



**Sistem Monitoring Telemedis Untuk
Pemantauan Detak Jantung Berbasis *Internet Of
Things (IoT)* Menggunakan MAX30102 Dan
Jaringan *LoRaWAN***

Proyek Akhir

**Oleh:
Rudcayanti Hutagaol (3232201022)**

**Program Studi Teknik Instrumentasi
Jurusan Teknik Elektro
Politeknik Negeri Batam
2024**

Pernyataan Keaslian Proyek Akhir

Saya yang bertandatangan dibawah ini menyatakan bahwa isi sebagian maupun keseluruhan Proyek Akhir saya yang berjudul : "Sistem Monitoring Telemedis Untuk Pemantauan Detak Jantung Berbasis *Internet Of Things (IoT)* Menggunakan MAX30102 Dan Jaringan *LoRaWAN*" adalah hasil **karya sendiri**, **diselesaikan tanpa menggunakan bahan-bahan yang tidak diizinkan**, dan **bukan merupakan karya pihak lain yang saya akui sebagai karya sendiri**. Semua referensi yang dikutip atau dirujuk telah ditulis secara lengkap pada daftar pustaka. Apabila ternyata pernyataan saya ini tidak benar, saya bersedia menerima sanksi sesuai peraturan yang berlaku.

Batam, 04 Juli 2024

A 10,000 Rupiah Indonesian postage stamp featuring the Garuda Pancasila emblem and the text "METER TEMPEL" and "DEANK015218929". A handwritten signature is written over the stamp.

Rudcayanti Hutagaol
NIM: 3232201022

Lembar Pengesahan

Proyek Akhir disusun untuk memenuhi salah satu syarat memperoleh gelar
Ahli Madya Teknik (A.Md.T.)
di
Politeknik Negeri Batam

Oleh:

Rudcayanti Hutagaol (3232201022)

Tanggal Sidang: 04 Juli 2024

Disetujui oleh :

Penguji 1



Ir. Kamarudin S.T.,M.T.,IPM.
NIK: 110071

Pembimbing 1



Mu'thiana Gusnam, S.Kom, M.T
NIK: 123293

Penguji 2



Muhammad Jaka Wimbang
Wicaksono, S.T., M.T.
NIK: 122272



Dessy Oktani S.T., M.T
NIK: 11075

[Pemantauan Detak Jantung Berbasis *Internet Of Things (IoT)* Menggunakan MAX30102 Dan Jaringan *LoRaWAN*]

Abstrak

Pengembangan perangkat pemantauan detak jantung dan tekanan darah berbasis *Internet of Things (IoT)* dengan teknologi *Long Range (LoRa)* sangat penting dalam menangani penyakit kardiovaskular, penyebab utama kematian global. Alat ini memungkinkan pemantauan detak jantung dan tekanan darah secara mandiri oleh pasien, serta pemantauan *real-time* oleh tenaga medis dari jarak jauh melalui aplikasi Android dan *realtime firebase*. Tujuan penelitian ini adalah menciptakan perangkat yang efisien untuk mendeteksi dan memantau kondisi jantung, menggunakan sensor MAX30102 dan MPX5700DP yang terhubung ke jaringan *LoRaWAN*. Metode penelitian meliputi studi literatur, desain mekanikal dan elektrik, pengembangan program komunikasi dan sensor, serta pengujian untuk kalibrasi sensor dan pengumpulan data. Hasil menunjukkan bahwa perangkat ini dapat mentransmisikan data detak jantung dengan akurasi 97,7% dengan presentase error sebesar 2,3% dan tekanan darah sistolik dengan akurasi 97,5% dengan presentase error 2,5%, tekanan darah diastolik dengan akurasi 96,8% dengan presentase error 3,2% yang memungkinkan pemantauan kesehatan yang lebih responsif dan efektif. Kesimpulannya, perangkat pemantauan detak jantung dan tekanan darah berbasis *IoT* dengan teknologi *LoRa* ini berpotensi meningkatkan deteksi dini dan pengobatan kondisi kardiovaskular, serta memberikan akses yang lebih baik bagi pasien dan tenaga medis.

Kata kunci: detak jantung, tekanan darah, *IoT*, *LoRa*, pemantauan kesehatan.

[Telemedicine Monitoring System For Internet Of Things (IoT) Based Heart Rate Monitoring Using MAX30102 And LoRaWAN Network]

Abstract

The development of Internet of Things (IoT)-based heart rate and blood pressure monitoring devices with Long Range (LoRa) technology is very important in dealing with cardiovascular disease, the leading cause of global death. This tool allows independent monitoring of heart rate and blood pressure by patients, as well as real-time monitoring by medical personnel remotely via the Android application and realtime Firebase. The aim of this research is to create an efficient device for detecting and monitoring heart conditions, using the MAX30102 and MPX5700DP sensors connected to the LoRaWAN network. Research methods include literature study, mechanical and electrical design, communication and sensor program development, as well as testing for sensor calibration and data collection. The results show that this device can transmit heart rate data with an accuracy of 97.7% with an error percentage of 2.3% and systolic blood pressure with an accuracy of 97.5% with an error percentage of 2.5%, diastolic blood pressure with an accuracy of 96.8%. % with an error percentage of 3.2% which allows more responsive and effective health monitoring. In conclusion, this IoT-based heart rate and blood pressure monitoring device with LoRa technology has the potential to improve early detection and treatment of cardiovascular conditions, as well as provide better access for patients and medical personnel.

Keywords: heart rate, blood pressure, IoT, LoRa, health monitoring.

Kata Pengantar

Puji syukur atas kehadiran Tuhan Yang Maha Esa, karena atas berkat dan rahmat-Nya penulis dapat menyelesaikan proyek akhir yang berjudul “Sistem Monitoring Telemedis Untuk Pengukuran Tekanan Darah Berbasis *Internet Of Things (IoT)* Menggunakan MPX5700DP Dan Jaringan *LoRaWAN*”. Penulisan laporan ini dilakukan dalam rangka memenuhi salah satu syarat untuk memperoleh gelar Ahli Madya Teknik (Amd. T) dari jenjang Diploma 3, Program Studi Teknik Instrumentasi, Jurusan Teknik Elektro di Politeknik Negeri Batam. Saya menyadari bahwa, tanpa bantuan dan bimbingan dari berbagai pihak pada masa perkuliahan saya sampai pada penyusunan ini, sangat sulit bagi saya untuk dapat menyelesaikannya. Oleh karena itu, saya mengucapkan terima kasih kepada:

1. Ibu Mu’thiana Gusnam, S.Kom,M.T dan Ibu Dessy Oktani S.T.,M.T selaku dosen pembimbing yang telah membimbing, serta memberikan masukan, saran dan juga arahan hingga akhir.
2. Seluruh dosen dan laboran pada Program Studi Teknik Instrumentasi Politeknik Negeri Batam.
3. Orang tua dan keluarga tercinta yang selalu memberikan nasehat, semangat dan motivasi untuk menyelesaikan proyek akhir ini.
4. Seluruh sahabat, teman se-angkatan, kakak tingkat, dan seluruh teman-teman seperjuangan baik secara langsung maupun tidak langsung terlibat dalam proses pengerjaan proyek akhir.

Dalam penyusunan laporan proyek akhir ini, penulis menyadari bahwa masih jauh dari kata sempurna. Oleh sebab itu, penulis sangat mengharapkan adanya kritik dan saran yang bersifat membangun guna penyempurnaan dalam penulisan ini. Akhir kata, penulis mengucapkan terima kasih kepada semua pihak yang telah membantu, baik dalam penyusunan laporan ini maupun masukan yang telah diberikan dalam bentuk kritik dan saran yang membangun. Semoga laporan proyek akhir ini membawa manfaat bagi pengembangan ilmu pengetahuan.

Batam, 04 April 2024

Rudcayanti Hutagol

Daftar Isi

Pernyataan Keaslian Proyek Akhir	Error! Bookmark not defined.
Abstrak	iii
Abstract	iv
Kata Pengantar	v
Daftar Isi	vi
Daftar Gambar	viii
Daftar Tabel	ix
Bab 1. Pendahuluan	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	1
1.3 Tujuan	2
1.4 Manfaat	2
1.5 Batasan	2
1.6 Work Breakdown Structure (Opsional)	3
Bab 2. Tinjauan Pustaka	4
2.1 Landasan Teori	4
1.1.1 Detak Jantung	4
1.1.2 Tekanan Darah	4
4.2 Teknologi Yang Digunakan	5
2.2.1 Teknologi Hardware	5
4.2.2 Teknologi <i>Software</i>	6
4.3 Penelitian Terdahulu	7
Bab 3. Metode Pelaksanaan	10
3.1 Teknik dan Cara Pelaksanaan	10
3.1.1 Perancangan	10
3.1.2 Perancangan Desain elektrikal	11
3.1.3 Perancangan Desain mekanikal	12
3.1.4 Pembuatan program	12

3.1.5	Pembuatan Aplikasi Android	12
3.2	Alat dan Bahan	15
3.3	Pengujian	16
Bab 4.	Hasil dan Pembahasan	22
4.1	Data Hasil Pengujian	22
4.1.1	Data Hasil Pengujian Sensor MAX30102	22
4.1.2	Data Hasil Pengujian Sensor MPX5700DP	27
4.1.3	Pengujian Aplikasi Android	30
4.2	Pembahasan	34
Bab 5.	Kesimpulan dan Saran	36
5.1	Kesimpulan	36
5.2	Saran	37
Daftar Pustaka	38
Lampiran	40
Lampiran 1.	Coding Pengujian <i>LoRa</i> Untuk Mendapatkan Nilai <i>RSSI</i>	40
Lampiran 2.	Coding Sensor MAX30102 Untuk Mengukur Detak Jantung .	43
Lampiran 3.	Coding Sensor MPX5700DP Untuk Mengukur Tekanan Darah	47
Lampiran 4.	Coding Sensor MPX5700DP Untuk Mengukur Tekanan Darah	52

Daftar Gambar

Gambar 2. 1 <i>LoRa Transmitter dan Receiver</i>	5
Gambar 2. 2 Sensor MAX30102	6
Gambar 2. 3 Sensor MPX5700DP	6
Gambar 3. 1 Flowchart Pelaksanaan Project	10
Gambar 3. 2 Perancangan Desain Elektrikal	11
Gambar 3. 3 Perancangan Desain Mekanikal <i>Transmitter</i>	12
Gambar 3. 4 Perancangan Desain Mekanikal <i>Receiver</i>	12
Gambar 3. 5 Perancangan aplikasi <i>android</i>	13
Gambar 3. 6 <i>Firebase Realtime Database</i>	14
Gambar 3. 7 <i>Use case diagram</i>	15
Gambar 3. 8 Flowchart Cara Kerja Sistem	20
Gambar 3. 9 Blok Diagram Cara Kerja Sistem	20
Gambar 4. 1 Denah Lokasi Pengujian <i>LoRa</i> (1)	22
Gambar 4. 2 Denah Lokasi Pengujian <i>LoRa</i> (2)	23
Gambar 4. 3 Denah Lokasi Pengujian <i>LoRa</i> (3)	24
Gambar 4. 4 Denah Lokasi Pengujian <i>LoRa</i> (4)	26
Gambar 4. 5 Tampilan Register	30
Gambar 4. 6 Tampilan Login	31
Gambar 4. 7 Tampilan Dashboard Hasil Pengukuran	32
Gambar 4. 8 Tampilan Dashboard masing-masing user	33
Gambar 4. 9 Tampilan Tab Informasi	34

Daftar Tabel

Tabel 1. 1 Work Breakdown Structure	3
Tabel 2. 1 Penelitian Terdahulu.....	7
Tabel 3. 1 Pin yang digunakan komponen.....	11
Tabel 3. 2 Alat dan Bahan.....	15
Tabel 3. 3 Nilai <i>RSSI</i> Pada Daerah Perkotaan.....	17
Tabel 3. 4 Nilai Detak Jantung Normal	18
Tabel 3. 5 Nilai Tekanan Darah Normal	18
Tabel 4. 1 Data Sensor MAX30102 (1).....	23
Tabel 4. 2 Data Sensor MAX30102 (2).....	24
Tabel 4. 3 Data Sensor MAX30102 (3).....	25
Tabel 4. 4 Data Sensor MAX30102 (4).....	26
Tabel 4. 5 Data Sensor MPX5700DP (1).....	27
Tabel 4. 6 Data Sensor MPX5700DP (2).....	28
Tabel 4. 7 Data Sensor MPX5700DP (3).....	29

Bab 1. Pendahuluan

1.1 Latar Belakang

Jantung berperan sebagai pompa utama dalam sistem peredaran darah manusia, memastikan darah beroksigen terdistribusi ke seluruh tubuh sehingga sel dan jaringan mendapatkan nutrisi yang diperlukan [1]. Penyakit jantung seperti Kardiomiopati, Aritmia, Penyakit Jantung Koroner, dan Infeksi Jantung merupakan ancaman serius bagi kesehatan global dan penyebab utama kematian, sering kali terdeteksi terlambat [2]. Oleh karena itu, penting untuk memiliki alat deteksi detak jantung dan tekanan darah yang efisien. Pada orang dewasa, denyut jantung normal berkisar antara 60 hingga 100 bpm dan tekanan darah normal berada di antara 90/60 mmHg hingga 120/80 mmHg [3].

Seiring dengan kemajuan teknologi komunikasi, alat pemantauan detak jantung dan tekanan darah *real-time* dan nirkabel menggunakan teknologi *LoRa* (*Long Range*) yang telah dirancang. *LoRa*, sebagai solusi dalam *Internet of Things* (*IoT*), menyediakan komunikasi nirkabel jarak jauh dengan konsumsi daya rendah dan *bit rate* yang sesuai [4]. Alat pemantauan ini mencerminkan kemajuan teknologi dan merupakan langkah penting dalam deteksi dini masalah kardiovaskular serta pemeliharaan kesehatan. Pemantauan detak jantung sangat penting, terutama dalam mendeteksi masalah jantung yang berbahaya dan sulit terdeteksi, menegaskan pentingnya alat berbasis *IoT* dengan teknologi *LoRa*.

Penelitian ini mengembangkan alat pengukur detak jantung dan tekanan darah menggunakan sensor MAX30102 dan MPX5700DP. Sensor MAX30102 mengukur cahaya inframerah yang dipantulkan dari pembuluh darah untuk memberikan informasi detak jantung, sementara sensor MPX5700DP mengukur tekanan darah secara *non-invasif* dengan memonitor perbedaan tekanan pada *manset cuff* yang dikenakan di lengan pasien. Data dari kedua sensor ini dikirim melalui jaringan *LoRaWAN* untuk pemantauan luas dan akses jarak jauh, memberikan informasi yang lengkap dan akurat mengenai kesehatan jantung pengguna. Alat ini akan menampilkan data BPM, sistolik, dan diastolik melalui aplikasi *android* dan *realtime firebase* yang dapat dipantau, sehingga memungkinkan pemantauan yang lebih efisien dan efektif oleh tenaga medis.

1.2 Rumusan Masalah

1. Bagaimana cara merancang alat pemantauan detak jantung dan tekanan darah yang efisien?
2. Bagaimana langkah-langkah mengintegrasikan alat pemantauan detak jantung dan tekanan darah yang efisien?

3. Bagaimana mengevaluasi efektivitas dari alat pemantauan detak jantung dan tekanan darah?

1.3 Tujuan

1. Merancang dan mengembangkan alat pemantauan detak jantung dan tekanan darah yang efisien dengan menggunakan sensor MAX30102 dan MPX5700DP yang mampu mengirimkan data melalui jaringan *LoRaWAN*.
2. Merumuskan dan menerapkan langkah-langkah teknis untuk mengintegrasikan alat pemantauan dengan aplikasi *Android* dan *platform web Firebase* sehingga memungkinkan pemantauan kondisi pasien secara *real-time* oleh tenaga medis dari jarak jauh.
3. Melakukan pengukuran dan evaluasi terhadap efektivitas alat pemantauan dalam mendeteksi dini dan mendukung perawatan kondisi kardiovaskular, termasuk menganalisis data yang dihasilkan untuk meningkatkan performa alat.

1.4 Manfaat

1. Bagi tenaga medis, aplikasi ini mempermudah akses data pasien dan pemantauan kesehatan secara *real-time*, serta mendukung diagnosis dan perencanaan perawatan berdasarkan data yang akurat.
2. Bagi pasien, aplikasi memungkinkan pemantauan mandiri detak jantung dan tekanan darah, memberikan kenyamanan dan rasa aman dengan akses informasi kesehatan kapan saja.
3. Bagi pengembangan teknologi kesehatan, aplikasi ini berkontribusi pada pengembangan teknologi kesehatan terintegrasi dan mendorong inovasi dalam aplikasi kesehatan berbasis *Android* dan *web*.

1.5 Batasan

1. Pada penelitian ini hanya menggunakan dua jenis sensor spesifik, yaitu sensor detak jantung (MAX30102) dan tekanan darah (MPX5700DP). Penggunaan sensor tambahan atau penggantian sensor lain tidak akan dicakup dalam penelitian ini.
2. Pada penelitian ini penambahan data pasien melalui aplikasi dan *website* hanya dapat diakses oleh tenaga medis.
3. Pada penelitian ini masih dibuat untuk 1 pengirim dan 1 penerima.
4. Spesifikasi alat:
 - Sensor detak jantung : MAX30102
 - Sensor tekanan darah : MPX5700DP
 - Modul komunikasi : *LoRaWAN*
 - Platform aplikasi : *Android* dan *Firebase*
 - *Current* : 1A

- *Voltage* : 9V
- *Supporting Frequency* : 915-925MHz
- Jarak komunikasi *point to point* : Max 1km

1.6 Work Breakdown Structure (Opsional)

Tabel 1. 1 Work Breakdown Structure

No.	Nama	Tugas dan Tanggung Jawab dalam Tim
1	Cindy Kristine Pasaribu	Akuisisi data sensor MPX5700DP, sensor MAX30102 dan Komunikasi <i>LoRaWAN</i>
2	Rudcayanti Hutagaol	Aplikasi <i>Andorid Studio</i>
3	Wisnu Ferdinan	<i>Firebase Realtime Database</i>

Bab 2. Tinjauan Pustaka

2.1 Landasan Teori

Penyakit jantung merupakan penyebab kematian utama di seluruh dunia [5], seringkali terdeteksi terlambat karena tanda-tandanya tidak selalu terlihat secara langsung. Oleh karena itu, dibutuhkan alat efisien untuk mendeteksi jantung. Denyut jantung orang dewasa normalnya berkisar antara 60 hingga 100 bpm, dan nilai tekanan darah normal berada di kisaran 90/60 mmHg hingga 120/80 mmHg. Dalam menghadapi perkembangan teknologi komunikasi, telah dirancang alat pemantauan detak jantung *real-time* dan nirkabel dengan menggunakan teknologi *LoRa (Long Range)*, solusi *IoT* dengan *telekomunikasi nirkabel* jarak jauh, konsumsi daya rendah, dan *bit rate* sesuai. Alat ini mencerminkan kemajuan teknologi dan merupakan langkah penting dalam deteksi dini masalah kardiovaskular.

Penelitian sebelumnya menggunakan sensor MAX30100 dan *Arduino Uno* untuk mengukur kadar oksigen *non-invasif* dengan tambahan alarm [6]. Riset lain memanfaatkan *pulse* sensor dan *Raspberry Pi*, serta sensor MAX30102 dengan ESP8266 dan *Blynk* untuk mendeteksi detak jantung, namun dengan keterbatasan jarak pemantauan [7]. Riset ini akan menggunakan sensor MAX30102 dan MPX5700DP terhubung pada *LoRa Transmitter* untuk mendeteksi detak jantung dan tekanan darah. Data ditransfer ke *mikrokontroler LoRa Receiver* dan ditampilkan melalui *Liquid Crystal Display (LCD)* serta *Android Studio*.

1.1.1 Detak Jantung

Detak jantung adalah denyutan yang terjadi saat jantung memompa darah melalui pembuluh darah, dapat dirasakan di titik-titik seperti pergelangan tangan dan leher. Frekuensinya diukur dalam denyutan per menit (BPM), dengan orang dewasa sehat biasanya memiliki 60-100 BPM saat istirahat, dan atlet sering lebih rendah. Detak jantung memompa darah kaya oksigen ke seluruh tubuh dan mengembalikan darah mengandung karbon dioksida ke jantung untuk dipompa ke paru-paru. Aktivitas fisik, emosi, suhu tubuh, dan kondisi medis dapat mempengaruhi detak jantung. Pengukuran bisa dilakukan secara manual atau dengan alat seperti stetoskop dan perangkat wearable. Memantau detak jantung penting untuk memastikan jantung bekerja dengan baik; detak jantung yang terlalu cepat (takikardia) atau terlalu lambat (bradikardia) bisa menunjukkan masalah kesehatan yang perlu diperiksa lebih lanjut.

1.1.2 Tekanan Darah

Tekanan darah adalah kekuatan yang dihasilkan oleh darah saat mengalir melalui pembuluh darah, diukur dalam milimeter merkuri (mmHg) dan terdiri dari dua angka: tekanan sistolik (saat jantung berdetak) dan tekanan diastolik (saat jantung beristirahat). Tekanan darah normal pada orang dewasa adalah sekitar

120/80 mmHg. Hipertensi (tekanan darah tinggi) dan hipotensi (tekanan darah rendah) dapat menunjukkan masalah kesehatan serius. Faktor-faktor seperti gaya hidup, pola makan, aktivitas fisik, berat badan, usia, dan kondisi medis mempengaruhi tekanan darah. Pengukuran dilakukan dengan sphygmomanometer. Pemantauan rutin penting untuk mencegah komplikasi kesehatan, dan perubahan signifikan harus dikonsultasikan dengan dokter.

4.2 Teknologi Yang Digunakan

2.2.1 Teknologi Hardware

4.2.1.1 LoRa

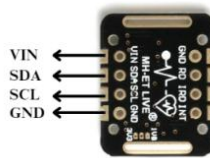
Pada Gambar 2.1 merupakan mikrokontroler yang digunakan sebagai komunikasi yang terdiri dari *LoRa Transmitter* (pengirim) dan *LoRa Receiver* (penerima). *LoRa* adalah komunikasi nirkabel (*wireless*) yang memberikan informasi jarak jauh tanpa menggunakan tegangan listrik atau kawat dimana komunikasi nirkabel menggunakan media udara sebagai tempat transmisinya. *LoRa* adalah singkatan dari *Long Range* dan merupakan teknologi jaringan nirkabel berdaya rendah dengan jangkauan luas, transmisi dua arah, dan volume data rendah [8]. Digunakan sebagai komunikasi sensor, *LoRa* mentransmisikan data sensor dari *transmitter* ke *receiver* dengan jarak *line of sight*. Nantinya *LoRa* akan digunakan sebagai komunikasi sensor dengan jarak jauh, dimana data sensor MAX30102 dan MPX5700DP akan direkam oleh *LoRa transmitter* dan akan dikirimkan ke *LoRa receiver* yang berada di tempat lain.



Gambar 2. 1 *LoRa Transmitter dan Receiver*

4.2.1.2 MAX30102

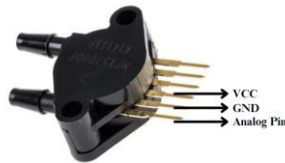
Pada Gambar 2.2 merupakan sensor MAX30102 yang dikembangkan oleh Maxim Integrated merupakan sensor *infra-red* yang dirancang khusus untuk pengukuran detak jantung secara *non-invasif*. MAX30102 mampu mengukur perubahan volume darah dalam pembuluh darah manusia. Keunggulan MAX30102 meliputi ukuran kecil, konsumsi daya rendah, dan kemampuan untuk mengukur detak jantung dan SpO2 dengan akurasi yang tinggi agar data yang diterima pihak medis mengenai kesehatan pasien lengkap dan lebih mudah.



Gambar 2. 2 Sensor MAX30102

4.2.1.3 MPX 5700DP

Pada Gambar 2.3 merupakan sensor tekanan MPX5700DP yang digunakan untuk mengukur tekanan *fluida* dalam berbagai aplikasi. Sensor MPX5700DP mengonversi perbedaan tekanan antara tekanan di dalam manset dan tekanan atmosfer menjadi sinyal listrik. Sinyal ini kemudian diproses oleh perangkat elektronik yang terintegrasi dengan sensor untuk menghitung tekanan darah pasien. Penggunaan sensor MPX5700DP dalam pengukuran tekanan darah memiliki beberapa keunggulan, termasuk akurasi yang tinggi dan respons yang cepat terhadap perubahan tekanan. Selain itu, sensor ini juga dapat diandalkan untuk pengukuran tekanan darah yang konsisten dan dapat diulang. Dengan menggabungkan sensor MPX5700DP dengan *manset cuff*, dokter atau tenaga medis dapat memantau tekanan darah pasien dengan cepat dan akurat.



Gambar 2. 3 Sensor MPX5700DP

4.2.2 Teknologi Software

4.2.2.1 Android Studio

Android Studio dilengkapi dengan berbagai fitur, termasuk editor kode yang kuat dengan dukungan penuh untuk bahasa pemrograman *Java*, serta berbagai alat pengembangan visual yang memudahkan desain antarmuka pengguna aplikasi *Android*. Pembuatan aplikasi *Android* digunakan sebagai tampilan *output* dari sensor. Pembuatan Aplikasi *Android* ini menggunakan *software Android Studio* yang terhubung ke database *Firebase*, yang nantinya Aplikasi *Android* akan menampilkan data yang masuk ke *database*.

4.2.2.2 Firebase Realtime Database

Firebase Realtime Database adalah layanan *cloud-hosted NoSQL* di bawah *Firebase SDK*. Menyediakan penyimpanan data yang dapat disinkronkan *real-time*

ke semua klien terhubung. Aplikasi yang terhubung juga dapat memberikan data saat *offline*, berkat fitur *Firebase SDK* yang menyinkronkan data pada perangkat penyimpanan klien.

4.2.2.3 Desain Mekanikal

Pembuatan desain mekanikal menggunakan perangkat *software Auto CAD*, yang dimana desain mekanikal nantinya akan digunakan sebagai casing dari komponen-komponen yang akan dipakai.

4.2.2.4 Desain Elektrikal

Pembuatan desain elektrikal dibuat melalui perangkat *software EasyEDA*, yang dimana komponen-komponen yang digunakan yaitu Sensor *MAX30102*, sensor *MPX5700DP*, *LoRa*, *solenoid valve*, *motor*, *resistor*, *transistor*, *push button* dan kabel *jumper*.

4.3 Penelitian Terdahulu

Peradaban manusia saat ini mengalami kemajuan pesat, khususnya di bidang teknologi. Berbagai alat *portabel*, termasuk yang digunakan untuk pengukuran detak jantung dan tekanan darah, telah dikembangkan melalui penelitian dengan pemanfaatan teknologi yang beragam.

Tabel 2. 1 Penelitian Terdahulu

No.	Nama Peneliti	Judul	Persamaan	Perbedaan
1.	Inda, dkk.2021.[9].	Sistem Monitoring <i>Heart Rate</i> dan Oksigen Dalam Darah Berbasis <i>LoRa</i> .	Persamaannya adalah: Menggunakan teknologi <i>LoRa</i> untuk mengirimkan data dari alat pengukuran detak jantung dari <i>transmitter</i> ke <i>receiver</i> . Data ini kemudian ditampilkan di <i>LCD</i> dan	Perbedaan utama antara penelitian sebelumnya dan penelitian saat ini adalah Penelitian sebelumnya hanya memusatkan pada pemantauan <i>Heart Rate</i> dan Oksigen Dalam Darah menggunakan <i>LoRa</i> , dengan data ditampilkan pada aplikasi <i>Android</i> . Sementara itu, penelitian saat ini memiliki kemampuan

			aplikasi <i>Android</i> .	untuk memonitor detak jantung dan tekanan darah secara <i>remote</i> melalui jaringan <i>LoRaWAN</i> , dengan hasil pengukuran yang ditampilkan baik pada <i>platform web</i> maupun aplikasi <i>Android Studio</i> .
2.	Isyanto, H., dkk.2022 [10].	Rancang Bangun Alat Pendeteksi Detak Jantung, Suhu Tubuh, dan <i>Tensimeter</i> Berbasis <i>Arduino Uno</i> serta <i>Smartphone Android</i>	Persamaannya adalah: pada penelitian ini sama sama melakukan pengukuran detak jantung dan tekanan darah.	Perbedaan penelitian sebelumnya dan penelitian saat ini terletak pada komponen dan jaringan. Penelitian sebelumnya menggunakan sensor suhu DS18B20, sensor detak jantung pulsemeter, dan Sensor Tekanan MPX5050GP dengan <i>Arduino Uno</i> dan <i>smartphone Android</i> . Sementara itu, penelitian saat ini memungkinkan pemantauan detak jantung dan tekanan darah dari jarak jauh melalui jaringan <i>LoRaWAN</i> , dengan hasil pengukuran ditampilkan di platform web dan aplikasi <i>Android Studio</i> , memberikan fleksibilitas lebih tanpa terbatas pada jaringan <i>Wi-Fi</i> .
3.	Yogiarditha, Agus N.2020 [11].	Perancangan Alat Ukur Tekanan Darah Dan	Persamaannya adalah: pada penelitian ini sama sama	Perbedaan penelitian sebelumnya dan penelitian saat ini terletak pada sensor,

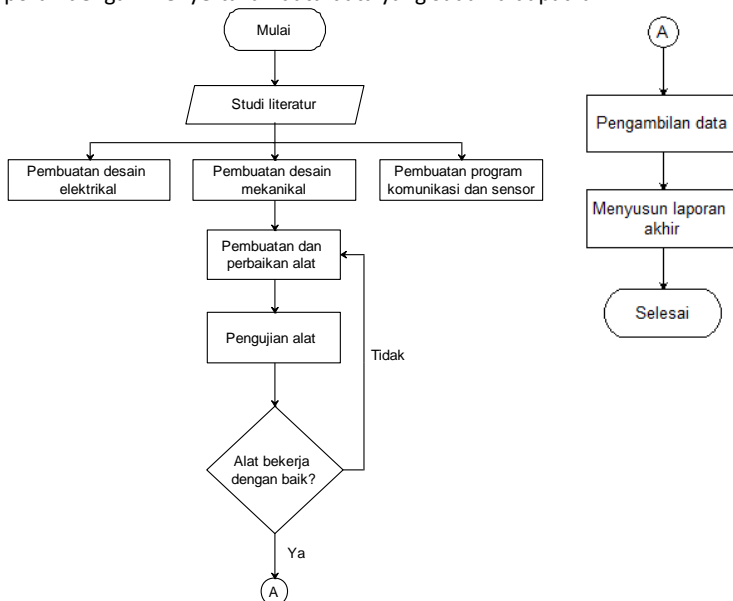
		Jumlah Denyut Jantung Berbasis <i>Arduino Uno</i> .	melakukan pengukuran detak jantung dan tekanan darah.	metode pengiriman data, dan jaringan yang digunakan. Penelitian sebelumnya menggunakan sensor MPX5700DP untuk tekanan darah dan sensor APDS9008 untuk denyut jantung. Data diproses oleh <i>mikrokontroler Arduino Uno</i> dan dikirim melalui ESP8266 dengan <i>Wi-Fi</i> ke <i>web browser</i> . Sementara itu, penelitian saat ini menggunakan jaringan <i>LoRaWAN</i> untuk memonitor detak jantung dan tekanan darah dari jarak jauh, dengan hasil pengukuranditampilkan pada <i>platform web</i> dan aplikasi <i>Android Studio</i> .
--	--	---	---	---

Bab 3. Metode Pelaksanaan

3.1 Teknik dan Cara Pelaksanaan

3.1.1 Perancangan

Tahap perancangan sistem dapat dilihat pada Gambar 3.1, yang menjelaskan secara umum urutan pelaksanaan yang akan dilakukan. Tahap perencanaan dimulai dari *studi literatur* untuk menyusun kerangka penelitian yang sistematis, membantu dalam penulisan laporan penelitian dengan memberikan konteks yang diperlukan dalam tinjauan pustaka, serta menginspirasi ide-ide baru yang mendorong inovasi dan pemikiran kritis. Setelah itu, dilakukan pembuatan desain elektrik, mekanikal, serta pembuatan program komunikasi dan sensor yang digunakan. Selanjutnya, dilakukan pembuatan dan perbaikan alat di mana sensor yang digunakan dihubungkan ke mikrokontroler sesuai dengan alur desain elektrik, dan fungsi dari mekanikal yang dibuat sebagai wadah untuk meletakkan sensor agar lebih rapi. Setelah semua alat selesai dibuat, dilanjutkan dengan pengujian alat untuk mengukur detak jantung dan tekanan darah. Setelah alat diuji, data sensor diambil untuk memastikan keakuratan alat. Terakhir, menyusun laporan dengan menyertakan data-data yang sudah didapatkan.



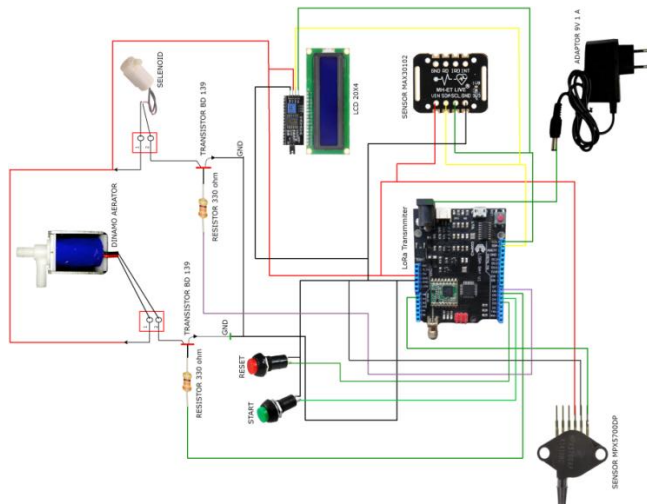
Gambar 3. 1 Flowchart Pelaksanaan Project

3.1.2 Perancangan Desain elektrikl

Pada Gambar 3.2 menunjukkan tahap perancangan elektrikl dengan panduan *wiring* komponen. Selain itu pada desain elektrikl ini menekankan pentingnya penggunaan pin yang sesuai agar komponen berfungsi maksimal. Pada tahap perancangan elektrikl diperlukan *referensi* untuk dapat melakukan *wiring* pada komponen-komponen yang akan dipakai. Pin yang akan dipakai harus sesuai agar komponen yang digunakan agar dapat berjalan maksimal. Pembuatan desain elektrikl dibuat melalui *software EasyEDA*, yang dimana komponen-komponen yang digunakan yaitu *LoRa Ray V1*, *LoRa Aurora V2*, sensor MAX30102 , MPX5700DP, *Selenoid*, *Dinamo Aerator*, *Lcd I2C* dan *Push Button*. Berikut tabel pin yang digunakan dalam desain elektrikl ini:

Tabel 3. 1 Pin yang digunakan komponen

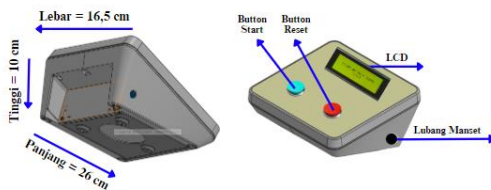
Komponen	Terhubung Ke-	Komponen	Terhubung Ke-	Terhubung Ke-
Sensor MPX5700DP		Motor Pump		
Analog Pin	Pin A0	VCC	5V	
GND	GND	GND	GND	Transistor BD139 - Collector
VCC	VCC	Selenoid Valve		
Sensor MAX30102		VCC	5V	
Vin	5V	GND	GND	Transistor BD139 - Collector
SDA	SDA	Transistor Motor Pump		
SCL	SCL	Emitter	GND	
GND	GND	Collector	Motor Pump	
Button Start		Base	Resistor 330 ohm	Pin D5
VCC	Pin D4	Transistor Selenoid Valve		
GND	GND	Emitter	GND	
Button Stop		Collector	Selenoid Valve	
VCC	Pin D3	Base	Resistor 330 ohm	Pin D6
GND	GND			
LCD				
GND	GND			
VCC	5V			
SDA	SDA			
SCL	SCL			



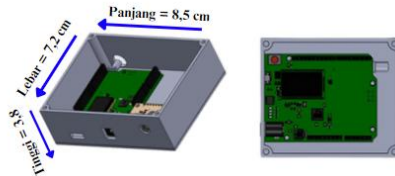
Gambar 3. 2 Perancangan Desain Elektrikal

3.1.3 Perancangan Desain mekanikal

Pada Gambar 3.3 adalah tahap setelah perancangan desain elektrik yaitu desain mekanikal yang diperlukan untuk membuat sebuah perangkat keras atau sebuah kotak yang digunakan untuk meletakkan komponen yang telah di wiring agar tidak berantakan dan lebih terlihat rapi. Bentuk dari desain mekanikal ukurannya juga harus sesuai dengan total ukuran komponen yang akan dipakai. Untuk membuat perancangan desain mekanikal diperlukan sebuah *software* maka dari itu akan digunakan *software SOLIDWORKS*. Pada *case* mekanikal yang digunakan terdapat 2 *button* sebagai *start* dan *reset*, *LCD* sebagai *interface* untuk menampilkan data detak jantung dan tekanan darah kemudian ada beberapa lubang untuk selang manset, antena dan adaptor.



Gambar 3. 3 Perancangan Desain Mekanikal *Transmitter*



Gambar 3. 4 Perancangan Desain Mekanikal *Receiver*

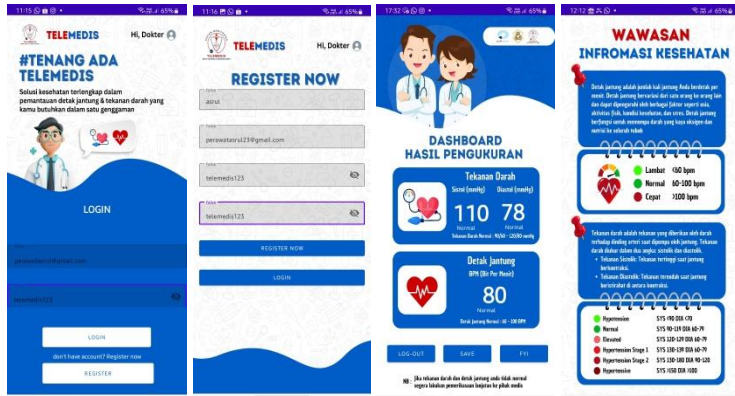
3.1.4 Pembuatan program

Perancangan program pada penelitian ini menggunakan *software Arduino IDE* yang digunakan untuk memprogram sensor MAX30102 untuk mengukur detak jantung dan sensor MPX5700DP untuk mengukur tekanan darah yang dihubungkan menggunakan *mikrokontroler LoRa*. Kemudian membuat program agar hasil nilai yang didapatkan dapat ditambahkan ke *serial monitor*, *LCD* dan *firebase*. Program ini juga harus disesuaikan dengan spesifikasi sensor yang digunakan seperti *library* sensor dan harus memperhatikan penggunaan pin sesuai desain elektrik.

3.1.5 Pembuatan Aplikasi Android

Saat melakukan pengukuran akan ada data yang akan ditampilkan maka dari itu diperlukan *interface* yang dimana berguna untuk menampilkan data tersebut.

Interface yang akan digunakan yaitu aplikasi *android* yang dimana aplikasi tersebut akan lebih memudahkan penggunaannya karena pemakaiannya aplikasi yang mudah dan tidak membingungkan penggunaannya. Aplikasi *android* tersebut akan dibuat di *software Android Studio* dan aplikasi tersebut akan terhubung dengan *database* yang digunakan.

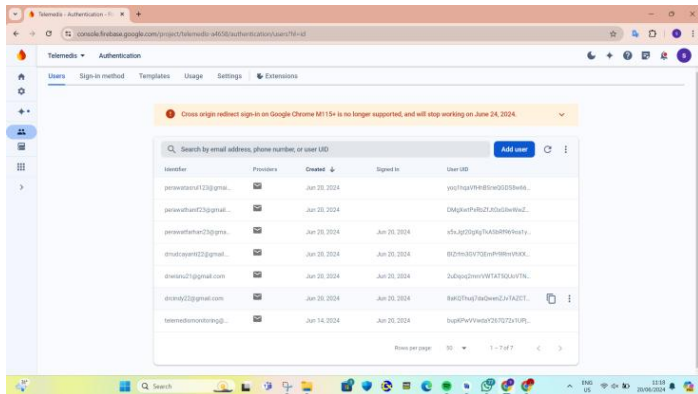
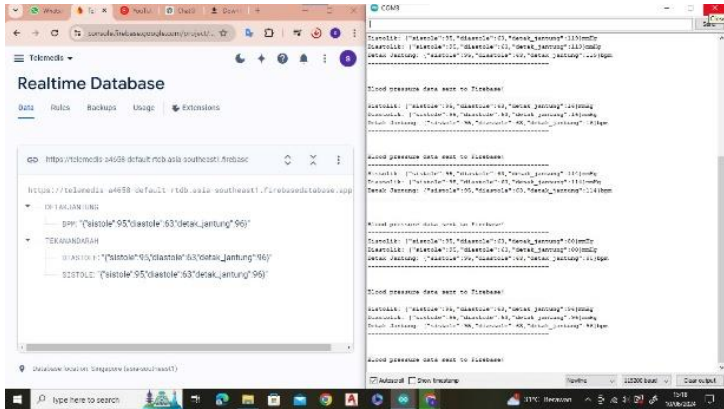


Gambar 3. 5 Perancangan aplikasi *android*

Pada gambar 3.4 ini terdapat beberapa fitur yaitu Aplikasi ini memiliki fitur *login* yang meminta pengguna untuk memasukkan *email* dan *password* mereka. Selain itu, pada fitur *register*, pengguna diminta untuk memasukkan nama lengkap, alamat *email*, *password*, dan konfirmasi ulang *password* untuk keamanan. Setelah *login* berhasil, tampilan selanjutnya menampilkan nilai detak jantung dan tekanan darah pengguna. Dengan demikian, aplikasi ini memberikan akses yang aman dan memudahkan pengguna untuk memantau nilai detak jantung dan tekanan darah pasien.

3.1.6 Pembuatan *Firestore Realtime Database*

Selain aplikasi *android* yang digunakan sebagai *interface*, *firebase* juga akan digunakan sebagai *interface firebase* yang akan terhubung ke sebuah *database* yang menerima data detak jantung dan tekanan darah secara *realtime*. Pada Gambar 3.5 merupakan tampilan yang ada pada *realtime firebase* ini yaitu menampilkan nilai tekanan darah yaitu sistolik dan diastolik kemudian menampilkan nilai detak jantung pasien. Selain itu juga data pasien yang sudah di daftarkan pada aplikasi *android* akan muncul dan tersimpan di *firebase*. Kemudian jika dokter memeriksa pasien dan memasukkan email dan *password* yang sudah didaftarkan juga akan muncul di *firebase* beserta dengan data tekanan darah dan detak jantung pasien tersebut.

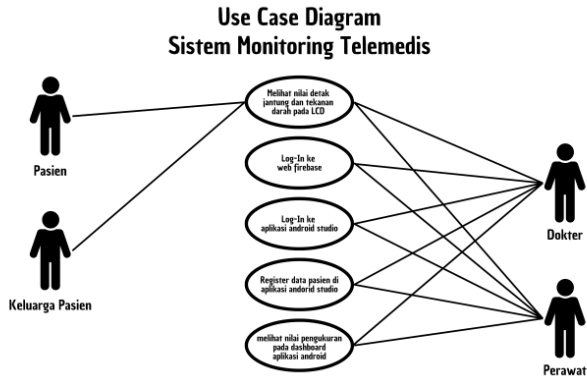


Gambar 3. 6 Firebase Realtime Database

3.1.7 Use Case Diagram

Use case diagram pada gambar 3.6 menunjukkan interaksi antara aktor dan fungsi-fungsi dalam sistem monitoring telemedis. Aktor-aktor yang terlibat adalah pasien, keluarga pasien, dokter, dan perawat. Masing-masing aktor memiliki akses ke berbagai fungsi dalam sistem. Pasien dan keluarga pasien dapat melihat nilai detak jantung dan tekanan darah pada layar LCD. *Login ke web Firebase*, *login ke aplikasi Android Studio*, dan melihat hasil pengukuran kesehatan pada *dashboard* aplikasi *Android* juga dapat dilihat dan diakses oleh dokter dan perawat selain itu juga dokter dan perawat memiliki kemampuan untuk melakukan registrasi data pasien di aplikasi *Android Studio*. Setiap *use case* dihubungkan dengan garis ke aktor-aktor yang berinteraksi dengan *use case* tersebut, memperlihatkan hubungan antara aktor dan fungsi sistem. Diagram ini memberikan gambaran yang

jelas tentang bagaimana berbagai pengguna berinteraksi dengan sistem untuk memonitor, melihat, dan mendaftarkan data kesehatan pasien, menunjukkan peran dan tanggung jawab masing-masing aktor dalam ekosistem telemedis ini.



Gambar 3. 7 Use case diagram

3.2 Alat dan Bahan

Dalam pelaksanaan pembuatan proyek ini dibutuhkan alat dan bahan yang dibutuhkan dapat dilihat pada table 3.2 sebagai berikut :

Tabel 3. 2 Alat dan Bahan

Alat dan Bahan	Banyaknya	Harga Satuan (Rp)	Nilai (Rp)
Sensor <i>MPX5700DP</i>	1 buah	300.00	300.000
Sensor <i>MAX30102</i>	1 buah	245.000	245.000
<i>Pulse Oximeter Fingertip</i>	1 buah	155.000	155.000
Tensimeter Digital	1 buah	440.000	440.000
<i>LoRa Aurora ESP32 ESP 32 Arduino 915 MHz 915MHz - SMA</i>	1 buah	749.000	749.000
<i>LoRa Arduino Development Board 915MHz 915 MHz + Antenna + Charger - LoRa Ray V3 SMA</i>	1 buah	605.000	605.000
Kabel data <i>micro USB</i>	1 buah	76.000	76.000
Kabel data <i>USB type C</i>	1 buah	50.000	50.000
Adaptor 9V 2A	2 buah	35.000	70.000

Jumper <i>Male To Female</i> 10cm	2 buah	15.000	30.000
Jumper <i>Male To Male</i> 10cm	2 buah	15.000	30.000
<i>Breadboard</i>	1 buah	8.000	8.000
Lem tembak + stik lem5	1 buah	55.000	55.000
Solder 60W <i>with 5 Tips</i>	1 buah	55.000	55.000
Isolasi bakar 4 mm	3 buah	2.500	7.500
Case OKW <i>Size M</i>	1 buah	1.000.000	1.000.000
LCD 20X4 <i>I2C Interface</i>	1 buah	75.000	75.000
<i>Push Button ON OFF</i> 22 mm	2 buah	7.000	14.000
Obeng Set – USD603	1 set	144.000	144.000

3.3 Pengujian

Dalam pengujian ini, terdapat beberapa tahapan yang akan dilakukan. Sebelum melakukan pengujian secara keseluruhan, dilakukan beberapa tahap pengujian sebagai berikut:

3.3.1 Pengujian *LoRa (Long Range)*

Pengujian ini bertujuan untuk memastikan komunikasi antara *transmitter* dan *receiver* dapat berjalan dengan baik tanpa terhalang oleh benda apapun sehingga pengiriman data dapat diterima oleh *receiver*. Pengujian *LoRa* ini dilakukan dengan mengambil data nilai *RSSI (Receive Signal Strength Indicator)* merupakan parameter untuk mengukur indikator kekuatan sinyal yang diterima. Nilai *RSSI* sangat bergantung pada kondisi lingkungan yaitu jarak dan penghalang. Semakin jauh dan semakin banyak penghalangnya maka nilai *RSSI* akan menurun. Hal ini dapat mempengaruhi kinerja dari *LoRa*. Rumus untuk menghitung rata-rata *RSSI* adalah:

$$RSSI \text{ (dBm)} = \frac{\text{Total nilai RSSI yang diterima}}{\text{Jumlah sampel RSSI yang diterima}} \quad (1)$$

Transmisi Pada Lingkungan Outdoor

terdapat 4 faktor-faktor transmisi sinyal pada lingkungan outdoor diantaranya:

2. Jarak

Jarak antara pemancar sinyal dan penerima sangat akan menentukan kualitasnya. Semakin dekat maka akan semakin kuat, dan sebaliknya jika jarak yang berjauhan maka sinyal akan semakin lemah.

3. Kekuatan sinyal pemancar

Dari sisi pemancar dapat perlu diperhatikan. Karena besarnya daya pancar yang dihasilkan dari antena akan sangat berpengaruh pada sisi penerimanya. Faktor interferensi dan jarak adalah penyebab utamanya. Jadi, semakin kuat sinyal yang dipancarkan, maka akan semakin besar sinyal yang diterima oleh penerima.

4. *Interferensi* (gangguan sinyal)

Interferensi ini sering terjadi karena penggunaan *channel frequency* yang sama oleh dua bahkan lebih perangkat, sehingga hal tersebut dapat menyebabkan gangguan atau hambatan. Jadi yang perlu diingat adalah jika ada dua *access point* pastikan keduanya itu beroperasi pada *channel* yang berbeda.

5. *LoS* (*Line of Sight*)

LoS merupakan sebuah kondisi dimana pada area yang berupa garis lurus antara pemancar sinyal dan penerima yang tidak terhalang oleh benda apapun.

Tabel 3. 3 Nilai RSSI Pada Daerah Perkotaan

Nilai RSSI untuk komunikasi LoRa di lingkungan perkotaan dengan kepadatan penduduk yang tinggi dan bangunan-bangunan tinggi	
RSSI (dBm)	Kualitas Sinyal
-80	Kuat
-120	Lemah

3.3.2 Pengujian Sensor MAX30102

Pengujian pada perangkat sensor MAX30102 ini bertujuan untuk memastikan keakuratan serta ketepatan alat ini dalam mengukur detak jantung manusia. Denyut jantung orang dewasa normalnya berkisar antara 60 hingga 100 bpm. Sebelum memulai pengukuran harus membuat program sensor MAX30102 setelah program sudah selesai lanjut mengambil data pengukuran sebanyak 10 kali dalam satu pengambilan data. nantinya alat ini juga akan dibandingkan dengan alat ukur *heart rate* yaitu *pulse oxymeter*. Pada kolom data akan berisi data pengujian detak jantung dari MAX30102 dan data detak jantung dari *Pulse Oxymeter* dan setelah pengujian dilakukan data tersebut akan dicari *persentase error* sebagai nilai seberapa akurat MAX30102 dibandingkan dengan alat *Pulse Oxymeter*.

Tabel 3. 4 Nilai Detak Jantung Normal

Denyut Jantung Normal			
Usia	Minimal	Maximal	Rata-Rata
Baru Lahir	100	180	140
1 Bulan - 1 Tahun	80	160	120
1 Tahun - 3 Tahun	80	130	105
3 Tahun - 6 Tahun	80	120	100
6 Tahun - 12 Tahun	65	100	83
12 Tahun - 18 Tahun	60	90	85
19 Tahun - 69 Tahun	60	100	80
> 70 Tahun	60	100	80

3.3.3 Pengujian Sensor MPX5700DP

Pengujian pada perangkat sensor MPX5700DP ini bertujuan memastikan keakuratan serta ketepatan alat ini dalam mengukur tekanan darah dan agar sensor ini berfungsi untuk mengukur tekanan darah manusia harus membuat program dimana menggunakan rumus dari tegangan diubah ke tekanan lalu ke mmHg. dan nilai tekanan darah normal berada di kisaran 90/60 mmHg hingga 120/80 mmHg. Setelah program selesai sensor MPX5700DP harus dihubungkan ke *manset cuff*, *solenoid* dan *dianamo aerator*. Setelah semuanya terhubung kemudian lanjut untuk pengambilan data sistolik dan diastolik manusia sebanyak 10 kali dalam satu pengambilan data. Nantinya akan dibandingkan dengan alat ukur tensimeter digital. Pada kolom data akan berisi data pengujian tekanan darah dari MPX5700DP dan data tekanan darah dari tensimeter digital dan setelah pengujian dilakukan data tersebut akan dicari *persentase error* sebagai nilai seberapa akurat MPX5700DP dibandingkan dengan tensimeter digital.

Tabel 3. 5 Nilai Tekanan Darah Normal

Tekanan Darah (mmHg)		
Sistolik	Diastolik	Kategori
<120	<80	Normal
120-139	80-89	Prahipertensi
140-159	90-99	Hipertensi Stadium 1
>160	>100	Hipertensi Stadium 2

3.3.4 Cara mendapatkan %Error

Sistem diuji dengan mengambil data detak jantung dan tekanan darah dari beberapa orang. Hasilnya dibandingkan dengan alat komersial seperti *PulseMeter* dan tensimeter digital untuk mengetahui % error alat yang dibuat dengan alat

pembandingan. Data dari sensor akan ditampilkan menggunakan grafik untuk mengetahui apakah data yang didapat *linear* atau tidak.

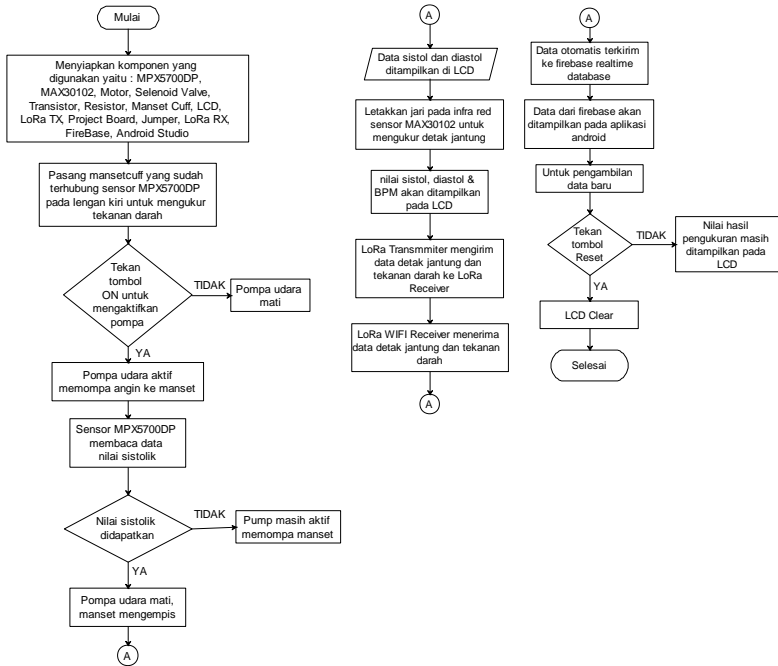
$$\%Error = ABS \frac{(Sensor - Pembanding)}{Pembanding} * 100 \quad (2)$$

Setelah semua Data detak jantung dan tekanan darah dikumpulkan, lanjut untuk membuat program pada *receiver* agar dapat terkoneksi ke *firebase realtime database* sehingga data detak jantung dan tekanan darah pasien yang diterima oleh *receiver* dari *transmitter* akan ditampilkan pada *firebase* dan juga *android studio*.

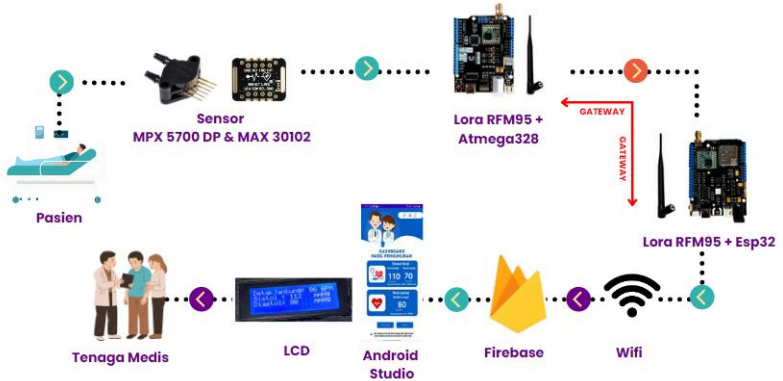
3.4 Cara Kerja Sistem

Berikut *flowchart* yang menjelaskan cara kerja sistem monitoring telemedis. Pada Gambar 3.6 Sistem pengukuran detak jantung dan tekanan darah dimulai dengan menghubungkan komponen utama, yaitu MPX5700DP (sensor tekanan darah), MAX30100 (sensor detak jantung), motor, *solenoid valve*, *transistor*, *resistor*, *manset*, *cuff*, *LCD*, *LoRa Transmitter*, *Project Board*, *jumper*, *LoRa Receiver*, *Firebase*, dan *Android Studio*. Langkah awal melibatkan pemasangan semua komponen yang terhubung ke sensor MPX5700DP pada lengan kiri untuk mengukur tekanan darah. Sistem dioperasikan melalui tombol *Start* dan *reset*. Ketika tombol *Start* ditekan, pompa udara akan diaktifkan, dan sensor MPX5700DP mulai membaca tekanan darah. Data tekanan sistolik akan disiapkan untuk ditampilkan. Jika tekanan darah sudah diukur, pompa udara akan dimatikan. Selanjutnya, pengguna harus meletakkan jarinya pada infra merah sensor MAX30100 untuk mengukur detak jantung. Nilai sistol, diastol, dan BPM akan ditampilkan pada *LCD*.

LoRa Transmitter kemudian mengirim data detak jantung dan tekanan darah ke *LoRa Receiver*. *Receiver* akan menerima data ini dan mengirimkannya ke *database Firebase*. Data tersebut akan disimpan dan ditampilkan di aplikasi *Android* yang dikembangkan menggunakan *Android Studio*. Jika tombol *reset* ditekan *LCD* akan *clear* dan dapat memulai pengukuran baru. Proses selesai setelah semua data dikirim, disimpan, dan ditampilkan sesuai kebutuhan pengguna. Sistem ini memungkinkan pemantauan kesehatan secara otomatis dan *real-time* melalui perangkat seluler.



Gambar 3. 8 Flowchart Cara Kerja Sistem



Gambar 3. 9 Blok Diagram Cara Kerja Sistem

Pada Gambar 3.7 merupakan blok diagram yang menggambarkan alur sistem pemantauan kesehatan pasien secara nirkabel menggunakan teknologi

LoRa dan *WiFi*. Sistem dimulai dari pasien yang menggunakan sensor MPX5700DP untuk mengukur tekanan darah dan MAX30102 untuk mengukur detak jantung. Data yang diperoleh dari sensor ini kemudian dikirimkan ke modul *LoRa RFM95* yang terhubung dengan mikrokontroler *Atmega328*. Modul *LoRa RFM95* ini bertindak sebagai pengirim data ke *gateway* yang juga menggunakan *LoRa RFM95*, tetapi terhubung dengan mikrokontroler *Esp32*. *Gateway* ini berfungsi untuk menerima data dari modul *LoRa* sebelumnya dan meneruskannya melalui koneksi *WiFi* ke server *web*.

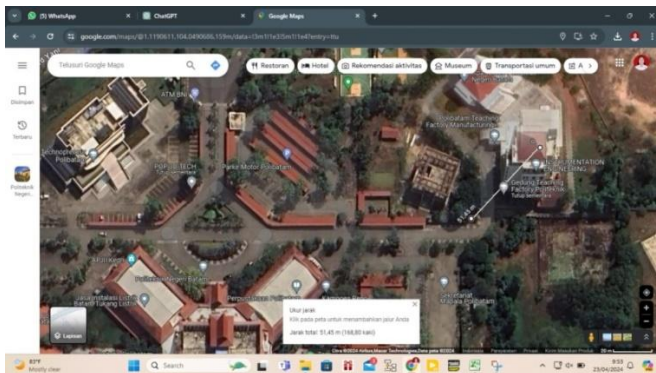
Dari server *web*, data tersebut bisa diakses oleh dokter dan keluarga pasien melalui perangkat yang terhubung ke internet. Data ini juga bisa ditampilkan pada layar *LCD* untuk pemantauan lokal. Sistem ini memungkinkan pengawasan kesehatan pasien secara *real-time* dan jarak jauh, memberikan akses cepat dan mudah bagi dokter dan keluarga untuk memonitor kondisi pasien, serta memastikan data kesehatan tersimpan dengan baik di server *web* untuk analisis lebih lanjut.

Bab 4. Hasil dan Pembahasan

4.1 Data Hasil Pengujian

4.1.1 Data Hasil Pengujian Sensor MAX30102

Pengujian pada sensor MAX30102 dilakukan pengambilan data detak jantung sebanyak 10 kali percobaan secara berurutan. Data detak jantung diambil dari beberapa lokasi dengan jarak yang berbeda seperti pada gambar 4.1 ini merupakan denah lokasi dan jarak pengujian *LoRa transmitter* dan *receiver* dari Gedung RTF ke Dasar RTF dengan jarak jauh 51,45 m (168,80 kaki).



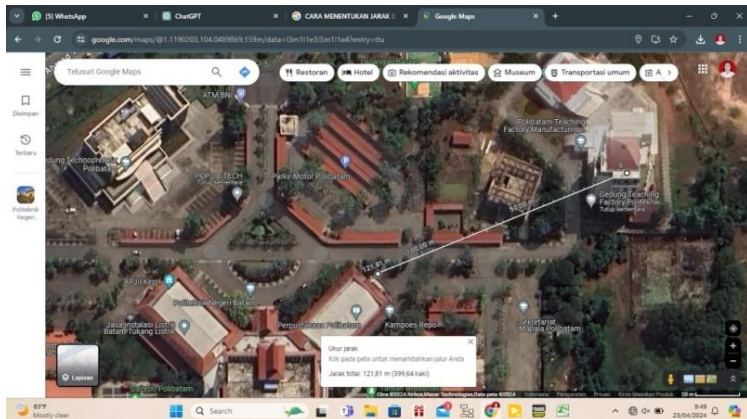
Gambar 4. 1 Denah Lokasi Pengujian LoRa (1)

Pada tabel 4.1 dibawah ini merupakan hasil pengujian yang dilakukan dengan mengambil 10 data detak jantung dari 1 pasien tanpa riwayat hipertensi. Pengambilan data detak jantung sebanyak 10 kali bertujuan untuk memperoleh nilai rata-rata dari pengukuran yang dihasilkan oleh sensor. Hasil pengukuran dari sensor kemudian dibandingkan dengan hasil pengukuran dari alat pembanding, yaitu *pulse meter*, yang berfungsi sebagai tolak ukur untuk hasil penghitungan dari sensor yang digunakan. Hal ini dilakukan untuk menentukan seberapa besar kesalahan atau gangguan (*noise*) yang terjadi saat pembacaan dengan sensor. Berdasarkan pengujian yang telah dilakukan, rata-rata % *error* pengukuran detak jantung menggunakan sensor MAX30102 dan alat pembanding yaitu *pulse* sensor adalah 2,8%, dengan nilai kesalahan tertinggi sebesar 4,8% dan terendah sebesar 0%.

Tabel 4. 1 Data Sensor MAX30102 (1)

Lokasi : RTF LT 7 Ke LT Dasar RTF					RECAIVER				
Jarak TX Ke RX : 51,45 m (168,80 kaki)									
TRANSMITTER									
No.	Time	Sensor MAX30102 (BPM)	Pulse Meter (BPM)	% ERROR	No.	Time	Sensor MAX30102 (BPM)		
PERCOBAAN KE-	1	16:35:03	101	98	3,1	PERCOBAAN KE-	1	16:35:10	101
	2	16:36:32	82	80	2,5		2	16:36:38	82
	3	16:36:35	84	83	1,2		3	16:36:41	84
	4	16:37:35	101	97	4,1		4	16:37:42	101
	5	16:38:18	105	107	1,9		5	16:38:25	105
	6	16:38:53	104	107	2,8		6	16:39:02	102
	7	16:39:58	90	86	4,7		7	16:40:04	90
	8	16:40:41	87	83	4,8		8	16:40:48	87
	9	16:41:02	91	88	3,4		9	16:41:08	91
	10	16:41:52	94	94	0,0		10	16:41:35	94
MIN		82	80	0,0					
MAX		105	107	4,8					
AVG				2,8					

Pada gambar 4.2 ini merupakan denah lokasi dan jarak pengujian *LoRa transmitter* dan *receiver* dari Gedung RTF ke depan Gedung Utama dengan jarak jauh 121,81 m (399,64 kaki).



Gambar 4. 2 Denah Lokasi Pengujian *LoRa* (2)

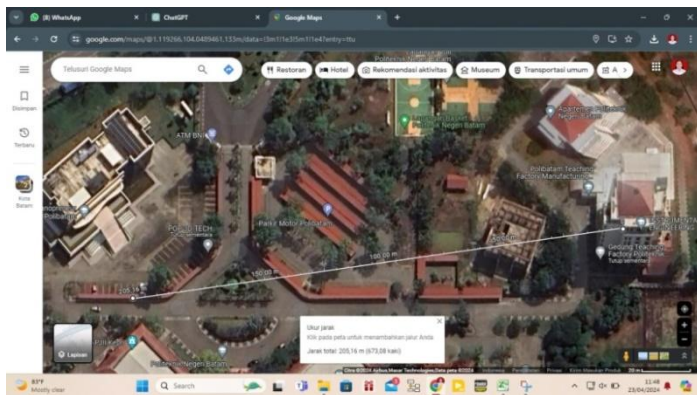
Pada tabel 4.2 dibawah ini merupakan hasil pengujian yang dilakukan dengan mengambil 10 data detak jantung dari 1 pasien tanpa riwayat hipertensi. Pengambilan data detak jantung sebanyak 10 kali bertujuan untuk memperoleh nilai rata-rata dari pengukuran yang dihasilkan oleh sensor. Hasil pengukuran dari sensor kemudian dibandingkan dengan hasil pengukuran dari alat pembanding,

yaitu *pulse* meter, yang berfungsi sebagai tolak ukur untuk hasil penghitungan dari sensor yang digunakan. Hal ini dilakukan untuk menentukan seberapa besar kesalahan atau gangguan (*noise*) yang terjadi saat pembacaan dengan sensor. Berdasarkan pengujian yang telah dilakukan, rata-rata % error pengukuran detak jantung menggunakan sensor MAX30102 dan alat pembanding yaitu pulse sensor adalah 1,9%, dengan nilai kesalahan tertinggi sebesar 4,8% dan terendah sebesar 0%.

Tabel 4. 2 Data Sensor MAX30102 (2)

Lokasi : RTF LT 7 Ke Depan GU									
Jarak TX Ke RX : 121,81 m (399,64 kaki)									
TRANSMITTER					RECAIVER				
No.	Time	Sensor MAX30102 (BPM)	Pulse Meter (BPM)	% ERROR	No.	Time	Sensor MAX30102 (BPM)		
PERCOBAAN KE.	1	16:41:19	87	83	4,8	PERCOBAAN KE.	1	16:44:25	87
	2	16:42:45	85	83	2,4		2	16:45:52	85
	3	16:47:43	93	92	1,1		3	16:47:49	93
	4	16:47:45	91	92	1,1		4	16:47:52	91
	5	16:48:11	98	95	3,2		5	16:48:17	98
	6	16:51:56	99	98	1,0		6	16:52:03	99
	7	16:52:50	74	75	1,3		7	16:52:56	74
	8	16:52:52	76	76	0,0		8	16:52:58	76
	9	16:52:54	78	76	2,6		9	16:53:01	78
	10	16:52:57	80	79	1,3		10	16:53:03	80
MIN		74	75	0,0					
MAX		99	98	4,8					
AVG				1,9					

Pada gambar 4.3 ini merupakan denah lokasi dan jarak pengujian *LoRa transmitter* dan *receiver* dari Gedung RTF ke Techno dengan jarak jauh 205,16 m (673,08 kaki).



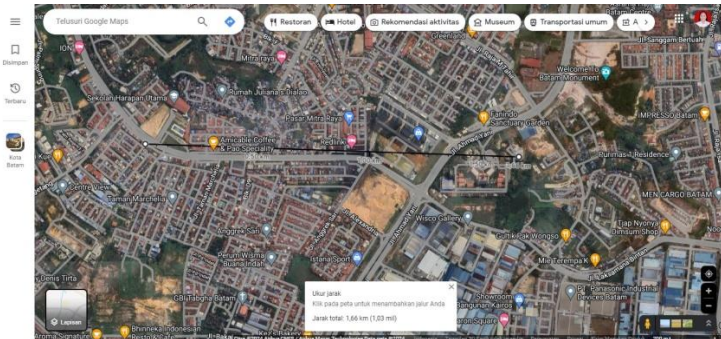
Gambar 4. 3 Denah Lokasi Pengujian LoRa (3)

Pada tabel 4.3 dibawah ini merupakan hasil pengujian yang dilakukan dengan dengan mengambil 10 data detak jantung dari 1 pasien tanpa riwayat hipertensi. Pengambilan data detak jantung sebanyak 10 kali bertujuan untuk memperoleh nilai rata-rata dari pengukuran yang dihasilkan oleh sensor. Hasil pengukuran dari sensor kemudian dibandingkan dengan hasil pengukuran dari alat pembanding, yaitu *pulse* meter, yang berfungsi sebagai tolak ukur untuk hasil penghitungan dari sensor yang digunakan. Hal ini dilakukan untuk menentukan seberapa besar kesalahan atau gangguan (*noise*) yang terjadi saat pembacaan dengan sensor. Berdasarkan pengujian yang telah dilakukan, rata-rata % error pengukuran detak jantung menggunakan sensor MAX30102 dan alat pembanding yaitu pulse sensor adalah 2,1%, dengan nilai kesalahan tertinggi sebesar 4,6% dan terendah sebesar 0%.

Tabel 4. 3 Data Sensor MAX30102 (3)

Lokasi : RIF LT 7 Ke Depan Techno									
Jarak TX Ke RX : 205,16 m (673,08 kaki)									
TRANSMITTER					RECAIVER				
No.	Time	Sensor MAX30102 (BPM)	Pulse Meter (BPM)	% ERROR	No.	Time	Sensor MAX30102 (BPM)		
PERCOBAAN KE-	1	16:55:12	86	85	1,2	PERCOBAAN KE-	1	16:55:19	86
	2	16:55:15	87	87	0,0		2	16:55:21	87
	3	16:55:17	87	86	1,2		3	16:55:23	87
	4	16:55:19	99	98	1,0		4	16:55:26	99
	5	17:12:33	107	105	1,9		5	17:02:50	107
	6	17:03:51	103	105	1,9		6	17:03:58	103
	7	17:05:33	91	87	4,6		7	17:05:40	91
	8	17:05:35	94	90	4,4		8	17:05:42	94
	9	17:05:38	92	90	2,2		9	17:05:45	92
	10	17:09:10	101	98	3,1		10	17:09:17	101
MIN		86	85	0,0					
MAX		107	105	4,6					
AVG				2,1					

Pada gambar 4.4 ini merupakan denah lokasi dan jarak pengujian LoRa transmmiter dan receiver yang dilakukan dengan mengambil data nilai detak jantung pada jarak maksimal komunikasi LoRa yaitu 1 kilometer. Pengiriman data dari LoRa pengirim dan penerima sangat bergantung pada kondisi lingkungan yaitu jarak dan penghalang.



Gambar 4. 4 Denah Lokasi Pengujian LoRa (4)

Pada tabel 4.4 dibawah ini merupakan hasil pengujian yang dilakukan dengan mengambil 10 data detak jantung dari 1 pasien tanpa riwayat hipertensi. Pengambilan data detak jantung sebanyak 10 kali bertujuan untuk memperoleh nilai rata-rata dari pengukuran yang dihasilkan oleh sensor. Hasil pengukuran dari sensor kemudian dibandingkan dengan hasil pengukuran dari alat pembanding, yaitu pulse meter, yang berfungsi sebagai tolak ukur untuk hasil penghitungan dari sensor yang digunakan. Hal ini dilakukan untuk menentukan seberapa besar kesalahan atau gangguan (noise) yang terjadi saat pembacaan dengan sensor. Berdasarkan pengujian yang telah dilakukan, rata-rata % error pengukuran detak jantung menggunakan sensor MAX30102 dan alat pembanding yaitu pulse sensor adalah 1,9%, dengan nilai kesalahan tertinggi sebesar 3,8% dan terendah sebesar 0%.

Tabel 4. 4 Data Sensor MAX30102 (4)

Lokasi : RIF LI 7 Ke Sekolah Harapan Utama									
Jarak TX Ke RX : 1,66 km (1,03 mil)									
TRANSMITTER					RECAIVER				
No.	Time	Sensor MAX30102	Pulse Meter	% ERROR	No.	Time	Sensor MAX30102		
PERCOBAAN KE-	1	16:20:09	77	80	3,8	PERCOBAAN KE-	1	16:22:07	77
	2	16:21:31	76	74	2,7		2	16:23:29	76
	3	16:22:14	78	78	0,0		3	16:24:13	78
	4	16:23:05	77	79	2,5		4	16:25:04	77
	5	16:23:43	73	72	1,4		5	16:25:41	73
	6	16:24:25	73	75	2,7		6	16:26:23	73
	7	16:25:08	66	66	0,0		7	16:27:07	66
	8	16:25:44	69	69	0,0		8	16:27:43	69
	9	16:26:19	75	73	2,7		9	16:28:17	75
	10	16:27:06	73	71	2,8		10	16:29:05	73
MIN		66	66	0,0					
MAX		78	80	3,8					
AVG				1,9					

Dari data yang sudah didapatkan pada beberapa tabel diatas merupakan pengujian komunikasi *LoRa* untuk mengirimkan data detak jantung manusia menggunakan sensor MAX30102. Data ini diambil dengan beberapa titik lokasi yang berbeda dengan jarak yang juga berbeda, dan dari hasil pengujian komunikasi *LoRa* ini dapat disimpulkan semakin jauh jarak antara *LoRa* pengirim dan penerima dan semakin banyak penghalang menyebabkan *delay* dalam pengiriman data paling lama 2 menit. Untuk pasien yang dijadikan sebagai percobaan pengambilan data detak jantung merupakan 1 orang yang sama dengan status kesehatan jantung yang normal tanpa riwayat hipertensi.

4.1.2 Data Hasil Pengujian Sensor MPX5700DP

Pada tabel 4.5 dibawah ini merupakan hasil pengujian yang dilakukan dengan dengan mengambil 5 data tekanan darah dari 1 pasien tanpa riwayat hipertensi. Pengambilan data tekanan darah sebanyak 5 kali bertujuan untuk memperoleh nilai rata-rata dari pengukuran yang dihasilkan oleh sensor. Hasil pengukuran dari sensor kemudian dibandingkan dengan hasil pengukuran dari alat pembanding, yaitu tensimeter digital, yang berfungsi sebagai tolak ukur untuk hasil penghitungan dari sensor yang digunakan. Hal ini dilakukan untuk menentukan seberapa besar kesalahan atau gangguan (*noise*) yang terjadi saat pembacaan dengan sensor. Berdasarkan pengujian yang telah dilakukan, rata-rata % error pengukuran tekanan darah menggunakan sensor MPX5700DP dan alat pembanding yaitu tensimeter digital adalah 1,9% untuk sistol dan 3,2% untuk diastol, dengan nilai kesalahan tertinggi sebesar 4,5% dan terendah sebesar 1,0%.

Tabel 4. 5 Data Sensor MPX5700DP (1)

		Lokasi : RTF LI 7 Ke LI Dasar RTF									
		Jarak TX Ke RX : 51,45 m (168,80 kaki)									
		TRANSMITTER					TRANSMITTER				
PERCOBAAN KE -	NO.	Time	SENSOR MPX5700DP	TENSIMETER DIGITAL	%ERROR SISTOL	PERCOBAAN KE -	NO.	Time	SENSOR MPX5700DP	TENSIMETER DIGITAL	%ERROR DIASTOL
			SISTOL (mmHg)	SISTOL (mmHg)					DIASTOL (mmHg)	DIASTOL (mmHg)	
	1	17:26:00	97	96	1,0		1	17:26:00	69	67	3,0
	2	17:28:35	100	98	2,0		2	17:28:35	72	74	2,7
	3	17:33:00	93	91	2,2		3	17:33:00	69	72	4,2
	4	17:34:48	100	102	2,0		4	17:34:48	72	73	1,4
	5	17:36:37	93	95	2,1		5	17:36:37	69	66	4,5
	MIN		93	91	1,0		MIN		69	66	1,4
	MAX		100	102	2,2		MAX		72	74	4,5
	AVG				1,9		AVG				3,2

RECAIVER				
PERCOBAAN KE -	NO.	Time	SENSOR MPX5700DP	
			SISTOL (mmHg)	DIASTOL (mmHg)
	1	17:26:00	97	69
2	17:28:35	100	72	
3	17:33:00	93	69	
4	17:34:48	100	72	
5	17:36:37	93	69	

Pada tabel 4.6 dibawah ini merupakan hasil pengujian yang dilakukan dengan dengan mengambil 5 data tekanan darah dari 1 pasien tanpa riwayat hipertensi. Pengambilan data tekanan darah sebanyak 5 kali bertujuan untuk memperoleh nilai rata-rata dari pengukuran yang dihasilkan oleh sensor. Hasil pengukuran dari sensor kemudian dibandingkan dengan hasil pengukuran dari alat pembanding, yaitu tensimeter digital, yang berfungsi sebagai tolak ukur untuk hasil penghitungan dari sensor yang digunakan. Hal ini dilakukan untuk menentukan seberapa besar kesalahan atau gangguan (*noise*) yang terjadi saat pembacaan dengan sensor. Berdasarkan pengujian yang telah dilakukan, rata-rata % error pengukuran tekanan darah menggunakan sensor MPX5700DP dan alat pembanding yaitu tensimeter digital adalah 3,0% untuk sistol dan 3,2% untuk diastol, dengan nilai kesalahan tertinggi sebesar 4,8% dan terendah sebesar 0%.

Tabel 4. 6 Data Sensor MPX5700DP (2)

Lokasi : RTF LT 7 Ke LT Dasar RTF					Jarak TX Ke RX : 51,45 m (168,80 kaki)						
TRANSMITTER					TRANSMITTER						
PERCOBAAN KE -	NO.	Time	SENSOR MPX5700DP	TENSIMETER DIGITAL	%ERROR SISTOL	PERCOBAAN KE -	NO.	Time	SENSOR MPX5700DP	TENSIMETER DIGITAL	%ERROR DIASTOL
			SISTOL (mmHg)	SISTOL (mmHg)					DIASTOL (mmHg)	DIASTOL (mmHg)	
	1	17:39:09	100	98	2,0		1	17:39:09	73	76	3,9
2	17:42:00	92	94	2,1	2	17:42:00	67	67	0,0		
3	17:43:11	109	104	4,8	3	17:43:11	69	67	3,0		
4	17:44:25	99	97	2,1	4	17:44:25	61	64	4,7		
5	17:46:12	93	97	4,1	5	17:46:12	69	66	4,5		
MIN		92	94	2,0	MIN		61	64	0,0		
MAX		109	104	4,8	MAX		73	76	4,7		
AVG		3,0			AVG		3,2				

RECAIVER				
PERCOBAAN KE -	NO.	Time	SENSOR MPX5700DP	
			SISTOL (mmHg)	DIASTOL (mmHg)
	1	17:39:09	100	73
2	17:42:00	92	67	
3	17:43:11	109	69	
4	17:44:25	99	61	
5	17:46:12	93	69	

Pada tabel 4.7 dibawah ini merupakan hasil pengujian yang dilakukan dengan dengan mengambil 5 data tekanan darah dari 1 pasien tanpa riwayat hipertensi. Pengambilan data tekanan darah sebanyak 5 kali bertujuan untuk memperoleh nilai rata-rata dari pengukuran yang dihasilkan oleh sensor. Hasil pengukuran dari sensor kemudian dibandingkan dengan hasil pengukuran dari alat pembanding, yaitu tensimeter digital, yang berfungsi sebagai tolak ukur untuk hasil penghitungan dari sensor yang digunakan. Hal ini dilakukan untuk menentukan seberapa besar kesalahan atau gangguan (noise) yang terjadi saat pembacaan dengan sensor. Berdasarkan pengujian yang telah dilakukan, rata-rata % error pengukuran tekanan darah menggunakan sensor MPX5700DP dan alat pembanding yaitu tensimeter digital adalah 2,2% untuk sistol dan 3,0% untuk diastol, dengan nilai kesalahan tertinggi sebesar 4,3% dan terendah sebesar 1,0%.

Tabel 4. 7 Data Sensor MPX5700DP (3)

Lokasi : RT/FLT 7 Ke Sekolah Harapan Utama					
Jarak TX Ke RX : 1,66 km (1,03 mil)					
TRANSMITTER					
PERCOBAAN KE -	NO.	Time	SENSOR MPX5700DP	TENSIMETER DIGITAL	%ERROR SISTOL
			SISTOL (mmHg)	SISTOL (mmHg)	
			1	19:34:21	
2	19:35:36	95	96	1,0	
3	19:37:44	94	93	1,1	
4	19:39:03	97	95	2,1	
5	19:42:24	105	102	2,9	
MIN			94	93	1,0
MAX			105	102	3,9
AVG			2,2		

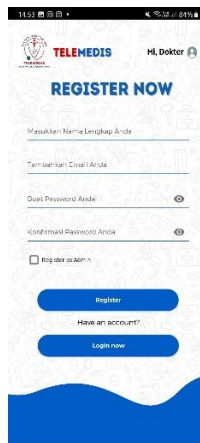
TRANSMITTER					
PERCOBAAN KE -	NO.	Time	SENSOR MPX5700DP	TENSIMETER DIGITAL	%ERROR DIASTOL
			DIASTOL (mmHg)	DIASTOL (mmHg)	
			1	19:34:21	
2	19:35:36	67	70	4,3	
3	19:37:44	75	72	4,2	
4	19:39:03	72	75	4,0	
5	19:42:24	68	69	1,4	
MIN			67	69	1,3
MAX			77	78	4,3
AVG			3,0		

RECAIVER				
PERCOBAAN KE -	NO.	Time	SENSOR MPX5700DP	
			SISTOL (mmHg)	DIASTOL (mmHg)
			1	19:35:21
2	19:36:37	95	67	
3	19:38:49	94	75	
4	19:40:04	97	72	
5	19:43:25	105	68	

Dari data yang sudah didapatkan pada beberapa tabel diatas merupakan pengujian komunikasi LoRa untuk mengirimkan data tekanan darah manusia menggunakan sensor MPX5700DP. Data ini diambil dengan beberapa titik lokasi yang berbeda dengan jarak yang juga berbeda. Untuk pasien yang dijadikan sebagai percobaan pengambilan data detak jantung merupakan 1 orang yang sama dengan status kesehatan jantung yang normal tanpa riwayat hipetensi.

4.1.3 Pengujian Aplikasi Android

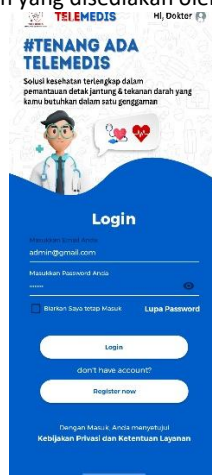
Pembacaan data nilai detak jantung dan tekanan darah dilakukan menggunakan aplikasi Android yang telah dibuat menggunakan software Android Studio atau kode yang telah diintegrasikan dengan Firebase agar dapat berkomunikasi dengan server. Pada aplikasi terdapat beberapa tampilan seperti registrasi, log-in, dashborard data sensor, history dan informasi mengenai kesehatan jantung manusia. Pada gambar 4.5 merupakan tampilan tab register pada aplikasi telemedis yang dirancang sebagai pintu masuk bagi pengguna baru untuk membuat akun pribadi. Fitur-fitur seperti pengisian nama lengkap, email, dan pembuatan kata sandi yang kuat menjadi dasar pembuatan akun. Opsi "Register as Admin" umumnya ditujukan untuk tenaga medis atau pengelola sistem yang memiliki akses penuh. Setelah proses pendaftaran selesai, pengguna dapat mengakses berbagai layanan kesehatan secara online, mulai dari konsultasi dengan dokter hingga melihat riwayat kesehatan. Tujuan utama tab register adalah untuk memfasilitasi pembuatan akun baru, memverifikasi identitas pengguna, dan mengatur preferensi akun awal.



Gambar 4. 5 Tampilan Register

Pada gambar 4.6 merupakan tampilan fitur login pada aplikasi telemedis ini berfungsi sebagai pintu masuk bagi pengguna yang sudah memiliki akun untuk mengakses berbagai layanan kesehatan yang disediakan. Dengan memasukkan alamat email dan kata sandi yang telah didaftarkan sebelumnya pada tab registrasi, pengguna dapat masuk ke dalam akun mereka. Setelah berhasil login, pengguna dapat melakukan berbagai aktivitas seperti berkonsultasi dengan dokter secara online, memantau kondisi kesehatan, mengakses riwayat kesehatan, dan mendapatkan informasi kesehatan lainnya. Fitur "Biarkan Saya

tetap Masuk" memungkinkan pengguna untuk tetap login secara otomatis tanpa perlu memasukkan ulang kata sandi setiap kali membuka aplikasi. Sementara itu, fitur "Lupa Password" disediakan untuk pengguna yang lupa dengan kata sandi mereka dan ingin meresetnya. Secara singkat, fitur login ini berperan sebagai autentikasi untuk memastikan bahwa hanya pengguna yang berhak yang dapat mengakses layanan kesehatan yang disediakan oleh aplikasi telemedis tersebut.



Gambar 4. 6 Tampilan Login

Pada gambar 4.7 merupakan tampilan dashboard hasil pengukuran pada aplikasi kesehatan ini berfungsi sebagai pusat informasi bagi pengguna untuk melihat hasil pengukuran kesehatan mereka secara real-time. Dalam gambar yang Anda berikan, ditampilkan hasil pengukuran tekanan darah dan detak jantung.

- **Tekanan darah:** Menampilkan hasil pengukuran tekanan darah sistolik dan diastolik, beserta status normal atau tidak normalnya berdasarkan rentang nilai yang umum diterima.
- **Detak jantung:** Menampilkan hasil pengukuran detak jantung per menit (BPM) dan status normal atau tidak normalnya.
- **Info:** Kemungkinan besar fitur ini akan memberikan informasi lebih lanjut mengenai hasil pengukuran, seperti penjelasan tentang apa itu tekanan darah dan detak jantung, serta faktor-faktor yang dapat memengaruhinya.
- **Histori:** Fitur ini memungkinkan pengguna untuk melihat riwayat hasil pengukuran sebelumnya, sehingga dapat memantau perkembangan kondisi kesehatan mereka dari waktu ke waktu.
- **Logout:** Fitur ini digunakan oleh pengguna untuk keluar dari akun mereka.



Gambar 4. 7 Tampilan Dashboard Hasil Pengukuran

Pada gambar 4.8 merupakan tampilan dashboard masing-masing user yang berfungsi sebagai pusat informasi bagi pengguna untuk melihat hasil pengukuran kesehatan mereka secara real-time. Dalam gambar yang Anda berikan, ditampilkan hasil pengukuran tekanan darah dan detak jantung.

- **BPM (Detak Jantung):** Menampilkan jumlah detak jantung per menit.
- **Sistolik dan Diastolik:** Menampilkan hasil pengukuran tekanan darah sistolik (saat jantung berkontraksi) dan diastolik (saat jantung berelaksasi).
- **Simpan Data:** Tombol ini kemungkinan digunakan untuk menyimpan data pengukuran terbaru ke dalam riwayat data pengguna.
- **Histori Data Pengguna:** Fitur ini menampilkan riwayat hasil pengukuran sebelumnya, sehingga pengguna dapat melacak perubahan kondisi kesehatannya dari waktu ke waktu.



Gambar 4. 8 Tampilan Dashboard masing-masing user

Pada gambar 4.9 merupakan tampilan tab informasi yang berfungsi memberikan gambaran umum tentang kesehatan jantung dan pembuluh darah Anda berdasarkan dua indikator utama, yaitu detak jantung dan tekanan darah. Detak jantung menunjukkan seberapa cepat jantung Anda berdetak, sedangkan tekanan darah mengukur kekuatan darah mendorong dinding arteri. Angka-angka yang tertera pada tampilan umumnya dibandingkan dengan rentang nilai normal untuk menentukan apakah kondisi Anda termasuk dalam kategori sehat, terlalu cepat, terlalu lambat, atau mengalami tekanan darah tinggi atau rendah. Namun, perlu diingat bahwa ini hanya gambaran awal. Untuk mendapatkan diagnosis dan penanganan yang tepat, selalu konsultasikan dengan dokter.



Gambar 4. 9 Tampilan Tab Informasi

4.2 Pembahasan

Dapat dilihat dari tabel 4.1 sampai 4.7 terdapat beberapa data yang diperoleh dari pengujian sensor yang membaca nilai detak jantung dan tekanan darah. Ada beberapa data dari hasil pengujian sensor yang tidak sesuai dengan alat ukur pembanding, namun di sisi lain terdapat pula data pengujian yang sesuai dengan alat ukur pembanding. Hal ini mungkin disebabkan oleh masalah atau kesalahan saat sensor membaca hasil pengujian atau mungkin juga karena kondisi pasien yang kondisinya tidak stabil yang menyebabkan pembacaan sensor menjadi tidak akurat. Dari tabel 4.1 sampai 4.7 merupakan beberapa pengukuran detak jantung dan tekanan darah data ini didapat dari sensor MAX30102 dan MPX5700DP yang terhubung ke *transmitter* (pengirim).

Pada pengujian sensor MAX30102 untuk mengukur detak jantung manusia. Pengambilan data dilakukan sebanyak 10 kali dalam 1 kali percobaan dengan orang yang sama. Dalam pengambilan data detak jantung digunakan *pulsemeter* sebagai alat pembanding untuk membandingkan nilai sensor dengan alat komersil, untuk memastikan apakah sensor sudah akurat atau belum dengan cara mencari % error dengan membandingkan nilai sensor dan pembanding. Dan dari beberapa data detak jantung yang sudah diambil didapatkan rata-rata persentase error nya sebesar 2,2%.

Selanjutnya adalah pengujian sensor MPX5700DP untuk mengukur tekanan darah manusia. Pengambilan data dilakukan 5 kali dalam 1 kali percobaan dengan orang yang sama. Dalam pengambilan data tekanan darah digunakan tensimeter

digital sebagai alat pembanding untuk membandingkan nilai sensor dengan alat komersial yang sudah sesuai standar, untuk memastikan apakah sensor sudah akurat atau belum dengan cara mencari % error dengan membandingkan nilai sensor dan pembanding. Dan dari beberapa data tekanan darah yang sudah diambil didapatkan rata-rata persentase error sistoliknya sebesar 2,4% dan persentase error diastoliknya sebesar 3,1%.

Hasil menunjukkan bahwa sensor detak jantung MAX30102 memberikan hasil yang belum konsisten pada beberapa pengujian, dikarenakan terdapat perbedaan kecil antara nilai yang diukur oleh sensor dengan nilai yang diukur oleh perangkat komersial *Pulse Oximeter*, yang menandakan perlunya investigasi lebih lanjut terkait penyebab perbedaan tersebut. Sementara itu, sensor tekanan darah MPX5700DP juga menunjukkan beberapa perbedaan hasil dalam pengukuran tekanan darah dengan alat pembanding tensimeter digital, yang menandakan perlunya investigasi lebih lanjut terkait penyebab perbedaan tersebut.

Lebih lanjut, integrasi kedua sensor dengan teknologi komunikasi *LoRa* memiliki potensi besar untuk meningkatkan pemantauan kesehatan jantung secara *real-time*. Dengan mentransmisikan data detak jantung dan tekanan darah secara nirkabel, sistem ini dapat memberikan manfaat yang signifikan dalam pemantauan pasien jantung dari jarak jauh, termasuk pemantauan kondisi jantung secara kontinu, deteksi dini terhadap masalah kesehatan, dan intervensi cepat jika diperlukan. Dengan demikian, hasil pembahasan ini menyoroti potensi besar sistem pemantauan detak jantung dan tekanan darah berbasis IoT dalam meningkatkan kualitas pemantauan kesehatan jantung dan memberikan manfaat yang nyata bagi pasien dan penyedia layanan kesehatan.

Bab 5. Kesimpulan dan Saran

5.1 Kesimpulan

1. Rancangan Alat:

Penelitian ini berhasil merancang alat pemantauan detak jantung dan tekanan darah yang efisien menggunakan sensor MAX30102 dan MPX5700DP yang terhubung melalui jaringan *LoRaWAN*. Hasil pengujian menunjukkan bahwa alat ini dapat mentransmisikan data detak jantung dan tekanan darah secara nirkabel dengan performa yang baik.

2. Integrasi dengan Aplikasi:

Alat pemantauan berhasil diintegrasikan dengan aplikasi *Android* dan *platform web Firebase*, yang memungkinkan tenaga medis untuk memantau kondisi pasien secara *real-time* dari jarak jauh. Aplikasi ini memberikan kemudahan bagi tenaga medis dalam menambah data pasien serta memantau kondisi kesehatan pasien secara efisien.

3. Efektivitas Alat:

Pengujian menunjukkan bahwa alat ini memiliki tingkat akurasi yang tinggi dalam pengukuran detak jantung dan tekanan darah. Data detak jantung yang diambil menggunakan sensor MAX30102 dibandingkan dengan *pulse oximeter* menunjukkan akurasi pengukuran sebesar 97,7% dengan persentase kesalahan sebesar 2,3%. Data tekanan darah yang diambil menggunakan sensor MPX5700DP dibandingkan dengan tensimeter digital menunjukkan akurasi pengukuran sistolik sebesar 97,5% dan diastolik sebesar 96,8%, dengan persentase kesalahan masing-masing sebesar 2,5% dan 3,2%.

4. Pengujian Komunikasi LoRa:

Pada pengujian komunikasi *LoRa* dengan pengiriman data tekanan darah dan detak jantung pada jarak maksimal komunikasi *LoRa* sejauh 1 kilometer, pengiriman data paling lama yaitu 2 menit. Hal ini menunjukkan bahwa semakin jauh dan semakin banyak penghalang, maka menyebabkan delay dalam pengiriman data.

Dengan demikian, alat pemantauan detak jantung dan tekanan darah berbasis *IoT* dengan teknologi *LoRa* ini efektif dan dapat diandalkan dalam memberikan informasi kesehatan yang penting bagi tenaga medis dan pasien. Selain itu, alat ini berpotensi meningkatkan deteksi dini dan pengobatan kondisi kardiovaskular, memberikan akses yang lebih baik bagi pasien dan tenaga medis, serta mendukung pemantauan kesehatan jantung secara kontinu. Implementasi alat ini dapat membantu dalam deteksi dini terhadap masalah kesehatan kardiovaskular, yang merupakan salah satu penyebab utama kematian di dunia, dan memberikan manfaat signifikan dalam pemantauan pasien jantung dari jarak jauh.

5.2 Saran

Adapun saran yang ingin kami sampaikan adalah sebagai berikut:

- a. Kami sangat mengharapkan masukan yang membangun dari pembaca dan praktisi kesehatan untuk meningkatkan efektivitas dan efisiensi alat pemantauan kesehatan ini.
- b. Kami menyarankan penelitian lebih lanjut untuk melakukan uji presisi dan akurasi pada hasil pengukuran sensor, termasuk pengujian sensor terhadap kondisi kesehatan yang beragam.
- c. Kami berharap agar seluruh tim proyek, termasuk pengembang, peneliti, dan praktisi kesehatan, dapat lebih memperhatikan peran dan tanggung jawab mereka masing-masing.

Daftar Pustaka

- [1] Putriana, "Pemberian Pemahaman Mengenai Sediaan Herbal Yang Berfungsi Untuk Pemeliharaan Kesehatan Jantung Dan Ginjal," *Department of Pharmaceutics and Pharmaceutical Technology, faculty of pharmacy Unpad*, 2019.
- [2] N. I. S. Lilik and I. Budiono, "Risiko Kematian Pasien Gagal Jantung Kongestif (GJK): *Studi Kohort Retrospektif* Berbasis Rumah Sakit," *Indones. J. Public Heal. Nutr.*, vol. 1, no. 3, pp. 388–395, 2021.
- [3] Siloam Hospital, "Detak Jantung Normal Manusia," URL:<https://www.siloamhospitals.com/informasi-siloam/artikel/detak-jantung-normal>, 2022. Diakses tanggal 9 November 2023.
- [4] Augustin, "Jaringan Jarak Jauh & Berdaya Rendah untuk *Internet of Things*," Politeknik École Route de Saclay, 91128 Palaiseau, Prancis, 2016.
- [5] RS Soewandhi Surabaya, "Jenis, Gejala, dan Penyebab Penyakit Jantung," URL:<https://rs-soewandhi.surabaya.go.id/jenis-gejala-dan-penyebab-penyakit-jantung/>, 2021. Diakses tanggal 11 November 2023.
- [6] L. Aditya and D. Riska, "Rancang Bangun Alat Pengukur Kadar Oksigen Non Invasive Menggunakan Sensor Max30100," <https://Medium.Com/>, vol. 4, no. 3, pp. 248–253, 2020.
- [7] M. E. Gumilang and W. Sugeng, "Implementasi Alat Pendeteksi Detak Jantung Berbasis *Raspberry Pi*," vol. 2, no. 1, pp. 28–29, 2016.
- [8] Fadhullah, Muhamad Irfan, "Pengukuran Dan Analisis Efek Interferensi Sinyal *Global System For Mobile Communication 915 Mhz* Terhadap Long Range (LoRa) 920-923 Mhz," Fadhullah, Telkom University, 2020.
- [9] I. R. Sofiani, R. Kharisma, and L. Syafa'ah, "Sistem Monitoring *Heart Rate* dan Oksigen Dalam Darah Berbasis LoRa," *Med. Tek. J. Tek. Elektromedik Indones.*, vol. 2, no. 2, 2021.
- [10] Sofiani, R. Kharisma, and L. Syafa'ah, "Sistem Monitoring *Heart Rate* dan Oksigen Dalam Darah Berbasis LoRa," *Med. Tek. J. Tek. Elektromedik Indones.*, vol. 2, no. 2, 2021.
- [11] P. Alat, U. Tekanan, D. Dan, J. Denyut, J. Berbasis, and A. Uno, "Perancangan Alat ukur Tekanan Darah dan Jumlah Denyut Jantung Berbasis *Arduino Uno*," *Jur. Tek. Elektro Univ. Mataram*, vol. 3, no. 1, pp. 1–12, 2019.

Biodata



Nama : Rudcayanti Hutagaol
TTL : Hutagaol, 07 Januari 2004
Agama : Kristen Protestan
Alamat : Perum Legenda Malaka blok G4
Email : rudcayantihuatagol@gmail.com
Riwayat Pendidikan SMA/SMK : SMA N 1 Siantar
Narumonda
SMP : SMP Negeri 1 Sigumpar

Lampiran

Lampiran 1. Coding Pengujian *LoRa* Untuk Mendapatkan Nilai *RSSI*

Transmitter

```
#include <SPI.h>
#include <LoRa.h>

// LoRa <-> Pro Mini Interface
#define LORA_RAY_NSS 10
#define LORA_RAY_RST 9
#define LORA_RAY_DIO0 2
#define LORA_RAY_DIO1 6
#define BAND 433E6

#define LORA_TX_POWER 20
#define LORA_SPREADING_FACTOR 12

int counter = 0;

void setup() {
  //initialize Serial Monitor
  Serial.begin(115200);
  while (!Serial);
  Serial.println("LoRa Sender");

  //setup LoRa transceiver module
  LoRa.setPins(LORA_RAY_NSS, LORA_RAY_RST, LORA_RAY_DIO0);

  //replace the LoRa.begin(---E-) argument with your location's frequency
  //Indonesia 920.0 - 923.0 MHz
  if (!LoRa.begin(BAND)) {
    Serial.println("Starting LoRa failed!");
    while (1);
  }

  LoRa.setTxPower(LORA_TX_POWER, PA_OUTPUT_PA_BOOST_PIN);
  LoRa.setSpreadingFactor(LORA_SPREADING_FACTOR);

  // Change sync word (0xF3) to match the receiver
```

```

// The sync word assures you don't get LoRa messages from other LoRa
transceivers
// ranges from 0-0xFF
LoRa.setSyncWord(0xF3);
Serial.println("LoRa Initializing OK!");
}

void loop() {
Serial.print("Sending packet: ");
Serial.println(counter);

//Send LoRa packet to receiver
LoRa.beginPacket();
LoRa.print("hello ");
LoRa.print(counter);
LoRa.endPacket();

counter++;

delay(10000);
}

```

Receiver

```

#include <SPI.h>
#include <LoRa.h>

//define the pins used by the transceiver module
#define LORA_AURORA_V2_NSS 15
#define LORA_AURORA_V2_RST 0
#define LORA_AURORA_V2_DIO0 27
#define LORA_AURORA_V2_EN 32

#define LORA_TX_POWER 20
#define LORA_SPREADING_FACTOR 12

void setup() {
// Initiate the LoRa Enable pin
pinMode(LORA_AURORA_V2_EN, OUTPUT);
// LoRa chip is Active High
digitalWrite(LORA_AURORA_V2_EN, HIGH);

//initialize Serial Monitor

```

```

Serial.begin(115200);
while (!Serial);
Serial.println("LoRa Receiver");

//setup LoRa transceiver module
LoRa.setPins(LORA_AURORA_V2_NSS,LORA_AURORA_V2_RST,
LORA_AURORA_V2_DIO0);

//replace the LoRa.begin(---E-) argument with your location's frequency
//433E6 for Asia
//866E6 for Europe
//915E6 for North America
while (!LoRa.begin(433E6)) {
  Serial.println(".");
  delay(500);
}

LoRa.setTxPower(LORA_TX_POWER, PA_OUTPUT_PA_BOOST_PIN);
LoRa.setSpreadingFactor(LORA_SPREADING_FACTOR);

// Change sync word (0xF3) to match the receiver
// The sync word assures you don't get LoRa messages from other LoRa
transceivers
// ranges from 0-0xFF
LoRa.setSyncWord(0xF3);
Serial.println("LoRa Initializing OK!");
}

void loop() {
  // try to parse packet
  int packetSize = LoRa.parsePacket();
  if (packetSize) {
    // // received a packet
    // Serial.print("Received packet ");

    // read packet
    while (LoRa.available()) {
      String LoRaData = LoRa.readString();
      Serial.print(LoRaData);
    }

    // // print RSSI of packet

```

```

    Serial.print(" with RSSI ");
    Serial.println(LoRa.packetRssi());
  }
}

```

Lampiran 2. Coding Sensor MAX30102 Untuk Mengukur Detak Jantung

```

#include <MAX3010x.h>
#include "filters.h"
#include <SPI.h>
#include <LoRa.h>

// LoRa <-> Pro Mini Interface
#define LORA_RAY_NSS 10
#define LORA_RAY_RST 9
#define LORA_RAY_DIO0 2
#define LORA_RAY_DIO1 6
#define BAND 433E6

#define LORA_TX_POWER 20
#define LORA_SPREADING_FACTOR 12

int counter = 0;
// Sensor (adjust to your sensor type)
MAX30105 sensor;
const auto kSamplingRate = sensor.SAMPLING_RATE_400SPS;
const float kSamplingFrequency = 400.0;

// Finger Detection Threshold and Cooldown
const unsigned long kFingerThreshold = 10000;
const unsigned int kFingerCooldownMs = 500;

// Edge Detection Threshold (decrease for MAX30100)
const float kEdgeThreshold = -2000.0;

// Filters
const float kLowPassCutoff = 5.0;
const float kHighPassCutoff = 0.5;

// Averaging

```

```

const bool kEnableAveraging = true;
const int kAveragingSamples = 50;
const int kSampleThreshold = 5;

void setup() {
  Serial.begin(115200);
  while (!Serial);
  Serial.println("LoRa Sender");

  //setup LoRa transceiver module
  LoRa.setPins(LORA_RAY_NSS, LORA_RAY_RST, LORA_RAY_DIO0);

  //replace the LoRa.begin(---E-) argument with your location's frequency
  //Indonesia 920.0 - 923.0 MHz
  if (!LoRa.begin(BAND)) {
    Serial.println("Starting LoRa failed!");
    while (1);
  }

  LoRa.setTxPower(LORA_TX_POWER, PA_OUTPUT_PA_BOOST_PIN);
  LoRa.setSpreadingFactor(LORA_SPREADING_FACTOR);

  // Change sync word (0xF3) to match the receiver
  // The sync word assures you don't get LoRa messages from other LoRa
  transceivers
  // ranges from 0-0xFF
  LoRa.setSyncWord(0xF3);
  Serial.println("LoRa Initializing OK!");

  if(sensor.begin() && sensor.setSamplingRate(kSamplingRate)) {
    Serial.println("Sensor initialized");
  }
  else {
    Serial.println("Sensor not found");
    while(1);
  }
}

// Filter Instances
HighPassFilter high_pass_filter(kHighPassCutoff, kSamplingFrequency);
LowPassFilter low_pass_filter(kLowPassCutoff, kSamplingFrequency);
Differentiator differentiator(kSamplingFrequency);

```

```

MovingAverageFilter<kAveragingSamples> averager;

// Timestamp of the last heartbeat
long last_heartbeat = 0;

// Timestamp for finger detection
long finger_timestamp = 0;
bool finger_detected = false;

// Last diff to detect zero crossing
float last_diff = NAN;
bool crossed = false;
long crossed_time = 0;

void loop() {
  // Serial.print("Sending packet: ");
  // Serial.println(counter);

  // //Send LoRa packet to receiver
  // LoRa.beginPacket();
  // LoRa.print("hello ");
  // LoRa.print(counter);
  // LoRa.endPacket();

  // counter++;

  // delay(10000);

  auto sample = sensor.readSample(1000);
  float current_value = sample.red;

  // Detect Finger using raw sensor value
  if(sample.red > kFingerThreshold) {
    if(millis() - finger_timestamp > kFingerCooldownMs) {
      finger_detected = true;
    }
  }
  else {
    // Reset values if the finger is removed
    differentiator.reset();
    averager.reset();
    low_pass_filter.reset();
  }
}

```

```

high_pass_filter.reset();

finger_detected = false;
finger_timestamp = millis();
}

if(finger_detected) {
  current_value = low_pass_filter.process(current_value);
  current_value = high_pass_filter.process(current_value);
  float current_diff = differentiator.process(current_value);

  // Valid values?
  if(!isnan(current_diff) && !isnan(last_diff)) {

    // Detect Heartbeat - Zero-Crossing
    if(last_diff > 0 && current_diff < 0) {
      crossed = true;
      crossed_time = millis();
    }

    if(current_diff > 0) {
      crossed = false;
    }

    // Detect Heartbeat - Falling Edge Threshold
    if(crossed && current_diff < kEdgeThreshold) {
      if(last_heartbeat != 0 && crossed_time - last_heartbeat > 300) {
        // Show Results
        int bpm = 60000/(crossed_time - last_heartbeat);
        if(bpm > 50 && bpm < 250) {
          // Average?
          if(kEnableAveraging) {
            int average_bpm = averager.process(bpm);

            // Show if enough samples have been collected
            if(averager.count() > kSampleThreshold) {

              Serial.print("Heart Rate (avg, bpm): ");
              Serial.println(average_bpm);

              LoRa.beginPacket();
              LoRa.print("Heart Rate (avg, bpm): ");

```

```

        LoRa.print(average_bpm);
        LoRa.endPacket();

        counter++;

        delay(100);
    }

}
else {
    Serial.print("Heart Rate (current, bpm): ");
    Serial.println(bpm);
}
}
}

crossed = false;
last_heartbeat = crossed_time;
}
}

last_diff = current_diff;
}
}

```

Lampiran 3. Coding Sensor MPX5700DP Untuk Mengukur Tekanan Darah

```

#include <SPI.h>
#include <LoRa.h>
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
#include <ArduinoJson.h>

LiquidCrystal_I2C lcd(0x27, 20, 4); // Inisialisasi LCD 20x4

// LoRa <-> Pro Mini Interface
#define LORA_RAY_NSS 10
#define LORA_RAY_RST 9
#define LORA_RAY_DIO0 2
#define BAND 433E6

```

```

#define LORA_TX_POWER 20
#define LORA_SPREADING_FACTOR 12

const int motorPin = 7;
const int solenoidPin = 6;
const int buttonPin = 4;
const int stopButtonPin = 3; // Pin untuk tombol stop
int sistole = 0; // Nilai awal sistole adalah 0
int diastole;
int mark = 0;

// Konstanta kalibrasi
const float VOLTAGE_MIN = 0.0; // Tegangan minimum pada kisaran tekanan
(dalam volt)
const float VOLTAGE_MAX = 5.0; // Tegangan maksimum pada kisaran tekanan
(dalam volt)
const float PRESSURE_MIN = 0; // Tekanan minimum yang sesuai dengan tegangan
minimum (dalam mmHg)
const float PRESSURE_MAX = 200; // Tekanan maksimum yang sesuai dengan
tegangan maksimum (dalam mmHg)

// Fungsi untuk mengonversi tegangan menjadi tekanan dalam mmHg
float voltageToPressure(float voltage) {
    // Menggunakan konversi linier untuk mengonversi tegangan ke tekanan
    float pressure = (voltage - VOLTAGE_MIN) * ((PRESSURE_MAX - PRESSURE_MIN)
/ (VOLTAGE_MAX - VOLTAGE_MIN)) + PRESSURE_MIN;
    return pressure;
}

void setup() {
    lcd.begin(); // Inisialisasi LCD
    lcd.backlight();
    lcd.noCursor();
    Serial.begin(115200);
    while (!Serial);
    Serial.println("LoRa Sender");

    //setup LoRa transceiver module
    LoRa.setPins(LORA_RAY_NSS, LORA_RAY_RST, LORA_RAY_DIO0);

    //replace the LoRa.begin(---E-) argument with your location's frequency

```

```

//Indonesia 920.0 - 923.0 MHz
if (!LoRa.begin(BAND)) {
  Serial.println("Starting LoRa failed!");
  while (1);
}

LoRa.setTxPower(LORA_TX_POWER, PA_OUTPUT_PA_BOOST_PIN);
LoRa.setSpreadingFactor(LORA_SPREADING_FACTOR);

// Change sync word (0xF3) to match the receiver
// The sync word assures you don't get LoRa messages from other LoRa
transceivers
// ranges from 0-0xFF
LoRa.setSyncWord(0xF3);
Serial.println("LoRa Initializing OK!");

pinMode(motorPin, OUTPUT);
pinMode(solenoidPin, OUTPUT);
pinMode(buttonPin, INPUT_PULLUP);
pinMode(stopButtonPin, INPUT_PULLUP); // Mengatur pin tombol stop sebagai
input dengan pull-up internal

lcd.setCursor(0, 0);
lcd.print("Sistolik : ");
lcd.setCursor(0, 1);
lcd.print("Diastolik : ");

Serial.println("Setup selesai.");
}

void loop() {
  Serial.println("Loop dimulai.");

  int buttonState = digitalRead(buttonPin);

  if (buttonState == LOW && mark == 0) {
    mark = 1;
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("Sistolik : ");
    Serial.print("Sistolik : "); // Menambahkan untuk Serial Monitor
    lcd.setCursor(0, 1);

```

```

    lcd.print("Diastolik : ");
    Serial.println("Diastolik : "); // Menambahkan untuk Serial Monitor
    delay(1000);
    digitalWrite(motorPin, HIGH);
    digitalWrite(solenoidPin, HIGH);
}

if (mark > 0) {
    mulai(); // Panggil fungsi untuk melakukan pengukuran tekanan
}

int stopButtonState = digitalRead(stopButtonPin); // Membaca status tombol
stop

if (stopButtonState == LOW) { // Jika tombol stop ditekan
    digitalWrite(motorPin, LOW); // Menghentikan motor memompa manset
    digitalWrite(solenoidPin, LOW); // Menonaktifkan solenoid
    mark = 0; // Mengatur mark kembali ke 0

    // Hitung nilai diastole sesuai dengan rumus yang diberikan
    diastole = sistole * 2 / 3; // Misalnya, menggunakan rumus diastole = 2/3 *
sistole

    // Tampilkan nilai pengukuran sistol dan diastol pada Serial Monitor
    Serial.print("Sistolik: ");
    Serial.print(sistole);
    Serial.println(" mmHg"); // Tampilkan satuan mmHg
    Serial.print("Diastolik: ");
    Serial.print(diastole);
    Serial.println(" mmHg"); // Tampilkan satuan mmHg

    // Kirim data melalui LoRa
    String dataToSend = "{"sistole\":" + String(sistole) + ",\\"diastole\":" +
String(diastole) + "}";
    LoRa.beginPacket();
    LoRa.print(dataToSend);
    LoRa.endPacket();

    delay(100);

    lcd.clear(); // Bersihkan layar LCD sebelum menampilkan teks baru
    lcd.setCursor(0, 0);

```

```

    lcd.print("Sistolik : ");
    lcd.print(sistole); // Pastikan ini dipanggil setelah perhitungan sistole terakhir
    lcd.print(" mmHg");
    lcd.setCursor(0, 1);
    lcd.print("Diastolik : ");
    lcd.print(diastole);
    lcd.print(" mmHg");

    delay(1000); // Penundaan untuk membaca nilai
}
delay(10); // Penundaan kecil untuk stabilitas loop
}

void mulai() {
    Serial.println("Fungsi mulai() dipanggil.");

    int dataadc = analogRead(A0);
    float voltage = (dataadc * 5.0) / 1023.0; // Konversi nilai ADC menjadi tegangan

    int mmhg = (voltage - VOLTAGE_MIN) * ((PRESSURE_MAX - PRESSURE_MIN) /
(VOLTAGE_MAX - VOLTAGE_MIN)) + PRESSURE_MIN; // Konversi tegangan
menjadi tekanan dalam mmHg

    if (mark == 1) { // Jika tombol start ditekan
        Serial.println("Pantau Tekanan");

        lcd.setCursor(11, 0);
        lcd.print(sistole); // Tampilkan nilai sistole pada layar LCD
        lcd.print(" mmHg"); // Tampilkan satuan mmHg
        lcd.setCursor(11, 1);
        lcd.print(diastole);
        lcd.print(" mmHg"); // Tampilkan satuan mmHg

        // Tambahkan penundaan kecil di sini untuk mengurangi kecepatan
peningkatan nilai sistolik
        delay(800);

        sistole++; // Tambahkan nilai sistole
        if (sistole >= 120) { // Jika sistole mencapai 120 mmHg, hentikan pompa
            digitalWrite(motorPin, LOW); // Matikan pompa
            digitalWrite(solenoidPin, LOW); // Tutup valve
            mark = 0; // Reset mark
        }
    }
}

```

```

        Serial.println("Sistolik mencapai 120 mmHg, pengukuran selesai.");
        return;
    }
}
}

```

Lampiran 4. Coding Sensor MPX5700DP Untuk Mengukur Tekanan Darah

```

#include <SPI.h>
#include <LoRa.h>
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
#include <ArduinoJson.h>
#include <MAX3010x.h>
#include "filters.h"

// Pin Definitions
LiquidCrystal_I2C lcd(0x27, 20, 4);
#define LORA_RAY_NSS 10
#define LORA_RAY_RST 9
#define LORA_RAY_DIO0 2
#define BAND 433E6
#define LORA_TX_POWER 20
#define LORA_SPREADING_FACTOR 12

const int motorPin = 7;
const int solenoidPin = 6;
const int buttonPin = 4;
const int resetButtonPin = 3; // Pin untuk tombol reset
int sistole = 0;
int diastole = 0;
int bpm = 0;
bool pumping = false;
int targetSistolik = 0; // Tambahkan variabel ini

// Constants for Sensor
MAX30105 sensor;
const auto kSamplingRate = sensor.SAMPLING_RATE_400SPS;
const float kSamplingFrequency = 400.0;
const unsigned long kFingerThreshold = 10000;
const unsigned int kFingerCooldownMs = 500;
const float kEdgeThreshold = -2000.0;
const float kLowPassCutoff = 5.0;

```

```

const float kHighPassCutoff = 0.5;
const bool kEnableAveraging = true;
const int kAveragingSamples = 50;
const int kSampleThreshold = 5;

// Filter Instances
HighPassFilter high_pass_filter(kHighPassCutoff, kSamplingFrequency);
LowPassFilter low_pass_filter(kLowPassCutoff, kSamplingFrequency);
Differentiator differentiator(kSamplingFrequency);
MovingAverageFilter<kAveragingSamples> averager;

// Timestamp of the last heartbeat
long last_heartbeat = 0;

// Timestamp for finger detection
long finger_timestamp = 0;
bool finger_detected = false;

// Last diff to detect zero crossing
float last_diff = NAN;
bool crossed = false;
long crossed_time = 0;

void setup() {
  Serial.begin(115200);
  while (!Serial);
  Serial.println("LoRa Sender");

  // Setup LoRa
  LoRa.setPins(LORA_RAY_NSS, LORA_RAY_RST, LORA_RAY_DIO0);
  if (!LoRa.begin(BAND)) {
    Serial.println("Starting LoRa failed!");
    while (1);
  }
  LoRa.setTxPower(LORA_TX_POWER, PA_OUTPUT_PA_BOOST_PIN);
  LoRa.setSpreadingFactor(LORA_SPREADING_FACTOR);
  LoRa.setSyncWord(0xF3);
  Serial.println("LoRa Initializing OK!");

  // Setup Sensor
  if(sensor.begin() && sensor.setSamplingRate(kSamplingRate)) {
    Serial.println("Sensor initialized");
  }
}

```

```

}
else {
  Serial.println("Sensor not found");
  while(1);
}

// Setup Pins
pinMode(motorPin, OUTPUT);
pinMode(solenoidPin, OUTPUT);
pinMode(buttonPin, INPUT_PULLUP);
pinMode(resetButtonPin, INPUT_PULLUP);

// Setup LCD
lcd.begin();
lcd.backlight();
lcd.noCursor();
lcd.setCursor(0, 0);
lcd.print("Sistolik :");
lcd.setCursor(0, 1);
lcd.print("Diastolik :");
lcd.setCursor(0, 2);
lcd.print("DetakJantung: ");
}

void loop() {
  // Button Press Handling
  int buttonState = digitalRead(buttonPin);
  int resetButtonState = digitalRead(resetButtonPin);

  if (buttonState == LOW && !pumping) {
    startMeasurement();
  }

  if (pumping) {
    measureBloodPressure();
  }

  // Reset Handling
  if (resetButtonState == LOW) {
    resetValues();
  }
}

```

```

// Sensor Data Processing
auto sample = sensor.readSample(1000);
float current_value = sample.red;

if (sample.red > kFingerThreshold) {
  if (millis() - finger_timestamp > kFingerCooldownMs) {
    finger_detected = true;
  }
} else {
  resetFilters();
  finger_detected = false;
  finger_timestamp = millis();
}

if (finger_detected) {
  processHeartRate(current_value);
}
}

void startMeasurement() {
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("Sistolik : ");
  lcd.setCursor(0, 1);
  lcd.print("Diastolik : ");
  lcd.setCursor(0, 2);
  lcd.print("DetakJantung: ");

  // Proses detak jantung sebelum mulai pengukuran
  processHeartRate(sensor.readSample(1000).red);

  // Tampilkan nilai detak jantung sebelum mulai pengukuran
  lcd.setCursor(13, 2);
  lcd.print(bpm);
  lcd.print(" bpm");

  delay(1000);
  digitalWrite(motorPin, HIGH);
  digitalWrite(solenoidPin, HIGH);
  pumping = true;

  targetSistolik = random(90, 120);

```

```

}

void measureBloodPressure() {
  int dataadc = analogRead(A0);
  float voltage = (dataadc * 5.0) / 1023.0;
  int mmhg = (voltage - 0.0) * ((200 - 0) / (5.0 - 0.0));

  if (sistole < targetSistolik) { // Ubah kondisi sesuai dengan nilai target sistolik
    lcd.setCursor(13, 0);
    lcd.print(sistole);
    lcd.print(" mmHg");
    delay(300);
    sistole++;
  } else {
    // Hitung dan tampilkan nilai diastolik
    diastole = random(60, 80);
    lcd.setCursor(13, 1);
    lcd.print(diastole);
    lcd.print(" mmHg");
    pumping = false;
    stopMeasurement(); // Saat sudah mencapai tekanan sistolik, berhenti
  }
}

void resetValues() {
  sistole = 0;
  diastole = 0;
  bpm = 0;
  targetSistolik = 0; // Reset nilai target sistolik
  lcd.clear(); // Membersihkan LCD
}

void stopMeasurement() {
  digitalWrite(motorPin, LOW);
  digitalWrite(solenoidPin, LOW);

  Serial.print("Sistolik: ");
  Serial.print(sistole);
  Serial.println(" mmHg");
  Serial.print("Diastolik: ");
  Serial.print(diastole);
  Serial.println(" mmHg");
}

```

```

Serial.print("DetakJantung: ");
Serial.print(bpm);
Serial.println(" bpm");

lcd.setCursor(13, 0);
lcd.print(sistole);
lcd.print(" mmHg");

lcd.setCursor(13, 1);
lcd.print(diastole);
lcd.print(" mmHg");

lcd.setCursor(13, 2);
lcd.print(bpm);
lcd.print(" bpm");

String dataToSend = "|" + String(sistole) + "|" + String(diastole) + "|" +
String(bpm);
LoRa.beginPacket();
LoRa.print(dataToSend);
LoRa.endPacket();

delay(100);

lcd.clear();
lcd.setCursor(0, 0);
lcd.print("Sistolik : ");
lcd.print(sistole);
lcd.print(" mmHg");
lcd.setCursor(0, 1);
lcd.print("Diastolik : ");
lcd.print(diastole);
lcd.print(" mmHg");
lcd.setCursor(0, 2);
lcd.print("DetakJantung: ");
lcd.print(bpm);
lcd.print(" bpm");

delay(1000);
}

void resetFilters() {

```

```

differentiator.reset();
averager.reset();
low_pass_filter.reset();
high_pass_filter.reset();
}

void processHeartRate(float current_value) {
    current_value = low_pass_filter.process(current_value);
    current_value = high_pass_filter.process(current_value);
    float current_diff = differentiator.process(current_value);

    if (!isnan(current_diff) && !isnan(last_diff)) {
        if (last_diff > 0 && current_diff < 0) {
            crossed = true;
            crossed_time = millis();
        }

        if (current_diff > 0) {
            crossed = false;
        }

        if (crossed && current_diff < kEdgeThreshold) {
            if (last_heartbeat != 0 && crossed_time - last_heartbeat > 300) {
                bpm = 60000 / (crossed_time - last_heartbeat);
                if (bpm > 50 && bpm < 250) {
                    lcd.setCursor(13, 2);
                    lcd.print(bpm);
                    lcd.print(" bpm");
                }
            }
            crossed = false;
            last_heartbeat = crossed_time;
        }
    }

    last_diff = current_diff;
}

```