

PENGEMBANGAN APLIKASI PEMANTAUAN KESEHATAN MENGUNAKAN WSN BERBASIS ARDUINO

CRISTINE ARTANTY—2017730050

1 Data Skripsi

Pembimbing utama/tunggal: **Elisati Hulu, S.T., M.T.**

Pembimbing pendamping: -

Kode Topik : **ELH4905**

Topik ini sudah dikerjakan selama : **1 semester**

Pengambilan pertama kali topik ini pada : **Semester 7 - Ganjil 20/21**

Pengambilan pertama kali topik ini di kuliah : **Skripsi 1**

Tipe Laporan : **B** - Dokumen untuk reviewer pada presentasi dan **review Skripsi 1**

2 Latar Belakang

Seiring dengan perkembangan teknologi komputer yang berkembang semakin maju dan kecepatan internet yang semakin cepat, teknologi komputer semakin banyak digunakan untuk membantu manusia dalam menjalani kehidupan sehari-hari. Penggunaannya tidak hanya sebatas bidang komunikasi, tetapi sudah pada berbagai bidang seperti bidang industri, bidang perbankan, bidang kedokteran, dan masih banyak lagi. Pada bidang kedokteran teknologi komputer banyak dimanfaatkan mulai dari hal yang sederhana hingga kompleks, Seperti melakukan administrasi data dokter dan pasien hingga memonitor kesehatan pasien lewat rekam medis pasien.

Kesehatan menjadi salah satu faktor untuk meningkatkan kualitas dan kuantitas hidup seseorang, Untuk menjamin kesehatan, seseorang harus melakukan pemantauan kesehatan secara berkala. Untuk melakukan pemantauan kesehatan seseorang harus pergi ke rumah sakit(puskesmas) yang dapat melakukan hal tersebut, tetapi tidak semua rumah sakit(puskesmas) menyediakan alat pemantauan kesehatan tersebut. Selain itu tidak semua masyarakat mempunyai tempat tinggal yang dekat dengan rumah sakit yang menyediakan alat pemantauan. Oleh karena itu masyarakat yang tinggal jauh dari rumah sakit tersebut akan kesulitan untuk dapat melakukan pengecekan kesehatan secara berkala. Maka dari itu untuk memudahkan pemantauan dapat dilakukan dengan cara memanfaatkan *Wireless Sensor Network* (WSN) yang diintegrasikan dengan sensor-sensor medis(*biosensor*). *Wireless Sensor Network* merupakan kumpulan *node-node* yang terdistribusi untuk melakukan pengukuran-pengukuran sesuai dengan penggunaannya. *Node-node* dapat berupa sensor, Adapun sensor merupakan perangkat elektronik yang dapat melakukan pengukuran/mendeteksi perubahan besaran misalnya suhu, tekanan, dan kelembapan. Dengan memanfaatkan teknologi tersebut seseorang tidak harus terus pergi ke rumah sakit secara berkala untuk melakukan pemantauan. Teknologi tersebut dapat digunakan untuk melakukan pemantauan kesehatan secara *real-time* sehingga seseorang dapat mengetahui kondisi kesehatannya.

Pada skripsi ini, akan dibangun sebuah perangkat lunak pemantau kesehatan dengan *Wireless Sensor Network*(WSN) yang berbasis arduino. WSN akan diintegrasikan dengan beberapa sensor medis sehingga seseorang nantinya dapat melakukan pemantauan dengan beberapa parameter. Dengan perangkat lunak pemantauan kesehatan tersebut diharapkan dapat membawa banyak manfaat. Beberapa manfaat yang dapat diperoleh seperti mempersingkat waktu diagnosis sehingga seseorang dapat segera ditangani, dapat melakukan pemantauan lebih banyak orang, mengurangi penggunaan kertas yang biasanya digunakan untuk menunjukkan hasil pemantauan kesehatan konvensional dan seseorang dapat melakukan tindakan pencegahan sejak dini terkait kesehatannya.

3 Rumusan Masalah

Adapun rumusan masalah dari deskripsi yang telah dipaparkan adalah:

- Bagaimana cara kerja WSN dan cara mengintegrasikan dengan sensor-sensor kesehatan?
- Bagaimana membangun perangkat lunak menggunakan WSN yang berbasis arduino untuk pemantauan kesehatan?
- Bagaimana menguji kualitas hasil pemantauan yang dilakukan?

4 Tujuan

Adapun tujuan penelitian dari rumusan masalah yang telah dipaparkan adalah:

- Mempelajari cara kerja WSN dan mengintegrasikan WSN dengan sensor-sensor kesehatan.
- Membangun perangkat lunak menggunakan WSN yang berbasis *arduino* untuk pemantau kesehatan.
- Menguji kualitas hasil pemantauan yang dilakukan.

5 Detail Perkembangan Pengerjaan Skripsi

Detail bagian pekerjaan skripsi sesuai dengan rencan kerja/laporan perkembangan terakhir :

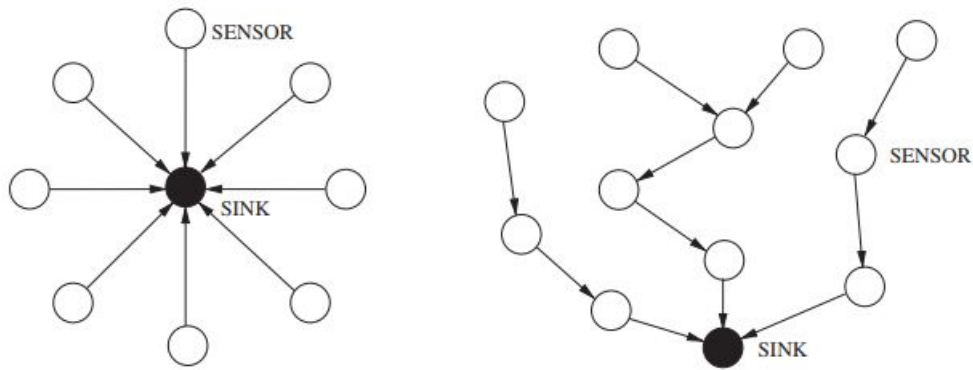
1. Melakukan studi literatur mengenai WSN *arduino*, dan sensor.

Status : Ada sejak rencana kerja skripsi.

Hasil : Hasil dari studi literatur ini terdiri dari 3 bagian yaitu WSN, Arduino dan sensor. Berikut hasil dari ketiga bagian tersebut yang akan dijelaskan satu per satu secara berurutan. *Wireless sensor network* (WSN) adalah jaringan nirkabel yang terdiri dari kumpulan node sensor yang saling berkomunikasi melakukan pengukuran pada lingkungan atau area diletakannya. Adapun komponen dari sensor node pada WSN adalah

- Embedded Processor yang berfungsi untuk melakukan penjadwalan dan pemrosesan data. Contoh dari embedded processor pada sensor node adalah *microcontroller*.
- Transceiver yang berfungsi untuk mengatur komunikasi antar sensor node. Contoh dari transceiver pada sensor node adaah *radio frequency*(RF), laser dan infrared.
- Memory yang berfungsi sebagai tempat penyimpanan pada sensor node. Contoh dari memory pada sensor node adalah *in-chip flash memory*, RAM dan *external flash memory*.
- Power source yang berfungsi sebagai sumber daya dari sensor node. Contoh dari power source yang digunakan oleh sensor node adalah baterai.
- Sensor yang merupakan komponen yang akan melakukan *sensing* atau pengukuran.

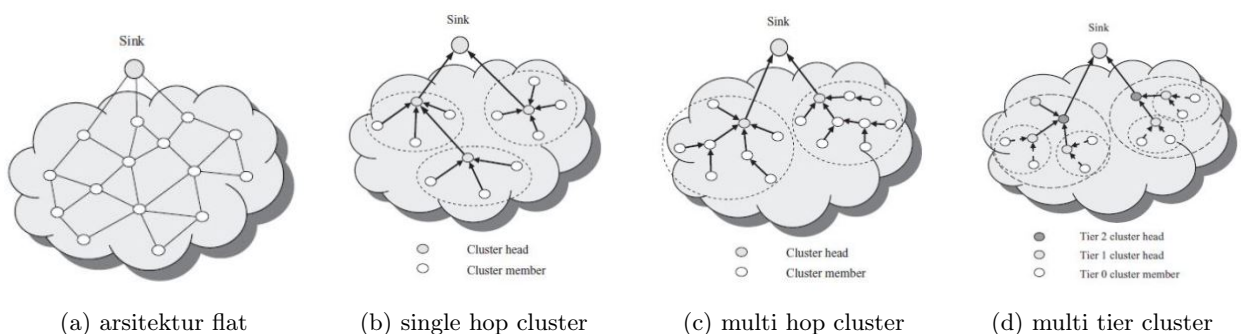
Data hasil *sensing* atau pengukuran oleh setiap sensor akan dikirimkan ke *base station*. Network layer pada WSN bertugas untuk mencari jalur dari sensor *data source* menuju ke *base station device base-station*. Proses pencarian jalur tersebut dinamakan *routing*.

Gambar 1: *single-hop* dan *multi-hop*

Dapat dilihat pada gambar 1 bahwa terdapat 2 jenis protokol komunikasi yang digunakan oleh WSN yaitu :

- Single-hop, merupakan jenis *routing* dimana setiap node sensor dapat langsung samapi pada *base-station* tanpa harus melewati node sensor lain.
- Multi-hop, merupakan jenis *routing* dimana untuk mencapai *base-station* setiap node sensor harus melewati node sensor lain.

WSN dapat dibangun dengan 2 jenis arsitektur yaitu *flat architecture* dan *hierarchy architecture*. Pemilihan kedua arsitektur ini akan berpengaruh pada tugas-tugas dari setiap node sensor. Pada arsitektur flat (gambar 2a), setiap node sensor akan memiliki tugas yang sama dalam melakukan *sensing* dan Setiap node sensor akan mengirimkan data hasil *sensing* ke *base-station*. Setiap node sensor akan menjadi peers untuk masing-masing node sensor sehingga komunikasi akan terjadi secara multi-hop. Sementara pada arsitektur hirarki, node sensor dikelompokkan menjadi *cluster-cluster* atau kelompok-kelompok. Pada tiap cluster terdiri atas *cluster head* dan *cluster member*, dimana *cluster head* bertugas untuk menerima data dari *cluster member* lalu meneruskan data ke *base-station*. Pemilihan *cluster head* dilakukan berdasarkan jarak *cluster head* ke *cluster member* yaitu *single-hop* (gambar 2b) atau *multi-hop* (gambar 2c) dan dapat berdasarkan jumlah tier (2d).



Gambar 2: Arsitektur WSN

Jalur *routing* dan cara komunikasi antar node sensor dengan node sensor atau dengan *base-station* bergantung pada topologi yang akan digunakan oleh WSN tersebut. Topologi pada jaringan WSN terdapat beberapa jenis yaitu :

- Topologi Bus, topologi ini menghubungkan semua node sensor pada sebuah jalur yang sama untuk menuju *base-station*. Topologi ini cocok digunakan pada jaringan yang kecil, karena akan

boros *bandwidth* jika digunakan pada jaringan besar. Hal ini dikarenakan sebuah node sensor akan melakukan broadcast ke semua node sensor yang terhubung saat mengirimkan pesan. Keuntungan topologi ini adalah implementasinya yang mudah, sedangkan kelemahannya adalah karena menggunakan jalur tunggal maka jika jalur terputus semua node sensor tidak akan dapat berkomunikasi.

- Topologi Ring, semua node sensor akan memiliki 2 buah node sensor tetangga dan akan dihubungkan secara melingkar menuju node tujuan. Topologi ini cocok digunakan pada jaringan kecil, karena pengiriman pesan akan dilakukan secara searah jarum jam ataupun sebaliknya maka akan memakan waktu yang cukup lama untuk dapat samapai pada node tujuan atau *base-station* jika diterapkan pada jaringan besar. Keuntungan dari topologi ini adalah mudah diimplementasi sedangkan kelemahan dari topologi ini adalah jika terdapat jalur putus maka komunikasi tidak dapat dilakukan.
- Topologi Star, setiap node sensor akan langsung terhubung dengan *base-station*, sehingga jika sebuah node sensor akan berkomunikasi dengan node sensor lain maka harus melewati *base-station*. Sementara jika node sensor akan mengirimkan hasil *sensing* ke *base-station* dapat dilakukan secara langsung tanpa melewati node sensor lain. Keuntungan dari topologi ini adalah setiap node sensor memiliki jalur sendiri menuju ke *base-station* sehingga jika jalur sebuah node sensor putus, komunikasi tetap dapat dilakukan.
- Topologi Tree, node sensor pada topologi ini dihubungkan secara hirarkial dengan sensor lain. Pada topologi ini terdapat *root-node* pada *first-level* dan *children-node* pada level-level di bawahnya. *Root-node* merupakan *base-station* sedangkan *children-node* merupakan node-sensor yang terhubung. Keuntungan dari topologi ini adalah untuk memudahkan proses *maintenance* dan komunikasi tetap dapat dilakukan walaupun terdapat sebuah jalur yang putus.
- Topologi Mesh, semua node sensor terhubung satu sama lain. Topologi ini mempunyai 2 jenis yaitu *full mesh* dan *partial mesh*. *Full mesh* jika semua node sensor terhubung satu sama lain, sedangkan *partial mesh* jika node sensor hanya terhubung dengan sebagian node sensor lain. Kelemahan dari topologi ini adalah proses *maintenance*-nya yang sulit dilakukan. Topologi ini cocok digunakan untuk jaringan besar.

Selain jalur komunikasi, WSN juga membutuhkan sistem operasi untuk mengatur kerja sensor node. Selain itu sistem operasi juga bertugas untuk mengatur komunikasi dengan *hardware*, melakukan *task scheduling* termasuk mengatur prioritas pengerjaan task serta melakukan manajemen terhadap memori, file dan daya. Berikut beberapa contoh sistem operasi yang digunakan WSN :

- TinyOS
- Contiki
- MANTIS
- Nano-RK
- PreonVM
- LiteOS

Sistem operasi pada WSN digunakan untuk mengatur kerja sensor node, dalam mengatur kerja sensor node tersebut terdapat 3 jenis *management* pada WSN yaitu :

- *Power Management Plane* digunakan untuk mengatur power level dari sensor node untuk melakukan *sensing*, *processing*, *transmission* dan *reception*. Sebagai contoh transceiver saat tidak ada data yang akan dikirim atau diterima akan dimatikan oleh sensor node.

- *Connection Management Plane* digunakan untuk melakukan pengaturan terhadap koneksi sensor node saat adanya perubahan topologi yang terjadi pada WSN. Perubahan tersebut seperti adanya penambahan node sensor baru dan perubahan peletakan node sensor.
- *Task Management Plane* digunakan untuk melakukan pembagian tugas atau mendistribusikan tugas pada sensor node untuk meningkatkan efisiensi energi pada WSN.

Dari 3 *management* tersebut terdapat 5 layer protokol stack pada WSN yaitu :

- **Application Layer** : Pada *application layer* terdapat bermacam-macam protokol untuk *network sensor application* seperti *query dissemination*, *node localization*, *time synchronization* dan *network security*.
- **Transport Layer** : digunakan untuk mengirimkan data, baik dari sensor ke sensor lain atau dari sensor menuju ke sink. Pengiriman dilakukan secara reliable. Terdapat 2 cara pengiriman yaitu *upstream* (sensor node mengirimkan data ke sink) dan *downstream* (sink mengirimkan data ke sensor node). Kebutuhan pengiriman secara reliable berbeda untuk kedua cara pengiriman. *Downstream* membutuhkan pengiriman reliable karena *sink* hanya mengirimkan data ke sensor node satu kali. Sedangkan *upstream* tidak terlalu membutuhkan pengiriman secara reliable karena sensor node mengirimkan data hasil pengukuran *sense* ke *sink* secara terus menerus yang dapat dilakukan dengan cara pengulangan.
- **Network Layer** : digunakan untuk membuat rute komunikasi dari sensor node menuju ke *sink*. Pada *network layer* juga dilakukan penentuan protokol komunikasi apakah menggunakan *single-hop* / *multi-hop* sehingga dapat membuat komunikasi lebih optimal.
- **Data Link Layer** : salah satu fungsi terpenting dari *data link layer* adalah *medium access control* (MAC). MAC digunakan untuk menjamin agar semua sensor node dapat dengan adil serta efisien berkomunikasi sehingga performa dan komunikasi energi optimal. Hal yang dilakukan pada *data link layer* adalah *multiplexing*, *data frame creation and detection*, *medium access*, dan *error control* untuk *reliable point-to-point* dan *point-to-multipoint transmissions*.
- **Physical Layer** : digunakan untuk mengonversi bit *stream* dari *data link layer* menjadi sinyal yang dapat ditransmisi pada media komunikasi.

Arduino merupakan sebuah *open-source platform* yang dirancang sangat sederhana, arduino dapat membaca dari sensor dan mengontrol komponen misalkan lampu led. Arduino mempunyai beberapa jenis model / *board*, sehingga untuk memilih *board* arduino perlu melakukan pertimbangan sesuai dengan kebutuhan penggunaannya. Berikut beberapa jenis-jenis arduino *board* : Arduino Uno R3, Arduino Uno SMD, Arduino MO, Arduino Mega, Arduino Leonardo dan Arduino Nano. Setiap jenis *board* arduino mempunyai spesifikasi yang berbeda-beda. Berdasarkan pertimbangan akan spesifikasi dari *board* arduino, pada penelitian ini akan digunakan *Arduino Mega*. Jika pengguna membutuhkan fitur lebih tetapi *board* arduino tidak memiliki fungsi tersebut, pengguna dapat menambahkan *shield*. *Shield* merupakan sebuah tambahan yang memiliki bentuk mirip dengan *board* arduino tetapi hanya digunakan untuk melengkapi fungsionalitas dari arduino tersebut. Pada sebuah *board* arduino terdapat beberapa komponen, berikut komponen-komponen penting pada sebuah *board* arduino :

- **Digital pins**: *Arduino Mega* mempunyai jumlah pins sebanyak 54 pins. Digital pins digunakan untuk baik input maupun output. State dari digital pins ada 2 yaitu *high* dan *low*. Sebagai contoh digital pins digunakan untuk menghidupkan dan mematikan lampu led. maka untuk menghidupkan lampu led state yang digunakan adalah *high* dan untuk mematikan menggunakan state *low*.
- **Analog pins**: *Arduino Mega* mempunyai 16 analog pins. Pembacaan dari analog pins mempunyai range dari 0 hingga 1023. Contoh sensor yang melakukan pembacaan secara analog adalah

sensor yang mengukur tingkat kelembapan tanah. Dengan menggunakan range tersebut akan memudahkan untuk mengetahui tingkat kelembapan tanah tersebut.

- **USB connector:** *USB connector* digunakan untuk menghubungkan *Arduino* dengan komputer.
- **Battery power:** komponen IoT yang diletakkan pada lokasi tertentu akan membutuhkan daya/(power). Untuk itu biasa digunakan baterai untuk memberikan daya pada komponen dan menghubungkannya dengan *board*.

Software Yang digunakan sebagai IDE(Integrated Development Environment) pada arduino dapat diunduh secara gratis pada website www.arduino.cc. Flow programming pada arduino adalah sebagai berikut :

- menyambungkan board arduino ke komputer atau laptop melalui port USB.
- menuliskan sketch atau program untuk menyalakan board arduino.
- upload sketch pada board arduino melalui koneksi USB dan tunggu beberapa detik untuk board arduino melakukan load sketch tersebut.
- board arduino akan melakukan hal yang dituliskan pada sketch.

Sensor adalah perangkat atau *device* yang mengubah pengukuran fisik menjadi signal yang dapat dibaca oleh observer atau alat pengukuran. Berdasarkan fungsi dan penggunaannya sensor dapat dibagi menjadi beberapa kelompok yaitu *mechanical sensor*, *optical sensor*, *semiconductor sensor*, *electrochemical sensor* dan *biosensors*. Jenis sensor yang akan digunakan dalam melakukan penelitian ini adalah *biosensor*. Terdapat berbagai parameter kesehatan yang dapat dipantau oleh biosensor yaitu detak jantung, level saturasi oksigen, peredaran darah pada tubuh, tingkat pernafasan, kerja otot pada tubuh, banyaknya oksigen yang di hirup, dan pola pergerakan manusia. Berikut beberapa contoh sensor yang dapat digunakan untuk pemantauan kesehatan:

- *pulse oxygen saturation sensor* digunakan untuk melakukan pengukuran besaran persentase oksigen yang di-ikat oleh hemoglobin(SpO2) dan heart rate (HR);
- *blood pressure sensors* digunakan untuk mengukur tekanan darah;
- *electrocardiogram(ECG)* digunakan untuk mengukur detak jantung;
- *electromyogram(EMG)* digunakan untuk mengukur aktifitas kerja otot;
- *temperature sensors* digunakan untuk mengukur suhu tubuh;
- *respiration sensors* digunakan untuk mengukur pernafasan;
- *blood flow sensors* digunakan untuk mengukur peredaran darah;
- *blood oxygen level sensor (oximeter)* digunakan untuk mengukur level oksigen pada darah.

2. Melakukan studi literatur mengenai kesehatan.

Status : Baru ditambahkan di semester ini.

Hasil : Penulis melakukan studi mengenai kesehatan dengan membaca beberapa buku yang berhubungan dengan bidang kedokteran dan kesehatan. Berikut hasil dari studi yang telah dilakukan untuk melengkapi pemahaman penulis untuk melakukan penelitian ini. Pengertian kesehatan dalam UU Kesehatan No 23,1992 adalah keadaan sejahtera dari badan, jiwa dan sosial yang memungkinkan setiap orang hidup produktif secara sosial dan ekonomi. Oleh karena itu kesehatan merupakan hal yang sangat penting untuk diperhatikan karena dapat menjadi dasar untuk meningkatkan kualitas dan kuantitas hidup dalam masyarakat. Untuk dapat meningkatkan kualitas hidup manusia dalam bidang kesehatan dapat dilakukan dengan beberapa upaya yaitu peningkatan (*promotif*), pencegahan(*preventif*),

penyembuhan(*kuratif*, dan pemulihan(*rehabilitatif*). Untuk melakukan pencegahan(*preventif*) masyarakat ada baiknya melakukan pemeriksaan kesehatan secara berkala. Pemeriksaan kesehatan atau biasa disebut *medical test* adalah pemeriksaan kesehatan secara menyeluruh yang penting dilakukan untuk mengetahui kondisi kesehatan seseorang. Pemeriksaan ini dilakukan untuk mengidentifikasi kondisi kesehatan sehingga dapat dilakukan pencegahan atau pengobatan secara dini. Berdasarkan tujuannya pemeriksaan kesehatan dapat dibagi menjadi *diagnostic*, *screening*, dan *monitoring*.

Diagnostic : *diagnostic test* adalah sebuah prosedur yang dilakukan untuk menentukan kebenaran dari diagnosis penyakit yang telah dilakukan oleh seorang dokter. Diagnosis yang dilakukan oleh seorang dokter biasanya dilakukan berdasarkan *symptoms* atau keluhan pasien dan riwayat penyakit pasien. Beberapa contoh *diagnostic test* adalah Mengukur *blood glucose* pada pasien yang diduga mempunyai penyakit diabetes dan Memonitor pembacaan *electrocardiogram* pada pasien yang menderita sakit pada dada untuk mendiagnosa adanya penyakit pada jantung pasien.

Screening : *Screening* merupakan jenis *medical test* yang digunakan untuk mendeteksi atau memprediksi adanya penyakit yang beresiko pada kelompok tertentu seperti keluarga, tenaga kerja, atau suatu populasi. Tujuan dilakukannya hal ini adalah untuk memantau penyebaran suatu penyakit, mengelola epidemiologi, membantu pencegahan suatu penyakit, dan untuk kebutuhan statistik.

Monitoring : *Monitoring* adalah *medical test* yang bertujuan untuk memantau progress atau respon dari sebuah *medical treatment*.

Dari hasil dari test yang telah dilakukan, dapat digunakan untuk membantu dokter atau tenaga medis dalam melakukan *patient management*. Adapun contoh tindakan *patient management* yang dilakukan adalah mengevaluasi tingkat keparahan penyakit yang diderita oleh pasien, memperkirakan prognosis¹, memantau perkembangan penyakit, mendeteksi kekambuhan penyakit yang diderita pasien, dan memilih obat dan menyesuaikan terapi. Pada zaman teknologi yang semakin berkembang ini, untuk memudahkan pemantauan kesehatan telah banyak dibangun sistem-sistem pemantauan kesehatan. Jika dikelompokkan menurut penggunaan komponen *hardware*, maka sistem pemantauan kesehatan dapat di kategorikan menjadi 3 yaitu

- (a) *Sensor-based health monitoring systems*,
- (b) *Smartphone-based health monitoring systems*,
- (c) *Microcontroller-based health monitoring systems*.

Diantara ketiga kategori diatas *microcontroller* adalah komponen yang paling sering digunakan untuk membangun sistem pemantauan kesehatan. MCU's (Microcontroller Unit) sangat baik digunakan untuk pemrosesan data mentah dari sensor selain itu karena ukurannya yang kecil tingkat portability-nya tinggi. Dalam membangun sistem pemantauan kesehatan ini *arduino*, *raspberry*, dan *ZigBee* banyak digunakan. Terdapat beberapa parameter yang biasa di pantau oleh sistem pemantauan kesehatan seperti detak jantung, saturasi oksigen, tekanan darah, suhu/temperatur tubuh manusia. Pada tiap parameter tersebut mempunyai batas normal yang berbeda, batas normal tersebut digunakan sebagai parameter yang menandakan keadaan seseorang apakah berada dibatas normal atau melebihi batas normal. Jika seseorang berada diatas batas normal maka hal itu menandakan seseorang sedang menderita sakit dan perlu untuk berkonsultasi dengan dokter atau tim medis. Pada tabel 1 dapat dilihat bahwa tiap parameter mempunyai batasan yang berbeda, batasan tersebut juga dapat berbeda bergantung pada beberapa aspek misalnya usia, penyakit yang diderita, dan gender. Batasan pada tabel 1 merupakan batasan normal untuk orang dewasa dengan usia ≥ 20 tahun yang tidak menderita penyakit apapun.

¹<https://medlineplus.gov/genetics/understanding/consult/prognosis/>

No	Parameter Pemantauan	Batas Normal
1	Suhu / temperatur tubuh	36°c - 37°c
2	Detak jantung	60 - 100 bpm
3	Saturasi oksigen	<95%
4	Tekanan darah	120 /80 mm Hg

Tabel 1: Batas Normal Parameter Pengukuran

3. Melakukan riset ketersediaan sensor-sensor yang dibutuhkan.

Status : Ada sejak rencana kerja skripsi.

Hasil : Sensor yang digunakan untuk membuat sistem pemantauan kesehatan merupakan sensor berjenis biosensor. Terdapat berbagai parameter kesehatan yang dapat dipantau oleh biosensor yaitu detak jantung, level saturasi oksigen, peredaran darah pada tubuh, tingkat pernafasan, kerja otot pada tubuh, banyaknya oksigen yang di hirup, dan pola pergerakan manusia. Setelah melakukan berbagai pertimbangan, pada penelitian ini akan dilakukan pemantauan kesehatan khususnya untuk kesehatan jantung dan paru-paru. Untuk melakukan pemantauan tersebut ada 3 buah sensor yang akan digunakan pada penelitian ini yaitu

- Pulse sensor, digunakan untuk melakukan pengukuran terhadap denyut jantung.



Gambar 3: Pulse Sensor

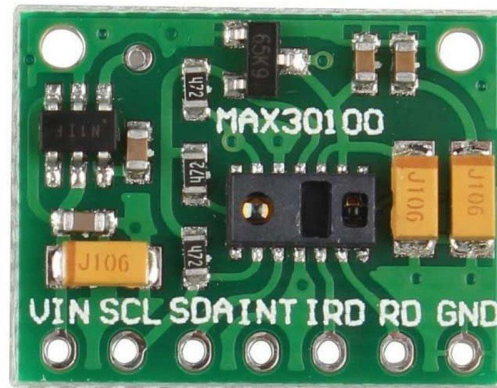
Berdasarkan gambar 3, pulse sensor mempunyai 3 buah pin yang harus disambungkan pada *board* arduino. Sensor ini dapat bekerja dengan daya 3V atau 5V. Berikut pemetaan pin sensor terhadap arduino.

Tabel 2: Pemetaan Pin Pulse Sensor

Pin Pulse Sensor	Arduino Pin
-	Gnd pin
+	5V
s	arduino A0

Cara kerja pulse sensor : Pada pulse sensor terdapat sebuah infra-red LED yang akan digunakan untuk mengukur denyut jantung. Saat seseorang menaruh tangannya pada bagian depan dari pulse sensor, refleksi dari lampu infra-red akan bergantung pada volume darah pada pembuluh darah kapilar yang terdapat pada jari. Saat jantung memompakan darah maka volume darah pada pembuluh darah kapilar akan tinggi, dan begitu juga sebaliknya saat jantung sedang tidak memompakan darah maka volume darah pada pembuluh darah kapilar akan berkurang. Variasi dari refleksi infra-red LED ini yang digunakan untuk mengukur denyut jantung seseorang.

- SPO2 Sensor, digunakan untuk melakukan pengukuran detak jantung dan saturasi oksigen dalam darah. Modul dari sensor ini adalah MAX30100.



Gambar 4: SP02 Sensor

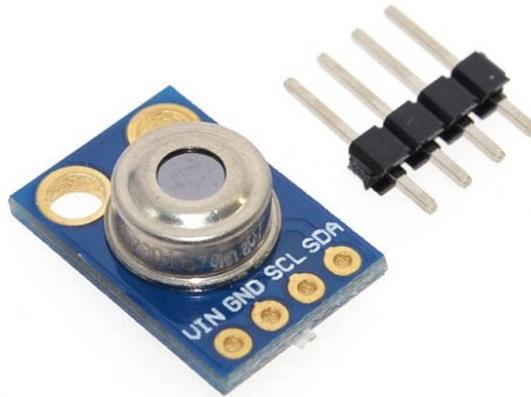
Berdasarkan gambar 4, modul MAX30100 terdapat 5 buah pin yang harus disambungkan pada *board* arduino. Sensor ini dapat bekerja dengan daya 1.8V dan 3.3V. Berikut pemetaan pin modul dengan *board* arduino

Tabel 3: Pemetaan Pin Modul MAX30100

Pin Pulse Sensor	Arduino Pin
VIN	+3.3V
GND	GND
SCL	arduino A5
SDA	arduino A4
INT	-

Cara Kerja modul MAX30100 : Cara kerja dari modul sensor ini mirip dengan cara kerja pulse sensor, yang membedakan kedua modul sensor ini adalah pada modul MAX30100 dapat melakukan pengukuran saturasi oksigen pada darah. Saturasi oksigen adalah jumlah persentase hemoglobin pada pembuluh darah yang membawa oksigen. Sama seperti cara kerja modul pulse sensor, modul MAX30100 juga dilengkapi dengan infra-red LED dan menggunakan refleksi dari lampu infra-red LED untuk mengukur detak jantung dan saturasi oksigen berdasarkan variasi refleksi yang terjadi.

- Sensor Temperatur, digunakan untuk mengukur suhu tubuh manusia. Modul dari sensor ini adalah MLX90614.



Gambar 5: Sensor Temperatur

Berdasarkan gambar 5, terdapat 4 buah pin yang harus disambungkan pada *board* arduino. Range dari temperatur yang dapat diukur modul ini adalah -20 hingga 120°C. Berikut pemetaan pin modul dengan *board* arduino

Tabel 4: Pemetaan Pin Modul MLX90614

Pin Pulse Sensor	Arduino Pin
VCC	5V
GND	GND
SCL	arduino A5
SDA	arduino A4

Cara Kerja modul MLX90614 : Modul MLX90614 sensor dapat mengukur temperatur tanpa perlu bersentuhan langsung dengan objek atau permukaan suatu objek. Meski tidak terlihat oleh mata manusia, semua objek memancarkan sinar infra-red dan konsentrasi dari sinyal tersebut berbeda tergantung dengan temperaturnya. Dengan mendeteksi sinyal infra-red maka dapat mengetahui kisaran suhu sebuah objek.

Dengan menggunakan ketiga sensor yang telah disebutkan diatas, dapat dilakukan pemantauan terhadap denyut jantung, saturasi oksigen dan temperatur tubuh seseorang. Dari ketiga parameter tersebut dapat menjadi indikator yang menunjukkan seseorang sehat atau tidak. Perubahan denyut jantung dan saturasi oksigen seseorang dipengaruhi oleh tingkat pernafasan, jika seseorang mengalami gangguan pernafasan maka akan menyebabkan saturasi oksigen dalam darah seseorang menjadi rendah. Rendahnya saturasi oksigen akan menyebabkan denyut jantung lebih cepat sehingga akan mengalami kelelahan, gangguan penglihatan dan pusing. Selain kedua parameter tersebut, pengecekan suhu tubuh juga merupakan salah satu indikator kesehatan seseorang. Maka dari itu pengecekan akan ketiga parameter ini menjadi penting.

4. Melakukan analisis masalah dan analisis kebutuhan perangkat lunak untuk membangun perangkat lunak pemantau kesehatan.

Status : Ada sejak rencana kerja skripsi.

Hasil : Pada penelitian ini penulis telah melakukan beberapa analisis terkait sistem yang akan digunakan oleh perangkat lunak pemantauan kesehatan. Sistem ini akan bekerja pada arduino untuk

melakukan *sensing* atau pengukuran dan juga pada komputer pengguna untuk menampilkan hasil *sensing* tersebut. Untuk menghemat waktu pemantauan, sistem pemantuan kesehatan akan dapat melakukan pengukuran terhadap beberapa orang sekaligus pada satu waktu yang sama. Sistem ini akan bekerja dengan 2 tahap yaitu

- pada tahap pertama, Orang yang ingin melakukan pemantauan kesehatan harus dipasangkan atau meletakkan semua sensor pada bagian tubuh yang telah ditentukan. Setelah semua dipasangkan akan dilakukan pengecekan apakah sensor telah siap digunakan untuk melakukan pengukuran dan dilakukan sinkronisasi waktu terhadap semua sensor yang akan digunakan. Setelah memastikan bahwa sensor sudah siap digunakan, maka admin akan memberikan perintah *sense* pada *basestation*.
- Pada tahap kedua, sensor node yang telah melakukan pengukuran akan mengirimkan hasil pengukurannya ke *base-station* lalu sistem akan menampilkan semua hasil pengukuran yang telah dilakukan pada komputer pengguna.

Pada penelitian ini akan dibangun 3 buah program, yaitu program pertama adalah program yang akan berjalan atau digunakan pada sensor node dan melakukan *sensing* terhadap pengguna. Program kedua adalah program yang akan berjalan atau digunakan pada *base-station* yang akan berguna untuk menerima hasil *sensing* yang telah dilakukan oleh setiap sensor dan mengirimkan hasil *sensing* kepada komputer. sementara Program ketiga adalah program yang akan berjalan pada komputer pengguna yang akan digunakan untuk interaksi antara sistem dan pengguna, seperti menampilkan hasil *sensing* tersebut kepada pengguna dan memberikan perintah *sense*. Pada program yang akan berjalan pada komputer terdapat 2 jenis pengguna yaitu admin dan user. kedua jenis pengguna mempunyai fitur yang berbeda. Ketiga program tersebut akan saling berkomunikasi agar dapat bekerja dengan baik. Program pada komputer akan menerima masukan dari pengguna dan menerima keluaran dari *base-station*. Berikut penjelasan singkat mengenai alur komunikasi pada perangkat lunak yang akan dibangun.

- Perangkat lunak pada komputer akan mengirimkan perintah ke *sink/base-station* untuk meminta sensor node melakukan proses *sensing*.
- *Base-station/sink* mengirimkan pesan ke sensor node untuk melakukan proses *sensing*.
- sensor node mengirimkan hasil *sensing* yang telah dilakukan kepada *base-station/sink*.
- *Base-station* mengembalikan hasil *sensing* ke antarmuka untuk dapat ditampilkan kepada pengguna.

Untuk membangun sebuah WSN dibutuhkan arsitektur dan topologi, sehingga pada penelitian ini akan digunakan arsitektur flat dan menggunakan topologi star. Alasan dipilihnya arsitektur flat adalah karena tiap sensor node memiliki tugas yang sama yaitu untuk melakukan *sensing* sehingga tidak dibutuhkan hirarki selain itu jaringan yang akan dibangun adalah jaringan kecil sehingga *clustering* juga tidak dibutuhkan.



Gambar 6: Wsn dengan topologi *star* dan protokol komunikasi *single-hop*

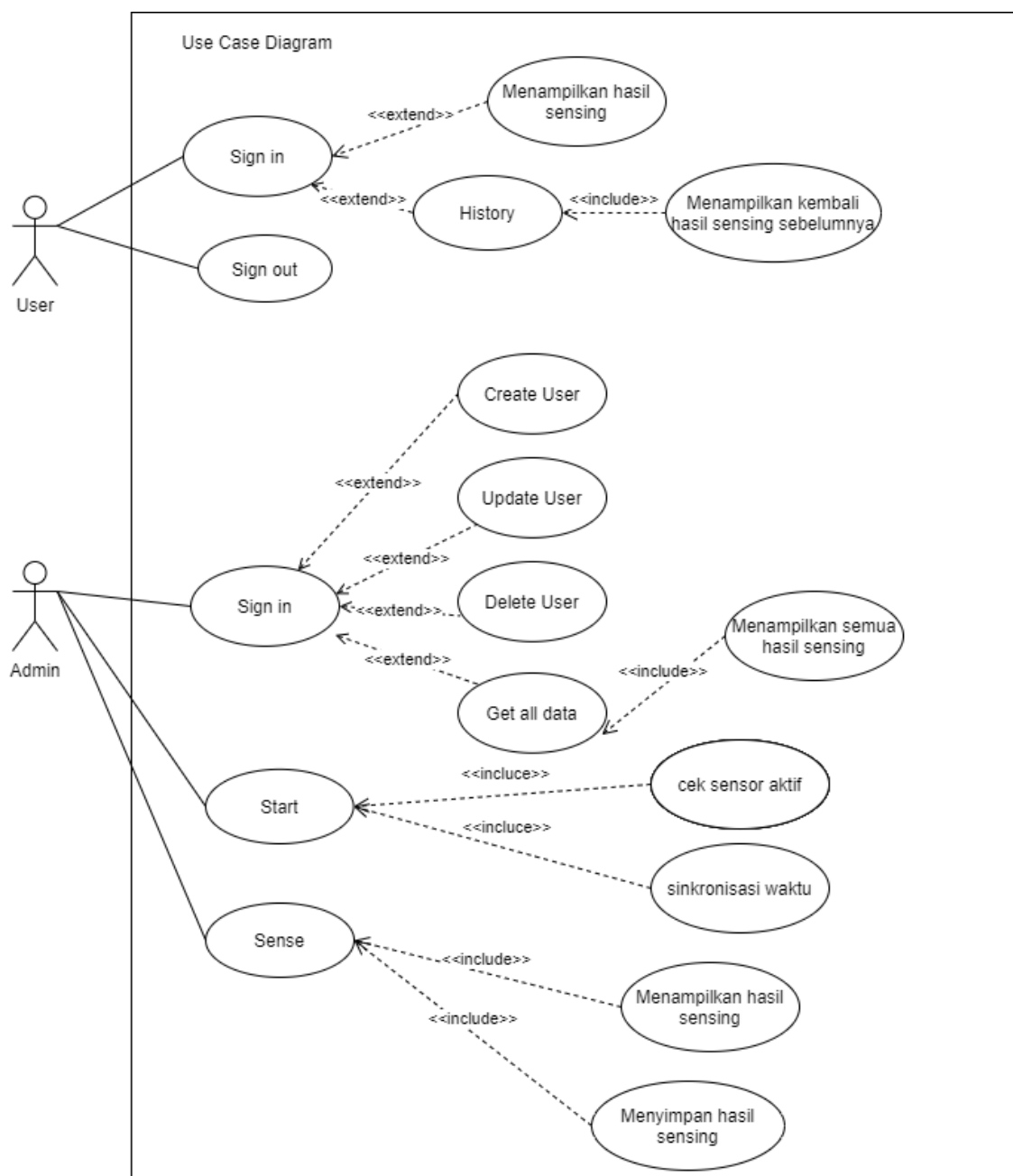
Seperti yang dapat dilihat pada 6, Pada topologi *star* protokol komunikasi yang digunakan adalah *single-hop*. Setiap sensor node (ditandai dengan warna putih) yang telah melakukan *sensing* akan mengirimkan data langsung menuju *base-station/sink* (ditandai dengan warna biru) setelah itu data hasil *sensing* akan dikirimkan oleh *base-station/sink* kepada komputer untuk dapat ditampilkan kepada user. Tujuan menggunakan komunikasi dan topologi tersebut adalah untuk memudahkan proses konfigurasi hubungan antara *base-station* dengan sensor node dan pertimbangan jarak antar sensor node yang masih dapat dijangkau dengan menggunakan topologi ini. Pengguna pada sistem yang dibangun terdiri dari 2 jenis yaitu user dan admin. Berikut fungsi-fungsi yang dapat dilakukan oleh user pada perangkat lunak.

- (a) Melakukan Sign in
- (b) Menampilkan hasil sense yang telah dilakukan sensor node
- (c) Menampilkan history hasil sensing pengguna

Penjelasan fungsi user : Fungsi melakukan sign in digunakan agar user dapat mengakses perangkat lunak. User harus melakukan pendaftaran kepada admin terlebih dahulu sehingga dapat mengakses perangkat lunak untuk melihat data hasil pemantauan dan history pemantauan yang telah dilakukan user. Fungsi menampilkan hasil *sensing* yang telah dilakukan sensor dan memberikan peringatan terhadap variabel pemantauan yang melebihi ambang batas normal, pada fungsi ini perangkat lunak akan menampilkan hasil sense yang telah dikirimkan oleh sensor node. Sistem akan melakukan peringatan terhadap variabel pemantauan yang melebihi ambang batas normal dengan melakukan highlight terhadap hasil pengukuran tersebut. Selain menampilkan hasil *sensing* sistem juga akan menyimpan hasil pengukuran pengguna juga akan disimpan pada database sehingga dapat ditampilkan kembali kepada pengguna. Fungsi menampilkan history hasil sensing pengguna digunakan untuk menampilkan data sensing pengguna yang sebelumnya telah dilakukan oleh pengguna. Sistem akan melakukan pencarian data pengguna pada database untuk ditampilkan kembali. Admin akan memiliki fungsi / fitur yang berbeda dari user, Berikut fungsi / fitur yang dapat dilakukan oleh admin.

- Melakukan penambahan user
- Melakukan update data user
- Melakukan penghapusan user
- Melakukan get all data user
- Memeriksa apakah sensor siap melakukan sensing / pengukuran
- Memberikan perintah sense kepada sensor node

Penjelasan fungsi admin : Fungsi melakukan penambahan user digunakan untuk admin dapat melakukan penambahan user, setelah admin menambahkan user maka user dapat mengakses perangkat lunak untuk dapat melihat hasil pemantauan kesehatannya. Fungsi melakukan update data user digunakan untuk mengganti data user jika terdapat data user yang telah berubah dan harus diganti dengan data baru. Fungsi penghapusan user digunakan untuk melakukan penghapusan user. Fungsi get all data user digunakan untuk menampilkan semua data hasil pemantauan yang telah dilakukan oleh user. Fungsi memeriksa apakah sensor siap melakukan sensing / pengukuran digunakan untuk memastikan sensor node pada WSN sedang aktif dan dapat melakukan sensing. Selain memastikan sensor aktif, juga akan dilakukan sinkronisasi waktu untuk semua sensor. Jika dideteksi sensor node terdapat error pengguna tidak dapat melakukan sensing. Fungsi memberikan perintah sense kepada sensor node digunakan untuk memulai sensing pada setiap sensor, hasil sensor akan langsung dikirimkan ke sink / *base-station*. Untuk memudahkan penggambaran interaksi yang dapat dilakukan oleh user dan admin maka dibuat use-case diagram dan skenario untuk tiap use-case.



Gambar 7: Use-Case Diagram untuk pengguna(user)

Setiap fungsi pada diagram use-case dilengkapi dengan skenario untuk memodelkan interaksi antara pengguna dan sistem. Berikut skenario dari setiap fungsi pada use-case.

Tabel 5: Tabel Skenario Saat User Menekan Tombol Sign in

Nama	Sign in
Deskripsi	Pengguna menekan tombol sign in untuk melakukan login pada sistem
Aktor	User
Pre-Kondisi	Pengguna telah melakukan pendaftaran pada admin
Alur Skenario	<ul style="list-style-type: none"> (a) User telah melakukan pendaftaran pada admin dan mendapatkan akun. (b) Pengguna melakukan klik pada tombol / button sign in. (c) Pengguna mengisi form login. (d) Sistem memeriksa apakah data yang user masukkan sesuai dengan data-base. (e) jika sesuai, user masuk ke halaman home. (f) jika tidak sesuai, sistem akan mengeluarkan pesan error.

Tabel 6: Tabel Skenario Saat User Menekan Tombol History

Nama	History
Deskripsi	Pengguna menekan tombol history, perangkat lunak menampilkan hasil pengukuran yang telah dilakukan user sebelumnya
Aktor	User
Pre-Kondisi	Pengguna telah berhasil melakukan login
Alur Skenario	<ul style="list-style-type: none"> (a) Pengguna melakukan klik pada tombol / button history. (b) Perangkat lunak mendeteksi apakah terdapat data pengukuran sebelumnya. (c) Perangkat lunak menampilkan hasil pengukuran yang telah dilakukan user sebelumnya, jika tidak terdapat data maka akan muncul pesan tidak terdapat data pengukuran sebelumnya.

Tabel 7: Tabel Skenario Saat Admin Menekan Tombol Create User

Nama	Create User
Deskripsi	Admin menekan tombol create user
Aktor	Admin
Pre-Kondisi	pengguna telah berhasil melakukan login
Alur Skenario	<ul style="list-style-type: none"> (a) Pengguna membuka tab create user. (b) Pengguna mengisi data user pada form pendaftaran. (c) Pengguna menekan tombol create user, sistem melakukan pengecekan apakah user sudah pernah didaftarkan pada sistem. (d) Jika belum pernah terdaftar maka pendaftaran user berhasil, jika sudah pernah terdaftar maka muncul pesan error.

Tabel 8: Tabel Skenario Saat Admin Menekan Tombol Update User

Nama	Update User
Deskripsi	Pengguna menekan tombol update user
Aktor	Admin
Pre-Kondisi	Pengguna telah berhasil melakukan login
Alur Skenario	(a) Pengguna mencari user untuk melakukan update data. (b) Pengguna mengganti data user. (c) Pengguna menekan tombol update.

Tabel 9: Tabel Skenario Saat Admin Menekan Tombol Delete User

Nama	Delete User
Deskripsi	Pengguna menekan tombol delete user
Aktor	Admin
Pre-Kondisi	Pengguna telah berhasil melakukan login
Alur Skenario	(a) Pengguna mencari user yang akan dihapus. (b) Pengguna menekan tombol / button delete user.

Tabel 10: Tabel Skenario Saat Admin Menekan Tombol Get All Data

Nama	Get All Data
Deskripsi	Pengguna menekan tombol get all data
Aktor	Admin
Pre-Kondisi	Pengguna telah berhasil melakukan login
Alur Skenario	(a) Pengguna melakukan klik pada tombol / button get all data (b) Perangkat menampilkan semua data pengukuran yang telah dilakukan user.

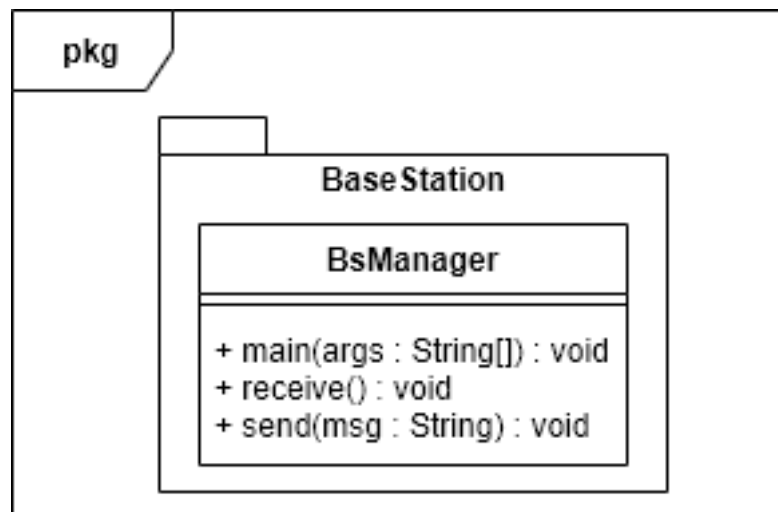
Tabel 11: Tabel Skenario Saat Admin Menekan Tombol Start

Nama	Start
Deskripsi	Pengguna memberikan perintah start kepada <i>basestation</i> untuk melakukan pengecekan dan sinkronisasi
Aktor	Admin
Pre-Kondisi	Sensor telah dipasang
Alur Skenario	(a) Pengguna memberikan perintah start kepada <i>basestation</i> . (b) Sistem mendeteksi apakah semua sensor telah siap melakukan pengukuran dan melakukan sinkronisasi waktu. (c) Sistem mengeluarkan pesan siap melakukan sensing atau pesan error jika sensor belum siap untuk melakukan sensing.

Tabel 12: Tabel Skenario Saat Admin Menekan Tombol Sense

Nama	Sense
Deskripsi	Pengguna memberikan perintah sense, Semua sensor melakukan sensing dan hasil sensing dikirimkan dan akan ditampilkan pada perangkat lunak, hasil sensing tersebut juga akan disimpan.
Aktor	Admin
Pre-Kondisi	Tidak ada pesan error saat perintah start dijalankan
Alur Skenario	<p>(a) Semua sensor telah siap melakukan pengukuran/sensing dan telah dilakukan sinkronisasi waktu.</p> <p>(b) Sensor melakukan pengukuran/sensing.</p> <p>(c) Sensor mengirimkan hasil pengukuran/sensing kepada perangkat lunak.</p> <p>(d) Perangkat lunak menampilkan hasil pengukuran dan menyimpan hasil pengukuran.</p>

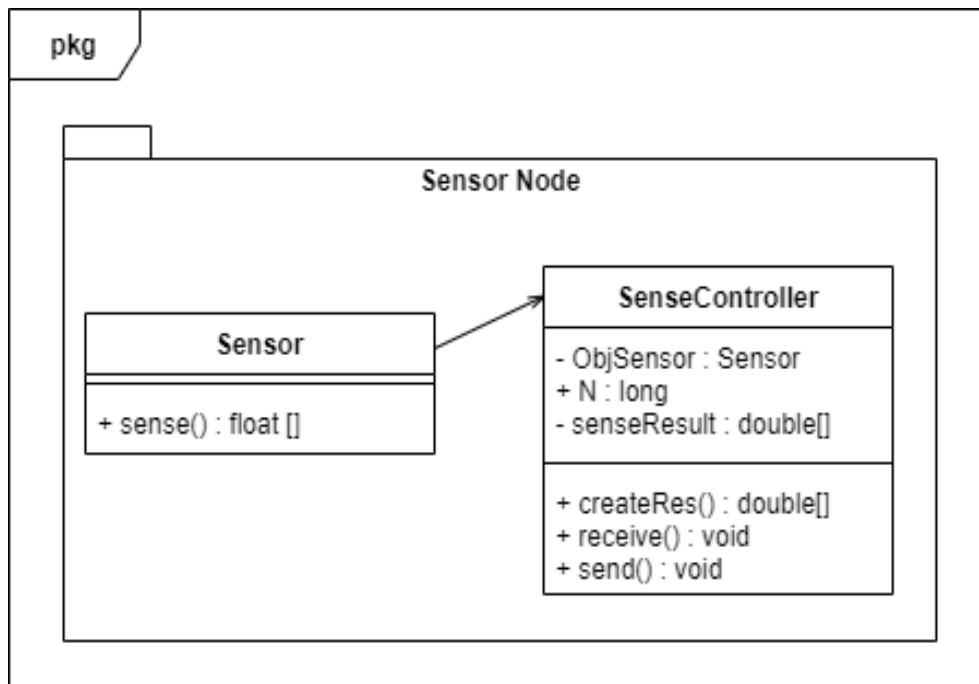
Pembuatan perangkat lunak akan menggunakan Arduino IDE untuk *sensor node Arduino Mega 256*. Setelah melakukan analisis maka dibuat diagram kelas sederhana untuk aplikasi ini. Berikut diagram kelas sederhana yang dibutuhkan untuk aplikasi ini.



Gambar 8: Kelas Diagram Sederhana Pada Aplikasi di BaseStation

Keterangan Package BaseStation :

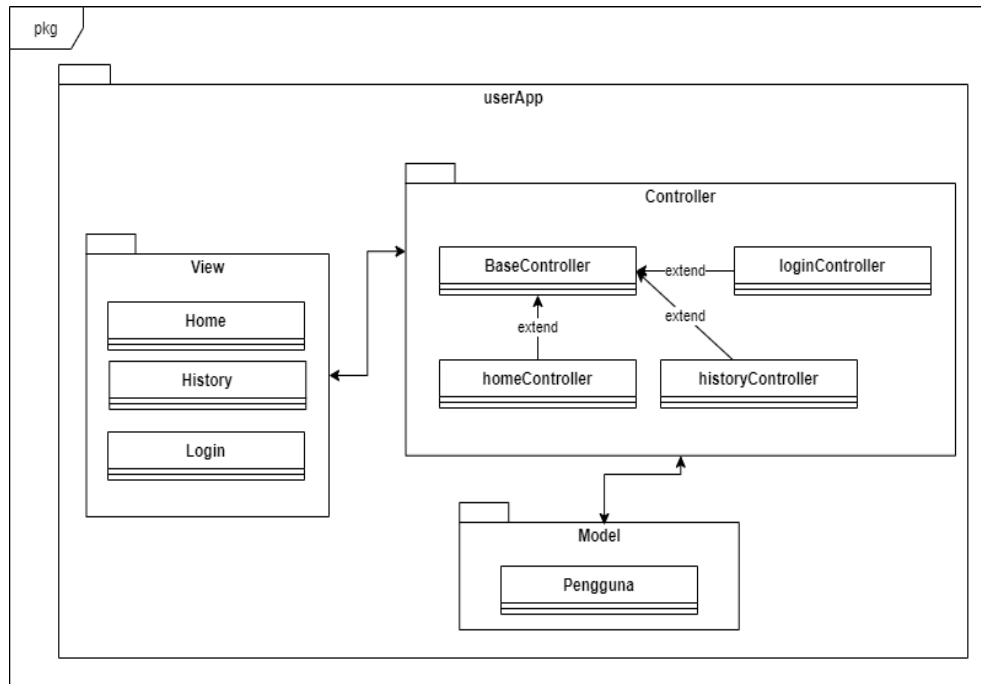
- Kelas BSManager : Kelas ini digunakan sebagai main class dari package ini. Kelas ini akan berfungsi untuk menerima pesan dari sensor node, menerima perintah dari UserApp, dan akan mengirim perintah tersebut ke sensor node. Metode yang bertugas untuk mengirimkan perintah dalam bentuk pesan kepada sensor node adalah metode *send()*. Metode yang bertugas untuk menerima pesan yang dikirimkan oleh sensor node adalah metode *receive()*.



Gambar 9: Kelas Diagram Sederhana Pada Aplikasi di SensorNode

Keterangan package SensorNode :

- Kelas Sensor : Kelas ini digunakan untuk menginisialisasi sensor pada sensor node. Metode *sense()* digunakan untuk melakukan *sensing* atau pengukuran.
- Kelas SensorController : Kelas ini menjadi main class dari package ini dan digunakan untuk menerima dan mengirim pesan / hasil *sensing* atau pengukuran yang telah dilakukan ke *base-station*. Atribut *ObjSensor* digunakan sebagai objek dari sebuah sensor, *N* digunakan sebagai panjang dari hasil yang akan ditampilkan, *senseResult* digunakan untuk menyimpan hasil *createRes()* digunakan untuk menyimpan semua hasil *sensing* atau pengukuran yang akan dikirimkan ke *base-station*. Metode *send()* digunakan untuk mengirim pesan ke *baseStation*. dan Metode *receive()* untuk menerima pesan dari *baseStation*.



Gambar 10: Kelas Diagram Sederhana Pada Aplikasi di UserApp

Keterangan Package UserApp : Pada Package ini terdapat 3 folder yaitu View, Controller, dan Model. View digunakan untuk mengatur tampilan yang akan dilihat oleh user, Controller digunakan sebagai penghubung antara View dan Model, Sementara Model digunakan untuk mengatur hal-hal terkait *logic* pada App. Pada folder View terdapat 3 kelas yaitu :

- Kelas Home : Kelas ini digunakan untuk mengatur tampilan home user. Pada halaman ini akan ditampilkan hasil *sensing* atau pengukuran yang dilakukan user.
- Kelas History : Kelas ini digunakan untuk mengatur tampilan history user. Pada halaman ini akan ditampilkan riwayat hasil *sensing* atau pengukuran yang dilakukan user.
- Kelas Login : Kelas ini digunakan untuk mengatur tampilan sign-up dan sign-in user. Pada halaman ini user akan diminta mengisi form pendaftaran atau login.

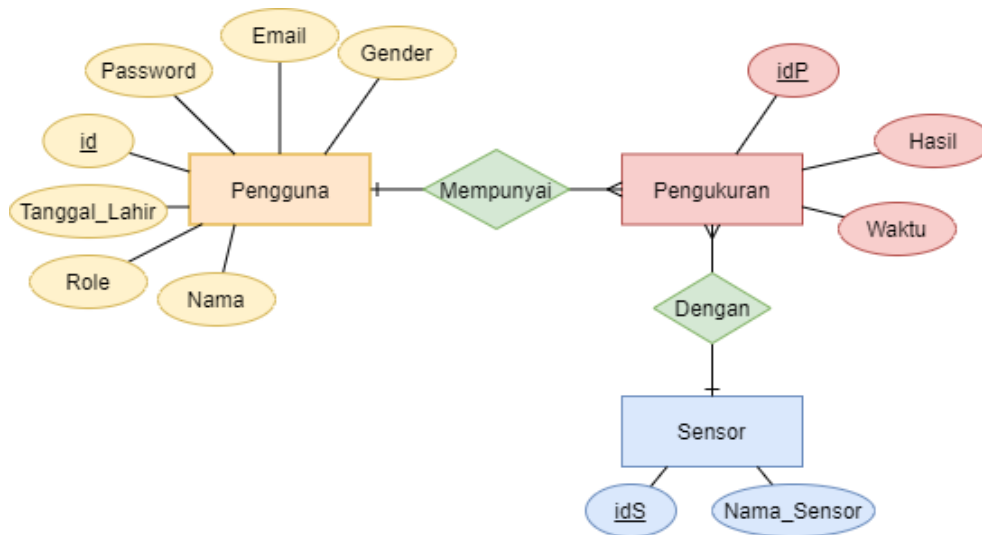
Pada folder Controller terdapat 3 kelas yaitu :

- Kelas BaseController : Kelas ini digunakan sebagai controller utama pada folder ini.
- Kelas homeController : Kelas ini digunakan sebagai controller dari view Home.
- Kelas historyController : Kelas ini digunakan sebagai controller dari view History.
- Kelas loginController : Kelas ini digunakan sebagai controller dari view Login.

Pada folder Model terdapat 1 kelas yaitu :

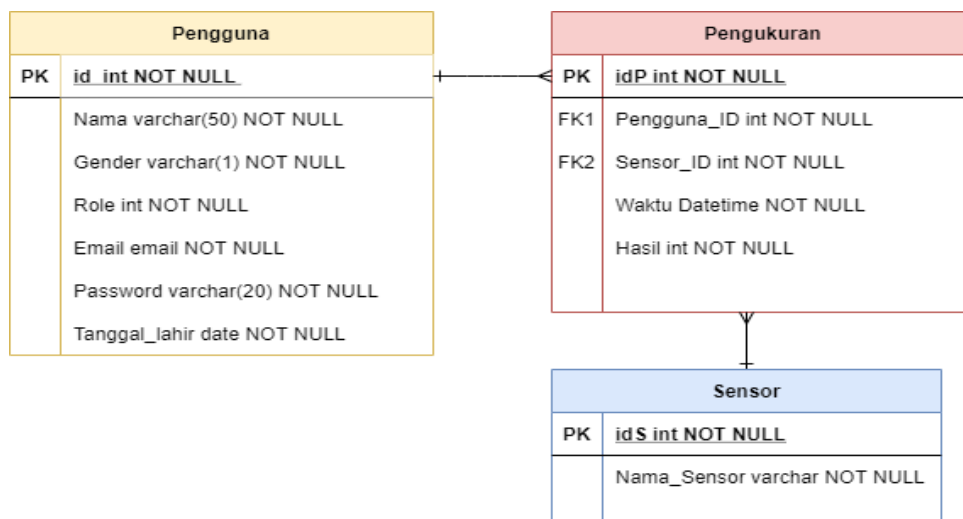
- Kelas Pengguna : Kelas ini digunakan untuk mengatur pengambilan data user dari database.

Untuk dapat menampilkan dan memproses data hasil pemantauan kesehatan pengguna, perangkat lunak membutuhkan database sebagai tempat untuk menyimpan data hasil pemantauan tersebut. Berikut rancangan ERD(*Entity Relationship Diagram*) dan rancangan tabel yang akan digunakan dalam membangun database.



Gambar 11: Rancangan ER-Diagram

ER-Diagram pada 11 akan mempunyai 3 entitas yaitu entitas Pengguna, entitas Pengukuran dan entitas Sensor. Tiap entitas mempunyai beberapa atribut dan Tiap entitas juga mempunyai hubungan dengan entitas lain. Hubungan antara entitas Pengguna dan Pengukuran adalah One to Many, yang berarti satu pengguna dapat mempunyai 2 atau lebih Pengukuran. Sementara hubungan entitas Pengukuran dan entitas sensor adalah Many to One, yang berarti satu sensor dapat digunakan untuk banyak pengukuran. Entitas pada ER-Diagram akan dibuat menjadi tabel-tabel pada database. Rancangan tabel dari ER-Diagram dapat dilihat pada gambar dibawah.



Gambar 12: Rancangan tabel dari ER-Diagram

Entitas dan atribut pada ER-Diagram akan dirancang menjadi tabel-tabel pada database, Pada gambar 12 dapat dilihat bahwa tiap atribut mempunyai tipe data yang berbeda-beda. Berikut akan dijelaskan rancangan tabel dan atribut dari tiap tabel.

- (a) Rancangan tabel Pengguna, akan digunakan untuk menyimpan data pengguna. Tabel ini mempunyai beberapa atribut yaitu :
- id, yang digunakan sebagai Primary-key adalah nilai unik dimana tiap pengguna akan mempunyai sebuah nilai ID yang berbeda satu sama lain. Atribut akan menggunakan tipe data

int.

- Gender, yang digunakan untuk menunjukkan apakah pengguna merupakan pria atau wanita. Atribut akan menggunakan tipe data varchar(1). Yang akan berisi hanya satu huruf, huruf L untuk pria dan P untuk perempuan.
 - Role, yang digunakan untuk menunjukkan apakah pengguna merupakan user atau admin selain itu juga akan digunakan untuk penandaan bahwa data telah dihapus. Atribut akan menggunakan tipe data int, dimana 0 menunjukkan pengguna adalah admin, 1 menunjukkan pengguna adalah user, dan 2 untuk menunjukkan bahwa user telah dihapus.
 - Email, yang digunakan untuk menyimpan email pengguna, tiap pengguna harus memasukkan satu alamat email. Atribut akan menggunakan tipe data email.
 - Password, yang digunakan untuk menyimpan password yang telah dibuat oleh pengguna saat melakukan pendaftaran. Atribut akan menggunakan tipe data varchar.
 - Tanggal_lahir, yang digunakan untuk menyimpan data tanggal lahir dari seorang pengguna. Atribut akan menggunakan tipe data date.
- (b) Rancangan tabel Pengukuran, akan digunakan untuk menyimpan data hasil pemantauan yang telah dilakukan pengguna sehingga dapat ditampilkan. Tabel ini mempunyai beberapa atribut yaitu :
- IdP, yang digunakan sebagai Primary-Key adalah nilai unik dimana tiap baris memiliki nilai yang berbeda. Atribut ini menggunakan tipe data int.
 - Pengguna_ID, yang digunakan sebagai foreign-key yang menunjuk pada id pengguna. Tipe data dari atribut ini adalah int.
 - Sensor_ID, yang digunakan foreign-key yang menunjuk pada id sensor. Tipe data dari atribut ini adalah int.
 - Waktu, yang digunakan untuk menyimpan tanggal dan waktu dilakukannya pengukuran. Nilai dari atribut ini akan berguna untuk menunjukkan riwayat pengukuran pengguna. Atribut ini akan menggunakan tipe data datetime.
 - Hasil, yang digunakan untuk menyimpan riwayat hasil pemantauan pengguna. Tipe data yang digunakan adalah int.
- (c) Rancangan tabel Sensor, akan digunakan untuk menyimpan data sensor-sensor yang digunakan untuk melakukan pengukuran. Tabel ini mempunyai beberapa atribut yaitu :
- idS, yang digunakan sebagai Primary-Key adalah nilai unik dimana tiap baris memiliki nilai yang berbeda. Atribut ini menggunakan tipe data int.
 - Nama_Sensor, yang digunakan untuk menyimpan nama sensor yang digunakan untuk melakukan pengukuran. Atribut ini menggunakan tipe data varchar.

Alur kerja sistem yang akan dibangun adalah sebagai berikut, User akan melakukan pendaftaran pada admin lalu Sensor akan dipasangkan pada bagian tubuh user telah ditentukan, lalu Admin akan memberikan perintah seperti start dan sense, Perintah-perintah tersebut akan dikirimkan kepada *base-station*. Setelah perintah tersebut diterima oleh *base-station*, format pesan akan diubah sehingga dapat dibaca oleh sensor node. Dan pesan tersebut diteruskan kepada sensor node dengan cara *broadcast*. Jika pesan/perintah yang diberikan oleh admin adalah Start maka setelah sensor node menerima pesan, sensor node akan mengirimkan status(apakah sedang menyala atau tidak) dari tiap sensor node kepada *base-station* dan melakukan sinkronisasi waktu terhadap semua sensor. Jika pesan/perintah yang diberikan oleh admin adalah sense maka setelah sensor node menerima pesan, sensor node akan melakukan sensing dan mengirimkan hasil sensing tersebut Kembali kepada *base-station*. Dan jika user melakukan login dengan akun yang telah didaftarkan oleh admin, sistem akan menampilkan hasil

pemantauan yang dilakukan user. Jika user memberikan pesan/perintah yang diberikan adalah history maka sistem akan melakukan pengecekan apakah pengguna mempunyai data pemantauan sebelumnya, dan melakukan *retrieve* data dan menampilkan data tersebut kepada user.

Untuk menjamin kelancaran komunikasi antara sensor node, *base-station*, dan perangkat lunak pada komputer maka dibutuhkan sebuah format pesan yang dapat digunakan untuk menerima perintah atau pesan. Komunikasi akan terjadi saat dilakukan pengiriman pesan atau perintah dan penerimaan pesan. Berikut format pesan yang akan digunakan untuk setiap komunikasi.

- Komunikasi antara perangkat lunak pada komputer dan *base-station* serta komunikasi antara *base-station* dan sensor node akan dilakukan dengan format pesan sebagai berikut : (**{kodePengenal};{NamaPerintah}**).
- Komunikasi antara sensor node dan *base-station* serta komunikasi antara *base-station* dan perangkat lunak pada komputer akan dilakukan dengan format pesan sebagai berikut : (**{kodePengenal};{sensorNodeID};{namaSensor};{hasil}**).

Keterangan :

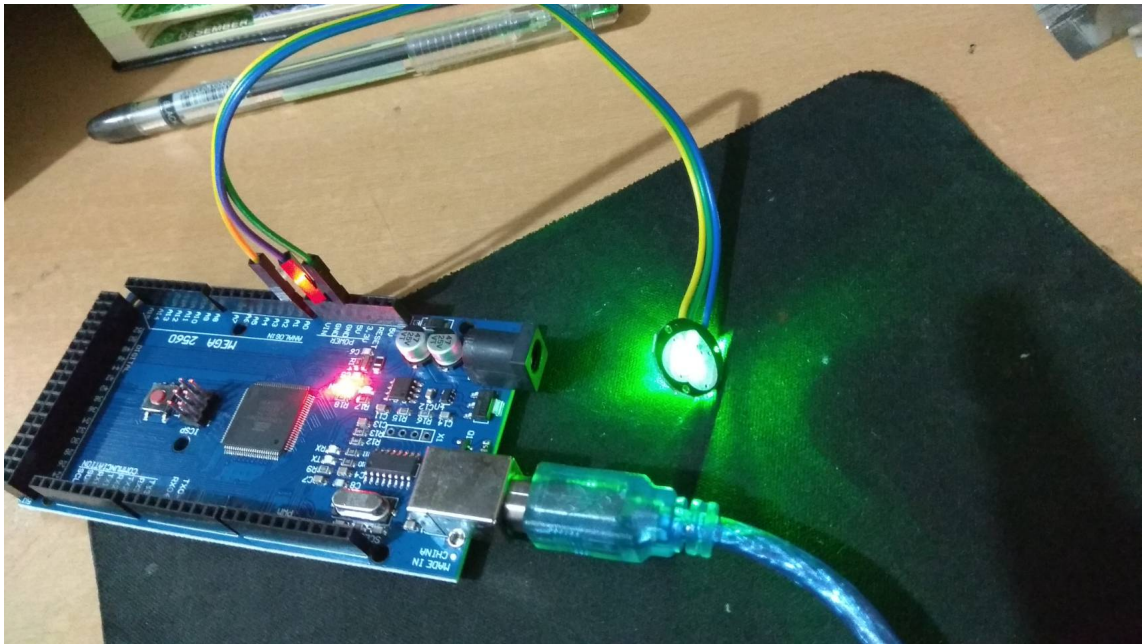
- kodePengenal pada format pesan akan berisi Q(*Query*) yang menunjukkan pemberian perintah / pesan sementara R(*Result*) yang menunjukkan hasil dari perintah / penerimaan pesan.
- NamaPerintah pada format pesan akan berisi start, sense dan history.
- SensorNodeID pada format pesan akan berisi id dari sensor node yang mengirimkan balasan. Id tersebut akan bertipe int.
- namaSensor pada format pesan akan berisi nama dari sensor yang melakukan *sensing*.
- hasil pada format pesan akan berisi nilai hasil *sensing* yang telah dilakukan.

Setelah melakukan analisis terhadap *base-station*, dimana tugas utama dari *base-station* adalah menerima pesan dari seluruh sensor node yang terhubung dan meneruskan pesan baik kepada perangkat lunak maupun meneruskan pesan kepada setiap sensor node. Tetapi tiap sensor node yang terhubung pada *base-station* akan melakukan *sensing* pada waktu yang sama sehingga tidak menutup kemungkinan bahwa tiap sensor node juga akan mengirimkan hasil *sensing* pada waktu yang bersamaan, hal ini tentu dapat menimbulkan collision. Bagaimana *base-station* tetap dapat menerima hasil *sensing* dari setiap sensor node sehingga dapat ditampilkan pada perangkat lunak? Langkah yang digunakan untuk mengatasi hal tersebut adalah dengan mengatur pengiriman yang dilakukan oleh sensor node. Tiap sensor node akan melakukan pengiriman ulang hasil *sensing* hingga diterima oleh *base-station*. Dengan cara tersebut maka *base-station* dapat menerima hasil *sensing* dari tiap sensor node yang terhubung.

5. Melakukan studi perakitan terhadap *arduino* dan sensor.

Status : Ada sejak rencana kerja skripsi.

Hasil : Untuk melakukan studi perakitan terhadap *arduino* dan sensor, penulis membeli satu buah *arduino mega* dan satu buah pulse sensor. Setelah melakukan studi perakitan, penulis mencoba melakukan perakitan sederhana seperti yang dapat dilihat pada gambar 13. Setelah berhasil menghubungkan *arduino* serta pulse sensor pada laptop, Penulis mencoba menuliskan kode program sederhana untuk mencoba apakah pulse sensor dapat bekerja. Pulse sensor digunakan untuk mengukur detak jantung pengguna, sensor ini digunakan dengan cara meletakkan jari tangan pengguna pada bagian depan sensor yang memiliki bentuk hati. Setelah melakukan percobaan dengan menaruh jari penulis pada sensor, dapat diketahui bahwa sensor dapat bekerja dengan baik. Seperti dapat dilihat pada gambar 14, saat jari diletakkan pada bagian depan sensor akan dilakukan pengukuran detak jantung dan akan hasil pengukuran tersebut akan dikirimkan pada laptop sehingga akan menampilkan hasil seperti pada gambar 15 dapat dilihat oleh pengguna.



Gambar 13: Perakitan Arduino dan Pulse Sensor



```
#define USE_ARDUINO_INTERRUPTS true
#include <PulseSensorPlayground.h>
const int PulseWire = 0;
const int LED13 = 13;
int Threshold = 550;

PulseSensorPlayground pulseSensor;

void setup() {

  Serial.begin(9600);

  pulseSensor.analogInput(PulseWire);
  pulseSensor.blinkOnPulse(LED13);
  pulseSensor.setThreshold(Threshold);

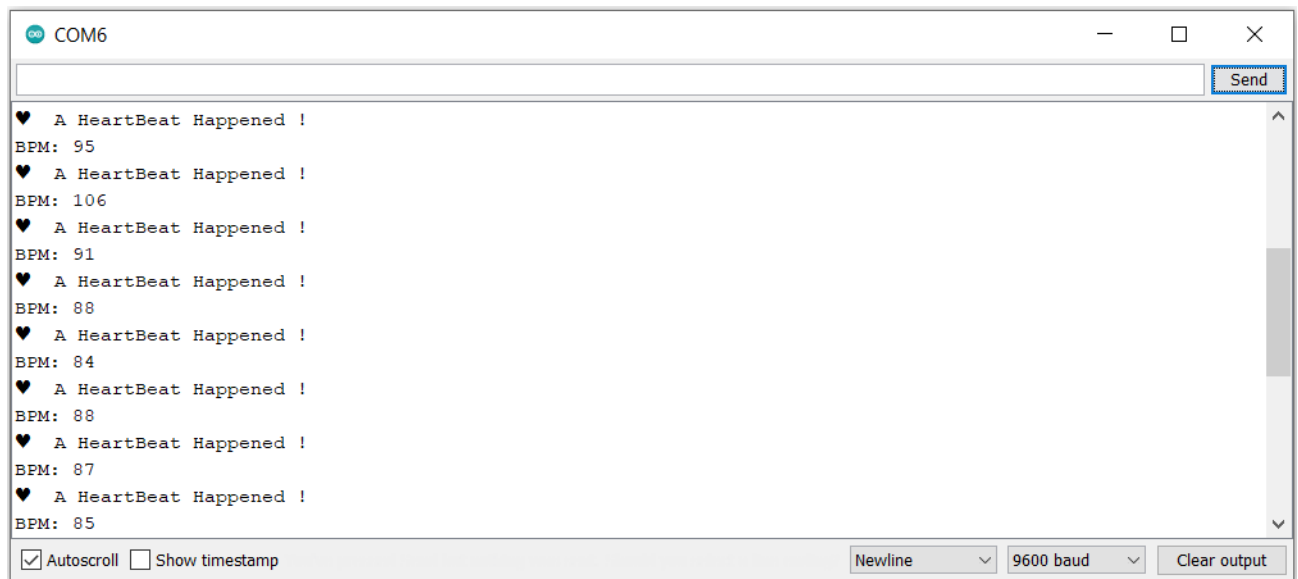
  if (pulseSensor.begin()) {
    Serial.println("We created a pulseSensor Object !");
  }
}

void loop() {

  int myBPM = pulseSensor.getBeatsPerMinute();

  if (pulseSensor.sawStartOfBeat()) {
    Serial.println("♥ A HeartBeat Happened ! ");
    Serial.print("BPM: ");
    Serial.println(myBPM);
  }
  delay(20);
}
```

Gambar 14: Kode Program



Gambar 15: Hasil Output

6. Menulis dokumen skripsi.

Status : Ada sejak rencana kerja skripsi.

Hasil : Menulis dokumen skripsi dilakukan bersamaan dengan melakukan studi literatur dan analisis. Sampai dengan dokumen ini dibuat, Penulis telah berhasil menyelesaikan 3 bab dari dokumen skripsi yaitu bab 1 latar belakang, bab 2 landasan teori dan bab 3 analisis.

6 Pencapaian Rencana Kerja

Langkah-langkah kerja yang berhasil diselesaikan dalam Skripsi 1 ini adalah sebagai berikut:

1. Melakukan studi literatur mengenai WSN, arduino dan sensor.
2. Melakukan studi literatur mengenai kesehatan.
3. Melakukan riset ketersediaan sensor-sensor yang dibutuhkan.
4. Melakukan analisis masalah dan analisis kebutuhan perangkat lunak untuk membangun perangkat lunak pemantau kesehatan
5. Melakukan studi perakitan terhadap arduino dan sensor.
6. Menulis dokumen skripsi

7 Kendala yang Dihadapi

Kendala - kendala yang dihadapi selama mengerjakan skripsi :

- Terdapat banyak tugas dari matakuliah lain yang harus dikerjakan.

Bandung, 16/01/2021

Cristine Artanty

Menyetujui,

Nama: Elisati Hulu, S.T., M.T.
Pembimbing Tunggal