

SISTEM PEMANTAUAN DETAK JANTUNG DAN SUHU TUBUH MENGUNAKAN PROTOKOL KOMUNIKASI MQTT

SKRIPSI

Untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun oleh:

Fathul Hakim

NIM: 135150201111095



PROGRAM STUDI TEKNIK INFORMATIKA

JURUSAN TEKNIK INFORMATIKA

FAKULTAS ILMU KOMPUTER

UNIVERSITAS BRAWIJAYA

MALANG

2020



PENGESAHAN

SISTEM PEMANTAUAN DETAK JANTUNG DAN SUHU TUBUH MENGGUNAKAN PROTOKOL KOMUNIKASI MQTT

SKRIPSI

Diajukan untuk memenuhi sebagian persyaratan memperoleh gelar Sarjana Komputer

Disusun Oleh : Fathul Hakim NIM: 135150201111095

Skripsi ini telah diuji dan dinyatakan lulus pada 8 Januari 2020

Telah diperiksa dan disetujui oleh:

Dosen Pembimbing I

[Handwritten signature]

Heru Nurwarsito, M.Kom NIP: 196504021990021001

Mengetahui Ketua Jurusan Teknik Informatika



[Handwritten signature]

Tri Astoro Kurniawan, S.T, M.T, Ph.D NIP: 19710518 200312 1 001

PERNYATAAN ORISINALITAS

Saya menyatakan dengan sebenar-benarnya bahwa sepanjang pengetahuan saya, di dalam naskah skripsi ini tidak terdapat karya ilmiah yang pernah diajukan oleh orang lain untuk memperoleh gelar akademik di suatu perguruan tinggi, dan tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis disitasi dalam naskah ini dan disebutkan dalam daftar referensi.

Apabila ternyata didalam naskah skripsi ini dapat dibuktikan terdapat unsur-unsur plagiasi, saya bersedia skripsi ini digugurkan dan gelar akademik yang telah saya peroleh (sarjana) dibatalkan, serta diproses sesuai dengan peraturan perundang-undangan yang berlaku (UU No. 20 Tahun 2003, Pasal 25 ayat 2 dan Pasal 70).

Malang, 8 Januari 2020



Fathul Hakim

NIM: 135150201111095

PRAKATA

Puji syukur kehadirat Allah SWT, karena berkat rahmat dan bimbinganNya, penulis dapat menyelesaikan tugas akhir dengan baik.

Penulisan skripsi ini diajukan memenuhi sebagian persyaratan untuk memperoleh gelar Sarjana Komputer pada Program Studi Teknik Informatika Fakultas Ilmu Komputer Universitas Brawijaya Malang. Judul yang penulis kerjakan adalah Sistem Pemantauan Detak Jantung dan Suhu Tubuh menggunakan Protokol Komunikasi MQTT.

Banyak pihak yang berperan atas terselesaikannya penelitian dan penulisan skripsi ini. Atas bantuan yang telah diberikan, penulis ingin menyampaikan ucapan terimakasih kepada terimakasih kepada:

1. Bapak Achmad Fatchurrahman dan Ibu Latifah selaku kedua orangtua penulis yang selalu memberikan doa, semangat dan motivasi dan dukungan segalanya selama ini.
2. Bapak Heru Nurwarsito, M.Kom selaku dosen pembimbing 1, atas bimbingan dan kesabarannya demi tercapainya keutuhan pada penyusunan tugas akhir ini.
3. Wayan Firdaus Mahmudy, S.Si, M.T, Ph.D., selaku dekan Fakultas Ilmu Komputer.
4. Tri Astoto Kurniawan, S.T, M.T, Ph.D., selaku ketua jurusan Teknik Informatika.
5. Segenap bapak dan ibu dosen FILKOM yang telah mendidik dan berbagi ilmu kepada penulis.
6. Teman-teman angkatan 2013 informatika yang selalu memberi semangat dan arahan demi kelancaran penulisan tugas akhir ini.
7. Teman hidup yang selalu sabar dan memberikan motivasi untuk penulis agar cepat selesai
8. Keluarga bani Amin dan bani Musa yang selalu memberikan dukungan dan motivasi untuk segera menyelesaikan tugas akhir ini

Malang, 14 Desember 2019

Penulis

Fathulhakim57@gmail.com

ABSTRAK

Banyaknya pasien yang tidak memiliki dana sehingga harus rawat jalan menyebabkan persoalan yang baru. Banyaknya penyakit yang ditimbulkan dari organ jantung dan suhu tubuh manusia menjadi permasalahan yang penting. Berdasarkan kasus yang telah terjadi, diperlukanlah sebuah sistem untuk memantau detak jantung dan suhu tubuh. Untuk menyelesaikan masalah tersebut dilakukanlah perancangan sistem detak jantung dan suhu tubuh menggunakan pengiriman MQTT (*Message Queuing Telemetry Transport*). Pembacaan detak jantung menggunakan sensor *pulse oxymeter* dengan cara menempelkan ibu jari ke sensor dan sensor suhu tubuh *lm35* dengan cara di genggam. Pemrosesan data dilakukan dengan menggunakan *Wemos D1* dan penyimpanan online yaitu *cloud* dengan memanfaatkan protokol komunikasi *mqtt*. Proses menampilkan suatu data dilakukan secara *realtime*. Hasil pengujian akurasi memberikan hasil yang berbeda yaitu 6,1% untuk sensor detak jantung dan 0,9% untuk sensor suhu tubuh. Sedangkan untuk pengujian waktu *RTT (Round Trip Time)* sistem dapat menyelesaikan kurang dari 1 menit.

Kata kunci: Detak jantung, Suhu Tubuh, *Sensor Node*, *cloud*, *Mqtt*



ABSTRACT

The large number of patients who do not have money so that they have to be treated outpatient causes new problems. The number of diseases caused by heart organs and human body temperature is an important problem. Based on the cases that have occurred, we need a system to monitor heart rate and body temperature. To solve this problem the design of the heart rate and body temperature was carried out using MQTT (Message Queuing Telemetry Transport) delivery. Heart rate reading uses a pulse oxymeter sensor by attaching the thumb to the sensor and the body temperature sensor LM35 by handheld. Data processing is done by using Wemos D1 and online storage namely cloud by utilizing the mqtt communication protocol. The process of displaying data is done in realtime. The test results on accurate give different results, namely 6.1% for heart rate sensors and 0.9% for body temperature sensors. Whereas for RTT (Round Trip Time) testing the system can complete less than 1 minute.

Keywords: Heartbeat, Body Temperature, Sensor Node, cloud, Mqtt



DAFTAR ISI

PENGESAHAN.....	Error! Bookmark not defined.
PERNYATAAN ORISINALITAS.....	ii
PRAKATA.....	iii
ABSTRAK.....	v
ABSTRACT.....	vi
DAFTAR GAMBAR.....	x
DAFTAR TABEL.....	xii
BAB 1 PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	2
1.3 Tujuan.....	2
1.4 Manfaat.....	2
1.5 Batasan Masalah.....	3
1.6 Sistematika Pembahasan.....	3
BAB 2 LANDASAN KEPUSTAKAAN.....	5
2.1 Kajian Pustaka.....	5
2.2 Jantung.....	7
2.3 Suhu Tubuh.....	7
2.4 Wemos D1.....	8
2.5 Max3010.....	8
2.6 Sensor suhu LM35.....	9
2.7 Internet of Things (IoT).....	9
2.8 MQTT.....	10
2.9 Cloud.....	11
2.10 Bahasa Pemrograman C.....	11
2.11 <i>Round Trip Time</i> (RTT).....	11
BAB 3 METODOLOGI PENELITIAN.....	13
3.1 Studi Literatur.....	13
3.2 Analisis Kebutuhan.....	14



3.3 Analisis Kebutuhan Perangkat Keras	14
3.4 Analisis Kebutuhan Perangkat Lunak.....	15
3.5 Perancangan Sistem.....	15
3.6 Implementasi Sistem	16
3.7 Pengujian dan Analisis	16
3.8 Kesimpulan dan Saran	17
BAB 4 REKAYASA KEBUTUHAN.....	18
4.1 Analisis Kebutuhan	18
4.1.1 Analisis Kebutuhan Fungsional	18
4.1.2 Analisis Kebutuhan Non-Fungsional	18
4.1.3 Kebutuhan Perangkat Lunak.....	18
4.1.4 Kebutuhan Perangkat Keras.....	19
BAB 5 PERANCANGAN DAN IMPLEMENTASI	20
5.1 Perancangan Sistem.....	20
5.1.1 Perancangan Sensor Node	20
5.1.2 Perancangan <i>Publisher</i> dan Subscriber Sistem Detak Jantung dan Suhu Tubuh.....	21
5.1.3 Perancangan Cara Kerja Node Sensor.....	24
5.1.4 Perancangan Antarmuka.....	26
5.2 Implementasi Sistem	28
5.2.1 Implementasi Sensor Detak Jantung.....	29
5.2.2 Implementasi Sensor Suhu Tubuh	29
5.3 Implementasi Perangkat Keras Sensor Node	30
5.3.1 Implementasi Perangkat lunak Sensor Node.....	31
5.3.2 Implementasi Program Subscriber.....	39
BAB 6 PENGUJIAN DAN ANALISIS.....	45
6.1 Prosedur dan Tujuan Pengujian Fungsional	45
6.1.2 Hasil dan Analisis Pengujian Fungsional	45
6.2 Pengujian Sensor Node	46
6.2.1 Tujuan Pengujian.....	46
6.2.2 Prosedur Pengujian Sistem	46
6.2.3 Hasil Pengujian.....	46



6.2.4 Analisis Pengujian.....	49
6.3 Pengujian Kinerja sistem.....	50
6.3.1 Tujuan Pengujian.....	51
6.3.2 Prosedur Pengujian.....	51
6.3.3 Hasil Pengujian.....	51
6.3.4 Analisis Pengujian.....	53
BAB 7 KESIMPULAN DAN SARAN.....	55
7.1 Kesimpulan.....	55
7.2 Saran.....	55
DAFTAR REFERENSI.....	56



DAFTAR GAMBAR

Gambar 2.1 Jantung Manusia	7
Gambar 2.2 Wemos D1	8
Gambar 2.3 Sensor Detak Jantung Max3010	9
Gambar 2.4 Sensor Suhu Tubuh Lm35	9
Gambar 2.5 Arsitektur MQTT	11
Gambar 3.1 Diagram Alur Metodologi	13
Gambar 3.2 Alur Rancangan Sistem Pemantauan Detak Jantung dan Suhu Tubuh	16
Gambar 5. 1 Rancangan Topologi Sistem	20
Gambar 5.2 Rangkaian Sensor Node	21
Gambar 5.3 Diagram alur Publisher	22
Gambar 5. 4 Alur Kerja Subscriber pada Node Sensor	23
Gambar 5. 5 Diagram Alur Cara Kerja Max3010	24
Gambar 5. 6 Diagram Alur Cara Kerja Sensor Lm35	25
Gambar 5. 7 Rancangan Antarmuka Sistem Pemantauan Detak Jantung dan Suhu Tubuh	26
Gambar 5. 8 Perancangan Antarmuka Perangkat lunak Subscriber	27
Gambar 5. 9 Perancangan Antarmuka Perangkat lunak Subscriber(lanjutan)	28
Gambar 5.10 Implementasi Sensor Detak Jantung	29
Gambar 5.11 Implementasi Sensor Suhu Tubuh	30
Gambar 5. 12 Implementasi Sensor Node	31
Gambar 5. 13 Implementasi Antarmuka suhu pada Adafuit	43
Gambar 5. 14 Implementasi Antarmuka Detak Jantung pada Adafuit	43
Gambar 5. 15 Implementasi Antarmuka Perangkat lunak Subscriber	43
Gambar 5. 16 Implementasi Perangkat Lunak Subscriber(Lanjutan)	44
Gambar 6.1 Grafik Data Perbandingan Program Sistem dengan Jam pintar	47
Gambar 6.2 Grafik Data Perbandingan Program sistem dengan Termometer	48
Gambar 6.3 Grafik perbandingan Program sistem dengan Jam pintar	49
Gambar 6.4 Grafik Perbandingan Sistem dengan Termometer	50
Gambar 6. 5 Grafik Hasil pengujian performa sistem	52



Gambar 6. 6 Grafik hasil pengujian performa sistem 53

Gambar 6. 7 Grafik Perbandingan RTT detak jantung dan suhu tubuh 54



DAFTAR TABEL

Tabel 2. 1 Kajian Pustaka..... 5

Tabel 2. 2 Suhu Tubuh Normal..... 7

Tabel 4. 1 Kebutuhan fungsional sistem..... 18

Tabel 4. 2 Kebutuhan Non-Fungsional..... 18

Tabel 4. 3 Kebutuhan Perangkat Lunak 19

Tabel 4. 4 Kebutuhan laptop..... 19

Tabel 4.5 Perangkat Node Sensor 19

Tabel 5. 1 Penjelasan Antarmuka Sistem Pemantauan Detak Jantung dan Suhu Tubuh 27

Tabel 5. 2 Perancangan Antarmuka Perangkat lunak Subscriber 27

Tabel 5. 3 Perancangan Antarmuka Perangkat lunak Subscriber..... 28

Tabel 5. 4 Keterangan Sensor node 31

Tabel 5. 5 Kode Program Max3010 dan Lm35..... 31

Tabel 5. 6 Kode Program Subscriber 39

Tabel 6. 1 Skenario Pengujian Fungsional..... 45

Tabel 6. 2 Hasil Pengujian Fungsional..... 45

Tabel 6. 3 Hasil dari Pengujian Sensor Detak Jantung Max3010..... 46

Tabel 6. 4 Hasil Pengujian Sensor Lm35..... 48

Tabel 6. 5 Hasil Pengujian performa detak jantung..... 51

Tabel 6. 6 Hasil pengujian performa suhu tubuh..... 52



BAB 1 PENDAHULUAN

1.1 Latar Belakang

Pemantauan kesehatan sangatlah penting dalam industri kesehatan. Karena kendala waktu ataupun dana, orang tidak dapat mengunjungi rumah sakit, yang mungkin dapat menyebabkan banyak masalah kesehatan dalam waktu yang singkat. Kesehatan menjadi topik yang paling banyak dibicarakan di beberapa negara-negara besar yang memiliki banyak penduduknya. Faktor kesehatan salah satu yang sangat penting dalam meningkatkan kualitas hidup manusia (Durga Amanarth, 2017).

Jantung adalah organ dalam badan manusia yang berperan sangat penting dalam system peredaran darah, jantung yaitu sebuah organ yang terdiri dari otot yang berongga sehingga berguna untuk mempompa darah dan menyebarkan ke pembuluh darah secara berulang-ulang. Detak jantung dapat memberikan indikasi seberapa sehat kondisi jantung dan dapat mendiagnosa kelainan pada jantung (Pendidikan, 2014). Begitupun juga yang harus di perhatikan adalah suhu tubuh, suhu tubuh yaitu kemampuan badan untuk memberikan panas dan mengeluarkan panas. Suhu tubuh dapat dilakukan di beberapa bagian tubuh, salah satunya ketiak, dubur dan telinga. (Quamila, 2017).

Internet of Things (IoT) adalah konsep yang menggabungkan berbagai objek fisik atau virtual melalui jaringan nirkabel atau kabel. Sehingga dapat berkomunikasi dengan mandiri. Bagian-bagian terpenting dari arsitektur IoT yaitu *Wireless Sensor Network (WSN)*, dimana kumpulan WSN tersebut akan terhubung ke jaringan internet yang lebih besar (Dejana Ugrenovic, 2015). Akan tetapi tidak semua protokol itu cocok untuk penerapan IoT, karena tidak tepatan pemilihan protokol hanya dapat membuat system menjadi rumit. Oleh sebab itu membutuhkan protokol komunikasi yang layak bekerja secara mudah dan efisien dalam penggunaannya.

Message Queue Telemetry Transport (MQTT) adalah protokol komunikasi mesin ke mesin menggunakan kabel ataupun tanpa kabel, *MQTT* adalah protokol komunikasi yang berbasis *TCP/IP*. Protokol *MQTT* memiliki ukuran pengiriman data yang kecil. Protokol komunikasi *MQTT* yaitu bagian yang terpenting dalam penerapan *IoT* karena berbasis *publish subscribe* yang mampu melakukan penerimaan maupun mengirim suatu pesan ataupun perintah pemantauan yang di inginkan pengguna. (Valerie Lampkin, 2012).

Beberapa peneliti telah melakukan riset. Menurut Hindarto, untuk menciptakan alat detak jantung menggunakan Max3010 dan arduino. Max3010 diletakkan di bagian jari dan rangkaian arduino. Max3010 akan merekam data dengan cara memberikan cahaya infra merah ketika jari telah di tempelkan pada Max3010. Max3010 digunakan untuk merekam detak jantung, aliran darah pada jari seiring dengan detak jantung dan akan direkam oleh phototransistor. Data detak jantung akan diolah oleh arduino dan dikirim ke computer melalui serial RS-232. Data yang ditampilkan di computer berupa *pulse* detak jantung permenit

dan data itu disimpan di microsoft excel. (Hindarto, 2015). Penelitian yang terakhir dilakukan oleh Riyanto. Penelitian tersebut bertujuan menghasilkan alat ukur suhu tubuh dan denyut jantung menggunakan telepon pintar *android*. Sensor yang dipakai oleh penulis yaitu ds18b20 sebagai alat mengukur suhu badan dan *Max3010* sebagai alat mendeteksi denyut jantung dan arduino. Data suhu tubuh dan detak jantung di tampilkan pada lcd yang berukuran 16x2 yang tersambung oleh arduino. Data juga akan ditampilkan ke telepon pintar melalui *Bluetooth HC-05*. (RIYANTO, 2016).

Berdasarkan penelitian-penelitian yang telah dilakukan, peneliti mencoba melakukan penelitian yang berbeda dengan cara menggunakan *Wemos D1* sebagai mikrokontroller, untuk sensor detak jantung peneliti memakai *Max3010* dan sensor untuk suhu tubuh yaitu *LM35* sehingga dapat menampilkan hasil olahan data ke dalam *Cloud adafruit.io*. Disini peneliti memakai protokol *mqtt* karena mampu berbasis *publish subscribe* yang mana data sensor *publisher* yang dikirimkan menuju broker mampu diterima dan *unsubscribe* secara *realtime*.

1.2 Rumusan Masalah

Rumusan masalah yang diterapkan dalam penelitian ini adalah:

1. Bagaimana cara menggunakan alat detak jantung dan suhu tubuh?
2. Bagaimana alur pengiriman data dari *publisher* ke *subscriber*?
3. Bagaimana performa kinerja dari sistem yang telah dibuat oleh penulis?

1.3 Tujuan

Tujuan dari pengerjaan penelitian ini adalah:

1. Mampu mengetahui cara menggunakan alat detak jantung dan suhu tubuh
2. Mampu mengetahui alur pengiriman data dari *publisher* ke *subscriber*
3. Mengetahui performa kinerja sistem yang telah dibuat oleh penulis

1.4 Manfaat

Manfaat yang diharapkan dalam penelitian ini adalah:

1. Mampu memberikan referensi untuk menggunakan protokol *publish subscribe* untuk membangun sebuah sistem yang memakai perangkat *IoT*.
2. Mampu menjadi referensi untuk penelitian berikutnya dan menerapkan dalam perkuliahan
3. Mampu diterapkan pada pasien rawat jalan

1.5 Batasan Masalah

Batasan masalah dari penelitian yang dilakukan adalah:

1. Mikrokontroler Wemos D1
2. Sensor Max3010 untuk mengukur detak jantung
3. Sensor Icm35 untuk mengukur suhu tubuh
4. *Cloud* yang di pakai adafruit.io
5. Protokol komunikasi yang digunakan *MQTT*

1.6 Sistematika Pembahasan

Penelitian yang dilakukan ini mempunyai sistematika pembahasan yang diuraikan menjadi beberapa bab:

BAB I : Pendahuluan

Menjelaskan latar belakang dari perkembangan *Internet of things (IoT)*, kesehatan detak jantung dan suhu tubuh dan komunikasi *Mqtt*. Bab ini juga menjabarkan tentang rumusan masalah, tujuan, manfaat, batasan masalah dan sistematika pembahasan yang dikerjakan oleh penulis

BAB II : Landasan Kepustakaan

Bab ini menjelaskan tentang kajian pustaka dan dasar teori yang berkaitan dengan penulisan skripsi.

BAB III : Metodologi

Menjelaskan mengenai langkah-langkah penelitian dari studi literatur, sistem yang akan di buat, analisis kebutuhan implementasi dan pengujian yang dilakukan sistem dan kesimpulan dari sistem yang dilakukan oleh peneliti.

BAB IV : Rekayasa Kebutuhan

Menguraikan kebutuhan untuk pengguna, kebutuhan sistem dan perangkat lunak maupun keras yang dimiliki oleh sistem.

BAB V : Perancangan dan Implementasi

Menguraikan cara merancang sistem dan proses penerapan sistem yang akan dibuat.

BAB VI : Pengujian dan Hasil Analisis

Pengujian dilakukan jika perancangan dan implementasi telah selesai dilakukan. Pengujian sistem bertujuan untuk menguji coba sistem, serta mengetahui kinerja dari sistem.

BAB VII : Penutup

Memberikan penarikan kesimpulan yang diambil oleh peneliti dan saran untuk pengembangan skripsi selanjutnya.



BAB 2 LANDASAN KEPUSTAKAAN

Teori-teori yang penulis gunakan demi menunjang penelitian tersebut berada pada bab ini, antara lain: wemos d1, jantung manusia, suhu tubuh manusia, sensor max3010 detak jantung manusia, sensor lm35 untuk mengukur suhu tubuh manusia, penyimpanan online (*cloud*), protokol MQTT dan bahasa pemrograman.

2.1 Kajian Pustaka

Tabel 2. 1 Kajian Pustaka

No	Nama Penulis, Tahun, Judul	Persamaan	Perbedaan	
			Penelitian Terdahulu	Rencana Penelitian
1	Anggraini, Nenny, 2015, RANCANG BANGUN ALAT PEREKAM FREKUENSI LISTRIK JANTUNG	Merancang sistem detak jantung	Lampu LED yang menyala sesuai tempo frekuensi mampu ditampilkan	Merancang sistem detak jantung dan suhu tubuh menggunakan protocol komunikasi mqtt
2	Wirawan, Gede Saindra Santyadiputra, Nyoman Sugihartini, 2017. SISTEM PEMANTAU SUHU LAB JARAK JAUH BERBASIS ARDUINO	Membuat suatu sistem untuk mengukur suhu	Mengukur suhu ruangan laboratorium memakai mikrokontroler <i>arduino</i> dan sensor DHT-11	Mengukur suhu tubuh menggunakan Lm35 dan Wemos D1
3	Solichin, Dr. I Dewa Gede Hari Wisana, ST, MT, Dra. Dwi Herry Andayani, MM, 2017. MONITORING DETAK JANTUNG, SUHU TUBUH, DAN RESPIRASI	Pembuatan alat monitoring suhu tubuh, respirasi dan detak jantung	Aplikasi berbasis android hasil dalam pembuatan sistem detak jantung, suhu tubuh dan respirasi	Sistem yang akan dirancang berkomunikasi dengan protocol mqtt yang berbasis publish-subscribe
4	Zennifa, Fadilla, 2013. Prototipe Alat Deteksi Dini	Pembuatan sistem detak jantung menggunakan	Pembuatan sistem penyakit detak jantung via android	Pembuatan sistem detak jantung dan suhu tubuh

Mandiri Penyakit Jantung Menggunakan Sistem Pakar VCSIR, Arduino dan Handphone Android	arduino	menggunakan sistem pakar menggunakan pulse sensor	menggunakan Max3010 dan lm35
--	---------	---	------------------------------

Penelitian pertama di lakukan oleh (Anggraini, 2015). Banyak pasien sakit jantung sehingga harus memakai alat EKG(Elektrokardioagram),namun jika pemeriksaan berkala akan menghabiskan biaya dan waktu ke rumah sakit,maka dari itu penulis menghubungkan Arduino Uno Grove Chest Strap Heart Rate Sensor untuk membuat alat portable perekam frekuensi jantung. Sistem dapat merubah data suhu tubuh dari sinyal analog menjadi digital dengan frekuensi dari denyut jantung ditampilkan dalam satuan *beats per minute(bpm)* berupa angka dan grafik. Data dapat ditampilkan melalui serial monitor pada komputer. Hasil dari perekaman detak jantung datanya akan tersimpan dan berformat .txt. (Anggraini, 2015).

Penelitian kedua di lakukan oleh (Wirawan,2017). Sistem pemantau suhu lab dengan Arduino ini sudah baik dan memudahkan penggunaanya. Ia memakai Arduino dan sensor DHT-11. Sistem yang telah dikerjakan oleh penulisan sudah mampu menampilkan kelembaban yang berada di ruang lab dan memberikan hasil dalam bentuk grafik. Untuk menguji kepekaan dari sensor dilakukanlah pemberian suhu panas dengan cara tidak langsung di tempelkan pada alat tersebut dan melihat tegangan yang telah dihasilkan dengan alat yaitu voltmeter (Wirawan, 2017).

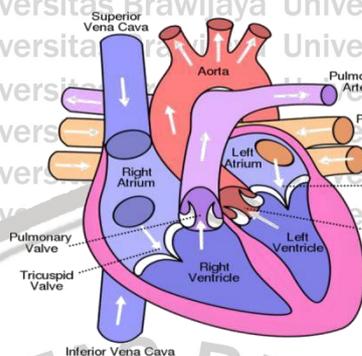
Penelitian ketiga dilakukan oleh (Solichin, 2017). Pembuatan aplikasi android untuk mendeteksi detak jantung, suhu tubuh dan respirasi berhasil dirancang. Kondisi ini ditunjukan bahwa semua rangkaian dan sensor dapat bekerja sesuai dengan pengaturan dan fungsinya, Begitu juga aplikasi android yang dibuat dapat bekerja dengan baik untuk menampilkan data yang di terima dari alat dan juga program arduino yang dibuat berjalan sesuai dengan rencana (Solichin, 2017) .

Penelitian keempat dilakukan oleh (Zennifa, 2013). Pembuatan alat detak jantung mandiri penyakit jantung menggunakan sistem pakar memiliki manfaat yaitu mampu mendeteksi sejak awal gejala sakit jantung dan dapat dilakukan secara sendiri. Hasil yang didapatkan oleh sistem pakar sesuai hasil diagnose dokter ataupun pakar. Penulis juga membandingkan sensor oximetri dan *pulse* sensor dengan hasil perbandingan dari kedua alat yaitu 2.60563158 bpm (Zennifa, 2013).



2.2 Jantung

Jantung yaitu salah satu organ yang paling berpengaruh didalam tubuh, ciri-ciri dari jantung ialah berongga dan memiliki empat ruang dan jantung terletak di dada tepat diatas diafragma. Kantong pada bagian jantung yang memiliki lapisan disebut kantong pericardial.



Gambar 2.1 Jantung Manusia

Kecepatan detak jantung sangat bervariasi, dalam satu menit jantung mampu menghasilkan detak jantung selama 60 hingga 100 denyutan. Parameter untuk mengetahui keadaan dari suatu jantung adalah BPM (*Beats Per Minute*), jumlah darah yang baik akan mampu dihasilkan oleh jantung yang sehat untuk apapun yang dilakukan oleh tubuh. (Kabo, 2008).

2.3 Suhu Tubuh

Suhu ialah hasil dari penyesuaian panas yang dihasilkan oleh bagian tubuh. Organ tubuh yang dapat menyeimbangkan suhu tubuh agar tetap normal yaitu kulit, musim dan lingkungan bisa jadi penyebab dari suhu tubuh yang berubah. Suhu tubuh normal sekitar 36-37,5C . Pada sebagian orang terdapat perubahan suhu tubuh saat malam hari suhu tubuh terasa tinggi, sedangkan pada pagi hari suhu rendah dan pada dini hari suhu tubuh akan minimum dan maksimum suhu tubuh saat sore hari (Gibson, 2002).

Tabel 2. 2 Suhu Tubuh Normal

USIA	SUHU (°C)
3 bulan	37,5
6 bulan	37,5
1 tahun	37,7
3 tahun	37,2
5 tahun	37,0
7 tahun	36,8
9 tahun	36,7
11 tahun	36,7
13 tahun	36,6
Dewasa	36,4
> 70 tahun	36,0

(Sumber: Sehat center)

2.4 Wemos D1

Mikrokontroler yang telah berelaborasi berbasis modul mikrokontroler ESP8266 yaitu adalah Wemos D1. Perangkat ini memiliki dimensi ukuran 68.6 mm x 53.4 mm, beroperasi pada tegangan 3.3V. Modul ini memiliki digital I/O pin sebanyak 11 buah dan 1 analog input pin. Mikrokontroler ini dapat terhubung dengan jaringan Wireless dan mampu melakukan perintah meskipun memakai command. Memiliki frekuensi 80/160 MHz chip ini dibekali dengan 4MB flash memory. (PUTRI, 2017)



Gambar 2.2 Wemos D1

(Sumber: Ebay)

2.5 Max3010

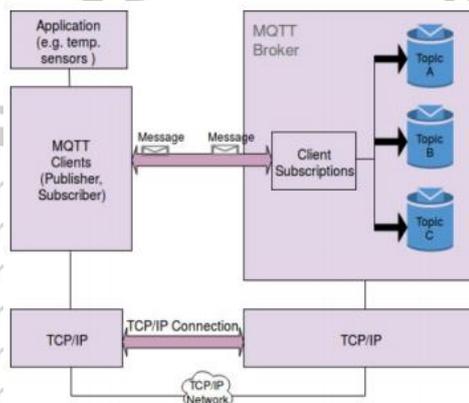
Salah satu sensor yang dirancang dan dikhususkan untuk detak jantung pada mikrokontroler yaitu Max3010. Sensor ini dapat merekam detak jantung dengan cara menempelkan di jari. Cara kerja dari sensor ini yaitu dengan metode transmisi pemancar yaitu LED & penerima yaitu detektor foto ditempatkan pada sisi yang berlawanan dari jari. Dalam metode ini jari ini akan ditempatkan di antara LED & detektor foto. Ketika jari diletakkan bagian cahaya akan diserap oleh jari dan sebagian akan mencapai detektor foto. Sekarang dengan setiap detak jantung akan ada peningkatan volume aliran darah ini akan menghasilkan lebih banyak cahaya yang diserap oleh jari sehingga lebih sedikit cahaya mencapai detektor foto. Max3010 ini memiliki daya 1.8V dan 3.3V yang terpisah dengan led. Ukuran dari sensor tersebut yaitu 5,6mm x 3,3mm x 1,55mm 14-pin. Sensor tersebut juga memiliki daya yang rendah. (Maxim, 2016).

2.8 MQTT

Message Queue Telemetry Transport (MQTT) adalah protokol komunikasi mesin ke mesin menggunakan kabel ataupun tanpa kabel, *MQTT* adalah protocol komunikasi yang berbasis *TCP/IP*. *MQTT* telah digunakan sebagai bagian dari banyak perangkat IoT dan pengiriman pesan secara instan protocol ini memiliki daya yang rendah sehingga bias disebut protocol yang ringan.

Konsep dari *MQTT* yaitu:

- a. *Publish/subscribe*: Dalam protokol *MQTT*, *publisher* mempublikasikan pesan dan pengguna yang berlangganan topik yang pada umumnya dianggap sebagai model *publish/subscribe*. *Subscriber* yang melakukan *subscribe* ke topik tertentu yang terkait, subscriber dengan yang menerima setiap pesan *publish* ketopik-topik tersebut. Di sisi lain, pengguna dapat *publish* pesan ke topik, sedemikian rupa sehingga memungkinkan semua subscriber mengakses pesan dari topik tersebut.
- b. Topik dan *subscriber*: Di *MQTT*, *publisher* mempublish pesan ke topik yang dapat dianggap sebagai subjek pesan. *Subscriber* berlangganan topik untuk mendapatkan pesan tertentu. Mensubscribe topik dapat dijelaskan, yang membatasi data yang dikumpulkan untuk topik tertentu.
- c. *QoS (Quality Of Service)*: *Quality of service* yaitu tingkat kualitas layanan dalam mendistribusikan data. Dibagi menjadi 3 yaitu:
 - QoS level 0: Di level ini pesan dikirim paling banyak sekali dan tidak memberikan jaminan pengiriman pesan.
 - QoS level 1: Di level ini, data dikirim setidaknya satu kali dan memungkinkan untuk mengirimkan pesan lebih dari satu kali dengan membutuhkan ACK namun bisa terjadi duplikasi data yang berada pada penerima.
 - QoS level 2: Dalam level ini, pesan dikirim tepat sekali dengan menggunakan 4 way *handshake* (Soni, 2017)



Gambar 2.5 Arsitektur MQTT

(Sumber: soni,2017)

2.9 Cloud

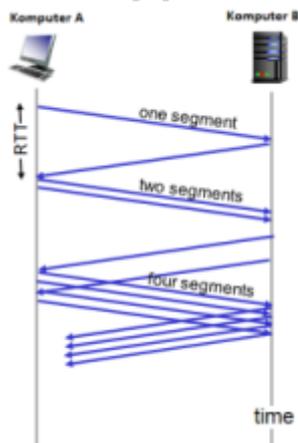
Cloud adalah penyimpanan data yang berbasis daring. Banyak kegunaan dari *cloud* yaitu untuk membagikan data, penyimpanan data, manajemen data besar dan informasi medis. *Cloud* bisa juga menjadi broker. Pengguna dapat mengakses *cloud* melalui *web browser* atau aplikasi yang berada di telepon pintar. Data dari *cloud* di simpan dilokasi server yang jauh. Kekurangan dari penyimpanan berbasis daring ini yaitu dalam hal keamanan dan privasi karena banyak data yang sangat sensitive yang disimpan dalam *cloud* itulah mengapa kita harus membaca syarat dan ketentuan yang berlaku saat kita ingin mendaftar. Contoh dari *cloud* yaitu io.adafruit, gmail dll. (Armbrust, 2010)

2.10 Bahasa Pemrograman C

Bahasa yang hampir tersedia di semua jenis komputer yaitu bahasa C, banyak aplikasi terdahulu maupun sekarang masih memakai bahasa C karena bahasa C ini memiliki sifat portable dan fleksib. Dalam pembuatan bahasa pemrograman *java*, C ikut ambil alih pembuatan bahasa ini sehingga *java* memiliki sifat terstruktur. Bahasa C ini bukan hanya bisa di gunakan pada windows namun di linux pun juga mampu di operasikan di linux namun berbeda di linux. Di linux bahasa c olah dahulu karena pada text editor tidak dapat di jalankan. (solichin, 2003)

2.11 Round Trip Time (RTT)

Round Trip Time (RTT) yaitu waktu yang diperlukan untuk sebuah packet untuk melakukan perjalanan dari sumber tertentu dan ke tujuan tertentu dan kembali ke sumber tersebut. RTT ditentukan dengan menjumlahkan waktu pulang pergi dari alat ke server dan kembali (Server RTT), dan waktu perjalanan pulang pergi dari alat ke klien dan kembali (RTT Klien).



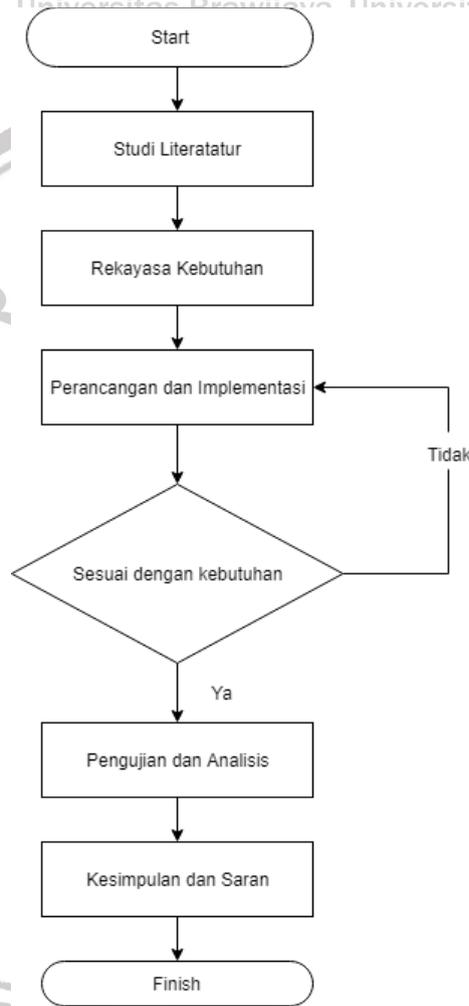
Gambar 2. 6 Round Trip Time (RTT)

Cara alternatif untuk memantau RTT jaringan adalah memantau waktu yang dibutuhkan untuk paket TCP untuk disetujui. Dalam menggunakan pendekatan ini, latency langsung dapat dipantau. Untuk mendapatkan perkiraan RTT jaringan yang akurat, nilai ini harus diukur selama waktu persiapan koneksi TCP sebelum terjadi keterlambatan level aplikasi.



BAB 3 METODOLOGI PENELITIAN

Bab metodologi ini menerangkan tentang alur penulisan dalam penelitian Sistem Pemantauan Detak Jantung dan Suhu Tubuh menggunakan Protokol Komunikasi MQTT. Dengan adanya langkah-langkah penulisan di bawah ini, proses pengerjaan ini dapat di mengerti. Adapun langkah-langkah yang akan di lakukan, terdapat di bawah ini:



Gambar 3.1 Diagram Alur Metodologi

3.1 Studi Literatur

Pada bagian studi literatur berisi tentang rujukan yang mendukung pengerjaan dari skripsi ini, yang berasal dari jurnal, e-book dan sumber lainnya. Dibawah ini adalah beberapa teori pendukung sebagai berikut :

1 Bahasa C



Dalam hal ini memaparkan tinjauan tentang pemrograman Bahasa

C.

2. Protokol MQTT

Dalam hal ini memaparkan tinjauan tentang protocol komunikasi

MQTT.

3. *Internet of things*

Dalam hal ini memaparkan tinjauan tentang *Internet Of Things*, dengan menemukan sisi karakteristik tersebut.

4. Sensor Node

Dalam hal ini memaparkan tinjauan tentang mikrokontroler dan macam-macam yang digunakan oleh sensor dalam mendapatkan nilai detak jantung dan suhu tubuh pada sistem.

5. *Java Language*

Pada bagian ini memaparkan kajian tentang pemrograman Bahasa

Java.

6. *Cloud*

Pada bagian ini memaparkan kajian tentang penyimpanan berbasis *online* (cloud), disini penulis menggunakan cloud *io.adafruit.co*.

3.2 Analisis Kebutuhan

Analisis kebutuhan memiliki tujuan yaitu agar peneliti mengerti tentang yang dibutuhkan saat mengerjakan pengujian agar sesuai dan mampu di pahami.

3.3 Analisis Kebutuhan Perangkat Keras

Pada sub bab ini berisi tentang kebutuhan perangkat keras berupa sensor detak jantung Max3010, Lm35 sensor suhu tubuh, Wemos D1 dan laptop

1. Sensor Node

a. Sensor Max3010 kegunaannya sebagai mengukur detak jantung pengguna.

b. Sensor Lm35 kegunaannya sebagai alat ukur suhu tubuh pengguna.

c. Mikrokontroler *Wemos D1* kegunaannya untuk menjalankan seluruh kegiatan pada perangkat yang akan digunakan.

2. LCD

Menampilkan hasil pengukuran detak jantung dan suhu tubuh dalam bentuk angka

3. Laptop

Laptop Lenovo kegunaannya sebagai menjalankan perangkat lunak

Subscriber

3.4 Analisis Kebutuhan Perangkat Lunak

Peneliti memakai perangkat lunak yang mampu diterapkan pada penelitian ini. Dibawah ini adalah beberapa perangkat lunak yang dapat digunakan dalam penelitian ini.

A. Program Sensor Node

Program sensor node disini menerapkan bahasa C, program sensor node harus mampu mengirim dan menerima perintah dari setiap rangkaian sensor node.

B. Cloud

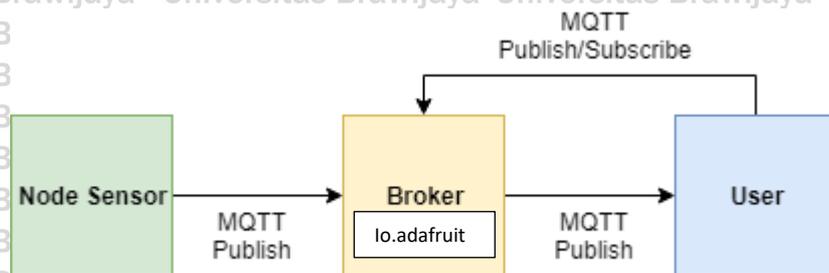
Cloud digunakan menerima dan meneruskan pengiriman data oleh rangkaian sensor node dan juga cloud digunakan sebagai Broker. Cloud disini menggunakan *io.adafruit.com*.

C. Wireshark

Wireshark digunakan untuk memantau paket yang menggunakan protokol mqtt yang telah di *publish* oleh rangkaian sensor node dan broker sampai di subscribe.

3.5 Perancangan Sistem

Peneliti akan merancang sebuah sistem pemantauan detak jantung dan suhu tubuh menggunakan protokol komunikasi MQTT. Peneliti memakai sensor Max3010 untuk detak jantung dan Lm35 sebagai sensor suhu, *wemos D1* sendiri kegunaannya untuk pemroses data sensor. Cara kerjanya yaitu node sensor *publish* sebuah data yang dihasilkan oleh detak jantung dan suhu tubuh menggunakan protokol komunikasi MQTT menuju *broker*, broker disini yaitu *cloud adafruit* yang tugasnya menampilkan dan meneruskan data ke user atau pengguna menggunakan protokol komunikasi MQTT. Ketika data sampai ke user atau pengguna, data akan di subscribe dan akan di tampilkan pada computer pengguna.



Gambar 3.2 Alur Rancangan Sistem Pemantauan Detak Jantung dan Suhu Tubuh

3.6 Implementasi Sistem

Dari penjelasan tersebut, sensor dihubungkan ke Wemos D1 supaya hasil pengkoleksian data dapat terkirim ke broker melalui jaringan *Wifi*. Pesan akan dikirimkan dengan bahasa C ke Wemos D1 begitu juga ke node sensor sehingga mampu melakukan pengiriman menggunakan koneksi *wifi*.

Ketika di broker, broker mampu mengirim data menggunakan protokol komunikasi MQTT menuju pengguna untuk untuk disubscribe di aplikasi tambahan subscriber dengan menggunakan bahasa *Java*. Untuk pengambilan data detak jantung dan suhu tubuh, diambil setiap satu menit sekali, dalam satuan bpm dan derajat celcius

3.7 Pengujian dan Analisis

Dalam bagian ini dilakukan pengujian dan analisis agar mampu melihat performa dan kinerja sistem yang telah di buat oleh peneliti.

A. Pengujian Sensor Node

Pengujian ini dikerjakan dengan memperbandingkan hasil dari sistem dengan alat yang sudah ada, agar dapat mengetahui tingkat akuratnya sebuah sensor dari sistem yang telah penulis buat.

B. Pengujian Fungsionalitas

Pengujian ini dilakukan dengan menjalankan seluruh fungsi dari sensor node ke broker hingga sampai ke pengguna mampu memsubscribe data detak jantung dan suhu tubuh menggunakan protokol komunikasi MQTT yang di koneksikan dengan *wifi* sehingga di harapkan dapat menjalankan semua fitur fitur yang sudah ada.

C. Pengujian Kinerja Sistem

Pengujian inii dilakukan dengan cara *Round Trip Time* (RTT) yaitu penghitungan waktu yang diperlukan untuk pengiriman data sampai ke broker dan diteruskan ke subscriber. Pengujian ini dilakukan untuk mengetahui tingkat keberhasilan dari sistem.

3.8 Kesimpulan dan Saran

Pengambilan kesimpulan dan saran dilaksanakan setelah dapat melalui alur dalam penelitian. Kesimpulan didapatkan setelah mengetahui pembahasan rangkuman pengujian dan analisis. Untuk saran, peneliti mampu menerima saran dari hasil tersebut dan supaya bisa digunakan penelitian akan datang.



BAB 4 REKAYASA KEBUTUHAN

Bagian ini di uraikan tentang kebutuhan fungsional dan nonfungsional yang harus mampu diterapkan. Kebutuhan fungsional meliputi fitur-fitur yang sudah ada dalam pembuatan sistem, sedangkan non fungsional meliputi alat-alat yang mensupport dalam pengerjaan pembuatan sistem.

4.1 Analisis Kebutuhan

4.1.1 Analisis Kebutuhan Fungsional

Kebutuhan fungsional yaitu bagian yang harus ada dalam sistem agar tercapainya sistem yang diinginkan. Kebutuhan fungsional dalam pengerjaan penelitian ini akan di jelaskan di Tabel 4.1 berikut ini:

Tabel 4. 1 Kebutuhan fungsional sistem

No.	Kebutuhan Fungsional
1	Node sensor mampu mempublish data ke <i>cloud</i>
2	Node sensor mempublish data melalui protokol MQTT via wifi
3	<i>Broker</i> mampu mempublish data ke pengguna untuk di subscribe
4	Pengguna dapat melihat detail data detak jantung dan suhu tubuh dari aplikasi subscriber
5	Data detak jantung dan suhu tubuh dapat dilihat oleh pengguna melewati cloud dan lcd

4.1.2 Analisis Kebutuhan Non-Fungsional

Non-fungsional yaitu membahas fungsi-fungsi yang telah dibatasi pada sistem tersebut. Dibawah ini yaitu kebutuhan non-fungsional, yang akan disebutkan pada tabel 4.2 berikut:

Tabel 4. 2 Kebutuhan Non-Fungsional

No	Kebutuhan Non-fungsional
1	Sistem mampu diakses melewati perangkat dekstop
2	Sistem mampu menghasilkan payload model data JSON

4.1.3 Kebutuhan Perangkat Lunak

Kebutuhan perangkat lunak sangat penting dari penyelesaian sistem pengembangan ini. Perangkat lunak yang terdapat pada sistem ini terbagi menjadi tigabagian. Peratama *Code editor*, Kedua yaitu *Compiler*, ketiga sebagai *broker*.

Tabel 4. 3 Kebutuhan Perangkat Lunak

Perangkat	Kebutuhan Perangkat Lunak
Arduino ide	Wadah untuk merancang kode pemrograman dan compiler dalam perancangan kode program untuk Wemos
Netbeans	Wadah merancang kode pemrograman dan compiler dalam perancangan kode program untuk pembuatan tampilan program
Adafruit	Sebagai penyedia <i>server cloud</i> dan broker dalam program
Bahasa Pemrograman C	Sebagai bahasa untuk sistem yang akan dibuat
Wireshark	Perangkat lunak yang berguna untuk melakukan pengujian

4.1.4 Kebutuhan Perangkat Keras

Dalam sistem yang akan dibuat terdiri dari dua kebutuhan perangkat keras. Pertama laptop dan node sensor. Berikut perangkat yang diperlukan:

Tabel 4. 4 Kebutuhan laptop

Komponen	Spesifikasi
Processor	Pentium Dual Core
Memory (RAM)	4GB
Harddisk	500gb
Sistem Operasi	Windows 10 Pro

Tabel 4.5 Perangkat Node Sensor

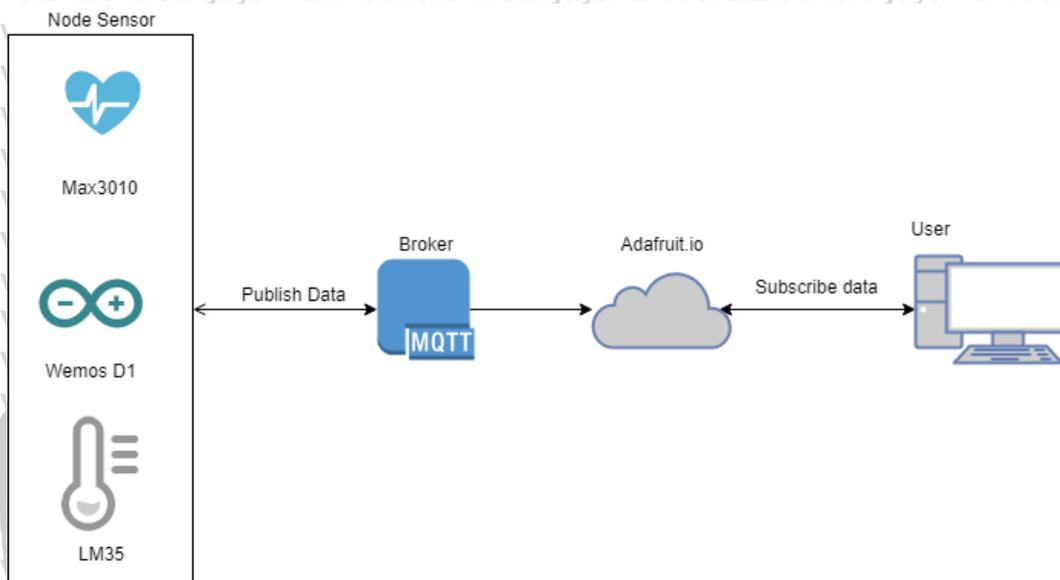
Komponen	Spesifikasi
WEMOS D1	CPU : ESP8266EX 80MHz/160MHz Flash : 4M bytes
Sensor Suhu Lm35	Daya: 4VDC-30VDC Daerah Pengukuran: 0-100C
Max30100	Daya: 1.8V dan 3.3V



BAB 5 PERANCANGAN DAN IMPLEMENTASI

5.1 Perancangan Sistem

Pada bagian ini berisi komponen penting yang menjelaskan bagaimana sistem akan bekerja dan terdiri dari perancangan node sensor, broker dan tampilan desktop. Di bawah ini yaitu rancangan dari topologi sistem yang akan dikerjakan peneliti, digambarkan pada Gambar 5.1:

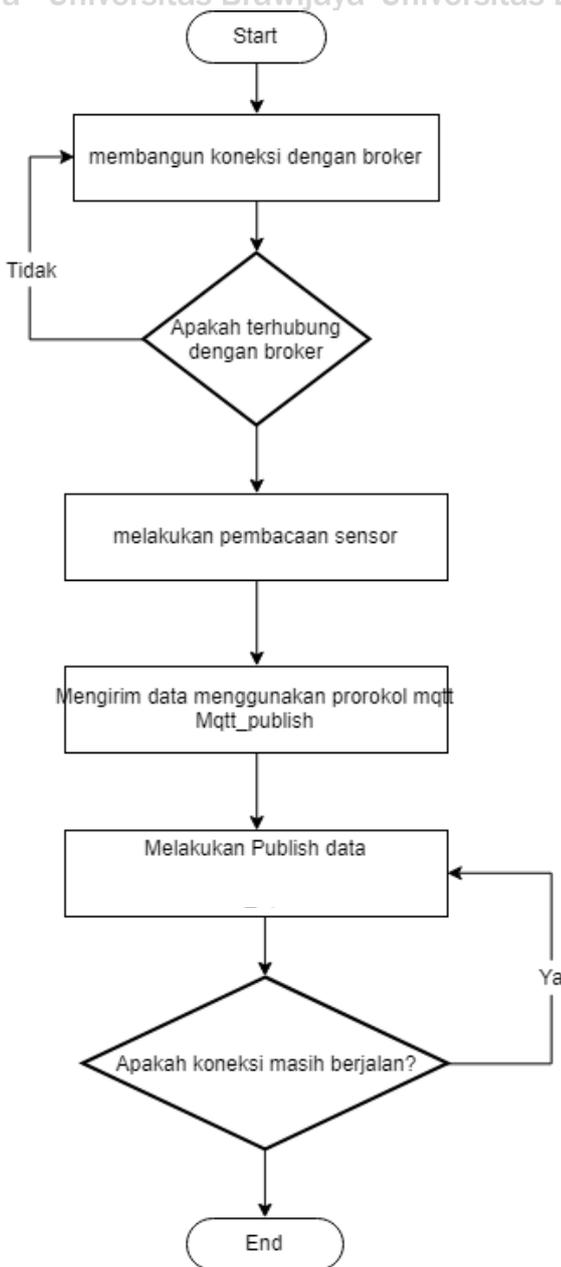


Gambar 5. 1 Rancangan Topologi Sistem

Gambar tersebut menerangkan sebuah gambaran sistem yang akan diterapkan. Dalam pengerjaan penelitian ini protokol MQTT digunakan sebagai protokol komunikasi antara node sensor dan *cloud*. Pada Gambar 5.1 dapat dilihat bahwa terdapat node sensor yang berfungsi sebagai publisher namun pengguna hanya dapat melihat hasil olah data dari aplikasi subscriber berbasis desktop yang telah disediakan. Disini broker yang digunakan juga sebagai cloud, cloud yang digunakan yaitu *io.adafruit.com*. Aplikasi desktop tersebut berbasis *GUI* (Graphic User Interface). Node sensor berupa Wemos D1, *Max3010* dan *Lm35*.

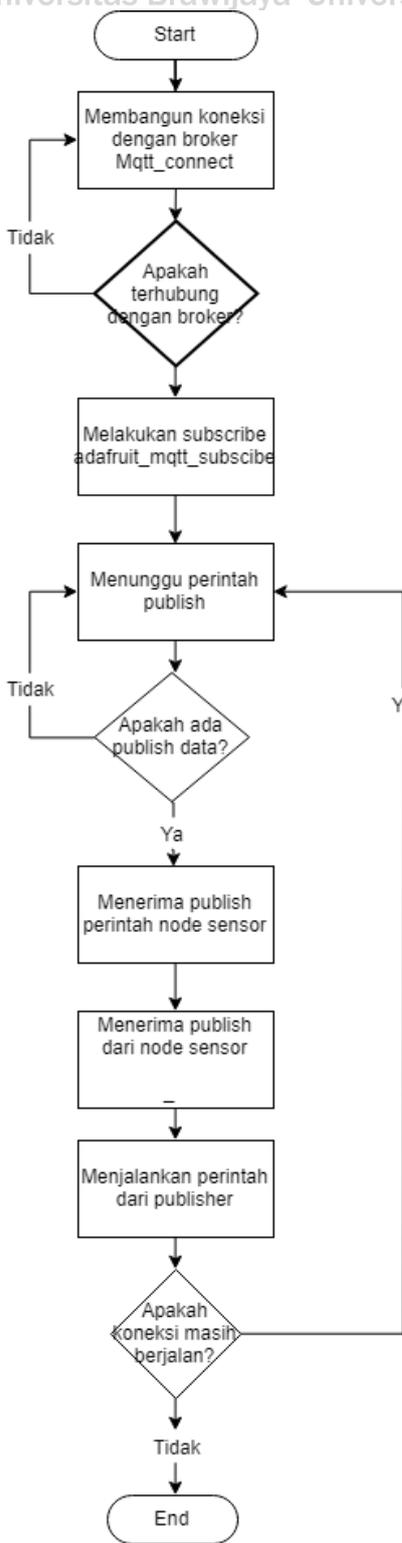
5.1.1 Perancangan Sensor Node

Untuk mengkoleksi data dari detak jantung dan suhu tubuh dilakukan pengaturan untuk sensor *max3010* dan *lm35* harus terintegrasi oleh *wemos d1*. Di bawah ini adalah sensor sensor node yang dirancang untuk pembuatan dari sistem yang dilakukan oleh penulis yang telah ditujukan pada gambar 5.2.



Gambar 5.3 Diagram alur Publisher

Pada gambar 5.3 memperlihatkan alur cara kerja dari *publisher* kepada *node sensor*. Dalam gambar tersebut *node sensor* sudah terhubung oleh koneksi internet sehingga *node sensor* memulai koneksi dengan broker MQTT atau adafruit, jika sudah terkoneksi *node sensor* sudah bisa mengerjakan publish pesan atau suatu data sensor pada topic yang diinginkan.



Gambar 5. 4 Alur Kerja Subscriber pada Node Sensor

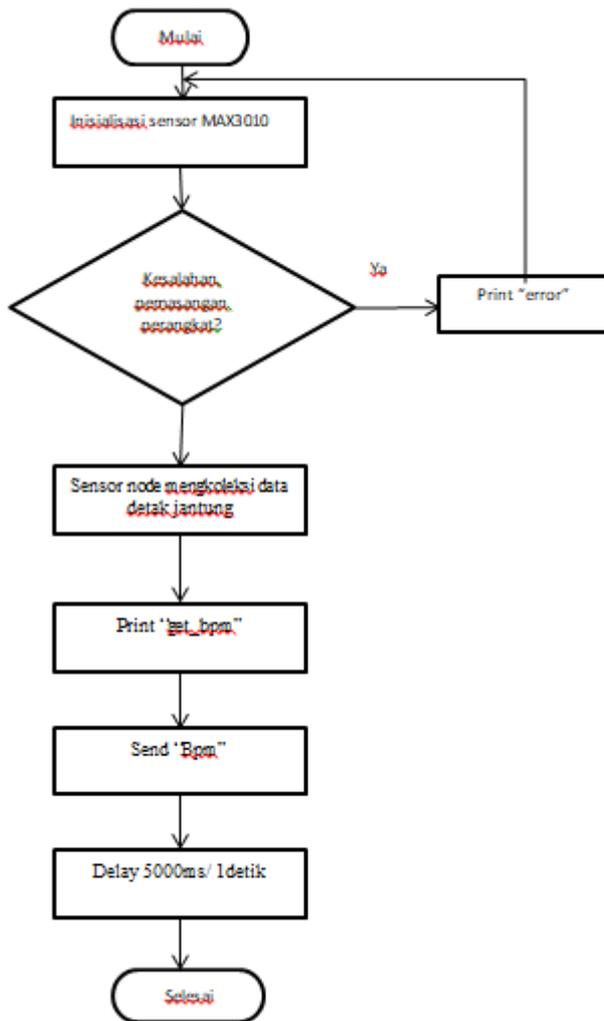
Gambar 5.4 memaparkan bagaimana alur dari *broker* ke *subscriber*. Dalam gambar tersebut node sensor sudah dihubungkan ke internet, lalu sensor memulai membuat koneksi dengan broker MQTT. Jikalau sudah terhubung, akan



melakukan subscribe pada topic. Ketika selesai, broker akan memulihkan data ke node sensor dan menampilkan dalam aplikasi subscriber.

5.1.3 Perancangan Cara Kerja Node Sensor

Pada bagian perancangan dipasang supaya sensor Max30100 yang terpasang pada mikrokontroler Wemos D1 dapat mengirimkan data nilai ke broker. Data yang telah didapatkan dari sensor Max3010 yaitu nilai BPM (*Beats per minute*) dan waktu saat pengambilan data tersebut. Pengambilan suatu data pada node sensor dijelaskan pada Gambar 5.5



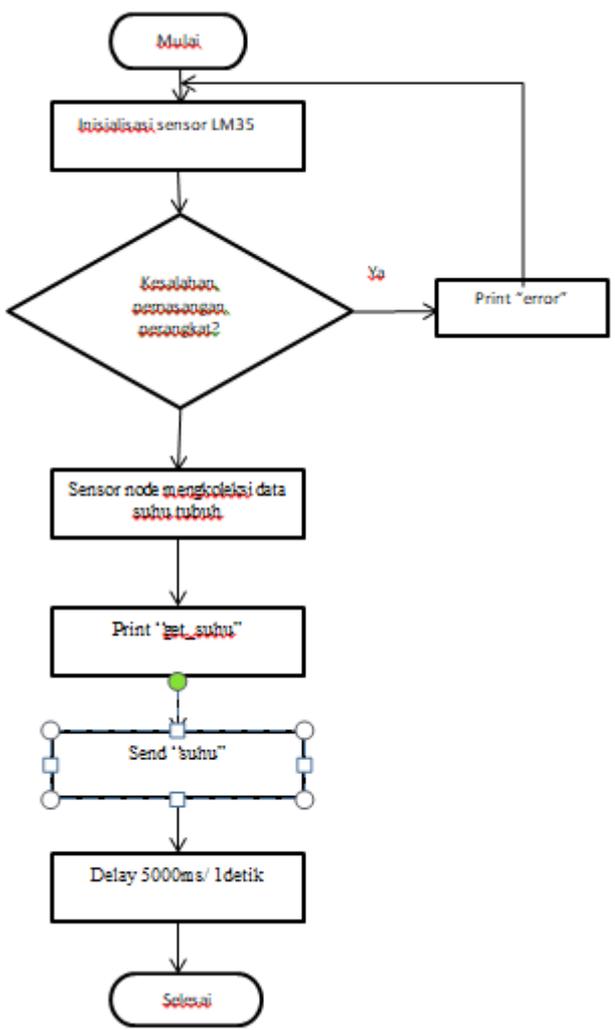
Gambar 5.5 Diagram Alur Cara Kerja Max3010

Pada gambar 5.5 dapat dilihat bahwa rancangan pada sensor max3010 yang terhubung dengan Wemos D1 mampu mengoleksi data detak jantung. Diagram alur pengambilan data detak jantung bpm dijelaskan dibawah ini.

1. Wemos D1 mulai melakukan inialisasi pada sensor untuk memberikan apakah perangkat sensor max3010 sudah bisa dan sesuai dalam pemasangannya.

2. Wemos d1 mulai melakukan fungsi pembacaan nilai BPM(Beats Per Minute) pada sensor max3010
3. Lalu Wemos D1 mampu memberikan nilai BPM, Wemos D1 akan menampilkan nilai BPM(Beats Per Minute) pada serial monitor dan LCD
4. Pada Wemos D1 menggunakan delay pengiriman data yaitu 5 detik untuk mendapatkan hasil yang diinginkan yaitu akurat

Setelah dari data detak jantung kita akan menjelaskan cara kerja dari sensor suhu yaitu LM35 Nilai yang telah didapatkan dari sensor LM35 yaitu nilai suhu dalam satuan derajat celcius dan waktu saat pengambilan data tersebut. Pengambilan suatu data pada node sensor dijelaskan pada Gambar 5.6



Gambar 5. 6 Diagram Alur Cara Kerja Sensor Lm35

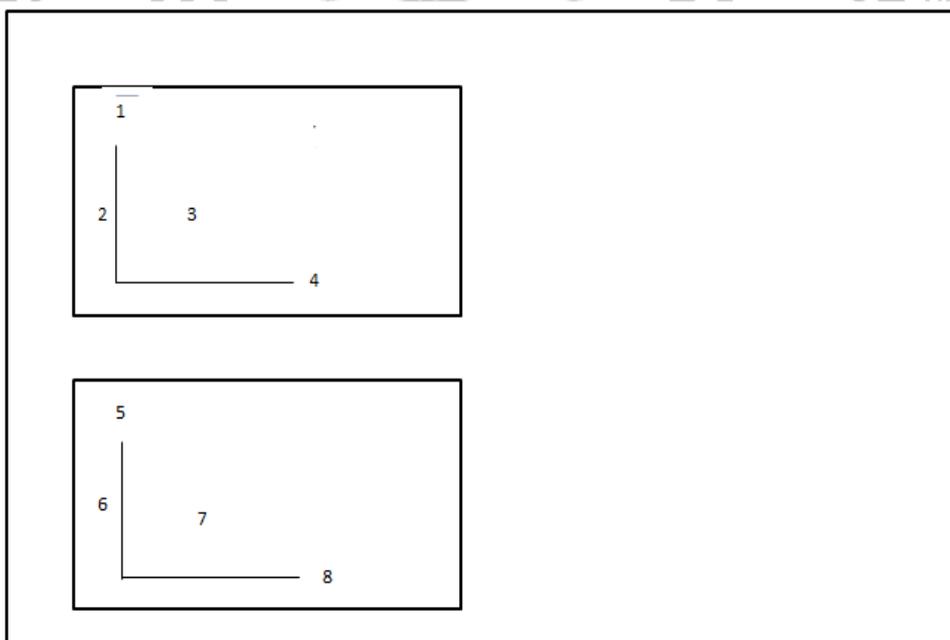
Pada gambar 5.6 dapat dilihat bahwa rancangan pada sensor LM35 yang terhubung dengan wemos d1 mampu mengkonversi data suhu tubuh. Diagram alur pengambilan data suhu tubuh dijelaskan dibawah ini.



1. Wemos d1 mulai melakukan inialisasi pada sensor untuk memberikan apakah perangkat sensor LM35 sudah bisa dan sesuai dalam pemasangannya.
2. Wemos d1 mulai melakukan fungsi pembacaan nilai suhu tubuh dalam satuan derajat celcius pada sensor LM35
3. Lalu Wemos d1 mampu memberikan nilai suhu, wemos d1 akan menampilkan nilai suhu tubuh pengguna pada serial monitor dan LCD
4. Pada wemos d1 menggunakan delay pengiriman data yaitu 5 detik untuk mendapatkan hasil yang diinginkan yaitu hasil yang akurat

5.1.4 Perancangan Antarmuka

Perancangan antarmuka memiliki kegunaan yaitu untuk memberikan tampilan sistem pemantauan perangkat dari sensor node dan akan selalu digunakan untuk menjalankan fungsi-fungsi yang penting dalam pemantauan. Gambar 5.5, Gambar 5.6 dan Gambar 5.7 adalah rancangan antarmuka dari sistem pemantauan detak jantung dan suhu tubuh.



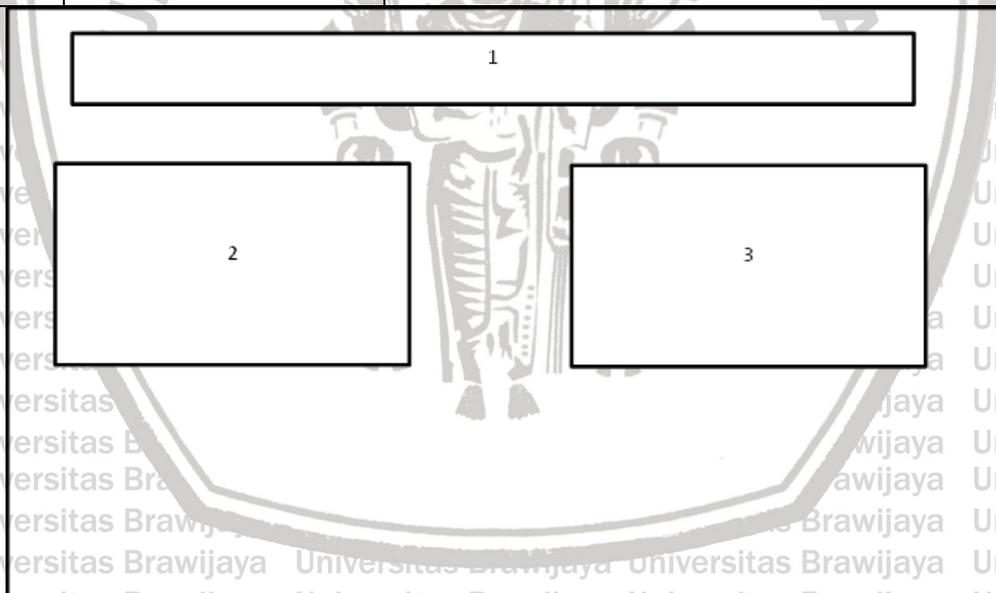
Gambar 5. 7 Rancangan Antarmuka Sistem Pemantauan Detak Jantung dan Suhu Tubuh

Rancangan Antarmuka ini berada pada *cloud* adafruit yang memiliki kegunaan yaitu menjadi broker dan memantau perangkat dari sensor yang telah terkoneksi ke mikrokontroler Wemos D1.

Dibawah ini ialah bagian-bagian penjelasan dari rancangan antarmuka dari sistem pemantauan Detak jantung dan Suhu tubuh dalam bentuk tabel.

Tabel 5. 1 Penjelasan Antarmuka Sistem Pemantauan Detak Jantung dan Suhu Tubuh

No	Bagian	Penjelasan
1	Judul Grafik	Berisi judul yang dimuat di halaman
2	Nilai Suhu	Menampilkan nilai suhu dalam satuan derajat celcius
3	Grafik Indeks	Berisi Grafik indeks hasil pengambilan nilai suhu tubuh dalam satu menit
4	Waktu	Berisi waktu kapan didapatkan informasi nilai suhu dalam satuan derajat <i>celcius</i>
5	Judul Grafik	Berisi judul yang dimuat di halaman
6	Nilai Detak Jantung	Menampilkan nilai detak jantung dalam satuan <i>BPM</i>
7	Statistik Indeks	Berisi statistic hasil indeks pengambilan nilai detak jantung dalam satu menit
8	Waktu	Berisi waktu kapan di dapatkan informasi nilai detak jantung



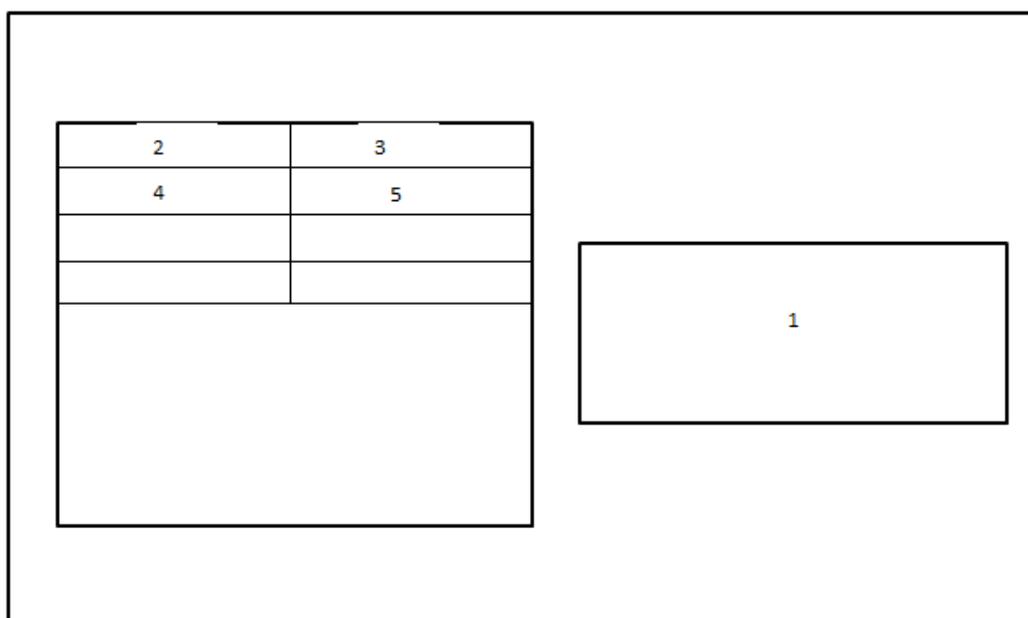
Gambar 5. 8 Perancangan Antarmuka Perangkat lunak Subscriber

Pada gambar 5.6 yaitu rancangan antarmuka dari perangkat lunak subscriber yang memiliki fungsi untuk mensubscribe data detak jantung maupun suhu tubuh. Dibawah ini adalah penjelasan bagian-bagian rancangan antarmuka dalam bentuk tabel

Tabel 5. 2 Perancangan Antarmuka Perangkat lunak Subscriber

No	Bagian	Penjelasan
----	--------	------------

1	Judul Perangkat Lunak	Berisi judul dari perangkat lunak
2	Tombol Button Detak Jantung	Berisi hasil pengkoleksian data detak jantung
3	Tombol Button suhu	Berisi hasil pengkoleksian data suhu tubuh



Gambar 5. 9 Perancangan Antarmuka Perangkat lunak Subscriber(lanjutan)

Tabel 5. 3 Perancangan Antarmuka Perangkat lunak Subscriber

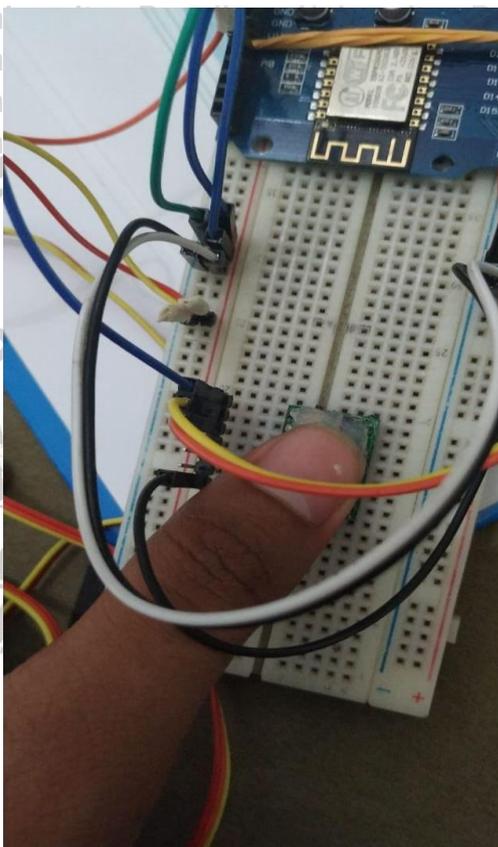
No	Bagian	Penjelasan
1	Button Pemberitahuan	Berisi pemberitahuan bahwa koneksi telah terhubung
2	Field Nilai	Berisi nilai BPM ataupun suhu
3	Field Waktu	Berisi waktu pengkoleksian data
4	Tabel nilai	Berisi nilai detak jantung ataupun suhu tubuh selama satu menit
5	Tabel waktu	Berisi waktu nilai pengkoleksian data selama satu menit

5.2 Implementasi Sistem

Pada hal ini implementasi sistem menerangkan apa saja komponen node sensor dan komponen perangkat lunak yang merujuk pada analisis kebutuhan dan perancangan.

5.2.1 Implementasi Sensor Detak Jantung

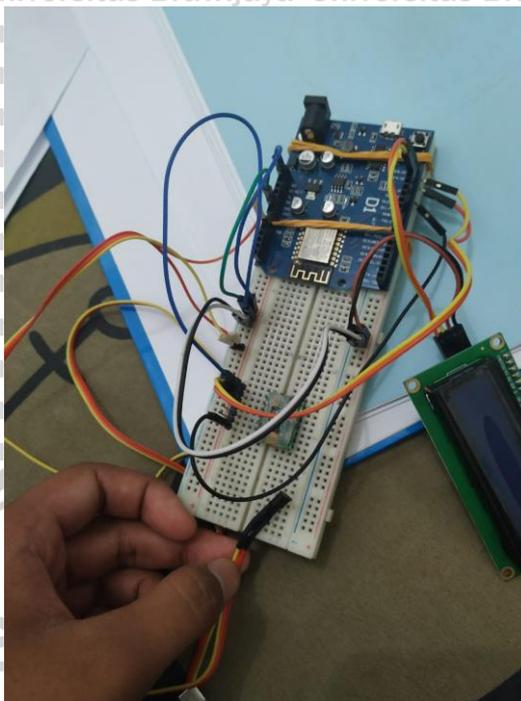
Dalam implementasi sensor detak jantung dihubungkan dengan kabel jumper dan diintegrasikan ke wemos D1 dengan cara menyambungkan ke port A0,GND dan 5V dan untuk pengambilan data,jari telunjuk ita tempelkan pada sensor pulse nya supaya data dapat diambil dan dikirimkan ke cloud dan ditampilkan lewat lcd. Untuk pengambilan datanya setiap satu menit sekali dalam satuan bpm.



Gambar 5.10 Implementasi Sensor Detak Jantung

5.2.2 Implementasi Sensor Suhu Tubuh

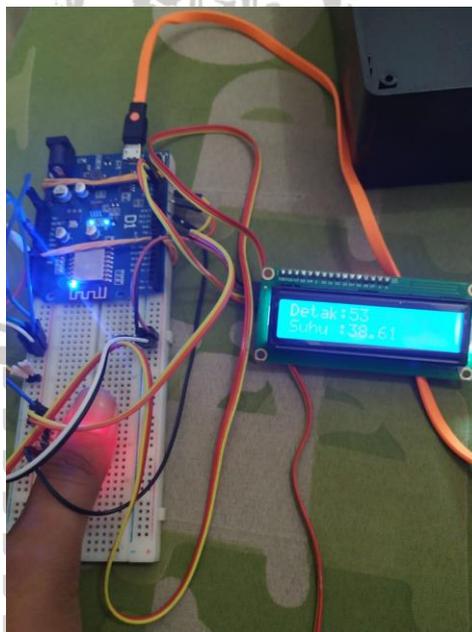
Dalam penerapan sensor suhu tubuh lm35 dihubungkan dengan kabel jumper dan diintegrasikan ke wemos d1 dengan cara menyambungkan kabel jumper ke 3.3V,GND dan SCK dan untuk pengambilan data,kita pasang sensor dengan cara di genggam dan kita tunggu satu menit untuk pengambilan data dalam satuan derajat celsius. Setelah satu menit data akan dikirim ke cloud.



Gambar 5.11 Implementasi Sensor Suhu Tubuh

5.3 Implementasi Perangkat Keras Sensor Node

Dalam penerapan sensor node telah dipaparkan pada rancangan sebelumnya. Implementasi perangkat keras sensor node ini terdiri dari Max3010, Lm35, wemos D dan Lcd. Implementasi perangkat keras tersebut dapat di lihat pada gambar 5.7.



Gambar 5. 12 Implementasi Sensor Node

Tabel 5. 4 Keterangan Sensor node

No.	Perangkat Keras Sensor node
1.	Sensor LM35
2.	Wemos D1
3.	Max3010
4.	Lcd

Gambar diatas menerangkan bahwa telah dilakukam implementasi sensor node. Sensor lm35 dan max3010 telah terhubung ke mikrokontroler wemos D1 dengan menggunakan kabel jumper yang telah dipasangkan pada VCC dan pin data. Untuk menyalakan node sensor cukup masukkan kabel micro ke dalam laptop dengan muatan listrik 3.2 volt.

5.3.1 Implementasi Perangkat lunak Sensor Node

Pada bagian ini memaparkan program dari perangkat lunak menggunakan sensor node dan memakai bahasa pemrograman C untuk sensor Max3101 dan lm35. Ditujukan pada Script 5.1 dibawah ini.

Tabel 5. 5 Kode Program Max3010 dan Lm35

```

1 #include <Wire.h>
2 #include <LiquidCrystal_I2C.h>
3
4 LiquidCrystal_I2C lcd(0x3F, 2, 1, 0, 4, 5, 6, 7, 3, POSITIVE);
5
6 #define merah 2
7 #define biru 13
8 #define hijau 12
9 #define merah_on digitalWrite(merah,1); digitalWrite(biru,0);
10 digitalWrite(hijau,0);
11 #define biru_on digitalWrite(merah,0); digitalWrite(biru,1);
12 digitalWrite(hijau,0);
13 #define hijau_on digitalWrite(merah,0); digitalWrite(biru,0);
14 digitalWrite(hijau,1);
15
16 #define tempPin A0
17 int val;
18 float mv, cel, fahr;
19
20 long previousMillis = 0;
21 long interval = 60000;
22 int BPM =0;
23 int jantung;
24 int dataadc1;

```





```

23 int x;
24 uint32_t detak = 10;
25
26
27 #include <ESP8266WiFi.h>
28 #include "Adafruit_MQTT.h"
29 #include "Adafruit_MQTT_Client.h"
30
31 /***** WiFi Access Point *****/
32
33
34
35 #define WLAN_SSID       "Miphone"
36 #define WLAN_PASS       "hakim888"
37
38 /***** Adafruit.io Setup *****/
39 #define AIO_SERVER      "io.adafruit.com"
40 #define AIO_SERVERPORT  1883
41 #define AIO_USERNAME    "hakim99"
42 #define AIO_KEY         "054e7ddc8bfc4e1190f1f83e0afbb6d3"
43 /***** Global State (you don't need to change this!) *****/
44
45 // Create an ESP8266 WiFiClient class to connect to the MQTT
46 // server.
47 // or... use WiFiClientSecure for SSL
48 //WiFiClientSecure client;
49
50 // Setup the MQTT client class by passing in the WiFi client and
51 // MQTT server and login details.
52 Adafruit_MQTT_Client mqtt(&client, AIO_SERVER, AIO_SERVERPORT,
53 AIO_USERNAME, AIO_KEY);
54
55 /***** Feeds *****/
56 Adafruit_MQTT_Publish bpm = Adafruit_MQTT_Publish(&mqtt,
57 AIO_USERNAME "/feeds/bpm");
58 Adafruit_MQTT_Publish temperature = Adafruit_MQTT_Publish(&mqtt,
59 AIO_USERNAME "/feeds/temperature");
60
61 Adafruit_MQTT_Subscribe onoffbutton =
62 Adafruit_MQTT_Subscribe(&mqtt, AIO_USERNAME "/feeds/tombol");
63
64 uint8_t txfailures = 0;
65 #define MAXTXFAILURES 3
66
67 int count_publish;
68 String t1, t2, t3, t4;

```



```

64 void MQTT_connect();
65
66 #include "MAX30100_PulseOximeter.h"
67 #define REPORTING_PERIOD_MS 1000
68 PulseOximeter pox;
69 uint32_t tsLastReport = 0;
70 void onBeatDetected()
71 {
72     // Serial.println("Beat!");
73 }
74
75
76 const int numReadings = 7;
77 int readings[numReadings];      // the array of numbers to be averaged
78 int readIndex = 0;              // the index of the current read
79 int total = 0;                  // the sum of all readings
80 int average = 0;                // the average of all readings
81 //XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
82 //XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
83 void setup()
84 {
85     Serial.begin(9600);
86     delay(10);
87
88     lcd.begin( 16, 2);
89     .
90     lcd.backlight();
91     lcd.setCursor(0, 0);
92     lcd.print("Starting....");
93     delay(1000);
94     lcd.clear();
95
96     pinMode(2, OUTPUT);
97     pinMode(13, OUTPUT);
98     pinMode(12, OUTPUT);
99     pinMode(15, OUTPUT);
100    digitalWrite(15, 0);
101
102    Serial.println(F("Adafruit MQTT demo"));
103
104
105
106
107

```

```
108
109
110
111 WiFi.begin(WLAN_SSID, WLAN_PASS);
112 while (WiFi.status() != WL_CONNECTED) {
113     delay(500);
114     Serial.print(">.");
115     lcd.setCursor(0, 0);
116     lcd.print("Connecting");
117     lcd.setCursor(7, 1);
118     lcd.print("wifi..");
119 }
120 lcd.clear();
121 Serial.println();
122
123 Serial.println("WiFi connected");
124 Serial.println("IP address: "); Serial.println(WiFi.localIP());
125
126 // Setup MQTT subscription for onoff feed.
127 mqtt.subscribe(&onoffbutton);
128
129 if (!pox.begin()) {
130     Serial.println("FAILED");
131     for (;;)
132 } else {
133     Serial.println("SUCCESS");
134 }
135 // MAX30100_LED_CURR_0MA
136 // MAX30100_LED_CURR_4_4MA
137 // MAX30100_LED_CURR_7_6MA
138 // MAX30100_LED_CURR_11MA
139 // MAX30100_LED_CURR_14_2MA
140 // MAX30100_LED_CURR_17_4MA
141 // MAX30100_LED_CURR_20_8MA
142 // MAX30100_LED_CURR_24MA
143 // MAX30100_LED_CURR_27_1MA
144 // MAX30100_LED_CURR_30_6MA
145 // MAX30100_LED_CURR_33_8MA
146 // MAX30100_LED_CURR_37MA
147 // MAX30100_LED_CURR_40_2MA
148 // MAX30100_LED_CURR_43_6MA
149 // MAX30100_LED_CURR_46_8MA
150 // MAX30100_LED_CURR_50MA
151
```

```

152     pox.setIRLedCurrent(MAX30100_LED_CURR_24MA);
153     pox.setOnBeatDetectedCallback(onBeatDetected);
154
155     for (int thisReading = 0; thisReading < numReadings;
156         thisReading++) {
157         readings[thisReading] = 0;
158     }
159     //xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
160     //xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
161     void loop()
162     {
163         waktu();
164         get_suhu();
165         bpm2();
166         // print_data();
167         // tampil_lcd();
168         // delay(500);
169     }
170     //xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
171     //xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
172     void tampil_lcd()
173     {
174         lcd.setCursor(0, 0);
175         lcd.print("Detak:"); lcd.print(average); lcd.print(" ");
176         lcd.setCursor(0, 1);
177         lcd.print("Suhu :"); lcd.print(cek); lcd.print(" ");
178     }
179     //xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
180     void print_data()
181     {
182         Serial.print("BPM = ");
183         Serial.print(average);
184         Serial.print("\tSuhu = ");
185         Serial.println(cek);
186     }
187     //xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
188     void waktu() {
189         unsigned long currentMillis = millis();
190         if (currentMillis - previousMillis > interval) {
191             previousMillis = currentMillis;
192
193             get_suhu();
194
195

```



```

196     datatampil();
197
198     if (!pox.begin()) {
199         Serial.println("FAILED");
200         for (;;)
201     } else {
202         Serial.println("SUCCESS");
203     }
204
205
206 }
207 }
208 //xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
209 void datatampil()
210 {
211     MQTT_connect();
212     //mqtt_counter();
213     //mqtt_subscribe();
214     mqtt_publish();
215
216     lcd.setCursor(9, 0);
217     lcd.print(" ");
218 }
219 //xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
220 void mqtt_counter() {
221     Serial.print("==counter==> ");
222     Serial.println(count_publish);
223     count_publish++; if (count_publish >= 10) {
224         count_publish = 0;
225         mqtt_publish();
226     }
227 }
228 //xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
229 // Function to connect and reconnect as necessary to the MQTT
230 // server. Should be called in the loop function and it will take care if
231 // connecting.
232 void MQTT_connect() {
233     int8_t ret;
234     // Stop if already connected.
235     if (mqtt.connected()) {
236         return;
237     }
238     Serial.println();
239     Serial.print("Connecting to MQTT.. ");

```





```

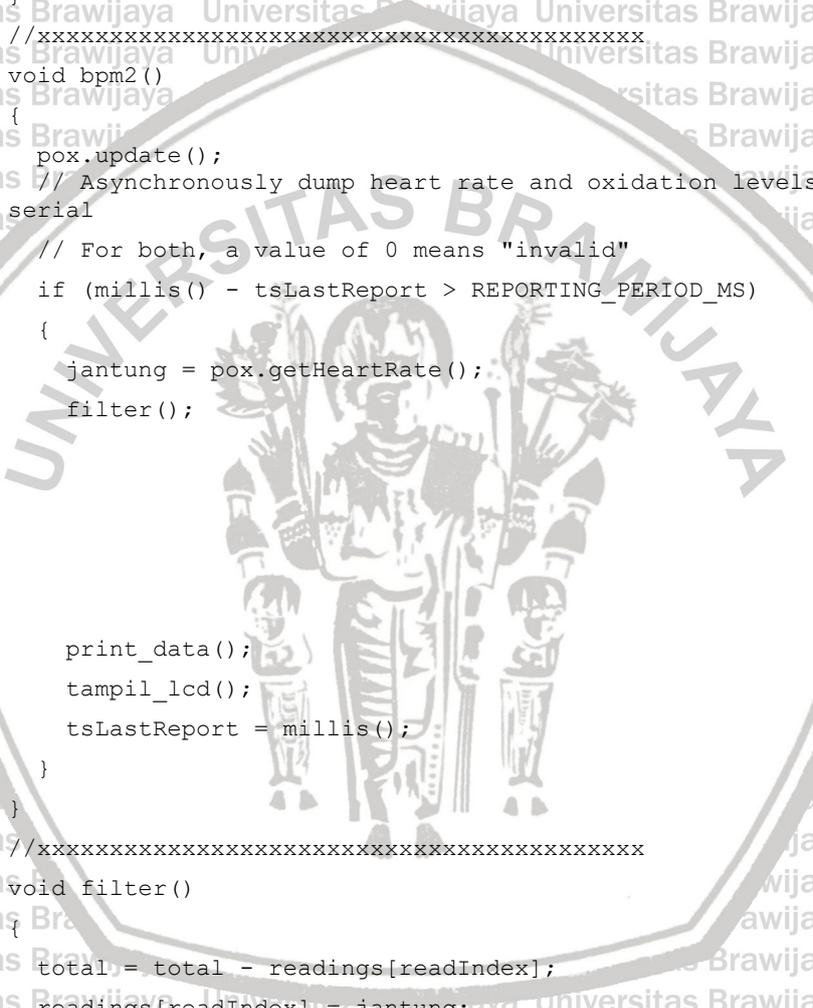
240
241 lcd.setCursor(9,0);
242 lcd.print("Upload...");
243
244 uint8_t retries = 3;
    while ((ret = mqtt.connect()) != 0) { // connect will return 0
245 for connected
246 Serial.println(mqtt.connectErrorString(ret));
247 Serial.println("Retrying MQTT connection in 5 seconds...");
248 mqtt.disconnect();
249 delay(5000); // wait 5 seconds
250 retries--;
251 bpm2();
252 if (retries == 0) {
253 // basically die and wait for WDT to reset me
254 while (1);
255 }
256 }
257 Serial.println("MQTT Connected!");
258 }
259 //xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
260 void mqtt_publish() {
261 detak = jantung;
262 MQTT_connect();
263 Serial.println(); Serial.println("Start publish");
264 temperature.publish(cel);
265 bpm.publish(average);
    Serial.println(); Serial.println("end off publish");
266 Serial.println();
267 }
268 //xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
269 void mqtt_subscribe() {
270 MQTT_connect();
271 Adafruit_MQTT_Subscribe *subscription;
272 while ((subscription = mqtt.readSubscription(5000))) {
273 Serial.println(); Serial.println(); Serial.println();
274 if (subscription == &onoffbutton) {
275 Serial.print(F("saklar 1: "));
276 Serial.println((char *)onoffbutton.lastread);
277 t1 = (char *)onoffbutton.lastread;
278 if (t1 == "ON") {
279 digitalWrite(9, 1);
280 } else {
281 digitalWrite(9, 0);
282 }
283 }

```

```

284 Serial.println(); Serial.println(); Serial.println();
285 }
286 }
287 //xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
288 void get_suhu() {
289     val = analogRead(tempPin);
290     mv = (val / 1580.0) * 5000;
291     cel = (mv / 10);
292     farh = (cel * 9) / 5 + 32;
293 }
294 //xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
295 void bpm2()
296 {
297     pox.update();
298     // Asynchronously dump heart rate and oxidation levels to the
299     serial
300     // For both, a value of 0 means "invalid"
301     if (millis() - tsLastReport > REPORTING_PERIOD_MS)
302     {
303         jantung = pox.getHeartRate();
304         filter();
305
306
307
308
309         print_data();
310         tampil_lcd();
311         tsLastReport = millis();
312     }
313 }
314 //xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
315 void filter()
316 {
317     total = total - readings[readIndex];
318     readings[readIndex] = jantung;
319     total = total + readings[readIndex];
320     readIndex = readIndex + 1;
321     if (readIndex >= numReadings)
322     {
323         readIndex = 0;
324     }
325     average = total / numReadings;
326 }

```



1-11	Inisialisasi bentuk tampilan display lcd
13-24	Inisialisasi awal
27-42	Inisialisasi wifi access point dan adafruit.io
46-57	Digunakan untuk menghubungkan MQTT dengan atribut inisialisasi adafruit.io
59-80	Digunakan untuk inisialisasi pengecekan detak jantung
83-158	Digunakan untuk memulai MQTT dengan menggunakan atribut inisialisasi wifi access point dengan cara pengecekan koneksi serta deteksi detak
161-169	Untuk mengecek waktu, suhu dan bpm secara terus-menerus
172-178	Menampilkan data detak dan suhu pada lcd
180-186	Menampilkan data bpm dan suhu
188-207	Digunakan untuk mendapatkan waktu
209-218	Menampilkan data yang diterima berdasarkan koneksi MQTT
220-227	method tidak terpanggil
231-258	Memulai koneksi dengan MQTT pika koneksi belum dimulai
260-267	Mengirim data berdasarkan koneksi MQTT
269-286	Menerima data berdasarkan koneksi MQTT
288-293	Mendapatkan data suhu
295-313	Mendapatkan data bpm
315-326	Menghitung nilai bpm

5.3.2 Implementasi Program Subscriber

Tabel 5. 6 Kode Program Subsciber

```

1      btnDetak = new javax.swing.JButton();
2      btnTemp = new javax.swing.JButton();
3      jLabel1 = new javax.swing.JLabel();
4
5      setDefaultCloseOperation(javax.swing.WindowConstants.EXIT
6      ON_CLOSE);
7      btnDetak.setText("DEKAT JANTUNG");
8      btnDetak.addActionListener(new
9      java.awt.event.ActionListener() {
10         public void
11         actionPerformed(java.awt.event.ActionEvent evt) {
12             btnDetakActionPerformed(evt);
13         }
14     });
15     btnTemp.setText("TEMPERATUR");
16     btnTemp.addActionListener(new
17     java.awt.event.ActionListener() {
18         public void
19         actionPerformed(java.awt.event.ActionEvent evt) {

```



```
17 btnTempActionPerformed(evt);
18 }
19 });
20
21 jLabel1.setFont(new java.awt.Font("Tahoma", 1,
22 14)); // NOI18N
23
24 jLabel1.setText("APLIKASI SUBSCRIBER DETAK
25 JANTUNG DAN TEMPERATUR");
26
27 javax.swing.GroupLayout layout = new
28 javax.swing.GroupLayout(getContentPane());
29 getContentPane().setLayout(layout);
30 layout.setHorizontalGroup(
31 layout.createParallelGroup(javax.swing.GroupLayout.Alignment
32 .LEADING)
33 .addGroup(layout.createSequentialGroup()
34 .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment
35 .CENTER)
36 .addComponent(btnDetak,
37 javax.swing.GroupLayout.PREFERRED_SIZE, 132,
38 javax.swing.GroupLayout.PREFERRED_SIZE)
39 .addGap(80, 80, 80)
40 .addComponent(btnTemp,
41 javax.swing.GroupLayout.PREFERRED_SIZE, 123,
42 javax.swing.GroupLayout.PREFERRED_SIZE)
43 .addGap(javax.swing.GroupLayout.DEFAULT_SIZE,
44 Short.MAX_VALUE))
45 .addGroup(layout.createSequentialGroup()
46 .addComponent(jLabel1)
47 .addGap(0, 0, Short.MAX_VALUE))
48 );
49 layout.setVerticalGroup(
50 layout.createParallelGroup(javax.swing.GroupLayout.Alignment
51 .TOP)
52 .addGroup(layout.createSequentialGroup()
53 .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment
54 .TOP)
55 .addGroup(layout.createSequentialGroup()
56 .addComponent(btnDetak,
57 javax.swing.GroupLayout.DEFAULT_SIZE, 65, Short.MAX_VALUE)
58 .addComponent(btnTemp,
59 javax.swing.GroupLayout.DEFAULT_SIZE,
60 javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))
61 .addGap(62, 62, 62)
62 .addComponent(jLabel1))
63 .addContainerGap(15, Short.MAX_VALUE))
64 );
65 pack();
66 setLocationRelativeTo(null);
67 // </editor-fold>
```

```

53 private void
54 btnDetakActionPerformed(java.awt.event.ActionEvent evt) {
55 // this.setVisible(false);
56 FrameHeart subs = new FrameHeart();
57 subs.setTitle("MQTT SUBSCRIBER DETAK JANTUNG");
58 subs.setVisible(true);
59 subs.runMQTT("hakim99/feeds/bpm");
60 }
61 private void
62 btnTempActionPerformed(java.awt.event.ActionEvent evt) {
63 // this.setVisible(false);
64 FrameTemp subs = new FrameTemp();
65 subs.setTitle("MQTT SUBSCRIBER TEMPERATUR");
66 subs.setVisible(true);
67 subs.runMQTT("hakim99/feeds/temperature");
68 }
69
70 /**
71 * @param args the command line arguments
72 */
73 public static void main(String args[]) {
74 /* Set the Nimbus look and feel */
75
76
77
78 */
79 try {
80 for (javax.swing.UIManager.LookAndFeelInfo info :
81 javax.swing.UIManager.getInstalledLookAndFeels()) {
82 if ("Nimbus".equals(info.getName())) {
83 javax.swing.UIManager.setLookAndFeel(info.getClassName()
84 );
85 break;
86 }
87 } catch (ClassNotFoundException ex) {
88 java.util.logging.Logger.getLogger(HakimProj.class.getName()).lo
89 g(java.util.logging.Level.SEVERE, null, ex);
90 } catch (InstantiationException ex) {
91 java.util.logging.Logger.getLogger(HakimProj.class.getName()).lo
92 g(java.util.logging.Level.SEVERE, null, ex);

```



```

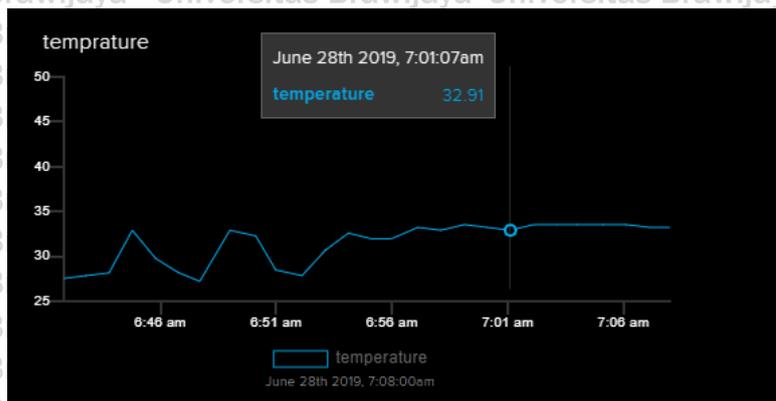
92     } catch
    (javax.swing.UnsupportedLookAndFeelException ex) {
93     java.util.logging.Logger.getLogger(HakimProj.class.getName()).lo
    g(java.util.logging.Level.SEVERE, null, ex);
94     }
95     //</editor-fold>
96
97     /* Create and display the form */
98     java.awt.EventQueue.invokeLater(new Runnable() {
99     public void run() {
100     new HakimProj().setVisible(true);
101     }
102     });
103 }
    
```

1-3	Inisialisasi teks label serta tombol detak dan suhu
7-13	Berguna untuk trigger tombol detak jantung
14-19	Berguna untuk trigger tombol temperatur
21-22	Kegunaannya untuk mengatur jenis font dan isi teks label
24-48	Memberikan pengaturan bentuk tampilan UI
54-60	Mendapatkan data detak jantung dengan memanggil fungsi MQTT
62-68	Mendapatkan temperatur dengan memanggil fungsi MQTT
79-103	Memulai dan menampilkan tampilan UI

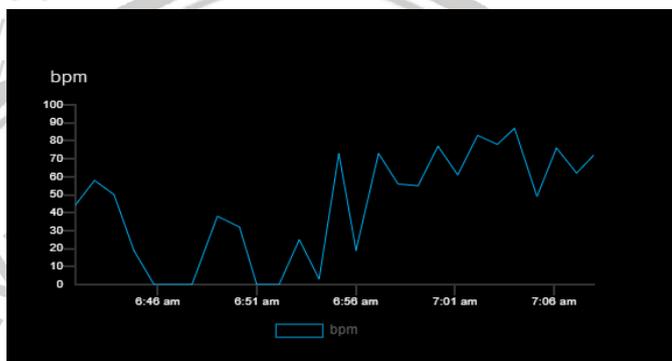
5.3.2.1 Implementasi Antarmuka

Implementasi antarmuka memiliki fungsi yaitu pengguna bisa melakukan pemantauan untuk sensor node melewati protocol komunikasi mqtt yang telah di sediakan oleh cloud adafruit dan dapat diakses melalui web browser. Di bawah ini adalah tampilan untuk antarmuka yang ditujukan pada Gambar 5.11.





Gambar 5. 13 Implementasi Antarmuka suhu pada Adafruit



Gambar 5. 14 Implementasi Antarmuka Detak Jantung pada Adafuit

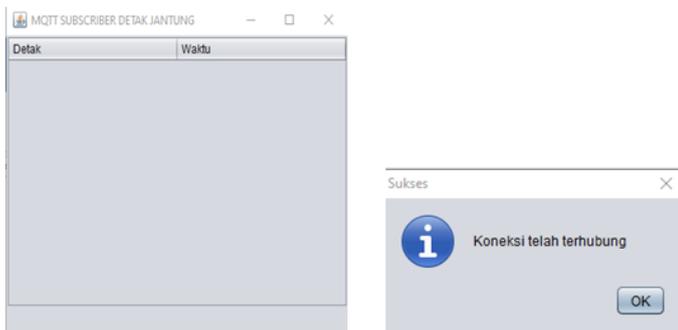
5.3.2.2 Implementasi Antarmuka Perangkat lunak Subscriber

Pada bagian implementasi antarmuka perangkat lunak subscriber dibuat menggunakan bahasa pemrograman *Javascript* untuk dapat membuat komunikasi protocol mqtt antara broker dan pengguna. Dibawah ini adalah tampilan antarmuka perangkat lunak subscriber yang ditujukan pada Gambar 5.13.



Gambar 5. 15 Implementasi Antarmuka Perangkat lunak Subscriber

Gambar diatas ialah antarmuka dari perangkat lunak subscriber yang belum kita klik button detak jantung maupun temperatur. Hasil dari jikalau kita klik salah satu dari dua button diatas dapat dilihat di Gambar 5.14.



Gambar 5. 16 Implementasi Perangkat Lunak Subscriber(Lanjutan)

Gambar 5.14 menunjukkan hasil program setelah kita klik button dari salah satu menu detak jantung atau temperatur. Dapat dilihat ada kolom detak dan waktu yang berfungsi memberikan hasil detak jantung dan waktu pengkoleksian data yang diambil setiap satu menit sekali.



BAB 6 PENGUJIAN DAN ANALISIS

Dalam bab ini berisi penjelasan tentang pengujian kepada sistem yang telah dikerjakan. Pengujian ini bermaksud mengetahui performa dari sistem tersebut apakah sudah sesuai dengan yang dibutuhkan.

6.1 Prosedur dan Tujuan Pengujian Fungsional

Prosedur dan tujuan pengujian fungsional yaitu mengetahui apakah sistem yang telah dibuat mampu melakukan yang diinginkan penulis secara baik dan benar, sehingga mampu memberi hasil yang diharapkan.

Tabel 6. 1 Skenario Pengujian Fungsional

No.	Deskripsi Pengujian	Skenario Pengujian
SPF_01	Pengujian sensor node	<ol style="list-style-type: none"> Melakukan komunikasi dengan broker Mendeteksi data sensor yang masuk dan memproses data
SPF_02	Pengujian aplikasi pengguna <i>Subscriber</i> live sensor node	<ol style="list-style-type: none"> Pengguna melakukan subscribe live sensor Broker menanggapi sehingga menampilkan data live sensor

6.1.2 Hasil dan Analisis Pengujian Fungsional

Dapat dilihat dibawah ini pada table 6.2 adalah hasil dari pengujian fungsional yang memiliki kaidah berhasil dan gagal. Berhasil jika fitur-fitur yang dibuat mampu berjalan semestinya dan tanpa ada error.

Tabel 6. 2 Hasil Pengujian Fungsional

No.	Deskripsi Pengujian	Skenario Pengujian	Hasil
SPF_001	Pengujian sensor node	<ol style="list-style-type: none"> Melakukan komunikasi dengan broker Mendeteksi data sensor yang masuk dan memproses data 	Berhasil
SPF_002	Pengujian aplikasi user <i>Subscriber</i> live sensor node	<ol style="list-style-type: none"> Pengguna melakukan subscribe live sensor Broker menanggapi sehingga menampilkan data live sensor 	Berhasil

6.2 Pengujian Sensor Node

Pengujian yang kedua yaitu pengujian sensor node yaitu dengan cara memulai menjalankan sistem yang kegunaannya agar mendapatkan informasi detak jantung dan suhu tubuh. Pengujian ini dibagikan menjadi dua yaitu, pengujian sensor detak jantung Max3010 dan sensor suhu Lm35.

6.2.1 Tujuan Pengujian

Tujuan dari pengujian pada sensor node tersebut agar dapat memantau dan memberikan nilai yang sama persis sehingga diambil rata-rata detak jantung dan suhu tubuh dengan sensor yang akan diuji kan.

6.2.2 Prosedur Pengujian Sistem

Dibawah ini adalah proses untuk melakukan pengujian sensor node.

1. Pastikan sensor node sudah di rancang dengan benar dan kondisi terkoneksi di computer
2. Melakukan *publish* ke *cloud* atau broker melewati *browser* computer pengguna, agar dapat memantau dan mendapatkan informasi detak jantung dan suhu tubuh
3. Melakukan penghitungan rata-rata dari hasil pemantauan sehingga dapat dibandingkan dengan perangkat lunak di jam pintar.

6.2.3 Hasil Pengujian

Pengujian yang di lakukan pada perangkat sensor node dikerjakan dalam pengimplementasian pemantauan detak jantung dan suhu tubuh. Pada pengujian tersebut dilakukan agar mendapatkan nilai rata-rata dan keakuratan dalam mendapatkan data dari sensor detak jantung dan suhu tubuh. Pengujian ini telah terbagi menjadi dua yaitu, pengujian sensor detak jantung dan suhu tubuh. Pada bagian ini, perangkat sensor node kita bandingkan pada erangkat lunak jam pintar dan pengambilan rata-rata.

1. Pengujian Sensor Detak Jantung MAX3010

Pengujian perangkat detak jantung adalah pengujian sensor MAX3010 yang kegunaannya mendapatkan nilai detak jantung pada jari. Pada proses pengujian sensor ini dilakukan supaya mampu mandapatkan nilai yang sama persis pada detak jantung dan menghitung rata-rata yang didapatkan dari sensor dengan alat pembanding lainnya. Hasil pengujian sensor detak jantung MAX3010 dapat dilihat di table berikut.

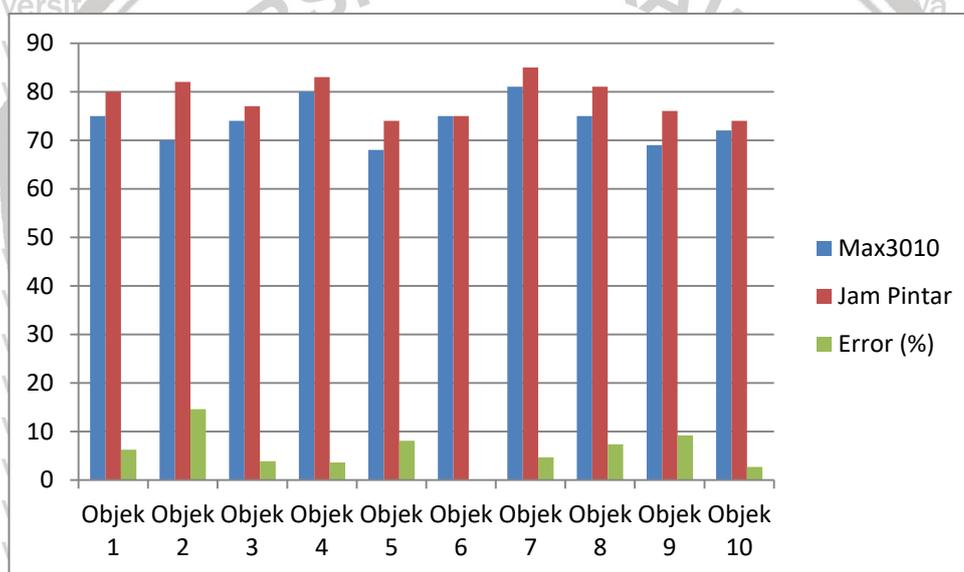
Tabel 6. 3 Hasil dari Pengujian Sensor Detak Jantung Max3010

No	Nama	Max3010	Jam Pintar	Error (%)



1	Objek 1	75	80	6,3
2	Objek 2	70	82	14,6
3	Objek 3	74	77	3,9
4	Objek 4	80	83	3,6
5	Objek 5	68	74	8,1
6	Objek 6	75	75	0,0
7	Objek 7	81	85	4,7
8	Objek 8	75	81	7,4
9	Objek 9	69	76	9,2
10	Objek 10	72	74	2,7
Rata-rata Error (%)				6,1

Dari table diatas berhasil didapatkan dari pengujian, hasil dari perbandingan nilai Max3010 yang telah diimplementasikan dan dibandingkan dengan hasil yang didapatkan dengan jam pintar dapat di aplikasikan dalam bentuk grafik yang tunjukkan pada gambar 6.1.



Gambar 6.1 Grafik Data Perbandingan Program Sistem dengan Jam pintar

Dapat dilihat dari grafik di Gambar 6.1 diatas, bahwa grafik perbandingan nilai detak jantung yang dilakukan memakai sistem dengan alat detak jantung lainnya. Hasil diatas dapat diartikan data yang telah di hasilkan dari sistem mendapatkan hasil yang tidak jauh berbeda dengan penghitungan menggunakan perangkat lunak Jam pintar. Program sistem detak jantung MAX3010 mengkoleksi data setiap satu menit sekali dalam satuan bpm. Presentase keakuratan pada sensor masih memiliki kesalahan yaitu 6.1%, namun masih bisa dikatakan sistem berjalan sesuai yang di inginkan.

2. Pengujian Lm35 sensor suhu

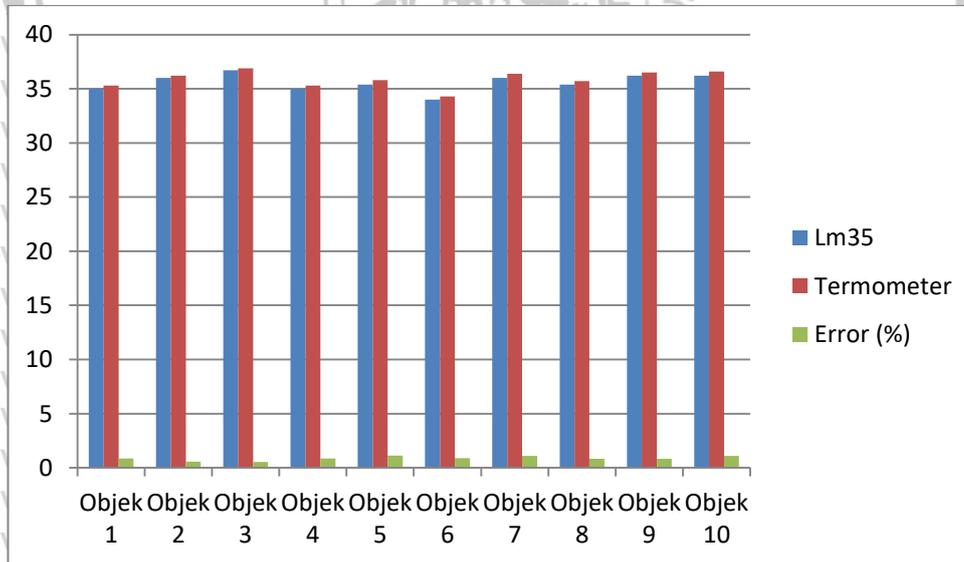
Pengujian perangkat sensor suhu tubuh LM35 yang kegunaanya mendapatkan nilai yang diperoleh dari suhu tubuh. Pada pengujian sensor ini

dilakukan supaya mampu mendapatkan nilai yang sama persis pada pada suhu tubuh dan menghitung rata-rata yang didapatkan dari sensor dengan termometer. Hasil pengujian sensor suhu tubuh LM35 dapat dilihat di table berikut.

Tabel 6. 4 Hasil Pengujian Sensor Lm35

No	Nama	Suhu tubuh		Error (%)
		Lm35	Termometer	
1	Objek 1	35	35,3	0,8
2	Objek 2	36	36,2	0,6
3	Objek 3	36,7	36,9	0,5
4	Objek 4	35	35,3	0,8
5	Objek 5	35,4	35,8	1,1
6	Objek 6	34	34,3	0,9
7	Objek 7	36	36,4	1,1
8	Objek 8	35,4	35,7	0,8
9	Objek 9	36,2	36,5	0,8
10	Objek 10	36,2	36,6	1,1
Rata-rata Error (%)				0,9

Dari table diatas berhasil diperoleh dari perbandingan nilai sensor Lm35 yang telah diimplementasikan dan dibandingkan dengan hasil termometer dapat di aplikasikan kedalam grafik yang berada gambar 6.2



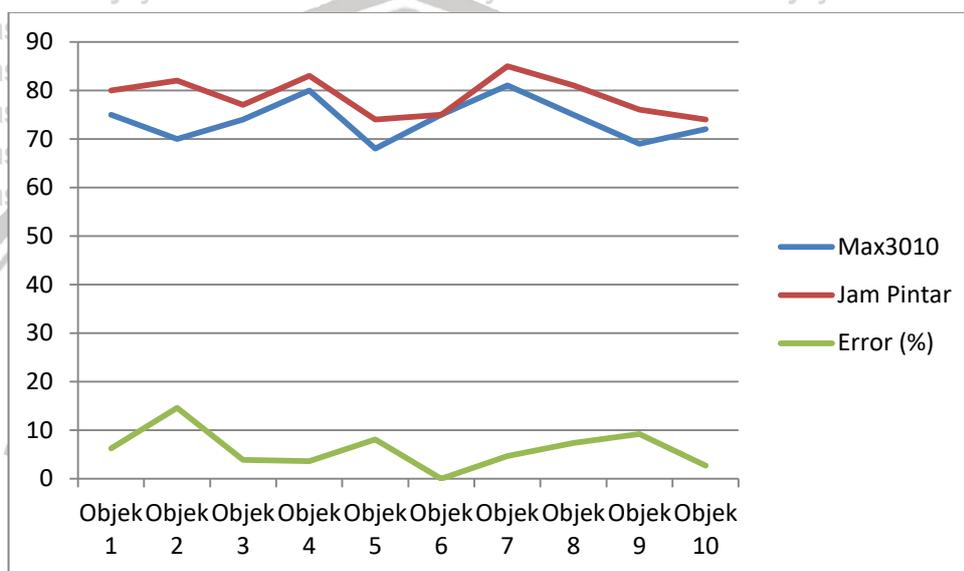
Gambar 6.2 Grafik Data Perbandingan Program sistem dengan Termometer

Menurut Gambar 6.2 diatas, bahwa grafik data perbandingan nilai suhu tubuh yang terapkan memakai sistem dengan termometer. Hasil diatas dapat diartikan bahwa nilai yang di hasilkan dari sistem mendapatkan hasil yang tidak jauh berbeda dengan penghitungan dengan termometer. Program dari LM35 mengkoliksi data setiap satu menit sekali dalam satuan derajat celcius. Dapat dilihat dari gambar tersebut bahwa, presentase keakuratan pada sensor masih

memiliki kesalahan yaitu 0.9%, namun masih bisa dikatakan sistem berjalan sesuai yang di inginkan, dan menghasilkan rata-rata eror yang kurang dari 1%.

6.2.4 Analisis Pengujian

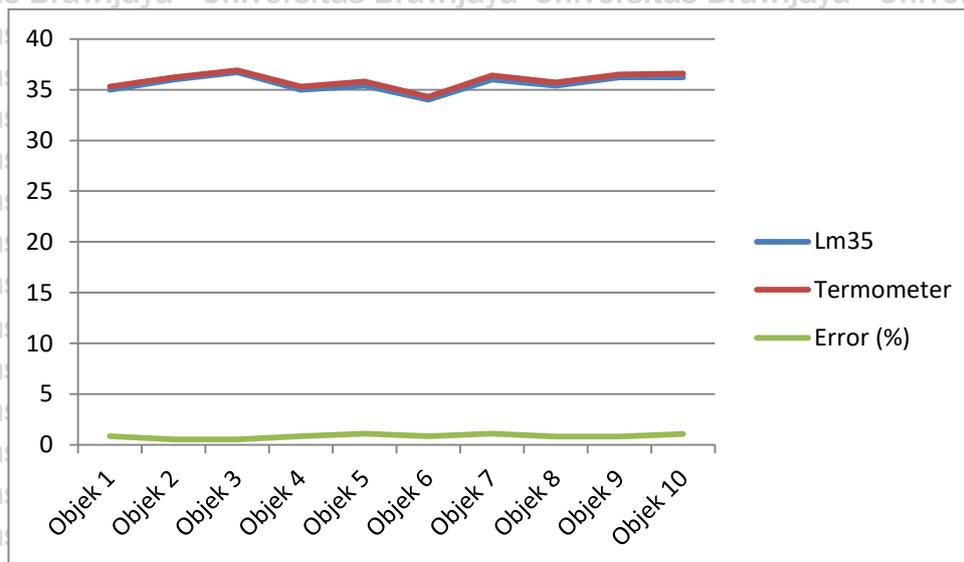
Sistem pemantauan detak jantung dan suhu tubuh berhasil di rancang dan mampu berjalan sesuai yang diinginkan sebelumnya, oleh sebab itu akan diberikan hasil analisis hasil pengujian yang usai dilkerjakan. Dibawah ini adalah hasil dari analisis pengujian perangkat sensor node dalam berupa grafik yang ditunjukkan pada gambar 6.3.



Gambar 6.3 Grafik perbandingan Program sistem dengan Jam pintar

Gambar 6.3 diatas adalah grafik pengaplikasian dari hasil yang dibandingkan sensor Max3010 yang sudah dirancang dengan perangkat lunak jam pintar. Dari grafik tersebut kita dapat menfsirkan bahwa nilai dari pebandingan keduanya memiliki nilai yang hampir mendekati sama dengan rata-rata atau *error rate* sebesar 6,1% dengan 10 kali percobaan dengan pengambilan data 1 menit sekali untuk ditampilkan. Untuk melakukan penghitungan *error rate* dengan cara di bawah ini.

$$\frac{(\text{Pengukuran Manual} - \text{Pengukuran Sensor})}{\text{Pengukuran Manual}} \times 100\% \quad (6.3)$$



Gambar 6.4 Grafik Perbandingan Sistem dengan Termometer

Gambar 6.4 tersebut adalah grafik pengaplikasian dari hasil perbandingan antara sensor Lm35 suhu yang sudah dirancang dengan alat termometer. Dari grafik garis di atas kita dapat menafsirkan bahwa nilai dari kedua alat mampu memiliki nilai yang hampir mendekati sama dengan rata-rata atau *error rate* 0.9% dengan 10 kali percobaan dengan pengambilan data 1 menit sekali untuk ditampilkan. Untuk melakukan penghitungan *error rate* dengan cara di bawah ini.

$$\frac{(\text{Pengukuran Manual} - \text{Pengukuran Sensor})}{\text{Pengukuran Manual}} \times 100\% \quad (6.4)$$

Dari hasil kedua pengukuran tersebut, dapat disimpulkan bahwa perancangan dari kedua alat tersebut sudah mampu memenuhi target yang diinginkan pengguna, yaitu mampu mengirimkan data detak jantung dan suhu tubuh yang akurat dengan mempunyai kesalahan pada sensor Max3010 6,1% dan pada sensor Lm35 0,9%. Dalam hasil pengujian ini dapat disimpulkan bahwa banyak faktor-faktor yang mempengaruhi terjadinya kesalahan, diantaranya kualitas dan tegangan dari komponen itu sendiri kurang maksimal.

6.3 Pengujian Kinerja sistem

Pengujian kinerja sistem yaitu penulis akan melakukan pengujian terhadap sistem untuk memberikan hasil yang diminta oleh pengguna. Perangkat lunak yang digunakan dalam pengujian kinerja sistem ini yaitu Wireshark. Pengujian ini dilakukan kepada 5 pengguna yang melakukan percobaan bersama-sama. Adapun bagian-bagian dalam yang akan dikerjakan dalam pengujian kinerja sistem ini yaitu tujuan pengujian, prosedur pengujian, hasil dan analisis pengujian.



6.3.1 Tujuan Pengujian

Tujuan dari pengujian ini adalah mengetahui hasil perhitungan waktu yang diberikan oleh sistem untuk mengirimkan paket dari yang diminta oleh subscriber dalam satu kali proses transmisi. Tujuan dari pengujian ini yaitu untuk mengetahui performa dari protokol mqtt pada sistem pemantauan ini yang berbasis publish dan subscribe.

6.3.2 Prosedur Pengujian

Dibawah ini adalah proses untuk melakukan pengujian performa

1. Dilakukan pengambilan data selama 5 kali dalam waktu setiap menit
2. Melakukan perbandingan setelah mendapatkan hasil dari sistem yang telah dicoba

6.3.3 Hasil Pengujian

Pengujian performa ini dikerjakan karena sesuai dengan prosedur yang telah dipaparkan pada bab sebelumnya, maka hasil yang didapat dari pengujian ini berada dibawah ini.

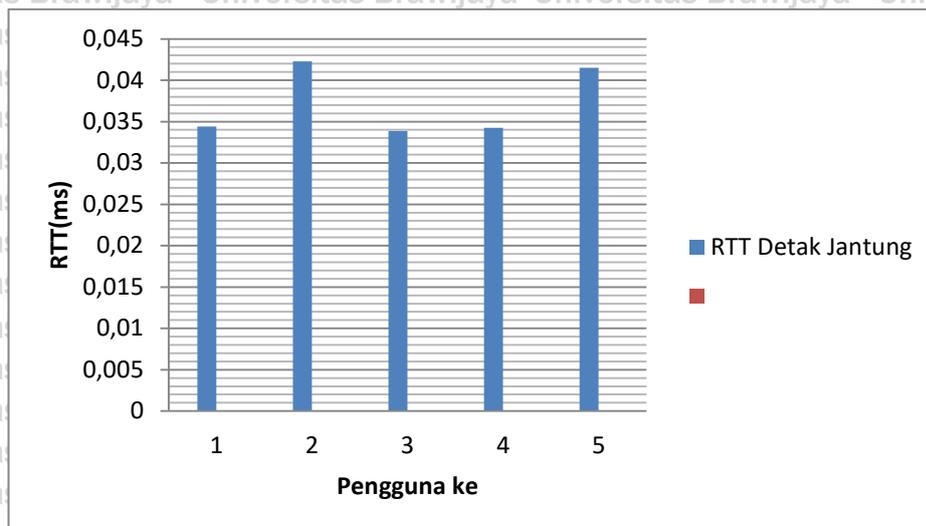
1. Pengujian proses mensubscribe data dengan waktu 5 menit pada detak jantung

Pada bagian pengujian performa yang dilakukan pertama adalah pada subscriber detak jantung. Pengujian ini agar mendapatkan nilai dari *Round Trip Time* (RTT) pada protokol mqtt. Dibawah ini adalah hasil dari pengujiannya yang ditujukan tabel 6.5.

Tabel 6. 5 Hasil Pengujian performa detak jantung

Pengguna ke	RTT Detak Jantung(ms)
1	0,034389918
2	0,042261105
3	0,033862239
4	0,034234477
5	0,041520713
Rata-rata	0,03725369

Tabel diatas adalah hasil dari perhitungan pengambilan data secara bersama-sama dengan waktu 5 menit dengan 5 kali percobaan untuk menguji performan dari sistem pemantauan detak jantung. Dari tabel diatas dapat ditunjukkan dengan grafik dibawah ini.



Gambar 6. 5 Grafik Hasil pengujian performa sistem

Gambar 6.5 adalah grafik proses transmisi data detak jantung yang telah di subscribe dengan waktu 5 menit. Dibawah ini adalah rumus mencari hasil RTT

$$\text{EstimatedRTT} = (1 - \alpha) \cdot \text{EstimatedRTT} + \alpha \cdot \text{SampleRTT} \quad (6.5)$$

Dari grafik diatas dapat dilihat bahwa pengambilan data selama 5 menit mempunyai nilai RTT dibawah 1 menit dan pada pengujian pengambilan data selama 5 menit sempat terjadi penurunan performa pada saat melakukan pengiriman 5 kali secara bersamaan.

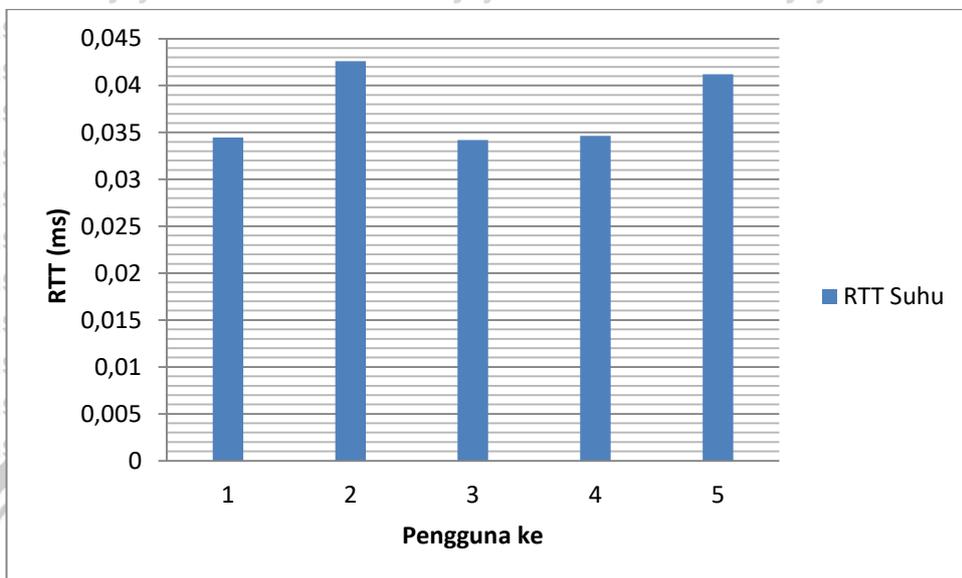
2. Pengujian proses mensubscribe data dengan waktu 5 menit pada suhu tubuh

Pada bagian pengujian performa yang dilakukan pertama adalah pada subscriber suhu tubuh. Pengujian ini agar mendapatkan nilai dari Round Trip Time (RTT) pada protokol mqtt. Dibawah ini adalah hasil dari pengujiannya yang ditujukan tabel 6.6

Tabel 6. 6 Hasil pengujian performa suhu tubuh

Pengujian ke	RTT Suhu (ms)
1	0,034472
2	0,042608
3	0,034185
4	0,034637
5	0,041202
Rata-rata	0,037421

Tabel diatas adalah hasil dari perhitungan pengambilan data secara bersama-sama dengan waktu 5 menit dengan 5 kali percobaan untuk menguji performa dari sistem pemantauan suhu tubuh. Dari tabel diatas dapat ditunjukkan dengan grafik dibawah ini.

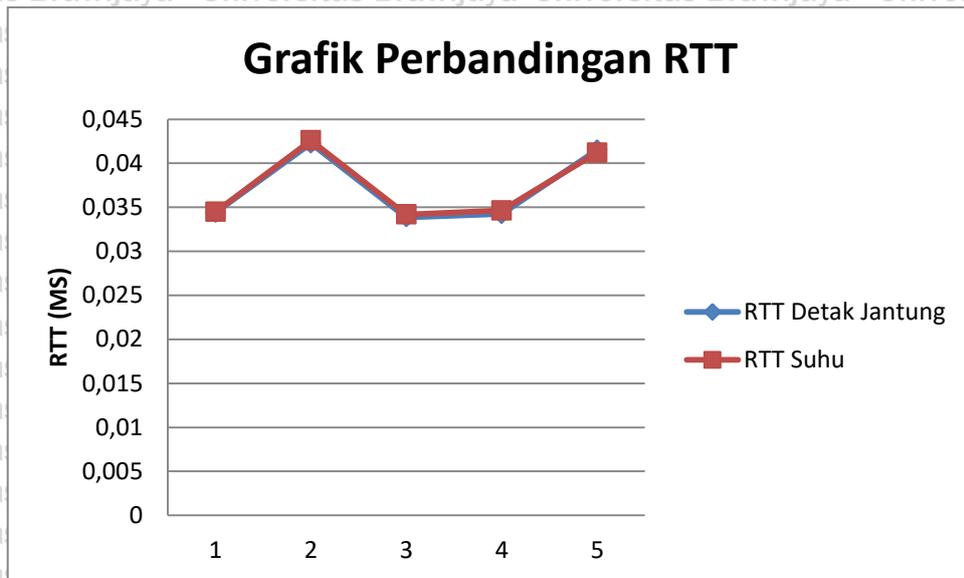


Gambar 6. 6 Grafik hasil pengujian performa sistem

Gambar 6.5 adalah grafik proses transmisi data suhu tubuh yang telah di subscribe dengan waktu 5 menit. Dari grafik diatas dapat dilihat bahwa pengambilan data selama 5 menit mempunyai nilai RTT dibawah 1 menit dan pada pengujian pengambilan data selama 5 menit sempat terjadi penurunan performa pada saat melakukan pengiriman 5 kali secara bersamaan.

6.3.4 Analisis Pengujian

Setelah berhasil melakukan pengujian, selanjutnya melakukan analisis terhadap pegujian yang berhasil dilakukan. Analisis pengujian sangat dibutuhkan karena untuk mengetahui kinerja sistem detak jantung dan suhu tubuh dan protokol mqtt sebagai komunikasi pertukaran data.



Gambar 6. 7 Grafik Perbandingan RTT detak jantung dan suhu tubuh

Gambar 6.7 tersebut adalah grafik pengaplikasian dari hasil perbandingan pengujian pada saat mensubscribe data. Dari grafik garis diatas kita dapat menafsirkan bahwa nilai dari kedua pengujian mendapatkan nilai RTT yang tidak begitu jauh berbeda. Dari kedua pengujian diatas, sistem mampu memiliki nilai RTT kurang dari satu menit ketika mensubscribe data.

$$\text{EstimatedRTT} = (1 - \alpha) \cdot \text{EstimatedRTT} + \alpha \cdot \text{SampleRTT} \quad (6.7)$$

Dari hasil kedua pengujian tersebut, dapat disimpulkan bahwa pengujian yang telah dilakukan saat melakukan subscribe data yang ke 5 mengalami peningkatan. peningkatan tersebut di karenakan protocol TCP memiliki *flow control* yang dapat memberi tempat kosong buffer ketika sudah terisi penuh.

BAB 7 KESIMPULAN DAN SARAN

Kesimpulan dan saran berdasarkan hasil implementasi dan pengujian yang telah dilaksanakan oleh peneliti.

7.1 Kesimpulan

Berdasarkan hasil dari bab-bab sebelumnya dalam perancangan sistem yang sudah dikerjakan oleh penulis, maka dapat ditarik kesimpulan:

1. Cara kerja dari sistem detak jantung dan suhu tubuh dilakukan dengan cara menempelkan ibu jari ke atas sensor MAX3010 dan menggenggam sensr suhu tubuh LM35 sehingga data detak jantung dapat dihasilkan dalam satuan *bpm* (*beats per minute*) dan suhu tubuh dalam satuan derajat *celcius* dengan menggunakan sensor MX3010 dan Lm35 sehingga pengkoleksian nilai data diambil setiap satu menit sekali.
2. Untuk memublish Node sensor mengirim data detak jantung dan suhu tubuh menggunakan protocol komunikasi *MQTT* mengirimkan dengan menjalankan node sensor ke *cloud* atau broker menggunakan jaringan *Wi-Fi* dan dapat di subscribe oleh pengguna.
3. Bahwa perangkat sensor node yang telah dirancang, memiliki tingkat kesalahan dalam melakukan pemantauan yaitu 0,9% untuk suhu tubuh dan 6.1% untuk detak jantung. Dan performa protokol komunikasi *MQTT* dalam penghitungan waktu pengiriman perintah untuk mensubscribe data dari broker menggunakan jaringan *wi-fi* membutuhkan waktu rata-rata kurang dari 1 menit.

7.2 Saran

Berdasarkan hasil dari penarikan kesimpulan yang telah dibuat, maka penulis mempunyai saran untu penelitian yang akan datang.

1. Sistem pemantauan detak jantung dan suhu tubuh ini bisa dikembangkan dengan menambahkan parameter penyakit yang diderita pasien
2. Penelitian selanjutnya mampu membangun system dengan protocol komunikasi yang lain
3. Dalam sisi pengujian dapat ditambahkan indikator uji yang lain, seperti fitur keamanan

DAFTAR REFERENSI

- Anggraini, N. (2015). RANCANG BANGUN ALAT PEREKAM FREKUENSI LISTRIK JANTUNG. *JURNAL TEKNIK INFORMATIKA*, 8(2).
- area, M. (2018). *Getting Started with Adafruit Circuit Playground Express: The Multipurpose Learning and development Board with Built in LEDs, Sensos, Accelerometer*. Canada: Maker Media.
- Dejana Ugrenovic, G. G. (2015). CoAP protocol for Web-based monitoring in IoT healthcare applications. *Telecommunications forum TELFOR 2015*, 24-26.
- Durga Amanarth, D. S. (2017). Design and Implementation of Smart HealthCare. *International Conference on Innovations in Information, Embedded and Communication Systems (ICIIECS)*, 1-7.
- Gibson, J. (2002). *Fisiologi dan Anatomi Modern untuk Perawat*. Jakarta: Buku Kedokteran EGC.
- Hillar, G. C. (2017). *MQTT Essentials - A Lightweight IoT Protocol*. Birmingham: Packt Publishing Ltd.
- Hindarto, I. A. (2015). APLIKASI PENGUKUR DETAK JANTUNG MENGGUNAKAN SENSOR PULSA. *Simposium Nasional Teknologi Terapan*.
- Kabo, P. D. (2008). *Mengungkap Pengobatan Penyakit Jantung Koroner*. Jakarta: PT Gramedia Pustaka Utama.
- Pendidikan, D. (2014). *Dosenpendidikan.com*. Retrieved April 2, 2018, from <http://www.dosenpendidikan.com/pengertian-jantung-dan-fungsinya-pada-manusia/>
- PUTRI, D. M. (2017). *ilmuti.org*. Retrieved April 3, 2018, from <http://ilmuti.org/2017/02/23/mengenal-wemos-d1-mini-dalam-dunia-iot/>
- Quamila, A. (2017). *hellosehat*. Retrieved april 2, 2018, from <https://hellosehat.com/hidup-sehat/tips-sehat/berapa-suhu-tubuh-normal/>
- RIYANTO, E. (2016). PERANCANGAN PENGUKUR DETAK JANTUNG DAN SUHU TUBUH.
- Shariman Ismadi Ismail, N. S. (2016). *Proceedings of the 2nd International Colloquium on Sports Science, Exercise, Engineering and Technology 2015*. Malaysia: Springer science+business media singapore.
- Solichin, D. I. (2017). MONITORING DETAK JANTUNG, SUHU TUBUH, DAN RESPIRASI. 1-10.
- Subhash Balla, V. B. (2017). *Intelligent Computing and Information and*

Communication: Proceedings. Singapore: Springer Nature.

Valerie Lampkin, w. t. (2012). *Building Smarter Planet* (1st ed.). U.S: WebSphere MQ.

Wirawan, G. S. (2017). SISTEM PEMANTAU SUHU LAB JARAK JAUH BERBASIS ARDUINO. *Seminar Nasional Vokasi dan Teknologi (SEMNASVOKTEK)*.

Zennifa, F. (2013). Prototipe Alat Deteksi Dini dan Mandiri Penyakit Jantung Menggunakan Sistem Pakar VCSIR, Arduino dan Handphone Android.

