

## RANCANG BANGUN ALAT PENDETEKSI TIDUR MENGUNAKAN SENSOR *HEART RATE* PADA *SECURITY GUARD* BERBASIS IOT (*INTERNET OF THINGS*)

Freean Asvian<sup>1)</sup> Pauladie Susanto<sup>2)</sup> Weny Indah Kusumawati<sup>3)</sup>

Program Studi/Jurusan Teknik Komputer  
Institut Bisnis dan Informatika Stikom Surabaya  
Jl. Raya Kedung Baruk 98 Surabaya, 60298

Email: 1)[Freeanasvian@gmail.com](mailto:Freeanasvian@gmail.com), 2)[Pauladie@stikom.edu](mailto:Pauladie@stikom.edu), 3)[Weny@stikom.edu](mailto:Weny@stikom.edu)

**Abstrak:** Satpam merupakan suatu kelompok petugas yang dibentuk oleh badan usaha yang bertujuan untuk melakukan keamanan fisik dalam rangka mengamankan aset dan properti perusahaan. Untuk memastikan keamanan dan mencegah kerugian maupun kerusakan yang disengaja maupun tidak disengaja. Dalam menjaga aset suatu perusahaan, Satpam harus serius dan tidak boleh tertidur karena Satpam memiliki peran utama dalam menjaga keamanan aset suatu perusahaan dan instansi. Pada penelitian ini penulis membuat rancang bangun alat pendeteksi tidur menggunakan sensor *Heart Rate* pada *security guard* berbasis IoT (*Internet of Things*). Tujuannya agar Satpam lebih fokus dalam bertugas menjaga aset dan properti perusahaan/instansi. Pada rancang bangun tersebut telah ditetapkan penulis bahwa detak jantung orang yang normal sekitar 60-120 Bpm. Orang yang beraktivitas pada umumnya sekitar (detak jantung > 80) sedangkan orang yang beristirahat sekitar (detak jantung < 80), sehingga penulis menetapkan apabila Satpam dengan detak jantung < 80 maka alat mendeteksi tertidur. Nilai detak jantung dan keterangan tertidur atau tidak, dikirimkan ke *cloud* Firebase sebagai monitoringnya. Hasil pengujian yang diperoleh dengan membandingkan Satpam tertidur dengan yang beraktivitas. Dari 6 sampel untuk HR yang tertidur < 80 Bpm dan yang beraktivitas > 80 Bpm. Pengiriman data dari sensor *node* menuju *cloud* Firebase memiliki *delay* dengan rata-rata *delay* 65.9555 detik.

**Kata Kunci:** *Heart rate*, *security guard*, Firebase, Detak Jantung.

### PENDAHULUAN

Pekerja keamanan (*security guard*) atau yang biasa dikenal dengan Satpam adalah suatu kelompok petugas yang dibentuk oleh instansi/badan usaha/proyek untuk melakukan keamanan fisik dalam rangka mengamankan aset, instansi, proyek bangunan, properti atau tempat pemantauan peralatan untuk memastikan keamanan dan mencegah kerugian maupun kerusakan yang disengaja. Satpam memiliki peran yang utama dalam menjaga aset milik perusahaan/intansi, apabila seorang Satpam tidak serius dalam pekerjaannya maka perusahaan/intansi mendapatkan kerugian. Bukan hanya perusahaan/intansi yang membutuhkan Satpam. Universitas, sekolah, tempat ibadah dll, juga membutuhkannya demi menjaga aset-aset

yang berharga. Khususnya Satpam yang bertugas pada malam hari sangat rawan dengan tertidur.

Untuk mengawasi agar Satpam tidak tertidur dan fokus dalam kerjanya, pimpinan perusahaan/intansi yang bersangkutan harus memonitoring Satpamnya agar tidak terjadi hal yang tidak diinginkan. Biasanya seorang pimpinan cukup melihat rekaman dari CCTV, tetapi dengan keterbatasan CCTV yang tidak bisa semua tempat dapat di *record* dan tidak bisa di monitoring secara *realtime* dari jauh maka pengawasan tersebut kurang efisien.

Selain CCTV, ada penelitian yang mengawasi kerja Satpam. Perusahaan itu membutuhkan pengontrolan yang teratur dalam mengawasi ruangan. Alat tersebut dinamakan cek point Satpam dalam penerapan alat tersebut Satpam diberikan *remote control* untuk memasukan ID-nya ketika melakukan pengontrolan ruangan. Laporan

tersebut dimonitoring disuatu ruangan yang khusus, ruangan mana dan jam berapa Satpam melakukan pengontrolan. (AhmadSofyan.2006).

Dalam penerapan IoT telah banyak diterapkan pada sistem monitoring dan pengontrolan. Semua sistem yang menggunakan IoT pasti terkoneksi pada internet dengan menggunakan *cloud* sebagai komunikasinya, seperti Firebase. Fitur yang dimiliki oleh Firebase tergolong lengkap, seperti *Realtime Database* fitur ini menyediakan pemrograman antarmuka yang bisa disinkronkan dengan beberapa *client* yang tersimpan dalam *cloud* Firebase.

Berdasarkan permasalahan diatas, penulis memberikan solusi bagaimana cara mengatasi permasalahan tersebut, dengan cara menerapkan sistem berbasis IoT (*Internet of Things*) agar sensor *node* bisa dibaca dan dapat dimonitoring dari jarak yang jauh, sehingga informasi tentang status Satpam tertidur atau tidak bisa dimonitoring dan di *record*. Dalam sensor *node* terdapat outputan *buzzer* untuk membangunkan Satpam agar tidak tertidur.

## METODE PENELITIAN

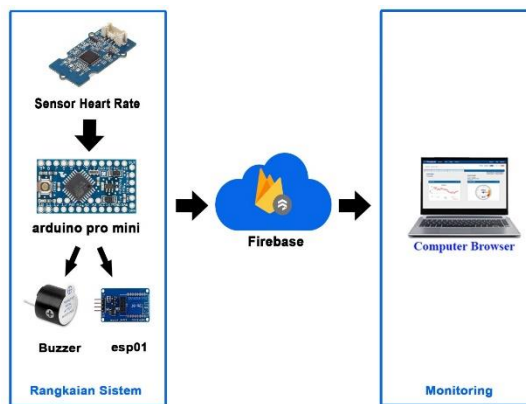
Rancang bangun alat pendeteksi tidur pada *Security Guard* berbasis IoT (*Internet of Things*) bertujuan untuk pengawasan pada Satpam yang notabennya bekerja pada suatu perusahaan dengan pengawasan yang extra. Yang tidak memiliki waktu untuk tidur dalam berjaga. Rancangan ini memiliki beberapa komponen alat diantaranya:

1. Arduino Pro Mini
2. Sensor Heart Rate
3. ESP8266
4. Buzzer

Alat-alat tersebut adalah pendukung terciptanya rancang bangun pendeteksi tidur pada Satpam. Untuk mengetahui Satpam tertidur atau tidak bisa dilihat dari detak jantungnya. Orang yang beraktivitas detak jantungnya cenderung lebih besar dari pada orang yang beristirahat. Dalam jurnal (Sandi, 2013) dijelaskan bahwa normalnya orang memiliki detak jantung 60-120 Bpm. Adapun orang yang beraktivitas memiliki detak jantung >80 Bpm sedangkan orang yang beristirahat memiliki detak jantung < 80. Penulis menyimpulkan bahwasanya seseorang dikatakan tertidur apabila detak jantungnya < 80. Metode ini hanya berlaku pada orang yang sehat, dikarenakan ada beberapa faktor yang dapat mempengaruhi kinerja detak jantung salah satunya memiliki penyakit dalam, diabetes menurut (Hidayah, 2012) bisa mengakibatkan detak jantungnya lebih rendah dari

normalnya orang. Hepatitis menurut (South et al, 2014) bisa mengakibatkan detak jantung lebih tinggi dari normalnya orang. Untuk hasil yang *maximal* penulis menyimpulkan bahwasanya dalam penerapan alat ini dikhususkan untuk orang yang normal dan tidak memiliki penyakit dalam.

Dalam membahas bagaimana proses komunikasi sensor *node* dengan *cloud* Firebase yang diterapkan maka penulis membagi menjadi beberapa bagian.

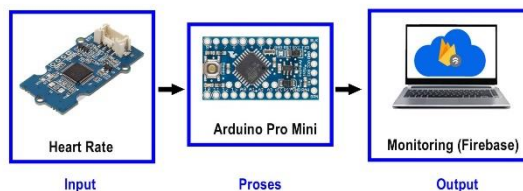


Gambar 1. Diagram blok sistem utama

Pada gambar 1 dijelaskan bahwa sensor *heart rate* dihubungkan dengan Arduino Pro Mini, untuk mengkoneksikan ke internet dibutuhkan modul ESP8266. Data yang dikirim oleh *Heart Rate* diproses Arduino Pro Mini dan diteruskan ke *cloud* Firebase melalui ESP8266. Monitoringnya menggunakan desktop atau smartphone dengan cara masuk ke akun Firebase. *Buzzer* sebagai outputan untuk membangunkan Satpam yang tertidur.

### A. Perancangan Perangkat Keras

Sistem deteksi tidur dengan sensor *Heart Rate* memiliki beberapa bagian utama yaitu *input*, proses dan *output*.

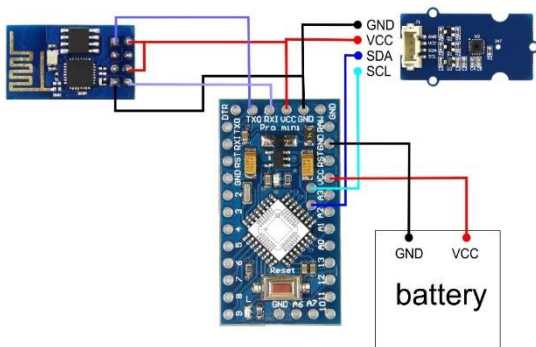


Gambar 2. Diagram blok perangkat keras alat pendeteksi tidur

Berikut penjelasan setiap bagian dari gambar 2 diatas.

1. **Input.**  
Sensor *Heart Rate*, sebagai sensor untuk mengambil data denyut jantung Satpam.
2. **Proses.**  
Arduino Pro Mini, sebagai pengelolah data yang diterima dari sensor *Heart Rate*, sehingga didapatkan nilai output.
3. **Output.**  
Firebase, sebagai monitoring *realtime database* yang bisa memonitoring dan menyimpan data yang ditampilkan melalui *web server*. *Buzzer*, sebagai notifikasi *alarm* kepada Satpam apabila nilai output yang didapatkan terdeteksi mengantuk.

Sensor *Heart Rate* memiliki 4 pin yaitu VCC, GND, SDA, dan SCL. Komunikasi yang digunakan adalah I2C. PIN I2C pada Arduino Pro Mini terletak pada pin A5 dan A4, sedangkan pin I2C pada sensor *Heart Rate* terletak pada SCL dan SDA.



Gambar 3. Skematik rangkaian alat

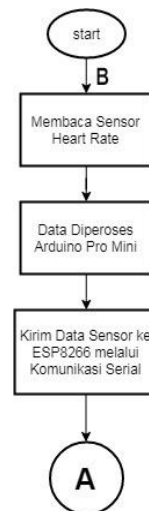
Untuk mengirimkan data yang diperoleh dari sensor *Heart Rate* ke Firebase diperlukan modul ESP8266. Komunikasi yang digunakan untuk mengirimkan data ke ESP8266 adalah komunikasi serial yaitu menggunakan RX (*Receiver*) TX (*Transmitter*). Pin RX pada Arduino dihubungkan ke pin TX pada ESP8266, begitu sebaliknya. Pin CH\_PD pada ESP8266 dihubungkan ke VCC. Setelah ESP8266 terkoneksi pada *internet/hostpot smartphone* data dari sensor *node* dikirimkan ke Firebase untuk monitoringnya.



Gambar 4. Disain perangkat keras

## B. Perancangan Software

1. *Flowchart* Arduino Pro Mini



Gambar 5. *Flowchart* Arduino Pro Mini

Gambar 5 adalah perancangan *software* dari Arduino Pro Mini. Ketika sistem dijalankan Arduino membaca data dari sensor *Heart Rate*, data yang diterima diproses dengan ketentuan  $HR > 80$  = beraktivitas, dan  $HR < 80$  = tertidur. Arduino Pro Mini mengirimkan data ke ESP8266 melalui komunikasi serial.

## 2. Flowchart ESP8266



Gambar 6. Flowchart ESP8266

Gambar 6 adalah perancangan software dari ESP8266. Ketika Arduino Pro Mini mengirimkan data ke ESP8266, data tersebut dibaca lalu dikirimkan ke *cloud* Firebase.

### 3. *Settingan* Pada Firebase

Agar bisa menampilkan data sensor *node* pada Firebase maka perlu melakukan pengaturan terlebih dulu. Berikut prosedur *settingan* pada Firebase:

1. Membuat *console* pada web resmi firebase yaitu: <https://firebase.google.com>.
2. Setelah itu ada pilihan dengan membuat proyek baru, mengisikan nama proyek dan negara.
3. Mengisikan data sesuai yang dibutuhkan sebagai contoh penulis mengisikan data awal dengan gelang-354.
4. Menyetting aturan menjadi *true*. *Read* dan *writenya* = *true*.
5. Mengopy rahasia basis data ke Arduino IDE dengan cara *setting* – akun layanan – rahasia database, untuk mengkoneksikan Arduino dengan firebase.
6. Mengopy link proyek ke Arduino IDE dengan cara Database – Data – Link. Contoh seperti [gelang-354.firebaseio.com](https://gelang-354.firebaseio.com).

## HASIL DAN PEMBAHASAN

### A. Pengujian *Heart Rate* dengan Tensimeter Detak Jantung

Pengujian *Heart Rate* sensor bertujuan untuk mengetahui berapa nilai detak jantung dari sampel yang ingin diuji. Sensor *Heart Rate* memiliki satuan Bpm (*Beat Per Minute*). Dalam

arti hasil yang akurat dalam pembacaan detak jantung yaitu ketukan per menit. Untuk membandingkan keakurasian hasil dari sensor *Heart Rate* diperlukan tensimeter detak jantung. Berikut prosedur pengujianya:

1. Menyambungkan Arduino Pro Mini dengan Arduino IDE menggunakan USB to TTL, lalu *Heart Rate* sensor disambungkan ke pin I2C Arduino Pro Mini (*SCL* dan *SDA*).
2. Menjalankan program Arduino.
3. Memverifikasi terlebih dahulu sebelum mengupload, agar tidak ada error.
4. Membandingkan data dari sensor *Heart Rate* dengan *tensimeter* detak jantung, *tensimeter* detak jantung yang digunakan bisa dilihat pada gambar 7 berikut ini.



Gambar 7. Tensimeter detak jantung

Tabel 1. Hasil Pengujian Sensor *Node* Dengan Tensimeter

| Sampel          | Percobaan | HR (Bpm)           |                   | Error |
|-----------------|-----------|--------------------|-------------------|-------|
|                 |           | <i>Finger clip</i> | <i>Tensimeter</i> |       |
| Revido          | 1         | 81                 | 81                | 0     |
|                 | 2         | 81                 | 81                | 0     |
|                 | 3         | 81                 | 81                | 0     |
|                 | 4         | 81                 | 81                | 0     |
|                 | 5         | 82                 | 81                | 1     |
|                 | 6         | 82                 | 81                | 1     |
| Rata-rata error |           |                    |                   | 0.33  |
| Afrizal         | 1         | 70                 | 70                | 0     |
|                 | 2         | 70                 | 70                | 0     |
|                 | 3         | 70                 | 70                | 0     |
|                 | 4         | 70                 | 70                | 0     |
|                 | 5         | 71                 | 70                | 1     |
|                 | 6         | 72                 | 70                | 2     |
| Rata-rata error |           |                    |                   | 0.33  |
| Agung           | 1         | 77                 | 77                | 0     |
|                 | 2         | 77                 | 77                | 0     |
|                 | 3         | 77                 | 77                | 0     |
|                 | 4         | 79                 | 77                | 2     |
|                 | 5         | 77                 | 77                | 0     |
|                 | 6         | 79                 | 77                | 2     |
| Rata-rata error |           |                    |                   | 0.33  |
| Fahmi           | 1         | 83                 | 83                | 0     |
|                 | 2         | 83                 | 83                | 0     |
|                 | 3         | 81                 | 83                | 2     |
|                 | 4         | 82                 | 83                | 1     |
|                 | 5         | 81                 | 83                | 2     |
|                 | 6         | 83                 | 83                | 0     |

| Sampel          | Percobaan       | HR (Bpm)    |            | Error |
|-----------------|-----------------|-------------|------------|-------|
|                 |                 | Finger clip | Tensimeter |       |
| Sukus           | Rata-rata error |             | 0.5        |       |
|                 | 1               | 80          | 80         | 0     |
|                 | 2               | 80          | 80         | 0     |
|                 | 3               | 81          | 80         | 1     |
|                 | 4               | 81          | 80         | 1     |
|                 | 5               | 80          | 80         | 0     |
|                 | 6               | 80          | 80         | 0     |
| Affandi         | Rata-rata error |             | 0.33       |       |
|                 | 1               | 85          | 85         | 0     |
|                 | 2               | 85          | 85         | 0     |
|                 | 3               | 85          | 85         | 0     |
|                 | 4               | 85          | 85         | 0     |
|                 | 5               | 84          | 85         | 1     |
|                 | 6               | 84          | 85         | 1     |
| Rata-rata error |                 | 0.33        |            |       |

Dari tabel 1, Hasil pengujian akurasi sensor *node* dengan *tensimeter* detak jantung pada sampel diatas dapat disimpulkan bahwa rata-rata *error* yaitu 0.358.

### B. Pengujian Membedakan Orang Yang Beraktivitas Dengan Yang Beristirahat

Berdasarkan teori yang diperoleh untuk membedakan orang yang beristirahat/tertidor dengan yang beraktivitas yaitu dilihat dari detak jantungnya. Normalnya orang dewasa beraktivitas berkisar 80-120 Bpm dan orang yang beristirahat < 80 Bpm. (Sandi, 2013). Hasil pengujian bisa dilihat pada tabel 2.

Tabel 2. Hasil Pengujian Membedakan Orang Yang Beraktivitas Dengan Beristirahat

| Sampel  | Percobaan | Nilai HR           | Nilai HR           |
|---------|-----------|--------------------|--------------------|
|         |           | Beraktivitas (Bpm) | Beristirahat (Bpm) |
| Revido  | 1         | 109                | 78                 |
|         | 2         | 90                 | 75                 |
|         | 3         | 93                 | 68                 |
|         | 4         | 89                 | 70                 |
|         | 5         | 89                 | 72                 |
|         | 6         | 86                 | 70                 |
| Afrizal | 1         | 93                 | 79                 |
|         | 2         | 92                 | 78                 |
|         | 3         | 94                 | 73                 |
|         | 4         | 95                 | 73                 |
|         | 5         | 95                 | 72                 |
|         | 6         | 90                 | 70                 |
| Agung   | 1         | 97                 | 69                 |
|         | 2         | 87                 | 70                 |
|         | 3         | 83                 | 68                 |
|         | 4         | 82                 | 72                 |
|         | 5         | 81                 | 70                 |
|         | 6         | 81                 | 70                 |
| Fahmi   | 1         | 91                 | 79                 |
|         | 2         | 90                 | 79                 |
|         | 3         | 83                 | 77                 |
|         | 4         | 80                 | 71                 |
|         | 5         | 83                 | 71                 |
|         | 6         | 80                 | 69                 |

| Sampel  | Percobaan | Nilai HR           | Nilai HR           |
|---------|-----------|--------------------|--------------------|
|         |           | Beraktivitas (Bpm) | Beristirahat (Bpm) |
| Sukus   | 1         | 109                | 77                 |
|         | 2         | 96                 | 75                 |
|         | 3         | 94                 | 80                 |
|         | 4         | 91                 | 70                 |
|         | 5         | 91                 | 68                 |
|         | 6         | 90                 | 75                 |
| Affandi | 1         | 102                | 78                 |
|         | 2         | 97                 | 75                 |
|         | 3         | 94                 | 75                 |
|         | 4         | 90                 | 72                 |
|         | 5         | 91                 | 68                 |
|         | 6         | 78                 | 70                 |

Dari tabel 2, dapat disimpulkan pengujian sampel diatas detak jantung orang yang beraktivitas dengan orang yang beristirahat berbeda. Rata-rata orang yang beraktivitas detak jantungnya > 80, sedangkan orang yang beristirahat/tidur detak jantungnya lebih rendah yaitu < 80. Pengujian ini mengambil sampel dengan kriteria umur 20 – 30 tahun, dan tidak memiliki riwayat penyakit dalam yang dapat mempengaruhi kinerja detak jantung..

### C. Pengujian Integrasi Arduino Pro Mini Dengan Sensor Heart Rate

Pengujian integrasi dilakukan agar sensor *Heart Rate* pada Arduino Pro Mini bisa berjalan dengan baik. Berikut prosedur pengujiannya:

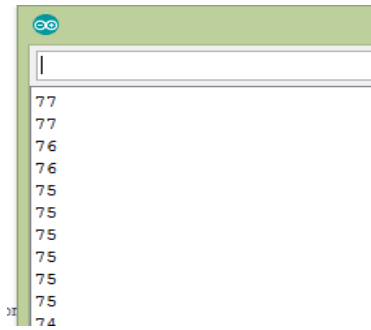
1. Menyambungkan Arduino Pro Mini dengan USB to TTL. Bisa dilihat pada gambar 8 berikut ini.



Gambar 8. Skematik Prototype Sensor Node

2. Membuka Arduino IDE dan memasukkan program untuk mencoba komunikasi serial antara Arduino Pro Mini dengan sensor *Heart Rate*.
3. Memverifikasi terlebih dahulu sebelum meng-*upload*, agar tidak ada error
4. Membuka *serial monitor* pada pojok kanan atas untuk melihat hasilnya.



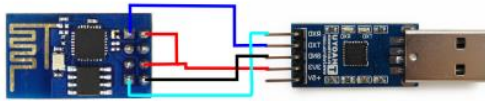


Gambar 9. Serial Monitor

#### D. Pengujian ESP8266

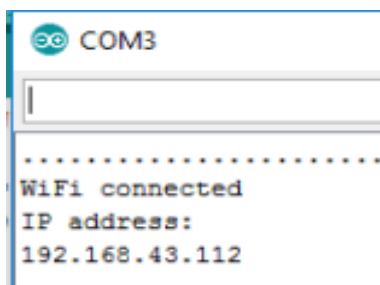
Pengujian modul ESP8266 dilakukan untuk memastikan apakah data dari Arduino Pro Mini bisa dikirimkan ke cloud Firebase dengan menggunakan komunikasi serial. Berikut prosedur pengujiannya:

1. Menghubungkan ESP8266 dengan USB to TTL, seperti pada gambar berikut ini.



Gambar 10. Skematik ESP8266 dengan USB to TTL

2. Membuka Arduino IDE dan jalankan program ESP8266.
3. Mengganti board dengan modul ESP8266 GENERIC..
4. Memverifikasi terlebih dahulu sebelum mengupload program agar tidak ada error, setelah itu upload program.
5. Membuat hotspot wifi menggunakan smartphone dengan SSID iPhone Freean dan PASS cakpreee.

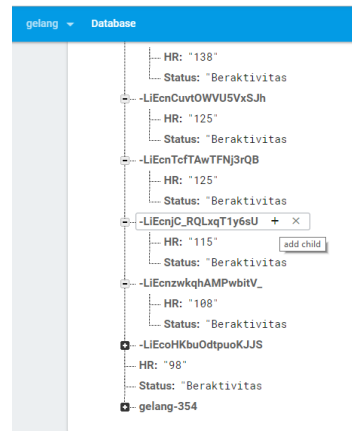


Gambar 11. Tampilan Connected ESP8266

#### E. Pengujian Firebase

Untuk melihat data pada tampilan Firebase maka dilakukan pengujian uji coba pengiriman data ke Firebase. Berikut prosedur pengujiannya:

1. Menyambungkan Arduino Pro Mini dengan ESP8266 dan sensor Heart Rate.
2. Mengaktifkan hotspot dan lihat apakah ESP8266 sudah terkoneksi dengan hotspot.
3. Memasang sensor Heart Rate di jari / pergelangan tangan
4. Membuka console Firebase dan lihat data yang masuk pada Firebase.



Gambar 12. Tampilan data pada Firebase

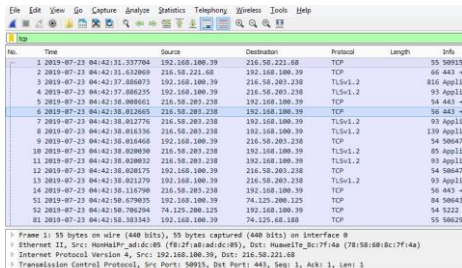
#### F. Pengujian Delay

1. Menjalankan aplikasi wireshark, dan memilih jaringan wifi pada capture, seperti gambar 13 berikut ini.



Gambar 13. Tampilan awal Wireshark

2. Membuka website firebase dan menyalakan sensor node untuk mendapatkan data detak jantung.
3. Menunggu 1-2 menit agar data yang diterima oleh firebase sesuai dengan kondisi detak jantung sebenarnya.
4. Melihat delay pada wireshark, dengan cara klik stop pada wireshark dan menuliskan tcp/http pada filter, seperti gambar 14 berikut ini.



Gambar 14. Tampilan Capture pada jaringan TCP

- Menyimpan *delay* yang telah *capture*, dengan cara *file – Export Packet Dissections – As CSV*.
- Membuka *file* yang telah disimpan dengan *excel* dan menghitung *delay*-nya. Seperti gambar 15 berikut ini.

Gambar 15. Delay yang tersimpan pada Excel

Hasil *delay* yang didapat dengan perhitungan dengan menggunakan *wireshark* bisa dilihat pada tabel 3 berikut ini.

Tabel 3. Pengujian Delay

| Sampel                  | Percobaan | Spek Jaringan  | Delay (detik) |
|-------------------------|-----------|----------------|---------------|
| Revido                  | 1         | Wifi (10 mbps) | 36.2          |
|                         | 2         | Wifi (10 mbps) | 57.4          |
|                         | 3         | Wifi (10 mbps) | 68.1          |
|                         | 4         | Wifi (10 mbps) | 33.8          |
|                         | 5         | Wifi (10 mbps) | 55.2          |
|                         | 6         | Wifi (10 mbps) | 50.0          |
| Rata-rata delay (detik) |           |                | 50.1166       |
| Afrizal                 | 1         | Wifi (10 mbps) | 109.1         |
|                         | 2         | Wifi (10 mbps) | 79.6          |
|                         | 3         | Wifi (10 mbps) | 57.0          |
|                         | 4         | Wifi (10 mbps) | 63.5          |
|                         | 5         | Wifi (10 mbps) | 119.9         |
|                         | 6         | Wifi (10 mbps) | 165.0         |
| Rata-rata delay (detik) |           |                | 99.0166       |
| Agung                   | 1         | Wifi (10 mbps) | 64.4          |
|                         | 2         | Wifi (10 mbps) | 87.1          |
|                         | 3         | Wifi (10 mbps) | 79.2          |
|                         | 4         | Wifi (10 mbps) | 129.4         |
|                         | 5         | Wifi (10 mbps) | 80.7          |
|                         | 6         | Wifi (10 mbps) | 55.6          |
| Rata-rata delay (detik) |           |                | 82.7333       |
| Fahmi                   | 1         | Wifi (10 mbps) | 62.1          |
|                         | 2         | Wifi (10 mbps) | 51.6          |

| Sampel                  | Percobaan               | Spek Jaringan  | Delay (detik)  |         |
|-------------------------|-------------------------|----------------|----------------|---------|
| Sukus                   | 3                       | Wifi (10 mbps) | 42.5           |         |
|                         | 4                       | Wifi (10 mbps) | 86.9           |         |
|                         | 5                       | Wifi (10 mbps) | 65.7           |         |
|                         | 6                       | Wifi (10 mbps) | 34.6           |         |
|                         | Rata-rata delay (detik) |                |                | 57.2333 |
|                         | Affandi                 | 1              | Wifi (10 mbps) | 51.1    |
| 2                       |                         | Wifi (10 mbps) | 66.1           |         |
| 3                       |                         | Wifi (10 mbps) | 57.5           |         |
| 4                       |                         | Wifi (10 mbps) | 67.8           |         |
| 5                       |                         | Wifi (10 mbps) | 52.7           |         |
| 6                       |                         | Wifi (10 mbps) | 55.8           |         |
| Rata-rata delay (detik) |                         |                | 58.5           |         |
| Sukus                   | 1                       | Wifi (10 mbps) | 65.6           |         |
|                         | 2                       | Wifi (10 mbps) | 51.8           |         |
|                         | 3                       | Wifi (10 mbps) | 48.0           |         |
|                         | 4                       | Wifi (10 mbps) | 42.7           |         |
|                         | 5                       | Wifi (10 mbps) | 42.2           |         |
|                         | 6                       | Wifi (10 mbps) | 38.5           |         |
| Rata-rata delay (detik) |                         |                | 48.1333        |         |

Dari tabel 3, dapat disimpulkan bahwa pengiriman data dari sensor *node* ke firebase menggunakan jaringan *wifi* memiliki *delay* dengan rata-rata *delay* 65.9555 detik.

## KESIMPULAN

Hasil dari pengujian rancang bangun alat pendeteksi tidur pada *security Guard* berbasis IoT (*Internet of Things*) pada tugas akhir ini terdapat kesimpulan dan saran dari penulis diantaranya:

- Disimpulkan bahwa orang yang beristirahat/tidur memiliki detak jantung lebih rendah dari pada orang yang beraktivitas. Orang yang beristirahat/tidur memiliki rata-rata detak jantung < 80 Bpm, sedangkan orang yang beraktivitas memiliki rata-rata detak jantung > 80 Bpm. Hasil pengujian akurasi sensor *node* dengan *tensimeter* detak jantung memiliki rata-rata *error* yaitu 0.358.
- Data dari sensor *Heart Rate* yang dikirimkan ke Firebase dapat dimonitoring dengan baik. Firebase menampilkan nilai detak jantung dan keterangan apakah Satpam tertidur atau tidak. Pengiriman data menggunakan ESP8266 ke *cloud* Firebase memiliki *delay* rata-rata 65.9555 detik.
- Data sensor *Heart Rate* dikirimkan ke Arduino Pro Mini melalui *pin* I2C, dengan pembacaan 1 detik sekali.

## Saran

- Untuk membuat sistem monitoring Satpam dibutuhkan catu daya yang cukup. Hasil perhitungan catu daya pada rancang bangun ini  $\pm$  3.5 jam, sedangkan umunya Satpam bertugas dalam shift 8 jam.

2. Demi kenyamanan Satpam dalam bertugas disarankan desain rancang bangun alat lebih diperkecil, agar lebih efisien dan ergonomis

## DAFTAR PUSTAKA

- Bashofi, M. E., & Abidin, R. Z. (2018). Implementasi Firebase Pada Sistem Kendali Lampu Jarak Jauh Berbasis Android. *Univesitas Yudharta Pasuruan*.
- Darwinsa, E. (2011). *Ratemonitor Sepuluh Denyut Per Satuan Waktu Dengan Transmisi Radio Frekuensi ( RF )*.
- Dita, D. (2009). Pengaruh Ekstrak Biji Pala (*Myristica fragrans* Houtt) DOSIS 7,5 mg/25grBB Terhadap Waktu Induksi Tidur Dan Lama Waktu Tidur Mencit Balb / C Yang Diinduksi Thiopental. *Fakultas Kedokteran Universitas Diponegoro*, 1–31.
- Hidayah, A. (2012). *Tingkat Pengetahuan Pasien Diabetes Mellitus tentang Risiko Terjadinya Ulkus Kaki Diabetes di Poli Klinik Penyakit Dalam Rumah Sakit Umum Pusat Haji Adam Malik Medan*.
- Ids, M. M. A. (n.d.). 1 2 3, 4. 214–222.
- Putra, N. D. (2018). Wireless Smart Tag Device Sebagai Sistem Keamanan Rumah Sistem Keamanan Rumah. *Fakultas Teknologi Industri Universitas Islam Indonesia*.
- Sandi, N. (2013). Hubungan Antara Tinggi Badan, Berat Badan, Indeks Massa Tubuh, Dan Umur Terhadap Frekuensi Denyut Nadi Istirahat Siswa Smkn-5 Denpasar. *1(1)*, 38–44.
- SeedTechnology. (2018). “GROVE-FINGER-CLIP- HATE RATE SENSOR.” Retrieved from [http://wiki.seeedstudio.com/Grove-Finger-clip\\_Heart\\_Rate\\_Sensor/](http://wiki.seeedstudio.com/Grove-Finger-clip_Heart_Rate_Sensor/)
- SOFYAN, A. (2006). Perancangan Alat Monitoring Cek Point Satpam.
- Suoth, M., Bidjuni, H., Malara, R. T., Studi, P., Keperawatan, I., Kedokteran, F., ... Manado, R. (2014). Pendahuluan Pelayanan Kesehatan Yang Diberikan Puskesmas Merupakan Pelayanan Yang Menyeluruh Yang Meliputi Pelayanan Kuratif (Pengobatan), Rehabilitative Kesehatan. Pelayanan Tersebut Ditujukan Kepada Semua Penduduk Dengan Tidak Membedakan Jenis Kelam. 2.
- WWW.FARNELL.COM. (2016). BUZZER. Retrieved from <http://www.farnell.com/datasheets/2171929.pdf>