

# Sistem monitoring denyut jantung berbasis IoT menggunakan protokol XMPP

Dany Salsabila<sup>1</sup>, Ahmad Tri Hanuranto<sup>2</sup>, Arif Indra Irawan<sup>3\*</sup>

<sup>1,2,3</sup>Program Studi Teknik Telekomunikasi, Universitas Telkom

Jl. Telekomunikasi No. 1, Terusan Buahbatu, Kabupaten Bandung, Jawa Barat 40257, Indonesia

<sup>1</sup>danysalsabila123@gmail.com, <sup>2</sup>athanuranto@telkomuniversity.ac.id, <sup>3\*</sup>arifirawan@telkomuniversity.ac.id

## ABSTRAK

Kardiovaskular merupakan suatu penyakit gangguan pada jantung. Penyakit Kardiovaskular dapat didiagnosis menggunakan alat yang bernama Elektrokardiogram (EKG). Dengan menggunakan teknologi revolusi *Internet of Things* (IoT), tujuan penelitian ini adalah merancang dan menguji sebuah sistem monitoring denyut jantung berbasis IoT menggunakan protokol *Extensible Messaging and Presence Protocol* (XMPP). Sistem dirancang dengan mengintegrasikan sebuah modul AD8232 yang dihubungkan dengan server XMPP melalui mikrokomputer Raspberry Pi dan akan dihubungkan dengan antarmuka *website*. Antarmuka ini dapat menampilkan hasil deteksi *beat per minute* (BPM), waktu pengiriman, dan waktu terima pesan XMPP berdasarkan hasil deteksi sensor AD8232. Pada penelitian ini dilakukan pengukuran performansi berdasarkan *Quality Of Service* (QoS) menggunakan Wireshark. Sedangkan untuk nilai BPM digunakan untuk mengukur tingkat akurasi sensor AD8232 yang dibandingkan dengan Oximeter sebagai referensi. Hasil implementasi dan pengukuran pada penelitian ini diperoleh *delay* 342,857 ms dan *throughput* 0,0009257 bps yang termasuk dalam kategori baik dalam sistem IoT. Hasil akurasi perbandingan data BPM antara sensor AD8232 dan Oximeter 59,918% dengan biaya implementasi yang murah namun dapat diakses di banyak perangkat dengan berbagai macam bentuk sistem monitor.

**Kata kunci:** *Extensible Messaging and Presence Protocol* (XMPP), Raspberry Pi, sensor AD8232, BPM, *Quality of Service* (QoS)

## ABSTRACT

*Cardiovascular is a disease of the heart. Cardiovascular disease can be diagnosed using a device called an electrocardiogram (ECG). Using the Internet of Things (IoT) revolution technology, the purpose of this research is to design and test an IoT-based heart rate monitoring system using the Extensible Messaging and Presence Protocol (XMPP) protocol. The system is designed by integrating an AD8232 module that is connected to the XMPP server via a Raspberry Pi microcomputer and will be connected to a website interface. This interface can display the results of the detection of beats per minute (BPM), the time of sending, and the time of receiving XMPP messages based on the detection results of the AD8232 sensor. In this research, performance measurement is based on Quality Of Service (QoS) using Wireshark. Meanwhile, the BPM value is used to measure the accuracy of the AD8232 sensor which is compared with the Oximeter as a reference. The results of the implementation and measurement in this study obtained a delay of 342.857 ms and a throughput of 0.0009257 bps which is included in the good category in the IoT system. The results of the BPM data comparison accuracy between the AD8232 sensor and the Oximeter 59.918% with a low implementation cost but can be accessed on many devices with various forms of monitor systems.*

**Keywords:** *Extensible Messaging and Presence Protocol* (XMPP), Raspberry Pi, AD8232 sensor, BPM, *Quality of Service* (QoS)

## 1. PENDAHULUAN

Saat ini banyak kasus kematian yang disebabkan oleh penyakit jantung yakni Kardiovaskular. Kardiovaskular merupakan suatu penyakit pada ketidakstabilan pada denyut jantung yang menimbulkan komplikasi penyakit lain hingga pasien dapat meninggal dunia secara tiba-tiba. Menurut data WHO, kematian akibat Kardiovaskular tahun 2019 di Asia Tenggara memiliki risiko rata-rata di atas 5% yang dominan diderita lansia di atas 40 tahun [1]. Penanganan yang dapat dilakukan adalah pencegahan dengan sistem monitor denyut jantung menggunakan alat yakni Elektrokardiogram

(EKG). Dengan fitur yang dimiliki oleh EKG, gejala Kardiovaskular dapat didiagnosis ketidaknormalannya melalui penggambaran dengan gelombang dan banyak denyut per menit (BPM). Pengambilan data EKG dilakukan dengan menempelkan beberapa elektroda di tubuh manusia yang mengandung Ag/ACL. Namun, saat ini untuk melakukan diagnosis Kardiovaskular masih dominan dilakukan dengan EKG yang berada di rumah sakit meskipun terdapat alat yang ditempel di jari yakni Oximeter. Alat ini hanya terbatas pada hasil deteksi yang divisualisasi pada satu monitor dan dominan hanya digunakan oleh individu pasien. Sehingga untuk deteksi dini penyakit Kardiovaskular perlu ditingkatkan [2], [3]. Selain itu, terdapat sebuah teknologi yang memungkinkan benda-benda mati dapat dihubungkan ke internet untuk dimonitor dan dikontrol secara otomatis yakni *Internet of Things* (IoT). Teknologi ini menjadi salah satu peluang jangka panjang untuk dikembangkan supaya menjadi solusi yang efektif dalam menyelesaikan permasalahan yang ada atau menjadi perkembangan dari teknologi sebelumnya [4]. Oleh karena itu, dengan teknologi IoT bisa menjadi solusi untuk membantu meningkatkan diagnosis penyakit Kardiovaskular kedepannya.

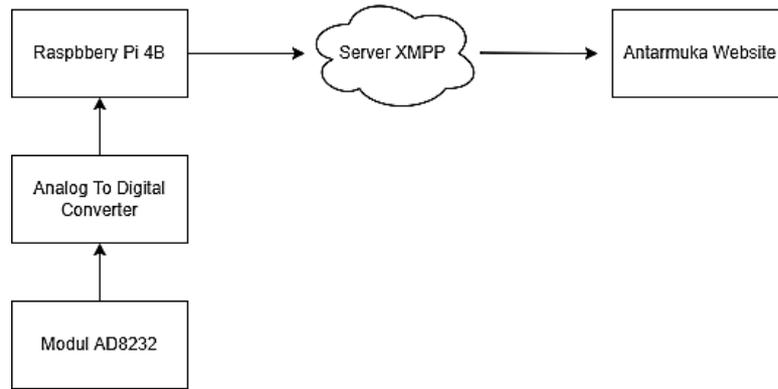
Salah satu penelitian melakukan perancangan sistem *real-time* EKG pada sebuah perangkat Android. Koneksi yang digunakan untuk mikrokontroler dan ponsel adalah Bluetooth sehingga memiliki keterbatasan pada jarak maksimal dari kedua perangkat tersebut [5]. Kemudian penelitian kedua merancang sebuah sistem IoT dengan protokol XMPP untuk memonitor kesehatan secara heterogen seperti EKG, tekanan darah, suhu, dll. XMPP merupakan sebuah protokol pengiriman pesan secara *real-time* dengan basis *server-client* yang berjalan di atas protokol *transmission control* (TCP). Implementasi sistem ini untuk membantu dokter dalam memonitor kondisi pasien dari jarak jauh menggunakan aplikasi Facebook. Namun, sistem ini dikhususkan di lingkungan rumah sakit untuk kalangan dokter dan perawat [6]. Penelitian selanjutnya dirancang sebuah sistem EKG menggunakan modul AD8232 khusus EKG yang dihubungkan ke mikrokontroler Arduino. Perangkat tersebut dihubungkan ke modul ESP8266/NodeMCU untuk mendukung koneksi melalui WiFi. Hasil deteksi sinyal EKG ditampilkan pada basis *web* dan LCD TFT [7]. Yang terakhir terdapat sebuah penelitian yang membandingkan antara protokol *constrained application* (CoAP) dan *Message Queuing Telemetry Transport* (MQTT) pada teknologi IoT yang diaplikasikan ke EKG. Untuk menampilkan datanya menggunakan antarmuka *web* dengan HTTP *restful* API. CoAP berjalan di atas protokol *user data* (UDP) yang bersifat *real-time*. Sedangkan untuk MQTT berjalan di atas protokol TCP dengan sistem pengiriman data *sync/ack* [8].

Berdasarkan beberapa hasil penelitian yang ada, penulis merancang sebuah sistem monitoring denyut jantung berbasis IoT dengan protokol XMPP. Penelitian ini bertujuan untuk membuat suatu sistem monitoring denyut jantung yang dapat diakses menggunakan *website* untuk membantu penderita Kardiovaskular dalam memonitor aktivitas denyut jantung secara mandiri tanpa harus datang ke rumah sakit.

## 2. METODE PENELITIAN

### 2.1 Desain Sistem

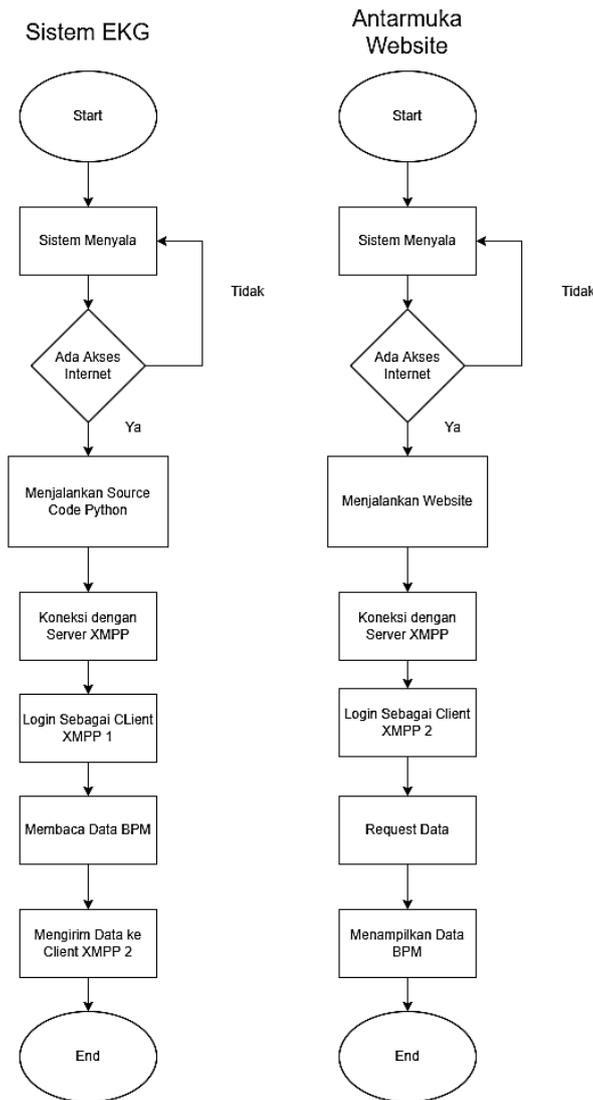
Gambar 1 menunjukkan diagram blok sistem secara keseluruhan. Sistem EKG terdiri dari beberapa komponen yakni tiga buah sensor elektroda, modul AD8232, MCP3008, dan Raspberry Pi. Sensor berupa elektroda ditempelkan pada anggota tubuh akan mendeteksi denyut jantung sebagai data *input* dan diproses pada modul AD8232 yakni ekstraksi, penguatan, dan pemfilteran sinyal hasil deteksi [9]. Setelah data diproses, data akan dikirimkan ke *server* XMPP oleh Raspberry Pi. *Server* XMPP digunakan sebagai pusat komunikasi antara sistem EKG dengan *website* dan sebagai penyimpanan secara *real-time* data hasil denyut jantung. *Server* akan memperbarui data apabila terdapat data *input* dari sistem EKG. Selain itu, *server* juga mengirimkan data hasil denyut jantung ke *website* ketika *website* melakukan *request* data. *Website* berfungsi untuk menampilkan hasil data denyut jantung berupa data *beat per minute* berdasarkan data yang disimpan secara *real-time* di *server* XMPP.



**Gambar 1. Diagram blok sistem monitoring denyut jantung**

**2.2 Diagram Alir Sistem**

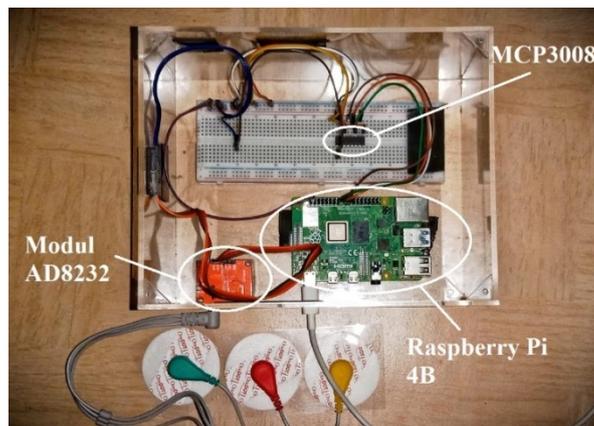
Gambar 2 menjelaskan diagram alir pada alat EKG dan antarmuka *website*. Pada alat yang dibuat, Raspberry Pi menjalankan *source code* Python dengan *login* ke *server* sebagai *client* XMPP 1 dan selanjutnya mengambil data BPM dari modul AD8232. *Website* juga melakukan *login* ke *server* sebagai *client* XMPP 2 lalu melakukan *request* data. Data hasil deteksi kemudian dikirim oleh Raspberry Pi ke *website* melalui perantara komunikasi *client* XMPP 1 dengan *client* XMPP 2.



**Gambar 2. Diagram alir sistem**

### 2.3 Perangkat Keras

Gambar 3 merupakan rangkaian komponen yang digunakan pada sistem EKG yang terdiri dari tiga komponen utama yakni modul AD8232, MCP3008, dan Raspberry Pi. Modul AD8232 berfungsi sebagai modul utama untuk sensor Elektrokardiogram. Modul AD8232 terhubung dengan 3 buah elektroda yang ditempel di bagian dada manusia. Modul AD8232 mendeteksi sinyal analog dari Ag-ACL pada elektroda. Modul ini membutuhkan tegangan 3V yang dikeluarkan oleh Raspberry Pi. Kemudian sinyal analog terlebih dahulu dikonversi ke digital menggunakan modul MCP3008 karena Raspberry Pi tidak bisa mendeteksi sinyal analog. Raspberry Pi berperan dalam menyimpan data sinyal EKG dan mengirim data ke *server* XMPP. Raspberry Pi diatur menjadi *client* XMPP.



Gambar 3. Bentuk fisik sistem

Pada penelitian ini digunakan beberapa perangkat keras dengan spesifikasi yang tertera pada Tabel 1.

Tabel 1. Spesifikasi perangkat keras

No.	Perangkat keras	Spesifikasi
1	Modul AD8232	<ul style="list-style-type: none"> <li>• Tegangan 3 V</li> <li>• 3 sensor elektroda</li> <li>• Indikator LED</li> </ul>
2	Raspberry Pi 4B	<ul style="list-style-type: none"> <li>• 64-bit Quad-core Cortex-A72 <i>processor</i></li> <li>• 4GB RAM</li> <li>• 2 port Micro HDMI</li> <li>• 2 port USB 2.0</li> <li>• 2 port USB 3.0</li> <li>• Port Ethernet Gigabit</li> <li>• Bluetooth 5.0</li> <li>• Port 5V/3A USB tipe C</li> </ul>
3	MCP3008	<ul style="list-style-type: none"> <li>• Analog to Digital Converter (ADC)</li> </ul>
4	Kabel Jumper	<ul style="list-style-type: none"> <li>• Male to Female</li> <li>• Male to Male</li> <li>• Female to Female</li> </ul>



Gambar 4. Penempatan elektroda modul AD8232

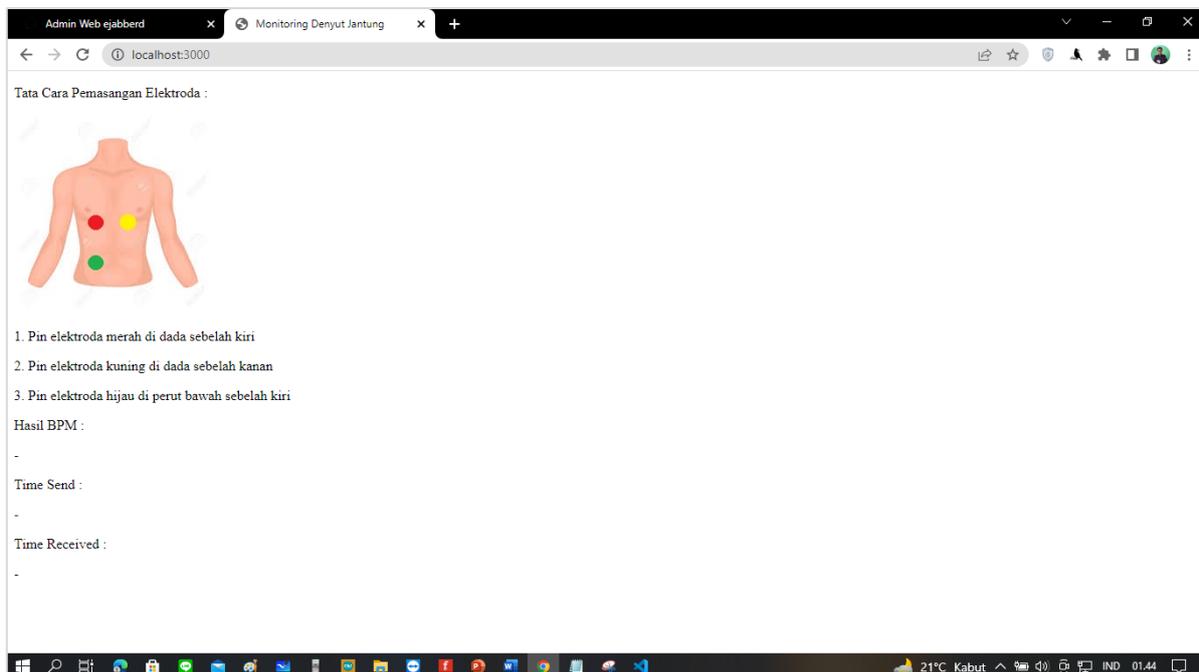
Gambar 4 menunjukkan tata cara pemasangan elektroda berdasarkan warna pin. Pada modul AD8232 pin elektroda warna kuning adalah (+) yang diletakkan di bagian dada kiri. Kemudian pin elektroda warna merah (-) diletakkan di bagian dada kanan. Elektroda hijau sebagai referensi diletakkan pada perut bawah sebelah kiri. Tabel 2 menunjukkan perbandingan antara alat yang dibuat dengan Oximeter.

**Tabel 2. Perbandingan modul AD8232 dan Oximeter**

Kategori	Sistem dengan Modul AD8232	Oximeter
Kecepatan pendeteksian	Protokol XMPP minimal 30 detik	1 detik
Sistem Monitoring	<ul style="list-style-type: none"> <li>• Serial plotter</li> <li>• LCD</li> <li>• Website</li> <li>• Aplikasi</li> <li>• Dapat diakses di banyak perangkat</li> </ul>	Layar monitor
Data yang dapat diukur	<ul style="list-style-type: none"> <li>• BPM</li> <li>• Plot grafik denyut jantung</li> </ul>	<ul style="list-style-type: none"> <li>• BPM</li> <li>• Kadar oksigen</li> </ul>
Estimasi biaya	Rp. 90.000	Rp. 100.000

## 2.4 Antarmuka Website

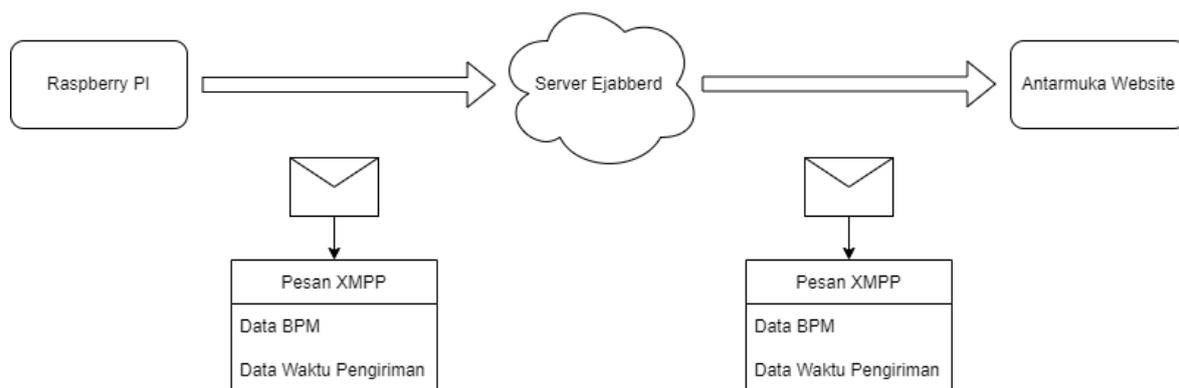
Gambar 5 adalah *website* sebagai visualisasi hasil monitoring denyut jantung. *Website* menampilkan tata cara pemasangan elektroda disertai dengan gambar realisasi pada dada manusia. Untuk hasil data yang ditampilkan adalah hasil BPM yang merupakan hasil deteksi yang dikirimkan melalui XMPP. Lalu *time send* merupakan hasil data pembacaan waktu pengiriman Raspberry Pi ke *website*. Dan yang terakhir adalah *time received* yakni hasil pembacaan waktu terima pesan XMPP oleh *website*.



**Gambar 5. Antarmuka website**

Antarmuka *website* dibuat menggunakan layanan Node.js sebagai sumber daya. *Website* menggunakan *express* sebagai *server* untuk menyediakan beberapa *route* untuk menjalankan tugas sebagai visualisasi hasil monitor denyut jantung. Beberapa fungsi *route* yang digunakan adalah sebagai berikut:

- a. *Route “/”*  
Route ini adalah sumber daya yang diakses oleh *website* pertama kali. Ketika *website* mengakses, maka fungsi *route “/”* akan memberikan respon dengan menjalankan *file index.html* yang berisi susunan *body* dari *website* itu sendiri, menampilkan teks dan gambar sesuai perintah, dan menampilkan data BPM, *time send*, dan *time receive* yang dibaca dan dioleh oleh Node.js.
- b. *Route “/bpm”*  
Route “/bpm” berperan dalam dua tugas utama yakni, memberikan informasi hasil pesan XMPP yang berupa data BPM dan *time send* yang telah di-request dari server XMPP, serta memberikan informasi hasil pembacaan *time receive* oleh Node.js.
- c. *Route “/ecg”*  
Route ini adalah khusus berperan untuk menampilkan data gambar penempatan elektroda dada manusia.



Gambar 6. Skenario pengiriman pesan XMPP

Gambar 6 menjelaskan skenario pengiriman data yang akan dilakukan untuk pengujian performansi QoS dan BPM. Raspberry Pi mengambil dan mengirim data dari modul AD8232 dan membaca waktu pengiriman. Kemudian data waktu pengiriman akan dikirimkan secara bersamaan dengan data BPM dalam bentuk pesan XMPP.

### 3. HASIL DAN PEMBAHASAN

#### 3.1 Skenario Pengujian

Skenario pengujian *Quality of Service* (QoS) pada penelitian ini adalah melakukan pengukuran performansi berdasarkan ITU-T Rec. G.1010 yang merupakan standar pengukuran QoS untuk pengiriman data seperti audio, video, teks, gambar, dll. Parameter yang diukur terdiri dari *delay*, *delay variation*, dan *information loss* [10]. Pengujian dilakukan sebanyak 10 kali pengiriman data dengan jeda waktu 30 detik setiap pengiriman. Sehingga untuk pengukuran *delay* adalah waktu dari Raspberry Pi mulai untuk mengirimkan data hingga antarmuka *website* menerima data tanpa menghitung waktu jeda 30 detik setiap pengiriman. Kemudian skenario pengujian BPM adalah membandingkan data menggunakan modul AD8232 sebagai detektor utama dan Oximeter sebagai detektor pembanding. Skema pengambilan data modul AD8232 dan Oximeter dipasang secara bersamaan dan pada waktu yang sama. Pengujian dilakukan sebanyak 50 kali pengambilan data dengan jeda waktu 30 detik untuk waktu pendeteksian sensor di modul AD8232.

#### 3.2 Hasil Pengujian QoS

Tabel 3 menunjukkan hasil ukur QoS untuk parameter *delay*, *throughput*, dan *packet loss*. Untuk pengujian QoS, TCP Dump terlebih dahulu dijalankan pada Raspberry Pi. Kemudian Raspberry Pi melakukan pengiriman data sehingga TCP Dump membaca setiap data yang masuk dan keluar pada Raspberry Pi. Kemudian pembacaan parameter QoS adalah dengan hasil visualiasi TCP Dump menggunakan Wireshark.

Tabel 3. Hasil ukur QoS

Pengiriman data ke-	Parameter		
	<i>Delay (ms)</i>	<i>Throughput (bps)</i>	<i>Packet loss</i>
1	351,011	0,00088	0
2	325,095	0,000947	0
3	301,94	0,001023	0
4	319,804	0,000966	0
5	316,443	0,000973	0
6	338,994	0,000912	0
7	307,283	0,001009	0
8	293,867	0,001051	0
9	446,471	0,000694	0
10	386,747	0,000802	0

Berdasarkan hasil pengujian, diperoleh parameter rata-rata *delay* yang didapatkan adalah 342,857 ms dari beberapa variasi *delay* yang didapatkan. Variasi *delay* dapat disebabkan oleh kecepatan internet yang tidak konstan. Kemudian *throughput* rata-rata yang dihasilkan adalah 0,0009257 bps untuk sekali pengiriman data BPM rata-rata sebesar 309,1 byte dengan *packet loss* pada XMPP adalah 0 karena berbasis pengiriman TCP.

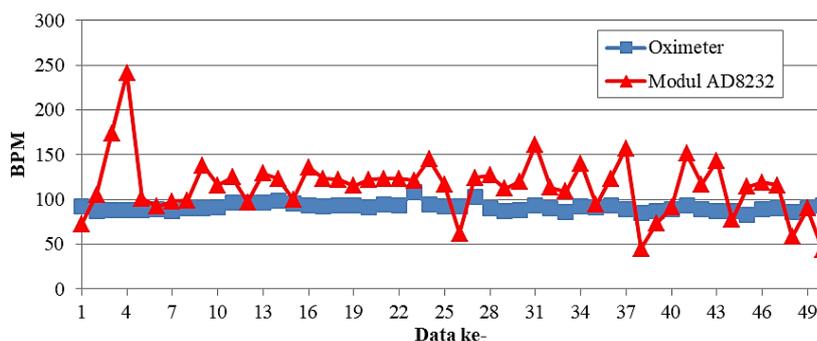
### 3.3 Hasil Pengujian BPM

Hasil pengujian BPM dapat dilihat pada Tabel 4. Untuk skenario pengujian BPM dengan menggunakan sensor AD8232 sebagai detektor utama dan Oximeter sebagai detektor pembanding. Modul AD8232 membutuhkan waktu 30 detik untuk pengambilan data. Dari 50 hasil data yang didapat, setiap pengambilan data BPM dari sensor AD8232 merupakan data yang berhasil dikirimkan dan ditampilkan di *website*. Dan hasil data BPM dari Oximeter yang diambil adalah waktu yang sama setelah 30 detik pendeteksian BPM dari sensor AD8232. Hasil perbandingan dengan Oximeter kemudian diukur tingkat akurasinya menggunakan *Root Mean Squared Root (RMSE)* [11] menggunakan persamaan berikut.

$$RMSE = \sqrt{\frac{(Data\ yang\ diukur - Data\ pembanding)^2}{Banyaknya\ data}} \times 100\% \quad (1)$$

$$Tingkat\ Akurasi = 100\% - RMSE \quad (2)$$

Adapun hasil perbandingan antara alat yang dibuat dengan Oximeter dapat dilihat pada Gambar 7. Berdasarkan hasil pengujian diperoleh nilai RMSE sebesar 40,062% sehingga didapatkan tingkat akurasi 59,918%. Perbedaan pada hasil perbandingan dengan oximeter sebagai detektor referensi untuk sensor AD8232 dapat disebabkan oleh beberapa hal yakni untuk sensor AD8232 merupakan *board* paling sederhana yang terdiri dari tiga elektroda dengan biaya yang cukup terjangkau. Kemudian elektroda dari sensor harus rutin untuk diganti agar mendapatkan nilai yang lebih akurat.



Gambar 7. Hasil perbandingan BPM Oximeter dan alat yang telah dibuat

#### 4. KESIMPULAN

Sistem monitoring denyut jantung berbasis IoT menggunakan protokol telah dirancang dan diuji berdasarkan parameter QoS. Berdasarkan hasil pengujian diperoleh *delay* pengiriman memiliki rata-rata 342,857 ms, *throughput* 0,0009257 bps, dan *packet loss* 0. Implementasi sistem monitoring denyut jantung menggunakan protokol XMPP ini dirancang tanpa menggunakan *database* dan dapat berjalan dengan baik dengan catatan data yang dikirimkan adalah BPM. Karena XMPP memiliki maksimal besar data yang dapat dikirimkan adalah 32 KB, sehingga XMPP cukup baik untuk sebagai salah satu protokol di IoT yang dapat berfungsi tanpa *database* tambahan. Dengan menggunakan modul AD8232 memungkinkan sebuah sistem monitoring denyut jantung berbasis IoT dapat dimonitor dengan banyak perangkat melalui protokol XMPP meskipun tingkat akurasi masih harus ditingkatkan. Pada penelitian selanjutnya dapat diterapkan metode untuk mengatasi akurasi dari keluaran BPM sensor AD8232 ini, misalnya dengan menerapkan regresi maupun algoritma filter.

#### REFERENSI

- [1] World Health Organization, "WHO updates Cardiovascular Risk Charts," 2019.
- [2] E. Nemati, M. J. Deen, and T. Mondal, "A wireless wearable ECG sensor for long-term applications," *IEEE Communications Magazine*, vol. 50, no. 1, pp. 36-43, 2012.
- [3] S. Lopez, "Pulse Oximeter Fundamentals and Design," Free. Semicond. Inc., pp. 1-39, 2012 [Online]. Available: [http://cache.freescale.com/files/32bit/doc/app\\_note/AN4327.pdf](http://cache.freescale.com/files/32bit/doc/app_note/AN4327.pdf).
- [4] A. H. Ngu, M. Gutierrez, V. Metsis, S. Nepal, and Q. Z. Sheng, "IoT Middleware: A Survey on Issues and Enabling Technologies," *IEEE Internet of Things Journal*, vol. 4, no. 1, pp. 1-20, 2017.
- [5] B. Mishra, N. Arora, and Y. Vora, "A Wearable Device for Real-Time ECG Monitoring and Cardiovascular Arrhythmia Detection for Resource Constrained Regions," *2018 8th International Symposium on Embedded Computing and System Design (ISED)*, Cochin, India, 2018, pp. 48-52.
- [6] S. R. Pawara, "Heterogeneous Health Monitoring System Using XMPP-Design and Implementation," *2017 Second International Conference on Electrical, Computer and Communication Technologies (ICECCT)*, Coimbatore, 2017, pp. 1-5.
- [7] P. Kamble and A. Birajdar, "IoT Based Portable ECG Monitoring Device for Smart Healthcare," *2019 Fifth International Conference on Science Technology Engineering and Mathematics (ICONSTEM)*, Chennai, India, 2019, pp. 471- 474.
- [8] I. Kassem and A. Sleit, "Elapsed Time of IoT Application Protocol for ECG: A Comparative Study Between CoAP and MQTT," *2020 International Conference on Electrical, Communication, and Computer Engineering (ICECCE)*, Istanbul, Turkey, 2020, pp. 1-6.
- [9] Analog Devices, "AD8232 Single-Lead ECG," Data Sheet, pp. 1-28, 2013, [Online]. Available: [www.analog.com/AD8232](http://www.analog.com/AD8232).
- [10] ITU-T, "G.1010: End-user multimedia QoS categories," Int. Telecommun. Union, vol. 1010, 2001, [Online]. Available: [http://scholar.google.com.au/scholar?hl=en&q=ITU-T+Recommendation+G.1010&btnG=&as\\_sdt=1,5&as\\_sdtp=#7](http://scholar.google.com.au/scholar?hl=en&q=ITU-T+Recommendation+G.1010&btnG=&as_sdt=1,5&as_sdtp=#7).
- [11] N. J. Salkind, "Root Mean Square Error," *Encyclopedia of Research Design*. 2010.