt
Input voltage	Optimal: 7-12 Volt Min: 6 Volt Max: 20 Volt
Digital I/O pins	14 pins (D0-D13)
Analog pins	8 pins (A0-A7)
Maximum electric current	40 mA
Clock speed	16 Mhz
SRAM	1 kbyte (ATmega168) and 2 kbyte (ATmega328)
EEPROM	512 byte (ATmega168) and 1 kbyte (ATmega328)
Flash memory	32 Mbyte for Arduino Nano 3.x 16 Mbyte for Arduino Nano 2.xz
Board size	4,5 mm x 18 mm
Weight	5 grams

**Tabel 2.1:** Spesifikasi Arduino Nano [2]

### 2.1.8 Arduino Nano

Perangkat Arduino yang digunakan pada penelitian ini yaitu Arduino Nano. Hal ini karena pada penelitian ini hanya memfokuskan untuk mengirimkan data sinyal dari arduino kepada dokter melalui aplikasi. Berdasarkan hal tersebut maka pada penelitian ini tidak membutuhkan banyak port untuk arduino. Perlu diketahui bahwa pada penelitian ini hanya membutuhkan 1 port analog untuk dihubungkan dengan pin output dari sensor AD8232. Arduino memiliki port ADC yang mana dapat mengkonversi sinyal analog menjadi digital. Pada Arduino Nano terdapat 8 port analog (A0-A7). Spesifikasi dari Arduino Nano dapat dilihat pada Tabel 2.1.

## **2.2 Penelitian Terkait**

### ***2.2.1 IoT based Real Time ECG Monitoring System using Cypress WICED***

Pada tahun 2017, Uttam U. Deshpande dan Milan A. Kulkarni mengerjakan penelitian, penelitian tersebut merancang dan menerapkan sistem pemantauan ECG berdasarkan *Cypress Wireless Internet Connectivity for Embedded Devices* (WICED) dan membandingkannya dengan Wi-Fi, Bluetooth, Zigbee dan BLE untuk membuktikan kecepatannya mengirim data yang lebih tinggi dan memiliki cakupan area yang lebih luas.[21]

### ***2.2.2 An IoT-cloud Based Wearable ECG Monitoring System for Smart Healthcare***

Pada tahun 2016, Zhe Yang, Qihao Zhou, Lei Lei, Kan Zheng, dan Wei Xiang melakukan penelitian, yang mana merancang dan menerapkan sistem pemantauan EKG berdasarkan teknik IoT cloud. Penetian ini juga menggunakan Wi-Fi, Bluetooth, Zigbee dan BLE untuk mengirim data ke cloud. Cloud IoT bertanggung jawab untuk memvisualisasikan data EKG kepada pengguna dan menyimpan data untuk analisis lebih lanjut, yang diimplementasikan atas dasar tiga server, yaitu server HTTP, Server MQTT, dan server penyimpanan.[22]

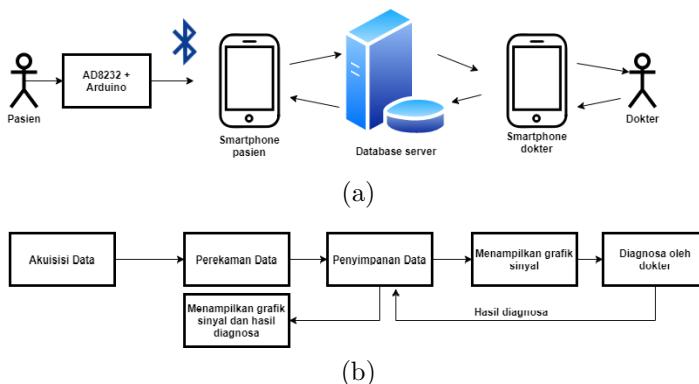
# BAB 3

## DESAIN DAN IMPLEMENTASI

Penelitian ini dilaksanakan sesuai dengan desain sistem berikut dengan implementasinya. Desain sistem merupakan konsep dari pembuatan dan perancangan infrastruktur kemudian diwujudkan dalam bentuk blok-blok alur yang harus dikerjakan. Pada bagian implementasi merupakan pelaksanaan teknis untuk setiap blok pada desain sistem.

### 3.1 Desain Sistem

Penelitian ini merupakan penerapan dari bidang studi *Mobile Programming*, *Internet of Things*, *Signal processing*, dan Basis data. Penitian ini memiliki tujuan untuk mengakuisisi data sinyal jantung menggunakan sensor AD8232 dan menyimpan data sinyal tersebut kedalam database untuk dapat ditampilkan pada aplikasi smartphone pasien dan dokter sehingga dokter dapat memantau kondisi pasien serta memberikan diagnosa, dokter dan pasien juga dapat berdiskusi melalui fitur chat yang ada pada aplikasi android. Berikut merupakan desain sistem dari penelitian ini.



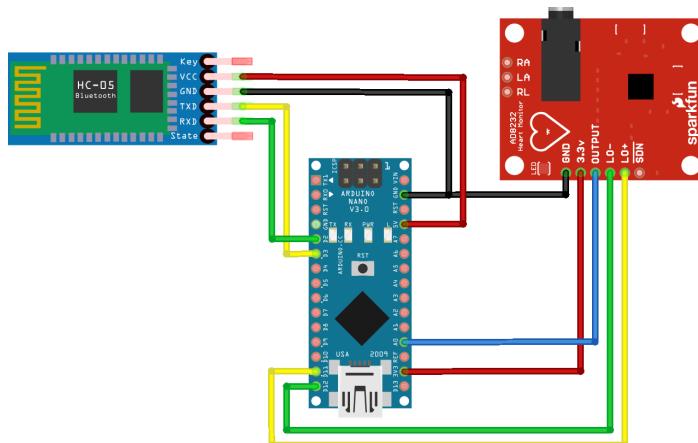
Gambar 3.1: Arsitektur (a) dan Blok Diagram (b) Alur Kerja Sistem

Gambar 3.1 merupakan arsitektur dan blok diagram alur kerja sistem. Aplikasi android menjadi dapat mengambil data sinyal jantung pasien yang telah dideteksi oleh sensor AD8232 melalui modul bluetooth HC-05. Aplikasi dapat mengirimkan data sinyal jantung pasien kepada dokter spesialis serta terdapat fitur chatting agar pasien dapat berkonsultasi dengan dokter. Penelitian ini memiliki bagian tahapan-tahapan yaitu:

1. Akuisisi data menggunakan sensor AD8232,
2. Pengiriman data ECG menuju aplikasi melalui bluetooth HC-05,
3. Menyimpan data ECG kedalam database,
4. Menampilkan grafik ECG pada aplikasi android,
5. Menambahkan fitur chat pada aplikasi android,
6. Pengujian sistem secara keseluruhan.

### 3.2 Desain Perangkat

Pada penelitian ini menggunakan 3 buah komponen yaitu Arduino Nano, modul bluetooth HC-05, dan sensor AD8232. Rangkaian ketiga komponen tersebut ditunjukkan oleh gambar berikut.

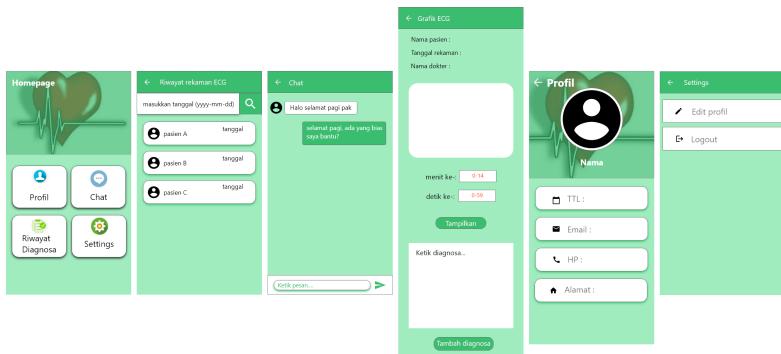


**Gambar 3.2:** Desain Perangkat ECG

Berdasarkan gambar 3.2 dapat dilihat sensor AD8232 ada pada posisi paling kanan yang memiliki board berwarna merah. Port AD82332 yang dihubungkan ke Arduino Nano adalah ground, VCC 3.3 Volt, output, LO+, dan LO-. Kemudian pada modul bluetooth HC-05 terdapat 4 port yang dihubungkan pada Arduino Nano yaitu RXD, TXD, GND, dan VCC sebesar antara 3.6 Volt sampai dengan 6 Volt.

### 3.3 Desain UI Aplikasi

Aplikasi android memiliki 2 desain yaitu untuk dokter dan pasien. Desain UI aplikasi untuk pasien memiliki fitur tambahan. Pada aplikasi pasien memiliki fitur untuk merekam sinyal ECG sedangkan pada aplikasi dokter tidak disediakan fitur tersebut. Desain aplikasi dokter ditunjukkan pada Gambar 3.3.



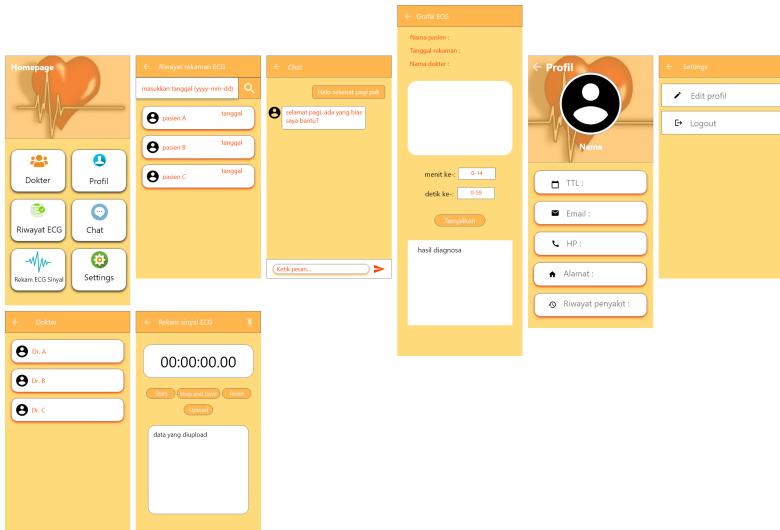
Gambar 3.3: Desain UI pada aplikasi dokter.

Dapat diketahui dari desain UI dokter tersebut terdapat 6 screen, yaitu:

1. Homepage screen, yang mana merupakan halaman utama pada aplikasi. Pada Homepage screen dokter tersedia empat buah menu yaitu Profil untuk berpindah ke halaman profil, Chat untuk berpindah ke halaman chat, Riwayat Diagnosa untuk berpindah halaman menuju riwayat diagnosa, dan Settings untuk berpindah ke halaman settings,

2. Riwayat rekaman ECG screen, yang mana merupakan halaman penelusuran yang dapat mencari riwayat ECG pasien pada waktu yang telah diinputkan. Pada screen ini terdapat field untuk menginputkan waktu rekaman dan tombol search untuk mulai mencari. Riwayat rekaman akan ditampilkan dalam bentuk list,
3. Chat screen, yang mana merupakan halaman untuk melakukan komunikasi dengan pasien melalui sebuah pesan,
4. Grafik ECG screen, yang mana merupakan halaman yang dapat menampilkan grafik data ECG hasil rekaman pasien. Pada halaman ini terdapat fitur untuk menampilkan grafik ECG pada waktu tertentu sesuai menit dan detik yang telah diinputkan. Pada halaman ini juga terdapat form isian yang merupakan tempat dokter menambahkan diagnosa untuk pasien,
5. Profil screen, yang mana merupakan halaman yang berisi tentang informasi pengguna. Informasi tersebut antara lain adalah foto, nama lengkap, tanggal lahir, email, nomor telepon, dan alamat,
6. Settings screen, yang mana merupakan halaman yang berisi menu untuk mengubah informasi pada halaman profil dan juga terdapat menu *logout* untuk keluar dari aplikasi.

Berikutnya adalah desain UI untuk pasien ditunjukkan pada Gambar 3.4.



**Gambar 3.4:** Desain UI pada aplikasi pasien.

Dapat diketahui dari desain UI pasien tersebut terdapat 8 screen, yaitu:

1. Homepage screen, yang mana merupakan halaman utama pada aplikasi. Pada Homepage screen dokter tersedia 6 buah menu yaitu Dokter untuk berpindah menuju halaman daftar dokter yang ada, Profil untuk berpindah ke halaman profil, Chat untuk berpindah ke halaman chat, Riwayat ECG untuk berpindah halaman menuju riwayat rekaman ECG, Rekam ECG Sinyal untuk berpindah menuju halaman perekaman data ECG, dan Settings untuk berpindah ke halaman settings,
2. Riwayat rekaman ECG screen yang sama seperti screen dokter, pada halaman ini pasien juga dapat mencari riwayat rekaman ECG pada waktu tertentu sesuai yang telah diinputkan,

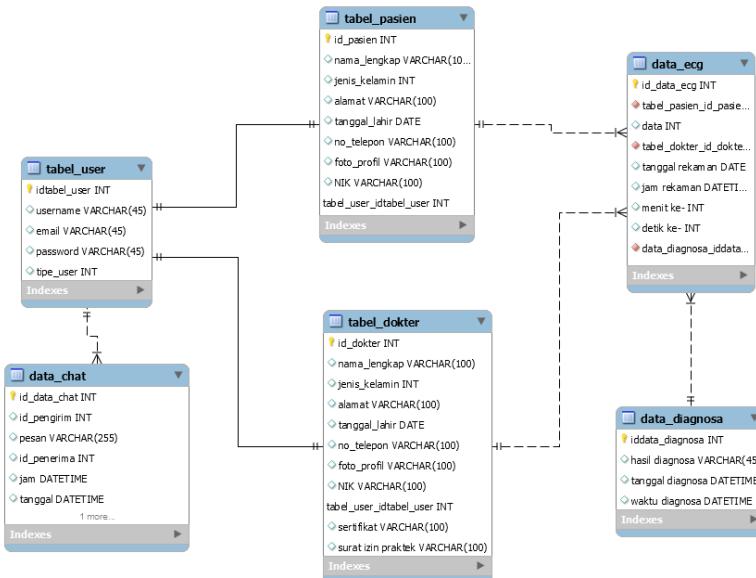
3. Chat screen, desain halaman chat untuk pasien juga sama seperti dokter,
4. Grafik ECG screen, sama seperti dokter, pasien juga dapat melihat grafik ECG miliknya yang telah direkam. Pasien juga dapat melihat grafik pada menit dan detik tertentu. Pada halaman ini pasien juga dapat melihat hasil diagnosa yang telah ditambahkan oleh dokter,
5. Profil screen, desain halaman profil untuk pasien juga sama seperti dokter. Pada halaman ini menampilkan informasi pasien yaitu foto, nama lengkap, tanggal lahir, email, nomor telepon, dan alamat,
6. Settings screen, yang mana merupakan halaman yang berisi menu untuk mengubah informasi pada halaman profil dan juga terdapat menu *logout* untuk keluar dari aplikasi.
7. Dokter screen, yang mana merupakan menu tambahan untuk pasien. Pada halaman ini pasien dapat melihat daftar dokter yang ada dan melihat informasi profil dokter.
8. Rekam sinyal ECG screen, yang mana merupakan halaman yang khusus untuk pasien. Pada halaman ini pasien dapat melakukan perekaman data ECG yang dikirim oleh arduino melalui bluetooth HC-05. Pada halaman ini terdapat icon bluetooth pada top bar yang mana berfungsi untuk menghubungkan bluetooth smartphone dengan modul HC-05. Halaman ini memiliki desain seperti stopwatch yang memiliki tombol start, stop, dan reset. Tombol start untuk memulai rekaman, stop untuk menghentikan rekaman sekaligus menyimpan data kedalam file.txt sementara, reset untuk mengembalikan ulang waktu stopwatch, dan terdapat tombol upload untuk mengupload data ke dalam database, data yang diupload dapat dilihat pada layar dibawahnya.

Pada kedua gambar tersebut terdapat sedikit perbedaan desain antara milik dokter dengan pasien. Perbedaan pertama desain tersebut terdapat pada jumlah menu di homepage. Pada desain UI homepage milik dokter memiliki menu yang lebih sedikit daripada pasien, hal ini dikarenakan dokter tidak perlu menggunakan fitur rekam sinyal sinyal dan juga fitur melihat dokter lain. Perbedaan

kedua terdapat pada layar grafik ECG yang berguna dapat menampilkan grafik sinyal jantung pada menit dan detik tertentu sesuai keinginan user. Perbedaan pada layar grafik ECG adalah pada layar dokter terdapat form dan tombol untuk menambahkan diagnosa sedangkan pada layar pasien hanya membutuhkan hasil diagnosa.

### 3.4 Desain Database

Pada penelitian ini menggunakan database MySQL. Berikut merupakan desain database pada penelitian ini ditunjukkan pada Gambar 3.5.



**Gambar 3.5:** Desain database.

Dalam database terdapat 6 tabel yang diperlukan untuk menyimpan masing-masing data, yaitu:

1. Tabel tabel\_user, memiliki 5 kolom. Tabel user digunakan sebagai tempat menyimpan data login pengguna aplikasi yaitu

username, email, password, dan tipe user untuk lebih mudah membedakan antara data pasien dan dokter. Tabel user memiliki relasi dengan 3 tabel lainnya yaitu tabel pasien, tabel dokter, dan tabel data *chat*.

2. Tabel tabel\_pasien, memiliki 9 kolom dengan 1 kolom merupakan *foreign key* dari tabel user. Tabel pasien digunakan sebagai tempat menyimpan data pasien berupa id pasien, nama lengkap, jenis kelamin, alamat, tanggal lahir, NIK, nomor telepon, foto profil , dan id user yang merupakan *foreign key* dari tabel user. Tabel pasien memiliki 2 relasi dengan 2 tabel lainnya yaitu tabel user dan tabel data ECG.
3. Tabel tabel\_dokter, memiliki 11 kolom dengan 1 kolom merupakan *foreign key* dari tabel user. Tabel dokter digunakan sebagai tempat menyimpan data dokter berupa id dokter, nama lengkap, jenis kelamin, alamat, tanggal lahir, NIK, nomor telepon, foto profil, sertifikat, surat izin praktek, dan id user yang merupakan *foreign key* dari tabel user. Tabel dokter memiliki 2 relasi dengan 2 tabel lainnya yaitu tabel user dan tabel data ECG.
4. Tabel data\_ecg, memiliki 9 kolom, 3 kolom merupakan foreign key dari tabel pasien, tabel dokter, dan tabel data\_diagnosa. Tabel data\_ecg digunakan untuk menyimpan data rekaman sinyal jantung pasien yang mana akan dapat ditampilkan pada aplikasi android. Tabel data ECG terdiri dari kolom id\_data\_ecg, id\_pasien yang merupakan *foreign key* dari tabel\_pasien, data, id\_dokter yang merupakan *foreign key* dari tabel\_dokter, tanggal rekaman, jam rekaman, jam, menit, detik, dan id\_data\_diagnosa yang merupakan *foreign key* dari tabel data diagnosa.
5. Tabel data\_diagnosa, memiliki 4 kolom, Tabel ini digunakan sebagai tempat penyimpanan hasil diagnosa yang telah ditambahkan oleh dokter. Tabel data diagnosa memiliki relasi dengan tabel data\_ecg. Tabel ini terdiri dari kolom id\_data\_diagnosa, hasil diagnosa, tanggal diagnosa, dan waktu diagnosa.
6. Dan tabel data\_chat yang memiliki 6 kolom. Tabel data\_chat diperlukan untuk menyimpan data chat antara pasien dan dokter. Tabel ini terdiri dari kolom id\_data\_chat, id\_pengirim, pesan, id\_penerima, jam, dan tanggal. Tabel data\_chat memiliki relasi dengan tabel tabel\_user.

### 3.5 Akuisisi Data

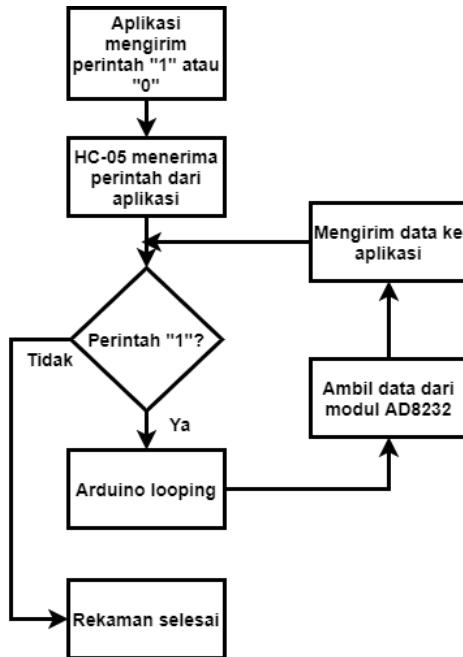
Akuisisi data pada penelitian ini menggunakan sensor ECG AD8232. Setelah sensor AD8232 membaca sinyal jantung pasien, kemudian data sinyal jantung akan diproses pada mikrokontroler arduino. Setelah data sinyal diproses oleh arduino, data sinyal akan dikirimkan melalui serial bluetooth HC-05. *Baudrate* yang digunakan pada modul bluetooth HC-05 adalah 57600. Data sinyal ECG yang dikirimkan pada aplikasi android akan disimpan sementara pada file.txt. Akuisisi data yang dilakukan pada penelitian ini adalah sebesar kurang lebih 128 Hz atau 128 data sampel sinyal tiap detik. Berikut merupakan perangkat ECG yang digunakan, ditunjukkan pada gambar 3.6.



**Gambar 3.6:** Perangkat ECG.

Mekanisme kerja pada akuisisi data adalah modul Bluetooth menunggu perintah dari aplikasi, kemudian aplikasi memberi perintah angka "1" yang mana berarti meminta alat mulai mengambil data. Disaat arduino menerima perintah "1" dari HC-05, arduino mulai melakukan looping mengambil data dari sensor AD8232. Apabila waktu rekaman sudah berjalan selama 15 menit, aplikasi akan mengirim perintah angka "0" pada HC-05 yang mana berarti untuk menghentikan rekaman. Pada saat arduino menerima perintah "0" dari HC-05, maka arduino akan berhenti melakukan looping untuk mengambil data. Berikut merupakan gambar diagram mekanisme

kerja akuisisi data Gambar 3.7.



Gambar 3.7: Diagram mekanisme akuisisi data.

### 3.6 Perekaman Data

Perekaman data dilakukan oleh aplikasi android. Proses rekaman pada pasien dalam kurang dari sampai dengan 15 menit. Berikut merupakan implementasi halaman rekaman data pada aplikasi android ditunjukkan oleh Gambar 3.8.



Gambar 3.8: Implementasi UI Rekam Sinyal ECG.

Selama proses rekaman berlangsung, aplikasi akan menerima data sinyal terus-menerus dari modul bluetooth HC-05. Aplikasi akan menyimpan data sinyal kedalam string saat proses rekaman berlangsung. Setelah rekaman selesai, aplikasi akan menyimpan data sinyal dari variabel string kedalam file.txt. Dapat dilihat pada layar rekaman (Gambar 3.8) terdapat stopwatch untuk mengetahui seberapa lama telah melakukan rekaman. Pada layar ini terdapat 4 tombol, yaitu tombol Start yang berfungsi untuk memulai rekaman, Stop and Save berfungsi untuk menghentikan rekaman dan menyimpan data dalam file.txt, tombol Reset berfungsi untuk mengembalikan stopwatch dengan keadaan waktu 00:00:00.00, tombol Upload yang berguna untuk mengupload data sinyal menuju database, dan yang terakhir adalah tombol Lihat Data yang berguna untuk menampilkan isi dari file.txt. Isi file.txt ditampilkan pada box putih seperti yang tertera dibawah tombol Lihat Data.

### **3.7 Penyimpanan Data**

Data rekaman sinyal ECG yang diupload oleh aplikasi pada database disimpan dalam tabel data\_ecg. Data ECG disimpan beserta atribut lainnya sesuai kolom yang ada pada tabel data\_ecg. Rincian kolom tersebut sebagai berikut:

1. Kolom id\_data\_ecg dengan tipe data BIGINT sebagai primary key dan auto increment untuk menyimpan nomor urut data ECG yang didapat,
2. Kolom id\_pasien dengan tipe data INT untuk menyimpan id pasien,
3. Kolom data dengan tipe data INT berguna untuk menyimpan data ECG hasil rekaman,
4. Kolom menit dengan tipe data INT untuk menyimpan waktu menit,
5. Kolom detik dengan tipe data INT untuk menyimpan waktu detik,
6. Kolom id\_dokter dengan tipe data INT untuk menyimpan id dokter,
7. Kolom tanggal dengan tipe data DATE berguna untuk menyimpan data tanggal rekaman,
8. Kolom jam dengan tipe data VARCHAR untuk menyimpan nomor sample data pada waktu mulai rekaman sampai akhir rekaman, (data ini digunakan sebagai sumbu X untuk menggambarkan grafik pada aplikasi),
9. Kolom channel dengan tipe data INT berfungsi untuk menyimpan channel lead ECG,
10. Kolom clock dengan tipe data VARCHAR untuk menyimpan waktu mulai rekaman,
11. Kolom id\_data\_diagnosa dengan tipe data INT untuk menyimpan nomor id diagnosa dari tabel data\_diagnosa,

Berikut merupakan potongan database pada tabel data\_ecg yang ditunjukkan pada Gambar 3.9.

<b>id_data_ecg</b>	<b>id_user</b>	<b>data</b>	<b>menit</b>	<b>detik</b>	<b>id_dokter</b>	<b>tanggal</b>	<b>jam</b>	<b>channel</b>	<b>clock</b>	<b>id_data_diagnosa</b>
21249	1	503	0	0	2	2021-03-02	0: 10:41	1	10:15	14
21250	1	496	0	0	2	2021-03-02	1: 10:41	1	10:15	14
21251	1	499	0	0	2	2021-03-02	2: 10:41	1	10:15	14
21252	1	4	0	0	2	2021-03-02	3: 10:41	1	10:15	14
21253	1	92	0	0	2	2021-03-02	4: 10:41	1	10:15	14
21254	1	497	0	0	2	2021-03-02	5: 10:41	1	10:15	14
21255	1	493	0	0	2	2021-03-02	6: 10:41	1	10:15	14
21256	1	4	0	0	2	2021-03-02	7: 10:41	1	10:15	14

**Gambar 3.9:** Tabel data\_ecg.

### 3.8 Menampilkan Data

Data yang telah disimpan pada database yaitu dalam tabel data\_ecg dapat diambil oleh aplikasi dokter dan pasien untuk ditampilkan dalam bentuk grafik. Pada screen grafik ECG, pengguna dapat melihat data ECG sesuai dengan waktu yang ditentukan yaitu pada (menit awal, detik awal) sampai dengan (menit akhir, detik akhir) dengan selisih waktu detik yang dapat diatur pada box (parameter range). Pengguna juga dapat melihat data selanjutnya dengan menekan tombol "selanjutnya" serta dapat kembali ke data sebelumnya dengan menakan tombol "sebelumnya". Berikut adalah tampilan dari screen Grafik sinyal ECG dokter ditunjukkan pada Gambar 3.10.



**Gambar 3.10:** Implementasi UI Grafik ECG dokter.

Pada grafik tersebut terdiri dari sumbu X dan sumbu Y. Sumbu X merupakan data ECG yang diambil dari kolom data pada tabel data\_ecg, sedangkan sumbu Y merupakan data dari kolom jam pada tabel data\_ecg (Gambar 3.9). Untuk tampilan Grafik sinyal ECG pasien tidak jauh berbeda. Berikut adalah tampilan dari screen Grafik sinyal ECG pasien ditunjukkan pada Gambar 3.11.



**Gambar 3.11:** Implementasi UI Grafik ECG pasien.

Dapat dilihat pada screen dokter (Gambar 3.10) terdapat box berwarna abu-abu dengan keterangan "Tulis diagnosa disini" yang berarti pada screen dokter dapat ditambahkan diagnosa melalui box tersebut dan kemudian diupload dengan cara menekan tombol "Upload data diagnosa". Sedang kan pada screen pasien (Gambar 3.11) hanya terdapat box putih tanpa ada tombol upload data diagnosa. Box putih tersebut adalah tempat hasil diagnosa yang

diberikan oleh dokter.

### 3.9 Diagnosa Data

Diagnosa sinyal ECG dilakukan secara manual yaitu oleh dokter. Dokter dapat menambahkan diagnosa pada rekaman ECG pasien dimenit dan detik tertentu, dapat dilihat pada Gambar 3.10 sebelumnya. Pada screen tersebut terdapat box untuk dokter yang berfungsi untuk tempat menambahkan diagnosa. Setelah dokter selesai menambahkan diagnosa, kemudian dapat mengupload ke database melalui tombol "Upload data diagnosa". Diagnosa yang telah diupload ke database akan disimpan pada tabel data\_diagnosa. Kemudian data diagnosa dapat ditampilkan pada aplikasi smarftpone pasien seperti yang ditunjukkan Gambar 3.12.



**Gambar 3.12:** Implementasi UI Riwayat Diagnosa.

### 3.10 Fitur Chat

Pada aplikasi ini tersedia fitur chat agar pasien dapat berkonsultasi dengan dokter spesialis. Untuk cara kerja pada screen chat ini, aplikasi akan melakukan refresh setiap detik dan akan mengambil data dari database yaitu pada tabel data.chat agar pesan terbaru pada screen chat dapat muncul. Saat mengirim pesan, pesan dari kedua user pasangan chat akan diupload pada tabel yang sama. Untuk membedakannya terdapat pada kolom id\_pengirim dan id\_penerima yang terdapat pada tabel data.chat (Gambar 3.9). Tampilan screen chat pasien dapat dilihat pada Gambar 3.13.



**Gambar 3.13:** Implementasi UI Chatscreen Pasien.

Sedangkan untuk tampilan screen chat dokter dapat dilihat pada Gambar 3.14.



**Gambar 3.14:** Implementasi UI Chatscreen Dokter.

Pada layar chat juga terdapat icon di pojok kanan atas yang berfungsi untuk jalan pintas. Pada Chatscreen pasien terdapat 2 icon yaitu icon sinyal ECG dan icon tiga garis. Icon sinyal ECG akan mengarahkan pasien menuju layar rekam data ECG sedangkan icon tiga garis akan mengarahkan pasien menuju layar riwayat rekaman ECG. Sedangkan pada Chatscreen dokter hanya terdapat satu icon yang akan mengarahkan dokter menuju layar riwayat rekaman ECG yang dikirimkan oleh pasien.

## BAB 4

# HASIL DAN PENGUJIAN

Pada bab ini dipaparkan hasil pengujian serta analisa dari desain sistem dan implementasi .Pengujian dilakukan guna mengetahui tingkat kesalahan dan menarik kesimpulan dari sistem yang telah dibuat.

Pada pengujian ini digunakan *Smartphone Android* dengan spesifikasi hardware seperti pada tabel 4.2.

**Tabel 4.1:** Spesifikasi *Smartphone* yang Digunakan

<b>CPU</b>	Octa-core (4x1.8 GHz Kryo 260 Gold & 4x1.6 GHz Kryo 260 Silver)
<b>Internal</b>	32GB 3GB RAM
<b>Chipset</b>	Qualcomm SDM636 Snapdragon 636 (14 nm)
<b>OS</b>	Android 8.1 (Oreo), upgradable to Android 9.0 (Pie)
<b>GPU</b>	Adreno 509
<b>Bluetooth</b>	5.0, A2DP, LE

**Tabel 4.2:** Spesifikasi *Smartphone* yang Digunakan

### 4.1 Pengujian Rekaman Data Selama 15 Menit

Pengujian ini dilakukan untuk menentukan baudrate bluetooth HC-05 yang akan digunakan dan delay yang diprogram pada arduino. Pada pengujian ini dilakukan dengan cara melakukan rekaman data dengan target durasi rekaman adalah 15 menit. Untuk menentukan delay dan baudrate yang akan digunakan maka pada pengujian ini dilakukan percobaan dengan delay dan baudrate yang berbeda-beda. Untuk delay sendiri memiliki 3 percobaan yaitu delay 1 ms, delay 5 ms, dan delay 10 ms. Sedangkan baudrate yang digunakan untuk percobaan terdapat 4 tahap yaitu 9600, 38400, 57600, dan 115200. Berikut ini merupakan tabel hasil pengujian rekaman data selama 15 (Tabel 4.3)

**Tabel 4.3:** Tabel hasil percobaan rekaman data dengan target 15 menit

Delay (ms)	Baudrate HC-05	Force <i>close</i>	Durasi dicapai	Sampling rate (sampel/detik)
10	9600	tidak	15 menit	50
5	9600	ya	2 menit 58.20 detik	68
1	9600	ya	4 menit 22.98 detik	90
10	38400	ya	3 menit 23.65 detik	72
5	38400	ya	1 menit 52.64 detik	114
1	38400	ya	2 menit 14.68 detik	216
10	57600	ya	2 menit 05.98 detik	112
5	57600	ya	2 menit 51.81 detik	125
1	57600	ya	2 menit 23.52 detik	270
10	115200	ya	3 menit 52.15 detik	117
5	115200	ya	1 menit 42.05 detik	139
1	115200	ya	1 menit 36.90 detik	327

Dapat dilihat pada tabel diatas memiliki 5 kolom yaitu Delay, Baudrate HC-05, Force close, Durasi dicapai, dan Frekuensi sampling. Delay merupakan durasi jeda waktu yang digunakan pada loop arduino. Baudrate HC-05 merupakan kecepatan pengiriman data oleh modul bluetooth HC-05. Force close merupakan kejadian dimana aplikasi menutup dengan sendirinya. Durasi dicapai merupakan lama waktu rekaman yang telah didapat sampai dengan aplikasi force close. Frekuensi sampling merupakan jumlah data

atau sampel yang didapat selama satu detik.

Pada percobaan pertama dilakukan dengan menggunakan delay 10 ms, baudrate HC-05 sebesar 9600, dan frekuensi sampling sebesar 50 data/detik. Pada percobaan pertama berjalan lancar hingga aplikasi dapat merekam data selama 15 menit. Percobaan kedua dilakukan dengan mengubah delay arduino menjadi 5 ms, frekuensi sampling yang didapat adalah 68 data/detik, pada percobaan ini aplikasi tertutup dengan sendirinya (*force close*) setelah merekam selama 2 menit 58.20 detik. Percobaan ketiga dilakukan menggunakan delay 1 ms, dengan baudrate tetap sebesar 9600, frekuensi sampling yang diperoleh sebesar 90 data/detik, pada percobaan ketiga aplikasi mengalami *force close* setelah merekam selama 4 menit 22.98 detik. Percobaan keempat menggunakan delay 10 ms, dengan baudrate HC-05 diperbesar menjadi 38400, frekuensi sampling pada percobaan ini sebesar 72 data/detik, pada percobaan keempat ini aplikasi juga mengalami *force close* setelah merekam selama 3 menit 23.65 detik. Percobaan kelima menggunakan delay sebesar 5 ms, dengan baudrate 38400, frekuensi sampling yang diperoleh sebesar 114 data/detik, dan seperti sebelumnya aplikasi mengalami *force close* setelah merekam selama 1 menit 52.64 detik. Percobaan keenam dilakukan menggunakan delay sebesar 1 ms, dengan baudrate yang sama yaitu 38400, frekuensi sampling yang diperoleh sebesar 216 data/detik, aplikasi mengalami *force close* setelah merekam selama 2 menit 14.68 detik. Pada percobaan ketujuh aplikasi juga mengalami *force close*, durasi yang dicapai selama 2 menit 05.98 detik, percobaan ini menggunakan delay arduino sebesar 10 ms, dengan baudrate yang diperbesar menjadi 57600, dan frekuensi sampling yang diperoleh sebesar 112 data/detik. Percobaan kedelapan dilakukan menggunakan delay arduino sebesar 5 ms, dengan baudrate sebesar 57600, diperoleh frekuensi sampling sebesar 125 data/detik, aplikasi mengalami *force close* setelah merekam selama 2 menit 51.81 detik. Percobaan kesembilan dilakukan dengan menggunakan delay arduino sebesar 1 ms, dengan baudrate sebesar 57600, diperoleh frekuensi sampling sebesar 270, aplikasi mengalami *force close* setelah merekam selama 2 menit 23.52 detik. Percobaan kesepuluh dilakukan dengan menggunakan delay arduino sebesar 10 ms, dengan baudrate diperbesar menjadi 115200, frekuensi sampling yang

diperoleh sebesar 117 data/detik, aplikasi mengalami *force close* setelah merekam selama 3 menit 52.15 detik. Percobaan kesebelas dilakukan menggunakan delay arduino sebesar 5 ms, dengan baudrate sebesar 115200, frekuensi sampling yang diperoleh sebesar 139 data/detik. Percobaan kedua belas yang merupakan percobaan terakhir menggunakan delay arduino sebesar 1 ms, dengan baudrate sebesar 115200, frekuensi sampling yang diperoleh sebesar 327 data/detik. Pada percobaan terakhir ini aplikasi juga mengalami *force close*, *force close* tersebut terjadi setelah merekam selama 1 menit 36.90 detik.

Berdasarkan tabel 4.3 dapat dilihat pada kolom force dimana hampir pada semua percobaan yang telah dilakukan mengalami force close pada aplikasi. Apabila dilihat pada kolom durasi dicapai, durasi waktu yang didapat saat aplikasi force close tidak pasti, tetapi sebagian besar aplikasi mengalami force close pada menit ke-2. Sedangkan apabila kolom Force close dihubungkan kolom Frekuensi sampling maka aplikasi hanya dapat berjalan dengan baik hingga menit ke-15 dengan menggunakan frekuensi sampling 50 sampel/detik, untuk frekuensi selain 50 sampel/detik tersebut aplikasi mengalami force close. Berdasarkan analisa maka dapat semakin besar frekuensi sampling yang digunakan maka semakin besar kemungkinan force close pada aplikasi.

## 4.2 Pengujian Persentase Data Hilang

Pada bagian ini, pengujian dilakukan dengan cara merekam data ECG yang dikirimkan oleh arduino. Pada pengujian ini menggunakan *baudrate* sebesar 9600. Pengujian ini dilakukan untuk mengetahui seberapa banyak data yang hilang dengan cara membandingkan data yang dikirim oleh mikrokontroler arduino dengan data yang diterima oleh aplikasi. Berikut adalah tabel yang merupakan hasil percobaan selama 1 menit (Tabel 4.4).

**Tabel 4.4:** Tabel hasil percobaan persentase data hilang

Delay (ms)	Jumlah Data dikirim	Jumlah Data diterima	Waktu uji (s)	Frekuensi sampling (sampel/detik)	Data hilang (%)
100	543	542	60	9	0.19
10	3107	3098	60	51	0.29
5	4168	4143	60	69	0.6
1	5748	5727	60	95	0.37

Tabel 4.4 merupakan tabel yang memiliki 6 kolom yaitu Delay, Jumlah Data dikirim, Jumlah Data diterima, Waktu uji, Frekuensi sampling, Data hilang. Delay merupakan durasi jeda waktu pada loop arduino. Jumlah Data dikirim merupakan banyak data yang dikirim oleh arduino menuju aplikasi. Jumlah Data diterima merupakan banyak data yang telah sampai di aplikasi. Waktu uji merupakan durasi melakukan pengujian. Frekuensi sampling merupakan banyak data yang diterima aplikasi pada setiap detik. Data hilang merupakan jumlah data yang tidak berhasil sampai pada aplikasi.

Pada pengujian ini, percobaan pertama dilakukan dengan menggunakan delay arduino sebesar 100 ms, jumlah data yang dapat dikirim sebanyak 543, jumlah data yang telah diterima pada aplikasi sebanyak 542, dengan frekuensi sampling sebesar 9 data/detik, dan diperoleh dengan cara perhitungan matematis data yang hilang sebesar 0.19%. Percobaan kedua dilakukan dengan menggunakan delay arduino sebesar 10 ms, jumlah data yang dapat dikirim sebanyak 3107, jumlah data yang telah diterima pada aplikasi sebanyak 3098, dengan frekuensi sampling sebesar 51 data/detik, dan diperoleh dengan cara perhitungan matematis data yang hilang sebesar 0.29%. Percobaan ketiga dilakukan dengan menggunakan delay arduino sebesar 5 ms, jumlah data yang dapat dikirim sebanyak 4168, jumlah data yang telah diterima pada aplikasi sebanyak 4143, dengan frekuensi sampling sebesar 69 data/detik, dan diperoleh dengan cara perhitungan matematis data yang hilang sebesar 0.6%. Percobaan kedua dilakukan dengan menggunakan delay arduino sebesar 1 ms, jumlah data yang dapat dikirim sebanyak 5748, jumlah data yang telah diterima pada aplikasi sebanyak 5727, dengan frekuensi sam-

pling sebesar 95 data/detik, dan diperoleh dengan cara perhitungan matematis data yang hilang sebesar 0.37%

Pada tabel 4.4 terdapat kolom frekuensi sampling yang mana merupakan jumlah sample yang berhasil diterima oleh aplikasi dalam 1 detik. Perhitungan frekuensi sampling dengan cara perbandingan antara jumlah data diterima dengan waktu uji. Selain itu juga terdapat kolom data hilang dimana merupakan persentase jumlah data yang tidak berhasil diterima oleh aplikasi. Perhitungan persentase data hilang adalah dengan cara membandingkan selisih jumlah data dikirim dengan jumlah data diterima dengan jumlah data dikirim kemudian dikalikan dengan 100 %.

### **4.3 Pengujian Waktu Pengiriman Data Menuju Database**

Pengujian ini dilakukan untuk mengetahui seberapa lama waktu yang dibutuhkan untuk mengupload data menuju database. Pengujian dilakukan dengan cara merekam data ECG secara bertahap untuk diupload yaitu dari 1 menit sampai 15 menit. Pada pengujian ini baudrate modul HC-05 yang digunakan adalah 9600 dan delay yang digunakan pada arduino sebesar 10 ms serta data diupload menuju server online. Dari masing-masing tahap rekaman tersebut akan dibandingkan berapa waktu yang dibutuhkan untuk upload data. Berikut merupakan tabel hasil pengujian durasi upload data (Tabel 4.5).

**Tabel 4.5:** Tabel hasil percobaan durasi upload

Durasi rekaman	Jumlah Data dikirim	Jumlah Data diterima	Durasi upload
1 menit	3004	2992	3.60 detik
2 menit	5876	5860	6.16 detik
3 menit	8927	8911	12.83 detik
4 menit	11814	11753	16.30 detik
5 menit	14699	14659	17.06 detik
6 menit	17651	17583	22.61 detik
7 menit	20510	20465	33.96 detik
8 menit	23382	23330	19.98 detik
9 menit	26330	26279	1 menit 31.64 detik
10 menit	29007	28924	34.72 detik
11 menit	31848	31784	22.69 detik
12 menit	33339	33260	19.25 detik
13 menit	35567	35513	23.07 detik
14 menit	38312	38244	28.66 detik
15 menit	41073	40973	27.01 detik

Dapat dilihat dari tabel diatas memiliki 4 kolom yaitu Durasi rekaman, Jumlah Data dikirim, Jumlah Data diterima, dan Durasi upload. Durasi rekaman merupakan jangka waktu yang dilakukan saat merekam data. Jumlah Data dikirim merupakan jumlah data yang dikirim melalui modul bluetooth HC-05 dari Arduino menuju aplikasi. Jumlah Data diterima merupakan jumlah data yang dapat disampaikan oleh modul bluetooth HC-05 dari Arduino. Dan yang terakhir adalah Durasi upload yang mana merupakan jangka waktu yang diperlukan untuk mengupload data dari aplikasi menuju database.

Pada percobaan pertama dilakukan rekaman selama 1 menit, jumlah data yang terkirim sebanyak 3004 sampel, sedangkan jumlah data yang diterima aplikasi sebanyak 2992 sampel, dan durasi upload ke database selama 3.60 detik. Percobaan kedua dilakukan rekaman selama 2 menit, jumlah data yang dikirim sebanyak

5876 sampel, sedangkan jumlah data yang diterima aplikasi sebanyak 5860 sampel, dan durasi upload ke database selama 6.16 detik. Percobaan ketiga dilakukan rekaman selama 3 menit, jumlah data yang dikirim sebanyak 8927 sampel, sedangkan jumlah data yang diterima aplikasi sebanyak 8911 sampel, dan durasi upload ke database selama 12.83 detik. Percobaan keempat dilakukan rekaman selama 4 menit, jumlah data yang dikirim sebanyak 11814 sampel, sedangkan jumlah data yang diterima aplikasi sebanyak 11753 sampel, dan durasi upload ke database selama 16.30 detik. Percobaan kelima dilakukan rekaman selama 5 menit, jumlah data yang dikirim sebanyak 14699 sampel, sedangkan jumlah data yang diterima aplikasi sebanyak 14659 sampel, dan durasi upload ke database selama 17.06 detik. Percobaan keenam dilakukan rekaman selama 6 menit, jumlah data yang dikirim sebanyak 17651 sampel, sedangkan jumlah data yang diterima aplikasi sebanyak 17583 sampel, dan durasi upload ke database selama 22.61 detik. Percobaan ketujuh dilakukan rekaman selama 7 menit, jumlah data yang dikirim sebanyak 20510 sampel, sedangkan jumlah data yang diterima aplikasi sebanyak 20465 sampel, dan durasi upload ke database selama 33.96 detik. Percobaan kedelapan dilakukan rekaman selama 8 menit, jumlah data yang dikirim sebanyak 23382 sampel, sedangkan jumlah data yang diterima aplikasi sebanyak 23330 sampel, dan durasi upload ke database selama 19.98 detik. Percobaan kesembilan dilakukan rekaman selama 9 menit, jumlah data yang dikirim sebanyak 26330 sampel, sedangkan jumlah data yang diterima aplikasi sebanyak 26279 sampel, dan durasi upload ke database paling lama dari percobaan lainnya yaitu selama 1 menit 31.64 detik. Percobaan kesepuluh dilakukan rekaman selama 10 menit, jumlah data yang dikirim sebanyak 29007 sampel, sedangkan jumlah data yang diterima aplikasi sebanyak 28924 sampel, dan durasi upload ke database selama 34.72 detik. Percobaan kesebelas dilakukan rekaman selama 11 menit, jumlah data yang dikirim sebanyak 31848 sampel, sedangkan jumlah data yang diterima aplikasi sebanyak 31784 sampel, dan durasi upload ke database selama 22.69 detik. Percobaan kedua belas dilakukan rekaman selama 12 menit, jumlah data yang dikirim sebanyak 33339 sampel, sedangkan jumlah data yang diterima aplikasi sebanyak 33260 sampel, dan durasi upload ke database selama 19.25 detik. Percobaan ketiga belas dilakukan rekaman

selama 13 menit, jumlah data yang dikirim sebanyak 35567 sampel, sedangkan jumlah data yang diterima aplikasi sebanyak 35513 sampel, dan durasi upload ke database selama 23.07 detik. Percobaan keempat belas dilakukan rekaman selama 14 menit, jumlah data yang dikirim sebanyak 38312 sampel, sedangkan jumlah data yang diterima aplikasi sebanyak 38244 sampel, dan durasi upload ke database selama 28.66 detik. Percobaan kelima belas yang merupakan percobaan terakhir dilakukan rekaman selama 15 menit, jumlah data yang dikirim sebanyak 41073 sampel, sedangkan jumlah data yang diterima aplikasi sebanyak 40973 sampel, dan durasi upload ke database selama 27.01 detik.

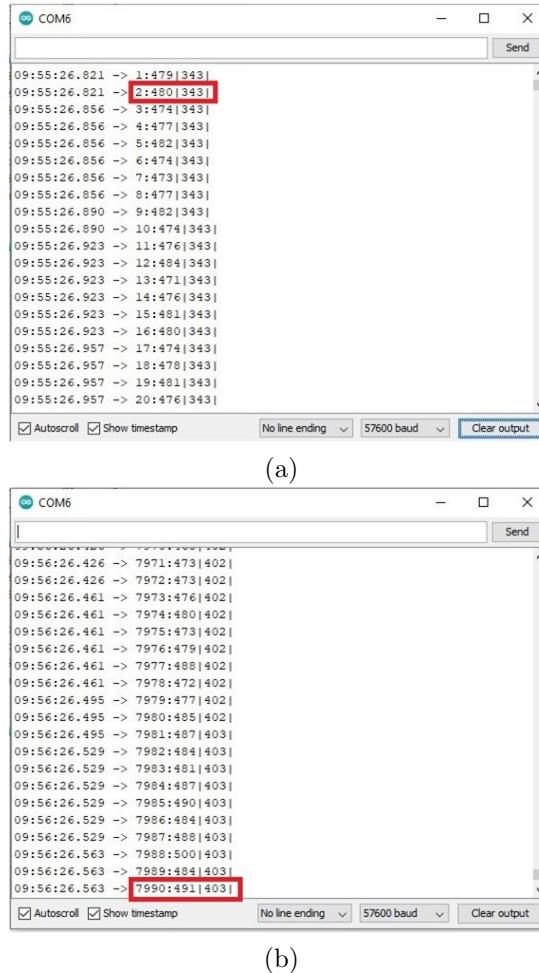
Berdasarkan tabel 4.5 dapat dilihat perbandingan durasi upload dari jumlah data yang diperoleh selama 1 menit sampai dengan 15 menit. Dari tabel tersebut dapat dilihat durasi upload yang diperlukan tidak pasti atau juga dapat dikatakan tidak linier apabila dibandingkan dengan durasi rekaman. Hal tersebut karena database ada pada server online maka tergantung jaringan internet yang digunakan. Apabila pada saat upload data memiliki jaringan internet yang bagus maka durasi upload akan berjalan lebih cepat. Sebaliknya apabila pada saat upload data memiliki jaringan internet yang kurang bagus maka proses upload akan berjalan lebih lama.

#### 4.4 Pengujian kesesuaian data dari arduino dan data di aplikasi

Pengujian ini dilakukan untuk melihat perbedaan data yang dikirim dari arduino dengan data yang telah diproses oleh aplikasi. Pengujian ini dilakukan dengan cara melakukan rekaman data selama 1 menit. Percobaan dilakukan sebanyak 5 kali percobaan, data percobaan kemudian dibandingkan

Data pada arduino ditampilkan menggunakan serial monitor dan untuk aplikasi ditampilkan pada box putih bagian bawah. Dapat dilihat pada gambar 4.1. Pada gambar tersebut dapat dilihat serial monitor telah menampilkan data yang dikirim oleh arduino terdapat 3 bagian data yaitu **sampel:data ECG|detik|** tetapi data yang dikirim hanya data **ECG|detik|** sedangkan untuk urutan sampel akan diproses pada aplikasi. Hal tersebut dikarenakan sam-

pel pertama yang dikirim dari arduino belum tentu sampai pada aplikasi, maka aplikasi perlu menentukan sampel yang pertama.



The terminal window shows the following data:

(a) Data awal yang dikirim dari arduino:

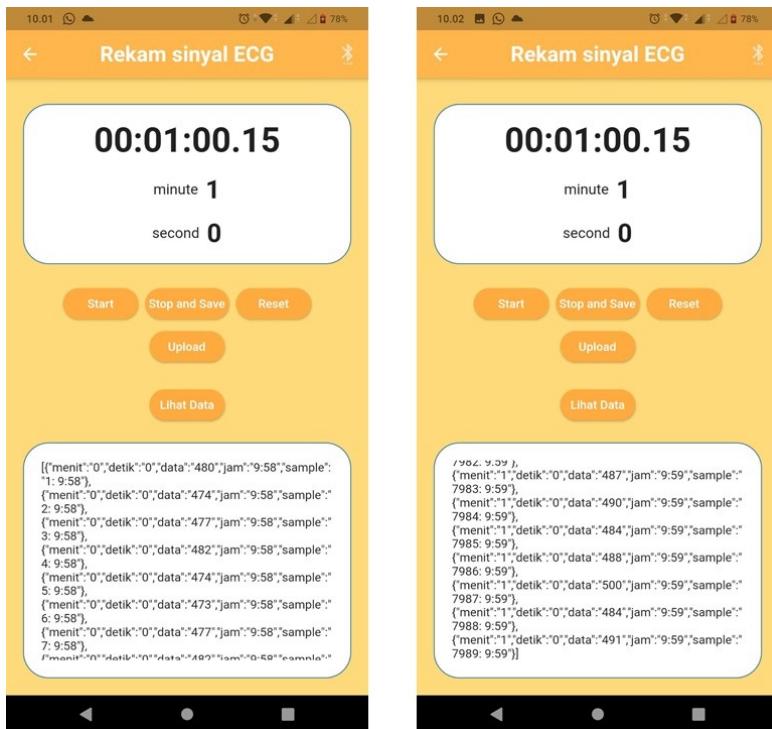
```
09:55:26.821 -> 1:479|343|
09:55:26.821 -> 2:480|343|
09:55:26.856 -> 3:474|343|
09:55:26.856 -> 4:477|343|
09:55:26.856 -> 5:482|343|
09:55:26.856 -> 6:474|343|
09:55:26.856 -> 7:473|343|
09:55:26.856 -> 8:477|343|
09:55:26.890 -> 9:482|343|
09:55:26.890 -> 10:474|343|
09:55:26.923 -> 11:476|343|
09:55:26.923 -> 12:484|343|
09:55:26.923 -> 13:471|343|
09:55:26.923 -> 14:476|343|
09:55:26.923 -> 15:481|343|
09:55:26.923 -> 16:480|343|
09:55:26.957 -> 17:474|343|
09:55:26.957 -> 18:478|343|
09:55:26.957 -> 19:481|343|
09:55:26.957 -> 20:476|343|
```

(b) Data akhir yang dikirim arduino:

```
09:56:26.426 -> 7971:473|402|
09:56:26.426 -> 7972:473|402|
09:56:26.461 -> 7973:476|402|
09:56:26.461 -> 7974:480|402|
09:56:26.461 -> 7975:473|402|
09:56:26.461 -> 7976:479|402|
09:56:26.461 -> 7977:488|402|
09:56:26.461 -> 7978:472|402|
09:56:26.495 -> 7979:477|402|
09:56:26.495 -> 7980:485|402|
09:56:26.495 -> 7981:407|403|
09:56:26.529 -> 7982:484|403|
09:56:26.529 -> 7983:481|403|
09:56:26.529 -> 7984:487|403|
09:56:26.529 -> 7985:490|403|
09:56:26.529 -> 7986:484|403|
09:56:26.529 -> 7987:488|403|
09:56:26.563 -> 7988:500|403|
09:56:26.563 -> 7989:484|403|
09:56:26.563 -> 7990:491|403|
```

**Gambar 4.1:** Percobaan 1 : (a)Data awal yang dikirim dari arduino, (b)Data akhir yang dikirim arduino.

Dapat dilihat pada Gambar 4.1, tanda "|" yang ada pada data tersebut diperlukan untuk melakukan split pada program aplikasi untuk memisahkan data ECG dengan detik. Pada tanda kotak berwarna merah berisi "2:480|343" yang mana angka "2" adalah urutan sampel dari arduino, kemudian angka "480" adalah data sensor, dan "343" merupakan detik. Pada percobaan pertama, data awal yang terkirim oleh arduino adalah 480 pada sampel ke-2 dan data terakhir yang terkirim adalah 491 pada sampel ke-7990. Apabila dihitung maka jumlah sampel yang terkirim adalah 7989 sampel dalam satu menit.



**Gambar 4.2:** Percobaan 1: (kiri) Data awal yang diterima aplikasi, (kanan) Data akhir yang diterima aplikasi.

Gambar 4.2 merupakan tampilan data yang diterima aplikasi. Dapat dilihat pada gambar tersebut, data yang ada aplikasi adalah menit, detik, data, jam, dan sampel. Menit tersebut didapatkan dari memproses detik yang telah dikirim dari arduino. Detik didapatkan dari data yang dikirim oleh arduino. Data merupakan data ECG yang dikirim oleh arduino. Jam merupakan data penjumlahan antara waktu mulai rekaman dengan detik yang diperoleh dari arduino. Sampel didapat dari proses pemotongan data pada aplikasi. Layar sebelah kiri menunjukkan data pertama yang diterima adalah 480 dan layar sebelah kanan menunjukkan data terakhir yang diterima aplikasi adalah 491. Jumlah data yang diterima oleh aplikasi dapat dilihat pada nomor "sample" yaitu 7989 sampel. Berikutnya adalah gambar percobaan kedua.

```
16:17:06.919 -> 1:344|23206|
16:17:06.919 -> 2:340|23206|
16:17:06.919 -> 3:333|23206|
16:17:06.919 -> 4:338|23206|
16:17:06.952 -> 5:347|23207|
16:17:06.952 -> 6:333|23207| 6:333|23207|
16:17:06.952 -> 7:337|23207|
16:17:06.952 -> 8:346|23207|
16:17:06.952 -> 9:335|23207|
16:17:06.952 -> 10:336|23207|
16:17:06.985 -> 11:343|23207|
16:17:06.985 -> 12:344|23207|
16:17:07.018 -> 13:336|23207|
16:17:07.018 -> 14:339|23207|
16:17:07.018 -> 15:347|23207|
16:17:07.018 -> 16:333|23207|
16:17:07.018 -> 17:338|23207|
16:17:07.052 -> 18:350|23207|
16:17:07.052 -> 19:336|23207|
16:17:07.052 -> 20:337|23207|
```

(a)

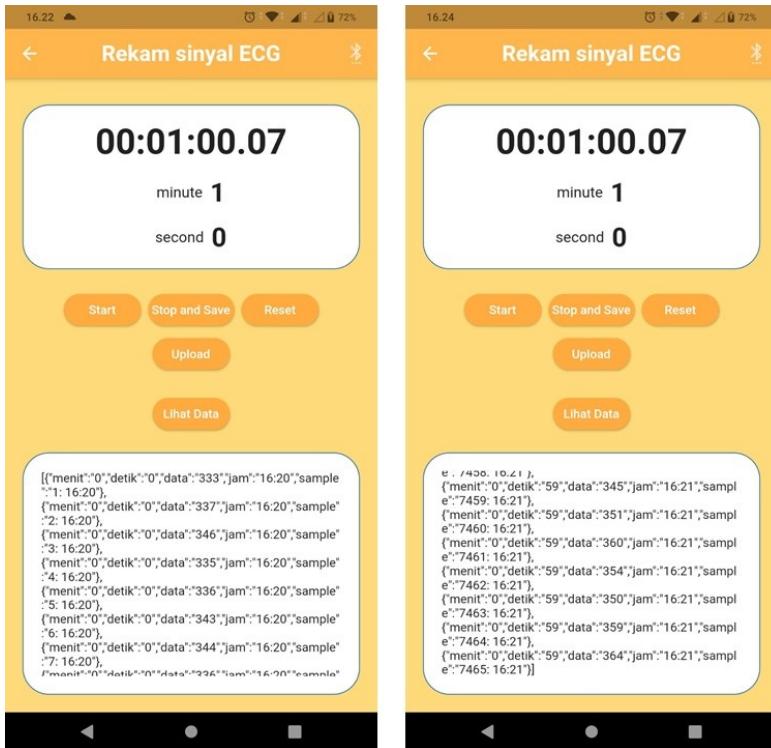
```
16:18:06.566 -> 7452:336|23266|
16:18:06.566 -> 7453:346|23266|
16:18:06.566 -> 7454:331|23266|
16:18:06.566 -> 7455:335|23266|
16:18:06.601 -> 7456:353|23266|
16:18:06.601 -> 7457:347|23266|
16:18:06.601 -> 7458:342|23266|
16:18:06.601 -> 7459:350|23266|
16:18:06.601 -> 7460:352|23266|
16:18:06.636 -> 7461:342|23266|
16:18:06.636 -> 7462:347|23266|
16:18:06.636 -> 7463:358|23266|
16:18:06.636 -> 7464:345|23266|
16:18:06.670 -> 7465:351|23266|
16:18:06.670 -> 7466:360|23266|
16:18:06.670 -> 7467:354|23266|
16:18:06.670 -> 7468:350|23266|
16:18:06.670 -> 7469:359|23266|
16:18:06.704 -> 7470:364|23266| 7470:364|23266|
16:18:06.704 -> 7471:359|23266|
```

(b)

**Gambar 4.3:** Percobaan 2 : (a)Data awal yang dikirim dari arduino,  
(b)Data akhir yang dikirim arduino.

Dapat dilihat pada Gambar 4.3, pada bagian (a) data awal yang terkirim oleh arduino adalah 333 yang merupakan sampel keenam dan pada bagian (b) data terakhir yang terkirim adalah 364

yang merupakan sampel ke-7470. Jumlah data yang dikirim oleh arduino adalah 7465 sampel.



Gambar 4.4: Percobaan 2: (kiri) Data awal yang diterima aplikasi, (kanan) Data akhir yang diterima aplikasi.

Gambar 4.4 merupakan tampilan data yang diterima aplikasi pada percobaan kedua. Layar sebelah kiri menunjukkan data pertama yang diterima adalah 333 dan layar sebelah kanan menunjukkan data terakhir yang diterima aplikasi adalah 364. Jumlah data yang diterima oleh aplikasi dapat dilihat pada nomor "sample" yaitu 7465 sampel. Berikutnya adalah gambar percobaan ketiga.

```
16:22:48.507 -> 1:341|23548|
16:22:48.507 -> 2:337|23548|
16:22:48.507 -> 3:344|23548|
16:22:48.507 -> 4:346|23548|
16:22:48.541 -> 5:340|23548|
16:22:48.541 -> 6341|23548|
16:22:48.541 -> 7:347|23548|
16:22:48.541 -> 8:336|23548|
16:22:48.541 -> 9:338|23548|
16:22:48.574 -> 10:350|23548|
16:22:48.574 -> 11:344|23548|
16:22:48.574 -> 12:340|23548|
16:22:48.608 -> 13:343|23548|
16:22:48.608 -> 14:355|23548|
16:22:48.608 -> 15:337|23548|
16:22:48.608 -> 16:340|23548|
16:22:48.608 -> 17:351|23548|
16:22:48.642 -> 18:340|23548|
16:22:48.642 -> 19:339|23548|
16:22:48.642 -> 20:351|23548|
```

Autoscroll  Show timestamp   No line ending

(a)

```
16:23:48.632 -> 7512:353|23608|
16:23:48.632 -> 7513:340|23608|
16:23:48.632 -> 7514:336|23608|
16:23:48.666 -> 7515:346|23608|
16:23:48.666 -> 7516:345|23608|
16:23:48.666 -> 7517:337|23608|
16:23:48.666 -> 7518:344|23608|
16:23:48.666 -> 7519:350|23608|
16:23:48.699 -> 7520:341|23608|
16:23:48.699 -> 7521:340|23608|
16:23:48.699 -> 7522:355|23608|
16:23:48.734 -> 7523:342|23608|
16:23:48.734 -> 7524:341|23608|
16:23:48.734 -> 7525:348|23608|
16:23:48.734 -> 7526:346|23608|
16:23:48.734 -> 7527:339|23608|
16:23:48.768 -> 7528:345|23608|
16:23:48.768 -> 7529:351|23608|
16:23:48.768 -> 7530:339|23608|
16:23:48.803 -> 7531:344|23608|
```

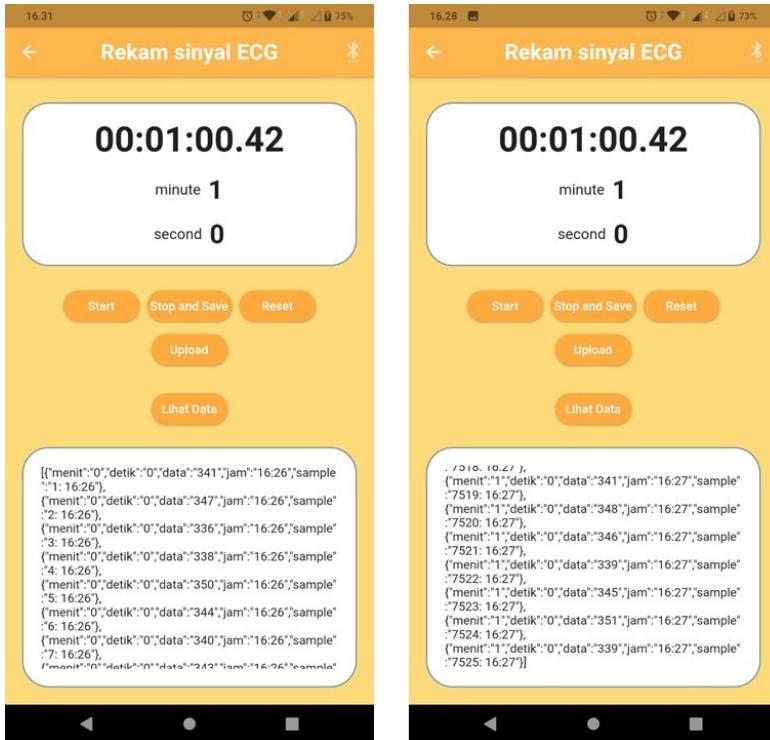
Autoscroll  Show timestamp   No line ending

(b)

**Gambar 4.5:** Percobaan 3 : (a)Data awal yang dikirim dari arduino, (b)Data akhir yang dikirim arduino.

Dapat dilihat pada Gambar 4.5, pada bagian (a) data awal yang terkirim oleh arduino adalah 341 yang merupakan sampel keenam dan pada bagian (b) data terakhir yang terkirim adalah 339

yang merupakan sampel ke-7470. Jumlah data yang dikirim oleh arduino adalah 7525 sampel.



**Gambar 4.6:** Percobaan 3: (kiri) Data awal yang diterima aplikasi, (kanan) Data akhir yang diterima aplikasi.

Gambar 4.6 merupakan tampilan data yang diterima aplikasi. Layar sebelah kiri menunjukkan data pertama yang diterima adalah 341 dan layar sebelah kanan menunjukkan data terakhir yang diterima aplikasi adalah 339. Jumlah data yang diterima oleh aplikasi dapat dilihat pada nomor "sample" yaitu 7525 sampel. Berikutnya adalah gambar percobaan keempat.

```
16:28:50.403 -> 1:346|23909|
16:28:50.403 -> 2:351|23909|
16:28:50.437 -> 3:346|23909|
16:28:50.437 -> 4:369|23909|
16:28:50.437 -> 5:400|23909|
16:28:50.437 -> 6:410|23909|
16:28:50.437 -> 7:437|23909|
16:28:50.437 -> 8:454|23909|
16:28:50.471 -> 9:417|23909|
16:28:50.471 -> 10:376|23909|
16:28:50.505 -> 11:306|23909|
16:28:50.505 -> 12:319|23909|
16:28:50.505 -> 13:338|23909|
16:28:50.505 -> 14:340|23909|
16:28:50.539 -> 15:333|23909|
16:28:50.539 -> 16:342|23909|
16:28:50.539 -> 17:346|23909|
16:28:50.539 -> 18:330|23909|
16:28:50.539 -> 19:340|23909|
16:28:50.539 -> 20:354|23909|
```

(a)

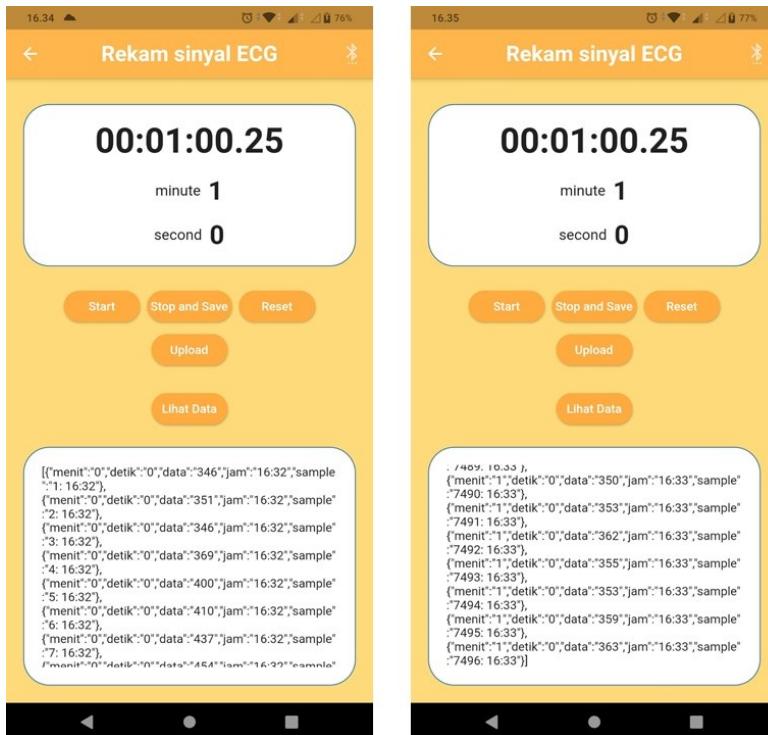
```
16:29:50.266 -> 7479:353|23969|
16:29:50.300 -> 7480:340|23969|
16:29:50.300 -> 7481:340|23969|
16:29:50.300 -> 7482:357|23969|
16:29:50.333 -> 7483:348|23969|
16:29:50.333 -> 7484:340|23969|
16:29:50.333 -> 7485:346|23969|
16:29:50.333 -> 7486:352|23969|
16:29:50.333 -> 7487:340|23969|
16:29:50.367 -> 7488:346|23969|
16:29:50.367 -> 7489:361|23969|
16:29:50.367 -> 7490:350|23969|
16:29:50.401 -> 7491:353|23969|
16:29:50.401 -> 7492:362|23969|
16:29:50.401 -> 7493:355|23969|
16:29:50.401 -> 7494:353|23969|
16:29:50.401 -> 7495:359|23969|
16:29:50.436 -> 7496:363|23969|
16:29:50.436 -> 7497:349|23969|
16:29:50.436 -> 7498:353|23969|
```

(b)

**Gambar 4.7:** Percobaan 4 : (a)Data awal yang dikirim dari arduino,  
(b)Data akhir yang dikirim arduino.

Pada gambar 4.7 dapat diketahui bagian (a) data awal yang terkirim adalah 346 yang merupakan sampel pertama pada arduino, kemudian data terakhir ditampilkan pada bagian (b) yaitu 363

merupakan data urutan ke-7496 serta merupakan jumlah data yang berhasil dikirim.



**Gambar 4.8:** Percobaan 4: (kiri) Data awal yang diterima aplikasi, (kanan) Data akhir yang diterima aplikasi.

Pada percobaan keempat seperti yang diperlihatkan Gambar 4.8, pada bagian kiri menunjukkan data awal yang diterima aplikasi yaitu 346 dan pada bagian kanan terdapat data terakhir yang diterima aplikasi yaitu 363. Jumlah data yang diterima aplikasi ditunjukkan pada nomor sampel data terakhir yaitu 7469. Kemudian berikut ini merupakan gambar percobaan terakhir.

```
16:33:31.423 -> 1:336|24189|
16:33:31.423 -> 2:336|24189|
16:33:31.423 -> 3:351|24189|
16:33:31.423 -> 4:347|24189|
16:33:31.456 -> 5:337|24189|
16:33:31.456 -> 61348|24189|
16:33:31.456 -> 7:350|24189|
16:33:31.456 -> 8:339|24189|
16:33:31.456 -> 9:344|24189|
16:33:31.490 -> 10:354|24189|
16:33:31.490 -> 11:339|24189|
16:33:31.490 -> 12:339|24189|
16:33:31.523 -> 13:352|24189|
16:33:31.523 -> 14:349|24189|
16:33:31.523 -> 15:339|24189|
16:33:31.523 -> 16:343|24189|
16:33:31.523 -> 17:354|24190|
16:33:31.557 -> 18:336|24190|
16:33:31.557 -> 19:343|24190|
16:33:31.557 -> 20:356|24190|
```

(a)

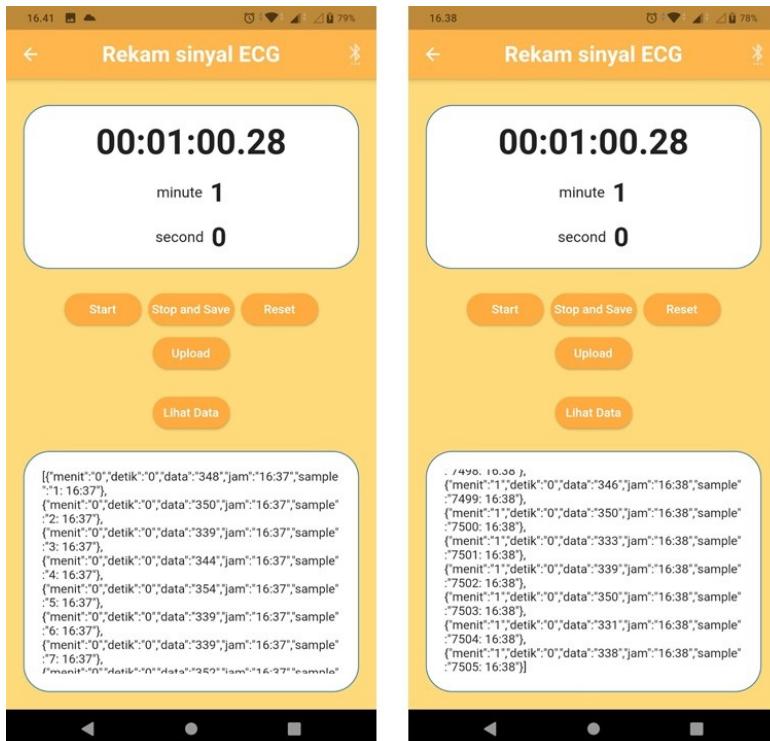
```
16:34:31.393 -> 7492:340|24249|
16:34:31.393 -> 7493:335|24249|
16:34:31.393 -> 7494:343|24249|
16:34:31.428 -> 7495:348|24249|
16:34:31.428 -> 7496:329|24249|
16:34:31.428 -> 7497:338|24249|
16:34:31.428 -> 7498:350|24249|
16:34:31.428 -> 7499:330|24249|
16:34:31.428 -> 7500:337|24249|
16:34:31.462 -> 7501:349|24249|
16:34:31.462 -> 7502:345|24249|
16:34:31.462 -> 7503:334|24249|
16:34:31.499 -> 7504:346|24249|
16:34:31.499 -> 7505:350|24249|
16:34:31.499 -> 7506:333|24249|
16:34:31.499 -> 7507:339|24249|
16:34:31.499 -> 7508:350|24249|
16:34:31.499 -> 7509:331|24249|
16:34:31.532 -> 7510:338|24249|
16:34:31.532 -> 7511:349|24249|
```

(b)

**Gambar 4.9:** Percobaan 5 : (a)Data awal yang dikirim dari arduino, (b)Data akhir yang dikirim arduino.

Dapat diketahui dari gambar 4.9, bagian (a) menunjukkan data awal yang terkirim yaitu 348 pada urutan sampel nomor 6, kemudian bagian (b) menunjukkan data terakhir yang terkirim yaitu 338

pada nomor sampel 7510. Apabila diperhitungkan melihat nomor sampel maka jumlah data yang terkirim adalah 7505 sampel.



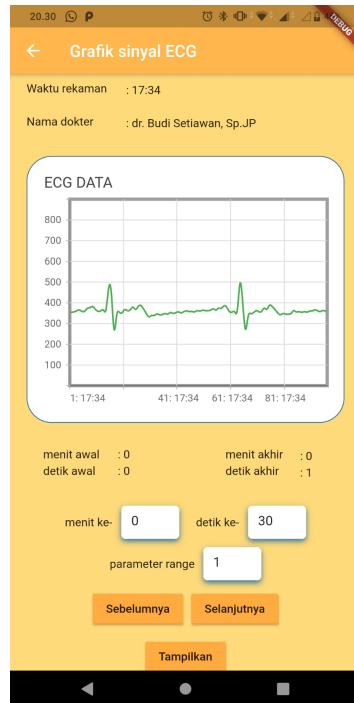
**Gambar 4.10:** Percobaan 5: (kiri) Data awal yang diterima aplikasi, (kanan) Data akhir yang diterima aplikasi.

Pada percobaan terakhir didapatkan data seperti yang diperlihatkan Gambar 4.10, pada bagian kiri menunjukkan data awal yang diterima aplikasi yaitu 348 dan pada bagian kanan terdapat data terakhir yang diterima aplikasi yaitu 338. Jumlah data yang diterima aplikasi ditunjukkan pada nomor sampel data terakhir yaitu 7505 sehingga jumlah data yang dikirim arduino sesuai dengan jumlah data yang diterima aplikasi. Untuk lebih jelasnya dapat dilihat pada tabel berikut.

Uji ke-	Arduino			Aplikasi		
	Data awal	Data akhir	Jumlah sampel	Data awal	Data akhir	Jumlah sampel

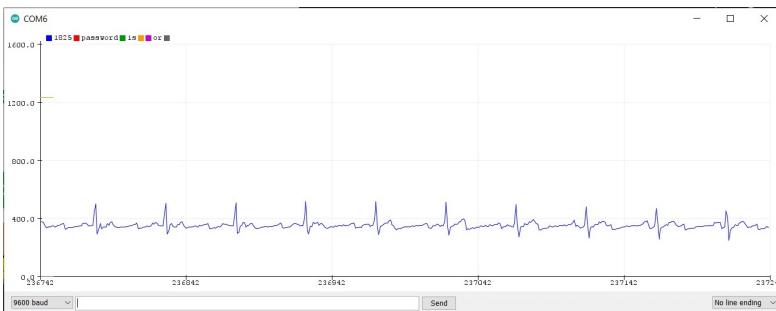
## 4.5 Pengujian Kesesuian Grafik ECG

Pada tahap ini, dilakukan pengujian dengan cara membandingkan grafik ECG yang telah ditampilkan oleh aplikasi android dengan grafik ECG yang ditampilkan oleh serial ploter yang merupakan tools dari Arduino IDE. Berikut adalah hasil grafik ECG yang ditampilkan oleh aplikasi Gambar 4.11.



**Gambar 4.11:** Grafik ECG yang ditampilkan pada aplikasi.

Sedangkan grafik ECG yang ditampilkan pada serial ploter Arduino IDE ditunjukkan pada gambar berikut.



**Gambar 4.12:** Grafik ECG yang ditampilkan pada serial ploter Arduino IDE.

Berdasarkan kedua gambar diatas, apabila dibandingkan maka kedua gambar tersebut cukup mirip. Dengan demikian berarti hasil grafik ECG yang ditampilkan oleh aplikasi android cukup sesuai dengan grafik ECG yang ditampilkan oleh serial ploter Arduino IDE.

*Halaman ini sengaja dikosongkan*

## BAB 5

# PENUTUP

### 5.1 Kesimpulan

Pada penelitian ini, dilakukan pengambilan data ECG dengan menggunakan sensor AD8232 dan kemudian dikirimkan menuju aplikasi android melalui modul bluetooth HC-05. Pengambilan data tersebut dilakukan selama 15 menit. Setelah semua data diterima oleh aplikasi, data kemudian akan diupload menuju database. Data yang telah diupload di database dapat diakses oleh Dokter Spesialis agar dapat diberi diagnosa melalui aplikasi android. Aplikasi android juga memiliki fitur chat sehingga pasien dapat berkonsultasi dengan dokter.

Dari hasil pengujian yang sudah dilakukan pada bab sebelumnya, dapat ditarik beberapa kesimpulan sebagai berikut:

1. Aplikasi sudah dapat merekam data ECG dari arduino melalui modul bluetooth HC-05, menampilkan garfik ECG, chatting.
2. Hasil pengujian untuk menghitung data yang hilang menunjukkan bahwa data yang hilang setiap sampling adalah kurang dari 1%.
3. Dari hasil pengujian kesesuaian grafik ECG menunjukkan bahwa hasil grafik ECG yang ditampilkan oleh aplikasi sudah cukup sesuai.
4. Aplikasi dapat merekam data selama 15 menit dengan frekuensi sampling 50 sampel per detik yaitu menggunakan baudrate 9600 pada HC-05 dan delay 10 ms pada arduino
5. Semakin besar frekuensi sampling maka semakin besar kemungkinan force close pada aplikasi
6. Durasi upload data menuju database tergantung pada kondisi jaringan internet

### 5.2 Saran

Untuk pengembangan penelitian selanjutnya terdapat beberapa saran sebagai berikut :

1. Mengirim data ECG menuju aplikasi selain menggunakan bluetooth,
2. Menambah fitur aplikasi untuk dapat mendeteksi adanya aritmia dari data ECG hasil rekaman.

## DAFTAR PUSTAKA

- [1] Wikipedia, “Arduino,” 2021. <https://id.wikipedia.org/wiki/Arduino>, Last accessed on December 2019. No citations.
- [2] A. wordpress, “Arduino nano,” 2015. <https://djukarna4arduino.wordpress.com/2015/01/19/arduino-nano/>, Last accesed on Juny 2021. (Dikutip pada halaman xiii, 13).
- [3] docdoc, “Apa itu kardiologi: Gambaran umum,” 2020. <https://www.docdoc.com/id/info/specialty/kardiologi>, Last accessed on December 2019. (Dikutip pada halaman 1).
- [4] World Health Organisationn, “The top 10 causes of death,” 2018. <https://www.who.int/news-room/fact-sheets/detail/the-top-10-causes-of-death>, Last accessed on December 2019. (Dikutip pada halaman 1).
- [5] KEMENTERIAN KESEHATAN REPUBLIK INDONESIA, “Indonesia dalam risiko penyakit kardiovaskular,” 2018. <http://p2ptm.kemkes.go.id/artikel-sehat/indonesia-dalam-risiko-penyakit-kardiovaskular>, Last accessed on December 2019. (Dikutip pada halaman 1).
- [6] F. K. U. Airlangga, “Profil kardiologi dan kedokteran vaskular,” 2013. <https://fk.unair.ac.id/archives/2013/10/24/profil-kardiologi-dan-kedokteran-vaskular.html>, Last accesed on Juny 2021. (Dikutip pada halaman 7).
- [7] medicalnewstoday, “What to know about arrhythmia,” 2020. <https://www.medicalnewstoday.com/articles/8887>, Last accesed on Juny 2021. No citations.

- [8] halodoc, “Elektrokardiogram (ekg),” 2021. <https://www.halodoc.com/kesehatan/elektrokardiogram-ekg>, Last accesed on Juny 2021. (Dikutip pada halaman 7).
- [9] hellosehat, “Elektrokardiogram/elektrokardiografi (ekg),” 2019. <https://hellosehat.com/jantung/aritmia/ekg-adalah-elektrokardiografi/>, Last accesed on Juny 2021. (Dikutip pada halaman 7, 8, 9).
- [10] T. Conversation, “Your apple watch can now record your ecg – but what does that mean and can you trust it?,” 2018. <https://theconversation.com/your-apple-watch-can-now-record-your-ecg-but-what-does-it>, Last accesed on December 2020. (Dikutip pada halaman 9).
- [11] H. to Electronocs, “Pemantauan grafik ekg dengan sensor ekg ad8232 & arduino,” 2019. <https://how2electronics.com/ecg-monitoring-with-ad8232-ecg-sensor-arduino/>, Last accessed on December 2019. (Dikutip pada halaman 10).
- [12] P. Supplies, “Ad8232 heart rate monitor module with electrodes,” 2021. <https://protosupplies.com/product/ad8232-heart-rate-monitor-module-with-electrodes/>, Last accesed on Juny 2021. (Dikutip pada halaman 11).
- [13] eTechnophiles, “Hc-05 pinout, specifications, datasheet and hc05 arduino connection,” <https://www.technophiles.com/hc-05-pinout-specifications-datasheet/>, Last accesed on Juny 2021. (Dikutip pada halaman 12).
- [14] techopedia, “Mobile application (mobile app),” 2020. <https://www.techopedia.com/definition/2953/mobile-application-mobile-app>, Last accesed on January. No citations.
- [15] wikipedia, “Adobe xd,” 2021. [https://en.wikipedia.org/wiki/Adobe\\_XD](https://en.wikipedia.org/wiki/Adobe_XD), Last accesed on Juny 2021. No citations.

- [16] ARDUINO, “What is arduino?,” 2018. <https://www.arduino.cc/en/Guide/Introduction>, Last accesed on Juny 2021. No citations.
- [17] wikipedia, “Visual studio code,” 2021. [https://en.wikipedia.org/wiki/Visual\\_Studio\\_Code](https://en.wikipedia.org/wiki/Visual_Studio_Code), Last acce-  
sed on Juny 2021. No citations.
- [18] dicoding, “Kenapa flutter layak kamu pelajari: 3  
alasan,” <https://www.dicoding.com/blog/kenapa-flutter/>, Last accesed on December 2020.  
No citations.
- [19] Qwords, “Pengertian xampp lengkap dengan fungsi dan  
cara instalasi,” 2019. <https://qwords.com/blog/pengertian-xampp/>, Last accesed on December 2020. No  
citations.
- [20] POSTMAN, “About postman,” <https://www.postman.com/company/about-postman/>, Last accesed on Juny 2021. No citations.
- [21] M. A. K. Uttam U. Deshpande, “Iot based real time  
ecg monitoring system using cypress wiced,” International  
Journal of Advanced Research in Electrical, Electronics and  
Instrumentation Engineering, vol. 6, pp. 710–720, 2017. (Dikutip pada halaman 14).
- [22] L. L. K. Z. W. X. Zhe Yang, Qihao Zhou1, “An iot-cloud based  
wearable ecg monitoring system for smart healthcare,” Journal  
of Medical Systems, vol. 40, 2016. (Dikutip pada halaman 14).

*Halaman ini sengaja dikosongkan*

## BIOGRAFI PENULIS



Robby Aldriyanto Raffly, atau biasa dipanggil Robby, lahir di Nganjuk Jawa Timur pada tanggal 12 Juli 1998. Merupakan anak tunggal. Penulis lulus dari SMP Negeri 2 Nganjuk dan melanjutkan ke SMA Negeri 2 Nganjuk. Penulis melanjutkan ke jenjang strata satu di Departemen Teknik Komputer Fakultas Teknologi Elektro dan Informatika Cerdas ITS. Dalam masa kuliah, penulis tertarik dengan pengembangan *Mobile Apps* dan *Machine Learning*. Penulis pernah aktif menjadi Staf Departemen Muamalah KALAM ELITS serta menjadi Koordinator Keamanan dan Perizinan MAGE 4. Bagi pembaca yang memiliki kritik, saran, atau pertanyaan mengenai tugas akhir ini dapat menghubungi penulis melalui email [robbyaraffly12@gmail.com](mailto:robbyaraffly12@gmail.com).

*Halaman ini sengaja dikosongkan*